CIÊNCIA DA COMPUTAÇÃO

JOGOS DIGITAIS

SISTEMAS DE INFORMAÇÃO



SIRC

SIMPÓSIO DE INFORMÁTICA

Universidade Franciscana

15 a 16 set | 2026

Interconectando

códigos dados inteligência inovação

ORGANIZADORES
Graziela Frainer Knoll
Herysson Rodrigues Figueiredo

editora UFN

UFN Universidade Franciscana

CIÊNCIA DA COMPUTAÇÃO

JOGOS DIGITAIS

SISTEMAS DE INFORMAÇÃO

SIRC

SIMPÓSIO DE INFORMÁTICA

Universidade Franciscana

15 a 16 set | 2026

Interconectando

códigos dados inteligência inovação

ORGANIZADORES

Graziela Frainer Knoll

Herysson Rodrigues Figueiredo

editora UFN

UFN Universidade Franciscana

Universidade Franciscana

Santa Maria, RS - 2026

editora.ufn.edu.br

EDITORA UFN
Rua Silva Jardim, 1535
Prédio 7, Sala 201A
Centro | Santa Maria, RS
97010-491 | (55) 3220.1306

COORDENAÇÃO EDITORIAL
Fagner Millani
Lucio Pozzobon de Moraes

CAPA
Lucio Pozzobon de Moraes

DIAGRAMAÇÃO
Lucio Pozzobon de Moraes

S619 SIRC - Simpósio de Informática - Universidade Franciscana - 15 a 16 set 2026 : interconectando códigos, dados, inteligência e Inovação [Ebook]
/ Organizadores Graziela Frainer Knoll, Herysson Rodrigues Figueiredo - Santa Maria, RS : Universidade Franciscana - UFN, 2026.
183 p. : il.

ISBN: 978-65-5852-595-0

1. Informática - simpósio 2. Ciência da Computação I. Knoll, Graziela Frainer II. Figueiredo, Herysson Rodrigues

CDU 004

Elaborada pela Bibliotecária Eunice de Olivera CRB 10/1491

DOI: XXXX

O conteúdo desta obra fora formulado exclusivamente pelo(s) autor(es). A Editora UFN não se responsabiliza pela manutenção, atualização e idioma dos sites, programas, materiais complementares ou similares referidos pelos autores nesta obra.

Marcas Registradas: Todos os termos mencionados e reconhecidos como Marca Registrada e/ou Comercial são de responsabilidade de seus proprietários. A editora informa não estar associada a nenhum produto e/ou fornecedor apresentado no livro.

SUMÁRIO

1	PREFÁCIO
3	SEGURANÇA DIGITAL: ARGUS, ASSISTENTE PARA APOIO À IDENTIFICAÇÃO DE GOLPES
19	DESENVOLVIMENTO E VALIDAÇÃO FUNCIONAL DE UM FLUXO AUTOMATIZADO PARA MONITORAMENTO DE PUBLICAÇÕES CIENTÍFICAS COM N8N E OPENALEX
33	PIPELINE DE CONVERSÃO DE ARQUIVOS DE SIMULAÇÃO MOLECULAR PARA VISUALIZAÇÃO EM REALIDADE MISTA
47	DA NARRATIVA AO JOGO: O PROCESSO DE DESENVOLVIMENTO DE *RESCUE HORSE*
61	DESENVOLVIMENTO DE UM SISTEMA WEB PARA GESTÃO DO ACERVO BIBLIOGRÁFICO DO MUSEU TREZE DE MAIO
77	AUTOMATING SCIENTOMETRICS: A GENERATIVE AI PIPELINE FOR ANALYZING NANOSCIENCE RESEARCH TRENDS
91	SISTEMA PARA CONTROLE DE VOO DE DRONE COM RESTRIÇÃO ESPACIAL BASEADA EM GEOLOCALIZAÇÃO
107	SISTEMA WEB PARA CATALOGAÇÃO E GESTÃO DO ACERVO DO MUSEU TREZE DE MAIO
123	AUTOMATED GUIDED VEHICLE PROTOTYPE FOR INDUSTRY 4.0 LOGISTICS
139	DESENVOLVIMENTO DE UM SERIOUS GAME PARA EDUCAÇÃO EM SAÚDE SOBRE O USO DOS SERVIÇOS DE URGÊNCIA E EMERGÊNCIA
155	SISTEMA DE GERENCIAMENTO DE CONTEÚDO PARA O PORTAL DA REDE NACIONAL DE MULHERES NA NANOCIÊNCIA
171	VULNERABILIDADES E SEGURANÇA NO CÓDIGO GERADO POR INTELIGÊNCIA ARTIFICIAL

PREFÁCIO

Apresentamos o ebook dos Anais do **XVII Simpósio de Informática (SIRC)**, evento organizado pelos cursos de Ciência da Computação, Sistemas de Informação e Jogos Digitais da Universidade Franciscana. Sob o tema *"Interconectando dados, códigos, inteligência e inovação"*, esta edição consolida-se como um espaço privilegiado para a convergência de saberes, reunindo pesquisas que refletem as fronteiras do conhecimento em Computação e suas profundas transformações no ecossistema tecnológico e social contemporâneo.

Estes Anais materializam a missão fundamental do SIRC: promover a integração da comunidade científica, o intercâmbio de experiências e o debate propositivo sobre as tendências emergentes da Ciência da Computação. O leitor encontrará nas páginas a seguir uma seleção de artigos originais, frutos do empenho de estudantes, pesquisadores e profissionais comprometidos com a excelência acadêmica e a inovação tecnológica.

As contribuições reunidas nesta obra cobrem um espectro abrangente de investigações teórico-práticas, fomentando a atualização contínua e estimulando novas colaborações interdisciplinares. Cada trabalho aqui publicado não apenas registra o estado da arte das tecnologias de informação, mas também lança pistas fundamentais para os desafios futuros da computação aplicada.

Expressamos nossa gratidão aos autores, pareceristas e à comissão organizadora, cuja dedicação garantiu a qualidade desta publicação. Esperamos que este volume sirva de fonte permanente de consulta, inspiração e impulso para novos projetos de pesquisa e desenvolvimento.

Boa leitura!

Profª. Graziela Frainer Knoll
Prof. Herysson Rodrigues Figueiredo
Organizadores

SEGURANÇA DIGITAL: ARGUS, ASSISTENTE PARA APOIO À IDENTIFICAÇÃO DE GOLPES

DIGITAL SECURITY: ARGUS, ASSISTANT FOR SUPPORT IN IDENTIFYING SCAMS

Bernardo Paz de Freitas[1] e Herysson Rodrigues Figueiredo[2]

RESUMO

Este trabalho aborda o avanço da digitalização e a consequente proliferação de fraudes eletrônicas no Brasil, investigando como um assistente de segurança, baseado em múltiplas camadas de verificação, pode apoiar usuários na identificação de conteúdo potencialmente fraudulento. O objetivo geral é apresentar o desenvolvimento do protótipo do sistema Argus, um assistente integrado ao WhatsApp, e investigar a percepção do público acerca dos golpes digitais e sua receptividade à proposta de solução. Como objetivos específicos, buscou-se mapear a exposição dos participantes a conteúdos suspeitos, avaliar a intenção de uso da ferramenta, modelar o fluxo de processamento e implementar um protótipo funcional para análise de diferentes tipos de mídia. A proposta fundamenta-se no princípio de defesa em profundidade e no padrão Roteador Baseado em Conteúdo. Metodologicamente, caracteriza-se como pesquisa aplicada e tecnológica, de caráter exploratório e descritivo, desenvolvida durante uma imersão de inovação. Foram realizados dois levantamentos exploratórios de opinião pública, com 94 e 45 participantes, e desenvolvido um Produto Mínimo Viável utilizando Python, FastAPI, Tesseract OCR, BERT e Whisper. Os resultados indicaram elevada exposição percebida a conteúdos suspeitos, especialmente no WhatsApp, além de receptividade favorável à solução proposta. O desenvolvimento resultou em um protótipo capaz de processar links, mensagens textuais, áudios e boletos, demonstrando a viabilidade técnica inicial da abordagem, embora sua eficácia na identificação de fraudes ainda demande avaliação experimental específica.

Palavras-chave: Fraude eletrônica, Processamento de linguagem natural, Proteção ao usuário

1 Acadêmico do curso Sistemas de Informação na Universidade Franciscana (UFN). E-mail: 2004be@gmail.com. ORCID: https://orcid.org/0009-0004-9674-6866.
2 Professor do curso de Sistemas de Informação na Universidade Franciscana (UFN). E-mail: herysson.figueiredo@ufn.edu.br. ORCID: https://orcid.org/0009-0009-2615-9785

ABSTRACT

This study addresses the advance of digitalization and the consequent proliferation of electronic fraud in Brazil, investigating how a conversational security assistant based on multiple verification layers can support users in identifying potentially fraudulent content. The general objective is to present the development of the Argus system prototype, an assistant integrated with WhatsApp, and to investigate public perception of digital scams and receptiveness to the proposed solution. The specific objectives were to map participants' exposure to suspicious content, assess their intention to use the tool, model the processing flow, and implement a functional prototype for analyzing different types of media. The proposal is based on the defense-in-depth principle and the Content-Based Router pattern. Methodologically, the study is characterized as applied and technological research with an exploratory and descriptive approach, developed during an innovation immersion event. Two exploratory public opinion surveys were conducted with 94 and 45 participants, and a Minimum Viable Product was developed using Python, FastAPI, Tesseract OCR, BERT, and Whisper. The results indicated a high perceived exposure to suspicious content, especially on WhatsApp, as well as favorable receptiveness to the proposed solution. The development resulted in a prototype capable of processing links, text messages, audio messages, and payment slips, demonstrating the initial technical feasibility of the approach, although its effectiveness in identifying fraud still requires specific experimental evaluation.

Keywords: *Electronic fraud, Natural language processing, User protection*

1 INTRODUÇÃO

A digitalização e a hiperconectividade transformaram a maneira como a sociedade se comunica. Aplicativos de mensagens, como o WhatsApp, deixaram de desempenhar exclusivamente uma função de comunicação pessoal e passaram a integrar atividades comerciais e a troca cotidiana de informações. Esse processo pode ser relacionado à penetrabilidade do paradigma tecnológico, como a informação é a base de todas as ações humanas, as redes digitais passam a fazer parte de todo o cotidiano social, privado e comercial (Castells, 1999).

Paralelamente à expansão do uso das tecnologias digitais, observa-se o crescimento dos golpes e fraudes praticados por meios eletrônicos. Em 2025, foram registrados 2.261.055 casos de estelionato no Brasil, dos quais 346.753 foram classificados como estelionato por meio eletrônico (Fórum Brasileiro de Segurança Pública, 2026, p. 120). Entre as modalidades de fraude mais recorrentes, destacam-se os golpes realizados por WhatsApp, mencionados por 34% dos entrevistados, e aqueles envolvendo Pix, com 24%, sendo a engenharia social um dos principais mecanismos empregados para induzir as vítimas a

fornecer informações ou realizar transações (Fórum Brasileiro de Segurança Pública, 2026, p. 126). A dimensão do problema também é evidenciada por pesquisa de vitimização segundo a qual 15,8% da população brasileira com 16 anos ou mais foi vítima de golpe ou perdeu dinheiro pela internet ou pelo celular no período de um ano, o equivalente a aproximadamente 26,3 milhões de pessoas (Fórum Brasileiro de Segurança Pública, 2026, p. 123).

Salahdine e Kaabouch (2019) classificam a engenharia social como um dos maiores desafios da segurança de redes por explorar a tendência humana natural de confiar. Dentre essas táticas, o *phishing* é apontado como o ataque mais comum, focado em manipular as vítimas para a extração fraudulenta de dados confidenciais. A dimensão do problema é alarmante, visto que os autores destacam que 84% dos ataques cibernéticos globais são baseados em engenharia social e apresentam altas taxas de sucesso (Salahdine; Kaabouch, 2019).

Nesse contexto, soluções computacionais podem contribuir para apoiar os usuários na identificação de conteúdo potencialmente fraudulento. Uma estratégia empregada em sistemas de segurança consiste na adoção da defesa em profundidade (*Defense in Depth*), abordagem baseada na utilização de múltiplas camadas de proteção, de modo que uma ameaça não identificada por determinado mecanismo possa ser submetida a verificações adicionais (NIST, 2017).

O presente trabalho tem como objetivo apresentar o desenvolvimento do sistema Argus e os resultados de um levantamento exploratório de opinião pública realizado para identificar a percepção do público em relação aos golpes digitais e sua receptividade à solução proposta. O Argus atua como um motor de segurança, orquestrando um fluxo que permite receber e analisar diferentes tipos de entradas, como textos, áudios, imagens de boletos e links, para detectar possíveis fraudes e golpes. A solução centraliza a segurança em uma interface já dominada pelo público, dispensando a instalação de novos aplicativos.

2 METODOLOGIA

Este trabalho caracteriza-se como uma pesquisa de natureza aplicada e tecnológica, de caráter exploratório e descritivo, orientada ao desenvolvimento de uma solução computacional para apoio à identificação de possíveis golpes digitais. O estudo compreendeu dois levantamentos exploratórios destinados à compreensão do problema e à avaliação da receptividade à proposta de solução, seguida do desenvolvimento de um Produto Mínimo Viável (MVP) do sistema Argus.

O projeto foi desenvolvido durante o Techstars Startup Weekend (TSW), evento de inovação voltado à concepção e ao desenvolvimento de propostas de negócio em um período de 52 horas. Durante o evento, as ideias apresentadas pelos participantes foram

submetidas a um processo de seleção e formação de equipes. A proposta que originou o Argus foi selecionada para desenvolvimento e, a partir disso, as atividades foram organizadas em duas etapas principais: realização de levantamentos exploratórios com potenciais usuários e construção do MVP.

2.1 LEVANTAMENTOS EXPLORATÓRIOS COM POTENCIAIS USUÁRIOS

A etapa de investigação junto aos potenciais usuários foi realizada por meio de dois levantamentos exploratórios, conduzidos presencialmente em um centro comercial de grande circulação. Os levantamentos tiveram como finalidade identificar inicialmente a percepção do público acerca dos golpes digitais e, posteriormente, avaliar a receptividade à solução proposta e as funcionalidades consideradas relevantes pelos potenciais usuários.

O primeiro levantamento foi direcionado à investigação da percepção do público acerca dos golpes digitais e contou com 94 participantes. O instrumento buscou identificar aspectos relacionados ao uso de redes sociais e aplicativos de comunicação, ao contato dos participantes com mensagens, links ou anúncios considerados suspeitos e aos ambientes digitais nos quais essas situações foram percebidas.

Após a análise inicial dessas informações, foi realizado um segundo levantamento, direcionado à avaliação da receptividade à proposta de solução. Essa etapa contou com 45 participantes e teve como objetivo investigar a intenção de utilização de um assistente integrado ao WhatsApp para apoio à identificação de possíveis golpes, bem como identificar quais funcionalidades seriam consideradas mais relevantes pelos usuários.

A pesquisa de opinião pública foi realizada coletando as respostas de forma anônima, sem o registro de dados pessoais, sensíveis ou de identificação dos participantes. Os resultados foram analisados por meio de estatística descritiva, utilizando frequências absolutas e relativas para sintetizar as respostas obtidas.

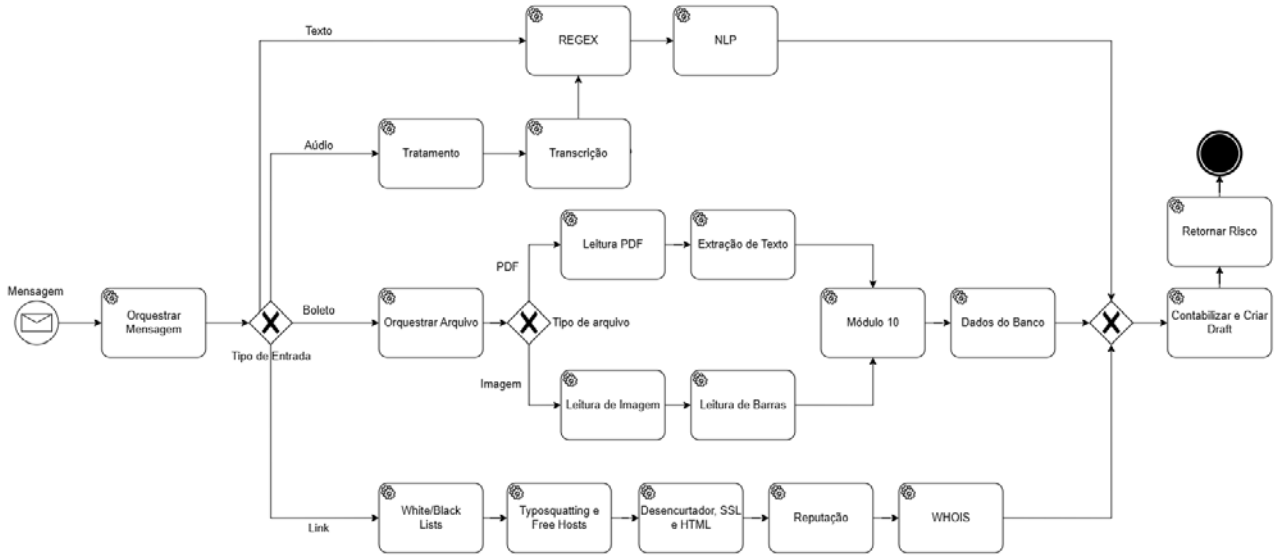
2.2 DESENVOLVIMENTO DO PROTÓTIPO

Com base nos requisitos identificados durante os levantamentos exploratórios, foi desenvolvido o MVP do Argus. A solução foi implementada em Python, utilizando o framework FastAPI (Ramírez, 2026) para a construção da Interface de Programação de Aplicativos (API - *Application Programming Interface*), que permite acesso às funcionalidades de um programa (Sommerville, 2011). Para a integração do sistema com o WhatsApp, foi utilizada a EvolutionAPI (Evolution Foundation, 2026).

A arquitetura central da aplicação foi inspirada no padrão Roteador Baseado em Conteúdo (*Content-Based Router*), em que, segundo Hohpe e Woolf (2003), o conteúdo de uma mensagem é analisado para determinar seu encaminhamento ao fluxo de processamento correspondente. No Argus, esse mecanismo é responsável por identificar o tipo de entrada recebida, link, texto, áudio, imagem ou documento e direcioná-la ao módulo de verificação apropriado.

A Figura 1 apresenta o fluxo de processamento do sistema por meio de um diagrama construído segundo a Notação e Modelo de Processos de Negócio (BPMN - *Business Process Model and Notation*) (OMG, 2013). O diagrama representa o percurso realizado desde o recebimento de uma mensagem pelo WhatsApp até a geração do retorno contendo o nível de risco identificado.

Figura 1 - Diagrama de processos do Argus



Fonte: elaborado pelos autores (2026).

Conforme representado na Figura 1, o módulo de verificação é composto por diferentes fluxos de processamento, definidos de acordo com o conteúdo recebido. Os principais mecanismos implementados são descritos a seguir.

As URLs recebidas são inicialmente submetidas à consulta de listas de domínios previamente classificados como confiáveis (*whitelist*) ou maliciosos (*blacklist*). Quando o domínio já consta em uma dessas listas, o sistema pode determinar o encaminhamento da análise sem executar todas as verificações subsequentes. Caso contrário, são realizadas verificações relacionadas a *typosquatting*, utilizado em domínios que exploram variações ortográficas de endereços legítimos, como utilizar ‘gogle.com’ no lugar de ‘google.com’, e à utilização de serviços gratuitos de hospedagem (*free hosts*).

Na sequência, o sistema analisa redirecionamentos de um Localizador Uniforme de Recursos (URL - *Uniform Resource Locator)* utilizando um desencurtador, uma ferramenta

que mostra o caminho real do link, analisa ainda a presença do certificado de segurança Camada de Soquetes Seguros (SSL - *Secure Sockets Layer*) e realiza a leitura de elementos do conteúdo da Linguagem de Marcação de Hipertexto (HTML - *HyperText Markup Language*) da página, em busca de palavras suspeitas. A reputação do endereço também é consultada por meio da API VirusTotal (2026). Por fim, informações relativas ao registro e à idade do domínio são verificadas por meio do serviço WHOIS, um protocolo público de consulta a dados de registro da internet (IETF, 2004). Os resultados obtidos nessas diferentes etapas são utilizados pelo sistema para determinar o nível de risco associado ao endereço analisado.

O processamento de boletos é iniciado pela identificação do formato do arquivo recebido. Caso a entrada corresponda a um documento em Formato de Documento Portátil (PDF - *Portable Document Format*), o arquivo é submetido à extração de texto, utilizando expressões regulares (Regex), uma sequência de caracteres que forma um padrão dinâmico de busca. Quando a entrada corresponde a uma imagem, são aplicadas técnicas de Reconhecimento Óptico de Caracteres (OCR - *Optical Character Recognition*), incluindo tratamento da imagem por escala de cinza, ajuste mediano e contraste, seguido da leitura dos caracteres impressos pelo mecanismo Tesseract (2026).

Além da leitura dos caracteres, o sistema utiliza a biblioteca ZBar (2026) para realizar a leitura do código de barras. Após a extração das informações, os dados são encaminhados ao módulo responsável pela verificação dos dígitos, empregando o algoritmo Módulo 10, definida pela FEBRABAN (2023) como um algoritmo de soma de verificação padronizado. Por fim, o código da instituição financeira identificado no documento é confrontado com uma lista local de instituições, complementando o processo de verificação.

As mensagens textuais são inicialmente analisadas por meio de expressões regulares, utilizadas para identificar termos e padrões associados a estratégias recorrentes de engenharia social, como mensagens que induzem à urgência financeira, falsas oportunidades de emprego ou ofertas de brindes. Quando são identificados links incorporados à mensagem, estes também são encaminhados ao módulo específico de análise de URLs.

Como camada complementar, o sistema emprega técnicas de Processamento de Linguagem Natural (NLP - *Natural Language Processing*), área que estuda as interações entre humanos e computadores por meio da linguagem natural (Zhou *et al.*, 2020). Para aplicar NLP, o sistema utiliza um modelo baseado na arquitetura de Representações de Codificador Bidirecional de Transformers (BERT - B*idirectional Encoder Representations from Transformers*) (Devlin *et al.*, 2018). O modelo classifica o conteúdo textual em uma escala de 1 a 5. Resultados situados nos extremos da escala são considerados pelo mecanismo de análise em conjunto com as demais verificações realizadas, contribuindo para a determinação do nível de risco da mensagem.

As mensagens de voz recebidas pelo WhatsApp são inicialmente submetidas a uma etapa de tratamento. Os arquivos enviados no formato utilizado pelo Whatsapp (.ogg/opus) são convertidos para o formato .wav utilizando o FFmpeg (2026). Em seguida, o conteúdo é transcrito utilizando o modelo de reconhecimento de fala Whisper (Radford *et al*., 2022).

Após a transcrição, o conteúdo textual obtido é encaminhado aos mesmos mecanismos utilizados na análise de mensagens escritas, incluindo as verificações baseadas em expressões regulares e NLP. Dessa forma, o sistema reutiliza o fluxo de análise textual para avaliar também conteúdos originalmente recebidos em formato de áudio.

Paralelamente aos mecanismos de análise, o Argus registra a recorrência dos conteúdos processados. A cada mensagem, arquivo ou link analisado, o sistema contabiliza a quantidade de ocorrências daquele conteúdo. Quando o limite definido na implementação, correspondente a dez ocorrências, é atingido, o sistema aciona a API do Google Gemini (2026) para gerar automaticamente um rascunho de alerta preventivo.

O conteúdo gerado não é publicado automaticamente. O rascunho é encaminhado ao painel administrativo do sistema web, no qual os administradores podem avaliar as informações e decidir pela publicação ou descarte do alerta. Dessa forma, a inteligência artificial é utilizada como mecanismo de apoio à produção do conteúdo, mantendo a decisão final de publicação sob responsabilidade dos administradores da plataforma.

3 RESULTADOS E DISCUSSÃO

Os resultados são apresentados considerando os dois levantamentos exploratórios descritos na metodologia e o desenvolvimento do MVP do Argus. Inicialmente, são analisados os dados obtidos no levantamento destinado à compreensão do problema e, posteriormente, os resultados relacionados à receptividade da solução proposta. Por fim, são apresentadas as principais funcionalidades implementadas no protótipo.

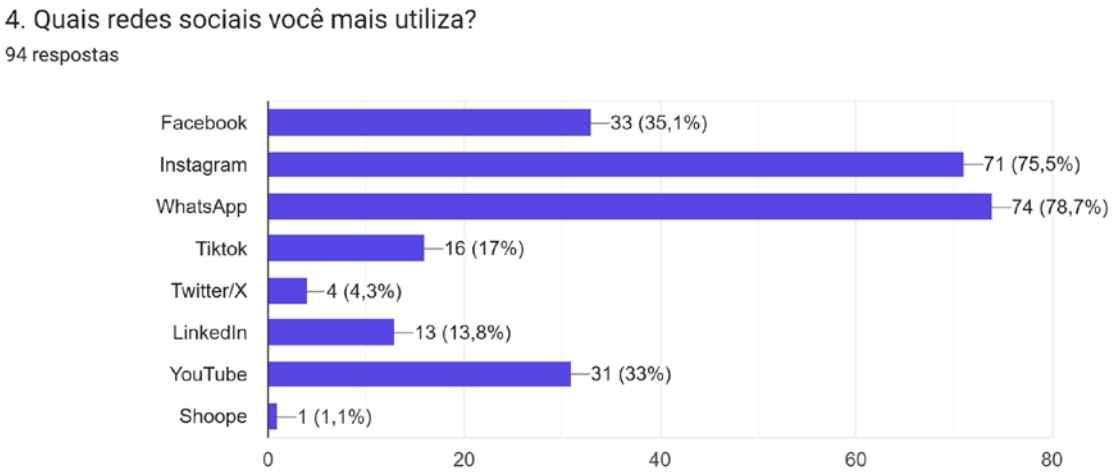
3.1 PERCEPÇÃO SOBRE GOLPES DIGITAL

O primeiro levantamento contou com 94 participantes e teve como objetivo identificar aspectos relacionados ao uso de plataformas digitais e à exposição dos participantes a conteúdos potencialmente suspeitos.

Entre as plataformas investigadas, o WhatsApp foi a plataforma mais frequentemente indicada pelos participantes, sendo selecionado por 78,7% dos respondentes, conforme apresentado na Figura 2. Esse resultado constituiu um dos elementos considerados na escolha da plataforma como principal meio de interação com o Argus, uma vez que a

utilização de um ambiente já presente no cotidiano dos usuários pode reduzir a necessidade de adaptação a uma nova aplicação.

Figura 2 - Plataforma digital mais utilizada pelos participantes.

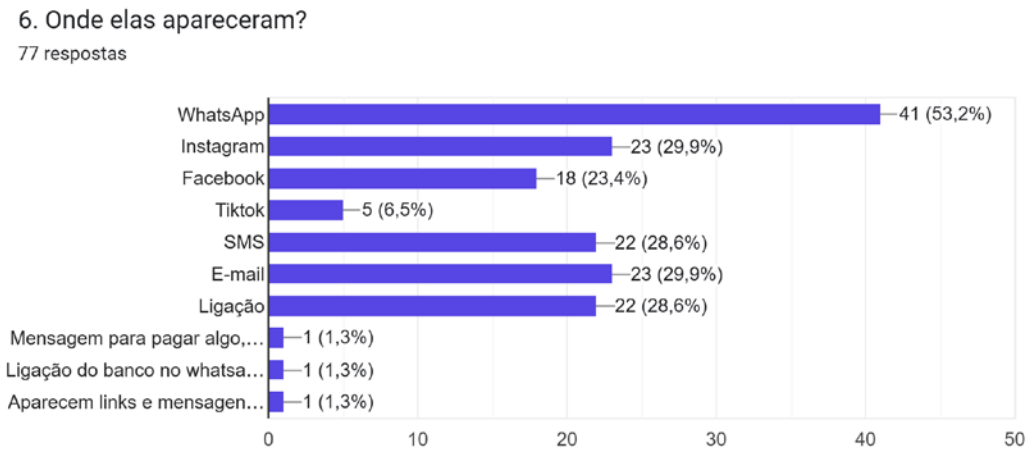


Fonte: dados da pesquisa (2026)

Além da ampla utilização das plataformas de comunicação, os resultados demonstraram uma recorrente exposição dos participantes a conteúdos considerados potencialmente suspeitos. Onde 79,8% dos respondentes afirmaram ter recebido mensagens, links ou anúncios duvidosos nos meses anteriores à realização do levantamento.

Embora esse resultado não permita determinar se os conteúdos recebidos correspondiam efetivamente a tentativas de fraude, ele evidencia a percepção de exposição dos participantes a situações de risco no ambiente digital. Esse aspecto é relevante para a proposta do Argus, cujo objetivo não é substituir mecanismos especializados de segurança, mas oferecer ao usuário um recurso adicional para análise de conteúdos sobre os quais exista dúvida.

Figura 3 - Ambiente de maior exposição a tentativas de golpes digitais.



Fonte: dados da pesquisa (2026)

Entre os 77 participantes que responderam à questão referente ao ambiente em que se percebiam mais expostos a tentativas de golpes, 53,2% indicaram o WhatsApp, conforme a Figura 3. Dessa forma, além de apresentar elevada utilização entre os participantes, a plataforma também foi apontada como o principal ambiente de exposição percebida a conteúdos potencialmente fraudulentos.

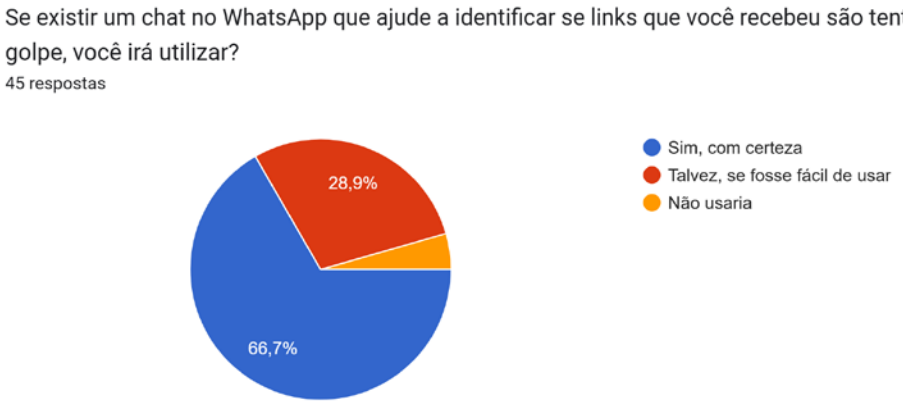
Os resultados do primeiro levantamento forneceram, portanto, indícios da existência do problema junto ao público consultado e contribuíram para direcionar a proposta de integração do Argus ao WhatsApp. Deve-se observar, entretanto, que os resultados representam a percepção dos participantes da amostra investigada e não permitem generalizações para a população em geral.

3.2 RECEPTIVIDADE À SOLUÇÃO PROPOSTA

Após o primeiro levantamento, foi realizado um segundo levantamento com 45 participantes, destinado a investigar a receptividade à proposta do Argus e identificar funcionalidades consideradas relevantes para uma ferramenta de apoio à identificação de possíveis golpes digitais.

Conforme apresentado na Figura 4, 66,7% dos participantes afirmaram que certamente utilizariam a solução apresentada, enquanto 28,9% indicaram que sua utilização dependeria da facilidade de uso da ferramenta.

Figura 4 - Intenção de utilização da solução proposta.



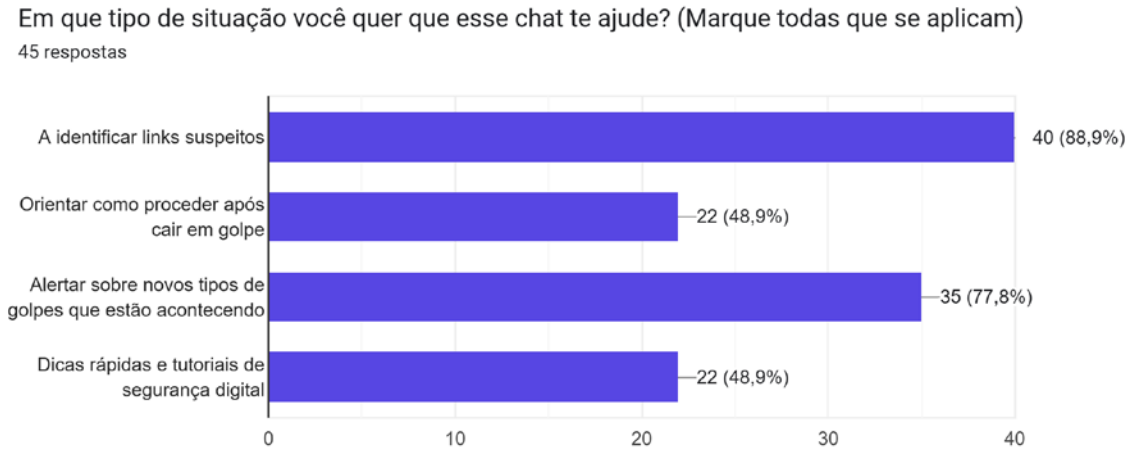
Fonte: dados da pesquisa (2026)

Os resultados indicam uma receptividade inicial favorável à proposta, mas também evidenciam a importância da usabilidade como requisito para adoção do sistema. Esse aspecto contribuiu para reforçar a decisão de utilizar o WhatsApp como principal interface de interação, buscando evitar que o usuário necessite aprender a utilizar uma nova aplicação apenas para consultar conteúdos suspeitos.

O levantamento também investigou quais funcionalidades eram consideradas mais relevantes pelos participantes. Conforme apresentado na Figura 5, a identificação de links suspeitos foi apontada por 88,9% dos respondentes, enquanto 77,8% indicaram interesse no recebimento de alertas preventivos sobre novos golpes.

Esses resultados foram utilizados como requisitos para orientar o desenvolvimento do MVP. A análise de links foi incorporada como um dos principais módulos de verificação do sistema, enquanto a demanda por alertas preventivos contribuiu para a implementação do mecanismo de identificação de conteúdo recorrentes e geração de rascunhos de alertas.

Figura 5 - Funcionalidades solicitadas pelo público-alvo.



Fonte: dados da pesquisa (2026)

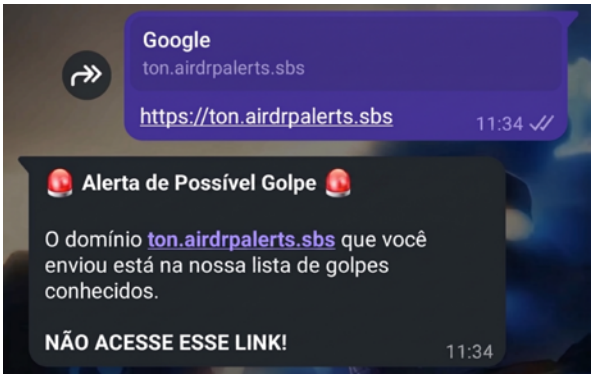
Dessa forma, observa-se uma relação entre os requisitos identificados no levantamento e as funcionalidades posteriormente incorporadas ao protótipo. Entretanto, os dados obtidos representam uma avaliação inicial da aceitação da proposta e não constituem uma validação da eficácia dos mecanismos utilizados pelo Argus para identificar fraudes.

3.3 RESULTADOS DO DESENVOLVIMENTO DO MVP

Como resultado do processo de desenvolvimento descrito na Seção 2.3, foi obtido um MVP funcional do Argus, integrado ao WhatsApp e capaz de processar diferentes tipos de conteúdo. O protótipo implementado contempla módulos para análise de links, mensagens textuais, boletos e áudios, além de funcionalidades destinadas à geração de alertas e à administração do serviço.

Na análise de links, o usuário pode encaminhar uma URL diretamente ao assistente. O conteúdo é submetido às diferentes verificações descritas na metodologia e, ao final do processamento, o sistema retorna uma mensagem contendo a classificação de risco identificada. A Figura 6 apresenta um exemplo da interação realizada por meio do WhatsApp.

Figura 6 - Exemplo de análise de link realizada pelo Argus



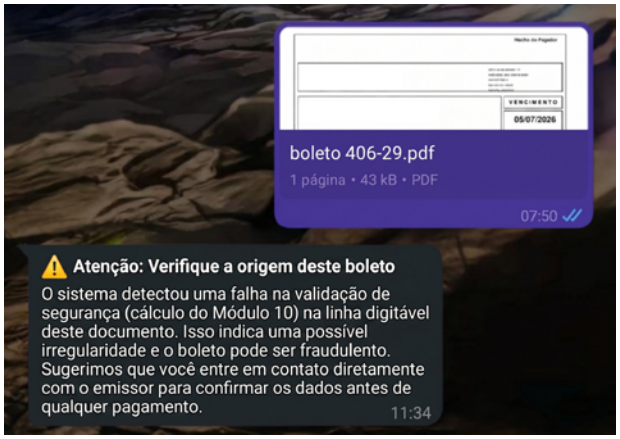
Fonte: elaborado pelos autores (2026)

A implementação desse módulo visa atender à funcionalidade mais solicitada pelos participantes no segundo levantamento, em que 88,9% indicaram interesse na identificação de links suspeitos. A integração ao WhatsApp permite ainda que o conteúdo seja encaminhado ao assistente utilizando a mesma plataforma na qual, segundo o primeiro levantamento, os participantes relataram maior percepção de exposição a tentativas de golpes.

O protótipo também possibilita a análise de mensagens textuais. Nesse caso, o conteúdo recebido é submetido a expressões regulares e aos mecanismos NLP descritos na metodologia. Quando são identificados elementos considerados relevantes pelas regras implementadas, o sistema utiliza essas informações na determinação do nível de risco informado ao usuário.

Outra funcionalidade implementada corresponde à análise de boletos. O usuário pode encaminhar documentos em formato PDF ou imagens do boleto para processamento. O sistema realiza a extração das informações disponíveis, a leitura da linha digitável ou do código de barras e as verificações estruturais implementadas no protótipo. A Figura 7 apresenta um exemplo de interação com esse módulo.

Figura 7 - Exemplo de análise de boleto realizada pelo Argus



Fonte: elaborado pelos autores (2026)

O módulo de áudio também foi incorporado ao MVP. As mensagens de voz encaminhadas ao assistente são convertidas e transcritas utilizando as ferramentas descritas na metodologia. O texto resultante da transcrição é posteriormente encaminhado aos mesmos mecanismos utilizados para a análise das mensagens textuais, permitindo reaproveitar a estrutura de processamento já implementada.

Além da interface integrada ao WhatsApp, foi desenvolvida uma aplicação web destinada ao gerenciamento do serviço. Essa interface contempla funcionalidades relacionadas à administração do sistema, gerenciamento de assinaturas, controle de cotas e avaliação dos alertas produzidos pelo mecanismo de geração de rascunhos.

O sistema de alertas foi implementado como uma funcionalidade complementar aos mecanismos de análise. Quando um mesmo conteúdo atinge o limite definido de ocorrências, o sistema utiliza a API do Google Gemini para produzir um rascunho de alerta preventivo. O conteúdo gerado é encaminhado ao painel administrativo e somente pode ser disponibilizado aos usuários após avaliação de um administrador. Esse mecanismo está relacionado à demanda identificada no levantamento, no qual 77,8% dos participantes manifestaram interesse em receber alertas preventivos sobre novos golpes.

Os resultados demonstram, portanto, que foi possível implementar no período disponível um protótipo que integra diferentes mecanismos de análise em uma única solução e oferece interação por meio do WhatsApp. Contudo, os exemplos apresentados nesta etapa demonstram o funcionamento técnico das funcionalidades implementadas, não sendo suficientes para estabelecer a acurácia do sistema na identificação de fraudes. Uma avaliação experimental envolvendo conjuntos previamente classificados de conteúdos legítimos e fraudulentos será necessária para mensurar indicadores como precisão, sensibilidade e ocorrência de falsos positivos e falsos negativos.

3.4 AVALIAÇÃO NO CONTEXTO DO EVENTO

Ao término do período de desenvolvimento, o MVP do Argus foi apresentado a três avaliadores da área empresarial durante o TSW. A apresentação abordou a proposta de valor, as principais funcionalidades do protótipo e aspectos relacionados à viabilidade do projeto. Como resultado da avaliação realizada no contexto do evento, o Argus obteve a terceira colocação entre os projetos apresentados.

Durante a avaliação, foram discutidos aspectos relacionados à viabilidade financeira da solução, à privacidade dos dados processados e à integração com o WhatsApp. Essas questões contribuíram para a identificação de aspectos que demandam aprofundamento em etapas posteriores de desenvolvimento, especialmente quanto à sustentabilidade

do modelo de negócio, ao tratamento de dados e à adequação contínua às políticas das plataformas utilizadas.

4 CONCLUSÃO

Este trabalho apresentou o desenvolvimento do Argus e uma avaliação exploratória inicial da proposta, buscando investigar a percepção pública acerca dos golpes digitais e sua receptividade a uma ferramenta de apoio à identificação de conteúdo potencialmente fraudulento. A partir das etapas de levantamento exploratório e desenvolvimento, foi possível implementar um MVP integrado ao WhatsApp, capaz de processar diferentes tipos de conteúdo, incluindo links, mensagens textuais, áudios e boletos.

Os levantamentos realizados indicaram a relevância do contexto considerado para o desenvolvimento da solução. O WhatsApp apresentou elevada utilização entre os participantes e foi apontado como o principal ambiente de exposição percebida a tentativas de golpes digitais. O segundo levantamento demonstrou receptividade inicial à proposta demonstrou receptividade inicial à proposta e permitiu identificar funcionalidades consideradas relevantes pelos participantes, especialmente a análise de links suspeitos e o recebimento de alertas preventivos. Esses resultados contribuíram para direcionar os requisitos incorporados ao MVP.

Do ponto de vista tecnológico, o desenvolvimento demonstrou a possibilidade de integrar, em um único protótipo, diferentes mecanismos de verificação e processamento de conteúdo, mantendo o WhatsApp como interface de interação com o usuário. Entretanto, os resultados apresentados neste estudo demonstram principalmente o funcionamento das funcionalidades implementadas e a receptividade inicial à proposta, não permitindo ainda estabelecer a eficácia do Argus na identificação de fraudes. Além disso, o desenvolvimento ocorreu durante um período de 52 horas, no contexto do Techstars Startup Weekend, o que restringiu a implementação de funcionalidades adicionais, o refinamento da interface e a realização de uma avaliação experimental mais abrangente.

Como trabalhos futuros, pretende-se realizar uma avaliação sistemática dos mecanismos de detecção utilizando conjuntos de conteúdos previamente classificados como legítimos e fraudulentos, permitindo mensurar indicadores como precisão, sensibilidade e ocorrência de falsos positivos e falsos negativos. Também se propõe ampliar os testes com usuários, avaliar aspectos de usabilidade, privacidade e segurança no tratamento das informações, além de aprimorar os mecanismos de análise e geração de alertas. Essas etapas poderão fornecer evidências mais consistentes sobre a efetividade da solução e orientar sua evolução para utilização em um contexto real.

REFERÊNCIAS

CASTELLS, Manuel. **A sociedade em rede**. 6. ed. São Paulo: Paz e Terra, 1999. (A Era da Informação: Economia, Sociedade e Cultura; v. 1).

DEVLIN, J. *et al.* BERT: **Pre-training of Deep Bidirectional Transformers for Language Understanding**. arXiv preprint arXiv:1810.04805, 2018. Disponível em: https://arxiv.org/abs/1810.04805. Acesso em: 5 ago. 2026.

EVOLUTION FOUNDATION. **Documentação do Evolution Foundation**. [S. l.]: Evolution Foundation, 2026. Disponível em: https://docs.evolutionfoundation.com.br. Acesso em: 23 ago. 2026.

FEBRABAN. **Especificações Técnicas para a Confecção de Boleto de Cobrança** (Layout - Código de Barras). Versão 7. São Paulo: Federação Brasileira de Bancos, 2023. Disponível em: https://portal.febraban.org.br/pagina/3425/33/pt-br/layout-febraban. Acesso em: 5 ago. 2026.

FFMPEG. **FFmpeg**. [S. l.], [2026]. Disponível em: https://ffmpeg.org/. Acesso em: 5 ago. 2026.

FÓRUM BRASILEIRO DE SEGURANÇA PÚBLICA. **20º Anuário Brasileiro de Segurança Pública 2026**. São Paulo: Fórum Brasileiro de Segurança Pública, 2026. Disponível em: https://publicacoes.forumseguranca.org.br/handle/fbsp/305. Acesso em: 23 ago. 2026.

GOOGLE. **Gemini API**. [S. l.]: Google LLC, 2026. Disponível em: https://ai.google.dev/. Acesso em: 23 ago. 2026.

HOHPE, Gregor; WOOLF, Bobby. **Enterprise Integration Patterns:** Designing, Building, and Deploying Messaging Solutions. Boston: Addison-Wesley, 2003.

INTERNET ENGINEERING TASK FORCE (IETF). **RFC 3912**: WHOIS Protocol Specification. [S. l.]: IETF, 2004. Disponível em: https://www.rfc-editor.org/rfc/rfc3912. Acesso em: 26 ago. 2026.

NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY (NIST). Defense-in-depth. In: **Computer Security Resource Center (CSRC) Glossary**. Gaithersburg, MD: NIST, [2017]. Disponível em: https://csrc.nist.gov/glossary/term/defense_in_depth. Acesso em: 5 ago. 2026.

OBJECT MANAGEMENT GROUP (OMG). **Business Process Model and Notation (BPMN)**. Versão 2.0.2. Needham: OMG, 2013. Disponível em: https://www.omg.org/spec/BPMN/2.0.2/PDF. Acesso em: 23 ago. 2026.

RADFORD, A. *et al.* Robust Speech Recognition via Large-Scale Weak Supervision. **arXiv preprint arXiv:2212.04356**, 2022. Disponível em: https://arxiv.org/abs/2212.04356. Acesso em: 5 ago. 2026.

RAMÍREZ, Sebastián. **FastAPI**. [S. l.], [2026]. Disponível em: https://fastapi.tiangolo.com/pt/. Acesso em: 5 ago. 2026.

SALAHDINE, Fatima; KAABOUCH, Naima. Social Engineering Attacks: A Survey. **Future Internet**, v. 11, n. 4, p. 89, abr. 2019. Disponível em: https://www.mdpi.com/1999-5903/11/4/89. Acesso em: 28 ago. 2026.

SOMMERVILLE, Ian. **Software Engineering**. 9. ed. Boston: Pearson, 2011.

TESSERACT OCR. **Tesseract Open Source OCR Engine**. [S. l.], [2026]. Disponível em: https://github.com/tesseract-ocr/tesseract. Acesso em: 5 ago. 2026.

VIRUSTOTAL. **VirusTotal**. [S. l.], [2026]. Disponível em: https://www.virustotal.com/. Acesso em: 5 ago. 2026.

ZBAR. **ZBar Bar Code Reader**. [S. l.], [2026]. Disponível em: https://zbar.sourceforge.net/. Acesso em: 21 ago. 2026.

ZHOU, M. *et al.* Progress in Neural NLP: Modeling, Learning, and Reasoning. **Engineering**, v. 6, p. 275-290, jan. 2020. Disponível em: https://doi.org/10.1016/j.eng.2019.12.014 Acesso em: 27 ago. 2026.

DESENVOLVIMENTO E VALIDAÇÃO FUNCIONAL DE UM FLUXO AUTOMATIZADO PARA MONITORAMENTO DE PUBLICAÇÕES CIENTÍFICAS COM N8N E OPENALEX

DEVELOPMENT AND FUNCTIONAL VALIDATION OF AN AUTOMATED WORKFLOW FOR MONITORING SCIENTIFIC PUBLICATIONS USING N8N AND OPENALEX

Fabrício Tonetto Londero[1], Deivison Shindi Takatu[2] e Adriano Valério Santos da Silva[3]

RESUMO

O acompanhamento contínuo da produção científica pode demandar a repetição de buscas, a organização de registros e a verificação de novas publicações em diferentes áreas. Este trabalho teve como objetivo desenvolver e validar funcionalmente uma prova de conceito para automatizar a recuperação e a distribuição de publicações científicas utilizando o n8n e a API do OpenAlex. O fluxo foi implantado de forma auto-hospedada em um notebook com Debian, utilizando Docker e armazenamento persistente. Foram configuradas 16 categorias temáticas, com expressões de busca em português e inglês, janela móvel de dez dias e limites entre três e cinco resultados por categoria. O processo contemplou consulta à API, extração e padronização de metadados, controle de registros processados, ordenação dos resultados e geração de um relatório em HTML. A validação funcional realizada em 1º de setembro de 2026 resultou no envio do Radar acadêmico semanal, contendo títulos, autores, fontes, datas, tópicos, resumos, hiperlinks e indicação de acesso aberto. Embora alguns documentos tenham apresentado correspondência temática parcial, a execução demonstrou a viabilidade técnica da integração. Conclui-se que a solução pode apoiar o monitoramento e a triagem inicial da literatura, sem substituir a avaliação crítica do pesquisador ou os procedimentos de uma revisão sistemática.

Palavras-chave: Automação, OpenAlex, recuperação da informação, literatura científica.

1Professor e pesquisador no Centro Universitário Facens (UniFacens). E-mail: fabriciotonettolondero@gmail.com. ORCID: https://orcid.org/0000-0002-4822-4981
2Professor no Centro Universitário Facens (UniFacens) e na Faculdade de Tecnologia de São Paulo (FATEC). E-mail: deivisontakatu@gmail.com. ORCID: https://orcid.org/0000-0002-8210-1002
3Professor e coordenador nos cursos de Inteligência Artificial no Centro Universitário Facens (UniFacens), e arquiteto de soluções (atuando com times de arquitetura e GenAI). E-mail: adrianovss@gmail.com. ORCID: https://orcid.org/0009-0002-2252-1272

ABSTRACT

Continuously monitoring scientific output may require repeated searches, record organization, and the identification of new publications across different research areas. This study aimed to develop and functionally validate a proof of concept for automating the retrieval and distribution of scientific publications using n8n and the OpenAlex API. The workflow was self-hosted on a Debian notebook using Docker and persistent storage. Sixteen thematic categories were configured with search expressions in Portuguese and English, a ten-day moving window, and limits ranging from three to five results per category. The process included API querying, metadata extraction and standardization, control of processed records, result ranking, and generation of an HTML report. The functional validation conducted on September 1, 2026, resulted in the successful delivery of a weekly academic radar containing titles, authors, publication sources, dates, topics, abstracts, hyperlinks, and open-access indicators. Although some documents showed only partial thematic correspondence, the execution demonstrated the technical feasibility of the integration. The solution can support literature monitoring and initial screening but does not replace researchers' critical assessment or the formal procedures required for a systematic review.

Keywords: *Automation, OpenAlex, information retrieval, scientific literature.*

1 INTRODUÇÃO

A identificação e o acompanhamento da literatura científica constituem atividades fundamentais para o desenvolvimento de pesquisas, uma vez que permitem reconhecer avanços, lacunas, métodos e resultados relacionados a determinado campo de conhecimento. Entretanto, o crescimento contínuo da produção acadêmica e sua distribuição entre diferentes periódicos, eventos e repositórios ampliam a quantidade e a heterogeneidade dos documentos que precisam ser consultados. Nessas condições, a realização recorrente de buscas, a organização dos registros recuperados e a identificação de trabalhos relevantes podem tornar-se atividades demoradas e sujeitas a inconsistências (Marcos-Pablos; García-Peñalvo, 2019; De La Torre-López; Ramírez; Romero, 2023).

Os pesquisadores precisam formular estratégias de busca, definir períodos, consultar diferentes fontes, excluir registros duplicados e avaliar a correspondência entre os documentos encontrados e seus interesses de investigação. Além disso, uma única expressão de busca nem sempre representa adequadamente a necessidade informacional do usuário, pois os termos empregados, os critérios de relevância e o direcionamento da pesquisa podem ser refinados ao longo do processo (Li *et al.*, 2026). Desse modo, embora as bases bibliográficas digitais facilitem o acesso à produção acadêmica, parte considerável do trabalho de recuperação e organização das informações ainda depende de procedimentos repetitivos.

A automação tem sido investigada como uma alternativa para apoiar diferentes etapas relacionadas à pesquisa bibliográfica. Entre as atividades passíveis de apoio computacional estão a construção ou execução de estratégias de busca, a recuperação de registros, a remoção de duplicidades, a seleção inicial de documentos e a extração de informações. Contudo, as tecnologias existentes costumam concentrar-se em etapas específicas e apresentam diferentes níveis de maturidade, aplicabilidade e desempenho (De La Torre-López; Ramírez; Romero, 2023; Tóth *et al.*, 2024). Por essa razão, a automação deve ser compreendida como suporte ao trabalho intelectual do pesquisador, preservando sua participação na definição dos critérios e na avaliação da relevância dos documentos. Ferramentas como o LiteRev demonstram a possibilidade de utilizar recursos computacionais para organizar documentos e recomendar publicações potencialmente relevantes (Orel *et al.*, 2023). Entretanto, a automação costuma abranger etapas específicas, sem substituir a condução de revisões sistemáticas ou a análise intelectual da literatura.

Entre as fontes abertas de metadados científicos encontra-se o OpenAlex, uma infraestrutura destinada à organização e recuperação de informações sobre trabalhos, autores, instituições e fontes de publicação, com acesso programático aos registros por meio de uma API (Priem; Piwowar; Orr, 2022; Borrego; Urbano, 2025; OpenAlex, 2026). Estudos anteriores demonstram a utilização dessa infraestrutura na recuperação, organização e enriquecimento de dados bibliográficos (Aria *et al.*, 2023; Harder, 2024; Wei, 2025), enquanto estudos comparativos apontam diferenças na cobertura do OpenAlex em relação a bases bibliográficas consolidadas, como Scopus e Web of Science (Maddi; Maisonobe; Boukacem-Zeghmouri, 2025). Apesar de sua cobertura de referências e de publicações ser comparável à de bases proprietárias em diferentes conjuntos, estudos identificam diferenças na cobertura, na completude e na precisão de determinados metadados do OpenAlex, o que exige que os registros recuperados sejam tratados e avaliados conforme a finalidade da aplicação (Culbert *et al.*, 2025; Céspedes *et al.*, 2025; Borrego; Urbano, 2025).

Apesar dos avanços na automação da recuperação e triagem da literatura, permanece pouco explorada a integração de plataformas de automação de *workflows* com infraestruturas abertas de metadados para o monitoramento recorrente e configurável de novas publicações. Nesse contexto, este estudo investiga em que medida a integração entre n8n [4]e OpenAlex pode viabilizar um fluxo automatizado, replicável e configurável para recuperação e distribuição periódica de publicações científicas. A questão de pesquisa que orienta o estudo é: é tecnicamente viável implementar, por meio da integração entre o n8n e OpenAlex, um fluxo automatizado e reprodutível para consulta, tratamento, deduplicação,

4 É uma plataforma de automação de fluxos de trabalho (*workflows*) de código aberto (*open-source*), que permite conectar aplicativos, APIs e bancos de dados de forma visual (N8n, 2026)

organização e distribuição periódica de publicações científicas? A utilização de uma plataforma de automação de *workflows* permite integrar, em um mesmo fluxo, diferentes etapas de processamento, comunicação com serviços externos e execução programada (Venkiteela, 2025). O fluxo contempla a execução programada de consultas, a delimitação temporal das publicações, o tratamento de duplicidades, a organização dos resultados por área temática e a geração automática de uma mensagem em formato HTML (N8n, 2026). Como contribuição, este trabalho apresenta a concepção, implementação e validação funcional de uma arquitetura aberta, configurável e reproduzível que integra uma plataforma de automação de *workflows* a uma infraestrutura aberta de metadados científicos para o monitoramento periódico da literatura. A solução automatiza as etapas de consulta, tratamento, deduplicação, organização e distribuição dos registros recuperados, mantendo a avaliação de pertinência sob responsabilidade do pesquisador. O artefato e os arquivos necessários à sua reprodução são disponibilizados publicamente, permitindo a sua adaptação a diferentes domínios temáticos.

2 TRABALHOS RELACIONADOS

A aplicação de recursos computacionais na recuperação e análise da literatura científica tem sido investigada em diferentes estudos, tais como o de Marcos-Pablos e García-Peñalvo (2019) que propuseram uma metodologia iterativa que combina mineração de textos, aprendizado de máquina e decisões supervisionadas pelo pesquisador para organizar e refinar resultados obtidos em bases científicas. De la Torre-López, Ramírez e Romero (2023), ao analisarem 34 estudos primários, identificaram o emprego de técnicas de inteligência artificial em tarefas como formulação de buscas, seleção de estudos e extração de informações. Os autores ressaltam que a automação pode reduzir atividades repetitivas, mas não deve eliminar a participação humana na avaliação dos documentos.

No contexto das revisões sistemáticas, Tóth *et al.* (2024) analisaram 123 estudos sobre automação e constataram que a maior parte das soluções se concentra em etapas específicas, principalmente na triagem de registros. Apenas 11 estudos, correspondentes a 8,9% da amostra, automatizaram múltiplas etapas do processo. Entre as ferramentas desenvolvidas para essa finalidade, Orel *et al.* (2023) apresentaram o LiteRev, que utiliza processamento de linguagem natural e aprendizado de máquina para organizar documentos por tópicos e recomendar trabalhos potencialmente relevantes. Em sua avaliação, a ferramenta reduziu em 56% o volume de documentos submetidos à triagem, embora não tenha recuperado todos os estudos encontrados pela análise manual.

Outros trabalhos exploraram especificamente os dados e a API do OpenAlex. Aria *et al.* (2023) desenvolveram o pacote openalexR, que permite consultar entidades como

trabalhos, autores, instituições e fontes por meio da linguagem R. Harder (2024) apresentou um procedimento baseado em Bash e SQL para recuperar dados das APIs do OpenAlex e do Scopus, voltado principalmente à síntese de evidências e às análises bibliográficas. Wei (2025), por sua vez, desenvolveu o OneCLick, uma aplicação web que consulta o OpenAlex para enriquecer registros bibliográficos com palavras-chave inferidas automaticamente. A ferramenta alcançou taxas de enriquecimento iguais ou superiores a 81% nos conjuntos avaliados, com latência aproximada de 0,30 segundo por DOI.

A confiabilidade dos dados do OpenAlex também foi examinada por Culbert *et al.* (2025), que compararam 16,8 milhões de publicações compartilhadas entre OpenAlex, Web of Science e Scopus. Os resultados indicaram cobertura média de referências comparável entre as bases, embora o OpenAlex apresentasse maior disponibilidade de identificadores ORCID e menor presença de resumos. Diferentemente dos trabalhos correlatos, a presente pesquisa integra o OpenAlex ao n8n para executar periodicamente estratégias de busca relacionadas a diferentes áreas, tratar duplicidades, organizar os registros recuperados e encaminhar os resultados por e-mail em formato HTML. O foco, portanto, não está na automatização integral de revisões sistemáticas nem no enriquecimento bibliométrico, mas no desenvolvimento e na avaliação de um fluxo configurável para o monitoramento recorrente de novas publicações.

3 METODOLOGIA

Foi conduzido um estudo aplicado de desenvolvimento tecnológico, estruturado como uma prova de conceito seguida de validação funcional. O procedimento metodológico foi organizado em quatro etapas: (i) definição das categorias temáticas e estratégias de busca; (ii) implementação do *workflow* e integração com a API do OpenAlex; (iii) processamento, deduplicação e organização dos registros; e (iv) execução controlada do fluxo e avaliação dos critérios funcionais previamente definidos. Foram considerados como critérios de validação: conclusão da execução sem falhas impeditivas, recuperação dos registros, aplicação das regras de deduplicação, organização por categoria, preservação dos metadados e entrega do relatório. O artefato consiste em um fluxo automatizado baseado no n8n e na API do OpenAlex, destinado à recuperação, organização e distribuição periódica de metadados de publicações científicas. A avaliação concentrou-se no funcionamento integrado de seus componentes, sem realizar comparação de desempenho com buscas manuais ou acompanhamento longitudinal dos resultados. Nesse sentido, a expressão validação funcional é empregada neste trabalho para designar a verificação do funcionamento integrado do artefato segundo os critérios previamente definidos, não correspondendo a uma avaliação de precisão, revocação ou superioridade do mecanismo de recuperação.

O n8n foi implantado de forma auto-hospedada em um notebook com sistema operacional Debian, utilizando Docker Compose. A escolha do n8n está relacionada à sua capacidade de estruturar *workflows* por meio da integração de diferentes etapas e serviços, característica que favorece a construção de processos automatizados e modulares (Venkiteela, 2025). O serviço foi configurado na porta 5678, com política de reinicialização unless-stopped e volume persistente para armazenamento dos fluxos, das configurações e do histórico de execuções. Essa configuração possibilitou a manutenção do serviço independentemente de uma assinatura da plataforma hospedada, preservando a possibilidade de execução da automação em infraestrutura própria, característica compatível com abordagens de automação baseadas em implantação auto-hospedada (Venkiteela, 2025).

Como fonte de dados, foi utilizado o endpoint de trabalhos da API do OpenAlex. Foram definidas 16 categorias temáticas, acompanhadas de expressões de busca em português e inglês e de limites entre três e cinco publicações por categoria. As consultas consideraram uma janela móvel de dez dias anteriores à execução, adotada para contemplar possíveis atrasos na indexação dos documentos. As categorias, expressões de busca e respectivos limites constituem parâmetros configuráveis do fluxo e encontram-se integralmente disponíveis, juntamente com o *workflow* exportado, no repositório público associado ao trabalho.

A validação funcional foi realizada em 1º de setembro de 2026, abrangendo publicações datadas entre 22 de agosto e 1º de setembro de 2026. Foram observados o término da execução sem falhas impeditivas, o processamento dos registros retornados pela API, a organização dos resultados por categoria, a apresentação dos metadados e o recebimento do relatório por e-mail. Os arquivos necessários para reprodução do fluxo estão disponíveis em repositório público no GitHub[5].

4 DESENVOLVIMENTO DO FLUXO AUTOMATIZADO

O fluxo foi desenvolvido no editor visual do n8n por meio da conexão sequencial de sete nós: Schedule Trigger, Definir assuntos, Buscar OpenAlex, Extrair artigos, Remove Duplicates, Montar Email e Send a message. Essa abordagem utiliza a estrutura de *workflows* da plataforma para organizar diferentes etapas de processamento em uma sequência automatizada e integrada (Venkiteela, 2025). Cada nó foi responsável por uma etapa específica, desde o acionamento periódico até o encaminhamento do relatório. Essa organização modular permite modificar as consultas, o tratamento dos registros ou o serviço de envio sem reconstruir integralmente a solução. A Figura 1 apresenta a arquitetura do fluxo desenvolvido.

5 Disponível em: https://github.com/Ernakh/radar-academico-n8n-OpenAlex

Figura 1 - Arquitetura do fluxo automatizado desenvolvido no n8n.



Fonte: elaborado pelos autores (2026).

O nó *Schedule Trigger* foi configurado para iniciar o processo semanalmente, aos sábados, às 7h. Em seguida, o nó “Definir assuntos”, implementado em JavaScript, estabelece as categorias, as expressões de busca e o número máximo de resultados apresentados em cada uma delas. Foram configuradas 16 categorias, distribuídas entre os eixos de jogos e educação, inteligência artificial, mobilidade e tecnologias veiculares. As expressões combinaram termos em português e inglês por meio dos operadores booleanos AND e OR. Nesse mesmo nó foram calculadas as datas inicial e final da janela de consulta, correspondentes aos dez dias anteriores à execução.

O nó “Buscar OpenAlex” realiza requisições HTTP do tipo GET ao *endpoint /works* da API do OpenAlex. Para cada categoria, são encaminhados a expressão de busca, o intervalo de publicação, a ordenação e o número de resultados solicitado. As consultas são executadas individualmente, com intervalo de cinco segundos, para reduzir a concentração de requisições. A autenticação é realizada por meio de uma chave da API armazenada no gerenciador de credenciais do n8n, sem sua inserção direta no código do fluxo.

As respostas da API são processadas pelo nó “Extrair artigos”, também implementado em JavaScript. Esse nó percorre os registros retornados e seleciona título, autores, data de publicação, fonte, tópico principal, DOI, endereço eletrônico, condição de acesso aberto, quantidade de citações e pontuação de relevância. A quantidade de autores apresentada foi limitada aos seis primeiros. Como o OpenAlex disponibiliza determinados resumos na forma de índice invertido, foi implementada uma função para reconstruir o texto com base nas palavras e em suas respectivas posições. Após a reconstrução, espaços excedentes são removidos e o resumo é limitado a 700 caracteres para manter o relatório conciso.

Para o controle de repetição, cada registro recebe uma chave formada pela categoria e pelo DOI ou, quando este não está disponível, pelo identificador do OpenAlex. O nó “Remove Duplicates” compara essa chave com os itens processados em execuções anteriores e mantém apenas os registros ainda não encaminhados naquela categoria. Esse procedimento evita a repetição semanal de uma publicação em uma mesma seção, embora permita que um artigo seja apresentado em categorias diferentes quando corresponder a mais de uma expressão de busca.

No nó Montar Email, os artigos são agrupados por categoria e ordenados, primeiramente, pela pontuação de relevância fornecida pelo OpenAlex, seguida pelo número de citações e pela data de publicação. Após a ordenação, são selecionados entre três e cinco artigos, conforme o limite definido para cada categoria. O conteúdo é convertido em HTML, com tratamento de caracteres especiais, formatação das datas e indicação visual das publicações de acesso aberto. Para cada registro, o relatório apresenta título com hiperlink, autores, fonte, data, tópico principal e resumo.

Por fim, o nó *Send a message* encaminha o relatório utilizando uma conta Microsoft Outlook autenticada por OAuth 2.0. O assunto da mensagem é composto pela expressão "Radar acadêmico semanal" e pela data da execução. O corpo é enviado em formato HTML para preservar a organização das categorias, os hiperlinks e os elementos visuais. O fluxo exportado do n8n, o arquivo Docker Compose, as expressões de busca e as instruções para implantação da solução estão disponíveis no repositório público (Londero, 2026).

5 RESULTADOS E DISCUSSÃO

A execução experimental demonstrou o funcionamento de todas as etapas previstas no *workflow*: acionamento programado, consulta à API, processamento dos registros, deduplicação, agrupamento temático e distribuição do relatório. Entretanto, o desempenho funcional não foi uniforme quanto à pertinência temática dos resultados. Foram identificados registros cuja recuperação ocorreu em razão de correspondências lexicais, embora apresentassem baixa aderência ao domínio pretendido. Esse resultado indica que a estratégia baseada predominantemente em correspondência textual permite recuperar documentos associados aos termos consultados, mas não é suficiente, isoladamente, para assegurar a pertinência semântica em relação ao domínio temático pretendido. O relatório contemplou publicações datadas entre 22 de agosto e 1º de setembro de 2026. A mensagem preservou a estrutura em HTML, apresentando títulos com hiperlinks, autores, fonte, data de publicação, tópico principal, resumo e indicação de acesso aberto. A Figura 2 apresenta um recorte do relatório produzido.

A organização por categorias permitiu reunir publicações relacionadas aos diferentes interesses definidos no fluxo. Na área educacional, foram recuperados trabalhos sobre aprendizagem baseada em jogos, desenvolvimento de jogos educacionais, gamificação, formação de professores e aplicação da inteligência artificial à educação. Entre os resultados encontram-se estudos sobre definição de aprendizagem baseada em jogos, seus efeitos na educação básica e utilização de jogos digitais na formação docente. Nas áreas de mobilidade e tecnologias veiculares, o relatório apresentou publicações relacionadas a

C-V2X, veículos autônomos, sistemas de assistência ao condutor, detecção de pedestres, redes automotivas e navegação com LiDAR. Esses resultados demonstram que uma única arquitetura pode acompanhar diferentes domínios mediante a substituição ou ampliação das expressões de busca.

Figura 2 - Recorte do e-mail recebido em 01/09/2026.



Radar acadêmico semanal

Artigos publicados entre **22/08/2026** e **01/09/2026**.

Jogos educacionais, serious games e aprendizagem baseada em jogos

INTEGRATION OF GAME-BASED LEARNING IN TEACHING MATHEMATICS 7

Autores: Mary Joy B. Etorma

Periódico: World Journal of Advanced Research and Reviews | **Publicação:** 28/08/2026 | Acesso aberto

Assunto: Educational Games and Gamification

Mathematics 7 serves as a crucial foundation for learners transitioning to junior high school, yet many students experience low engagement and difficulty in understanding abstract mathematical concepts. Traditional teaching approaches often fail to address diverse learning needs and motivation levels. In response, game-based learning (GBL) has emerged as a learner-centered instructional strategy that integrates play, challenge, and feedback to enhance understanding and participation. This study examined the integration of game-based learning—both ICT-based and non-ICT-based—in teaching Mathematics 7 and its influence on learner performance and classroom instruction using a quasi-experimental

Defining Game-Based Learning

Autores: Elīna Grāvelsiņa, Linda Daniela

Periódico: International Journal of Game-Based Learning | **Publicação:** 28/08/2026 | Acesso aberto

Assunto: Educational Games and Gamification

Game-based learning (GBL) has emerged as a widely discussed educational approach; however, this term may be referred to by various names, creating challenges for both research and practice. This article examines terminological gaps in the existing discourse on GBL and distinguishes it from related concepts such as gamification, serious games, etc. The focus is on understanding GBL and the related terminology provided. For this reason, a systematic literature analysis of 52 articles was conducted. Key findings contain an overview of the definition of "game-based learning" and its 72 variations. To gain a better understanding, terms were analyzed to answer the following questions: what specifi

Fonte: elaborado pelos autores (2026).

A recuperação também apresentou documentos com correspondência apenas parcial às categorias. Na seção relacionada a LiDAR, ROS e navegação autônoma, por exemplo, foram incluídos trabalhos sobre estimativa de biomassa florestal e navegação de robôs em silos de grãos. Embora esses documentos contenham termos como LiDAR, sensoriamento e navegação, seus contextos não correspondem diretamente ao interesse principal em mobilidade veicular. Esse comportamento evidencia que a correspondência entre palavras-chave não assegura a pertinência do resultado. Conforme discutem Marcos-Pablos e García-Peñalvo (2019), o aumento da abrangência das consultas também pode elevar a heterogeneidade dos documentos recuperados, tornando necessária a participação do pesquisador na seleção final.

A apresentação automática dos metadados mostrou-se adequada para uma etapa inicial de monitoramento e triagem. Os resultados corroboram aplicações anteriores da API do OpenAlex voltadas à recuperação e ao tratamento programático de dados bibliográficos (Aria *et al.*, 2023; Harder, 2024; Wei, 2025) e estudos recentes que investigam a automação de *workflows* utilizando o n8n, evidenciando seu potencial para reduzir operações manuais em processos estruturados (Amir; Atif, 2026). Entretanto, a disponibilidade dos campos variou entre os registros, especialmente em relação aos resumos e aos identificadores persistentes. Essa limitação é compatível com estudos que identificaram diferenças na completude e precisão dos metadados do OpenAlex, particularmente em relação às informações linguísticas e bibliográficas (Céspedes *et al.*, 2025; Culbert *et al.*, 2025). Por essa razão, o fluxo adotou alternativas para composição dos links e uma mensagem padrão quando determinados dados não estavam disponíveis.

A utilização do n8n em ambiente auto-hospedado possibilitou executar o processo sem dependência do serviço hospedado da plataforma. A combinação entre Docker Compose, volume persistente e política de reinicialização permitiu manter as configurações e o fluxo no equipamento utilizado como servidor. A modularidade dos nós também possibilita alterar categorias, consultas, limites de resultados e serviço de e-mail sem modificar toda a arquitetura.

Os resultados devem ser interpretados como uma validação funcional da prova de conceito. Foi analisada uma única execução no ambiente auto-hospedado, sem mensuração do tempo economizado, comparação com buscas manuais ou cálculo de precisão e revocação. O mecanismo de controle de repetição também precisará ser observado em execuções futuras para verificar seu comportamento ao longo do tempo. Portanto, os achados demonstram a viabilidade técnica da integração entre n8n e OpenAlex, mas não permitem concluir que todas as publicações recuperadas sejam relevantes nem que o fluxo identifique a totalidade dos trabalhos relacionados às categorias pesquisadas.

6 CONSIDERAÇÕES FINAIS

Este trabalho apresentou o desenvolvimento e a validação funcional de um fluxo automatizado baseado no n8n e na API do OpenAlex para o monitoramento periódico de publicações científicas. A execução realizada em ambiente auto-hospedado demonstrou que a solução foi capaz de consultar a fonte bibliográfica, processar os registros recuperados, organizá-los em 16 categorias temáticas e encaminhar um relatório em formato HTML. Os resultados permitem concluir que a integração entre n8n e OpenAlex é tecnicamente viável para a implementação de um mecanismo automatizado de monitoramento bibliográfico recorrente. A evidência obtida, contudo, sustenta uma conclusão restrita à viabilidade

funcional da arquitetura, não permitindo inferir sua precisão, cobertura ou superioridade em relação a estratégias manuais ou outras soluções automatizadas. A principal contribuição do estudo consiste, portanto, em demonstrar uma arquitetura aberta e reproduzível para operacionalização do monitoramento periódico da literatura, estabelecendo uma base para avaliações posteriores de pertinência, cobertura e eficiência

A principal contribuição está na integração de ferramentas acessíveis em uma arquitetura modular, capaz de apoiar pesquisadores no acompanhamento de diferentes áreas do conhecimento. A implantação por Docker Compose permite manter o fluxo em infraestrutura própria, enquanto a disponibilização dos arquivos no GitHub favorece sua reprodução e adaptação. Entretanto, a presença de documentos com correspondência temática parcial reforça que a solução deve ser utilizada como mecanismo de alerta e triagem inicial, sem substituir a avaliação crítica do pesquisador ou os procedimentos formais de uma revisão sistemática.

Como limitações, destaca-se a realização de uma única execução no ambiente auto-hospedado, sem comparação com buscas manuais, cálculo de precisão e revocação ou avaliação longitudinal do controle de repetição. Trabalhos futuros poderão acompanhar sucessivas execuções, aperfeiçoar as expressões de busca, avaliar a relevância dos resultados e incluir mecanismos de classificação semântica. Também poderão ser investigadas outras fontes bibliográficas, formas de personalização dos relatórios e indicadores de completude dos metadados recuperados.

REFERÊNCIAS

AMIR, Ahmed Raza; ATIF, Syed Muhammad. Evaluating workflow automation efficiency using n8n: a small-scale business case study. arXiv, 2026. **Preprint arXiv**:2602.01311. DOI: https://doi.org/10.48550/arXiv.2602.01311. Disponível em: https://arxiv.org/abs/2602.01311.

ARIA, Massimo; LE, Trang; CUCCURULLO, Corrado; BELFIORE, Alessandra; CHOE, June. openalexR: an R-tool for collecting bibliometric data from OpenAlex. **The R Journal**, [s. l.], v. 15, n. 4, p. 167-180, 2023. DOI: https://doi.org/10.32614/RJ-2023-089. Disponível em: https://journal.r-project.org/articles/RJ-2023-089/.

BORREGO, Ángel; URBANO, Cristóbal. OpenAlex: features, advantages and limitations of an open database for retrieving and analysing scholarly outputs. **Preprint arXiv**:2512.16434, 2025. DOI: 10.48550/arXiv.2512.16434.

CÉSPEDES, Lucía *et al.* Evaluating the linguistic coverage of OpenAlex: an assessment of metadata accuracy and completeness. **Journal of the Association for Information Science and Technology**, v. 76, n. 6, p. 884-895, 2025. DOI: 10.1002/asi.24979.

CULBERT, Jack H.; HOBERT, Anne; JAHN, Najko; HAUPKA, Nick; SCHMIDT, Marion; DONNER, Paul; MAYR, Philipp. Reference coverage analysis of OpenAlex compared to Web of Science and Scopus. **Scientometrics**, [s. l.], v. 130, p. 2475-2492, 2025. DOI: https://doi.org/10.1007/s11192-025-05293-3. Disponível em: https://link.springer.com/article/10.1007/s11192-025-05293-3.

DE LA TORRE-LÓPEZ, José; RAMÍREZ, Aurora; ROMERO, José Raúl. Artificial intelligence to automate the systematic review of scientific literature. **Computing**, [s. l.], v. 105, p. 2171-2194, 2023. DOI: https://doi.org/10.1007/s00607-023-01181-x. Disponível em: https://link.springer.com/article/10.1007/s00607-023-01181-x.

HARDER, Robin. Using Scopus and OpenAlex APIs to retrieve bibliographic data for evidence synthesis: a procedure based on Bash and SQL. **MethodsX**, [s. l.], v. 12, art. 102601, 2024. DOI: https://doi.org/10.1016/j.mex.2024.102601. Disponível em: https://www.sciencedirect.com/science/article/pii/S2215016124000554.

LI, Jisen; LI, Bingxuan; JIANG, Nanyi; NING, Xuying; WANG, Xiyao; SHEN, Yifan; WANG, Heng; JIAN, Yuqing; WU, Xiaoxia; ATHIWARATKUN, Ben; LU, Pan; YOU, Jiaxuan; ZHAO, Bingxin. Multi-turn agentic scientific literature search via workflow induction. arXiv, 2026. **Preprint arXiv**:2607.00597, versão 2. DOI: https://doi.org/10.48550/arXiv.2607.00597. Disponível em: https://arxiv.org/abs/2607.00597.

LONDERO, Fabrício Tonetto. Radar acadêmico n8n OpenAlex: fluxo automatizado para recuperação de publicações científicas. **GitHub**, 2026. Disponível em: https://github.com/Ernakh/radar-academico-n8n-OpenAlex. Acesso em: 1 set. 2026.

MADDI, Abdelghani; MAISONOBE, Marion; BOUKACEM-ZEGHMOURI, Chérifa. Geographical and disciplinary coverage of open access journals: OpenAlex, Scopus, and WoS. **PLoS One**, v. 20, n. 4, e0320347, 2025. DOI: 10.1371/journal.pone.0320347.

MARCOS-PABLOS, Samuel; GARCÍA-PEÑALVO, Francisco José. Information retrieval methodology for aiding scientific database search. **Soft Computing**, [s. l.], v. 23, p. 11287-11307, 2019. DOI: https://doi.org/10.1007/s00500-018-3568-0. Disponível em: https://link.springer.com/article/10.1007/s00500-018-3568-0.

N8N. **Understand workflows**. [S. l.], 2026. Disponível em: https://docs.n8n.io/build/understand-workflows/.

OPENALEX. OpenAlex: the open catalog to the global research system. [S. l.]: **OurResearch**, 2026. Disponível em: https://openalex.org/.

OREL, Erol; CIGLENECKI, Iza; THIABAUD, Amaury; TEMEREV, Alexander; CALMY, Alexandra; KEISER, Olivia; MERZOUKI, Aziza. An automated literature review tool (LiteRev) for streamlining and accelerating research using natural language processing and machine learning: descriptive performance evaluation study. **Journal of Medical Internet Research**, Toronto, v. 25, e39736, 2023. DOI: https://doi.org/10.2196/39736. Disponível em: https://www.jmir.org/2023/1/e39736.

PRIEM, Jason; PIWOWAR, Heather; ORR, Richard. OpenAlex: a fully-open index of scholarly works, authors, venues, institutions, and concepts. In: INTERNATIONAL CONFERENCE ON SCIENCE, TECHNOLOGY AND INNOVATION INDICATORS, 26., 2022, Granada. **Proceedings [...]. Granada: STI Conference**, 2022. Disponível em: https://arxiv.org/abs/2205.01833.

TÓTH, Barbara; BEREK, László; GULÁCSI, László; PÉNTEK, Márta; ZRUBKA, Zsombor. Automation of systematic reviews of biomedical literature: a scoping review of studies indexed in PubMed. **Systematic Reviews**, [s. l.], v. 13, art. 174, 2024. DOI: https://doi.org/10.1186/s13643-024-02592-3. Disponível em: https://systematicreviewsjournal.biomedcentral.com/articles/10.1186/s13643-024-02592-3.

VENKITEELA, Padmanabhan. n8n: an open-source workflow automation platform for enterprise integration and AI-driven orchestration. **International Journal of Computer Applications**, v. 187, n. 63, p. 1-11, 2025. DOI: 10.5120/ijca2025926031.

WEI, Loo Keat. OneCLick: streamlined metadata enrichment using machine-inferred keywords from OpenAlex. **SoftwareX**, [s. l.], v. 31, art. 102353, 2025. DOI: https://doi.org/10.1016/j.softx.2025.102353. Disponível em: https://www.sciencedirect.com/science/article/pii/S235271102500319X.

PIPELINE DE CONVERSÃO DE ARQUIVOS DE SIMULAÇÃO MOLECULAR PARA VISUALIZAÇÃO EM REALIDADE MISTA[1]

PIPELINE FOR CONVERTING MOLECULAR SIMULATION FILES FOR MIXED REALITY VISUALIZATION

Gabriel Azenha Fachim[2], Luiz Batista Cardoso[3], Maurício Pereira Braga[4] e Mirkos Ortiz Martins[5]

RESUMO

Compreender e interagir tridimensionalmente com moléculas em escala atômica é essencial, mas os dados brutos são frequentemente apenas coordenadas matemáticas ou figuras estáticas de difícil interpretação. Este estudo apresenta um sistema web automatizado que captura informações de repositórios como o banco de dados de proteínas e as transforma em modelos espaciais interativos para ambientes virtuais no motor gráfico Godot. O protótipo anterior exigia etapas manuais e era lento ao calcular as conexões atômicas. A nova versão automatiza o fluxo e utiliza um método eficiente de busca espacial para mapear essas ligações. O desempenho foi avaliado em cinco estruturas contendo de 327 a 4.779 átomos. As medianas da versão otimizada caíram para a faixa de 2,52 a 45,08 milissegundos, contra 18,58 a 4.062,82 milissegundos da anterior, representando aceleração de 7,39 a 90,13 vezes, mantendo a exatidão das conexões. Em testes práticos, o sistema processa corretamente três dos cinco arquivos. A comparação visual da estrutura 1CRN com o PyMOL confirmou a fidelidade da representação aos dados originais. Ainda que o tempo de geração tenha sido drasticamente reduzido, as avaliações indicam a necessidade de ampliar a compatibilidade de leitura para diferentes formatos e realizar testes em condições reais de uso.

Palavras-chave: Python; Modelagem molecular; Motor gráfico; Visualização científica.

---

1 Trabalho Parcial de Dissertação de Mestrado do Programa de Pós-Graduação em Nanociências.

2 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: gabriel.afachim@ufn.edu.br. ORCID: https://orcid.org/0009-0002-8511-4025

3 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: luiz.bcardoso@ufn.edu.br. ORCID: https://orcid.org/0009-0003-7344-4008

4 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: mauricio.pereira@ufn.edu.br. ORCID: https://orcid.org/0009-0003-4346-5836

5 Professor dos Cursos de Computação. Universidade Franciscana. E-mail: mirkos@ufn.edu.br. ORCID: https://orcid.org/0000-0002-3983-1624

ABSTRACT

Understanding and interacting three dimensionally with molecules at the atomic scale is essential, but raw data are often just mathematical coordinates or static figures that are difficult to interpret. This study presents an automated web system that captures information from repositories such as the protein data bank and transforms it into interactive spatial models for virtual environments in the Godot game engine. The previous prototype required manual steps and was slow when calculating atomic connections. The new version automates the workflow and uses an efficient spatial search method to map these bonds. Performance was evaluated on five structures containing from 327 to 4,779 atoms. The median processing times for the optimized version dropped to a range of 2.52 to 45.08 milliseconds, contrasting with the 18.58 to 4,062.82 milliseconds of the previous version, representing a speedup of 7.39 to 90.13 times while maintaining connection accuracy. In practical tests, the system successfully processed three out of five files. The visual comparison of the 1CRN structure with the PyMOL software confirmed the fidelity of the representation to the original data. Although the system drastically reduced generation time, evaluations indicate the need to expand reading compatibility for different formats and conduct tests under real usage conditions.

Keywords: *Python; Molecular modeling; Scientific visualization; Virtual environments.*

1 INTRODUÇÃO

A nanociência é um campo de natureza multidisciplinar, no qual se encontram contribuições da física, da química e da biologia para o estudo da matéria em escala nanométrica. Nesse contexto, a computação desempenha uma função fundamental na modelagem e na simulação de sistemas. Em escala nanométrica, o comportamento dos sistemas é determinado por interações entre átomos e moléculas (Glotzer; Solomon, 2007). O estudo dessas interações, como no *docking* molecular, depende da compreensão tridimensional da estrutura, pois envolve avaliar como uma molécula se posiciona em relação a um alvo (Schlick, 2010).

A visualização tridimensional interativa busca reduzir essa lacuna. Ferramentas consolidadas, como o ChimeraX, combinam representação, análise e interação em um mesmo ambiente (Pettersen *et al.*, 2021). Em ambientes imersivos, a literatura descreve aplicações educacionais, análise estrutural e colaboração, ainda que persistam limitações de renderização e de processamento dos dispositivos (Kuťák *et al.*, 2023). No ensino de geometria molecular, um estudo que comparou Realidade Virtual (RV), simulação em computador e modelos físicos não encontrou diferença significativa de desempenho entre os métodos, mas registrou maior entusiasmo e atitudes mais positivas no grupo que utilizou RV (Brown *et al.*, 2021). Há ainda iniciativas que aproximam os modelos do navegador e de dispositivos

de Realidade Aumentada (AR) ou virtual, reduzindo a distância entre obter os dados e explorá-los de forma interativa (Rodríguez; Peraro; Abriata, 2022).

Nesse contexto, o Godot Engine é um motor de jogos com suporte a ambientes tridimensionais e de realidade estendida que pode ser empregado para visualizar moléculas em um ambiente 3D interativo. Para visualizar estruturas depositadas em bases abertas, como o *Protein Data Bank* (PDB), precisam ser convertidas para um formato estruturado para o motor gráfico. O portal *Research Collaboratory for Structural Bioinformatics Protein Data Bank* (RCSB PDB) oferece busca e acesso a essas estruturas (Burley *et al.*, 2021) e representa o formato de arquivo distribuído pelo repositório.

O projeto já dispõe de um conversor que transforma arquivos PDB em um arquivo estruturado por meio de uma interface Tkinter. O fluxo atual depende de etapas manuais, como baixar o arquivo, selecioná-lo na interface, copiar a saída para o projeto no Godot e iniciar a visualização. Além disso, a inferência de ligações compara cada átomo com os demais, o que faz o tempo de conversão crescer de forma quadrática com o tamanho da estrutura. Estas limitações motivaram a pesquisa de uma solução que integrasse a busca, a conversão e a visualização em um fluxo automático, com desempenho melhorado.

1.1 OBJETIVOS

O objetivo geral deste trabalho foi desenvolver uma plataforma web capaz de buscar arquivos PDB em fontes de dados abertas, como o RCSB PDB, convertê-los em um documento em *JavaScript Object Notation* (JSON) e disponibilizá-lo de forma automática ao Godot Engine para visualização interativa. Para alcançá-lo, foram definidos cinco objetivos específicos: pesquisar bibliotecas e frameworks que permitam desenvolver sistemas web de forma rápida na linguagem Python; adaptar e otimizar o código funcional da plataforma anterior para o novo sistema web e desenvolver sua interface gráfica; implementar a conexão e a recuperação de arquivos PDB por meio do sistema web; conectar o sistema ao Godot Engine para que o visualizador recupere e aceite o documento JSON convertido; e testar e validar o sistema por meio de métricas de tempo, precisão e taxa de sucesso, comparando a eficiência das duas versões do conversor.

O escopo desta pesquisa limitou-se à integração computacional do sistema e à otimização de desempenho da inferência de ligações. A validação estrita dos parâmetros químicos e a avaliação da eficácia educacional nos ambientes interativos consistem em etapas fora da presente análise.

2 REFERENCIAL TEÓRICO

Um arquivo PDB reúne, entre outros registros, o elemento químico e as coordenadas tridimensionais dos átomos. Essas coordenadas permitem construir diferentes representações visuais. No modelo de esferas e ligações, os átomos são exibidos como esferas e as ligações como segmentos. A escolha da representação depende da tarefa e do tamanho da estrutura. À medida que os conjuntos crescem, o custo de processamento e renderização passa a fazer parte do problema de visualização (Kuťák *et al.*, 2023).

O formato de entrada não determina sozinho como todas as ligações devem ser mostradas. Neste projeto, o conversor aplica uma heurística própria, baseada na distância entre os centros, dos raios covalentes e em limites de valência, que produz uma cena utilizavel e adequada à visualização. Para apoiar a leitura e a inspeção das estruturas, o RDKit oferece recursos abertos de leitura, transformação e verificação de estruturas (Bento *et al.*, 2020). A separação entre leitura, padronização e validação adotada em uma *pipeline* de curadoria do ChEMBL orientou a organização dessas etapas no caminho de produção aqui apresentado.

O processamento numérico foi organizado com *arrays* do NumPy, que armazenam coordenadas e permitem operações vetorizadas (Harris *et al.*, 2020). Para evitar a comparação de pares muito distantes para formar uma ligação, foi utilizado o índice espacial cKDTree do SciPy, uma árvore de partição de pontos em várias dimensões. O SciPy reúne algoritmos científicos para Python (Virtanen *et al.*, 2020), e o método *query_pairs* retorna os pares de pontos encontrados dentro de um raio informado (SCIPY COMMUNITY, 2026).

A entrega do resultado exigiu um contrato entre o conversor e o visualizador. O JSON utilizado neste projeto contém metadados, informações dos átomos, quadros de coordenadas e ligações. Um esquema formal verifica a estrutura do documento, enquanto regras semânticas conferem índices, dimensões e relações internas. Essa separação permite rejeitar um objeto incompleto antes de enviá-lo ao cliente. No Godot, requisições pelo *Hypertext Transfer Protocol* (HTTP) recuperam uma revisão publicada e os dados podem ser convertidos em objetos tridimensionais. A documentação do motor descreve a comunicação por *HTTPRequest* e a instância com *MultiMesh* (Godot Engine, 2026a; Godot Engine, 2026b).

3 METODOLOGIA

O ambiente de desenvolvimento foi o *Windows Subsystem for Linux* 2 (WSL2), com o *Visual Studio Code* como editor e o Python 3.14.4 em um ambiente virtual isolado criado

com o módulo “venv”. O equipamento utilizado contava com processador Intel Core i7-13700 e 8 GB de memória RAM.

Para as bibliotecas e *frameworks*, foram utilizados NumPy 2.5.2 e SciPy 1.18.1 no processamento numérico, RDKit 2026.3.5 na leitura dos arquivos PDB, FastAPI 0.141.1 no serviço web e NiceGUI 3.16.0 na interface; o Godot Engine apoiou a visualização das estruturas e o PyMOL serviu de referência na comparação visual da estrutura 1CRN. A análise dos resultados brutos e a geração das figuras foram realizadas em um notebook, sem executar a aplicação web ou o Godot. As figuras quantitativas usam a paleta *viridis* e a tipografia Times New Roman, com os tempos expressos em milissegundos e as estatísticas calculadas com os tempos originais.

3.1 PIPELINE DE CONVERSÃO

A solução foi dividida em quatro partes. A interface escrita com NiceGUI aceitou um código PDB, uma busca textual ou o envio de um arquivo local. A biblioteca permitiu construir em Python uma interface executada no navegador (NiceGUI, 2026). Quando a busca foi realizada no RCSB PDB, o arquivo recebido foi armazenado com um nome derivado de seu conteúdo. Essa estratégia evitou que duas requisições diferentes sobrescrevessem silenciosamente o mesmo arquivo.

A segunda parte foi uma interface de programação de aplicações (API) escrita com FastAPI, tecnologia voltada à construção de serviços HTTP em Python (FastAPI, 2026). A API criou uma conversão, registrou seu estado e publicou o resultado como uma nova revisão. A gravação da revisão foi atômica. O cliente recebeu apenas um objeto concluído, e não um arquivo parcialmente escrito. Esse mecanismo utilizou uma trava válida dentro do processo e foi delimitado a uma única instância de trabalho do servidor *Uvicorn*. A terceira parte leu o PDB com RDKit, registrou a origem do arquivo e gerou o objeto molecular. No caminho de produção, o RDKit adicionou hidrogênios antes da exportação. A quarta parte foi o cliente Godot, cujo código consulta se existe uma revisão mais recente, baixa o JSON, repete as verificações essenciais e constrói a cena.

3.2 INTERFERÊNCIA DE LIGAÇÕES

No conversor anterior, cada átomo era comparado com todos os átomos seguintes. Denotou se por N o número de átomos e por C(*N*) a quantidade de pares examinados, calculada a partir da Equação 1.

$$C(N) = \binom{N}{2} = \frac{N(N-1)}{2} \quad (1)$$

Essa quantidade cresce de forma quadrática. Com 4.779 átomos, por exemplo, o laço pode visitar 11.414.031 pares antes de aplicar os limites químicos.

A otimização constrói uma cKDTree com as coordenadas e solicita apenas pares dentro do maior raio em que uma ligação poderia ser aceita. A distância entre os átomos de índices i e j foi definida pela Equação 2, na qual x, y e z são as coordenadas cartesianas.

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2} \quad (2)$$

Em ambas as versões, os pares com distância de até 0,4 Å foram descartados. O limite superior foi 1,30 vezes a soma dos raios covalentes simples. As ordens dupla e tripla foram propostas quando a distância não ultrapassou 1,05 vezes a soma dos raios correspondentes, sempre sujeitas ao limite de ligações derivado de *nvalence*. Os valores vieram da tabela local *custom_table.json*, herdada do protótipo.

A mudança ocorreu apenas na seleção dos pares candidatos. No código novo, a comparação inicial usa o quadrado da distância, o que evita calcular a raiz quadrada para pares rejeitados. A cKDTree procura primeiro dentro do maior limite possível e as mesmas regras químicas filtram os candidatos devolvidos.

As Listagens 1 e 2 mostram o núcleo da mudança. Elas foram reduzidas às instruções que distinguem a comparação total da consulta espacial.

Listagem 1 - Comparação total no conversor Tkinter.

```
for i in range(len(atomos)):
        for j in range(i + 1, len(atomos)):
                dist = math.sqrt(
                                (x[i] - x[j]) ** 2
                                + (y[i] - y[j]) ** 2
                                + (z[i] - z[j]) ** 2)
                        )
```

Fonte: Elaborada pelos autores (2026).

Listagem 2 - Consulta espacial no novo conversor.

```
tree = cKDTree(coords)
pares = tree.query_pairs(r=raio_maximo)

for a, b in pares:
        dist_sq = np.sum(
                        (coords[a] - coords[b]) ** 2
        )
```

Fonte: Elaborada pelos autores (2026).

3.3 CONJUNTO DE DADOS E DELINEAMENTO

Para o teste de desempenho, foram selecionadas cinco estruturas do PDB, com tamanhos variados, que serviram como conjunto de dados de *benchmark*. A escolha considerou a diversidade de elementos químicos e a presença de hidrogênios. A Tabela 1 apresenta os arquivos utilizados, com seus respectivos números de registros atômicos.

Tabela 1 - Estruturas utilizadas no *benchmark*.

PDB	ATOM	HETATM	Total atômico
1CRN	327	0	327
1UBQ	602	58	660
1LYZ	1.001	101	1.102
6LU7	2.387	113	2.500
4HHB	4.384	395	4.779

Fonte: Elaborada pelos autores a partir dos arquivos PDB locais (2026).

O teste de desempenho concentrou-se estritamente na etapa de inferência de ligações. Dessa forma, custos computacionais referentes à rede para download e comunicação HTTP, ao pré-processamento químico e geométrico na adição de hidrogênios e centralização e à apresentação de serialização e renderização foram isolados e não compõem as métricas de tempo. As duas implementações receberam os mesmos símbolos, coordenadas e tabela periódica. Para cada combinação entre estrutura e implementação foi feita uma execução de aquecimento, seguida por dez repetições válidas. A ordem entre as versões foi alternada. O relógio *time.perf_counter_ns*() registrou cada observação. Na Equação 3, N_{PDB} representa as cinco estruturas, N_{impl} as duas implementações, N_{rep} para as dez repetições e N_{obs} o total de medições.

$$N_{obs} = N_{PDB} \times N_{impl} \times N_{rep} = 5 \times 2 \times 10 = 100 \quad (3)$$

Os dez aquecimentos não entraram na estatística. Antes da medição, um teste de equivalência comparou o mapa normalizado de pares e ordens de ligação produzido por cada versão. Uma repetição cronometrada foi considerada bem-sucedida quando para calcular ligações terminou sem erro e seu mapa normalizado correspondeu à referência. Fora dos laços cronometrados, foram construídos e validados dez objetos completos, um por PDB e implementação, pelo esquema JSON e pelas regras semânticas do *backend*.

A mediana foi escolhida como medida principal porque sofre menor influência de uma execução ocasionalmente lenta. Na Equação 4, $t_{(5)}$ e $t_{(6)}$ representam o quinto e o sexto valores após a ordenação dos dez tempos, e $\tilde{t}$ representa a mediana.

$$\tilde{t} = \frac{t_{(5)} + t_{(6)}}{2} \quad (4)$$

A dispersão da metade central das observações foi expressa pelo intervalo interquartil da Equação 5. Q1 e Q3 representam os percentis 25 e 75, calculados pelo NumPy com interpolação linear. O intervalo interquartil, indicado por *IIQ*, permite mostrar variabilidade sem fazer o resultado depender do mínimo ou do máximo.

$$IIQ = Q_3 - Q_1 \quad (5)$$

A aceleração relativa A foi calculada pela Equação 6, na qual t_{antigo} e $t_{cKDTree}$ são as medianas das duas versões. Um valor maior que 1 indica que a mediana do método antigo foi maior.

$$A = \frac{\tilde{t}_{antigo}}{\tilde{t}_{cKDTree}} \quad (6)$$

A taxa de sucesso TS da Equação 7 relaciona o número de sucessos $N_{sucessos}$ ao total de tentativas $N_{tentativas}$, segundo o critério definido para cada teste.

$$TS = \frac{100 \times N_{sucessos}}{N_{tentativas}} \quad (7)$$

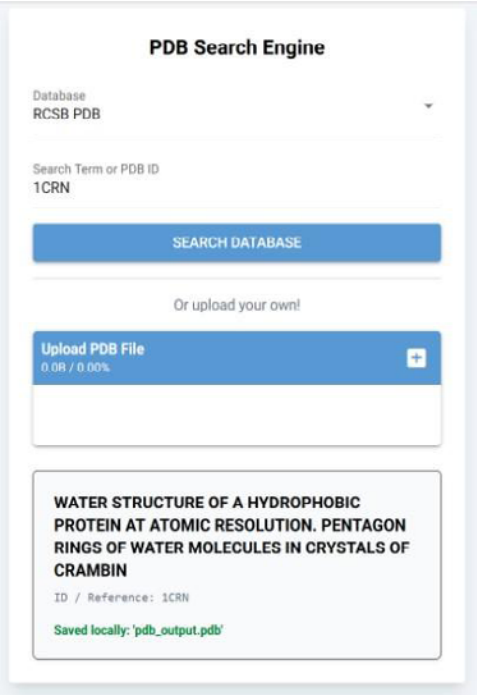
Além do *benchmark*, foi executado um teste funcional local do caminho de produção. Cada um dos cinco PDBs foi enviado uma vez ao *endpoint* de conversão por meio do cliente de testes do FastAPI. O armazenamento foi temporário, a rede externa ficou fora do teste e a adição de hidrogênios permaneceu ativa. O sucesso exigiu resposta HTTP 201 e publicação de um objeto validado. Como houve apenas uma tentativa por arquivo, esse teste descreve compatibilidade funcional e não estabilidade estatística nem tempo de execução.

4 RESULTADOS E DISCUSSÃO

4.1 INTEGRAÇÃO E PRESERVAÇÃO DA SAÍDA

A interface reuniu a busca no RCSB PDB e o envio de um arquivo local em uma única página. A Figura 1 mostra a recuperação da estrutura 1CRN em uma revisão anterior.

Figura 1 - Busca da estrutura 1CRN na interface web.



Fonte: Elaborado pelos autores (2026).

O fluxo descrito pelo código reduz as cópias manuais do protótipo Tkinter. A interface obtém ou recebe o PDB, a API inicia a conversão, o *backend* valida o objeto e publica uma revisão, e o cliente Godot contém a lógica para recuperar essa revisão. A publicação imutável permite identificar o resultado solicitado e comprova a organização do código, bem como a coerência do fluxo entre a interface e o Godot.

O teste funcional local do endpoint de produção com hidrogenação ativa mostrou que três dos cinco arquivos foram processados e publicados. A Tabela 2 apresenta os códigos PDB, os códigos de resposta HTTP, o número de átomos processados e o número de ligações inferidas.

Tabela 2 - Resultado funcional do endpoint de produção com hidrogenação ativa.

PDB	HTTP	Átomos processados	Ligações
1CRN	201	642	651
1UBQ	201	1.405	1.353
1LYZ	422	-	-
6LU7	201	5.050	5.020
4HHB	422	-	-

Fonte: Elaborada pelos autores (2026).

Três dos cinco arquivos foram processados e publicados, o que corresponde a 60% nesse teste descritivo. O RDKit rejeitou 1LYZ e 4HHB durante a leitura, com diagnósticos de valência explícita incompatível para átomos de carbono. O resultado mostra que a otimização da inferência não resolve, por si só, a compatibilidade do caminho de produção.

Os doze testes automatizados foram executados com sucesso. Dez deles verificaram, para cada PDB, a estabilidade da entrada e a igualdade dos mapas de ligação. Os dois restantes verificaram a preservação das observações brutas e o compartilhamento da mesma tabela periódica. Nas cinco verificações de equivalência, as versões antiga e nova produziram o mesmo conjunto de pares e as mesmas ordens de ligação. A Tabela 3 reúne os tempos, as ligações e as execuções válidas do *benchmark* isolado.

Tabela 3 - Medianas e execuções válidas do *benchmark* isolado de ligações.

PDB	Ligações	Antigo (ms)	cKDTree (ms)	Aceleração	Válidas
1CRN	337	18,58	2,52	7,39	20/20
1UBQ	608	76,77	5,00	15,37	20/20
1LYZ	1.033	202,84	7,66	26,46	20/20
6LU7	2.472	1.034,44	18,72	55,25	20/20
4HHB	4.593	4.062,82	45,08	90,13	20/20

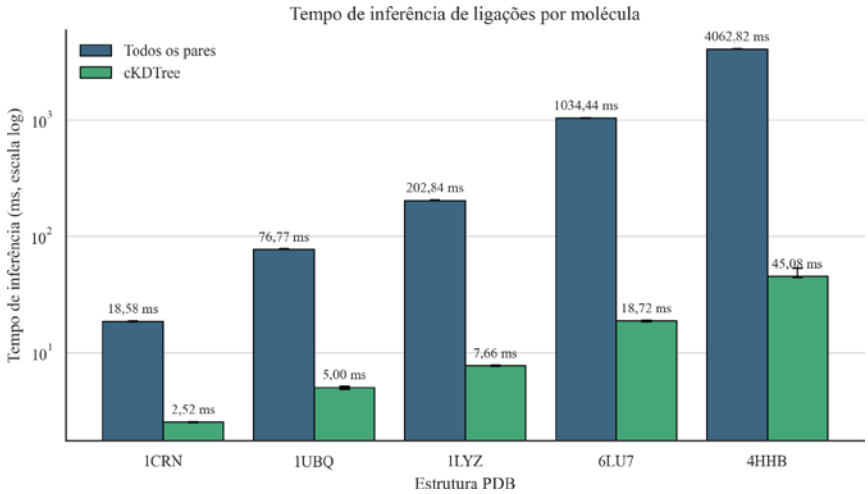
Fonte: Elaborada pelos autores (2026).

A taxa de sucesso foi de 100% nas 100 execuções cronometradas da função de inferência. Em cada linha da tabela, 20/20 representa dez repetições de cada implementação, o que demonstra a estabilidade da etapa avaliada nas entradas consideradas.

4.2 TEMPO E CRESCIMENTO OBSERVADO

A Figura 2 compara o tempo das duas implementações. Cada barra mostra a mediana das dez repetições e cada haste mostra o intervalo interquartil. O eixo vertical é logarítmico porque o menor e o maior tempo mediano diferem por aproximadamente 1,6 mil vezes. Em escala linear, os valores da versão otimizada ficariam comprimidos próximos de zero.

Figura 2 - Mediana e intervalo interquartil do tempo de inferência, em escala logarítmica.

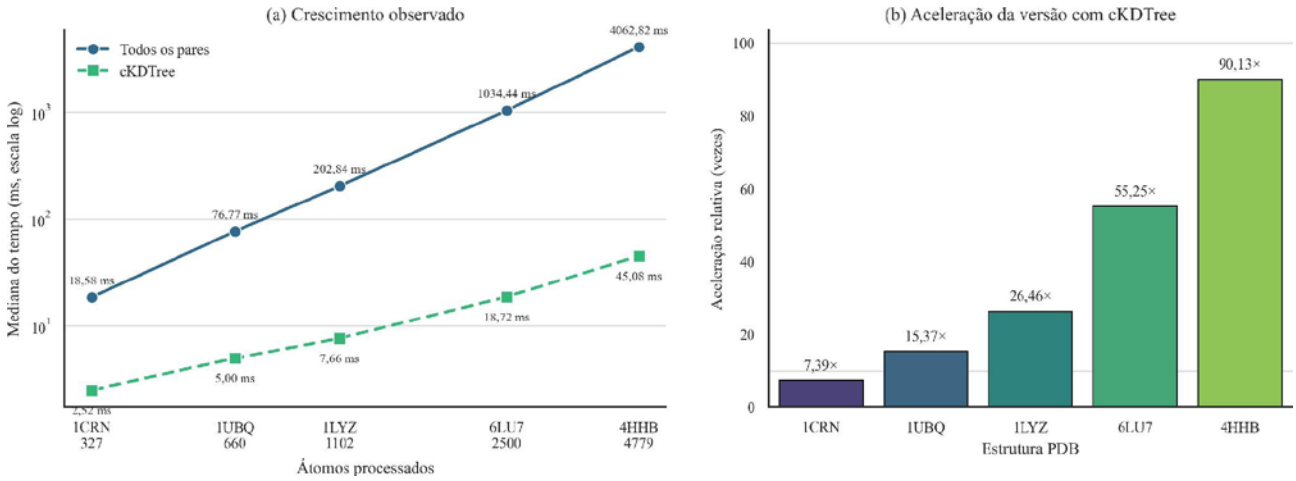


Fonte: Elaborada pelos autores (2026).

A versão com cKDTree foi mais rápida nas cinco estruturas. No conversor antigo, a mediana aumentou de 18,58 ms para a 1CRN até 4.062,82 ms para a 4HHB. Na versão otimizada, os valores correspondentes foram 2,52 ms e 45,08 ms. Para a maior entrada, a etapa avaliada caiu de aproximadamente 4,06 s para 45 ms.

A Figura 3 relaciona o número de átomos ao tempo mediano e mostra a aceleração relativa. As escalas logarítmicas do painel (a) facilitam a leitura do crescimento multiplicativo. O painel (b) usa escala linear para mostrar quantas vezes a mediana da versão otimizada foi menor.

Figura 3 - Tempo mediano em escala logarítmica no painel (a) e aceleração no painel (b).



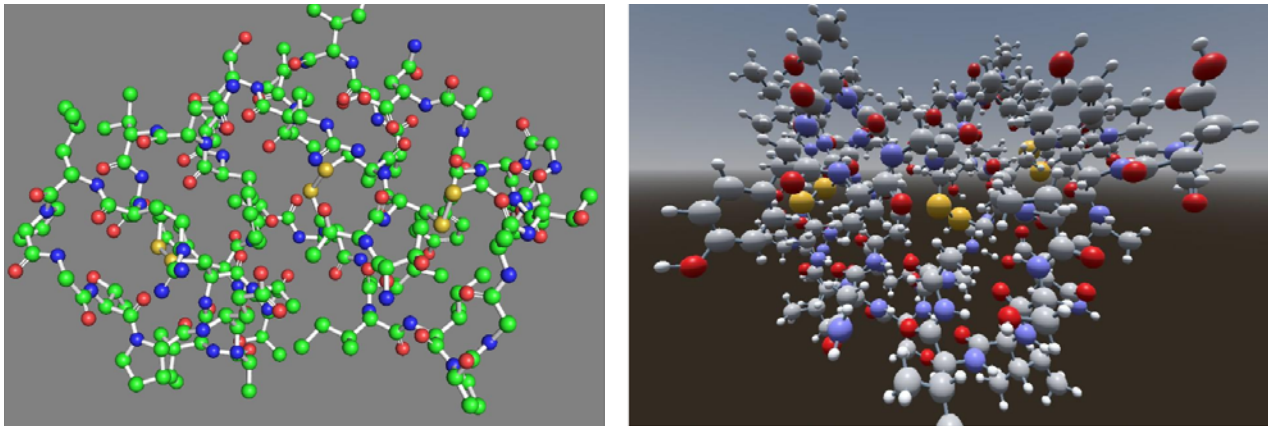
Fonte: Elaborada pelos autores (2026).

Entre a menor e a maior estrutura, o número de átomos aumentou cerca de 14,6 vezes. Nesse intervalo, a mediana do conversor antigo cresceu aproximadamente 218,7 vezes, enquanto a mediana da versão otimizada cresceu 17,9 vezes. A aceleração passou de 7,39 vezes para 90,13 vezes. A vantagem aumentou com o tamanho das entradas avaliadas porque a árvore descartou pares distantes antes da aplicação das regras de ligação.

4.3 RENDERIZAÇÃO E LIMITES DA COMPARAÇÃO

A Figura 4 compara a visualização da estrutura 1CRN no Godot com a mesma estrutura renderizada no PyMOL, sistema de visualização molecular com base de código aberto consolidado na área (SCHRÖDINGER, 2026). Para visualizar e comparar a efetividade e a precisão da conversão, a cena gerada no Godot foi contrastada com a referência profissional. As duas representações exibem as mesmas posições atômicas, elementos e ligações, reproduzindo a organização tridimensional esperada; as diferenças observadas restringem-se ao enquadramento e à coloração.

Figura 4 - Comparação da estrutura 1CRN renderizada no PyMOL e no Godot.



(a) PyMol (b) Godot

Fonte: Elaborada pelos autores (2026).

A renderização produzida pelo Godot evidencia que o pipeline utilizou posições, elementos e ligações para reconstruir a cena tridimensional. A correspondência visual com o PyMOL, um visualizador profissional, indica que a conversão preserva a estrutura molecular e que o cliente Godot apresenta o objeto com a precisão necessária para a análise. Dessa forma, a comparação lado a lado permite avaliar a efetividade da conversão e a fidelidade da representação em relação a uma ferramenta consolidada.

Os resultados quantitativos e visuais indicam que a otimização da inferência de ligações reduziu significativamente o tempo de processamento sem alterar a saída. No entanto, a rejeição de dois arquivos pelo RDKit evidencia que a compatibilidade do caminho de produção deve ser avaliada separadamente do desempenho. A comparação visual com o PyMOL confirma a precisão da representação no Godot, mas não substitui testes químicos formais.

5 CONCLUSÃO

O trabalho organizou a evolução de um conversor manual para uma *pipeline* com busca ou envio de PDB, conversão, validação, publicação por revisões e um cliente Godot. A principal modificação voltada ao desempenho substituiu a comparação de todos os pares de átomos por uma busca espacial com cKDTree, preservando as regras de ligação do protótipo anterior.

Nos cinco arquivos avaliados, a versão otimizada apresentou medianas menores em todas as estruturas. A aceleração variou de 7,39 a 90,13 vezes e aumentou com o número de átomos. As 100 execuções cronometradas da função de inferência terminaram com sucesso, e os mapas de ligações das duas versões foram idênticos. Assim, dentro deste conjunto e do escopo isolado, a busca espacial reduziu o gargalo sem alterar a saída anterior.

No teste funcional separado, o caminho de produção publicou três das cinco entradas. As duas rejeições no RDKit mostram que desempenho e compatibilidade devem ser avaliados separadamente.

A comparação visual da estrutura 1CRN entre o Godot e o PyMOL indicou que a renderização preserva posições atômicas, elementos e ligações, sustentando a precisão da representação em relação a uma ferramenta profissional. Como trabalhos futuros, projeta-se o tratamento das entradas rejeitadas pela biblioteca RDKit e a migração definitiva do cliente em realidade aumentada (AR) para a arquitetura baseada em revisões. Além disso, planeja-se validar o carregamento deste JSON otimizado diretamente em dispositivos de RV como o Meta Quest 3, onde a eficiência computacional é ainda mais crítica para manter a taxa de quadros e a estabilidade visual. Por fim, a saída gráfica gerada deverá ser avaliada contra uma referência química controlada e um teste completo separado por etapas deverá ser executado. Essas extensões permitirão avaliar tanto a velocidade do conversor quanto a confiabilidade da experiência desde a busca até a visualização interativa.

REFERÊNCIAS

BENTO, A. P. *et al.* **An open source chemical structure curation pipeline using RDKit**. Journal of Cheminformatics, Springer Science and Business Media LLC, v. 12, n. 1, p. 51, 2020. Disponível em: https://doi.org/10.1186/s13321-020-00456-1.

BROWN, C. E. *et al.* **Visualizing molecular structures and shapes: a comparison of virtual reality, computer simulation, and traditional modeling**. Chemistry Teacher International, Walter de Gruyter GmbH, v. 3, n. 1, p. 69-80, 2021. Disponível em: https://doi.org/10.1515/cti-2019-0009.

BURLEY, S. K. *et al.* **RCSB Protein Data Bank: powerful new tools for exploring 3D structures of biological macromolecules for basic and applied research and education in fundamental biology, biomedicine, biotechnology, bioengineering and energy sciences**. Nucleic Acids Research, Oxford University Press, v. 49, n. D1, p. D437-D451, 2021. Disponível em: https://doi.org/10.1093/nar/gkaa1038.

FastAPI. **FastAPI Tutorial: First Steps**. [S.l.], 2026. Versão 0.141.1 utilizada no ensaio funcional. Disponível em: https://fastapi.tiangolo.com/tutorial/first-steps/. Acesso em: 29 ago. 2026.

GLOTZER, S. C.; SOLOMON, M. J. **Anisotropy of building blocks and their assembly into complex structures**. Nature Materials, Nature Portfolio, v. 6, n. 8, p. 557-562, 2007. Disponível em: https://doi.org/10.1038/nmat1949.

Godot Engine. **HTTPRequest**. [S.l.], 2026. Documentação da versão 4.5. Disponível em: https://docs.godotengine.org/en/4.5/classes/class_httprequest.html. Acesso em: 29 ago. 2026.

Godot Engine. **Optimization Using MultiMeshes**. [S.l.], 2026. Documentação da versão 4.5. Disponível em: https://docs.godotengine.org/en/4.5/tutorials/performance/using_multimesh.html. Acesso em: 29 ago. 2026.

HARRIS, C. R. *et al.* **Array programming with NumPy**. Nature, Springer Science and Business Media LLC, v. 585, n. 7825, p. 357-362, 2020. Disponível em: https://doi.org/10.1038/s41586-020-2649-2.

KUŤÁK, D. *et al.* **State of the art of molecular visualization in immersive virtual environments**. Computer Graphics Forum, Wiley, v. 42, n. 6, p. e14738, 2023. Disponível em: https://doi.org/10.1111/cgf.14738.

NiceGUI. **NiceGUI Documentation**. [S.l.], 2026. Documentação correspondente à versão 3.16.0 declarada no projeto. Disponível em: https://nicegui.io/documentation/. Acesso em: 29 ago. 2026.

PETTERSEN, E. F. *et al.* **UCSF ChimeraX: Structure visualization for researchers, educators, and developers**. Protein Science, Wiley, v. 30, n. 1, p. 70-82, 2021. Disponível em: https://doi.org/10.1002/pro.3943.

RODRÍGUEZ, F. C.; PERARO, M. D.; ABRIATA, L. A. **Online tools to easily build virtual molecular models for display in augmented and virtual reality on the web.** Journal of Molecular Graphics and Modelling, Elsevier BV, v. 114, p. 108164, 2022. Disponível em: https://doi.org/10.1016/j.jmgm.2022.108164.

SCHLICK, T. **Molecular Modeling and Simulation: An Interdisciplinary Guide**. New York: Springer, 2010. (Interdisciplinary Applied Mathematics). Disponível em: https://doi.org/10.1007/978-1-4419-6351-2.

SCHRÖDINGER. **PyMOL: molecular visualization system**. [S.l.], 2026. Site da versão 3.1.8. Disponível em: https://pymol.org/index.php. Acesso em: 29 ago. 2026.

SCIPY COMMUNITY. **scipy.spatial.cKDTree.query_pairs**. [S.l.], 2026. Disponível em: https://docs.scipy.org/doc/scipy/reference/generated/scipy.spatial.cKDTree.query_pairs.html. Acesso em: 29 ago. 2026.

VIRTANEN, P. *et al.* **SciPy 1.0: fundamental algorithms for scientific computing in python**. Nature Methods, Springer Science and Business Media LLC, v. 17, n. 3, p. 261-272, 2020. Disponível em: https://doi.org/10.1038/s41592-019-0686-2.

DA NARRATIVA AO JOGO: O PROCESSO DE DESENVOLVIMENTO DE *RESCUE HORSE*

FROM NARRATIVE TO GAME: THE DEVELOPMENT PROCESS OF RESCUE HORSE

Guilherme Freitas Cardoso[1], Mateus Nedel dos Reis[2], Lucas Severo Portes[3] e Graziela Frainer Knoll[4]

RESUMO

Este artigo descreve o processo de desenvolvimento do jogo digital *Rescue Horse*, elaborado como atividade avaliativa da disciplina de Narrativa e Roteirização. A proposta parte da compreensão dos jogos digitais como artefatos interativos e multimodais, capazes de articular narrativa, regras, desafios, elementos visuais e participação do jogador. Fundamentado nas contribuições de Rojo e Moura (2012) sobre multiletramentos, de Salen e Zimmerman (2004) sobre design de jogos, e de Jenkins (2009) e Prensky (2001) acerca da cultura digital, da convergência e da interatividade, o estudo compreende o desenvolvimento do jogo como um processo de criação narrativa associado às especificidades da linguagem dos jogos digitais. Nesse sentido, são apresentadas as etapas de concepção, planejamento, criação e implementação de *Rescue Horse*, destacando a articulação entre narrativa e design na construção da experiência do jogador. Os resultados evidenciam a importância da integração entre os elementos narrativos e interativos para a criação de uma experiência digital coerente, participativa e significativa.

Palavras-chave: jogos digitais; multiletramentos; roteiro; interação.

ABSTRACT

This article presents the development process of the digital game Rescue Horse, created as an assessment activity for the Narrative and Screenwriting course. The proposal is based on an understanding of digital games as interactive and multimodal artifacts capable of integrating narrative, rules, challenges, visual elements, and player participation. Grounded in the contributions of Rojo

1 Acadêmico de Jogos Digitais - Universidade Franciscana. E-mail: guilherme.freitas@ufn.edu.br
2 Acadêmico de Jogos Digitais - Universidade Franciscana. E-mail: lucas.portes@ufn.edu.br
3 Acadêmico de Jogos Digitais - Universidade Franciscana. E-mail: m.nedel@ufn.edu.br
4 Professora Adjunta - Universidade Franciscana. E-mail: grazi.fknoll@gmail.com. ORCID: https://orcid.org/0000-0002-6014-2188

and Moura (2012) on multiliteracies, Salen and Zimmerman (2004) on game design, and Jenkins (2009) and Prensky (2001) on digital culture, convergence, and interactivity, this study understands game development as a narrative creation process associated with the specific characteristics of digital game language. In this regard, the stages of conception, planning, creation, and implementation of Rescue Horse are presented, highlighting the articulation between narrative and design in the construction of the player's experience. The results demonstrate the importance of integrating narrative and interactive elements to create a coherent, participatory, and meaningful digital experience.

Keywords: *digital games; multiliteracies; screenwriting; interaction.*

1 INTRODUÇÃO

A crescente presença das tecnologias digitais no cotidiano tem provocado transformações significativas nas formas de interação, comunicação e aprendizagem. Nesse contexto, os jogos digitais destacam-se não apenas como produtos de entretenimento, mas também como recursos capazes de promover experiências interativas, estimular a resolução de problemas e favorecer a construção de conhecimentos. Sua inserção no campo educacional amplia as possibilidades de desenvolvimento de práticas pedagógicas mais dinâmicas, participativas e alinhadas às experiências culturais dos estudantes na sociedade contemporânea.

Em disciplinas do curso de Jogos Digitais, o desenvolvimento de jogos constitui uma possibilidade de criação de experiências educativas intencionalmente planejadas. Esse processo envolve a articulação entre narrativa, desafios, regras, elementos visuais, interatividade e objetivos de aprendizagem, exigindo que os aspectos tecnológicos estejam integrados às finalidades propostas. Dessa maneira, pensar o jogo como objeto de desenvolvimento implica compreender que suas mecânicas e recursos podem contribuir para a construção de situações nas quais o jogador assume um papel ativo diante dos desafios propostos.

A perspectiva dos multiletramentos (Rojo; Moura, 2012) também contribui para compreender o potencial dos jogos digitais, uma vez que esses artefatos mobilizam diferentes linguagens e modos de significação, articulando textos verbais, imagens, sons, movimentos, interfaces e ações do jogador. Assim, a experiência proporcionada pelo jogo ultrapassa a leitura e a escrita em seu sentido convencional, demandando a interpretação e a produção de sentidos em ambientes multimodais e interativos.

A compreensão do jogo como uma experiência construída por meio da interação é fundamental para o desenvolvimento de projetos dessa natureza. Para Salen e Zimmerman (2004), o design de jogos está relacionado à criação de sistemas de regras e relações que produzem experiências significativas para os jogadores. Desse modo, o *designer* de jogos não

desenvolve apenas elementos visuais ou uma sequência de acontecimentos, mas estrutura as possibilidades de ação e interação que constituirão a experiência do jogador. Elementos como regras, objetivos, desafios, escolhas e recompensas precisam estar articulados de maneira coerente, contribuindo para que o jogador participe ativamente da experiência proposta.

Nesse cenário, o jogo apresentado neste artigo parte da compreensão de que as tecnologias podem atuar como mediadoras de processos de aprendizagem e de produção de conhecimentos. *Rescue Horse* foi concebido a partir de uma proposta de avaliação da disciplina de Narrativa e Roteirização, com foco na criação de uma narrativa original. Diante disso, este artigo tem como objetivo descrever o processo de desenvolvimento do jogo digital *Rescue Horse*, considerando suas etapas de concepção, planejamento, criação e implementação, caracterizando-se como um relato de experiência.

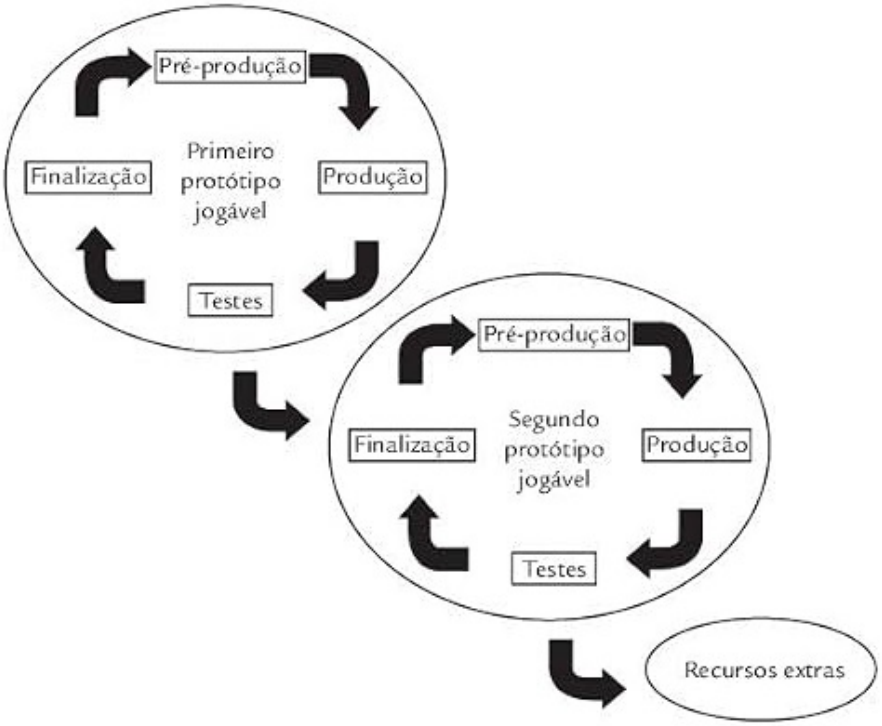
Compreende-se que o desenvolvimento de jogos digitais tem sido impulsionado pelas transformações nas formas contemporâneas de narrar e interagir com conteúdos. Jenkins (2009) destaca que a cultura da convergência amplia as possibilidades de circulação e construção de narrativas, favorecendo experiências nas quais diferentes mídias, linguagens e formas de participação se articulam. Nos jogos digitais, essa perspectiva manifesta-se na possibilidade de o jogador não apenas acompanhar uma história, mas interagir com seus elementos, tomar decisões e participar da construção da experiência narrativa. Prensky (2001), por sua vez, chama atenção para a familiaridade das novas gerações com ambientes digitais e jogos, ressaltando características como interatividade, experimentação, desafio e participação. A partir dessas contribuições, o jogo desenvolvido na disciplina de Narrativa e Roteirização foi concebido como uma experiência que articula os princípios da construção narrativa às especificidades da linguagem dos jogos digitais, explorando a interatividade e a participação do jogador como elementos constitutivos da experiência.

No desenvolvimento de jogos digitais, a construção narrativa está diretamente relacionada à elaboração dos personagens, uma vez que estes contribuem para estabelecer conflitos, objetivos, motivações e vínculos afetivos capazes de envolver o jogador na experiência interativa. Para Sheldon (2017), a criação de personagens e histórias constitui um dos elementos fundamentais da escrita para games, exigindo que os personagens sejam concebidos não apenas como componentes da narrativa, mas como elementos integrados às situações e possibilidades próprias do jogo. Nesse sentido, a narrativa dos games diferencia-se de outras formas de contar histórias por articular personagens, enredo e acontecimentos às ações do jogador, fazendo com que a experiência narrativa seja construída também a partir da interação. Assim, o desenvolvimento de personagens coerentes e de um enredo capaz de sustentar desafios e conflitos contribui para ampliar a imersão e atribuir sentido às ações realizadas durante o jogo.

2 ASPECTOS METODOLÓGICOS

Para o desenvolvimento do jogo, adotou-se como referência a metodologia de ciclos de produção proposta por Chandler (2009), compreendendo o desenvolvimento de jogos como um processo iterativo, no qual as diferentes etapas podem ser retomadas, avaliadas e aperfeiçoadas ao longo do projeto (Figura 1).

Figura 1 - Passos metodológicos



Fonte: Chandler (2009).

No contexto da disciplina de Narrativa e Roteirização, o processo foi adaptado e organizado nas seguintes etapas:

1. Definição da temática: estabelecimento da proposta central do jogo, considerando o gênero, a ambientação, a temática do conflito e a definição inicial dos objetivos da experiência;
2. *Brainstorming*: levantamento e discussão coletiva entre o grupo de possibilidades para a criação do jogo, contemplando ideias relacionadas à ambientação, aos personagens, aos conflitos, aos objetivos, aos desafios e às formas de interação;
3. Pesquisa e levantamento de referências: busca de referências relacionadas a narrativas, personagens, jogos digitais, elementos visuais, ambientações e mecânicas, com o objetivo de subsidiar as decisões criativas e ampliar as possibilidades de desenvolvimento da proposta;

4. Construção da narrativa: organização e estruturação do enredo, definição do arco narrativo, desenvolvimento dos personagens, estabelecimento do conflito central, elaboração dos objetivos e organização da progressão da experiência do jogador;
5. Desenvolvimento das mecânicas e integração entre narrativa e jogabilidade: definição das ações, desafios, obstáculos e possibilidades de interação do jogador, buscando estabelecer relações entre as características dos personagens, os acontecimentos narrativos e as mecânicas do jogo;
6. Prototipagem e experimentação: implementação das ideias desenvolvidas em uma versão inicial do jogo, permitindo testar a narrativa, as mecânicas, a movimentação, os desafios e a experiência de interação;
7. Revisão e avaliação: análise dos elementos produzidos, considerando a coerência do enredo, a caracterização dos personagens, a funcionalidade das mecânicas, a adequação dos desafios e a articulação entre narrativa, elementos visuais e interatividade;
8. Refinamento: realização de ajustes no roteiro, nos personagens, nas mecânicas e nos demais componentes do jogo a partir das avaliações realizadas, buscando solucionar problemas identificados e aprimorar a experiência do jogador;
9. Implementação da versão final: integração dos elementos desenvolvidos e revisados, resultando na versão final do protótipo de *Rescue Horse*.

Desse modo, embora as etapas sejam apresentadas de forma sequencial para fins de sistematização metodológica, o desenvolvimento ocorreu de maneira não linear, com retomadas e modificações entre as diferentes fases. A pesquisa, a experimentação, a produção, a avaliação e a revisão estiveram articuladas durante todo o processo, caracterizando uma experiência de criação colaborativa e iterativa, conforme a perspectiva de Chandler (2009).

3 RELATO DA PRODUÇÃO DO JOGO

Para a proposta deste projeto, foi idealizado o jogo digital *Rescue Horse*, um protótipo 2D dos gêneros quebra-cabeça (*puzzle*) e narrativo, apresentado em visão lateral e com movimentação contínua organizada em *grid*. A concepção da narrativa de *Rescue Horse* surgiu a partir da combinação das temáticas de “fantasia medieval”, “invasão e retomada” e “salvamento”, definidas e aleatorizadas nas etapas iniciais do desenvolvimento. Buscando afastar-se do clichê de um cavaleiro medieval que invade e retoma um castelo protegido por um dragão com o objetivo de salvar uma princesa, o protagonista do jogo é *Equituus*, um cavalo capaz de soprar energia curativa.

No enredo, *Equituus* é levado à linha de frente de uma guerra na qual uma metrópole próxima está sendo retomada por meio de uma invasão. Em meio ao conflito, o jogador orienta os movimentos de *Equituus* para percorrer o campo de batalha, evitando os soldados inimigos e curando os soldados aliados. Dessa forma, a proposta articula os elementos narrativos às mecânicas de movimentação e interação, colocando o jogador em uma posição ativa diante dos desafios apresentados (Figura 2).

Figura 2 - Tela inicial.



Fonte: elaborado pelos autores.

O projeto foi desenvolvido utilizando o motor de jogos *GameMaker*, escolhido em razão de sua acessibilidade, ampla documentação e eficiência para a criação de jogos digitais 2D. Para a composição sonora, foram selecionadas músicas e efeitos sonoros disponíveis em bibliotecas gratuitas, como o *site Freesound* e a Biblioteca de Áudio do *YouTube Studio*, utilizados como recursos complementares à construção da experiência do jogo (Figura 3).

Figura 3 - Narrativa.

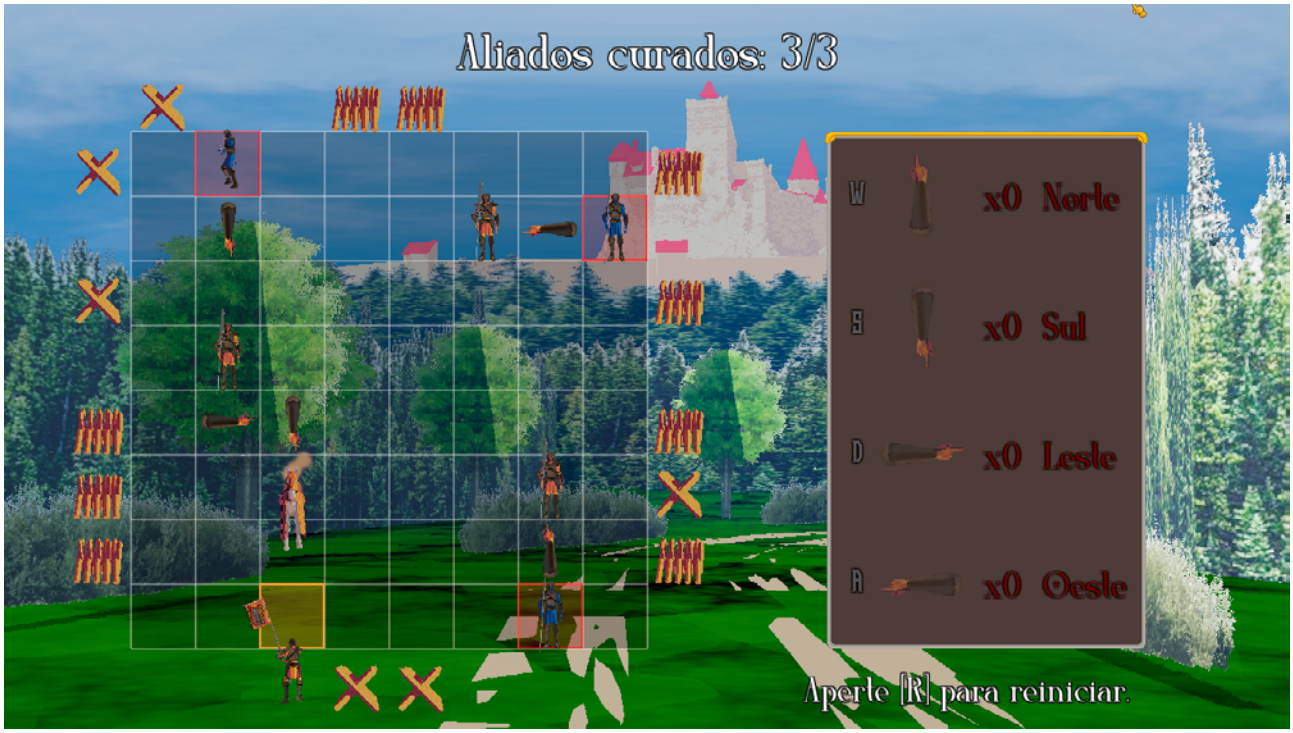


Fonte: elaborado pelos autores.

Durante o desenvolvimento, foram realizadas etapas de planejamento e experimentação voltadas à definição da estrutura do jogo, considerando aspectos como ambientação, personagens, objetivos, desafios e possibilidades de interação. A elaboração do protótipo partiu inicialmente da organização das ideias narrativas e da definição do papel desempenhado pelo protagonista, para, posteriormente, estabelecer de que maneira essas características poderiam ser traduzidas em ações dentro do jogo. Esse processo permitiu relacionar os acontecimentos do enredo às possibilidades de interação oferecidas ao jogador, buscando garantir que a narrativa não permanecesse apenas como elemento contextual, mas estivesse incorporada à própria experiência de jogo.

Outro aspecto considerado foi a construção progressiva dos desafios apresentados ao jogador. A organização do cenário em *grid* possibilitou estabelecer uma lógica de movimentação e posicionamento dos personagens, fazendo com que o deslocamento de *Equituus* exigisse atenção e planejamento. A presença simultânea de soldados aliados e inimigos contribui para a criação de situações nas quais o jogador precisa avaliar os movimentos disponíveis e escolher estratégias para alcançar os personagens que necessitam de cura. Dessa forma, as mecânicas foram pensadas de modo a reforçar o propósito narrativo do protagonista, estabelecendo uma relação entre a habilidade de cura de *Equituus* e o objetivo de salvamento apresentado pelo enredo (Figura 4).

Figura 4 - Gameplay.



Fonte: elaborado pelos autores.

A produção do protótipo também envolveu a integração entre diferentes recursos necessários à composição da experiência digital, incluindo programação, elementos gráficos, movimentação, efeitos sonoros e organização do cenário. A utilização desses recursos buscou produzir uma experiência simples, porém funcional, capaz de representar as principais características definidas para o projeto. Nesse sentido, o desenvolvimento de *Rescue Horse* foi compreendido como um processo experimental, no qual as decisões relacionadas à narrativa e às mecânicas foram sendo articuladas às possibilidades técnicas oferecidas pelo motor de jogos. O resultado foi um protótipo que materializa a proposta inicial e permite visualizar como uma ideia narrativa pode ser transformada em uma experiência interativa por meio da combinação entre história, desafios, personagens e recursos próprios dos jogos digitais.

As escolhas artísticas do projeto foram baseadas na adoção da perspectiva lateral; dessa forma, o jogo apresenta um estilo de *pixel art*, caracterizado por uma singularidade, uma vez que combina elementos 2D da interface feitos no *Aseprite*, enquanto personagens, objetos, mapas e cenários foram elaborados em 3D, sendo aplicada a técnica "*Pre-rendered 3D sprites*" para convertê-los em *pixel art 2D*.

Com a temática medieval, os elementos foram selecionados para simbolizar exércitos diferentes: os aliados, com coloração laranja, e os inimigos, com cor azul. *Equituus*, com sua pelagem branca, possui o *design* mais destacado, isto é, contrastante em relação ao fundo e às outras imagens, já que é o protagonista.

No contexto dos jogos digitais, a narrativa pode ser compreendida como um elemento estruturador da experiência, responsável por organizar acontecimentos, personagens, conflitos e objetivos em uma composição que atribui sentidos às ações realizadas pelo jogador. Diferentemente das narrativas lineares, nas quais o leitor acompanha uma sequência previamente estabelecida de acontecimentos, os jogos possibilitam uma relação mais dinâmica entre narrativa e interação, uma vez que o jogador participa da construção da experiência por meio de suas escolhas e ações. Nesse sentido, o enredo não se restringe à sucessão de acontecimentos, mas articula situações de conflito, objetivos e desafios que orientam a progressão do jogador.

Em *Rescue Horse*, essa construção se manifesta a partir da temática de fantasia medieval associada à invasão e ao salvamento, constituindo um enredo no qual o conflito bélico funciona como elemento motivador para as ações do protagonista e para os desafios propostos ao jogador. Assim, narrativa e mecânicas encontram-se integradas, de modo que avançar pelo campo de batalha, evitar inimigos e curar aliados não são apenas ações relacionadas à jogabilidade, mas também formas de vivenciar e desenvolver o enredo.

O personagem, por sua vez, desempenha papel fundamental na materialização da narrativa, pois constitui a instância por meio da qual o jogador estabelece uma relação de identificação e ação dentro do universo ficcional. Em *Rescue Horse*, a criação de *Equituus* representa uma estratégia de ruptura com os modelos recorrentes das narrativas medievais, substituindo a figura tradicional do cavaleiro-herói por um cavalo dotado de uma habilidade específica: o sopro de energia curativa. Essa característica atribui ao personagem uma função narrativa e, simultaneamente, mecânica, uma vez que sua habilidade interfere diretamente na forma como o jogador enfrenta os desafios do jogo. O protagonista, portanto, não é apenas um componente visual ou narrativo, mas constitui o eixo de articulação entre enredo, mecânicas e participação do jogador. Ao controlar *Equituus*, o jogador assume uma posição ativa diante do conflito, sendo responsável por conduzir o personagem pelo cenário, evitar os soldados inimigos e preservar os aliados. Dessa forma, a construção do personagem contribui para transformar a narrativa em uma experiência interativa, na qual as características do protagonista são incorporadas às possibilidades de ação oferecidas pelo sistema do jogo.

4 DISCUSSÃO

A narrativa constitui um dos elementos centrais na construção da experiência proporcionada pelos jogos digitais, especialmente porque, nesse tipo de produto, contar uma história implica também criar condições para que o jogador participe dela. Diferentemente

de narrativas lineares, nas quais o leitor acompanha acontecimentos previamente organizados, os jogos articulam enredo, personagens, conflitos e objetivos às possibilidades de ação oferecidas ao jogador. Nesse sentido, a narrativa não deve ser compreendida como um elemento isolado das mecânicas, mas como uma dimensão que atribui significado às ações realizadas durante a experiência de jogo. A partir das contribuições de Jenkins (2009) e Sheldon (2017), foi possível compreender que a construção narrativa em *games* envolve uma relação dinâmica entre história, personagem, espaço e interação, de modo que os acontecimentos narrados ganham sentido também por meio da participação do jogador. Em *Rescue Horse*, essa articulação manifesta-se na própria trajetória do protagonista, cuja missão de atravessar um cenário marcado pelo conflito e realizar ações de resgate estabelece uma relação direta entre o desenvolvimento do enredo e as possibilidades de interação. Assim, os desafios propostos deixam de constituir obstáculos meramente funcionais e passam a integrar a progressão narrativa, contribuindo para a construção de uma experiência na qual jogar e narrar são processos interligados.

O desenvolvimento experimental de um jogo digital também se mostrou relevante por permitir que a criação seja compreendida como um processo de descoberta, testagem e reelaboração, e não como uma sequência rígida de etapas previamente determinadas. A experiência de produção de *Rescue Horse*, realizada no contexto da disciplina de Narrativa e Roteirização, possibilitou experimentar diferentes possibilidades de construção do universo ficcional, do personagem, do enredo e de sua articulação com as mecânicas. Nesse processo, procedimentos como pesquisa, *brainstorming*, planejamento, prototipagem, revisão e avaliação contribuíram para que as ideias inicialmente concebidas fossem confrontadas com as possibilidades efetivas de implementação. Essa dimensão experimental aproxima-se da compreensão de Salen e Zimmerman (2004) acerca do *design* de jogos como construção de sistemas nos quais regras, ações, objetivos e relações produzem experiências significativas para o usuário.

Desse modo, desenvolver o jogo significou também aprender a partir do próprio processo de criação, identificando possibilidades, limitações e necessidades de reformulação. A experimentação, portanto, teve dimensão formativa, pois permitiu compreender que a autoria de um jogo é construída de maneira iterativa, mediante sucessivos processos de criação, observação, avaliação e aperfeiçoamento. Sob a perspectiva dos multiletramentos, essa experiência torna-se ainda mais relevante, uma vez que envolve exigiu dos desenvolvedores decisões relacionadas à linguagem verbal, à visualidade, à espacialidade, ao movimento e à interatividade.

Nesse percurso de criação, a escolha de um cavalo como personagem protagonista constitui uma decisão estética e narrativa que contribui para o caráter inovador da

proposta. Ao deslocar o protagonismo tradicionalmente associado a personagens humanos, guerreiros ou figuras heroicas para um animal, *Rescue Horse* rompe com determinados lugares-comuns presentes em narrativas de fantasia e aventura e abre possibilidades para a construção de uma identidade própria para o jogo. O personagem *Equituus* não foi concebido apenas como uma representação visual diferenciada, mas como elemento estruturante da narrativa e das mecânicas, uma vez que suas características e habilidades participam diretamente da resolução dos desafios e do desenvolvimento do enredo. A capacidade de produzir energia de cura, por exemplo, estabelece uma relação entre a caracterização do protagonista e sua função narrativa, fazendo com que a habilidade do personagem ultrapasse a dimensão estética e adquira relevância para a jogabilidade. Essa escolha também possibilita ressignificar a figura do herói, afastando-a de representações baseadas exclusivamente na força, no combate ou na conquista e aproximando-a de ações associadas ao cuidado, à proteção e ao resgate.

Assim, a inovação de *Rescue Horse* não reside somente na presença de um cavalo como protagonista, mas na integração entre personagem, narrativa e mecânica, demonstrando como decisões criativas aparentemente simples podem produzir novas possibilidades de significação e experiência no jogo digital. A proposta evidencia, assim, que a experimentação narrativa pode favorecer a criação de personagens não convencionais e, também, ampliar as formas de participação, interpretação e produção de sentidos características dos jogos enquanto artefatos multimodais e interativos.

5 CONCLUSÃO

A construção do enredo de *Rescue Horse* também pode ser compreendida a partir da organização de um arco narrativo, estruturado pela apresentação de uma situação inicial, pela inserção do protagonista em uma situação de conflito e pelo desenvolvimento de ações orientadas à sua resolução. A narrativa parte da introdução de *Equituus* e de sua habilidade singular de cura, conduzindo-o posteriormente ao cenário de guerra, onde a necessidade de salvar os soldados aliados estabelece o conflito central. A progressão do enredo ocorre, portanto, por meio dos desafios enfrentados durante o percurso, fazendo com que o avanço do jogador pelo espaço do jogo corresponda também ao desenvolvimento da narrativa. Essa estrutura evidencia uma característica própria da narrativa nos jogos digitais: o arco narrativo não se desenvolve exclusivamente por meio de acontecimentos previamente apresentados, mas também pela participação do jogador, cujas ações e escolhas contribuem para a progressão da experiência e para a concretização dos objetivos estabelecidos pelo enredo.

O desenvolvimento do jogo digital *Rescue Horse* possibilitou compreender a criação de jogos como um processo que ultrapassa a dimensão técnica e envolve decisões relacionadas à narrativa, ao enredo, aos personagens, às mecânicas e à experiência do jogador. A partir da metodologia de ciclos de produção, foi possível perceber que a construção de uma narrativa para jogos exige a articulação entre os elementos ficcionais e as possibilidades de interação oferecidas pelo sistema. Nesse processo, a concepção de *Equituus* como protagonista contribuiu para romper com representações recorrentes da fantasia medieval, ao mesmo tempo em que sua habilidade de cura foi incorporada às mecânicas do jogo. Dessa forma, narrativa e jogabilidade foram pensadas de maneira integrada, permitindo que as ações realizadas pelo jogador contribuíssem para a construção da experiência narrativa.

A experiência de produção também evidenciou o potencial dos jogos digitais como artefatos multimodais, nos quais diferentes linguagens e recursos tecnológicos participam da construção de sentidos. A combinação entre narrativa, elementos visuais, sonoros, desafios, movimentação e interação demonstrou que o desenvolvimento de um jogo envolve um processo criativo e interdisciplinar, no qual escolhas estéticas e narrativas estão diretamente relacionadas à experiência pretendida. Nesse sentido, *Rescue Horse* constitui não apenas o resultado de uma atividade avaliativa, mas também uma experiência de autoria e experimentação no campo dos jogos digitais, evidenciando a relevância da integração entre narrativa e design para a criação de experiências interativas significativas.

DECLARAÇÃO DE USO DE IA

Neste trabalho, foi utilizado o aplicativo de Inteligência Artificial “ChatGPT-5.6” estritamente para a revisão textual de trechos do artigo.

REFERÊNCIAS

CHANDLER, Heather Maxwell. ***The Game Production Handbook***. 2. ed. Hingham: Jones & Bartlett Learning, 2009.

JENKINS, Henry. **Cultura da convergência**. 2. ed. São Paulo: Aleph, 2009.

PRENSKY, Marc. ***Digital Game-Based Learning***. New York: McGraw-Hill, 2001.

ROJO, Roxane; MOURA, Eduardo (org.). **Multiletramentos na escola**. São Paulo: Parábola Editorial, 2012.

SALEN, Katie; ZIMMERMAN, Eric. ***Rules of Play****: Game Design Fundamentals*. Cambridge: MIT Press, 2004.

SHELDON, Lee. **Desenvolvimento de personagens e narrativas para games**. São Paulo: Cengage Learning, 2017.

DESENVOLVIMENTO DE UM SISTEMA WEB PARA GESTÃO DO ACERVO BIBLIOGRÁFICO DO MUSEU TREZE DE MAIO

DEVELOPMENT OF A WEB SYSTEM FOR THE BIBLIOGRAPHIC COLLECTION MANAGEMENT OF MUSEU TREZE DE MAIO

José Otávio Ribeiro Baggio[1], Guilherme Weber Henriques[2], Eduardo Oliveira da Fontoura[3], Nyx Barros Bento de Freitas[4], Janette Mariano Godois[5], Paula Ribeiro Ciochetto[6] e Herysson Rodrigues Figueiredo[7]

RESUMO

Este trabalho apresenta o desenvolvimento de um sistema web para apoiar a gestão do acervo bibliográfico do Museu Treze de Maio, em Santa Maria, Rio Grande do Sul. O projeto caracteriza-se como desenvolvimento tecnológico de natureza aplicada, em contexto extensionista na Universidade Franciscana. A solução foi construída a partir do levantamento e priorização de requisitos, modelagem em UML e adoção do *Feature-Driven Development*. A aplicação, desenvolvida com Java e Spring Boot, contempla gerenciamento e consulta pública do acervo, autenticação e controle de acesso. A validação extensionista com representantes do museu indicou percepção favorável quanto à interface e à compreensão dos fluxos, apontando possibilidades de aprimoramento. Os resultados demonstram correspondência entre os requisitos e as funcionalidades implementadas, evidenciando a aplicação de Engenharia de Software no atendimento a uma demanda concreta da comunidade.

Palavras-chave: Engenharia de Software; Extensão Universitária; Patrimônio Cultural; Spring Boot.

1 Acadêmico do Curso de Ciência da Computação - Universidade Franciscana - UFN. E-mail: jose.baggio@ufn.edu.br. ORCID: https://orcid.org/0009-0002-4697-9149

2 Acadêmico do Curso de Ciência da Computação - Universidade Franciscana - UFN. E-mail: guilherme.whenriques@ufn.edu.br. ORCID: https://orcid.org/0009-0001-0545-2482

3 Acadêmico do Curso de Ciência da Computação - Universidade Franciscana - UFN. E-mail: e.fontoura@ufn.edu.br. ORCID: https://orcid.org/0009-0007-5745-8206

4 Acadêmico do Curso de Ciência da Computação - Universidade Franciscana - UFN. E-mail: nyx.barros@ufn.edu.br. ORCID: https://orcid.org/0009-0008-1836-225X

5 Bibliotecária, Universidade Franciscana (UFN). E-mail: janette@ufn.edu.br. ORCID: https://orcid.org/0009-0002-5775-3743

6 Professora do Curso de História, Universidade Franciscana (UFN); coordenadora do projeto de extensão: “Território Negro do Rosário”. E-mail: paula.ciochetto@ufn.edu.br. ORCID: https://orcid.org/0000-0001-9949-7712

7 Professor dos Cursos da Área de Computação - Universidade Franciscana - UFN. E-mail: herysson.figueiredo@ufn.edu.br. ORCID: https://orcid.org/0009-0009-2615-9785

ABSTRACT

This paper presents the development of a web system to support the management of the bibliographic collection of Museu Treze de Maio, in Santa Maria, Rio Grande do Sul. The project is characterized as applied technological development, in an extension context at Universidade Franciscana. The solution was built from requirements gathering and prioritization, UML modeling, and adoption of Feature-Driven Development. The application, developed with Java and Spring Boot, includes collection management and public consultation, authentication, and access control. Extension-based validation with museum representatives indicated favorable perception of the interface and understanding of the flows, pointing to possibilities for improvement. The results demonstrate correspondence between requirements and implemented functionalities, evidencing the application of Software Engineering in addressing a concrete community demand.

Keywords: *Software Engineering; University Extension; Cultural Heritage; Spring Boot.*

1 INTRODUÇÃO

As instituições de memória, como museus, arquivos e bibliotecas, desempenham papel relevante na preservação e difusão do patrimônio histórico e cultural. Nesse contexto insere-se o Museu Comunitário Treze de Maio em Santa Maria, Rio Grande do Sul, originado da Sociedade Cultural Ferroviária Treze de Maio, antigo Clube Social Negro, fundado em 1903, período pós-Abolição (Diário de Santa Maria, 2023). Formalizado como museu comunitário em 2004, o espaço preserva registros da cultura afro-brasileira e um acervo bibliográfico composto por livros, jornais e revistas de relevância histórica para a comunidade (Santa Maria, 2013).

Nesse sentido, a gestão do acervo envolve atividades de catalogação, organização, consulta e atualização das informações, e assim a interação com a equipe do museu e com os integrantes do projeto de extensão “Território Negro do Rosário: o Museu Treze de Maio como espaço de educação antirracista, preservação da história e memória”, permitiu identificar dificuldades relacionadas à manutenção dos registros, à recuperação das informações e ao controle das operações, evidenciando a necessidade de uma solução informatizada específica para sua gestão. Destaca-se que este projeto de extensão é vinculado à UFN, e desenvolvido pelo curso de História em parceria com o curso de Ciência da Computação.

As demandas do acervo foram identificadas no contexto das atividades de curricularização da extensão desenvolvidas na disciplina de Projeto de Software, do curso de Ciência da Computação, da UFN. A aproximação entre estudantes e representantes do Museu Treze de Maio possibilitou conhecer os processos utilizados na gestão do acervo, compreender

necessidades da instituição e transformar uma demanda da comunidade em um problema tecnológico a ser trabalhado no ambiente acadêmico. Dessa forma, o projeto articula conhecimentos de Engenharia de Software com uma ação extensionista direcionada ao atendimento de uma necessidade concreta de uma instituição comunitária.

A partir dessa interação, foi proposta a construção de um sistema web destinado à organização e à consulta do acervo bibliográfico, contemplando mecanismos de gerenciamento dos itens, diferentes níveis de acesso aos usuários e recursos destinados à rastreabilidade das operações realizadas no sistema. A solução foi desenvolvida considerando as características e o fluxo de trabalho identificados junto à instituição, buscando proporcionar uma ferramenta adequada ao seu contexto de utilização.

Diante disso, este trabalho, caracterizado como um desenvolvimento tecnológico de natureza aplicada realizado em contexto extensionista, tem como objetivo apresentar o desenvolvimento de um sistema web para apoiar a gestão do acervo bibliográfico do Museu Treze de Maio. Para alcançar esse objetivo, foram realizadas atividades de levantamento das necessidades da instituição, definição e priorização de requisitos, modelagem, implementação das funcionalidades e verificação da solução junto aos representantes do museu e projeto de extensão “Território Negro do Rosário”. Além do produto tecnológico desenvolvido, o trabalho busca demonstrar como a articulação entre universidade e comunidade pode contribuir para a aplicação de conhecimentos de Computação no atendimento de demandas relacionadas à organização, preservação e acesso ao patrimônio cultural.

2 FUNDAMENTAÇÃO TEÓRICA E TRABALHOS CORRELATOS

Esta seção reúne os fundamentos teóricos que sustentam o desenvolvimento do trabalho, abordando a gestão de acervos em instituições de memória e a extensão universitária como contexto para o desenvolvimento de soluções tecnológicas. Na sequência, são apresentados sistemas e trabalhos correlatos relacionados à informatização e à gestão de acervos, permitindo situar a solução proposta em relação a iniciativas existentes.

2.1 GESTÃO DE ACERVOS EM INSTITUIÇÕES DE MEMÓRIA

Instituições de memória, como museus, arquivos e bibliotecas, desempenham papel relevante na preservação, organização e disponibilização de registros associados à história e à cultura das comunidades. No contexto museológico, a documentação e a gestão dos acervos contribuem para preservar as informações relacionadas aos objetos e documentos

sob responsabilidade da instituição, permitindo sua identificação, organização, recuperação e utilização em atividades de pesquisa, educação e difusão cultural (Padilha, 2014).

Entre os materiais que podem integrar esses acervos encontram-se livros, jornais, revistas e outros documentos bibliográficos. A gestão desses itens envolve atividades como registro, catalogação, organização, atualização e recuperação das informações, de modo a possibilitar sua adequada identificação e consulta. Segundo Padilha (2014), a documentação museológica constitui um instrumento importante para o gerenciamento das coleções, uma vez que permite organizar e sistematizar informações referentes aos bens preservados pela instituição.

Os sistemas de informação podem apoiar a gestão dos acervos ao centralizar os registros, facilitar sua recuperação e ampliar o controle sobre as operações realizadas. A informatização também pode reduzir dificuldades decorrentes da manutenção de registros dispersos ou predominantemente manuais, favorecendo a consulta, a atualização e a rastreabilidade das informações armazenadas. Em instituições comunitárias de menor porte, contudo, a adoção dessas soluções deve considerar não apenas as funcionalidades necessárias à gestão do acervo, mas também as características dos processos internos e dos usuários responsáveis por sua manutenção, de modo que o sistema desenvolvido esteja adequado ao contexto institucional.

2.2 A EXTENSÃO UNIVERSITÁRIA NO DESENVOLVIMENTO DE SISTEMAS

A extensão universitária articula ensino, pesquisa e sociedade, proporcionando aos estudantes experiências formativas em contextos reais. Na UFN, essa perspectiva busca integrar conhecimentos acadêmicos, demandas sociais e saberes dos territórios, fortalecendo a interação entre universidade e comunidade (Universidade Franciscana, 2026).

No processo de curricularização, as atividades extensionistas devem considerar demandas concretas das instituições parceiras e ser construídas com a participação da comunidade. Na formação em Computação, essa aproximação possibilita aplicar conhecimentos de Engenharia de Software na construção de soluções tecnológicas orientadas a problemas reais (Universidade Franciscana, 2026). Ritter *et al.* (2025), por exemplo, apresentam experiência semelhante de desenvolvimento de um sistema web em contexto extensionista na UFN.

O projeto apresentado neste trabalho segue essa perspectiva ao envolver estudantes do curso de Ciência da Computação e representantes do Museu Treze de Maio na identificação das necessidades e na definição das características da solução, relacionando formação acadêmica, desenvolvimento tecnológico e demanda comunitária.

2.3 SISTEMAS E TRABALHOS CORRELATOS

Diferentes soluções computacionais são utilizadas para organização, catalogação e disponibilização de acervos, sendo que no contexto bibliotecário, destacam-se sistemas como Koha (Koha Community, 2026) e BibLivre (Biblivre, 2026), enquanto plataformas como Omeka S (Omeka, 2026) e Tainacan (Tainacan, 2026) são direcionadas à organização e disponibilização de coleções digitais e patrimoniais.

Embora essas soluções apresentem recursos capazes de atender a diferentes cenários de gestão da informação, seus objetivos, formas de utilização e possibilidades de customização variam conforme o domínio para o qual foram concebidas (Tabela 1). Sistemas voltados ao contexto bibliotecário apresentam recursos específicos para catalogação e gerenciamento bibliográfico, enquanto plataformas orientadas a coleções digitais e museológicas priorizam a descrição e a disponibilização de objetos e conteúdos patrimoniais.

Tabela 1 - Comparação entre soluções para gestão de acervos

Solução	Contexto principal	Gestão bibliográfica	Coleções digitais / museológicas	Acesso público	Customização
Koha	Bibliotecas	Sim	Não é o foco principal	Sim	Sim
BibLivre	Bibliotecas	Sim	Não é o foco principal	Sim	Sim
Omeka S	Acervos e coleções digitais	Parcial	Sim	Sim	Sim
Tainacan	Repositórios e coleções digitais	Parcial	Sim	Sim	Sim

Fonte: Construção do Autor com base em Koha community, 2026; BibLivre, 2026; Omeka, 2026; Tainacan, 2026.

A análise dessas soluções demonstra a existência de diferentes alternativas para informatização de acervos. O sistema desenvolvido neste trabalho não tem como objetivo substituir ou estabelecer superioridade em relação a essas plataformas, mas atender aos requisitos específicos identificados durante a interação com o Museu Treze de Maio. Dessa forma, sua contribuição está relacionada principalmente à construção de uma solução tecnológica adaptada ao fluxo de trabalho e às necessidades identificadas junto à instituição no contexto da atividade extensionista.

3 METODOLOGIA

O presente trabalho caracteriza-se como um desenvolvimento tecnológico de natureza aplicada, realizado em contexto extensionista, tendo como finalidade a construção de

uma solução computacional para apoiar a gestão do acervo bibliográfico do Museu Treze de Maio. O projeto foi desenvolvido ao longo de um semestre letivo, no âmbito das atividades de curricularização da extensão da disciplina de Projeto de Software do curso de Ciência da Computação da UFN, em parceria com a instituição museal.

3.1 LEVANTAMENTO DAS NECESSIDADES E REQUISITOS

O levantamento das necessidades ocorreu por meio da interação direta com dois a três representantes do Museu Treze de Maio, buscando compreender os processos relacionados à organização e ao gerenciamento do acervo bibliográfico. Para isso, foram realizadas de uma a duas reuniões/visitas técnicas com os responsáveis pela instituição, durante as quais foram observadas as práticas utilizadas para controle do acervo, as necessidades relacionadas à rotina de trabalho e as dificuldades existentes na organização e recuperação das informações.

Complementarmente, foi realizada uma visita técnica à Biblioteca da Universidade Franciscana, com o objetivo de conhecer práticas adotadas na gestão de acervos bibliográficos, incluindo procedimentos de catalogação, classificação e identificação de livros, jornais e revistas. As informações obtidas nessas atividades subsidiaram a identificação das regras de negócio e a definição dos requisitos funcionais e não funcionais da aplicação.

Os requisitos funcionais foram utilizados para especificar as funcionalidades necessárias ao atendimento das demandas da instituição, enquanto os requisitos não funcionais contemplaram características relacionadas à qualidade da solução, como usabilidade, desempenho e confiabilidade. A partir desse levantamento, foi elaborado o documento de requisitos, utilizado como referência para as etapas posteriores de modelagem e desenvolvimento.

3.2 PROCESSO DE DESENVOLVIMENTO DO SISTEMA

Para orientar o processo de desenvolvimento da aplicação, foi adotado o *Feature-Driven Development* (FDD), ou Desenvolvimento Guiado por Funcionalidades. Segundo Pressman (2021), o FDD é um modelo de processo ágil orientado a objetos que organiza o desenvolvimento em pequenas funcionalidades, cada uma representando valor para o cliente, favorecendo a colaboração entre os membros da equipe e a realização de entregas incrementais.

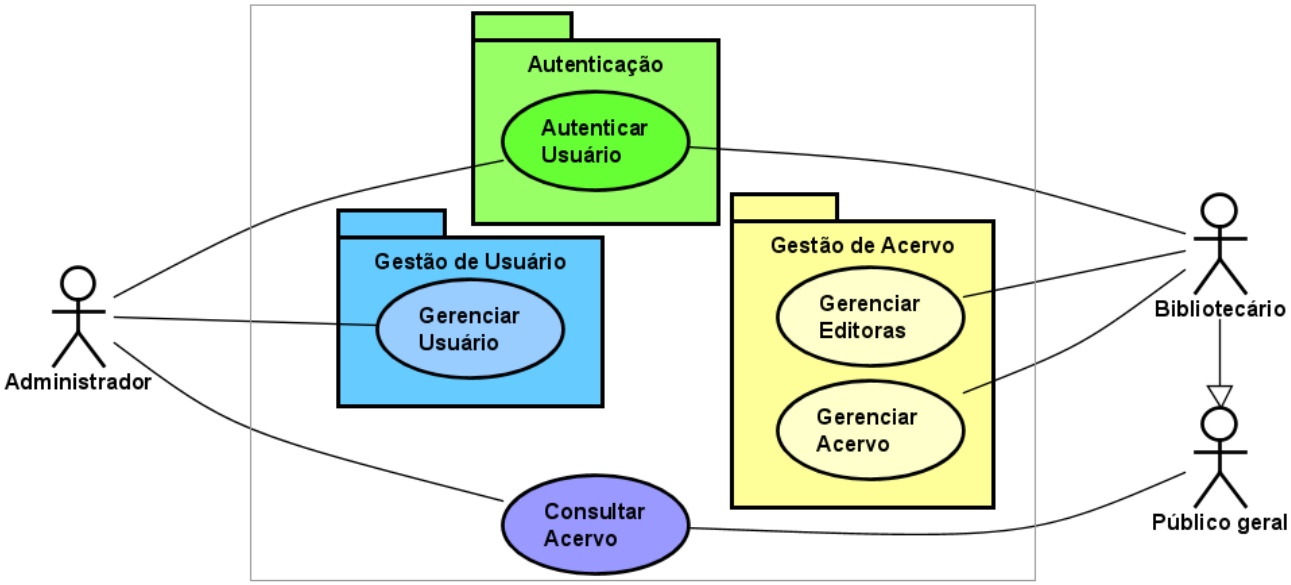
O processo foi organizado considerando as cinco etapas do FDD descritas por Pressman (2021): desenvolvimento de um modelo geral, elaboração da lista de funcionalidades, planejamento por funcionalidade, projeto por funcionalidade e construção por funcionalidade. No contexto deste trabalho, o levantamento das necessidades e a modelagem do

domínio subsidiaram a definição das funcionalidades do sistema, que posteriormente foram priorizadas, projetadas e implementadas de forma incremental.

3.3 MODELAGEM E PRIORIZAÇÃO DAS FUNCIONALIDADES

A modelagem do sistema foi realizada utilizando a Linguagem de Modelagem Unificada (UML), com o auxílio da ferramenta Astah UML. Inicialmente, foi elaborado o diagrama de casos de uso (Figura 1) para representar os atores e as principais funcionalidades da aplicação. Os casos de uso foram posteriormente detalhados por meio da descrição dos atores envolvidos, pré-condições, pós-condições e fluxos de interação.

Figura 1 - Diagrama de Casos de Uso do Sistema.



Fonte: Construção do Autor.

A partir dos casos de uso identificados, foi elaborada a lista de funcionalidades a serem implementadas. O acompanhamento das atividades de desenvolvimento foi realizado com o auxílio da ferramenta Trello, utilizando uma organização baseada em Kanban para distribuição e acompanhamento das tarefas.

As funcionalidades (Tabela 2) foram classificadas considerando três critérios: a essencialidade para o funcionamento básico do sistema, a dependência técnica em relação a outras funcionalidades e a urgência manifestada pela equipe do museu durante o levantamento de requisitos. Foram priorizadas inicialmente as funcionalidades relacionadas à autenticação, cadastro e consulta do acervo, enquanto funcionalidades administrativas e complementares foram organizadas para etapas posteriores do desenvolvimento.

Tabela 2 - Lista de Funcionalidades.

Funcionalidade	Código	Descrição	Prioridade
Gerenciar Acervo	UC001	Permite cadastrar, editar, ocultar/exibir, excluir e consultar itens do acervo (livros, jornais e revistas)	Alta
Consultar Acervo	UC002	Permite que qualquer visitante pesquise e visualize os itens do acervo, sem necessidade de autenticação	Alta
Autenticar Usuário	UC003	Permite que administradores e bibliotecários acessem o sistema mediante usuário e senha	Alta
Gerenciar Usuário	UC004	Permite que o Administrador cadastre, edite e exclua contas de usuários do tipo Bibliotecário	Média
Gerenciar Editoras	UC005	Permite cadastrar, editar, excluir e consultar editoras para associação aos itens do acervo	Média

Fonte: Construção do Autor.

O planejamento de desenvolvimento priorizou, inicialmente, a autenticação e o cadastro básico do acervo, funcionalidades essenciais ao funcionamento do sistema. Na sequência, foram implementados os recursos de listagem, busca, filtragem e paginação dos itens do acervo, seguidos por melhorias relacionadas à usabilidade, como o carregamento de imagens dos itens e a adaptação responsiva da interface para dispositivos móveis. Por fim, foram desenvolvidas as funcionalidades administrativas complementares, incluindo o gerenciamento de editoras e o registro da quantidade de acessos ao sistema.

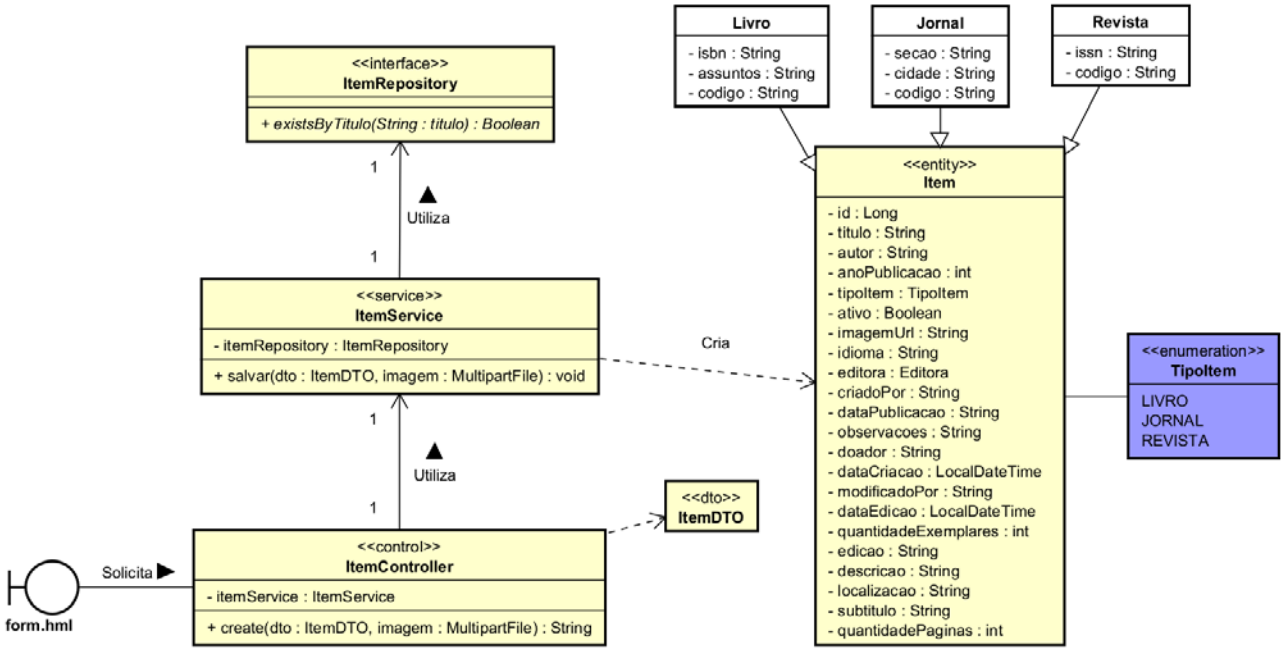
3.4 PROJETO E IMPLEMENTAÇÃO DA SOLUÇÃO

A aplicação foi desenvolvida utilizando a linguagem Java, versão 23, em conjunto com o framework Spring Boot. A camada de apresentação foi implementada com HTML5 (*HyperText Markup Language 5*), CSS (*Cascading Style Sheets*), JavaScript e Thymeleaf, enquanto a persistência dos dados foi realizada em banco de dados MySQL, utilizando Spring Data JPA (*Java Persistence API*) para acesso às informações. A autenticação e o controle de acesso conforme os diferentes perfis de usuários foram implementados com Spring Security.

A aplicação foi estruturada segundo a arquitetura *Model-View-Controller* (MVC), separando as responsabilidades relacionadas aos dados e regras de negócio, à interface com o usuário e ao tratamento das requisições (Fowler, 2002). Para prototipação das interfaces, foi utilizado o Figma, e a implementação foi realizada no ambiente de desenvolvimento IntelliJ IDEA.

Para representar a estrutura envolvida na funcionalidade de cadastro de itens do acervo, foi elaborado o diagrama de classes apresentado na Figura 2. O diagrama representa as principais classes envolvidas no fluxo de criação de um item e suas relações segundo a arquitetura adotada.

Figura 2 - Diagrama de Classes - Criação de Item.

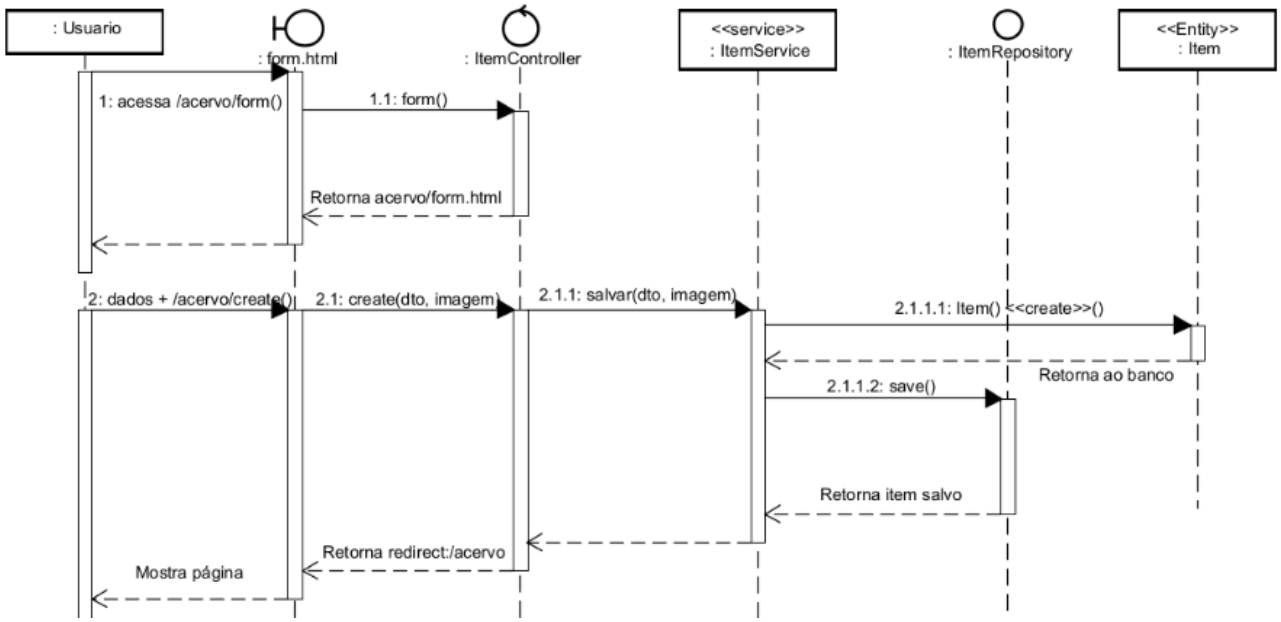


Fonte: Construção do Autor.

A classe ItemController recebe os dados provenientes do formulário e delega o processamento ao ItemService, responsável por converter o ItemDTO em uma instância de Item e persisti-la por meio do ItemRepository. A classe Item representa a generalização das especializações Livro, Jornal e Revista, cada uma contendo os atributos correspondentes ao tipo de material bibliográfico.

O fluxo de interação entre os componentes durante o cadastro de um item é representado pelo diagrama de sequência da Figura 3. Nele, o usuário acessa o formulário, informa os dados do item, e a requisição é encaminhada ao controlador, que aciona a camada de serviço e o repositório responsável pela persistência das informações.

Figura 3 - Diagrama de Sequência - Criando um Item.



Fonte: Construção do Autor.

A Figura 3 apresenta o diagrama de sequência referente ao processo de criação de um item do acervo, evidenciando a interação entre os componentes da aplicação ao longo da execução da funcionalidade. O fluxo inicia com o envio dos dados pelo usuário, recebidos pelo ItemController e encaminhados ao ItemService para processamento. Em seguida, o serviço realiza a criação do objeto correspondente e solicita sua persistência ao ItemRepository. Após a conclusão da operação, o sistema redireciona o usuário para a página de gerenciamento do acervo. O diagrama complementa a representação estrutural apresentada no diagrama de classes, demonstrando o comportamento dinâmico dos componentes envolvidos na funcionalidade.

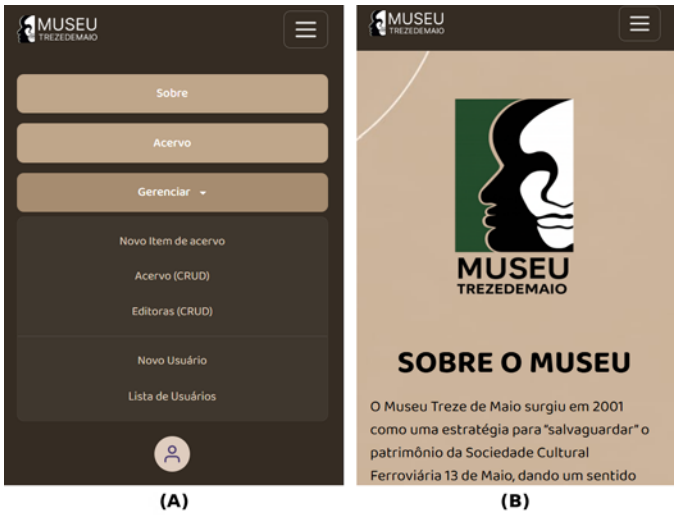
4 RESULTADOS E DISCUSSÃO

Esta seção apresenta os resultados obtidos a partir do desenvolvimento da solução proposta, considerando as funcionalidades implementadas, sua relação com as necessidades identificadas junto ao Museu Treze de Maio e as devolutivas obtidas durante a atividade de verificação e validação extensionista. Além da apresentação do produto tecnológico desenvolvido, são discutidas suas contribuições para a informatização dos processos relacionados à gestão do acervo bibliográfico da instituição.

4.1 FUNCIONALIDADES IMPLEMENTADAS

Como resultado do processo de desenvolvimento, foi implementado um sistema web responsivo para gerenciamento e consulta do acervo bibliográfico do Museu Treze de Maio. A solução contempla diferentes formas de interação conforme o perfil de acesso, permitindo que administradores e bibliotecários realizem operações de gerenciamento, enquanto visitantes podem consultar publicamente os itens disponibilizados no acervo.

Figura 4 - Navegação em Dispositivo Móvel.



(A) (B)

Fonte: Construção do Autor.

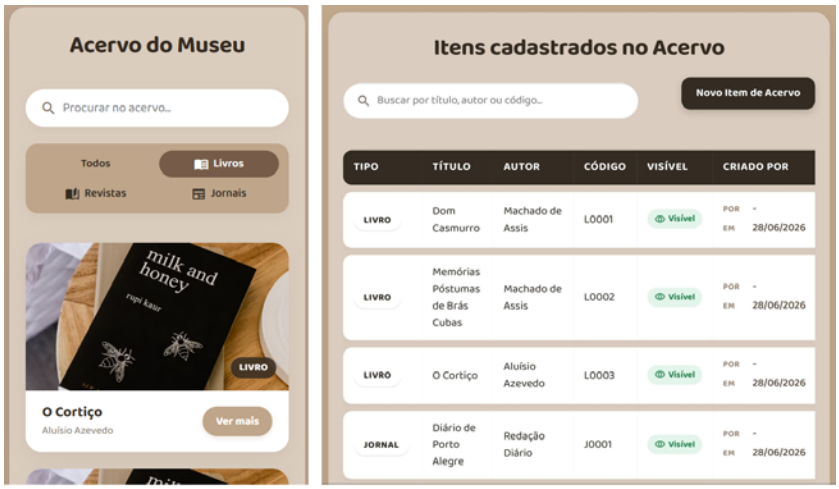
A implementação do controle de acesso considerou os diferentes perfis identificados durante o levantamento de requisitos. Os perfis Administrador e Bibliotecário possuem acesso às funcionalidades relacionadas à manutenção do acervo e ao gerenciamento de editoras, enquanto o gerenciamento de usuários é restrito ao Administrador. O público geral, por sua vez, pode consultar os itens disponibilizados no acervo sem necessidade de autenticação, conforme definido no diagrama de casos de uso.

A Figura 4A apresenta o menu de navegação disponibilizado ao perfil Administrador, evidenciando a implementação das regras de acesso associadas às funcionalidades administrativas. Por meio desse menu, são disponibilizadas opções para gerenciamento do acervo, usuários e editoras, relacionadas respectivamente aos casos de uso UC001, UC004 e UC005. A Figura 4B apresenta a página institucional do Museu Treze de Maio em dispositivo móvel, demonstrando a adaptação responsiva da interface a diferentes dimensões de tela.

A Figura 5A apresenta a implementação da funcionalidade Consultar Acervo (UC002). Conforme especificado durante a modelagem, essa funcionalidade permite que visitantes consultem os itens disponibilizados pela instituição sem necessidade de autenticação.

A interface possibilita realizar busca textual por título, autor ou descrição, bem como aplicar filtros conforme o tipo de item, contemplando livros, jornais e revistas. Os resultados são apresentados de forma paginada e organizados em cartões, contendo informações resumidas e acesso aos detalhes de cada registro.

Figura 5 - Acesso ao Acervo



(A) (B)

Fonte: Construção do Autor.

Na sequência, a Figura 5B, por sua vez, evidencia a implementação da funcionalidade Gerenciar Acervo (UC001), destinada aos usuários autorizados à manutenção dos registros. A interface apresenta os itens cadastrados em formato tabular e disponibiliza informações como tipo, título, autoria, código de identificação, visibilidade e dados de auditoria, além das operações necessárias para edição e exclusão dos registros. Dessa forma, a funcionalidade implementada contempla as principais operações previstas no caso de uso UC001 durante a etapa de modelagem.

Além das funcionalidades diretamente evidenciadas nas Figuras 4 e 5, foram implementados os casos de uso Autenticar Usuário (UC003), responsável pelo acesso de administradores e bibliotecários mediante credenciais; Gerenciar Usuário (UC004), destinado à administração das contas de usuários; e Gerenciar Editoras (UC005), voltado ao cadastro e à manutenção das editoras associadas aos itens bibliográficos.

Também foram incorporados recursos complementares identificados ao longo do desenvolvimento, como o carregamento de imagens dos itens, a adaptação responsiva da interface e o registro das informações de auditoria relacionadas à criação e edição dos registros. Outro recurso solicitado pela instituição foi a possibilidade de acompanhar a quantidade de acessos ao sistema, implementada por meio de um contador disponível aos usuários autenticados, sendo visualizado no rodapé da página.

A relação entre os casos de uso definidos na modelagem e as interfaces apresentadas permite verificar que as funcionalidades centrais previstas para a solução foram incorporadas ao sistema, estabelecendo rastreabilidade entre as necessidades levantadas, a especificação realizada e o produto tecnológico desenvolvido.

4.2 VALIDAÇÃO E DISCUSSÃO DOS RESULTADOS

Após a implementação, o sistema foi apresentado aos representantes do projeto de extensão “Território Negro do Rosário”. Durante a atividade, foram percorridos os principais fluxos da aplicação, incluindo autenticação, cadastro, consulta e edição de itens e mecanismos de busca e filtragem. As devolutivas foram registradas de forma qualitativa durante a apresentação do sistema, sendo posteriormente sistematizadas pelos autores em categorias de percepção favorável e sugestões de aprimoramento. Os representantes manifestaram percepção favorável quanto à organização visual e à compreensão dos fluxos de interação, destacando as informações de auditoria como recurso útil para a rastreabilidade das alterações.

Como aprimoramento, foi sugerida a implementação de um painel (*dashboard*) para acompanhamento dos acessos aos diferentes itens do acervo, complementando o contador geral já existente e possibilitando identificar o interesse do público pelos conteúdos disponibilizados.

Os resultados obtidos demonstram correspondência entre as necessidades levantadas e as funcionalidades implementadas. As operações de cadastro, edição, consulta e controle de acesso foram desenvolvidas de acordo com os casos de uso definidos na etapa de modelagem, atendendo tanto às atividades internas de gerenciamento quanto à consulta pública do acervo.

Além do produto tecnológico desenvolvido, destaca-se o caráter extensionista do projeto. A participação dos representantes do museu no levantamento das necessidades e na apreciação da solução possibilitou aplicar conhecimentos de Engenharia de Software em uma demanda concreta da comunidade, fortalecendo a articulação entre universidade e sociedade no contexto da curricularização da extensão.

A atividade de validação teve caráter qualitativo e extensionista, não constituindo uma avaliação formal de usabilidade. A ausência de instrumentos padronizados e de métricas quantitativas relacionadas à usabilidade, ao desempenho e à segurança constitui uma limitação deste trabalho, podendo ser abordada em estudos futuros por meio de procedimentos sistemáticos de avaliação.

5 CONCLUSÃO

O presente trabalho atingiu o objetivo de desenvolver um sistema web para apoiar a gestão do acervo bibliográfico do Museu Treze de Maio, a partir das necessidades identificadas junto à instituição. O processo de desenvolvimento possibilitou implementar funcionalidades de cadastro, consulta e gerenciamento dos itens bibliográficos, autenticação e controle de acesso por perfis, além de recursos de busca, filtragem, paginação e registro de informações de auditoria. Os resultados apresentados demonstraram correspondência entre os requisitos definidos durante a modelagem e as funcionalidades incorporadas à solução.

Além do produto tecnológico, o trabalho evidencia a contribuição da curricularização da extensão para aproximar a formação acadêmica de demandas concretas da comunidade. A interação entre estudantes e representantes do Museu Treze de Maio esteve presente desde a identificação das necessidades até a apreciação da solução desenvolvida, permitindo aplicar conhecimentos de Engenharia de Software em um contexto real. A atividade de verificação funcional e validação extensionista possibilitou obter devolutivas favoráveis quanto à organização da interface e à compreensão dos principais fluxos, bem como identificar possibilidades de aprimoramento do sistema. Entretanto, por não ter sido realizada uma avaliação formal baseada em instrumentos padronizados ou métricas quantitativas, esses resultados devem ser compreendidos no contexto qualitativo e extensionista em que foram obtidos.

Como trabalhos futuros, propõe-se a implementação de um painel (*dashboard*) para acompanhamento dos acessos aos diferentes itens do acervo, permitindo analisar indicadores de utilização e de interesse do público pelos conteúdos disponibilizados. Também poderão ser realizadas avaliações sistemáticas de usabilidade, desempenho e segurança, complementando a validação realizada nesta etapa. Dessa forma, o sistema desenvolvido constitui uma solução tecnológica adaptada às necessidades identificadas junto ao Museu Treze de Maio e demonstra como a articulação entre universidade e comunidade pode contribuir para o desenvolvimento de ferramentas voltadas à organização e ao acesso às informações relacionadas ao patrimônio cultural.

REFERÊNCIAS

BIBLIVRE. **Biblivre**. Disponível em: https://biblivre.org.br/index.php/sobre-biblivre/o-programa/recursos. Acesso em: 6 ago. 2026.

DIÁRIO DE SANTA MARIA. **Clube Social Negro, que deu origem ao Museu Treze de Maio, completa 120 anos**. Santa Maria, 12 maio 2023. Disponível em: https://diariosm.com.br/cultura/revista_mix/clube_social_negro_que_deu_origem_ao_museu_treze_de_maio_completa_120_anos.504329. Acesso em: 6 ago. 2026.

FOWLER, Martin. **Patterns of Enterprise Application Architecture**. Boston: Addison-Wesley, 2002.

KOHA COMMUNITY. **Koha**. Disponível em: https://koha-community.org/manual/latest/pt_BR/html/index.html. Acesso em: 28 ago. 2026.

OMEKA. **Omeka S User Manual**. Disponível em: https://omeka.org/s/docs/user-manual/. Acesso em: 28 ago. 2026.

PADILHA, Renata Cardozo. **Documentação Museológica e Gestão de Acervo**. Florianópolis: Fundação Catarinense de Cultura, 2014. (Coleção Estudos Museológicos, v. 2). Disponível em: https://cultura.rs.gov.br/upload/arquivos/carga20190653/17105304-documentacao-museologica-gestao-acervo.pdf. Acesso em: 25 ago. 2026.

PRESSMAN, Roger S. **Engenharia de Software: Uma Abordagem Profissional.** 9. ed. Porto Alegre: AMGH, 2021.

RITTER, João Vitor *et al.* **Desenvolvimento de sistema web para fichas técnicas nutricionais.** In: SIMPÓSIO DE ENSINO, PESQUISA E EXTENSÃO DA UNIVERSIDADE FRANCISCANA (SEPE), 29., 2025, Santa Maria. E-book [...]. Santa Maria: Universidade Franciscana, 2025. DOI: https://doi.org/10.48195/sepe2025.29908.

SANTA MARIA. **Prefeitura Municipal de Santa Maria. Museu Treze de Maio lança campanha dos 110 anos no Dia Internacional de Luta contra Discriminação Racial**. Santa Maria, 20 mar. 2013. Disponível em: https://www.santamaria.rs.gov.br/noticias/6254-museu-treze-de-maio-lanca-campanha-dos-110-anos-no-dia-internacional-de-luta-contra-discriminacao-racial. Acesso em: 22 ago. 2026.

TAINACAN. Tainacan - **A flexible and powerful repository platform for WordPress**. Disponível em: https://tainacan.github.io/tainacan-wiki/#/pt-br/README?id=wiki-do-tainacan. Acesso em: 22 ago. 2026.

UNIVERSIDADE FRANCISCANA. **Manual da curricularização da extensão**. Organização de Marcio Tascheto da Silva, Anelis Rolão Flôres, Gustavo Gonçalves Cougo e Laise Viviane Chaves da Rosa. Santa Maria, RS: Editora UFN, 2026.

AUTOMATING SCIENTOMETRICS: A GENERATIVE AI PIPELINE FOR ANALYZING NANOSCIENCE RESEARCH TRENDS

AUTOMATIZANDO A CIENTOMETRIA: UM PIPELINE DE IA GENERATIVA PARA ANALISAR TÉNDENCIAS DE PESQUISA EM NANOCIÊNCIAS

Luiz Batista Cardoso[1], Gabriel Azenha Fachim[2], Maurício Pereira Braga[3], William Leonardo da Silva[4] e Mirkos Ortiz Martins[5]

ABSTRACT

Institutional repositories holds a vast scientific knowledge, but extracting coherent insights from these unstructured collections remains challenging for academic programs. This study presents a computational framework to harvest, clean, and analyze the historical output of a Nanoscience Postgraduate Program. By integrating repository harvesting with a Large Language Model, the pipeline overcomes data quality issues such as broken links and missing metadata. Full PDF papers were converted into Markdown with IBM Docling to maximize semantic inference. The analysis of 241 theses and dissertations reconstructed the program's research keywords and revealed a thematic shift over the last decade, from carbon nanomaterials towards biological and pharmaceutical investigations. These results show that AI-driven workflows can automate advanced bibliometrics, giving coordinators and students dynamic tools to visualize scientific evolution.

Keywords: Academic Analytics; Data Visualization; Institutional Repositories; Large Language Model; Text Mining.

RESUMO

Os repositórios institucionais reúnem um grande volume de conhecimento científico, mas obter análises coerentes a partir dessas coleções não estruturadas ainda é um desafio considerável para os programas

1 Graduate Student in Nanosciences. Universidade Franciscana. E-mail: luiz.bcardoso@ufn.edu.br. ORCID: https://orcid.org/0009-0003-7344-4008.
2 Graduate Student in Nanosciences. Universidade Franciscana. E-mail: gabriel.afachim@ufn.edu.br. ORCID: https://orcid.org/0009-0002-8511-4025.
3 Graduate Student in Nanosciences. Universidade Franciscana. E-mail: mauricio.pereira@ufn.edu.br. ORCID: https://orcid.org/0009-0003-4346-5836.
4 Professor of Graduate Program in Nanosciences. Universidade Franciscana. E-mail: william.silva@ufn.edu.br. ORCID: https://orcid.org/0000-0002-7804-9678.
5 Professor of the Computer Programs. Universidade Franciscana. E-mail: mirkos@ufn.edu.br. ORCID: https://orcid.org/0000-0002-3983-1624.

acadêmicos. Este estudo apresenta um framework computacional de ponta a ponta para coletar, limpar e analisar a produção histórica de um Programa de Pós-Graduação em Nanociências. Ao integrar um repositório de coleta de dados a um Modelo de Linguagem de Grande Escala, o pipeline proposto contorna problemas comuns de qualidade dos dados, como links quebrados e metadados ausentes. Adotou-se uma nova estratégia de processamento de documentos, na qual os PDFs de texto completo foram convertidos para Markdown por meio do IBM Docling, a fim de ampliar a capacidade de inferência semântica. A análise de 241 teses e dissertações obteve uma taxa de retenção de dados de 99,2% e permitiu reconstruir a trajetória de pesquisa do programa. A análise de frequência lexical mostrou uma mudança temática clara na última década, com a passagem de pesquisas fundamentais sobre nanomateriais de carbono para investigações aplicadas nas áreas biológica e farmacêutica. Os resultados mostram que fluxos de trabalho baseados em IA podem automatizar o acesso à bibliometria avançada e oferecer a coordenadores e estudantes ferramentas dinâmicas para visualizar a evolução científica.

Palavras-chave: *Análise Acadêmica; Visualização de Dados; Repositórios Institucionais; Modelos de Linguagem de Grande Escala; Mineração de Texto.*

1 INTRODUCTION

Postgraduate programs are vital for advancing scientific research, yet navigating the historical evolution of interdisciplinary fields like Nanoscience can be challenging (Porter; Youtie, 2009). At the Franciscan University (UFN), the Programa de Pós-Graduação em Nanociências (PPGNano) has made significant strides in contributing to the global body of knowledge in this area. One of the key tools for accessing this corpora is the TEDE (Sistema de Publicação Eletrônica de Teses e Dissertações) (IBCT, 2026), an open-access repository with theses and dissertations from all its PPGs. This platform provides easy access to academic publications and allows for comprehensive data mining for research analysis. However, extracting coherent research trajectories from this repository remains difficult for students and faculty alike. With the increasing prominence of Artificial Intelligence (AI) in various academic disciplines (Buehler, 2024), there is an compelling opportunity to leverage technologies like Generative AI (GenAI) to gain insights into how research has evolved and to map these trends effectively (Zhao *et al.*, 2026).

This article proposes an automated computational pipeline to reconstruct the research history of PPGNano. By utilizing GenAI for metadata enrichment and summarization, we analyze the program's output to identify shifting patterns in central themes and keywords. The primary objective is to provide a comparative analysis of PPGNano's research evolution, using its research lines as a framework. This analysis will serve as a guide for incoming students, particularly those from diverse academic backgrounds, helping them navigate the multidisciplinary nature of the program and identify suitable research paths. In essence, this

approach offers a unique solution to an ongoing challenge. By creating an accessible, data-driven overview of the program's research history, we can not only guide new students but also provide a model that can be adapted to other postgraduate programs, or even undergraduate courses, to track and visualize academic trends.

2 METHODOLOGY

The methodological approach adopted in this study is structured as a sequential computational pipeline designed to automate the lifecycle of scientometric analysis. The workflow is divided into three distinct phases: (i) automated data extraction and filtration from institutional repositories, (ii) metadata enrichment and standardization using GenAI, and (iii) quantitative analysis and data visualization. The following subsections detail the specific protocols applied at each stage to ensure reproducibility and data integrity.

The computational framework was implemented in Python 3.12 (Rossum, 1995). All experiments were conducted on a local workstation equipped with 32GB of RAM and an NVIDIA RTX A4000 GPU, which was utilized for hardware-accelerated document parsing. The software stack relied on the following open-source libraries:

- Data Harvesting: Sickle (Loesch, 2020) for the TEDE repository interaction and BeautifulSoup4 (Richardson, 2004) for HTML parsing.
- Orchestration & Inference: LangChain (Chase, 2022) managed the workflow, utilizing the Google Gemini 2.5 Flash model (Comanici *et al.*, 2025) for long-context semantic processing.
- Document Processing: IBM Docling (Auer *et al.*, 2024) converted PDF documents into structured Markdown.
- Visualization: Matplotlib (Hunter, 2007), Seaborn (Waskom, 2021) and Worldcloud (Mueller, 2012) were used for quantitative plotting.

2.1 DATA HARVESTING STRATEGY

The primary dataset was sourced from the TEDE repository via the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH). Using the Sickle Python library and the "oai_dc" metadata prefix, a complete harvest of the repository was performed. However, utilizing the raw harvest yields a significant volume of unrelated documents.

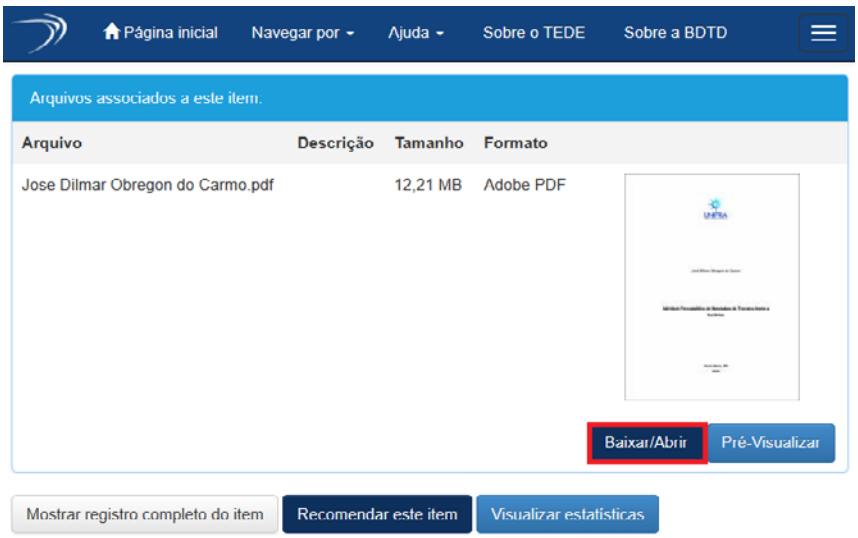
To isolate the specific corpus relevant to this study, a post-harvesting filtration algorithm was developed. This procedure parsed the aggregated JSON file, utilizing regular expressions to retain only records where the "pgp_name" metadata key contained the

substring pattern nano. This step effectively segregated the PPGNano documents from the broader institutional output.

2.2 AUTOMATED DOCUMENT ACQUISITION

Following metadata extraction, a full-text retrieval was performed to acquire the specific scientific content of each publication. A web crawler developed using BeautifulSoup utilized the URLs captured in the harvesting phase. The script navigated the landing page of each entry, identified the dynamic download element, and executed the retrieval of the PDF file. Files were automatically renamed using a composite of the author's surname and publication title to ensure consistent indexing. Figure 1 illustrates the target HTML elements identified by the crawler during this process.

Figure 1 - Target button identified by the crawler.



Source: Prepared by the authors (2026).

2.2 GENAI PROCESSING AND METADATA ENRICHMENT

To address inconsistencies in the source metadata such as missing keywords, absent publication years, or formatting errors common in older documents a GenAI pipeline was implemented. This phase utilized a Long-Context Prompting approach, leveraging the model's extended context window to analyze the full text of the downloaded documents without the need for retrieval-based chunking.

Prior to inference, raw PDF files were processed using the IBM Docling library. This library converted the unstructured PDF binary into structured Markdown text. This conversion was critical for two reasons:

- **Context Optimization**: It allowed for the programmatic exclusion of non-essential sections (e.g., References, Acknowledgments), retaining only the core narrative from Introduction to Conclusion.
- **Token Efficiency**: Markdown representation significantly reduced the token count compared to raw text extraction, optimizing the usage of the model's context window.

The optimized text was passed to the Google Gemini Flash 2.5 model via LangChain with a temperature value of zero, ensuring a deterministic approach. The model was instructed via a strict prompt template to perform two tasks: (1) generate a comprehensive, structured summary of the paper, and (2) categorize the research field according to standard PPGNano lines of research. The Figure 2 shows the Prompt Template used to achieve this tasks.

Figure 2 - Prompt Template string to enrich metadata.

```
prompt_template = """
  You are an expert Article Reviewer and Scientometric Analyst.
  Your task is to analyze the provided academic paper and extract specific metadata to enrich a dataset.

  You must analyze the text to:
  1. Validate/Fill Metadata: Identify the year, authors, and keywords if the provided 'current_metadata' has missing values.
  2. Summarize: Write a rich, technical summary (abstract-level depth) focusing on the methodology and results.
  3. Categorize: Classify the paper into one of the main Nanoscience Research Lines.
  Input Data:
  ---
  Current Metadata (Potentially Incomplete): {current_metadata}
  ---
  Paper Content (Markdown):
  {document_text}
  ---
  **OUTPUT FORMAT INSTRUCTIONS:**
  1) Use the following schema and build upon what was given.
  2) Add the three new fields (research_line, technical_summary, data_quality_notes).
  3) Return only the JSON object using markdown code block format:
  {{
      "paper_name": "paper_name",
      "author_name": "author_name",
      "main_supervisor_name": "main_supervisor_name",
      "abstract": "abstract",
      "keywords": [
          "keyword1",
          "keyword2",
          "keyword3..."
      ],
      "year_of_defense": "YYYY-MM-DD",
      "type": "Dissertation / Thesis",
      "pgp_name": "Mestrado/Doutorado Acadêmico em Something",
      "pdf_url": "http://www.tede.universidadefranciscana.edu.br:8080/handle/UFN-BDTD/XXX"
      "research_line": "Classify into: Nanomaterials / Nanobiotechnology / Computational Nanoscience / etc.",
      "technical_summary": "A detailed paragraph summarizing the objective, method, and key results...",
      "data_quality_notes": "Brief note on what was fixed (e.g., 'Added missing year', 'Fixed encoding in title')"
  }}
"""
```

Source: Prepared by the authors (2026).

2.4 DATA ANALYSIS AND VISUALIZATION

The enriched dataset was analyzed to map the temporal and thematic evolution of the postgraduate program. Python libraries Matplotlib (Hunter, 2007), Seaborn (Waskom, 2021) and Worldcloud (Mueller, 2012) were used to generate quantitative visualizations. To identify dominant scientific themes, a lexical frequency analysis was performed on the standardized

keywords generated by the LLM. This included the generation of a Word Cloud to visualize the most prominent terms across the entire corpus.

2.5 PERFORMANCE EVALUATION FRAMEWORK

To validate the efficacy of the proposed computational workflow, a quantitative analysis was established. This approach monitors data retention and integrity at each critical stage of the pipeline, utilizing the following metrics:

- **Count (N):** The absolute number of valid records or files processed at a specific stage.
- **Step Yield (%):** The efficiency of a specific processing module, calculated as the ratio of the current stage's count to the count of the immediately preceding logical stage (e.g., N_{PDF} /N_{JSON}). This isolates the performance of individual components, such as the PDF downloader or the LLM enrichment module
- **Global Yield (%):** The overall data preservation rate, calculated as the ratio of the current stage's count to the total number of records originally available in the source repository (N_{Stage}/N_{Source}). This metric reflects the cumulative impact of the pipeline on the final dataset size.

The integrity of the metadata was specifically assessed by verifying the presence of valid keywords, defined as records containing at least more than one in the JSON. The GenAI enrichment stage was evaluated by comparing the number of valid records before and after the application of the Large Language Model, quantifying the system's ability to impute missing information.

3 RESULTS AND DISCUSSIONS

The initial OAI-PMH harvesting process, conducted using the Sickle library, yielded a comprehensive dataset of the institution's academic output. While the retrieval mechanism proved robust and straightforward, the raw dataset required specific filtration to isolate the Nanoscience corpus from other graduate programs. To achieve this, a Python-based processing pipeline was deployed to parse the aggregated JSON file. The algorithm utilized a case-insensitive regex to filter the "pgp_name" field, retaining only records containing the nano substring. Figure 3 presents the core logic used for this filtration and the subsequent deduplication tasks.

Figure 3 - Code to filter the PPG and clean duplicates.

parse_repository_metadata.py > ...

```python
import json
import re

INPUT_FILE = 'repository_metadata_ppgnano.json'
OUTPUT_FILE = 'final_repository_metadata_ppgnano.json'
FILTER_PATTERN = r'nano'  # Regex pattern to identify the target program

# Extract simplified "Lastname, Firstname" to ensure consistent author matching
def normalize_author_name(full_name): ···

# Load JSON and filter entries by the postgraduate program keyword
def load_and_filter_repository(filepath): ···

# Remove duplicate entries, prioritizing those with a valid defense year
def remove_duplicates(data): ···

# Strip newlines and excessive whitespace from all metadata strings
def sanitize_metadata(data): ···

# Main execution pipeline
def main():
    raw_data = load_and_filter_repository(INPUT_FILE)
    unique_data = remove_duplicates(raw_data)
    final_data = sanitize_metadata(unique_data)

    with open(OUTPUT_FILE, 'w', encoding='utf-8') as f:
        json.dump(final_data, f, ensure_ascii=False, indent=4)

if __name__ == "__main__":
    main()
```

Source: Prepared by the authors (2026).

Despite the successful isolation of the target data, the intrinsic quality of the raw metadata presented several challenges. A notable issue was observed in older publications, specifically those dating back to 2009. These records frequently exhibited character encoding inconsistencies, particularly in Portuguese titles containing accented characters. To address this, the Large Language Model (LLM) pipeline was configured to identify and correct these encoding artifacts during the subsequent enrichment phase.

Another significant data quality issue involved the handling of URL identifiers. The extraction often returned a list containing multiple links for a single entry. Manual verification revealed a consistent pattern where the first link resulted in a 404 (Not Found) error, while the subsequent link remained valid. Consequently, a cleaning logic was implemented to automatically discard invalid links and retain only the functional URLs.

Furthermore, the dataset required rigorous filtration for duplicate entries. Due to the immutable nature of the repository, updated versions of papers often appeared alongside their original submissions. The selection algorithm, shown in Figure 3, prioritized records containing valid chronological metadata (Year of Defense) to ensure the dataset represented the final version of the work. Following these cleaning procedures, the final dataset comprised 241 unique documents, distributed as 183 master's dissertations and 58 doctoral

theses. Figure 4 illustrates an example of a raw metadata entry prior to AI-based cleaning. The red highlighted sections demonstrate the aforementioned issues, such as character formatting errors in the title and missing keywords replaced by generic placeholders.

Figure 4 - Raw JSON File schema of a paper from TEDE.


```
    "paper_name": "MODELAGEM E SIMULACÃO COMPUTACIONAL DE TEORIA QÂNTICA ORCH OR",
    "author_name": "Martins, Mirkos Ortiz",
    "main_supervisor_name": "Vizzotto, Juliana Kaizer",
    "abstract": "The realization of quantum computers has been investigated from the point
    "keywords": [
        "Nanociências"
    ],
    "year_of_defense": "2009-01-01",
    "type": "Dissertação",
    "pgp_name": "Mestrado Acadêmico em Nanociências",
    "pdf_url": "http://www.tede.universidadefranciscana.edu.br:8080/handle/UFN-BDTD/292"
},
```


Source: Prepared by the authors (2026).

To resolve these inconsistencies and enrich the metadata, the full text of the documents was required. The retrieval pipeline utilized the cleaned JSON dataset to locate the source URLs. A web crawler, built using the BeautifulSoup library, was deployed to access each document's landing page and programmatically identify the specific HTML elements corresponding to the "Baixar/Abrir" (Download/Open) button, as depicted in Figure 1. Figure 5 details the Python function developed to automate this fetching process across the dataset.

Figure 5 - Function to fetch and download PDFs from TEDE.

```python
from bs4 import BeautifulSoup
from urllib.parse import urljoin

JSON_FILE = 'final_repository_metadata_ppgnano.json'
SAVE_DIR = 'media/papers/tede'

# Load the dataset from the JSON metadata file
def load_dataset(filepath): ···
# Sanitize strings to create valid Windows filenames
def clean_filename(text): ···
# Scrape the repository page to find the direct PDF download link
def fetch_direct_link(handle_url): ···
# Download the PDF file stream to the local disk
def download_file(url, path): ···
# Orchestrate the validation and download for a single paper
def process_entry(paper): ···

def main():
    os.makedirs(SAVE_DIR, exist_ok=True)
    papers = load_dataset(JSON_FILE)

    for paper in papers:
        process_entry(paper)

if __name__ == "__main__":
    main()
```

Source: Prepared by the authors (2026).

Following the acquisition of the PDF files, the workflow transitioned to content conversion. IBM's Docling library was employed to convert the unstructured PDF data into Markdown format, which serves as a more efficient context format for LLM processing. This computational task leveraged hardware acceleration via an NVIDIA RTX A4000 GPU. The conversion performance was efficient, with processing times ranging between 3 to 10 seconds per document, depending on the page count and the complexity of graphical elements. Figure 6 presents the implementation of the conversion pipeline, which initiates only after the batch download of PDFs is complete.

Figure 6 - Function to convert PDF into Markdown

```python
def convert_markdown(source, markdown_path, num_pages=30):
  converter = DocumentConverter()
  try:
    result = converter.convert(source,
                               page_range=(1,num_pages))

    print("OK | Sucess on extracting data from PDF...")
    with open(markdown_path, "w") as f:
      f.write(result.document.export_to_markdown())
    return result
  except Exception as e:
    print(f"ERROR | Error extracting data from PDF: {e}")
    return None
```

Source: Prepared by the authors (2026).

With the documents converted to Markdown, the final stage involved semantic processing and summarization. The system integrated the Google Gemini Flash 2.5 model via API. A specific prompt template, illustrated in Figure 2, was designed to guide the model in summarizing the text and correcting the metadata errors identified in the initial extraction phase. The output was structured into a sanitized JSON format. Figure 7 displays the result of this process, showing the same record from Figure 4 with corrected character encoding, populated keywords, and enhanced attributes.

Figure 7 - Enhanced JSON output following GenAI processing.

Enhanced Metadata.json > ...

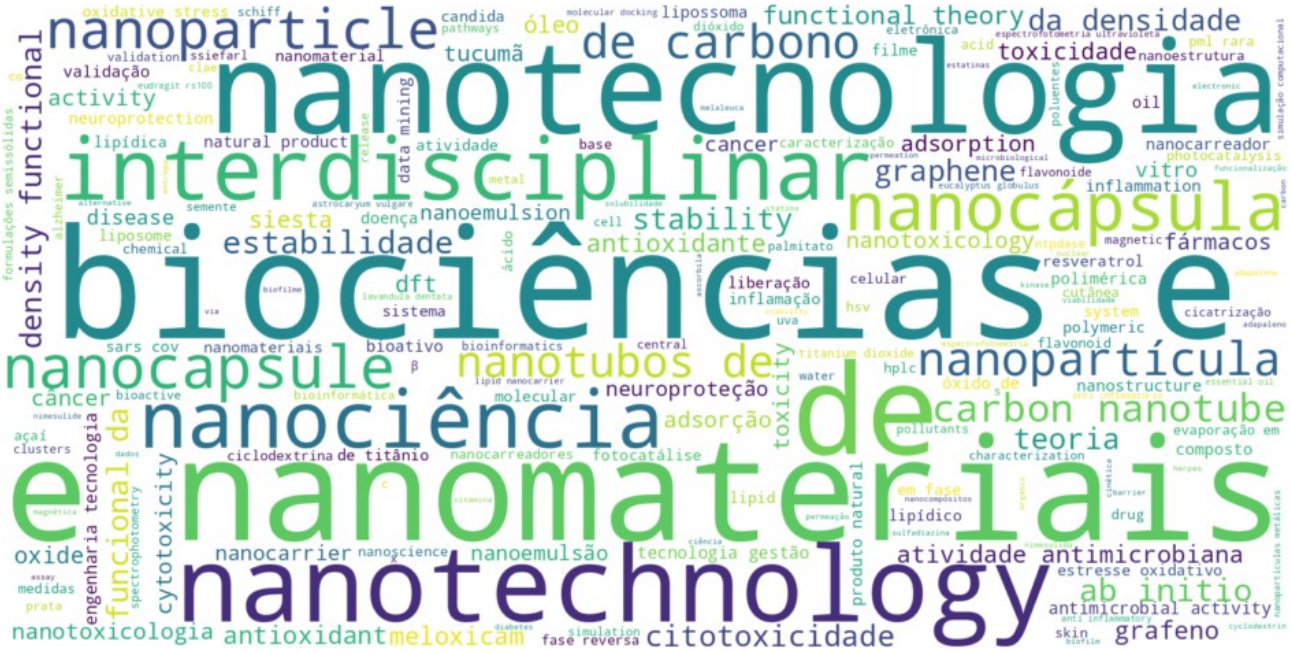

```json
{
    "paper_name": "MODELAGEM E SIMULAÇÃO COMPUTACIONAL DE TEORIA QUÂNTICA ORCH OR",
    "author_name": "Martins, Mirkos Ortiz",
    "main_supervisor_name": "Vizzotto, Juliana Kaizer",
    "abstract": "The realization of quantum computers has been investigated from the point
    "keywords": [
        "Nanociências",
        "Computação Quântica",
        "Orch OR",
        "Microtúbulos",
        "Autômatos Celulares",
        "Modelagem Computacional"
    ],
    "year_of_defense": "2009-01-01",
    "type": "Dissertação",
    "pgp_name": "Mestrado Acadêmico em Nanociências",
    "pdf_url": "http://www.tede.universidadefranciscana.edu.br:8080/handle/UFN-BDTD/292",
    "research_line": "Computational Nanoscience / Nanobiotechnology",
    "technical_summary": "This work presents a computational model and simulation for the Pe
    "data_quality_notes": "Added missing Portuguese and English keywords related to the core
}
```


Source: Prepared by the authors (2026).

To demonstrate the utility of the enriched metadata, a preliminary exploratory analysis was conducted to map the thematic evolution of the PPGNano program. By parsing the standardized keywords and “year_defense” fields from the structured JSON output, we generated visualizations that reveal both the dominant research themes and their temporal distribution.

Figure 8 presents a Word Cloud generated from the complete corpus of keywords. This visualization provides an immediate, high-level overview of the program’s scientific identity. The prominence of terms such as “nanotecnologia” (nanotechnology), “nanocápsulas” (nanocapsules), and “nanotubos de carbono” (carbon nanotubes) confirms the program’s adherence to its core scope. Furthermore, the presence of terms like “atividade antimicrobiana” (antimicrobial activity) and “citotoxicidade” (cytotoxicity) highlights a strong interdisciplinary link with biological and health sciences.

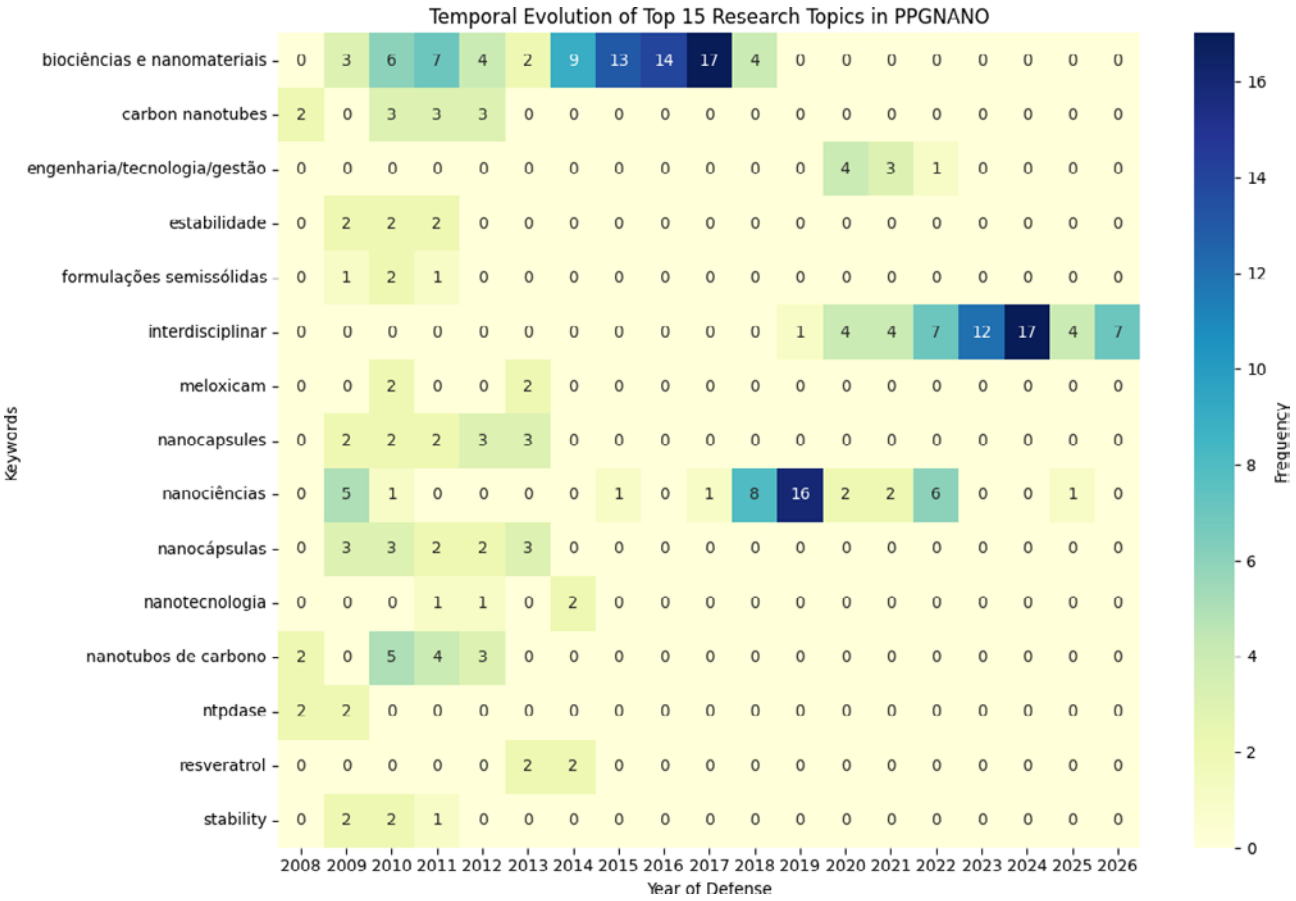
Figure 8 - Word cloud with most frequent terms in the PPGNano.



Source: Prepared by the authors (2026).

To understand how these topics have evolved over time, a frequency matrix was generated for the top 15 occurring keywords. Figure 9 displays this temporal evolution. The heatmap reveals a shift in scientific interest. For instance, ”nanotubos de carbono” shows consistent activity in the earlier years of the dataset, while terms related to specific applications, such as ”nanocápsulas” and ”atividade antioxidante,” show increased frequency in more recent years. This automated plotting capability confirms that the pipeline successfully structured the unstructured data into a format ready for immediate scientometric insight.

Figure 9 - Heatmap with frequency of the top 15 keywords in the PPGNano.



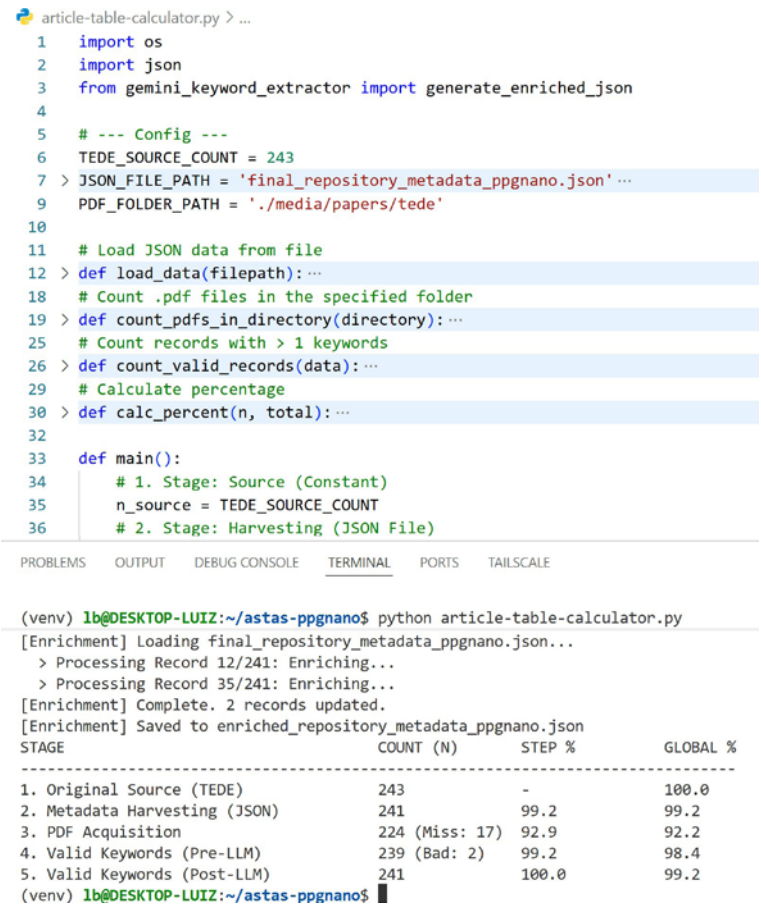
Source: Prepared by the authors (2026).

It is observed that this lexical approach relying solely on exact keyword matches has inherent limitations. Variations in terminology or synonymous phrasing are treated as distinct entities. This suggests that while keyword frequency provides a valid high-level overview, deeper semantic understanding requires more advanced modeling techniques.

3.1 PIPELINE EFFICIENCY AND DATA YIELD

The efficiency of the computational framework was evaluated using the defined analysis metrics, summarizing the data retention rate at each stage of the pipeline. Figure 10 illustrates the Python implementation used to calculate these yield metrics, ensuring reproducible results in the evaluation process.

Figure 10 - Function to calculate pipeline yield metrics.



Source: Prepared by the authors (2026).

Table 1 presents the quantitative results of this evaluation. The metadata harvesting phase demonstrated a successfully indexing of 99.2% (241/243) of the, at the time, records listed in the institutional repository. A reduction in yield was observed during the document acquisition phase, where 88.8% (214/241) of the full-text PDFs were retrieved. This attrition is attributed to broken or restricted access permissions in the older repository entries.

Table 1 - Pipeline data yield and efficiency.

Stage	Count (N)	Step Yield (%)*	Total Yield (%)*
Source (TEDE)	243	-	100%
Metadata Harvesting	241	99.2%	99.2%
PDF Aquisition	214	88.8%	88.1%
Data Integrity (Pre-LLM)	239	99.2%	98.4%
Data Integrity (Post-LLM)	241	100%	99.2%

*Step (%) is relative to the JSON count (N=241); Global (%) is relative to the source (N=243).

Source: Prepared by the authors (2026).

It is important to note the system's resilience handling missing files. For the 214 records where the PDF was successfully acquired, the LLM utilized the full document context

(Introduction to Conclusion). For the remaining 27 records (11.2%) where the PDF was unavailable, the system automatically fell back to a metadata-only inference mode, utilizing the title and abstract from the initial harvest to generate the missing keywords. This hybrid strategy ensures 100% metadata completeness even when full-text access is restricted.

Critically, the GenAI enrichment phase proved highly effective in restoring data utility. Prior to LLM processing, only 239 records contained valid metadata (specifically keywords). By generating synthetic metadata for records with missing fields (e.g., using the title and remaining metadata to infer missing keywords), the system achieved a final metadata completeness score of 100% relative to the harvested JSON. This resulted in a global yield of 99.2%, meaning nearly the entire corpus was successfully structured and made available for downstream scientometric analysis, mitigating the impact of unavailable PDF files.

4 CONCLUSION

In this work, we presented an computational framework for the extraction, enrichment, and analysis of academic metadata from institutional repositories. By integrating open-source scraping tools with advanced generative language models such as Gemini Flash 2.5, the system successfully structured 241 academic documents from the Nanoscience graduate program, achieving a global data retention yield of 99.2% from the repository at the time of the study.

The results demonstrate that while the document acquisition phase faces inherent challenges regarding file availability of 88.8% yield due to repository limitations, the integration of Large Language Models provides a robust mechanism for metadata imputation. The AI module effectively recovered missing information for incomplete records, saving data points that would otherwise be discarded and ensuring a comprehensive dataset for analysis. The use of Large Language Models offered a favorable balance between context window capacity and cost-efficiency, although the document conversion process via IBM Docling remains a computationally intensive step for larger corpora.

Despite these successes, limitations remain. The current exploratory analysis relies on lexical keyword matching, which does not account for synonymous terms or semantic nuances. Furthermore, the definition of research lines currently lacks a standardized taxonomy, leading to potential overlaps in categorization.

Future work will focus on standardizing research lines through a predefined taxonomy for consistent LLM classification, alongside implementing semantic trend analysis via topic modeling. Moving beyond lexical keyword matching will uncover latent semantic clusters, providing a more granular representation of the program’s scientific evolution.

REFERENCES

AUER, C. *et al.* **Docling Technical Report**. Rüschlikon: IBM Research, 2024. Available from: https://doi.org/10.48550/arXiv.2408.09869.

BUEHLER, M. J. **Accelerating Scientific Discovery with Generative Knowledge Extraction, Graph-Based Representation, and Multimodal Intelligent Graph Reasoning**. Machine Learning: Science and Technology, v. 5, n. 3, p. 035083, 2024. Available from: https://doi.org/10.1088/2632-2153/ad7228.

CHASE, H. **LangChain**. [S.l.], 2022. Available from: https://github.com/langchain-ai/langchain. Accessed on: 11 Sept. 2026.

COMANICI, G. *et al.* **Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities**. [S.l.], 2025. Available from: https://doi.org/10.48550/arXiv.2507.06261. Accessed on: 11 Sept. 2026.

HUNTER, J. D. **Matplotlib: A 2D Graphics Environment**. Computing in Science & Engineering, v. 9, n. 3, p. 90–95, 2007. Available from: https://doi.org/10.1109/MCSE.2007.55.

IBCT. **TEDE: Biblioteca Digital de Teses e Dissertações**. Santa Maria, RS: Universidade Franciscana, 2026. Available from: http://www.tede.universidadefranciscana.edu.br:8080/. Accessed on: 11 Sept. 2026.

LOESCH, M. **Sickle: OAI-PMH for Humans**. Version 0.7.0. [S.l.], 2020. Available from: https://github.com/mloesch/sickle. Accessed on: 11 Sept. 2026.

MUELLER, A. C. **wordcloud: A little wordcloud generator in Python**. [S.l.], 2012. Available from: https://github.com/amueller/word_cloud. Accessed on: 11 Sept. 2026.

PORTER, A. L.; YOUTIE, J. **How interdisciplinary is nanotechnology?** Journal of Nanoparticle Research, v. 11, n. 5, p. 1023–1041, 2009. Available from: https://doi.org/10.1007/s11051-009-9607-0.

RICHARDSON, L. **BeautifulSoup**. [S.l.], 2004. Available from: https://www.crummy.com/software/BeautifulSoup/. Accessed on: 11 Sept. 2026.

ROSSUM, G. van. **Python Reference Manual**. Amsterdam: Centrum voor Wiskunde en Informatica (CWI), 1995.

WASKOM, M. L. **seaborn: statistical data visualization**. Journal of Open Source Software, v. 6, n. 60, p. 3021, 2021. Available from: https://doi.org/10.21105/joss.03021.

ZHAO, W. X. *et al.* **A Survey of Large Language Models**. Frontiers of Computer Science, v. 20, n. 12, p. 2012627, 2026. Available from: https://doi.org/10.1007/s11704-026-60308-3.

SISTEMA PARA CONTROLE DE VOO DE DRONE COM RESTRIÇÃO ESPACIAL BASEADA EM GEOLOCALIZAÇÃO

DRONE FLIGHT CONTROL SYSTEM WITH GEOLOCATION-BASED SPATIAL RESTRICTION

Matheus Braga[1], Luiz Batista Cardoso[2], Lourdes Maria Muraro Favarin[3], Marcos Vinicius Pasqualoto Prior[4], Alessandro André Mainardi de Oliveira[5], Sylvio André Garcia Vieira[6], Mirkos Ortiz Martins[7] e André Flores dos Santos[8]

RESUMO

Este artigo apresenta o desenvolvimento e a avaliação de um sistema de geofencing tridimensional para delimitação da área de voo de drones. A solução foi projetada sob a metodologia Feature Driven Development e integra uma aplicação Python, um microcontrolador ESP8266 e um módulo GNSS AT6558 acoplados a um drone DJI Tello. A distância até o ponto de referência é calculada por uma adaptação 3D da fórmula de Haversine. Em teste prático com raio de 15 metros, o sistema identificou a ultrapassagem do limite e acionou o pouso automático. Em área aberta, o posicionamento obteve imprecisão de 0,8 metro e tempo de conexão de 4,24 segundos, enquanto ambientes fechados apresentaram alta imprecisão ou ausência de sinal. Como prova de conceito, a solução mostrou-se promissora para espaços abertos, embora sua confiabilidade dependa diretamente da qualidade do sinal de satélite.

Palavras-chave: ESP8266; Geofencing; Haversine; Posicionamento por satélite; Sistemas embarcados.

1 Egresso do curso de Ciência da Computação. Universidade Franciscana. E-mail: matheus.braga@ufn.edu.br. ORCID: https://orcid.org/0009-0004-6360-4511
2 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: luiz.bcardoso@ufn.edu.br. ORCID: https://orcid.org/0009-0003-7344-4008
3 Aluna de Graduação em Engenharia Biomédica. Universidade Franciscana. E-mail: lourdes.favarim@ufn.edu.br. ORCID: https://orcid.org/0009-0004-0972-8820
4 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: marcos.prior@ufn.edu.br. ORCID: https://orcid.org/0000-0002-3346-8289
5 Professor dos cursos de computação UFN. Universidade Franciscana. E-mail: alessandroandre@ufn.edu.br. ORCID: https://orcid.org/0009-0002-8579-2635
6 Egresso de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: sylvio@ufn.edu.br. ORCID: https://orcid.org/0000-0002-1484-4728
7 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: mirkos@ufn.edu.br. ORCID: https://orcid.org/0000-0002-3983-1624
8 Egresso de Pós-Graduação em Nanociências. Universidade Franciscana. E-mail: andre.flores@ufn.edu.br. ORCID: https://orcid.org/0000-0002-1348-7036.

ABSTRACT

This article presents the development and evaluation of a three-dimensional geofencing system for drone flight area delimitation. Designed using the Feature Driven Development methodology, the solution integrates a Python application, an ESP8266 microcontroller, and an AT6558 GNSS module attached to a DJI Tello drone. Distance to a reference point is calculated via a 3D adaptation of the Haversine formula. In a flight test with a 15-meter radius, the system detected boundary breaches and automatically triggered landing procedures. In open areas, it achieved a positioning error of 0.8 meters and a connection time of 4.24 seconds, whereas indoor testing resulted in high positioning error or signal loss. As a proof of concept, the system proved promising for open environments, though its reliability depends directly on satellite signal quality. structure with the PyMOL software confirmed the fidelity of the representation to the original data. Although the system drastically reduced generation time, evaluations indicate the need to expand reading compatibility for different formats and conduct tests under real usage conditions.

Keywords: *Embedded systems; ESP8266; Geofencing; Haversine; Satellite positioning.*

1 INTRODUÇÃO

O contexto atual tem consolidado a robótica como uma das áreas mais promissoras da ciência moderna (Fenerick; Volante, 2020), com aplicações que vão da saúde à exploração espacial (Hassanalian; Rice; Abdelkefi, 2018), e presença crescente na vida cotidiana (D'Alte, 2019), (Trevizan; Barros, 2007). Os drones, categoria específica de robôs aéreos, destacam-se em aplicações que incluem captura de imagens aéreas, entrega de suprimentos e vigilância de zonas sensíveis e de cultivos agrícolas (Shakhatreh *et al.*, 2019), (Rejeb *et al.*, 2022). Equipados com sistemas de navegação como o Sistema de Posicionamento Global (GPS - Global Positioning System), esses dispositivos voam com precisão e autonomia, permitindo programação de rotas e coleta eficiente de dados (Um *et al.*, 2020). Entretanto, a integração entre robótica, pilotagem de drones e GPS apresenta desafios, especialmente quanto à segurança e privacidade, pois drones com GPS podem coletar imagens ou vídeos não autorizados (Yaacoub *et al.*, 2020).

O controle de um drone por GPS dentro de uma área delimitada por geofencing (limite geográfico virtual em torno de áreas específicas) justifica-se pela crescente importância da robótica e da automação, pelas novas regulamentações sobre o uso de drones e pela necessidade de controlar e restringir o tráfego de aeronaves não tripuladas em espaços aéreos sensíveis (Zeng *et al.*, 2019). Segundo Alamouri, Lampert e Gerke (2023), em janeiro de 2021, a Agência Europeia para a Segurança da Aviação (EASA - European Union

Aviation Safety Agency) estabeleceu novas regras para o uso de drones, cujo impacto envolve limitações de alcance visual, sistemas de geofencing e considerações de privacidade. A pesquisa indicou que 75% dos participantes relataram dificuldades para implementar as novas normas de voo e que 44% enfrentaram problemas relacionados à privacidade, afetando missões de voo e levando ao cancelamento de alguns projetos. Assim, o foco deste projeto é o desenvolvimento de um sistema de geofencing que visa contribuir para o atendimento às novas normas de segurança estabelecidas por órgãos reguladores, promovendo práticas seguras e confiáveis em aplicações tecnológicas cada vez mais presentes em setores estratégicos da sociedade (Yaacoub *et al.*, 2020), (Sousa *et al.*, 2024), (Klidzio *et al.*, 2020).

1.1 OBJETIVOS

O objetivo geral deste trabalho foi projetar, implementar e avaliar um sistema de controle de voo com a funcionalidade de delimitação de área por meio de uma cerca virtual baseada em coordenadas geográficas, de modo a possibilitar o controle e o monitoramento do comportamento de um drone. Para alcançá-lo, foram definidos cinco objetivos específicos: programar e documentar um sistema desktop para controle de voo de drones; construir, programar e documentar um sistema embarcado GPS para a coleta de coordenadas geográficas; programar e documentar métodos de conversão de coordenadas em medidas lineares, expressas em metros; programar e documentar técnicas de geofencing para drones; e definir e aplicar um ambiente de teste para avaliar o desempenho do sistema, a precisão do GPS e a estabilidade de voo do drone.

2 REFERENCIAL TEÓRICO

O desenvolvimento seguiu a metodologia ágil Desenvolvimento Orientado a Funcionalidades (FDD - Feature-Driven Development), proposta por Jeff De Luca e Peter Coad em 1997, que favorece a divisão do projeto em etapas incrementais e bem definidas (Fernandes *et al.*, 2011). Como plataforma de validação, utilizou-se o drone DJI Tello, desenvolvido pela Ryze Tech em parceria com a DJI e a Intel, que se destaca pelo baixo custo, peso reduzido (cerca de 80 gramas) e compatibilidade com Python, Scratch e o Sistema Operacional do Robô (ROS - Robot Operating System) (Ryze Tech, 2023), (Yepes; Barone; Porciuncula, 2021), (Giernacki ,etal.2022)

O posicionamento do drone baseou-se no Sistema de Posicionamento Global (GPS - Global Positioning System), tecnologia por satélites que determina a localização por triangulação de sinais de pelo menos quatro satélites (Um *et al.*, 2020). Como latitude e longitude

são grandezas angulares, sua conversão em medidas lineares foi feita pela fórmula de Haversine (Munari; Setyawan; Fauzan, 2020), (DronEng Drones e Engenharia, 2024). A altitude foi considerada a componente vertical (Z), permitindo estimar deslocamentos tridimensionais e verificar os limites da geofencing.

A captação das coordenadas foi realizada por um sistema embarcado baseado no microcontrolador NodeMCU ESP8266, plataforma de comunicação sem fio de baixo custo compatível com Wi-Fi e com o modelo TCP/IP (Espressif Systems, 2024), acoplado ao módulo de posicionamento GNSS AT6558, capaz de rastrear até 32 canais e receber sinais de seis sistemas de navegação por satélite (DFRobot, 2022). Na implementação, utilizaram-se as linguagens Python (Python Software Foundation, 2025), pela versatilidade, e C/C++ (Microsoft, 2025), (Ebeid *et al.*, 2018), usadas em sistemas embarcados que exigem controle direto de hardware, como os controladores de voo de drones. A comunicação entre o sistema embarcado e o computador seguiu o modelo TCP/IP, baseado nos conceitos de socket e porta lógica, que identificam processos específicos em uma rede (Mendes, 2015).

A entrega do resultado exigiu um contrato entre o conversor e o visualizador. O JSON utilizado neste projeto contém metadados, informações dos átomos, quadros de coordenadas e ligações. Um esquema formal verifica a estrutura do documento, enquanto regras semânticas conferem índices, dimensões e relações internas. Essa separação permite rejeitar um objeto incompleto antes de enviá-lo ao cliente. No Godot, requisições pelo *Hypertext Transfer Protocol* (HTTP) recuperam uma revisão publicada e os dados podem ser convertidos em objetos tridimensionais. A documentação do motor descreve a comunicação por *HTTPRequest* e a instância com *MultiMesh* (Godot Engine, 2026a; Godot Engine, 2026b).

3 TRABALGOS CORRELATOS

Giernacki *et al.* (2022) destaca o drone DJI Tello (Ryze Tech, 2023) como plataforma experimental acessível e eficaz em ambientes acadêmicos, adotada em projetos como o “Tello-Bird”, gerador de trajetórias baseado em código de barras (QR Code - Quick Response Code), e o “Sentry drone”, para detectar e rastrear pessoas em áreas específicas. Setiawan, Sari e Hardyanto (2021) desenvolveu um sistema de rastreamento de animais de estimação por geofencing, baseado em Arduino e integrado a um aplicativo Android, empregando os módulos GPS U-Blox Neo 6M e GSM SIM800L, gerenciados pelo microcontrolador Arduino Pro Mini. O aplicativo recebe notificações quando o animal ultrapassa os limites predefinidos. Graça (2017) integrou sensores a um microcontrolador ESP8266 NodeMCU, transmitindo os dados via Wi-Fi para uma plataforma IoT, permitindo visualização em tempo real e armazenamento para análises futuras.

Esses trabalhos oferecem uma base consistente para este projeto: Giernacki *et al.* (2022) auxilia na compreensão do drone Tello e de seu controle; Setiawan, Sari e Hardyanto (2021) demonstra a geofencing para delimitação de área; e Graça (2017) aborda a integração de módulos ao ESP8266, utilizado no circuito GPS.

4 METODOLOGIA

Esta seção descreve detalhadamente a metodologia adotada para o desenvolvimento do projeto, a qual compreende a condução de um estudo de caso, o levantamento e a definição de requisitos, o projeto de software, a elaboração de protótipos e a montagem do sistema, além da implementação do software. Material suplementar do projeto pode ser acessado em um repositório público (Braga, 2025).

4.1 ESTUDO DE CASO

O estudo de caso é uma abordagem empírica que permite aproximar a teoria da prática, avaliando a aplicabilidade e eficácia em um contexto real (Sommerville, 2018). Com base na metodologia ágil FDD, foi elaborado um esboço visual da interface do projeto, representando os componentes necessários para a construção do sistema e facilitando o levantamento de requisitos, revelando necessidades específicas de hardware e software e antecipando desafios técnicos.

4.2 LEVANTAMENTO E DEFINIÇÃO DE REQUISITOS

As especificações de um sistema solicitadas pelo cliente devem ser organizadas em dois grupos. De acordo com Sommerville (2018), os Requisitos Funcionais (RF) correspondem às especificações das funcionalidades que o sistema deve ser capaz de desempenhar. Já os Requisitos Não Funcionais (RNF) definem como o sistema deve operar, abrangendo aspectos como desempenho, segurança, usabilidade e confiabilidade. Neste projeto, foram identificados requisitos funcionais, como a delimitação da área de voo do drone e a montagem do circuito GPS, além de requisitos não funcionais relacionados a atributos de qualidade e desempenho do sistema, demonstrados na Tabela 1.

Tabela 1 - Estruturas utilizadas no *benchmark*.

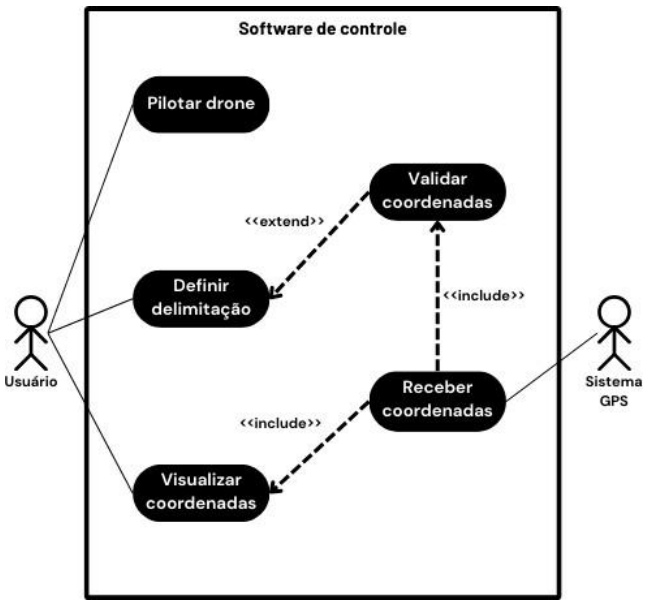
Requisito	Descrição
RF01	O drone deve estabilizar o voo automaticamente.
RF02	O drone pode se mover baseado nas coordenadas do GPS.
RF03	O drone deve realizar movimentos direcionais (frente, trás, lados e giros).
RF04	O drone deve permanecer no ar por mais de um minuto.
RF05	O drone deve permanecer dentro da área definida pelo usuário.
RF06	O software deve limitar a área de voo com base no valor informado pelo usuário.
RNF01	O drone deve ser controlado por rede Wi-Fi.
RNF02	O drone deve ser controlado por uma aplicação desktop.
RNF03	O drone deve ter quatro hélices.
RNF04	A placa utilizada deve ser uma ESP8266.
RNF05	O protocolo de comunicação feito por TCP/IP.
RNF06	O módulo GPS utilizado deve ser o AT6558.
RNF07	O circuito do GPS deve estar fora do hardware do drone.

Fonte: Construção dos autores (2026).

4.3 PROJETO DE SOFTWARE

Na fase inicial deste projeto, foi criado e adaptado um diagrama de caso de uso com o objetivo de demonstrar, de forma simples, as funcionalidades do sistema de geofencing, facilitando o entendimento do seu funcionamento como um todo. A Figura 1, a seguir, apresenta o diagrama de caso de uso, que contém componentes acessíveis ao usuário, como pilotar o drone, definir a área de delimitação e visualizar coordenadas. O diagrama também representa os componentes com os quais o sistema interage, como o circuito GPS externo, responsável por enviar as coordenadas ao software, que realiza a validação da área restrita.

Figura 1 - Diagrama do caso de uso.



Fonte: Construção dos autores (2026).

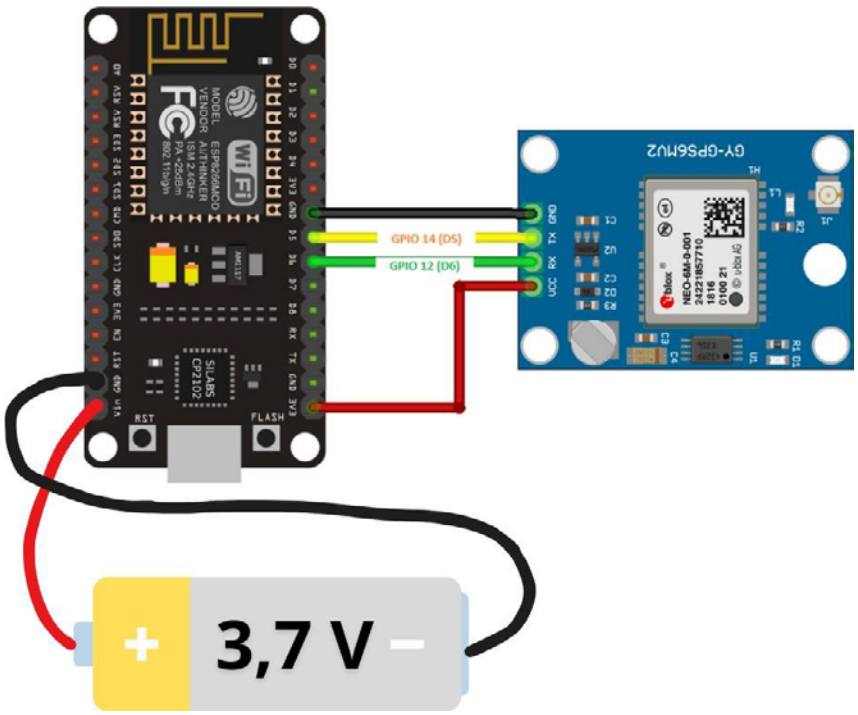
Para detalhar o funcionamento do sistema, foram identificadas as responsabilidades de cada componente no ciclo de vida da aplicação: o usuário define comandos e restrições de voo; o software realiza os cálculos de delimitação e controla o drone; o circuito GPS fornece e transmite as coordenadas geográficas; e o drone executa os comandos de voo recebidos.

4.4 PROTÓTIPOS E MONTAGEM

A prototipagem foi adotada em conjunto com a metodologia FDD como método de verificação dos requisitos do projeto, buscando validar as principais funcionalidades do sistema, ainda em caráter exploratório. Foi implementado o sistema embarcado GPS para a coleta de coordenadas, utilizando um microcontrolador ESP8266 programado em C++. A placa está conectada a um módulo GPS, responsável por captar as coordenadas, e é alimentada por uma bateria de 3,7V e 1000 mAh (miliampere). A Figura 2 apresenta o esquema eletrônico dos componentes. Embora o GPS seja representado no diagrama como um módulo NEO-6M, a aplicação real utiliza um módulo GNSS AT6558, o que atende à necessidade de compactar o circuito para integrá-lo externamente ao chassi do drone.

A montagem do circuito viabilizou a obtenção de coordenadas geográficas, que são analisadas pelo sistema desktop, permitindo ou não o controle do drone dentro da área delimitada.

Figura 2 - Esquema eletrônico



Fonte: Construção dos autores (2026).

4.5 IMPLEMENTAÇÃO DO SOFTWARE

Para o desenvolvimento deste trabalho, foi implementado um sistema desktop em Python, empregando as bibliotecas CustomTkinter, para a interface gráfica; math, para os cálculos matemáticos; threading, para a execução simultânea de múltiplas tarefas; e socket, para a comunicação e o controle de voo do drone Tello (Python Software Foundation, 2025). A implementação do geofencing utilizou a fórmula de Haversine (Munari; Setyawan; Fauzan, 2020). Para isso, foi desenvolvido um código em C++ para o módulo ESP8266, conforme ilustrado na Figura 2, que transmite as coordenadas geográficas do sistema embarcado ao sistema desktop, onde são realizados os cálculos da fórmula de Haversine. O sistema utiliza o protocolo TCP/IP, com o ESP8266 atuando como cliente e o computador como servidor. Os dados GPS são extraídos via interface serial, processados pela biblioteca TinyGPS e enviados como strings de texto no formato LAT: x , LON: y, ALT: z.

A conexão é mantida via rede Wi-Fi, e a cada segundo uma nova leitura é transmitida, permitindo o monitoramento da posição do drone em tempo de execução. A Listagem 1 apresenta o código utilizado para o envio destas coordenadas.

Listagem 1 - Implementação do envio das coordenadas em C++.

```
// se tiver novas informações
if (gps.location.isUpdated()) {
    String message = "";
    message += "LAT:" + String(gps.location.lat(), 6) + ",";
    message += "LON:" + String(gps.location.lng(), 6) + ",";
    message += "ALT:" + String(gps.altitude.meters(), 2);

    Serial.println("Enviando: " + message);
    client.println(message); // envia para o servidor

    delay(1000); // delay de 1 segundo para nova leitura/envio
}
```

Fonte: Elaborada pelos autores (2026).

A Listagem 2 apresenta uma versão adaptada da fórmula de Haversine (Munari; Setyawan; Fauzan, 2020) na linguagem Python para realizar operações de geofencing em ambientes tridimensionais. Diferentemente da fórmula original, esta adaptação também leva em conta a altitude, permitindo uma análise mais precisa em cenários que envolvem variações de elevação. A lógica baseia-se na comparação entre as coordenadas geográficas coletadas em tempo de execução por um circuito embarcado e um ponto de referência previamente definido, verificando se o objeto monitorado permanece dentro dos limites estabelecidos de uma zona geográfica virtual.

Listagem 2 - Implementação do cálculo de geofencing em Python.

```
def haversine_3d(lat1, lon1, alt1, lat2, lon2, alt2):
    R = 6371000  # raio médio da Terra, em metros
    phi1 = math.radians(lat1)
    phi2 = math.radians(lat2)
    delta_phi = math.radians(lat2 - lat1)
    delta_lambda = math.radians(lon2 - lon1)

    a = math.sin(delta_phi / 2)**2 + \
        math.cos(phi1) * math.cos(phi2) * \
        math.sin(delta_lambda / 2)**2
    c = 2 * math.atan2(math.sqrt(a), math.sqrt(1 - a))
    horizontal_distance = R * c  # distância no plano horizontal

    delta_alt = alt2 - alt1  # diferença das altitudes

    # distância 3D combinando distância horizontal e vertical
    return math.sqrt(horizontal_distance**2 + delta_alt**2)
```

Fonte: Elaborada pelos autores (2026).

O método haversine_3d (linha 1) recebe as coordenadas atuais do drone (lat1, lon1, alt1) e a posição de referência definida pelo usuário (lat2, lon2, alt2), fornecidas pelo módulo GPS conectado ao sistema.

Listagem 3 - Implementação da condição da geofencing em Python.

```
# verificar a geofencing apenas no modo estático
if self.operation_mode.get() == "static":
    distance = self.haversine(
        self.origin_lat,
        self.origin_lon,
        self.origin_alt,
        self.current_pos["lat"],
        self.current_pos["lon"],
        self.current_pos["alt"]
    )

    self.in_geofence = distance <= self.geofence_radius.get()

    if not self.in_geofence and self.drone_connected:
        self.land()
```

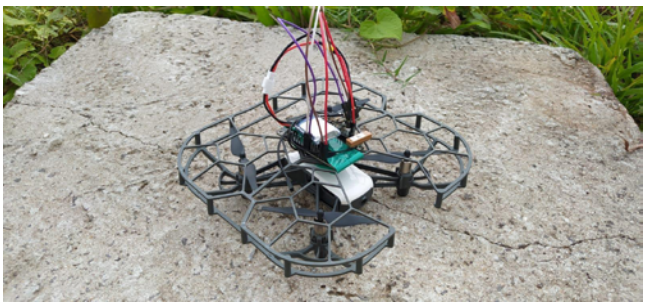
Fonte: Elaborada pelos autores (2026).

Com a fórmula de Haversine adaptada, foram desenvolvidos diferentes métodos de controle de voo, utilizando a biblioteca socket para a comunicação entre o drone Tello e o sistema de geofencing. A estrutura condicional apresentada na Listagem 3 identifica quando o drone ultrapassa os limites da área delimitada e define a ação a ser executada. O sistema também emprega threads para a execução simultânea de múltiplas tarefas (Python Software Foundation, 2025), sendo uma dedicada ao recebimento, a cada segundo, das coordenadas atualizadas da posição do drone. Em seguida, iniciou-se a fase de testes para validar os critérios estabelecidos.

5 RESULTADOS E DISCUSSÃO

Inicialmente, propôs-se a adaptação de um ESP-Drone (Espressif Systems, 2025) com um módulo GPS ao hardware, para testes de geofencing baseados na localização geográfica do drone. No entanto, ao alcançar a etapa de voo, o ESP-Drone apresentou instabilidades, não se mantendo no ar de forma adequada para a análise precisa das coordenadas. Diante disso, optou-se pela substituição pelo drone Tello, da empresa DJI, que passou a transportar o sistema embarcado GPS. Essa mudança exigiu remodelar algumas etapas, pois o módulo GPS não poderia mais ser integrado ao hardware do drone, já que o Tello possui estrutura fechada. Assim, foi projetado o circuito GPS externo, responsável por enviar as coordenadas ao sistema via TCP/IP. O circuito GPS foi construído com base no diagrama da Figura 2 e acoplado ao chassi do drone Tello, como mostra a Figura 3.

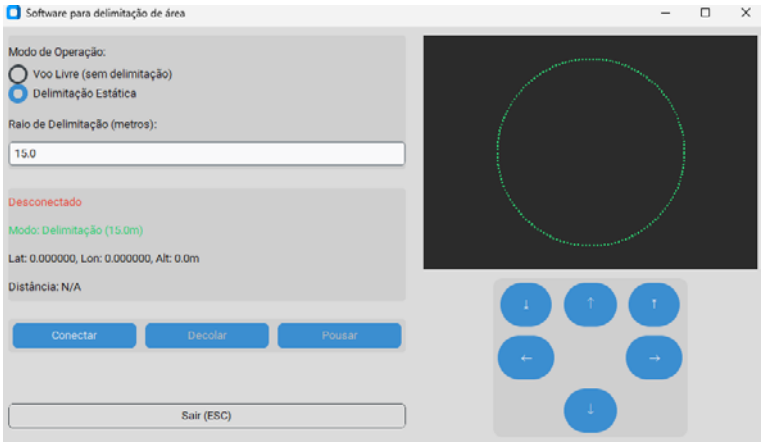
Figura 3 - Drone Tello e circuito GPS.



Fonte: Construção dos autores (2026).

O controle e o monitoramento da posição do drone foram realizados por uma aplicação desktop em Python, integrando o módulo ESP8266 via TCP/IP. A Figura 4 apresenta os principais componentes da interface, como as coordenadas do drone, o modo de operação, a área delimitada, a posição do drone, o status de conexão e os controles de voo.

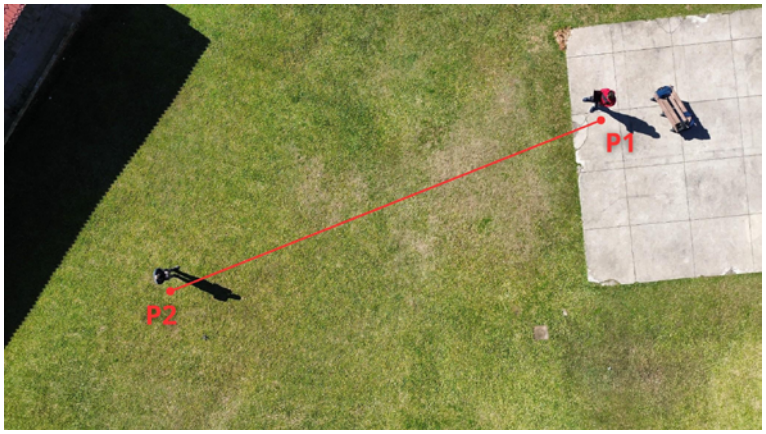
Figura 4 - Componentes da interface do sistema de *geofencing*.



Fonte: Construção dos autores (2026).

Realizou-se um teste de voo para avaliar a precisão do sistema de geofencing, delimitando-se uma área com raio de 15 metros em ambiente aberto, conforme a Figura 5. Essa distância foi medida entre o ponto de partida do drone (P1) e o limite máximo permitido (P2). O sistema foi configurado para reconhecer a área com base na posição inicial do drone. Essa resposta foi previamente programada com o objetivo de verificar o funcionamento do controle de área em uma situação de violação do perímetro.

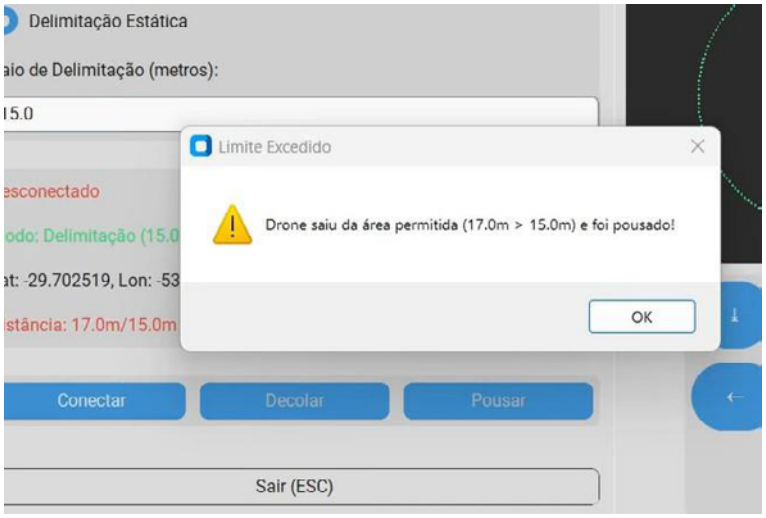
Figura 5 - Área de teste da *geofencing.*



Fonte: Construção dos autores (2026).

A Figura 6 apresenta a notificação emitida quando o drone ultrapassa a área delimitada, acionando a funcionalidade de pouso.

Figura 6 - Alerta da *geofencing*.



Fonte: Construção dos autores (2026).

A partir do teste de voo realizado, foi possível analisar o desempenho do módulo GPS. Observou-se mau desempenho em ambientes muito fechados e variações em relação à posição original, que impactam a definição dos limites. A Tabela II apresenta os valores de

imprecisão coletados em diferentes locais, permitindo analisar as condições mais adequadas para a geofencing.

Tabela 2 - Dados de impresição do GPS

Ambiente	Leituras	Imprecisão GPS (m)	Desvio Padrão (m)	Tempo Conexão GPS	Observação
Sala de Aula I	30	Não Conectado	Não Conectado	Não Conectado	Janelas fechadas
Sala de Aula II	30	34,5 m	6,2 m	37,32 segundos	Janelas abertas
Área Aberta I	30	9,4 m	2,7 m	28,74 segundos	Muitos edifícios
Área Aberta II	30	0,8 m	0,3 m	4,24 segundos	Poucos edifícios

Fonte: Construção dos autores (2026).

As métricas da Tabela II incluem o número de leituras do GPS (30 leituras, uma por segundo, durante 30 segundos), a imprecisão do GPS em metros, o desvio padrão das leituras em metros, o tempo necessário para a conexão com o sinal e observações qualitativas sobre o ambiente. No ambiente “Sala de Aula I”, área de aproximadamente 10×10 metros no térreo de um prédio de três andares, a quantidade de paredes, a presença de edifícios vizinhos e as janelas fechadas dificultaram a conexão com o sinal de GPS. Após a abertura da janela em “Sala de Aula II”, no mesmo local, foi possível estabelecer conexão e receber coordenadas, embora com elevada imprecisão. No ambiente “Área Aberta I”, apesar de ser um espaço ao ar livre, havia edificações ao redor, o que permitiu a conexão, mas resultou em imprecisão e variações nas coordenadas. Já no ambiente “Área Aberta II”, área de 15×15 metros a céu aberto com poucos edifícios, foi possível uma conexão rápida, com baixa imprecisão. A imprecisão foi calculada pela diferença em metros entre a coordenada real conhecida do ambiente e as médias das posições captadas durante os 30 segundos de leitura. A imprecisão e menor variação nos dados coletados.

6 CONCLUSÃO

Os resultados obtidos demonstram uma prova de conceito funcional que atende às necessidades para a delimitação de área em condições controladas. O sistema de geofencing, avaliado em um único teste de voo relatado, foi capaz de delimitar a área de voo do drone e executar os comandos de controle, enquanto a interface gráfica contribuiu para a organização e o monitoramento das operações. No entanto, observou-se variação na precisão do GPS em ambientes muito fechados.

Para trabalhos futuros, recomenda-se a realização de testes adicionais, em diferentes condições e ambientes, a fim de validar os resultados apresentados, bem como a utilização de GPS e antenas de maior precisão e a aplicação do sistema em áreas abertas.

Com a Programação Orientada a Objetos (POO), o sistema pode ser implementado em drones de diferentes marcas, inclusive com o módulo GPS incorporado ao hardware, facilitando novas soluções alinhadas às regulamentações demonstradas por Alamouri, Lampert e Gerke (2023).

REFERÊNCIAS

ALAMOURI, A.; LAMPERT, A.; GERKE, M. **Impact of drone regulations on drone use in geospatial applications and research: Focus on visual range conditions, geofencing and privacy considerations**. PFG - Journal of Photogrammetry, Remote Sensing and Geoinformation Science, v. 91, p. 381-389, 2023. Disponível em: https://doi.org/10.1007/s41064-023-00246-y.

BRAGA, M. **geofencing-system**. 2025. Repositório GitHub. Disponível em: https://github.com/matheuskbraga/geofencing-system. Acesso em: 29 ago. 2026.

D'ALTE, P. G. **Imagologia robótica: robôs e inteligência artificial nas narrativas para crianças e jovens**. Revista Observatório, v. 5, n. 5, p. 586-614, 2019. Disponível em: https://doi.org/10.20873/uft.2447-4266.2019v5n5p586.

DFROBOT. **GPS+BDS BeiDou Dual Module (SKU: TEL0132)**. 2022. Disponível em: https://wiki.dfrobot.com/GPS_+_BDS_BeiDou_Dual_Module_SKU_TEL0132. Acesso em: 29 ago. 2026.

DRONENG DRONES E ENGENHARIA. **O que são sistemas de referência e de coordenadas?** 2024. Disponível em: https://blog.droneng.com.br/geodesia-sistemas-de-referencia--e-de-coordenadas/. Acesso em: 29 ago. 2026.

EBEID, E. *et al.* **A survey of open-source uav flight controllers and flight simulators**. Microprocessors and Microsystems, v. 61, p. 11-20, 2018. Disponível em: https://doi.org/10.1016/j.micpro.2018.05.002.

ESPRESSIF SYSTEMS. **ESP8266EX Datasheet**. [S.l.], 2024. Disponível em: https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf. Acesso em: 29 ago. 2026.

ESPRESSIF SYSTEMS. **ESP32 Series Datasheet**. [S.l.], 2025. Disponível em: https://www.espressif.com/sites/default/files/documentation/esp32_datasheet_en.pdf. Acesso em: 29 ago. 2026.

FENERICK, J. A.; VOLANTE, C. R. **A evolução das indústrias, os benefícios da automação e as perspectivas do mercado da robótica no Brasil e no mundo**. Revista Interface Tecnológica, v. 17, n. 1, p. 734-745, 2020. Disponível em: https://doi.org/10.31510/infa.v17i1.805.

FERNANDES, B. B. *et al.* **Metodologia fdd aplicada a especificação dos requisitos no processo gre do nível g de maturidade do modelo mps.br**. In: ENCOINFO - CONGRESSO DE COMPUTAÇÃO E TECNOLOGIAS DA INFORMAÇÃO. [S.l.: s.n.], 2011. p. 199-208.

GIERNACKI, W. *et al.* **Dji tello quadrotor as a platform for research and education in mobile robotics and control engineering**. In: INTERNATIONAL CONFERENCE ON UNMANNED AIRCRAFT SYSTEMS (ICUAS), 2022, Dubrovnik. Anais [...]. Dubrovnik, Croatia: [s.n.], 2022. p. 735-744. Disponível em: https://doi.org/10.1109/ICUAS54217.2022.9836168.

GRAÇA, P. C. **Sistema de aquisição de dados utilizando o módulo ESP8266 NodeMCU**. 2017. 42 f. Monografia (Trabalho de Conclusão de Curso) - Faculdade de Engenharia de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá, 2017. Disponível em: http://hdl.handle.net/11449/156909. Acesso em: 29 ago. 2026.

HASSANALIAN, M.; RICE, D.; ABDELKEFI, A. **Evolution of space drones for planetary exploration: A review**. Progress in Aerospace Sciences, v. 97, p. 61-105, 2018. Disponível em: https://doi.org/10.1016/j.paerosci.2018.01.003.

KLIDZIO, A. M. *et al.* **Uso de drones em logística**. In: FATECLOG, 11., 2020, Bragança Paulista. Os desafios da logística real no universo virtual [...]. Bragança Paulista, SP: [s.n.], 2020.

MENDES, D. R. **Redes de computadores: teoria e prática**. 2. ed. São Paulo: Novatec, 2015.

MICROSOFT. **C/C++ Documentation - Visual C++**. 2025. Disponível em: https://learn.microsoft.com/pt-br/cpp/. Acesso em: 29 ago. 2026.

MUNARI, A. S.; SETYAWAN, M. Y. H.; FAUZAN, M. N. **Panduan Lengkap Algoritma Haversine Formula Pada Sistem Monitoring Mahasiswa Internship Berbasis GPS**. [S.l.]: CV Kreatif Industri Nusantara, 2020.

PYTHON SOFTWARE FOUNDATION. **What is Python? Executive Summary**. 2025. Disponível em: https://www.python.org/doc/essays/blurb/. Acesso em: 29 ago. 2026.

REJEB, A. *et al.* **Drones in agriculture: A review and bibliometric analysis**. Computers and Electronics in Agriculture, v. 198, p. 107017, 2022. Disponível em: https://doi.org/10.1016/j.compag.2022.107017.

RYZE TECH. **Tello - Feel the Fun**. 2023. Disponível em: https://www.ryzerobotics.com/tello. Acesso em: 29 ago. 2026.

SETIAWAN, D.; SARI, M. W.; HARDYANTO, R. H. **Geofencing technology implementation for pet tracker using arduino based on android**. Journal of Physics: Conference Series, v. 1823, p. 012055, 2021. Disponível em: https://doi.org/10.1088/1742-6596/1823/1/012055.

SHAKHATREH, H. *et al.* **Unmanned aerial vehicles (uavs): A survey on civil applications and key research challenges**. IEEE Access, v. 7, p. 48572-48634, 2019. Disponível em: https://doi.org/10.1109/ACCESS.2019.2909530.

SOMMERVILLE, I. **Engenharia de Software**. 10. ed. [S.l.]: Pearson, 2018.

SOUSA, M. P. *et al.* **Avanços e aplicações de drones na gestão de recursos naturais e monitoramento ambiental no semiárido brasileiro**. Revista de Gestão e Secretariado, v. 15, n. 7, p. e4030, 2024. Disponível em: https://doi.org/10.7769/gesec.v15i7.4030.

TREVIZAN, F. W.; BARROS, L. N. de. **Robótica cognitiva: programação baseada em lógica para controle de robôs**. Controle & Automação, v. 18, n. 2, p. 187-198, 2007. Disponível em: https://doi.org/10.1590/S0103-17592007000200005.

UM, I. *et al.* **Configuring rtk-gps architecture for system redundancy in multi-drone operations**. IEEE Access, v. 8, p. 76228-76242, 2020. Disponível em: https://doi.org/10.1109/ACCESS.2020.2989276.

YAACOUB, J.-P. *et al.* **Security analysis of drones systems: Attacks, limitations, and recommendations**. Internet of Things, v. 11, p. 100218, 2020. Disponível em: https://doi.org/10.1016/j.iot.2020.100218.

YEPES, I.; BARONE, D. A. C.; PORCIUNCULA, C. M. D. **Use of drones as pedagogical technology in stem disciplines**. Informatics in Education, v. 21, n. 1, p. 201-233, 2021. Disponível em: https://doi.org/10.15388/infedu.2022.08.

ZENG, G. *et al.* **An airport airspace flow control method for drones**. In: IEEE INTERNATIONAL CONFERENCE ON UNMANNED SYSTEMS (ICUS), 2019, Beijing. Anais [...]. Beijing, China: [s.n.], 2019. p. 178-182. Disponível em: https://doi.org/10.1109/ICUS48101.2019.8996012.

SISTEMA WEB PARA CATALOGAÇÃO E GESTÃO DO ACERVO DO MUSEU TREZE DE MAIO

WEB-BASED CATALOGATION AND MANAGEMENT SYSTEM FOR THE TREZE DE MAIO MUSEUM COLLECTION

Luiza Lewandowski Karlec[1], Diana Frazzon Rolim[2], Vanessa Cezar Nascimento[3], Eunice Olivera[4], Paula Ribeiro Ciochetto[5] e Hérysson Rodrigues Figueiredo[6]

RESUMO

O presente trabalho trata sobre o desenvolvimento de um Sistema Web de Catalogação e Gestão para o Acervo do Museu Treze de Maio ao longo de disciplinas extensionistas do Curso de Ciência da Computação. A demanda por um ambiente que proporcione gerenciamento e organização sobre os itens armazenados, disponibilidade destes itens ao público em geral e que ajude a preservar a história do Museu motivou o desenvolvimento do projeto. A partir do levantamento de requisitos e da modelagem de domínio, o sistema foi construído com a metodologia *Feature-Driven Development* (FDD) integrada ao método *Kanban*, utilizando *Java*, *Spring Boot*, *Spring Security*, *MySQL*, *Thymeleaf* e *Bootstrap*. Como resultado, obteve-se uma versão funcional do sistema, contemplando o cadastro, a edição e a consulta livros, revistas e jornais, contribuindo para a organização e preservação do acervo do Museu.

Palavras-chave: Aplicação Web; Catalogação Digital; Extensão Universitária; Metodologias Ágeis; Spring Boot.

1 Acadêmica do Curso de Ciência da Computação UFN. E-mail: luiza.karlec@ufn.edu.br. ORCID: https://orcid.org/0009-0007-5430-6135
2 Acadêmica do Curso de Ciência da Computação UFN. E-mail: diana.frazzon@ufn.edu.br. ORCID: https://orcid.org/0009-0007-5430-6135
3 Acadêmica do Curso de Ciência da Computação UFN. E-mail: vanessa.cezar@ufn.edu.br. ORCID: https://orcid.org/0009-0009-4641-2821
4 Bibliotecária, Universidade Franciscana (UFN). E-mail: eunice.olivera@ufn.edu.br. ORCID: https://orcid.org/0000-0002-2063-4786.
5 Professora do curso de História UFN. E-mail: paula.ciochetto@ufn.edu.br. ORCID: https://orcid.org/0000-0001-9949-7712
6 Professor dos Cursos da Área de Computação UFN. E-mail: herysson.figueiredo@ufn.edu.br. ORCID: https://orcid.org/00090009-2615-9785

ABSTRACT

This paper discusses the development of a web-based archival system for the Treze de Maio Museum's collection, throughout extension courses within the Computer Science program. The project was driven by the need for a system that enables the management and organization of stored items, making these items accessible to the general public, and helping to preserve the Museum's history. Based on requirements gathering and domain modeling, the system was built using Feature-Driven Development (FDD) integrated with Kanban method, employing Java, Spring Boot, Spring Security, MySQL, Thymeleaf, and Bootstrap. The project resulted in a functional system capable of registering, editing, and querying records for books, magazines, and newspapers, contributing to the organization and preservation of the Museum's collection.

Keywords: *Web Application; Digital Catalogation; University Extension; Agile Methodologies; Spring Boot.*

1 INTRODUÇÃO

A extensão na Universidade Franciscana (UFN) fortalece a relação entre universidade e sociedade, promovendo a transformação social por meio da integração com o ensino e a pesquisa aplicando o conhecimento técnico em soluções para as realidades locais (Universidade Franciscana, 2026). Com base nesse princípio, este artigo aborda o processo de construção de um sistema web de catalogação e gestão desenvolvido para o Museu Treze de Maio no decorrer de disciplinas extensionistas do curso de Ciência da Computação, estando vinculado ao projeto de extensão "Território Negro do Rosário: o Museu Treze de Maio como espaço de educação antirracista, preservação da história e memória".

Na década de 1990, Hayes (1993 *apud* Silva; Ribeiro, 2002) previa o avanço do uso de ferramentas tecnológicas para modernizar a busca de dados e a elaboração bibliográfica. Atualmente, essa previsão se concretiza por meio do desenvolvimento de sistemas dedicados à gestão digital de coleções físicas, dito isso, garantir a preservação de fontes históricas exige ferramentas que organizem e protejam o patrimônio documental. Essa necessidade reflete diretamente a realidade em que o Museu Treze de Maio se encontrava, possuindo um importante patrimônio que precisa ser digitalizado, catalogado e disponibilizado. Esse acervo, no entanto, não possuía um sistema digital de catalogação, dificultando a busca, organização e divulgação dos itens tanto para a equipe interna quanto para o público externo, com isso, surge a necessidade de desenvolver um sistema de gerenciamento eficiente e acessível.

Criado no ano de 2001, na cidade de Santa Maria/RS, o Museu Treze de Maio é um museu comunitário que preserva a história e a memória afro-brasileira, e se localiza no

espaço que anterior anteriormente abrigou um Clube Social Negro, a Sociedade Cultural Ferroviária Treze de Maio, que existiu entre 1903 e o início da década de 1990. Ao decorrer dos anos, o espaço do Treze possibilitou o desenvolvimento de mecanismos de valorização de memória, tornando-se um importante local de sociabilidade para a comunidade negra na cidade (Weber; Zanini, 2022). Diante da relevância desse patrimônio, o trabalho justifica-se pela necessidade de preservar a sua memória por meio da organização dos registros, além de disponibilizar o acesso às informações ao público geral, como pesquisadores e demais interessados. Destaca-se que o museu possui acervo referente ao clube, contando com carteiras e fichas de sócios, atas, fotografias, e possui um acervo de cerca de 3 mil livros, que será transformado em biblioteca comunitária, além de coleções de revistas e jornais.

Nesse sentido, o presente trabalho tem como objetivo principal relatar o desenvolvimento de um sistema web de catalogação voltado à organização e gestão do acervo do Museu Treze de Maio. A fim de alcançar esse objetivo, foram estabelecidos os seguintes objetivos específicos: levantar os requisitos funcionais e não funcionais por meio de visitas técnicas e entrevistas; desenvolver uma solução web utilizando a arquitetura MVC (Model-View-Controller) com as tecnologias Java e Spring Boot; e realizar uma validação preliminar dos resultados obtidos com a implementação do sistema quanto ao atendimento das necessidades do acervo.

Além desta introdução, o artigo está organizado em outras três seções. Inicialmente, apresenta-se os procedimentos metodológicos adotados para o levantamento das necessidades do Museu Treze de Maio, a modelagem e o desenvolvimento da solução proposta. Na sequência, discute-se os principais resultados obtidos com a implementação do sistema, considerando as funcionalidades desenvolvidas e sua relação com as necessidades identificadas. Por fim, apresenta-se as conclusões do trabalho, suas limitações e as perspectivas para a continuidade e implantação do sistema.

2 METODOLOGIA

Este trabalho caracteriza-se como uma pesquisa de natureza aplicada e tecnológica, desenvolvida em contexto extensionista, uma vez que busca empregar conhecimentos da área de Ciência da Computação na construção de uma solução para uma demanda concreta identificada junto ao Museu Treze de Maio. Quanto aos seus objetivos, o estudo apresenta caráter descritivo, pois contempla o levantamento e a descrição das necessidades relacionadas à organização, catalogação e consulta do acervo, bem como o processo de desenvolvimento da solução computacional proposta.

O desenvolvimento do trabalho ocorreu de forma integrada às disciplinas extensionistas do curso de Ciência da Computação, envolvendo a participação dos acadêmicos, docentes e representantes das instituições envolvidas. A identificação do problema e das necessidades do Museu Treze de Maio ocorreram por meio de visita técnica ao local e de diálogos com os responsáveis pelo acervo. Posteriormente, foram realizadas entrevistas com bibliotecárias da Universidade Franciscana, com o objetivo de identificar os dados e atributos necessários para a adequada catalogação dos materiais contemplados pelo sistema.

A partir das informações levantadas, o desenvolvimento da solução foi organizado em quatro etapas principais: levantamento das necessidades e definição dos requisitos do sistema; modelagem do domínio e da estrutura de dados; definição e condução do processo de desenvolvimento; e implementação da solução computacional. Ao final do desenvolvimento, foi realizada uma validação preliminar da solução por meio de sua apresentação aos responsáveis pelo Museu Treze de Maio e às bibliotecárias envolvidas no levantamento das informações, possibilitando a interação com as principais funcionalidades implementadas e a verificação de sua correspondência com as necessidades inicialmente identificadas.

As etapas metodológicas adotadas são detalhadas nas subseções seguintes, abrangendo o levantamento das necessidades, a modelagem do domínio e do banco de dados, o processo de desenvolvimento e a implementação da solução.

2.1 LEVANTAMENTO DAS NECESSIDADES

Este projeto teve início na disciplina de Implementação de Banco de Dados, sendo uma das ações propostas pelo projeto de extensão “Território Negro do Rosário”, na busca pela preservação do acervo do Museu Treze de Maio. No âmbito da referida disciplina realizou-se uma visita técnica ao museu (Figura 1a), onde o grupo teve a chance de conhecer um pouco da história do local, bem como compreender o contexto de aplicação do sistema.

Figura 1 - Fotografias da Visita ao Museu Treze de Maio.



(a) Visita ao museu (b) Materiais armazenados fisicamente

Fonte: Fotografada pelos Autores (2025).

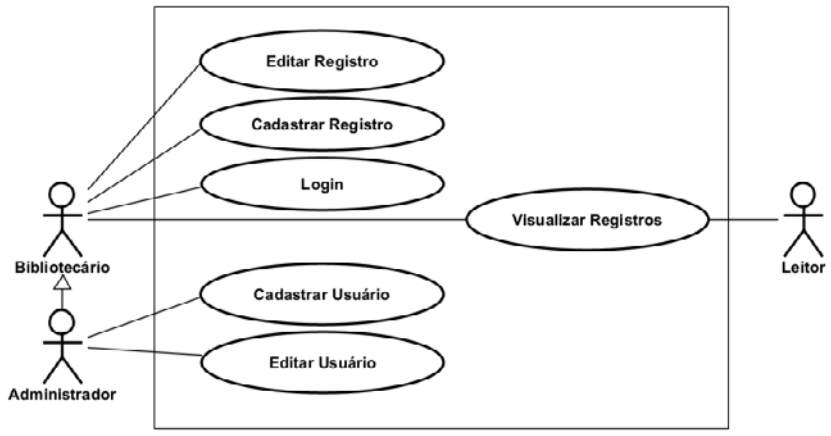
No decorrer do encontro, foi possível ter contato com os materiais armazenados fisicamente (Figura 1b), além dialogar com os responsáveis pelo espaço sobre as principais dificuldades enfrentadas na organização e na catalogação dos itens do acervo, e assim decidiu-se que primeiramente o sistema deveria se propor apenas a catalogação de livros, revistas e jornais. Essa vivência permitiu à equipe compreender a realidade do museu e as demandas que o sistema deveria atender.

A partir das informações coletadas durante a visita, realizou-se a etapa de levantamento de requisitos com o propósito de identificar as necessidades do sistema. Definem-se como requisitos o conjunto de funções que a aplicação deve realizar, combinadas com as restrições impostas ao seu funcionamento, refletindo diretamente as necessidades dos usuários e clientes do produto (Sommerville, 2018, p. 89).

Com base nos requisitos funcionais identificados, foi desenvolvido o Diagrama de Casos de Uso (Figura 2) utilizando a Linguagem de Modelagem Unificada (UML), padrão adotado na modelagem de programas orientados a objetos, desenvolvido por meio da ferramenta Astah UML, uma ferramenta para criação de diagramas UML (Change Vision, 2026). De acordo com Bezzera (2007, p. 54) “o modelo de caso de uso MCU é uma representação das funcionalidades externamente observáveis do sistema e dos elementos externos ao sistema que interagem com ele. O MCU é um modelo de análise que representa um refinamento dos requisitos funcionais do sistema em desenvolvimento”.

O objetivo do Diagrama de Casos de Uso é representar todas as interações possíveis que serão descritas nos requisitos do sistema por meio gráfico e um texto estruturado, para assim, determinar as funcionalidades e características de software. Os atores, ou agentes, representam os seres humanos ou outros sistemas, são identificados pelos ‘bonecos palito’. Cada funcionalidade é representada por uma elipse nomeada. As interações são feitas por meio de linhas que conectam os atores e as ações entre si (Sommerville, 2018, p. 108).

Figura 2 - Diagrama de Caso de Uso.



Fonte: Construção dos Autores

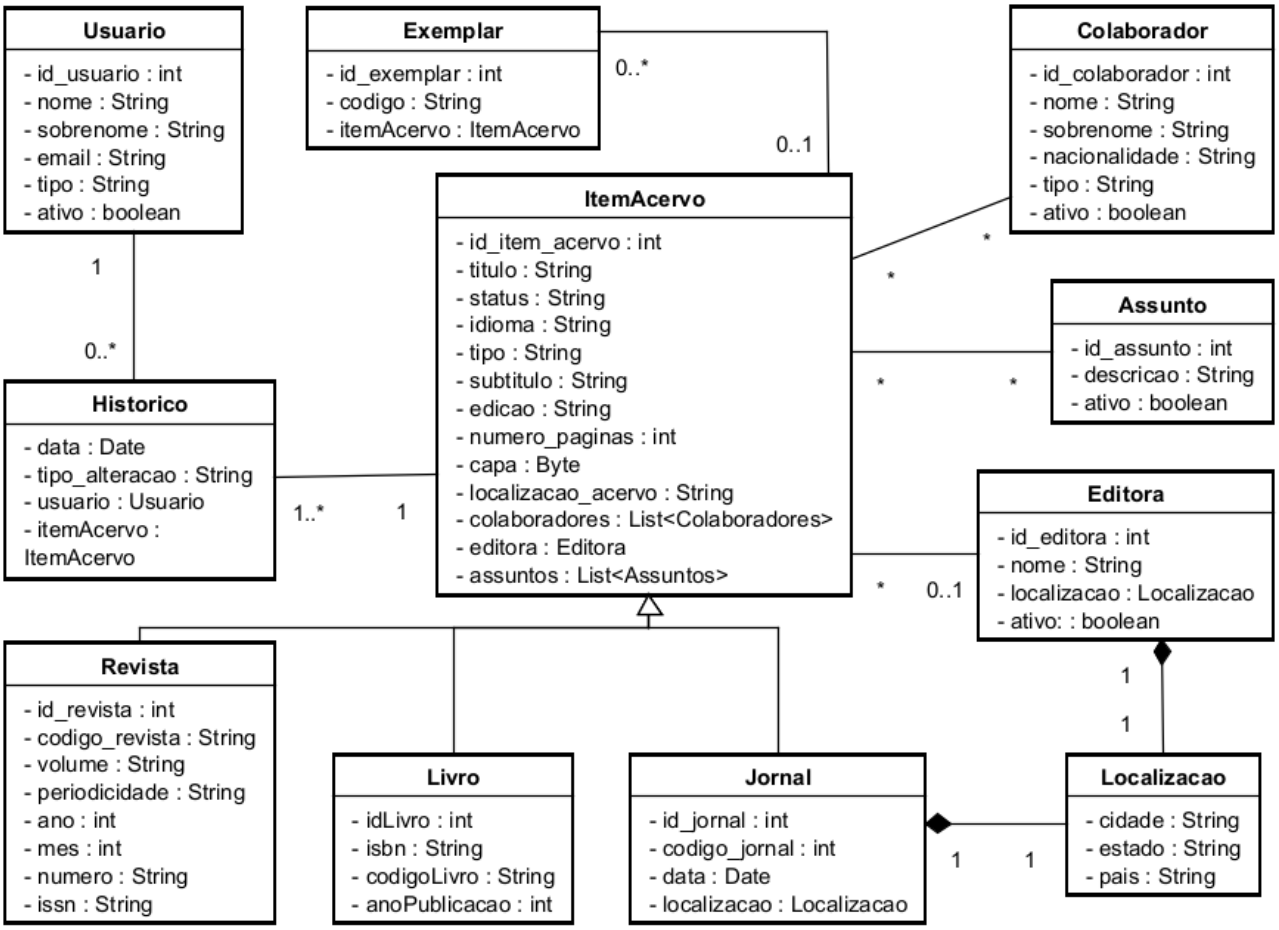
A Figura 2 apresenta o Diagrama de Caso de uso com os três possíveis atores do sistema: o bibliotecário, o administrador e o leitor. Para garantir a segurança e integridade dos dados, existem diferentes níveis de permissão restritos por meio de autenticação (login e senha). O bibliotecário tem acesso ao cadastro e à edição de registros específicos do acervo (abrangendo livros, jornais e revistas). O administrador, por sua vez, possui essas mesmas atribuições, além do cadastro e da edição de novos usuários. Por fim, o leitor pode realizar buscas e visualizar as informações dos itens catalogados sem necessidade de autenticação prévia. Para evitar a perda acidental de dados importantes, a opção de excluir foi substituída por ‘Inativar’, disponível quando um registro é editado.

2.2 MODELAGEM DO DOMÍNIO E BANCO DE DADOS

A partir da necessidade de se identificar quais os atributos considerados fundamentais para a catalogação e exibição das obras no acervo digital, realizou-se uma série de entrevistas com bibliotecárias da UFN. E assim, com base no que foi mencionado sobre os livros jornais e revistas, elaborou-se o Diagrama de Classe de Domínio (Figura 3), que serviu para organizar os diferentes tipos de materiais, seus atributos principais (como título, autor, idioma, editora, código e localização física) e como eles se relacionam entre si. Esse modelo garantiu que todas as regras de organização informadas pelas profissionais fossem transferidas corretamente para o banco de dados.

O Diagrama de Classe de Domínio é uma representação visual de classes conceituais ou de objetos do mundo real dentro de um contexto específico. De forma simples, funciona como um dicionário visual das abstrações notáveis, do vocabulário e do conteúdo de informações de um determinado domínio. Ele representa essencialmente três elementos: classes conceituais (ideias coisas ou entidades significativas do mundo real), associações (relacionamentos significativos entre as classes) e atributos (dados pertencentes a uma classe conceitual) (Larman, 2001, p. 127-129).

Figura 3 - Diagrama de Classe de Domínio



Fonte: Construção dos Autores

A Figura 3 representa o Diagrama de Classe de Domínio modelado para a gestão do acervo do Museu Treze de Maio. No centro do modelo se encontra a entidade genérica ItemAcervo, da qual derivam, por meio de especialização: Livro, Revista e Jornal. O diagrama mostra como esses itens interagem com o ecossistema do museu, estabelecendo associações *many-to-many* (muitos para muitos) com Colaborador e Assunto, além de registrar o vínculo com Editora e sua Localização física. Por fim, o modelo conecta cada ItemAcervo aos seus respectivos Exemplares e a um registro de Histórico, o qual monitora as movimentações e ações executadas pelos Usuários do sistema.

2.3 PROCESSO DE DESENVOLVIMENTO

As próximas etapas do projeto partiram da disciplina de Projeto de Software onde foi estabelecido que o desenvolvimento do projeto seguiria uma metodologia ágil, assim foi determinado dois modelos: a metodologia ágil Desenvolvimento Guiado por Funcionalidades (FDD - *Feature-Driven Development*) que se baseia na gerência do projeto dividindo os

problemas em funcionalidades (pequenas funções); E o método *Kanban*, uma técnica visual usada para gerenciamento de fluxo de produção em etapas de desenvolvimento.

A metodologia FDD trabalha seu fluxo de desenvolvimento com cinco processos: 1°) desenvolvimento de um modelo geral, anteriormente realizado na etapa de levantamento de requisitos pelo Diagrama de Caso de Uso; 2°) construção de uma lista de funcionalidades, desenvolvida com base nos requisitos identificados; 3°) planejamento por funcionalidades, etapa que se define a data de conclusão para as funcionalidades, além disso, a atribuição das responsabilidades aos membros da equipe, separadas por classes; 4°) projetar por funcionalidade, foram desenvolvidos os Diagramas de Classe por funcionalidade e seus respectivos Diagramas de Sequência; 5°) desenvolver por funcionalidade, nesta etapa ocorre a codificação e integração da funcionalidade ao sistema. O detalhamento sobre do desenvolvimento das funcionalidades será abordado no Tópico 2.4. Nota-se que, os processos 4° e 5° ocorrem em um ciclo iterativo, com base nos processos anteriores cada funcionalidade é projetada e desenvolvida sequencialmente, assim que concluída a próxima entra em execução. (Palmer; Felsing, 2002, p. 56-58 e 63-72)

O *Kanban* participa como complemento à metodologia FDD, com o objetivo de auxiliar o planejamento das funcionalidades quanto ao gerenciamento de fluxo de produção. Como meio, foi utilizada a ferramenta *Notion*, ambiente virtual para espaço de trabalho compartilhado (Notion, 2026). A partir disso, foi aplicado o *Kanban*, que consiste em colunas, para as etapas de desenvolvimento, e cartões, para as tarefas que avançam em cada etapa do processo. As etapas de desenvolvimento básicas são “A fazer”, “Em Andamento” e “Concluído”. Cada cartão representa uma funcionalidade, para cada cartão foi definido: uma data de conclusão, membro responsável e grau de prioridade (baixo, médio ou alto). Assim, foi organizada, dividida e delegadas as funcionalidades entre a equipe para desenvolvimento simultâneo. (Pressman, 2021, p. 48-50)

2.4 IMPLEMENTAÇÃO DA SOLUÇÃO

No desenvolvimento do sistema, foi utilizada a linguagem orientada a objetos *Java*, que atua como uma plataforma de desenvolvimento (Oracle, 2026), combinado com o ecossistema *Spring Boot*, que simplifica o processo de produção ao fornecer recursos e configurações prontas (Spring, 2026).

As informações de classes e atributos definidas no Diagrama de Domínio precisam ser armazenadas em um banco de dados relacional, e, para que seja possível adicionar, acessar e processar os dados armazenados, é necessário utilizar um sistema de gerenciamento de banco de dados (SGBD). Nesse contexto, foi escolhido o *MySQL*

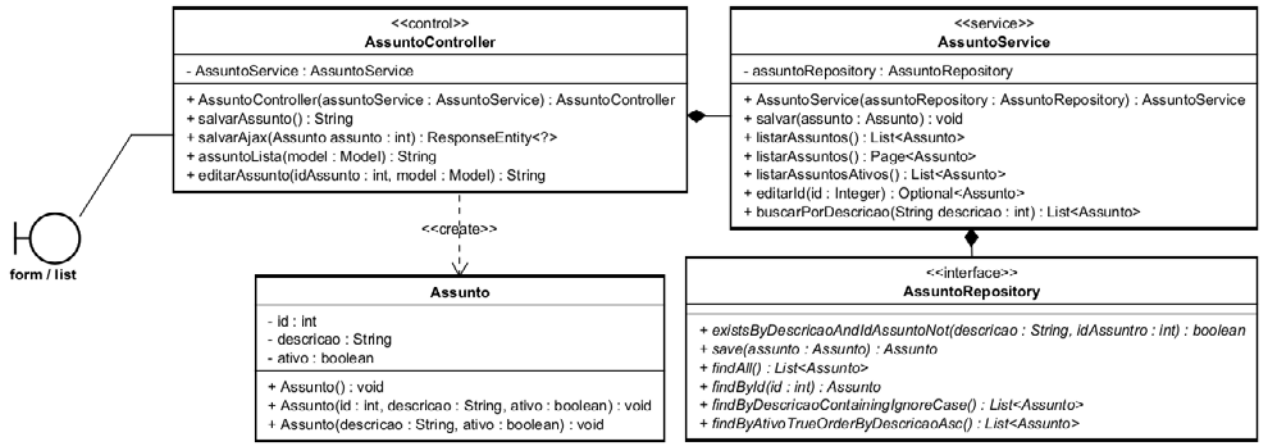
como SGBD, onde a manipulação dos desses dados é realizada por meio da Linguagem de Consulta Estruturada (SQL - *Structured Query Language*) (MySQL, 2026). Além disso, a integração com o banco de dados foi viabilizada pelo *Spring JPA* (*Java Persistence API*) que cria ponte inteligente entre os objetos Java e o banco de dados relacional (Spring, 2026).

Para a estrutura do sistema foi utilizado o padrão de arquitetura *Model-View-Controller* (MVC) que separa a aplicação em três componentes interconectados, favorecendo o desenvolvimento de cada parte do sistema independentemente. Complementarmente, foram implementados os padrões de arquitetura de camadas, o *Service Pattern* e o *Repository Pattern, c*om o objetivo de organizar sistematicamente as responsabilidades do sistema.

No padrão arquitetural MVC são atribuídas as seguintes funções: *Model* - define a estrutura de dados das entidades do sistema identificadas pelo Diagrama de Classe de Domínio; *View* - apresenta a interface ao usuário carregando os elementos da model; *Controller* - traduz as interações do usuário e valida requisições e respostas; *Service* - concentra a regras de negócios e regras da aplicação, atua como intermediário entre os dados e o *controller*; *Repository* - é intermediário *entre* o sistema e o banco de dados, isolando a camada de negócio dos detalhes técnicos de infraestrutura e de comandos de banco de dados. (Oracle, 2026; Evans, 2003, p. 104-107 e 136-138).

Como próximo passo, com base nas entidades identificadas pelo Diagrama de Domínio, foi elaborado os Diagramas de Classe para as entidades do sistema. Os Diagramas de Classe são utilizados para desenvolvimento de um modelo de sistema orientado a objetos, com objetivo de destacar as classes, atributos, operações e as associações entre classes. Assim, foi desenvolvido por meio do Diagrama de Classe a modelagem baseada nas funcionalidades do sistema, conforme abaixo. (Sommerville, 2018, p. 131-132 e 183-184).

Figura 4 - Diagrama de Classe para Entidade Assunto.



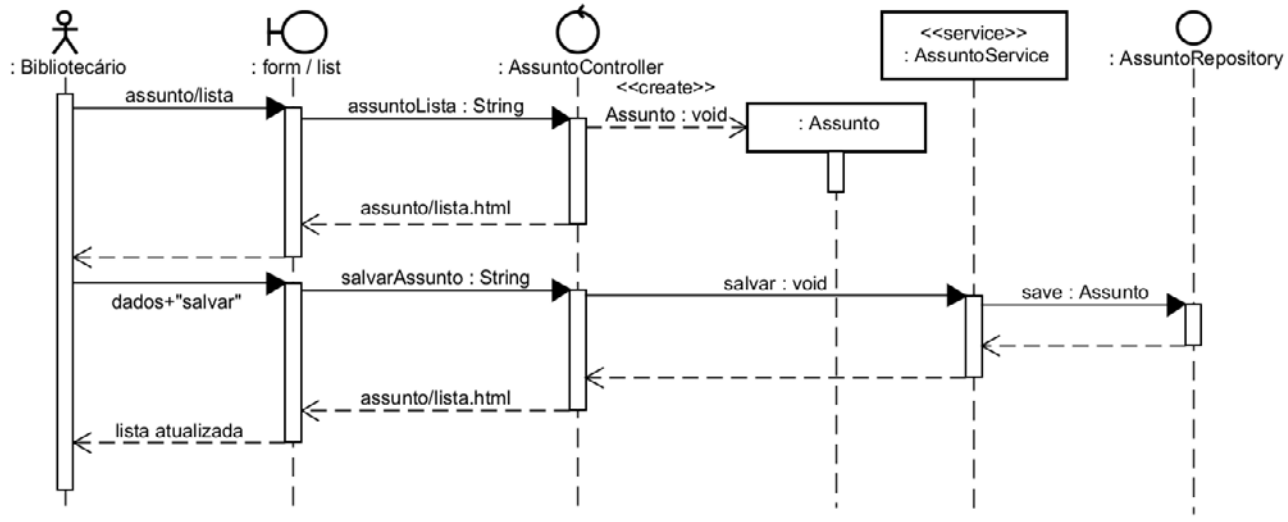
Fonte: Construção dos Autores.

Na Figura 4, foi construído o Diagrama de Classe para a funcionalidade de gerenciamento de assunto, desenvolvido utilizando a ferramenta de modelagem ***Astah UML***. A ***view***, representada pelo símbolo circular atribuído o nome "***form/list***" se refere tanto à visualização do formulário quanto à listagem do sistema. Os demais padrões são representados pelos retângulos, cada um possui: tipo, nome, atributos e operações.

A ***model*** é retratada em «Assunto» e contém os atributos dessa funcionalidade, bem como os seus construtores. Em «Assunto Controller", que se refere à ***Controller***, ocorre o recebimento das requisições feitas no sistema e o encaminhamento destas para a classe Assunto Service. A "Assunto Service", que representa a ***Service***, engloba as regras de negócio. E, por fim, a "Assunto Repository", que retrata a ***Repository***, é responsável pelas operações de acesso e persistência realizadas no banco de dados do sistema. Ademais, no diagrama também é representada a relação de composição - losangos preenchidos - entre as classes "Assunto Controller" e "Assunto Service", e entre "Assunto Service" e "Assunto Repository", o que exprime uma relação na qual uma classe mantém a outra como parte de sua estrutura.

A partir do Diagrama de Classe é desenvolvido o Diagrama de Sequência para modelar as interações entre atores e objetos em um sistema além das interações entre os próprios objetos. O modelo exibe a sequência de interações respectivas a uma funcionalidade do sistema em uma estrutura de fluxo temporal (Pressman, 2016, p. 205-206; Sommerville, 2018, p. 130-131).

Figura 5 - Diagrama de Sequência para Funcionalidade "Cadastrar Assunto".



Fonte: Construção dos Autores.

Na Figura 5, o diagrama representa a funcionalidade "Cadastrar Assunto", desenvolvido utilizando *Astah UML*. O diagrama é formado pelo ator "Bibliotecário" e sua interação com as etapas do fluxo, aquelas que modelamos no Diagrama de Classe. Cada etapa representada pelo nome e símbolo possui uma linha para traçar o tempo, cada retângulo vertical

indica o tempo entre a solicitação e resposta que o ator/objeto espera. Flechas conectam as etapas e evidenciam o fluxo das mensagens, elas possuem legendas indicando qual operação foi utilizada. O fluxo vai desde a tela inicial do cadastro quando o bibliotecário a acessa, até o envio do formulário para o banco de dados e a validação da solicitação.

No desenvolvimento das interfaces do sistema *web*, foram utilizados o *framework Bootstrap* e o mecanismo de *template Thymeleaf.* O *Bootstrap* é um framework que possui uma biblioteca repleta de recursos de *HyperText Markup Language* (HTML), *Cascading Style Sheets* (CSS) e *JavaScript* (JS) que potencializam a criação de páginas de maneira rápida e responsiva (Bootstrap, 2026). Já o *Thymeleaf* é um motor de templates para o ecossistema *Java* voltado ao processamento de páginas no lado do servidor, usado para a geração de páginas HTML dinâmicas, atualizando em tempo real com os registros vindo do banco de dados (Thymeleaf, 2026).

Para questões de segurança e login foi utilizado o *framework Spring Security. D*e acordo com Spring (2026) o *Spring Security* se concentra em fornecer autenticação e autorização para aplicações *Java.* No processo de autenticação ocorre a validação da credencias, e-mail e senha, do usuário que está tentando acessar ao sistema, após o usuário estar logado no sistema ocorre a verificação das permissões que ele possui, determinando assim quais ações ele poderá realizar. Por fim, o ambiente de desenvolvimento integrado (IDE) adotado para a construção do projeto foi o *IntelliJ IDEA*, devido ao seu suporte nativo ao ecossistema *Spring* (Jetbrains, 2026). Já o planejamento visual e a prototipação das interfaces de navegação que foram previamente apresentadas para os usuários para a validação dos requisitos do sistema foram produzidas na ferramenta *Figma*, uma área de trabalho para o processo de desenvolvimento de produtos. (Figma, 2026).

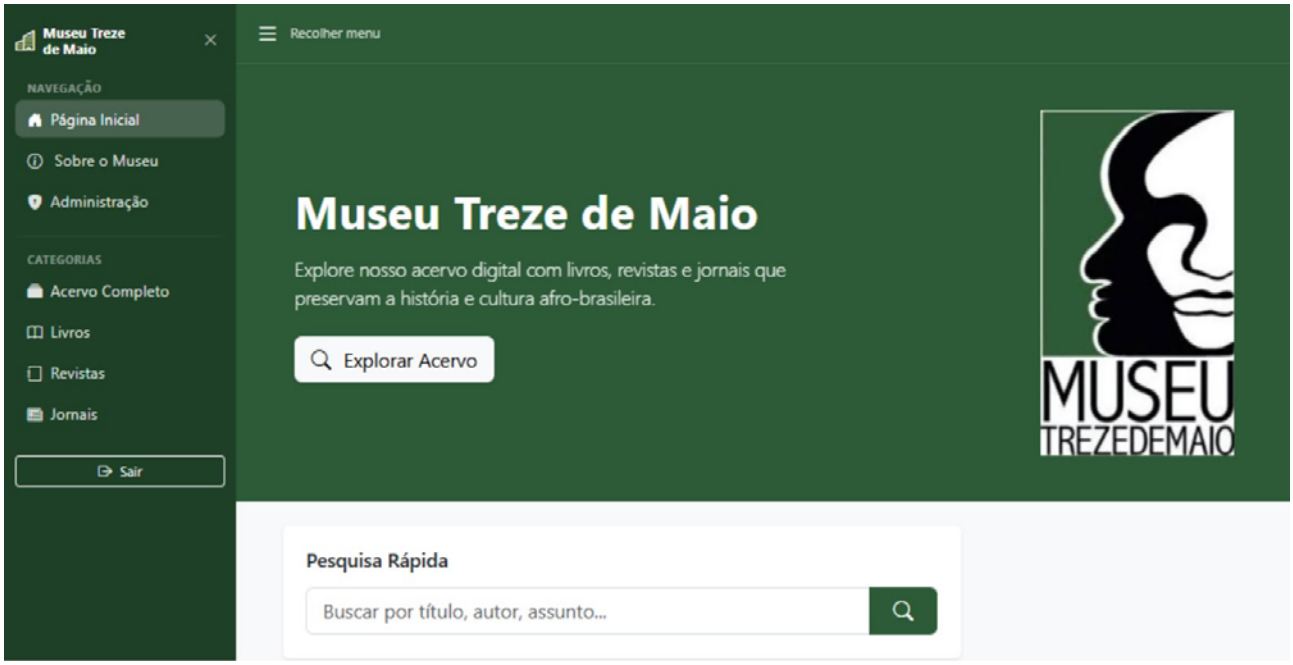
3 RESULTADOS E DISCUSSÕES

Desenvolveu-se, como resultado, uma versão funcional do sistema web de catalogação e gestão, contendo um conjunto de funcionalidades proveitosas para o gerenciamento do acervo do Museu Treze de Maio. Dentre as funcionalidades implementadas, destacam-se a autenticação dos usuários, as operações de cadastro e edição de registros (livros, revistas e jornais), a área de consulta com a suporte para pesquisa, além do painel de controle e gerenciamento de usuários do sistema.

O sistema dispõe de interfaces gráficas responsivas que apresentam todas as funcionalidades previamente propostas. A Figura 6 apresenta a tela inicial, que serve como ponto de entrada para acessar as funcionalidades de catalogação e consulta. A partir dessa interface, o usuário pode acessar a tela de pesquisa e consulta (Figura 7), que possibilita a

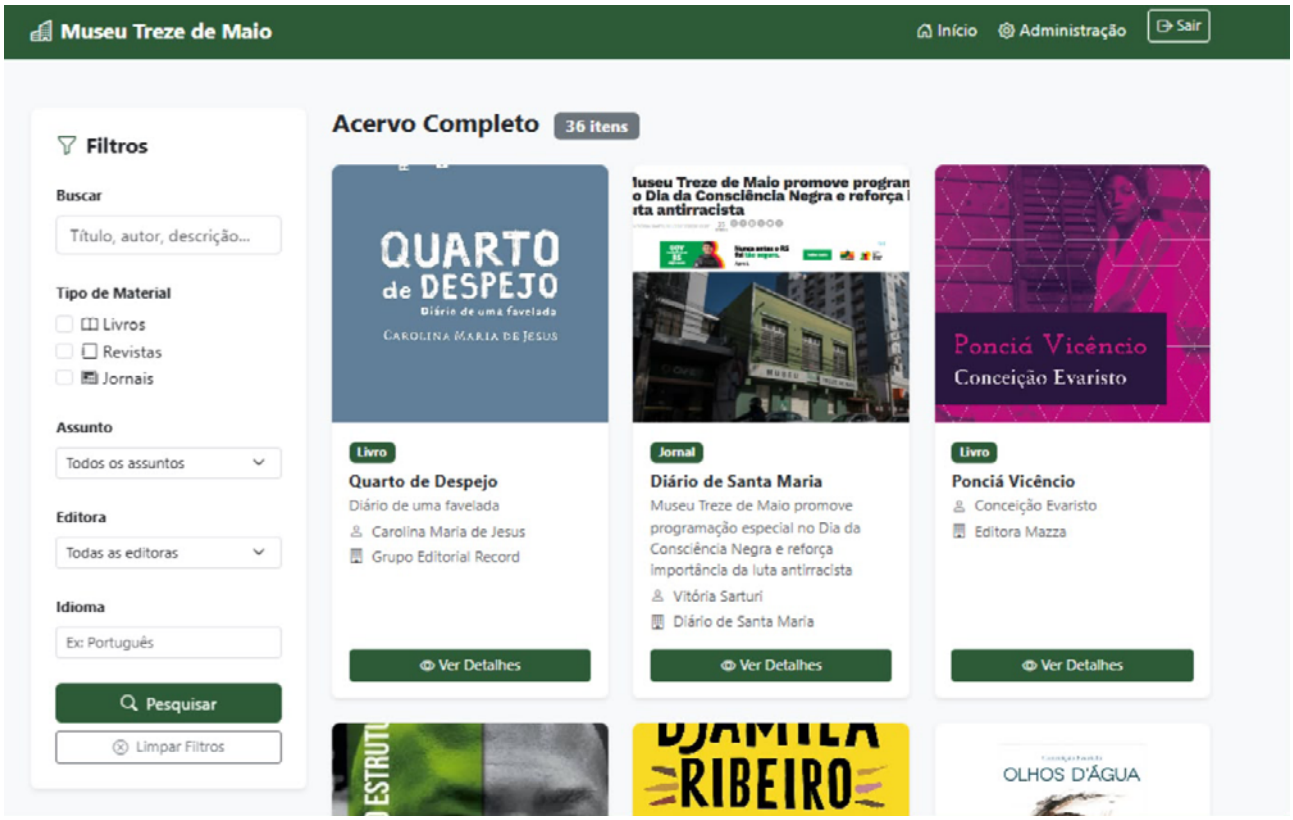
localização de itens por meio de filtros como título autor, tipo de material e assuntos vinculados, exibindo informações detalhadas da obra e sua localização física no acervo.

Figura 6 -Tela Inicial do Sistema Web.



Fonte: Construção dos Autores.

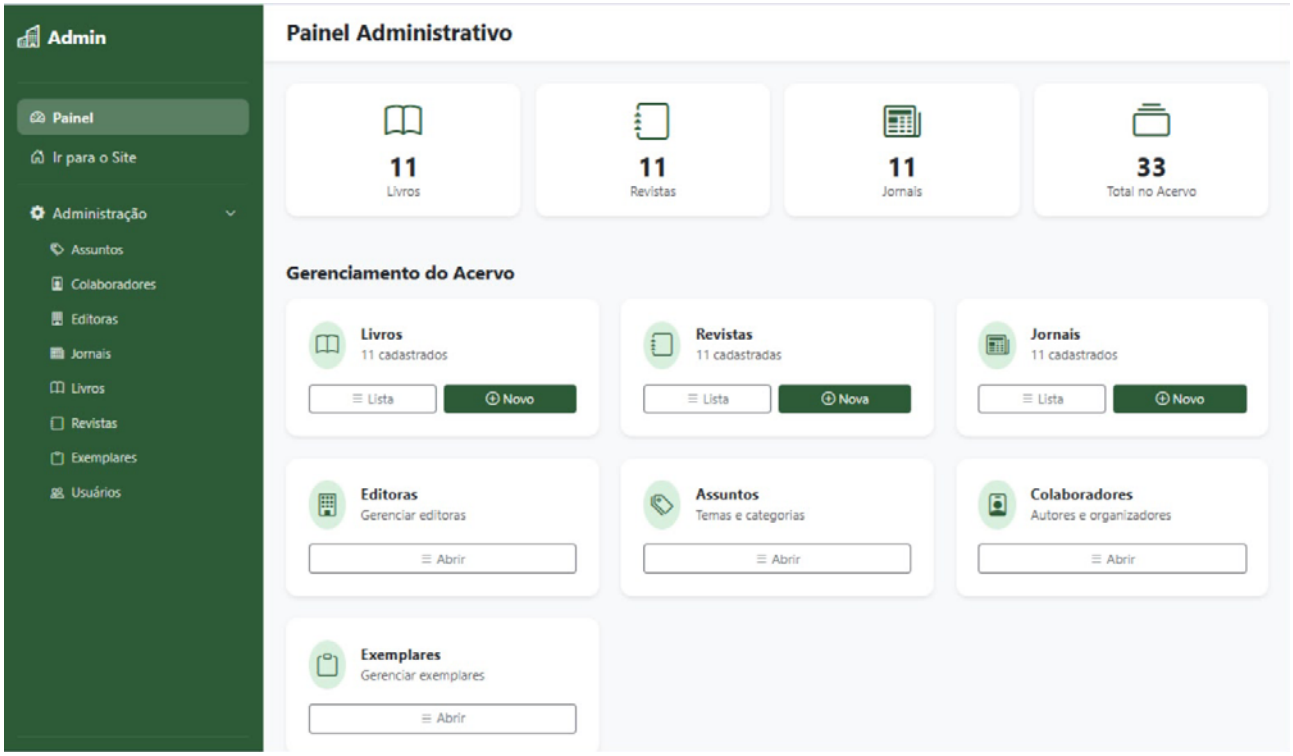
Figura 7 -Tela de Pesquisa e Consulta do Acervo



Fonte: Construção dos Autores.

Para o controle e monitoramento do acervo, desenvolveu-se a área administrativa contendo um painel administrativo (Figura 8). Esta interface restrita aos usuários autenticados (bibliotecários e administradores) possui indicadores quantitativos que exibem o total de livros, revistas e jornais cadastrados. Também, a tela contém atalhos para listagem e cadastros de novos registros, bem como controle das entidades auxiliares como editoras, assuntos e colaboradores, sendo o atalho de gestão de usuários disponível exclusivamente para o perfil administrador.

Figura 8 - Tela do Painel Administrativo do Sistema Web.



Fonte: Construção dos Autores.

Considerando o problema inicialmente apresentado, a ausência de um sistema digital de catalogação que dificultava a busca, organização e divulgação dos itens no acervo, cada uma das funcionalidades desenvolvidas buscou responder uma necessidade específica identificada junto com a equipe do museu. A Tabela 1 comprova essa relação, associando as principais necessidades identificadas às soluções implementadas no sistema, bem como às evidências que comprovam seu atendimento.

Tabela 1 - Soluções para as Necessidades Identificadas.

Necessidade identificada	Solução implementada	Evidência
Organizar o acervo	Cadastro estruturado de livros revistas e jornais	Tela de cadastro / banco de dados
Facilitar a localização	Pesquisa com filtros	Figura 8
Controlar acesso administrativo	Autenticação e perfis	Sistema implementado
Gerenciar registros	Cadastro, edição e inativação	Painel administrativo
Disponibilizar consulta	Interface web de pesquisa	Figura 8

Fonte: Construção dos Autores

Conforme demonstrado na Tabela 1, os resultados obtidos demonstram o atendimento às necessidades identificadas no início do projeto. A ausência de um sistema digital era o principal problema que dificultava a busca, organização e divulgação do patrimônio do museu. A solução implementada resolve essa problemática ao centralizar as informações em uma base estruturada acessível via web.

A validação da solução ocorreu por meio de uma sessão de apresentação do sistema finalizado, somada a testes práticos com a equipe responsável pelo museu e as bibliotecárias da instituição. Durante a demonstração, os participantes interagiram diretamente com as interfaces, executando tarefas como a busca no acervo, o cadastro e a edição de itens. Essa etapa permitiu que os usuários finais realizassem uma avaliação preliminar a respeito da usabilidade do sistema e clareza de informações.

4 CONCLUSÃO

O presente trabalho apresenta o desenvolvimento de um sistema web para catálogo e gestão do acervo do Museu Treze de Maio, resultado de uma colaboração entre o curso de Ciência da Computação e o Museu Treze de Maio. O sistema desenvolvido soluciona as necessidades do acervo em catalogar itens digitalmente, facilitando o processo de organização, gestão e busca de itens.

O projeto teve êxito na integração entre teoria e prática, onde foi possível aplicar os conceitos de engenharia de software e implementação de banco de dados na construção de um sistema completo para atender problemas reais. Ademais, visando as necessidades dos usuários, destaca-se a utilização de técnicas para mapeamento e documentação do problema por meio das visitas técnicas e entrevistas, além do uso de metodologias ágeis, padrões de arquitetura e ferramentas atuais.

Como perspectivas futuras, encontram-se em desenvolvimento os ajustes finais para a aplicação do sistema no ambiente de produção para aderir, por meio de feedbacks, melho-

rias contínuas. Além disso, pretende-se ampliar a variedade de materiais disponíveis para catalogação, podendo contemplar diferentes tipos de itens do acervo.

Conclui-se, portanto, que o projeto atende ao problema que motivou o trabalho, ao proporcionar à equipe do Museu Treze de Maio uma ferramenta capaz de catalogar e disponibilizar os itens do acervo. Ademais, a ação evidencia a relevância universitária ao conectar a Universidade Franciscana à comunidade, promovendo aprendizado prático e significativos para os acadêmicos envolvidos e valorizando a história do museu.

REFERÊNCIAS

BEZZERA, Eduardo. **Princípios de análise e projeto de sistemas com UML**. 2. ed. Rio de Janeiro: Elsevir, 2007.

BOOTSTRAP. **Bootstrap**: The most popular HTML, CSS, and JS library in the world. [S. l.]: 2026. Disponível em: https://getbootstrap.com/. Acesso em: 20 ago. 2026.

CHANGE VISION. **Inc. Astah UML**. [S. l.]: 2026. Disponível em: https://astah.net/products/astah-uml. Acesso em: 27 ago. 2026.

EVANS, Erick. **Domain-Driven Design**: Tackling complexity in the heart of software. Upper Saddle River, NJ: Addison-Wesley, 2003.

FIGMA. **Figma**: A ferramenta de design colaborativa para a criação de produtos digitais. [S. l.]: 2026. Disponível em: https://www.figma.com/pt-br/. Acesso em: 20 ago. 2026.

JETBRAINS. **IntelliJ IDEA**: The Leading Java and Kotlin IDE. [S. l.]: 2026. Disponível em: https://www.jetbrains.com/idea/. Acesso em: 20 ago. 2026.

LARMAN, Craig. **Applying UML and Patterns**: An Introduction to Object-Oriented Analysis and Design and the Unified Process. 2nd ed. Upper Saddle River: Prentice Hall, 2001.

NOTION. **Introdução aos espaços de trabalho**. [S. l.]: 2026. Disponível em: https://www.notion.com/pt/help/intro-to-workspaces. Acesso em: 27 ago. 2026.

ORACLE. **Java Documentation**. [S. l.]: 2026. Disponível em: https://docs.oracle.com/en/java/. Acesso em: 20 ago. 2026.

ORACLE. **Java SE Application Design With MVC**. [S. l.]: 2026. Disponível em: https://www.oracle.com/technical-resources/articles/javase/mvc.html. Acesso em: 27 ago. 2026.

ORACLE. **MySQL Documentation**: Reference Manual. [S. l.]: 2026. Disponível em: https://dev.mysql.com/doc/. Acesso em: 20 ago. 2026.

PALMER, Stephen R.; FELSING, John M.. **A Practical Guide to Feature Driven Development.** Upper Saddle River, NJ: Prentice-Hall, 2002.

PRESSMAN, Roger S.; MAXIM, Bruce R.. **Engenharia de Software**: Uma Abordagem Profissional. 8. ed. Porto Alegre: AMGH, 2016.

PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de Software**: Uma Abordagem Profissional. 9. ed. Porto Alegre: AMGH, 2021.

SILVA, Armando Malheiro da; RIBEIRO, Fernanda. **Das "ciências" documentais à ciência da informação**: ensaio epistemológico para um novo modelo formativo. Porto: Edições Afrontamento, 2002.

SOMMERVILLE, Ian. **Engenharia de software**. 10. ed. São Paulo: Pearson, 2018. E-book. Disponível em: https://plataforma.bvirtual.com.br. Acesso em: 15 ago 2026.

SPRING. **Spring Boot**. [S. l.]: 2026. Disponível em: https://spring.io/projects/spring-boot. Acesso em: 20 ago. 2026.

SPRING. **Spring Data JPA**: Reference Documentation. [S. l.]: 2026. Disponível em: https://spring.io/projects/spring-data-jpa. Acesso em: 20 ago. 2026.

SPRING. **Spring Security**. [S.l.]: 2026. Disponível em: https://spring.io/projects/spring-security. Acesso em: 20 ago. 2026.

UNIVERSIDADE FRANCISCANA. **Extensão UFN.** Santa Maria: 2026. Disponível em: https://site.ufn.edu.br/pagina/extensao-ufn. Acesso em: 09 ago. 2026.

WEBER, L. I.; ZANINI, M. C. C. Museu Treze De Maio: Um Espaço de Memória e Identidade Negra em Santa Maria/Rs. **Identidade!**, [S. l.], v. 19, n. 1, p. 44-54, 2022. Disponível em: https://revistas.est.edu.br/Identidade/article/view/1774. Acesso em: 09 ago. 2026.

AUTOMATED GUIDED VEHICLE PROTOTYPE FOR INDUSTRY 4.0 LOGISTICS

PROTÓTIPO DE VEÍCULO AUTOGUIADO PARA A LOGÍSTICA DA INDÚSTRIA 4.0

Mauricio Pereira Braga[1], Luiz Batista Cardoso[2], Gabriel Azenha Fachim[3], Mirkos Ortiz Martins[4], Alessandro André Mainardi de Oliveira[5] e Herysson Rodrigues Figueiredo[6]

ABSTRACT

Industry 4.0 is transforming industrial processes through advanced technologies such as the Internet of Things and Artificial Intelligence, which drive automation and efficiency in manufacturing operations. This study presents a solution based on an Automated Guided Vehicle to enhance material transport in industrial environments, minimizing the need for human intervention and increasing safety. The proposed vehicle integrates navigation sensors and tag-based identification systems, providing precision and autonomy in load handling. In addition to a functional vehicle prototype, a web application was developed for monitoring and control. The project was validated in a simulated industrial environment, which allowed its effectiveness and operational applicability to be evaluated.

Keywords: Smart factory; Mobile robotics; RFID; MQTT; Web application.

RESUMO

A Indústria 4.0 vem transformando os processos industriais por meio de tecnologias avançadas, como a Internet das Coisas e a Inteligência Artificial, que impulsionam a automação e a eficiência

1 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana (UFN). E-mail: mauricio.pereira@ufn.edu.br. ORCID: https://orcid.org/0009-0003-4346-5836.
2 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana (UFN). E-mail: luiz.bcardoso@ufn.edu.br. ORCID: https://orcid.org/0009-0003-7344-4008.
3 Acadêmico de Pós-Graduação em Nanociências. Universidade Franciscana (UFN). E-mail: gabriel.afachim@ufn.edu.br. ORCID: https://orcid.org/0009-0002-8511-4025.
4 Professor dos cursos de computação da UFN. Universidade Franciscana (UFN). E-mail: mirkos@ufn.edu.br. ORCID: https://orcid.org/0000-0002-3983-1624.
5 Professor dos cursos de computação da UFN. Universidade Franciscana (UFN). E-mail: alessandro.andre@ufn.edu.br. ORCID: https://orcid.org/0009-0002-8579-2635.
6 Professor dos cursos de computação da UFN. Universidade Franciscana (UFN). E-mail: herysson.figueiredo@ufn.edu.br. ORCID: https://orcid.org/0009-0009-2615-9785.

das operações de manufatura. Este trabalho apresenta uma solução baseada em um Veículo Autoguiado para aprimorar o transporte de materiais em ambientes industriais, reduzindo a necessidade de intervenção humana e aumentando a segurança. O veículo proposto integra sensores de navegação e sistemas de identificação por etiquetas, proporcionando precisão e autonomia na movimentação de cargas. Além do protótipo funcional do veículo, foi desenvolvida uma aplicação web para monitoramento e controle. O projeto foi validado em um ambiente industrial simulado, o que permitiu avaliar sua efetividade e aplicabilidade operacional.

Palavras-chave: *Fábrica inteligente; Robótica móvel; RFID; MQTT; Aplicação web.*

1 INTRODUCTION

The fourth industrial revolution, known as Industry 4.0, was launched by the German government during the Hannover Industrial Fair in 2011 (Kagermann; Wahlster; Helbig, 2013). Industrial automation uses control and information technologies to increase efficiency, precision, and safety in processes, replacing manual practices with automated systems (Pires, 2019). This era is characterized by the integration of technologies such as the Internet of Things (IoT), cyber-physical systems, and Artificial Intelligence. These innovations are transforming manufacturing environments by establishing new standards for automation and operational efficiency, emphasizing the interconnectivity and digitalization of production systems (Ustundag; Cevikcan, 2018).

Within this technological landscape, Automated Guided Vehicles (AGVs) emerge as a prime example of the practical application of Industry 4.0 principles. AGVs are material transport technologies that use autonomous vehicles to follow predefined routes in locations such as warehouses and factories, operating without physical rails, which allows for less intrusive integration into workspaces (Groover, 2014). Used in contexts ranging from driverless train operations to complex flexible manufacturing systems, AGVs coordinate the transport of materials among multiple loading and unloading points, integrating with automated systems to improve logistics and workflow.

However, the operational safety of these vehicles is a priority; therefore, strict adherence to standards is imperative. The Verein Deutscher Ingenieure (VDI) 2510 standard specifies requirements for the safe operation of AGVs, including limiting the maximum speed to 1 m/s (3.6 km/h) to protect people and structures (Verein Deutscher Ingenieure, 2021). The ISO 3691-4 standard, in turn, specifies safety requirements and the means for their verification for driverless industrial trucks and their systems (International Organization for Standardization, 2023). Additionally, Regulatory Standard No. 26 (NR-26) recommends the use of the color yellow for transport equipment to improve visibility and operational safety

(Ministério do Trabalho e Emprego, 2012). Adherence to these standards guides safe operations and protects both workers and facilities.

The motivation for this work lies in the need to enhance operational efficiency in industrial environments, reducing the dependence on manual labor and the risks of errors and accidents. Automation in load transportation is a component of this transformation, with the potential not only to increase workplace safety but also to reduce operating costs. To address these challenges, this work aims to improve internal logistics by developing a functional prototype of an Automated Guided Vehicle, including the creation and integration of a web-based application for its management and control.

By implementing a technological solution that integrates sensors for obstacle detection and RFID technology, the proposed system is designed to navigate autonomously within industrial complexes, using Radio-Frequency Identification (RFID) tags to support dynamic decision-making (Liu; Fan; Song, 2013). This integration of physical and digital systems is intended not only to increase the accuracy of locomotion and load delivery but also to validate the efficiency and safety of the integrated system in a simulated industrial environment.

2 RELATED WORK

This section presents articles related to the use and development of AGVs and their associated technologies, which contributed to the development of this project. The articles presented here provide information on the use of RFID for navigation and recognition, ideas for modeling the structure of an AGV prototype, and the connection between AGVs and IoT.

2.1 SMART AUTOMATED GUIDED VEHICLES FOR MANUFACTURING IN THE CONTEXT OF INDUSTRY 4.0

Mehami, Nawi, and Zhong (2018) explore the adoption of AGVs in smart factories, equipping two AGVs with RFID readers for navigation and for the recognition of tags placed on the floor and on objects, which improved internal logistics. The results showed that integrating AGVs with RFID technology can considerably improve operational efficiency in factory environments, and this integration represents the main contribution of the article to the prototype developed in this work.

2.2 MODELING AND IMPLEMENTATION OF AN AUTOMATED GUIDED VEHICLE PROTOTYPE FOR SMART FACTORIES

Sasamoto *et al.* (2021) investigate the modeling and implementation of an AGV for material transport. The design uses differential drive and laser-cut steel plates, ensuring structural robustness capable of supporting up to 700 kg, confirmed by Finite Element Analysis. The tests validated the AGV's effectiveness in navigating automatically through the factory, transporting heavy loads, and avoiding obstacles; the robustness of the mechanical structure was the main element that contributed to the present work.

2.3 IOT-BASED AGV SYSTEM FOR HEALTHCARE

Ananthi *et al.* (2022) present an AGV adapted for hospital environments, aiming to reduce physical contact during the COVID-19 pandemic. The AGV distributes medication, measures vital signs, and allows video consultations, transmitting information to healthcare professionals via IoT. Beyond minimizing contagion risks during the pandemic, the system integrates effectively into daily hospital operations, and this capacity for operational integration stood out as the most influential characteristic for the present work.

3 METHODOLOGY

Agile methodology emerged to overcome the limitations of traditional software development methods, which were bureaucratic and inflexible, especially for smaller projects. Instead of rigid planning and extensive documentation, agile methodology promotes incremental development, adapting to changes in requirements (Sommerville, 2011). For this work, the Kanban method was adopted.

The Kanban method is a management approach initially developed at Toyota and later adapted to the field of software development by David Anderson (2010). This methodology focuses on continuous improvement of processes and workflows, and its main tool is the Kanban board, which visualizes the workflow and facilitates task management across the different stages of the development process.

In this project, the board was used to manage the mechanical, electronic, and software tasks in a single workflow, which was necessary because the vehicle went through several modifications before reaching its final version. Two of these modifications illustrate how the method influenced the product. The first was the obstacle-avoidance functionality, initially defined as a deviation and later replaced by stopping upon detection, for reasons of safety

and practicality. The second was the removal of the reflectance sensor initially planned for line following, described in Section 4, which changed the navigation strategy to the RFID reader only. In both cases, the change was treated as a new card and the affected requirements were revised, keeping the scope visible and organized.

For the effective implementation of Kanban practices in this project, a case study was conducted to detail how the project operates, resulting in the elicitation and definition of the project requirements.

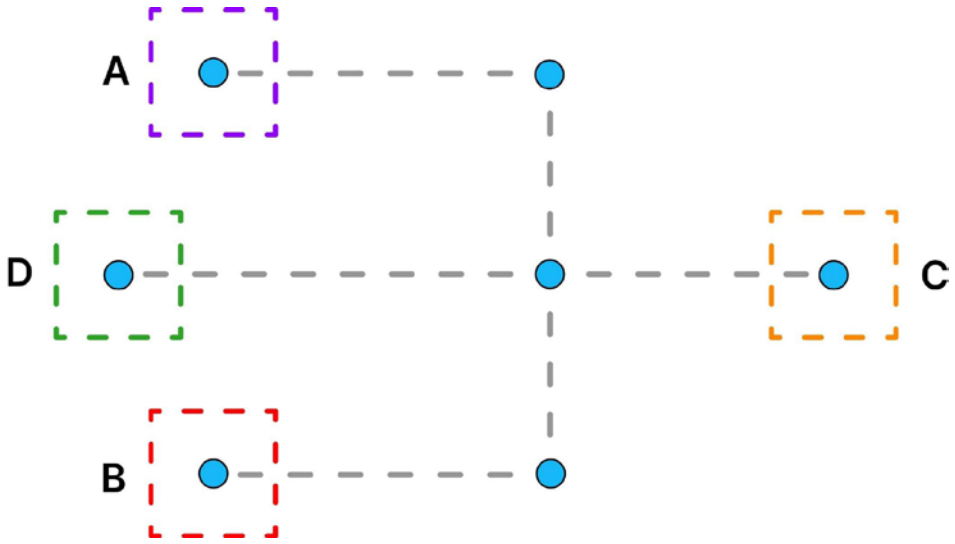
3.1 CASE STUDY

In this work, as illustrated in Figure 1, the AGV operates in an environment with a predefined trajectory that includes four main areas: unloading area A, unloading area B, the loading area, and the maintenance area.

Unloading areas A and B are the possible destinations for the items transported by the AGV, and the decision to unload in one of them is made based on the information received via RFID tags. The loading area (C) is the location where the AGV loads the items to be transported, and the maintenance area (D) is the space designated for maintenance and inspection of the vehicle, ensuring that it always operates under optimal conditions.

The AGV uses RFID tags, represented by blue circles, to decide which path to follow. Each RFID tag is programmed with specific information that guides the AGV along the route. When reading a tag, the AGV processes the information and decides whether to proceed to the loading, maintenance, unloading A, or unloading B area, or to continue to the next intersection.

Figure 1 - AGV route map.



Source: Prepared by the authors (2026).

3.2 REQUIREMENTS ELICITATION AND DEFINITION

The elicitation and definition of requirements for the AGV followed a structured methodology, encompassing the capture, approval, refinement, and detailing of requirements. Functional requirements were identified, such as autonomous navigation and load transportation, as well as nonfunctional requirements, focusing on quality attributes such as safety and performance (Pressman *et al.*, 2021).

In the capture phase, the potential functionalities of the vehicle within an industrial environment were explored, producing an initial list of requirements. These requirements were then submitted to approval, in which each item was reviewed to ensure alignment with the project objectives and technical feasibility, so that only relevant and executable requirements advanced to the next phase. During refinement, ambiguities were eliminated and the initial obstacle-avoidance functionality was changed so that the vehicle stops upon detecting an obstacle, for reasons of safety and practicality. The result was the objective definition of the functional and nonfunctional requirements presented in Tables 1 and 2, organized by priority. Finally, in the detailing phase, these requirements were decomposed into software, mechanical, and electronic designs, described through diagrams, electrical schematics, and prototypes.

Table 1 - Functional requirements.

RF01	Move in all directions.
RF02	Navigate along predefined lines on the floor.
RF03	Identify control points using RFID tags.
RF04	Transport loads.
RF05	Stop when identifying obstacles that block the path.
RF06	Communicate with the web application.
RF07	Return to the loading area after unloading.
RF08	Allow remote tracking of the vehicle's trajectory.
RF09	Allow the user to define the type of load to be transported.
RF10	Enable the user to stop/start operations remotely.
RF11	Display metrics related to the AGV (distance traveled; operating time; load type).

Source: Prepared by the authors (2026).

Table 2 - Nonfunctional requirements.

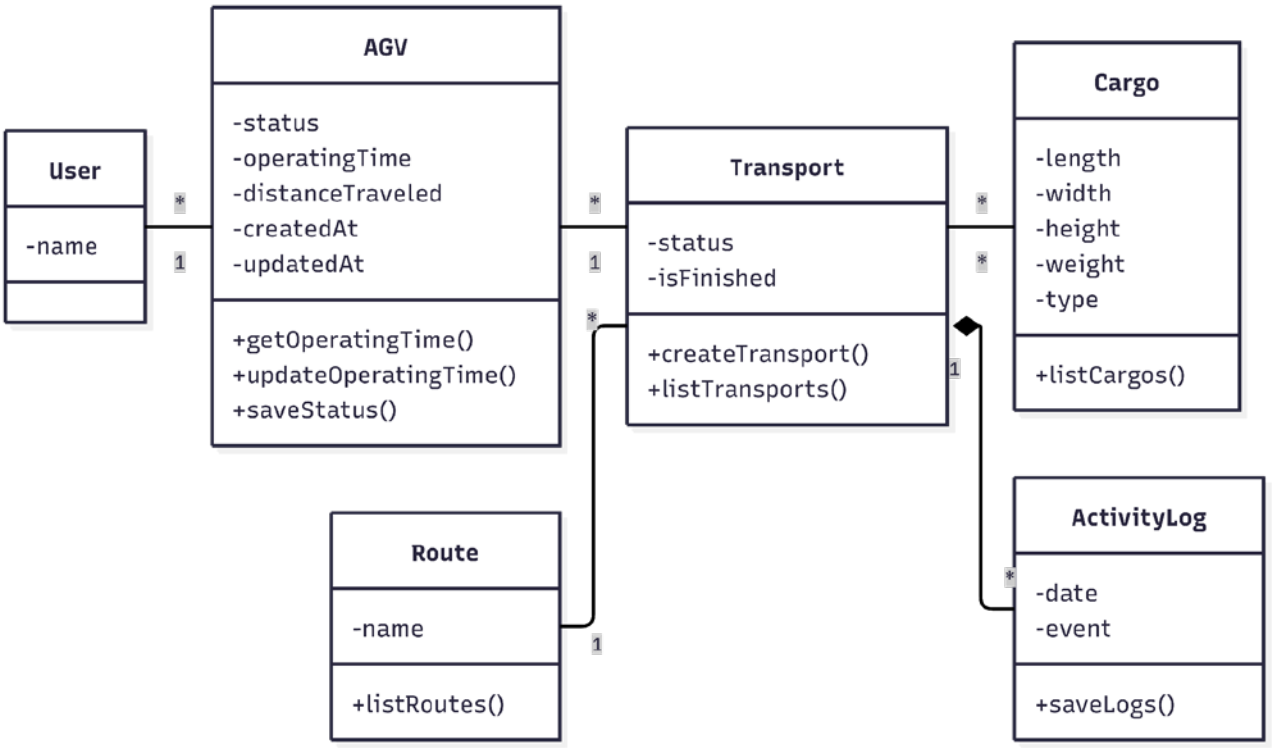
RNF01	The web system must facilitate remote diagnostics and repairs.
RNF02	The AGV must comply with all applicable safety standards (VDI 2510 and NR-26).
RNF03	The AGV's mechanical design must minimize accident risks, including appropriate protections against intentional misuse.
RNF04	The system must record all AGV actions.

Source: Prepared by the authors (2026).

3.3 DESIGN

In the initial design phase, a domain diagram was created to clarify the system's functionalities and relationships, highlighting the main classes: AGV, with attributes such as status and operating methods; User, who supervises the AGV; Route and Transport, which define paths and list routes; Load, to specify the transported items; and Activity Log, which maintains an event log for analysis. These elements form the foundation of the system, visually represented in Figure 2, which helps to understand their interdependencies.

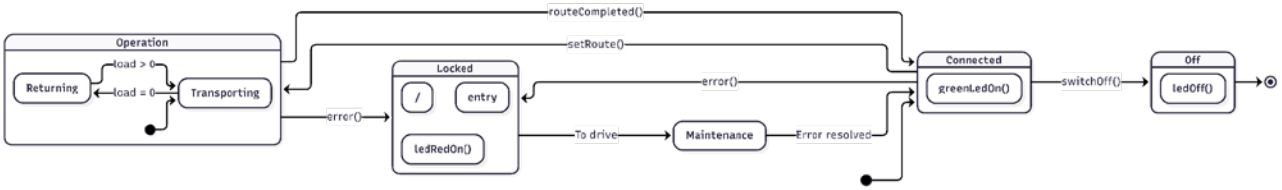
Figure 2 - Domain diagram.



Source: Prepared by the authors (2026).

To detail the AGV's operational behavior, a state machine diagram was used. This diagram describes the vehicle's main states, such as "Off", "On", "Operation", "Maintenance", and "Locked". During "Operation", the AGV alternates between the substates "Transporting" and "Returning", depending on whether a load is present. The representation also highlights transitions triggered by events, such as "desligar()", "erro()", and "rotaFinalizada()", as illustrated in Figure 3. These transitions include specific actions, such as turning on the green lights in the "On" state to signal that the AGV is powered on.

Figure 3 - State machine diagram.

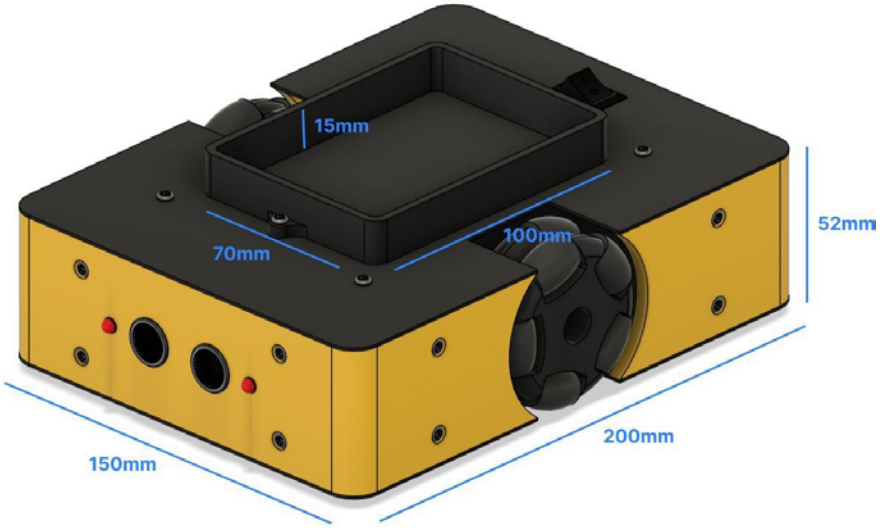


Source: Prepared by the authors (2026).

The design was also materialized in the three-dimensional model shown in Figure 4. This model was conceived to transport unit loads of up to 100 mm in length and 70 mm in width, with a support height of 15 mm. The AGV itself has dimensions of 200 mm in length, 150 mm in width, and 52 mm in height, meeting efficiency requirements for industrial environments.

The mechanical design also follows NR-26, which requires the color yellow to ensure visibility, contributing to operator safety by minimizing the risk of accidents in the work environment.

Figure 4 - Three-dimensional drawing of the AGV.

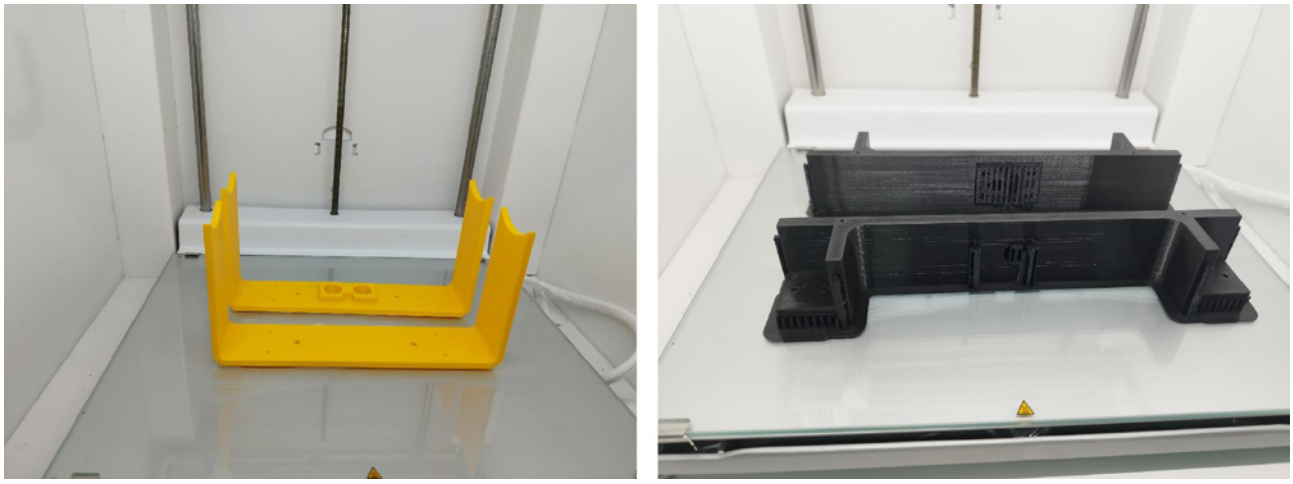


Source: Prepared by the authors (2026).

3.4 CONSTRUCTION

The prototype was built using Fused Deposition Modeling (FDM). This process enabled design adjustments, such as increasing the width from 150 mm to 180 mm to accommodate internal components. In addition, the overall height was reduced to 47 mm, and thicker covers were produced to improve strength. The AGV's mechanical structure is predominantly made of plastic, selected for its lightness and suitability for the project, as illustrated in Figure 5.

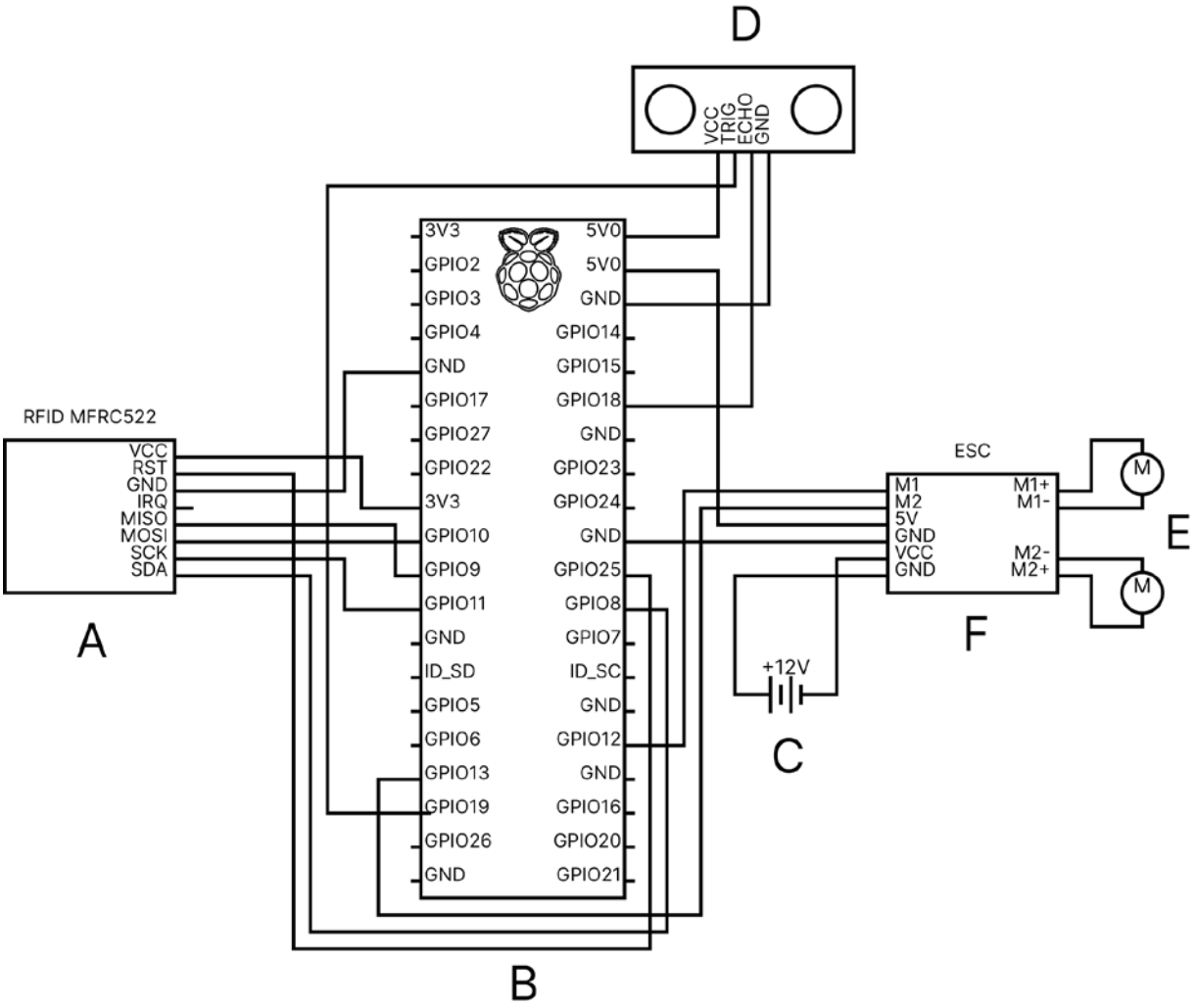
Figure 5 - Manufacturing process of the parts.



Source: Prepared by the authors (2026).

With the physical structure completed, the electronic system was implemented using a Raspberry Pi board programmed in MicroPython. The board handles the vehicle's entire control logic, using an ultrasonic sensor to detect obstacles and an RFID reader to identify reference points in the environment. Figure 6 presents the electronic schematic with the main components: the RFID reader (A), the Raspberry Pi (B), the battery (C), the ultrasonic sensor (D), the motors (E), and the motor driver (F). Although the schematic represents the driver as an L298N H-bridge, the prototype uses an Electronic Speed Controller with an integrated Battery Eliminator Circuit (BEC). This arrangement allows the use of a single 3S LiPo battery (C), with a nominal voltage of 11.1 V, to drive the motors while simultaneously stepping down the voltage to provide a stable 5 V supply to the Raspberry Pi, eliminating the need for a secondary power source.

Figure 6 - Electronic schematic.



Source: Prepared by the authors (2026).

AGV operation is managed by an algorithm that defines the route to be followed based on the specified destination. The method receives the destination acronym as input and sets the AGV's movement sequence, using predetermined points, where "CRG" represents the loading area, "ITM" is the central intersection, "ITA" and "ITB" are intersections A and B, and "DGA" or "DGB" are unloading areas A and B. The symbol "#" indicates the end of the route and enables returning to the initial point at the end of the trajectory.

Finally, the system was integrated with a web application developed in Next.js with the TypeScript Remote Procedure Call (tRPC) library, forming an end-to-end application. tRPC is a library for TypeScript that allows the creation of fully typed APIs, facilitating communication between client and server without the need for boilerplate code or additional code generation (tRPC, 2025). This architecture unifies the client and server sides, enabling direct integration between layers. The server connects to a relational database, ensuring persistence and retrieval of data related to the AGV.

Communication between system components occurs in two primary ways. Data exchange between client and server is performed via Hypertext Transfer Protocol (HTTP), while runtime communication between the Raspberry Pi and the client uses the Message Queuing Telemetry Transport (MQTT) protocol. MQTT is the preferred protocol for IoT and is defined by rules that allow devices to publish and subscribe to data. Because it is driven by events, it facilitates communication between decoupled devices through topics, mediated by an MQTT broker (OASIS, 2019). The application publishes the AGV destination to the MQTT topic "transport/start", translating the specified destination into an encoded message (*e.g.*, "DGA" for Unloading A), allowing the Raspberry Pi to receive and process the information.

4 RESULTS

The prototype was built to navigate based on RFID tags and to integrate with the web application, so that the proposed set of integrations could be validated. The tests were performed in the environment illustrated in Figure 1, with the tags fixed to the floor and the vehicle always starting from the loading area.

During development, the reflectance sensor presented functional limitations that led to its removal from the project. The imprecision of the measurements performed by the sensor, combined with compatibility problems with the Electronic Speed Controller, compromised the effectiveness of the system and made its use unfeasible. The RFID reader was therefore adopted as the only means of guiding the vehicle along the route, which means that RF02, defined as navigation along predefined lines on the floor, was not met.

This change imposed a fixed limit on load weight, since the turns toward the destination are controlled by time: heavier loads affect motor performance and, consequently, the position where the AGV stops after a turn. Despite this characteristic, the approach based on the RFID reader enabled a more consistent operation.

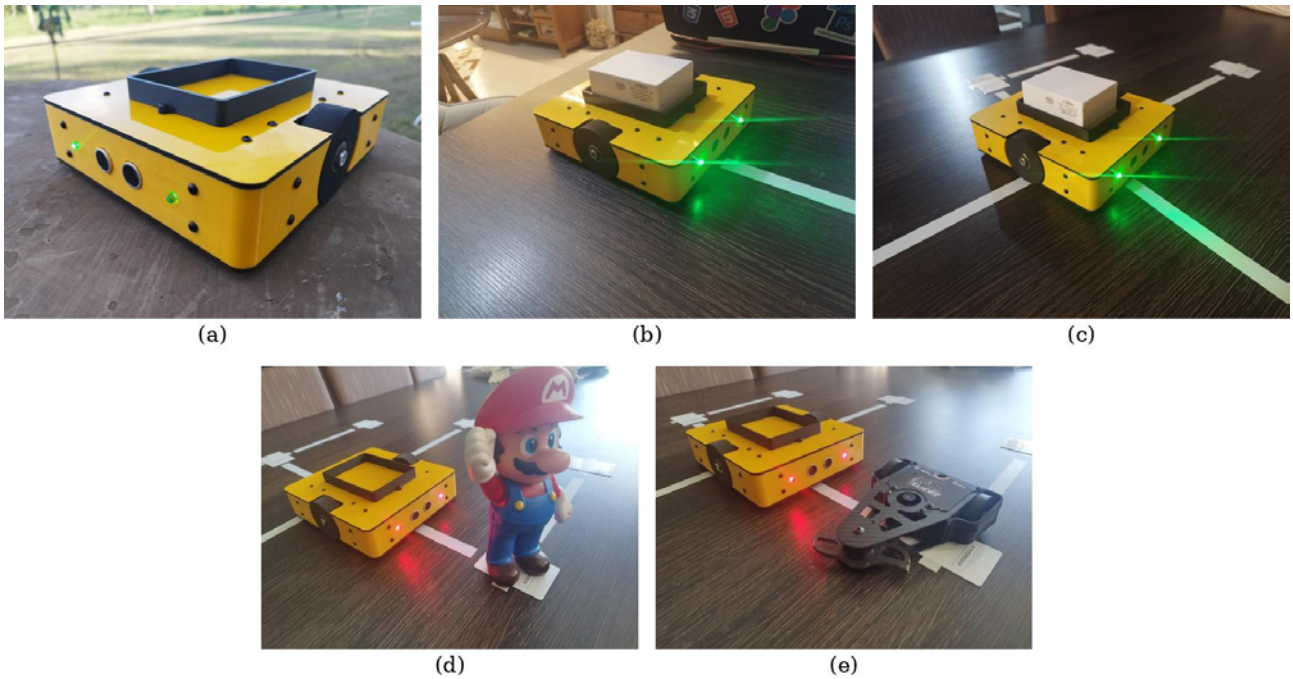
The AGV was built in its final version based on the design in Figure 4, with minor changes made during manufacturing to improve stability and internal organization. The prototype has six wheels: two main wheels coupled to the motors for traction and four support casters positioned at the lower corners of the vehicle. The RFID reader was installed on the lower front section. In Figure 7(a), the vehicle is shown with the green lights on, which indicate that it is powered on.

In the transport tests, considering a load 92 mm in length, 68 mm in width, 42 mm in height, and an approximate weight of 10 g, the AGV completed the route in about 6 seconds. Figures 7(b) and 7(c) illustrate the path from the loading point to unloading area A, passing through the central intersection and intersection A. For loads above 800 g, the motors had

difficulty performing the turns and realigning afterward, which shows the need for improvements to handle heavier loads.

To increase operational safety, the anti-collision system was validated. The ultrasonic sensor, programmed to stop the vehicle when detecting an object at 10 cm, operated effectively on straight trajectories. In Figures 7(d) and 7(e), the objects used in the tests can be observed, along with the AGV's red lights, which indicate an alert condition. However, the sensor showed limitations when detecting objects outside the frontal angle, due to the short programmed distance and the simple sensor model used.

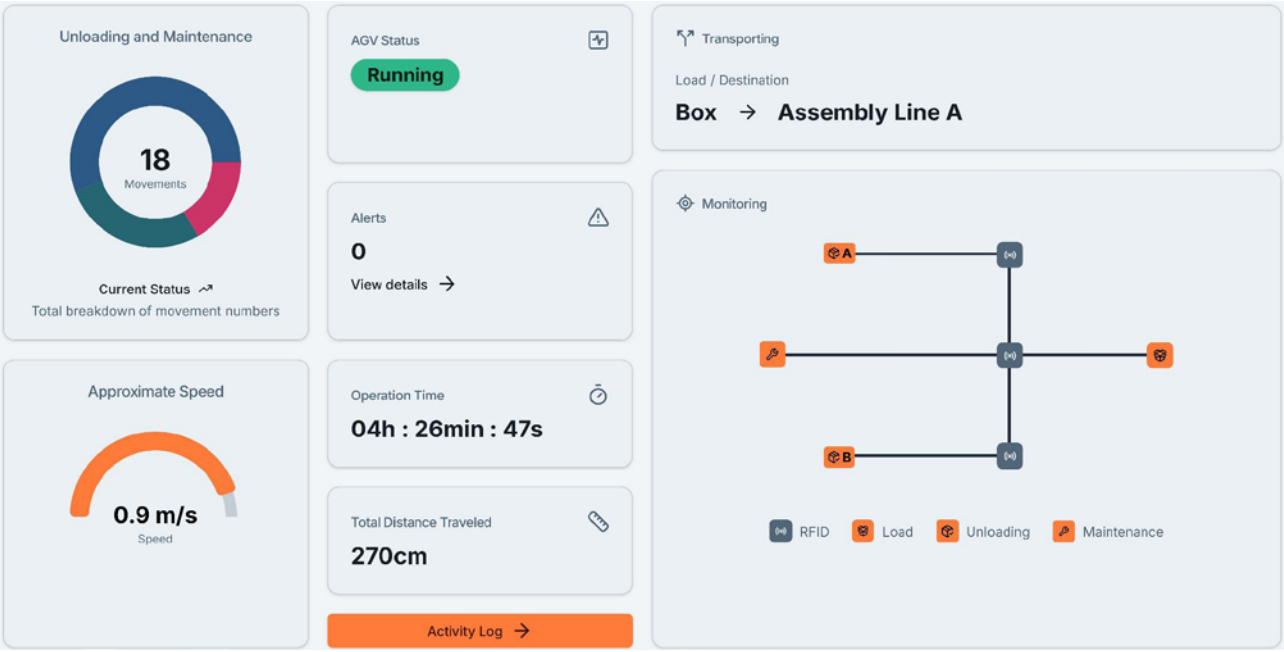
Figure 7 - Prototype and operational tests: (a) final version of the AGV; (b) and (c) AGV performing the route; (d) and (e) safety tests with the ultrasonic sensor.



Source: Prepared by the authors (2026).

AGV monitoring was carried out through a web application developed in Next.js and integrated with the Raspberry Pi via the MQTT protocol, ensuring runtime communication. Figure 8 illustrates the main components of the application's Overview screen, such as the number of completed transports, speed, status, time, alerts, destination, and current position along the route.

Figure 8 - Components of the web application Overview screen.



Source: Prepared by the authors (2026).

In addition, Figure 9 presents the main components of the Transport Management screen, where detailed information about the ongoing transport is displayed. This interface also provides functionalities such as switching to maintenance mode, registering a new load, and starting and canceling transports. To start a transport, the operator selects a previously registered load, including its type, dimensions, and destination, and the confirmed request is sent to the AGV, which begins the transport. In maintenance mode, the operator can assume full control of the AGV using a radio controller, allowing it to be moved in case of issues or directed to the maintenance area.

Figure 9 - Components of the web application Transport Management screen.



Source: Prepared by the authors (2026).

Finally, the application includes an Activity Log feature, which records the actions performed by the operator and by the AGV, facilitating problem identification and resolution while maintaining an operational history of the system.

Table 3 summarizes how the requirements were verified in the tests. The verification was carried out as a proof of concept, with a single prototype and without repetition of runs, and therefore metrics such as success rate, RFID reading failures, braking distance, and communication latency were not measured.

Table 3 - Verification of the requirements.

RF02	Not met. Line following with the reflectance sensor was abandoned after the functional limitations reported in this section, and navigation was performed with the RFID reader only.
RF03	Met. The control points of the route were identified from the tags fixed to the floor, and the decisions about the path were taken from them.
RF04	Met for a unit load of 92 mm in length, 68 mm in width, and 42 mm in height, with an approximate weight of 10 g. Above 800 g the motors lose alignment after the turns.
RF05	Met at 10 cm on straight trajectories. Objects outside the frontal angle of the ultrasonic sensor were not detected.
RF06	Met. The destination published on the MQTT topic was received and processed by the Raspberry Pi during operation.
RNF02	Partially verified. The color yellow required by NR-26 was adopted in the mechanical design, and the application displays the approximate speed during operation, but the limit of 1 m/s established by the VDI 2510 standard was not systematically measured.
RNF04	Met. The Activity Log records the actions performed by the operator and by the AGV.

Source: Prepared by the authors (2026).

5 CONCLUSION

This work presented the development of a low-cost Automated Guided Vehicle (AGV) prototype integrated with a web application for Industry 4.0 logistics. The system demonstrated the feasibility of using RFID for localization and MQTT for runtime control in a simulated environment, and the integration of Next.js with tRPC provided a single interface for monitoring and control.

However, limitations were observed regarding the navigation method. The removal of the reflectance sensor and the reliance on open-loop control of the turns, based on time, reduced precision with heavier loads, above 800 g, and restricted the vehicle to short and simple trajectories. The ultrasonic safety system proved effective for frontal obstacles but lacked peripheral coverage. The evaluation was also limited, since it was conducted as a proof of concept, with a single prototype and without repetition of runs or consolidated metrics, such as success rate, operating speed, and braking distance, which restricts the generalization of the results.

Future work will focus on implementing closed-loop control using wheel encoders or computer vision to improve navigation accuracy, and on enhancing the obstacle detection

system with LiDAR or multiple ultrasonic sensors to cover the entire perimeter of the vehicle. A systematic evaluation is also planned, with repetition of runs and measurement of success rate, operating speed, reading failures, braking distance, and communication latency, using the records already produced by the Activity Log.

REFERENCES

ANANTHI, K. *et al.* IoT based AGV system for medicare. In: **2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC)**. Coimbatore: IEEE, 2022. p. 410-414. DOI: 10.1109/ICESC54411.2022.9885709.

ANDERSON, D. J. **Kanban: successful evolutionary change for your technology business**. Sequim: Blue Hole Press, 2010.

GROOVER, M. P. **Automation, production systems, and computer-integrated manufacturing**. Upper Saddle River: Pearson, 2014. (Pearson Custom Library).

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. **ISO 3691-4: industrial trucks - safety requirements and verification - part 4: driverless industrial trucks and their systems**. 2. ed. Geneva: ISO, 2023.

KAGERMANN, H.; WAHLSTER, W.; HELBIG, J. **Recommendations for implementing the strategic initiative INDUSTRIE 4.0: final report of the Industrie 4.0 Working Group**. Frankfurt: Acatech, 2013.

LIU, C.; FAN, M.; SONG, R. A novel intelligent and agile warehouse system for energy meter storage. In: **2013 IEEE Third International Conference on Information Science and Technology (ICIST)**. Yangzhou: IEEE, 2013. p. 129-133.

MEHAMI, J.; NAWI, M.; ZHONG, R. Y. Smart automated guided vehicles for manufacturing in the context of Industry 4.0. **Procedia Manufacturing**, v. 26, p. 1077-1086, 2018. ISSN 2351-9789. DOI: 10.1016/j.promfg.2018.07.144.

MINISTÉRIO DO TRABALHO E EMPREGO. **Norma Regulamentadora nº 26 (NR-26)**. Brasília, 2012.

OASIS. **MQTT version 5.0: OASIS standard**. 2019. Available at: https://docs.oasis-open.org/mqtt/mqtt/v5.0/mqtt-v5.0.html. Accessed on: Set. 11, 2026.

PIRES, J. N. **Automação e controlo industrial: Indústria 4.0**. Lisboa: Lidel, 2019.

PRESSMAN, R. S. *et al.* **Engenharia de software**. Porto Alegre: AMGH, 2021.

SASAMOTO, H. *et al.* Modeling and prototype implementation of an automated guided vehicle for smart factories. In: **2021 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT)**. Soyapango: IEEE, 2021. p. 1-6. DOI: 10.1109/ICMLANT53170.2021.9690543.

SOMMERVILLE, I. **Engenharia de software**. São Paulo: Pearson Prentice Hall, 2011.

TRPC. **tRPC: introduction**. Available at: https://trpc.io/docs. Accessed on: Set. 11, 2026.

USTUNDAG, A.; CEVIKCAN, E. **Industry 4.0: managing the digital transformation**. Cham: Springer, 2018. DOI: 10.1007/978-3-319-57870-5.

VEREIN DEUTSCHER INGENIEURE. **Specification for the operation of automated guided vehicles**. Berlin, 2021.

DESENVOLVIMENTO DE UM SERIOUS GAME PARA EDUCAÇÃO EM SAÚDE SOBRE O USO DOS SERVIÇOS DE URGÊNCIA E EMERGÊNCIA

DEVELOPMENT OF A SERIOUS GAME FOR HEALTH EDUCATION ON THE USE OF URGENT AND EMERGENCY CARE SERVICES

Nícholas Doviggi Meyer[1], Marianne Machado[2], Mateus Nedel dos Reis[3] e Gabriel Rodrigues Felipetto[4]

RESUMO

Os *serious games* podem articular entretenimento e aprendizagem para abordar questões relacionadas à educação em saúde. Este trabalho teve como objetivo desenvolver o *serious game P.A. - Plantão Absurdo*, voltado à associação entre sinais e sintomas, gravidade do quadro e serviço de saúde mais adequado. Trata-se de um estudo aplicado, desenvolvido durante uma *Game Jam* de quatro dias, envolvendo etapas de pré-produção, produção, testes e análise do processo. Foram elaboradas situações clínicas e possibilidades de encaminhamento, além da implementação dos elementos visuais, interativos e audiovisuais do jogo. Como resultado, foi desenvolvida uma versão funcional composta por 40 casos clínicos e três possibilidades de encaminhamento, além de materiais audiovisuais complementares. O produto apresenta potencial como recurso de educação em saúde para abordar o uso adequado dos serviços, embora sua efetividade na aprendizagem ainda necessite ser investigada.

Palavras-chave: Aprendizagem, Aprendizagem por Associação, Tecnologia Educacional, Saúde Pública, Atenção Primária à Saúde.

ABSTRACT

Serious games can combine entertainment and learning to address issues related to health education. This study aimed to develop the serious game P.A. - Plantão Absurdo, designed to promote

1 Egresso do Curso de Jogos Digitais, Mestre e Doutorando em Saúde Materno Infantil. Universidade Franciscana (UFN). E-mail: nicholas.doviggi@gmail.com. ORCID: http://orcid.org/0000-0001-8329-0629
2 Acadêmica de Letras Português-Inglês. Bacharel em Artes Visuais. Universidade Federal de Minas Gerais (UFMG). E-mail: marianne.machado@hotmail.com. ORCID: https://orcid.org/0000-0002-8211-0574
3 Acadêmico de Jogos Digitais. Universidade Franciscana (UFN). E-mail: m.reis.nedel@gmail.com. ORCID: https://orcid.org/0009-0007-2833-0369
4 Egresso do Curso de Jogos Digitais. Universidade Franciscana (UFN). E-mail: gfelipetto@gmail.com. ORCID: http://orcid.org/0000-0001-5044-8215

the association between signs and symptoms, clinical severity, and the most appropriate healthcare service. This is an applied study developed during a four-day Game Jam, involving the stages of pre-production, production, testing, and process analysis. Clinical scenarios and referral possibilities were designed, along with the implementation of the game's visual, interactive, and audiovisual elements. As a result, a functional version of the game was developed, comprising 40 clinical cases and three referral options, in addition to complementary audiovisual materials. The product presents potential as a health education resource to address the appropriate use of healthcare services, although its effectiveness as a learning tool still requires investigation through studies involving players.

Keywords: *Learning, Association Learning, Educational Technology, Public Health, Primary Health Care.*

1 INTRODUÇÃO

Os *serious games* (jogos sérios) são tecnologias digitais desenvolvidas com propósitos que vão além do entretenimento, combinando mecânicas lúdicas a objetivos educativos, instrutivos ou de treinamento (Michael; Chen, 2005). No campo da saúde, essas ferramentas têm se consolidado como estratégias para a mediação do conhecimento, permitindo que os usuários experimentem cenários complexos, tomem decisões e compreendam rotinas institucionais em um ambiente virtual seguro e interativo (Graafland *et al.,* 2012).

Em contrapartida, o Sistema Único de Saúde (SUS) enfrenta desafios históricos relacionados ao letramento em saúde da população e à navegação pelos seus serviços. A falta de informação clara sobre as atribuições e o fluxo de atendimento na Rede de Atenção à Saúde faz com que muitos usuários busquem atendimento em níveis de complexidade inadequados para a sua demanda, como a procura excessiva por serviços de urgência e emergência para situações passíveis de resolução na Atenção Primária à Saúde (Lima *et al.*, 2015). Essa inadequação no direcionamento pode contribuir para a sobrecarga dos serviços de maior complexidade e compromete a eficiência e a continuidade do cuidado.

Embora a literatura reconheça o papel dos *serious games* na promoção do letramento em saúde e na mudança de comportamentos (Da Silva *et al.*, 2024), sua aplicação para a mediação da navegação do usuário na rede assistencial exige metodologias de desenvolvimento que articulem o rigor técnico-sanitário às mecânicas de engajamento lúdico (Hungaro *et al.*, 2021). Nesse contexto, a documentação e a análise sistematizada do processo de criação de jogos contribuem para a literatura ao apresentar modelos de desenvolvimento transdisciplinar de mídias educativas, evidenciando como o game design pode ser operacionalizado para abordar dinâmicas complexas, e no caso deste trabalho, da gestão do cuidado no SUS.

Diante do cruzamento entre a necessidade de qualificar o direcionamento do usuário no SUS e o potencial educativo das mídias interativas, os *serious games* emergem como

uma alternativa promissora para desmistificar a organização da rede de saúde. Ao simular situações de atendimento e tomada de decisão sobre o encaminhamento dos pacientes, o jogo pode atuar como um recurso pedagógico dinâmico para promover o uso consciente dos serviços públicos (Sátiro *et al.*, 2025). Dessa forma, este trabalho objetiva descrever o processo de produção e desenvolvimento de um *serious game* voltado à orientação de usuários sobre a organização e o fluxo de encaminhamento dos serviços no âmbito do SUS.

2 METODOLOGIA

Trata-se de um estudo aplicado, voltado ao desenvolvimento de um *serious game* sobre pronto atendimento e pronto-socorro, intitulado *P.A. - Plantão Absurdo*. Além do jogo digital, foram desenvolvidos materiais complementares para sua divulgação e apresentação, incluindo *gameplay*, *trailer* e apresentação em evento.

O desenvolvimento ocorreu durante uma *Game Jam*, realizada em outubro de 2025, com duração de quatro dias. O evento ocorreu em uma universidade comunitária localizada na região central do estado do Rio Grande do Sul, Brasil. Durante o período da atividade, os integrantes da equipe reuniram-se presencialmente em uma sala da instituição, no período noturno. Um dos integrantes encontrava-se em outra cidade e participou das atividades de forma remota, por meio de chamadas de voz e vídeo.

O tema proposto pela organização da *Game Jam* foi “Não devia estar aqui”, utilizado como ponto de partida para a concepção do jogo. A partir do tema, a equipe definiu como proposta a criação de uma simulação de pronto atendimento na qual o jogador deveria analisar diferentes pacientes e decidir o encaminhamento mais adequado para cada situação.

A equipe foi composta por quatro integrantes, com funções distribuídas de acordo com suas competências. Um integrante, com experiência em desenvolvimento de jogos digitais, foi responsável pela programação e implementação do jogo na *game engine Unity*. Uma integrante, com experiência em jogos digitais, ilustração e quadrinhos, ficou responsável pelas ilustrações 2D, interface e personagens. Um integrante com experiência em modelagem e desenvolvimento de jogos digitais ficou responsável pela produção dos cenários e demais elementos tridimensionais. O quarto integrante, Nutricionista e com experiência em *game design*, foi responsável pelo *game design*, gerenciamento do projeto e *sound design*.

Para o desenvolvimento do jogo, foi utilizada a *game engine Unity*. As ilustrações bidimensionais (2D) foram produzidas no software *Procreate*, enquanto os elementos tridimensionais (3D) foram desenvolvidos no *Blender*. Os arquivos de áudio utilizados no projeto foram obtidos a partir de fontes com disponibilização para uso sem restrições de direitos autorais e, adicionalmente, alguns efeitos sonoros foram produzidos pela própria equipe

utilizando o software *Audacity*. Para organizar o desenvolvimento, foi elaborado um mini *Game Design Document* (GDD), contendo informações relacionadas às mecânicas, fluxos de jogo e demais elementos necessários à implementação do projeto.

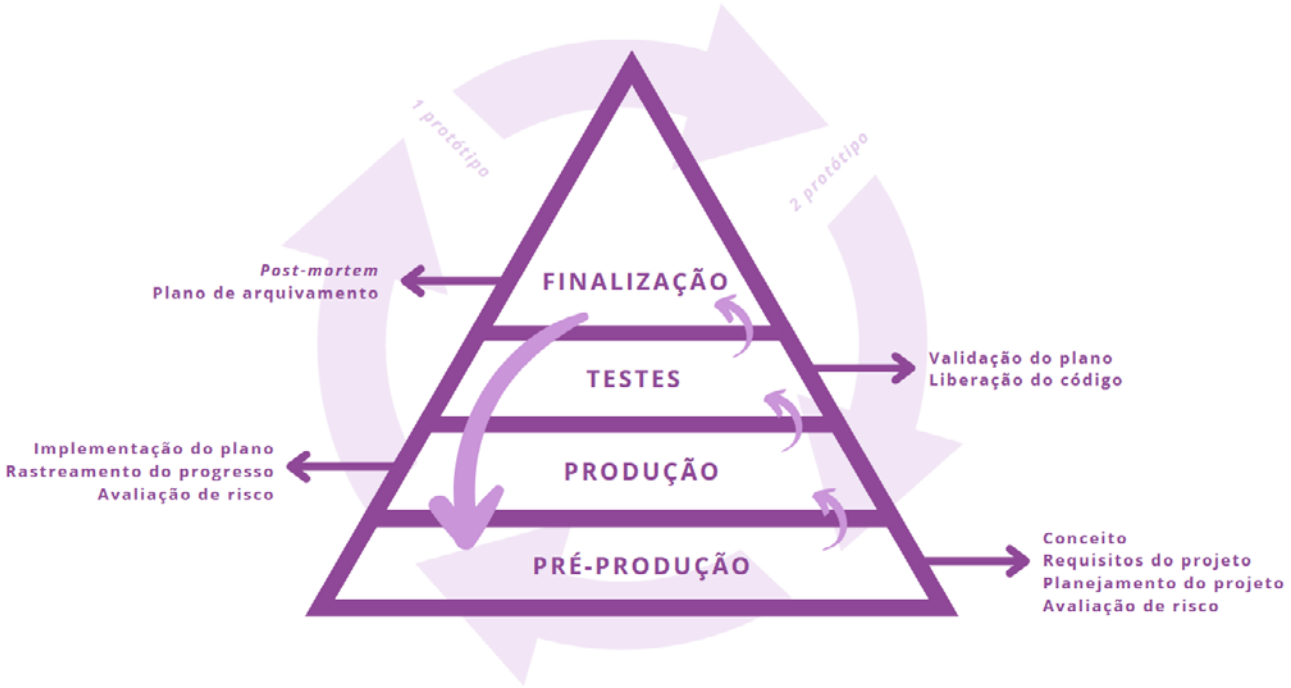
Os arquivos referentes aos *assets*, documentos e demais materiais do projeto foram organizados em uma pasta compartilhada no *Google Drive*. Para a comunicação e acompanhamento das atividades, foi utilizado um grupo no *Discord*, que também possibilitou a integração entre os integrantes que participaram presencialmente e o integrante que estava em outra cidade. Para a produção do material audiovisual, o *gameplay* foi gravado utilizando o *OBS Studio* e posteriormente editado no *DaVinci Resolve*.

O processo de desenvolvimento foi orientado pela metodologia proposta por Chandler (2009), apresentada na Figura 1, a qual já havia sido utilizada pela equipe em outros projetos de desenvolvimento de jogos, incluindo *Primeiras Refeições* (Meyer *et al.*, 2026b), *BebêXP* (Meyer *et al.*, 2026a) e *O Jardim do Ferro* (Beber *et al.*, 2024). A metodologia contempla quatro etapas principais: pré-produção, produção, testes e *post-mortem*.

Na etapa de pré-produção, foram definidas e organizadas as principais características do projeto, incluindo sua proposta, público, mecânicas, fluxo de jogo, elementos visuais e demais aspectos necessários para orientar o desenvolvimento. A etapa de produção correspondeu à implementação do projeto e à criação dos seus diferentes elementos, como programação, artes, cenários, interface, áudio e demais componentes do jogo, etapa na qual se utilizaram essencialmente todos os softwares de desenvolvimento mencionados anteriormente. Os testes corresponderam à verificação do funcionamento do jogo, permitindo identificar problemas técnicos, inconsistências e aspectos que necessitem de ajustes. Por fim, o *post-mortem* consistiu na análise do processo de desenvolvimento após sua conclusão, possibilitando identificar aspectos positivos, dificuldades encontradas e pontos que poderiam ser aprimorados em projetos futuros.

Considerando a duração reduzida da *Game Jam*, as etapas propostas por Chandler (2009) não ocorreram de maneira linear. Houve sobreposição e realização concomitante de atividades de pré-produção, produção e testes, de modo a permitir que decisões de planejamento fossem revistas conforme os elementos do jogo eram implementados. Essa organização possibilitou adaptar o processo às restrições de tempo da atividade, priorizando as funcionalidades essenciais para a construção de um protótipo jogável ao final dos quatro dias. O *post-mortem*, por sua vez, foi realizado posteriormente ao encerramento da *Game Jam*, permitindo à equipe analisar o processo de desenvolvimento e os resultados obtidos.

Figura 1 - Metodologia Cíclica de Chandler.



Fonte: Elaborado pelo autor através de uma adaptação de Chandler (2009).

3 RESULTADOS E DISCUSSÕES

O desenvolvimento do *P.A. - Plantão Absurdo* teve início durante uma sessão de *brainstorming* realizada pela equipe no primeiro momento da *Game Jam*. A partir do tema "Não devia estar aqui", foram levantadas diferentes possibilidades de jogos, incluindo propostas de comédia, terror, entretenimento e educação em saúde. A partir dessa etapa diagnóstica, optou-se pelo desenvolvimento de um *serious game* voltado à utilização inadequada dos serviços de pronto atendimento e pronto-socorro. Essa problemática caracteriza-se pela busca por atendimento de urgência e emergência para demandas de baixa gravidade que poderiam ser resolvidas na Atenção Primária à Saúde, fator apontado como uma das principais causas de sobrecarga e superlotação desses serviços (Lima *et al.*, 2015). Essa dinâmica compromete a eficiência assistencial, justificando a necessidade de estratégias educativas lúdicas para orientar o fluxo correto dos usuários na rede de atenção à saúde. A escolha buscou associar o caráter lúdico proposto pela *Game Jam* a uma situação concreta de saúde pública, utilizando o jogo como ferramenta para abordar o reconhecimento de sinais e sintomas e a escolha do serviço de saúde mais adequado.

A escolha da temática também esteve relacionada à organização dos serviços de saúde e às dificuldades da população em compreender os diferentes níveis de atenção. O SUS apresenta uma organização complexa, com diferentes serviços e atribuições, o que

pode contribuir para dúvidas da população sobre qual local buscar atendimento diante de determinada necessidade (Rodrigues *et al.*, 2024; Oliveira *et al.*, 2017). A utilização de serviços de urgência para situações que não demandariam esse nível de atendimento pode contribuir para a sobrecarga desses serviços, dificultando o atendimento de situações que efetivamente necessitam de assistência imediata (Oliveira *et al.*, 2017). Nesse sentido, identificou-se uma oportunidade para desenvolver uma ferramenta educativa capaz de apresentar essas situações de maneira interativa e acessível.

A partir dessa problemática, foi definida a principal mecânica do *P.A. - Plantão Absurdo*: o jogador assume o papel de um profissional responsável por avaliar pacientes que chegam ao serviço e decidir o encaminhamento mais adequado. Essa tomada de decisão ocorre diretamente na interface do jogo por meio de três carimbos interativos (Figura 2), que representam os destinos possíveis: “Pronto-Socorro (PS)”, destinado aos casos de maior gravidade; “Unidade de Pronto Atendimento (UPA)”, para situações de urgência que podem ser manejadas nesse serviço; e “Você não deveria estar aqui (VNDEA)”, categoria criada para representar situações que não justificariam a procura por um serviço de pronto atendimento. A terceira alternativa também funciona como um elemento humorístico, abordando de maneira lúdica situações em que o paciente procura o serviço por motivos inadequados, como a solicitação de atestados sem apresentar problemas de saúde. A pontuação do jogador é calculada com base na precisão dos encaminhamentos efetuados com os carimbos.

Figura 2 - Carimbos de encaminhamento.



Fonte: Elaborada pelos autores.

A mecânica foi construída de maneira a exigir que o jogador observe diferentes informações antes de tomar sua decisão. Inicialmente, o paciente apresenta uma queixa por meio de uma fala. O jogador pode acessar falas subsequentes, realizar perguntas

e observar informações relacionadas ao seu estado clínico. A partir desse conjunto de informações, deve identificar o possível problema e selecionar um dos três destinos disponíveis. Dessa forma, a identificação de sinais e sintomas deixa de ser apresentada apenas como conteúdo textual e passa a constituir parte da própria mecânica do jogo. Essa característica é particularmente relevante para *serious games*, uma vez que a integração entre conteúdo educativo e mecânica permite que o conhecimento seja mobilizado dentro da própria atividade de jogo, em vez de ser apresentado apenas como informação complementar (Gee, 2003; Mitgutsch; Alvarado, 2012; Meyer *et al.*, 2026b). Para garantir a acessibilidade do jogo a pessoas sem formação na área da saúde, foi incorporado um livro de consulta com informações sobre sintomas e encaminhamentos, auxiliando o jogador no processo de tomada de decisão.

Foram elaborados 40 casos clínicos seguindo um processo metodológico integrado de levantamento de literatura e expertise profissional. A definição dos temas e agravos fundamentou-se nos perfis epidemiológicos das demandas mais frequentes em Unidades Básicas de Saúde (UBS), UPAs e PS, conforme documentado na literatura (Camerro *et al.*, 2015; Aranguiz, 2019). Adicionalmente, utilizou-se a expertise técnica e clínica do nutricionista integrante da equipe de desenvolvimento do jogo, o qual realizou a seleção, adaptação e validação do conteúdo dos cenários, assegurando a fidelidade às rotinas assistenciais, o rigor condutual e a aplicabilidade pedagógica das situações simuladas. Para organizar esse processo, os casos foram estruturados em uma planilha contendo a fala inicial do paciente, duas falas subsequentes, fala ou pergunta adicional, informações obtidas durante o exame, problema real, características do paciente e destino de encaminhamento (Figura 3). A padronização dos casos permitiu organizar o conteúdo clínico e, simultaneamente, facilitar sua implementação durante o desenvolvimento do jogo. Como recurso auxiliar para a organização inicial dessas informações, foi utilizada inteligência artificial generativa *ChatGPT*, que contribuiu para estruturar os casos em formato tabular. Posteriormente, o conteúdo foi revisado e adaptado pela equipe, especialmente quanto às características das situações, aos encaminhamentos e à construção das falas. As falas dos pacientes foram posteriormente modificadas para aproximá-las da comunicação cotidiana. Foram incorporadas expressões coloquiais, hesitações, repetições, gírias e características linguísticas presentes na fala cotidiana, incluindo elementos associados à linguagem regional. Essa adaptação teve como objetivo tornar os personagens mais reconhecíveis e naturais, além de reforçar o caráter humorístico do jogo, buscando assim estabelecer um equilíbrio entre a apresentação de situações clínicas reais e uma linguagem capaz de manter o caráter lúdico da experiência (Marcuschi, 2001; Deterding *et al.*, 2011).

Figura 3 - Tabela de situações dentro do jogo.

Fala Inicial	Falas subsequêntes	Fala A/Pergunta	Examinar	Problema Real	Característica	Destino
Dr estou me sentindo com muita dor de cabeça, medi e estou com 38 de **febre**.	- Será que to com Covid? -To com dor no braço também... maldita vacina...	Tomei vacina para gripe ontem...	Paciente com 37° com aspecto febril.	Reação da Vacina.	Corado	UPA
Preciso de atestado pra ontem, bebi no carnaval e não fui trabalhar.	- Mas juro que foi só um pouquinho. - Tô melhor agora.	Hoje to bem de boa.	Sem sinal de intoxicação ou embriaguez	Falta de vergonha na cara	Rindo/Sorriso	VNDEA
Tô com **falta de ar** desde ontem, piorou hoje.	- Fumei por 30 anos. - Usei bombinha e não resolveu.	Tenho DPOC (Doença Pulmonar Obstrutiva Crônica)	SatO 85%, respiração ofegante, PA 100x60.	Crise de DPOC exacerbada.	Cianótico	PS
Eu não paro de **vomitar**!	- Dr vou vomitar no senhor. -Me ajuda!	Eu bebi só umas caipiras.	Paciente com forte cheiro de álcool.	Embriagado.	Corado	UPA
Tô com uma **dor** no dedo desde ontem.	- Cortei a unha muito fundo. - Mas é bom ver, vai que é infarto.	Só lateja quando encosto.	Vermelhidão leve no dedo, sem febre.	Unha encravada.	Random	VNDEA
EU TOMEI UM TIRO!	- ME AJUDA EU VOU MORRER. - POR FAVOR DR EU TE IMPLORO.	TU NÃO TA VENDO QUE EU TOMEI UM TIRO?	É, ele tomou um tiro e não para de sangrar.	Tomou um tiro	Sangrando	PS
Dr, eu não to legal... que **aperto no peito.**	- Eu to muito tonto... - Me segura, vou cair	Faz meia hora que estou com dificuldade de respirar.	Paciente suando frio, respiração rápida e PA 90x60.	Infarto agudo do miocárdio.	Cianótico	PS
Eu to com um **desarranjo**...	- Que dor de barriga... - Onde fica o banhe	Eu to que nem rei, o dia todo no trono.	Paciente desidratado, com maus odores e pálido	Intoxicação alimentar	Random	UPA
SOCORRO! Ta **sangrando** muito! Ta sangrando muito!	- Ta sangrando muito! - CADE O MEU BRAÇO?	Eu deixei a motossera cair em cima do meu antebraço...	Antebraço do paciente foi amputado.	Braço amputado e sangrando	Sangrando	PS
IAC que lindeza de lugar *IAC*	- Adoro vocês! - COMO EU SINTO FALTA DA MINHA EX!	Sempre estarei a sua disposiçaaaaumm	Paciente embriagado, sem sinais de vômito ou desidratação.	Paciente embriagado	Rindo/Sorriso	VNDEA
Doutor, uma **dor no peito**... apertando forte... parece que tem um elefante sentado aqui.	- O braço esquerdo tá todo dormente. - Tô suando frio, me sentindo enjoado.	Começou faz uns 20 minutos, do nada.	Paciente pálido, sudoreico, com a mão no peito. Pressão 90/60 mmHg.	Infarto agudo do miocárdio.	Cianótico	PS
Oi, tudo bom? Então... preciso de um atestado médico, por favor.	- É que amanhã tem a final do campeonato, não posso perder! - Minha vó ficou doente lá no interior, sabe como é...	Ah, dr... pra ser sincero, não tô sentindo nada. É só pra não ser demitido.	Paciente com aparência saudável, sinais vitais normais.	Falta de vergonha na cara	Random	VNDEA
Que linda flor!	- Rosas, mostardas, tulipas... - Mostarda é flor?	Minha mãe mandou eu vim ver vocês.	Paciente com forte cheiro alcoólico, sem sinais de vômito ou desidratação.	Paciente embriagado	Random	VNDEA
Oi dr, como você está?	- To meio tonto só. - Sentir um pouco de dor no peito é normal?	Me senti um pouco tonto e vim ver se está tudo bem.	Paciente pálido, sudoreico, com a mão no peito. Pressão 90/60 mmHg.	Infarto agudo do miocárdio.	Cianótico	PS

Fonte: Elaborada pelos autores.

A implementação dos casos ocorreu de maneira concomitante ao desenvolvimento da programação. Conforme os códigos, diálogos e mecânicas eram inseridos no jogo, a equipe realizava testes para verificar o funcionamento do fluxo de atendimento. Esses testes permitiram identificar problemas relacionados à sequência das falas, funcionamento das interações, comportamento dos elementos visuais e ocorrência de *bugs*. Quando eram identificados problemas ou situações que prejudicavam a compreensão ou o fluxo do jogo, eram realizados ajustes e novos testes. Assim, produção e testes ocorreram de forma iterativa ao longo dos quatro dias da *Game Jam*, permitindo que decisões fossem modificadas conforme o jogo era efetivamente experimentado pela equipe. Esse processo foi especialmente importante diante da limitação temporal da atividade, uma vez que possibilitou priorizar as funcionalidades essenciais e corrigir problemas durante o próprio desenvolvimento (Kultima, 2015; Olsen, 2017).

A produção dos personagens também buscou responder às limitações de tempo características de uma *Game Jam*. A identidade visual do jogo foi construída a partir da combinação de elementos bidimensionais e tridimensionais, utilizando uma estética inspirada em referências visuais da década de 1980 e do início dos anos 1990. A escolha desse estilo também esteve relacionada às competências dos integrantes da equipe, permitindo aproveitar diferentes experiências na produção de ilustrações e modelagem tridimensional (Figura 4).

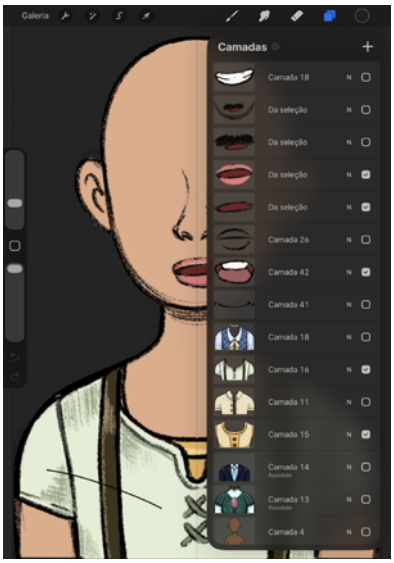
Figura 4 - Cenário 3D dentro do *Blender.*



Fonte: Elaborada pelos autores.

Para ampliar a quantidade de personagens sem aumentar proporcionalmente o tempo necessário para sua produção, foram desenvolvidos personagens a partir de sistemas de camadas. Elementos como corpo, cabelos, boca, nariz e roupas poderiam ser combinados de diferentes maneiras, permitindo a criação de múltiplas aparências a partir de um conjunto limitado de *assets* (Figura 5). Foram estabelecidas restrições para evitar a repetição excessiva das mesmas combinações e para impedir combinações visualmente inadequadas. Esse recurso permitiu aumentar a diversidade dos pacientes apresentados durante o jogo sem exigir a produção individual de cada personagem, constituindo uma estratégia de otimização compatível com o curto período disponível para desenvolvimento.

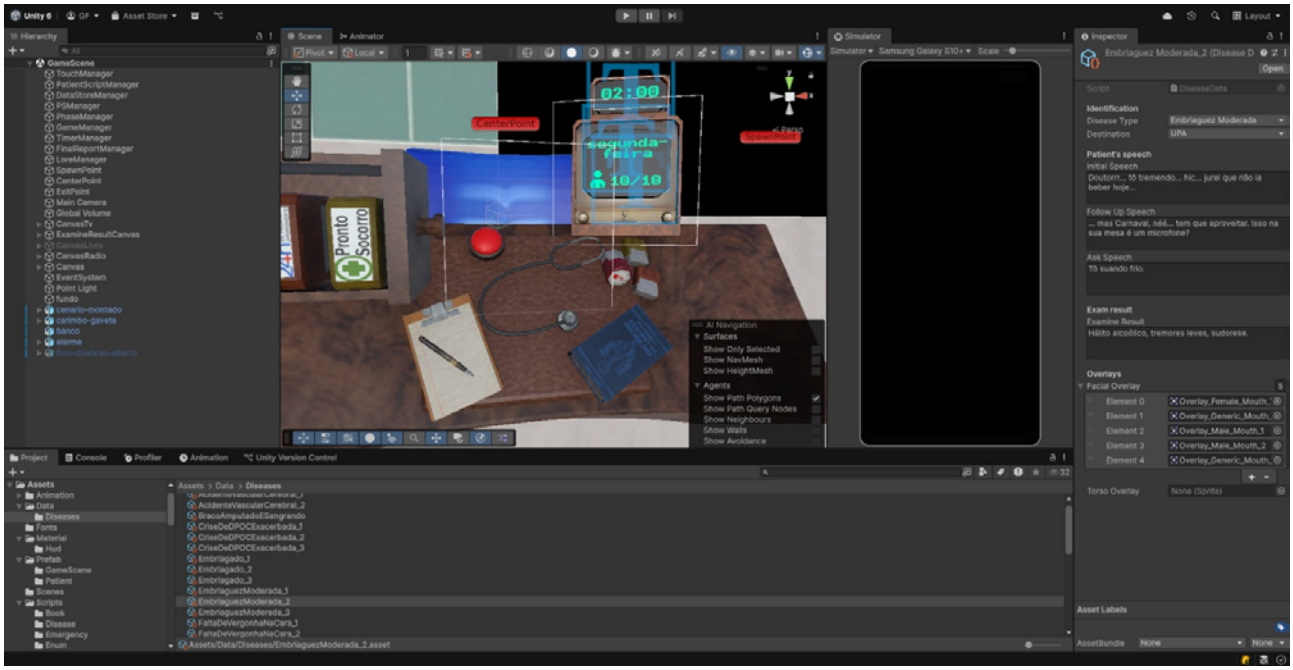
Figura 5 - *Assets* em camadas no *Procreate*.



Fonte: Elaborada pelos autores.

A programação foi realizada na *game engine Unity* (Figura 6), utilizando C#. A organização do código seguiu princípios de programação orientada a objetos, permitindo estruturar os diferentes componentes do jogo de maneira modular e facilitar sua reutilização e alteração durante o desenvolvimento (Nystrom, 2014). Essa característica foi particularmente relevante durante a *Game Jam*, considerando que o projeto passou por alterações sucessivas durante os testes e que diferentes elementos, como pacientes, diálogos, destinos e comportamentos, precisavam ser implementados e modificados rapidamente.

Figura 6 - Jogo dentro da *game engine Unity*.



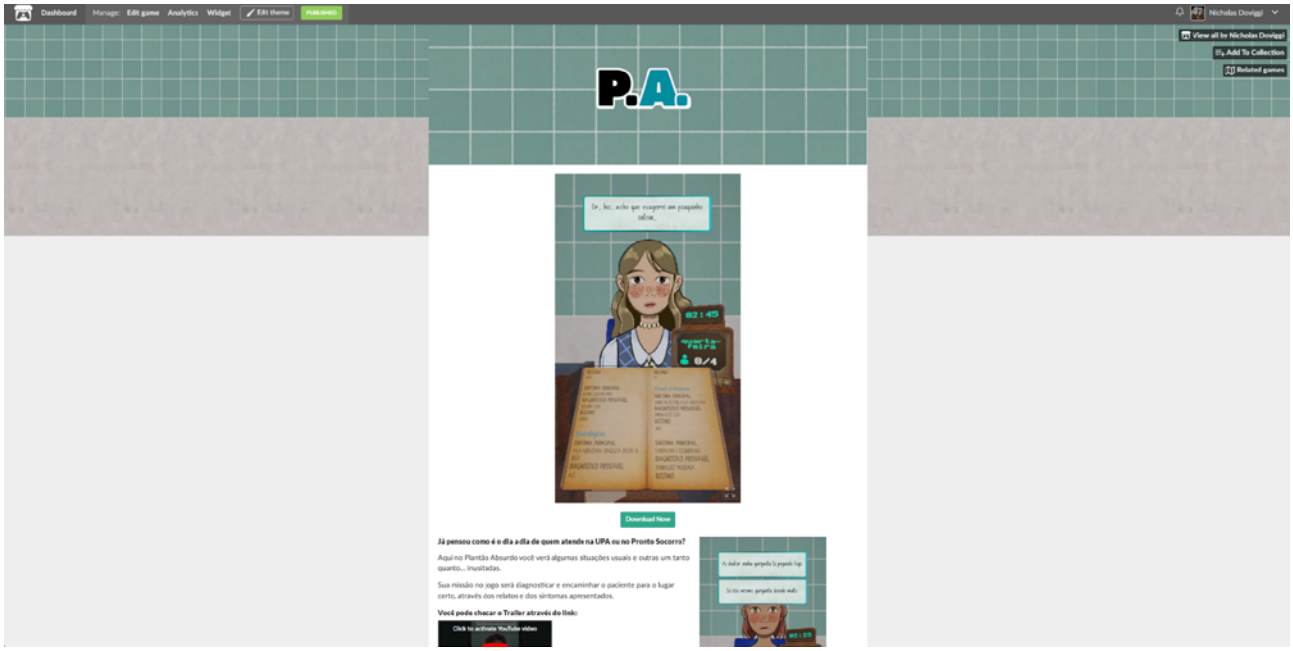
Fonte: Elaborada pelos autores.

A combinação entre desenvolvimento, testes e ajustes também contribuiu para adequar a proposta inicialmente planejada às limitações encontradas durante a produção. Como as etapas de pré-produção, produção e testes ocorreram concomitantemente, algumas decisões foram revistas à medida que novas funcionalidades eram implementadas. Essa dinâmica é característica de processos iterativos de desenvolvimento de jogos, nos quais a experimentação do produto permite identificar problemas e orientar alterações ainda durante sua construção (Fullerton, 2014; Chandler, 2009)

Como produto complementar ao jogo, foi desenvolvido um trailer com duração de aproximadamente 45 segundos. A produção audiovisual buscou apresentar inicialmente a problemática de saúde pública relacionada à utilização dos serviços de pronto atendimento, utilizando dados divulgados em notícia sobre os atendimentos realizados no município de Santa Maria do Rio Grande do Sul no Brasil (Aranguiz, 2019). O trailer intercala essa informação com cenas do *gameplay*, apresentando diferentes pacientes e situações

encontradas no jogo, além de frases de efeito relacionadas à temática. A utilização dos dados reais como ponto inicial do vídeo teve como objetivo estabelecer uma conexão entre o problema apresentado no jogo e uma situação existente no contexto local. A edição do trailer também buscou produzir contraste entre a gravidade da problemática e o caráter humorístico do jogo. As informações sobre os serviços de saúde são apresentadas juntamente a cenas rápidas do *gameplay* e uma trilha sonora de fundo, criando uma atmosfera dinâmica e de impacto. Dessa forma, o trailer não se limitou a apresentar as características do produto, mas também funcionou como uma peça de comunicação da problemática que motivou seu desenvolvimento. O produto final foi disponibilizado na plataforma itch.io[5] para as plataformas *Web* e *Android*, permitindo o acesso ao jogo após o encerramento da *Game Jam* (Figura 7). O trailer foi disponibilizado de forma não listada no YouTube[6].

Figura 7 - Jogo na plataforma itch.io.



Fonte: Elaborada pelos autores.

Por fim, o *P.A. - Plantão Absurdo* foi apresentado oralmente em uma mostra de produtos técnico-tecnológicos, realizada durante um evento internacional promovido por uma universidade comunitária. A apresentação contemplou a concepção do projeto, a composição da equipe, a ideia central do jogo e seu processo de desenvolvimento, além da exibição do trailer com trechos de *gameplay*. A participação possibilitou apresentar o produto, sua proposta educativa e os materiais desenvolvidos durante a *Game Jam* a um público ampliado.

5 https://nicholas-doviggi.itch.io/plantao-absurdo
6 https://youtu.be/ggDvnXbBmLA?si=j-_AKyboyUkUiMnO.

4 CONCLUSÕES

O *P.A. - Plantão Absurdo* constitui uma proposta de *serious game* como estratégia de educação em saúde voltada à compreensão dos serviços de urgência e emergência do SUS. Ao transformar situações clínicas e decisões de encaminhamento em elementos centrais da experiência de jogo, o produto aborda uma problemática relacionada ao uso inadequado dos serviços de saúde, aproximando o jogador de situações que fazem parte do cotidiano desses atendimentos. Dessa forma, a proposta possibilita explorar, de maneira interativa, a relação entre sinais e sintomas, gravidade do quadro e escolha do serviço de saúde mais adequado. Nesse contexto, a principal contribuição do projeto está na utilização de uma linguagem própria dos jogos, associada ao humor e à simulação, para abordar uma questão concreta da organização e utilização dos serviços de saúde, tornando a temática passível de ser explorada de forma contextualizada e interativa.

Embora o desenvolvimento em formato de *Game Jam* tenha imposto limitações de tempo e escopo, foi possível produzir uma versão funcional, acompanhada de materiais audiovisuais e apresentada em evento institucional. O processo também evidencia a possibilidade de desenvolver, em um período reduzido, um produto tecnológico que articule diferentes áreas de conhecimento e recursos de entretenimento a uma demanda de saúde pública. Portanto, o produto desenvolvido apresenta potencial para compor ações de educação em saúde e subsidiar discussões sobre o uso adequado dos serviços do SUS. Entretanto, sua efetividade enquanto instrumento de aprendizagem ainda deverá ser investigada por meio de estudos com jogadores, especialmente quanto à capacidade de associar sinais e sintomas, gravidade do quadro e serviço de atendimento mais adequado.

5 DECLARAÇÃO DE USO DE INTELIGÊNCIA ARTIFICIAL

Os autores declaram que as ferramentas de inteligência artificial generativa, especificamente Google Gemini e ChatGPT, foram empregadas exclusivamente para suporte na revisão gramatical e no refinamento da estruturação textual. Adicionalmente, a plataforma Consensus foi utilizada como instrumento auxiliar na prospecção de literatura científica para fundamentação teórica e citações. Ressalta-se que todo o conteúdo intelectual, a análise crítica e a redação final são de inteira responsabilidade dos autores, garantindo a originalidade e o rigor científico do presente trabalho.

REFERÊNCIAS

ARANGUIZ, Dandara Flores. Falha na rede básica de saúde e antigos hábitos lotam emergências em Santa Maria. **Diário de Santa Maria**, Santa Maria, 8 abr. 2019. Disponível em:https://diariosm.com.br/noticias/saude/falha_na_rede_basica_de_saude_e_antigos_habitos_lotam_emergencias_em_santa_maria.442234 . Acessado em: 9 set. 2026.

BEBER, Rejane Bermudes Costa *et al.* Digital educational game “O Jardim do Ferro”: a tool for the prevention of iron deficiency anemia in childhood. **Games for Health Journal**, v. 13, n. 1, p. 50-56, 2024. DOI: https://doi.org/10.1089/g4h.2023.0006.

CAMERRO, Aline *et al.* Perfil do atendimento de serviços de urgência e emergência. **Revista Fafibe On-Line**, Bebedouro São Paulo, p. 515-524, 2015. Disponível em: https://pt.scribd.com/document/401979813/Perfil-do-atendimento-de-servicos-de-urgencia-e-emergencia. Acessado em: 9 set. 2026.

CHANDLER, Heather Maxwell. **Manual de produção de jogos digitais.** Bookman Editora. 2009.

DA SILVA, Erielton Gomes *et al.* Jogos educativos para melhorar letramento em saúde de pessoas com diabetes: protocolo de scoping review. **Online Brazilian Journal of Nursing**, v. 23, n. Suppl 1, 2024. Disponível em: https://objnursing.uff.br/nursing/article/view/6732. Acessado em: 9 set. 2026

DETERDING, Sebastian *et al.* From game design elements to gamefulness: defining” gamification”. In: **Proceedings of the 15th international academic MindTrek conference: Envisioning future media environments.** 2011. p. 9-15. DOI: https://doi.org/10.1145/2181037.2181040

FULLERTON, Tracy. **Game design workshop: a playcentric approach to creating innovative games**. CRC press, 2014.

GEE, James Paul. What video games have to teach us about learning and literacy. **Computers**

in entertainment (CIE), v. 1, n. 1, p. 20-20, 2003. DOI: https://doi.org/10.1145/950566.95059

GRAAFLAND, Maurits; SCHRAAGEN, Jan M.; SCHIJVEN, Marlies P. Systematic review of serious games for medical education and surgical skills training. **Journal of British Surgery**, v. 99, n. 10, p. 1322-1330, 2012. DOI: https://doi.org/10.1002/bjs.8819

HUNGARO, Thiago Alves *et al.* Jogos sérios e gamificação: um novo modelo para educação em saúde. **Revista Eletrônica Acervo Saúde**, v. 13, n. 9, p. e8540, 2021. DOI: https://doi.org/10.25248/REAS.e8540.2021

KULTIMA, Annakaisa. Defining game jam. In: **FDG**. 2015. Disponível em: http://www.fdg2015.org/papers/fdg2015_paper_21.pdf. Acessado em: 21 ago. 2026

LIMA, Sayonara Arruda Vieira *et al.* Elementos que influenciam o acesso à atenção primária na perspectiva dos profissionais e dos usuários de uma rede de serviços de saúde do Recife. **Physis: Revista de Saúde Coletiva**, v. 25, n. 2, p. 635-656, 2015. DOI: https://doi.org/10.1590/S0103-73312015000200016

MARCUSCHI, Luiz Antônio. **Da fala para a escrita: atividades de retextualização**. Cortez Editora, 2001.

MEYER, Nícholas Doviggi *et al.* Development and validation of the serious game “BebêXP”: a food and nutritional education tool for children. **Millenium**, n. 22, p. 24, 2026a. DOI: https://doi.org/10.29352/mill0222e.41339

MEYER, Nícholas Doviggi *et al.* Primeiras refeições: Serious game to improve infant feeding knowledge. **International Journal of Serious Games**, p. 5-18, 2026b. DOI: https://doi.org/10.17083/j9g34329

MICHAEL, David R.; CHEN, Sandra L. **Serious games: Games that educate, train, and inform**. Muska & Lipman/Premier-Trade, 2005.

MITGUTSCH, Konstantin; ALVARADO, Narda. Purposeful by design? A serious game design assessment framework. In: **Proceedings of the International Conference on the foundations of digital games**. 2012. p. 121-128. DOI; https://doi.org/10.1145/2282338.2282364

NYSTROM, Robert. **Game programming patterns.** Carrollton: Genever Benning, 2014.

OLIVEIRA, Gabriela Silva *et al.* Superlotação das urgências e estratégias de gestão de crise: uma revisão de literatura. **Caderno de Graduação-Ciências Biológicas e da Saúde-UNIT-SERGIPE**, v. 4, n. 2, p. 114-114, 2017. Disponível em: https://periodicos.set.edu.br/cadernobiologicas/article/view/4596. Acessado em: 21 ago. 2026

OLESEN, Jeanette Falk. Design processes in game jams: studies of rapid design processes. In: **Extended Abstracts Publication of the Annual Symposium on Computer-Human Interaction in Play**. 2017. p. 723-726. DOI: https://doi.org/10.1145/3130859.3133226

RODRIGUES, Marcela Moura *et al.* Uso de serviço de emergência por motivos não urgentes: estudo qualitativo com usuários de um pronto atendimento, Vitória, ES, Brasil, 2019. **Interface-Comunicação, Saúde, Educação**, v. 28, p. e230493, 2024. DOI: https://doi.org/10.1590/interface.230493

SÁTIRO, Luana Silva Pereira *et al.* Serious Games Developed for Patient Health Education: A Scoping Review. **Enferm Foco**, v. 16, p. -, 2025. Disponível em:https://enfermfoco.org/en/article/serious-games-developed-for-patient-health-education-a-scoping-review/. Acessado em: 21 ago. 2026

SISTEMA DE GERENCIAMENTO DE CONTEÚDO PARA O PORTAL DA REDE NACIONAL DE MULHERES NA NANOCIÊNCIA

CONTENT MANAGEMENT SYSTEM FOR THE PORTAL OF THE NATIONAL NETWORK OF WOMEN IN NANOSCIENCE

Vanessa Cezar do Nascimento[1], Luiza Lewandowski Karlec[2], Alexandre Zamberlan[3], Robertson E. dos Santos[4], Laura Osmari Vendrame[5] e Solange Binotto Fagan[6]

RESUMO

Este artigo aborda o desenvolvimento de um Sistema de Gerenciamento de Conteúdo (CMS) para o portal da Rede Nacional de Mulheres na Nanociência. A necessidade de um ambiente que facilitasse a divulgação e o registro das atividades realizadas pelo projeto motivou a elaboração de um portal para a rede. Para facilitar a adição e o gerenciamento dos conteúdos que serão publicados, foi desenvolvido um CMS que proporciona um painel administrativo para a manutenção do portal por meio de interfaces gráficas, sem a necessidade de alteração de código-fonte. Para a construção desse sistema, empregou-se a metodologia Scrum e foram utilizados a linguagem de programação Python e os *frameworks* Django e Bootstrap. Para o controle de versionamento, utilizaram-se as ferramentas Git e GitHub e, para o gerenciamento da aplicação, o ecossistema Docker. Como resultado há o sistema web[7] disponível no portal da Universidade Franciscana.

Palavras-chave: Computação em Nuvem; Django; Mulheres na Ciência; Python.

1 Acadêmica do curso de Ciência da Computação na Universidade Franciscana. E-mail: vanessa.cezar@ufn.edu.br ORCID: https://orcid.org/0009-0009-4641-2821
2 Acadêmica do curso de Ciência da Computação na Universidade Franciscana. E-mail: luiza.karlec@ufn.edu.br ORCID: https://orcid.org/0009-0007-5430-6135
3 Professor nos cursos de Ciência da Computação, Jogos Digitais e Sistemas de Informação na Universidade Franciscana. E-mail: alexz@ufn.edu.br. ORCID: https://orcid.org/0000-0002-9815-2031
4 Parceiro da empresa ER Clinic junto ao Laboratório de Práticas da Computação UFN. E-mail: robertson@ersistemas.info. ORCID: https://orcid.org/0000-0003-0683-6063
5 Pesquisadora. Programa de Pós-graduação em Nanociência da Universidade Franciscana. E-mail: laura.o.vendrame@gmail.com ORCID: https://orcid.org/0000-0001-9662-7235
6 Professora, pesquisadora e coordenadora do projeto. Programa de Pós-graduação em Nanociência da Universidade Franciscana. E-mail: sfagan@prof.ufn.edu.br. ORCID: https://orcid.org/0000-0001-7878-1212
7 Link para o sistema: https://redemulheresnano.ufn.edu.br/.

ABSTRACT

This paper presents the development of a Content Management System (CMS) for the portal of the National Network of Women in Nanoscience. The need for a platform that facilitated the dissemination and registration of the activities carried out by the project motivated the development of a portal for the network. For the management of the content to be published, a CMS was developed providing an administrative panel for portal maintenance through graphical interfaces, eliminating the need to modify source code, such as HTML. For the development of this system, the Scrum methodology was employed, utilizing the Python programming language and the Django and Bootstrap frameworks. Regarding version control, the Git and GitHub tools were utilized, and for application management, the Docker ecosystem. As a result, the web system is available on the Franciscan University portal.

Keywords: *Cloud Computing; Django; Python; Women in Science;.*

1 INTRODUÇÃO

A persistência da desigualdade de gênero nas áreas de ciência e tecnologia é evidente na sociedade atual, uma vez que a presença de mulheres ainda é inferior à dos homens na vida acadêmica, no mercado de trabalho e em cargos de liderança. Ao analisar o número de pesquisadores registrados no Conselho Nacional de Desenvolvimento Científico e Tecnológico em 2018, apenas cerca de 30% das mulheres atuavam nas áreas de Química, Física e Matemática (Naideka *et al.*, 2020). A inserção de mulheres na área de exatas, em qualquer estágio de ensino, ainda é incipiente; incentivar essa participação é um importante agente global na busca por igualdade de gênero e na garantia de um desenvolvimento científico e tecnológico mais inclusivo e representativo da sociedade (Souto; Souto, 2022).

De acordo com a terceira edição do informativo Estatísticas de Gênero: Indicadores sociais das mulheres no Brasil, publicada pelo Instituto Brasileiro de Geografia e Estatística (IBGE), do total de mulheres que concluíram cursos de graduação no ano de 2022, somente 22% optaram por alguma graduação na área de Ciência, Tecnologia, Engenharia e Matemática (STEM). O número de concluintes mulheres na área da computação é ainda menor, representando cerca de 15% dos formandos. A baixa inserção nessa área, contudo, não é o único fator que corrobora esses dados; o alto índice de abandono durante o curso é outro aspecto importante que tem contribuído para a sub-representação feminina nas ciências. A evasão feminina nos cursos de graduação configura um problema relevante e está associada a fatores como a falta de representatividade e os estereótipos de gênero (Silva *et al.*, 2023).

O projeto Rede Nacional de Mulheres na Nanociência promove a inserção de meninas, professoras, pesquisadoras e empreendedoras nas áreas de Nanociência, Nanotecnologia

(N&N) e STEM. A iniciativa atua desde a educação básica até o ensino superior para despertar o interesse científico precoce, fortalecer a rede de pesquisadoras e reduzir a evasão universitária. Para otimizar as ações, o projeto divide-se em núcleos regionais e desenvolve atividades práticas, como visitas a laboratórios e criação de materiais educacionais. Com abrangência nacional, a rede busca impulsionar o avanço científico e tecnológico de forma homogênea no Brasil. A consolidação do projeto demanda um portal Web estratégico para centralizar a comunicação com a sociedade e instituições parceiras. A plataforma deve dar visibilidade às ações, eventos e materiais, além de registrar as atividades e o progresso das bolsistas. Atualmente, a rede utiliza um site em WordPress gratuito, mas que limita o desenvolvimento devido a restrições no gerenciamento de permissões e na inserção de vídeos. Por isso, é necessário a refatoração do site para um ambiente Web que ofereça maior controle e liberdade de desenvolvimento.

Portanto, o objetivo deste trabalho é apresentar o desenvolvimento do sistema de gerenciamento de conteúdo para o portal da Rede Nacional de Mulheres na Nanociência, o qual centraliza as informações e os materiais gerados pelo projeto. O sistema também auxilia no registro e na divulgação das atividades realizadas pela rede, viabilizando a criação e o gerenciamento de permissões para novos usuários, que alimentarão as informações do portal, além de facilitar a comunicação e a difusão da iniciativa junto à comunidade.

2 REFERENCIAL TEÓRICO

Esta seção aborda os conceitos centrais e fundamentos teóricos relacionados ao desenvolvimento do portal.

2.1 SISTEMA DE GERENCIAMENTO DE CONTEÚDO

Um Sistema de Gerenciamento de Conteúdo ou *Content Management System* (CMS) é uma plataforma tecnológica desenvolvida para estruturar, organizar e publicar conteúdos digitais de maneira sistematizada. De acordo com Silva e colegas (2024), o CMS funciona como uma camada de abstração sobre o código-fonte de uma aplicação Web. Ele viabiliza a operação, a edição e a manutenção de portais por múltiplos usuários de forma simultânea, dispensando a necessidade de conhecimentos avançados em programação ou Engenharia de Software para a realização de atualizações rotineiras.

Segundo a IBM (2025), a arquitetura de um CMS tradicional separa a lógica de apresentação dos dados brutos para garantir a consistência funcional e visual do sistema. Essa divisão ocorre através do *Content Management Application* (CMA), a interface gráfica do *back-end* que permite gerenciar conteúdos e permissões de forma visual, e do *Content*

Delivery Application (CDA), a infraestrutura de segundo plano que processa e transforma esses dados armazenados nas páginas do *front-end* visíveis aos usuários.

Na Engenharia de Software contemporânea, o CMS vai além da edição de textos, integrando fluxos de aprovação com privilégios customizáveis, controle de versão, histórico de segurança, modularidade e otimização para motores de busca (Silva *et al.*, 2024). Essa ferramenta torna-se estratégica academicamente por descentralizar o fluxo de informações e otimizar a difusão científica sem comprometer a integridade do código-fonte principal.

2.2 COMPUTAÇÃO EM NUVEM, SISTEMA WEB E DOCKER

A Computação em Nuvem refere-se à capacidade de acessar arquivos e executar serviços variados por meio da internet. Segundo Kolbe Júnior (2020), esse paradigma consiste no fornecimento de recursos computacionais, tais como servidores, bancos de dados e armazenamento, sob demanda através da rede, denominada de “nuvem”. Esse modelo fundamenta-se no conceito de virtualização, o qual se manifesta tanto no nível de servidores, que utilizam *hardware* físico compartilhado para hospedar múltiplas máquinas virtuais, quanto no nível de aplicações, no qual um único *software* hospedado é disponibilizado para diversos usuários de forma simultânea (Esférica Tecnologia, 2020).

Um sistema Web consiste em um modelo de aplicação distribuída no qual a lógica de negócio, o processamento e o armazenamento de dados residem centralizados em servidores remotos, enquanto a interação com o usuário final ocorre de forma descentralizada por meio de navegadores (*browsers*). Diferente dos *softwares desktop* tradicionais, os sistemas Web operam sob a arquitetura cliente-servidor e utilizam protocolos de comunicação padronizados, *Hypertext Transfer Protocol* (HTTP) e *HyperText Transfer Protocol Secure* (HTTPS), para a requisição e a transferência de dados (Sommerville, 2011). Essa infraestrutura confere alta acessibilidade à aplicação, eliminando a necessidade de instalações locais nas máquinas dos usuários e garantindo a entrega dinâmica de interfaces responsivas construídas com tecnologias como *HyperText Markup Language* (HTML), *Cascading Style Sheets* (CSS) e JavaScript (JS).

No desenvolvimento do sistema apresentado neste trabalho, o Docker foi utilizado para estruturar o ambiente de execução da aplicação Web e dos serviços necessários ao seu funcionamento. O Docker é uma plataforma de código aberto utilizada para desenvolver, distribuir e executar aplicações em *containers* de modo padronizado, rápido e portátil (Docker Inc., 2026). A plataforma propicia previsibilidade entre diferentes ambientes ao longo das etapas de desenvolvimento, teste e implementação, à medida que empacota uma aplicação junto com as bibliotecas e configurações necessárias para o seu funcionamento. De acordo com a Docker Inc. (2026), alguns dos conceitos-base relacionados ao funcionamento do

Docker são a imagem e o *container*. A imagem é um modelo somente de leitura que contém todas as informações necessárias para que uma aplicação seja executada por meio de um *container*. Os *containers* são unidades executáveis de uma imagem que reúnem o código da aplicação, dependências e bibliotecas, sendo, por padrão, bem isolados dos demais.

2.3 RECURSOS PARA PROJETO DE SOFTWARE

No desenvolvimento do portal da Rede Nacional de Mulheres na Nanociência, estruturado como um Sistema de Gerenciamento de Conteúdo (CMS), foram adotados o *framework* Scrum e a ferramenta Kanban para a gestão ágil do projeto. Adicionalmente, integraram-se outras tecnologias e ferramentas computacionais para viabilizar a implementação dos recursos do sistema.

As metodologias ágeis baseiam-se em uma abordagem incremental para a especificação, o desenvolvimento e a entrega do software, priorizando a satisfação do cliente (ou o dono do negócio) e promovendo entregas contínuas e funcionais de partes do sistema. A abordagem Scrum é composta por três fases: a primeira estabelece o objetivo geral do projeto e a arquitetura do sistema; após isso, ocorre uma série de *sprints*, em cada um dos quais incrementos são adicionados ao sistema; por fim, há a fase de encerramento do projeto, com a finalização da documentação e dos manuais do usuário (Sommerville, 2011).

Alguns dos elementos-chave do Scrum são as reuniões semanais entre os membros da equipe, a definição do *Backlog* do produto[8] e os *sprints*, que são períodos de tempo curtos, semanais ou mensais, nos quais as tarefas distribuídas aos membros da equipe devem ser entregues. A equipe responsável pelo emprego da metodologia Scrum é chamada de *Scrum Team* e é composta pelo *Product Owner*, pelo *Development Team* e pelo *Scrum Master*. O *Product Owner* é o dono do produto que será desenvolvido e tem como principal função definir os itens do *Backlog* do produto. O *Development Team* é composto por profissionais responsáveis pelo desenvolvimento do produto, em geral pequenos grupos multifuncionais que realizam as entregas incrementais. E por fim, há o *Scrum Master*, que é o responsável por garantir que o Scrum seja entendido e aplicado por meio da comunicação clara dos objetivos e itens do *Backlog* para o time de desenvolvimento na liderança e treinamento da organização na adoção do Scrum (Schwaber; Sutherland, 2013).

Um método que se integra adequadamente à metodologia Scrum é o Kanban, uma técnica visual utilizada para o gerenciamento do fluxo de trabalho. Segundo a *Kanban University* (2026), essa abordagem visa à melhoria evolutiva de processos, permitindo visualizar

8 Lista ordenada de tudo o que deve conter no sistema e de quais funcionalidades devem ser priorizadas.

e otimizar fluxos de trabalho em projetos de software sem impor papéis ou responsabilidades prescritivas (customização de processos, tarefas, papéis, etc).

Para o desenvolvimento *back-end*, foram utilizados a linguagem de programação Python e o *framework* Django. Segundo a *Python Software Foundation* (2026), o Python é uma linguagem de alto nível, de código aberto, tipada dinamicamente e orientada a objetos. Essa linguagem é mantida por uma comunidade ativa de desenvolvedores e possui diversas bibliotecas e *frameworks* que possibilitam sua aplicação em diferentes contextos da Engenharia de Software. O Django, por sua vez, é um *framework* escrito em Python para o desenvolvimento de aplicações Web; ele dispõe de uma camada de mapeamento objeto-relacional (ORM) que permite a interação com bancos de dados, o processamento de validações, um painel de administração nativo com gerenciamento de *Create, Read, Update* e *Delete* (CRUD) e um sistema de rotas, além de adotar a arquitetura *Model-View-Template* (MVT) A arquitetura MVT organiza a estrutura da aplicação e é composta pelo *Model*, responsável pela representação dos dados, pelas regras de negócio e pela interação com o banco de dados; pela *View*, que constitui a parte lógica do sistema, recebendo a requisição, processando os dados e selecionando o template para a renderização; e, por fim, pelo *Template*, que define a interface visual com a qual o usuário interage. (Django Software Foundation, 2026).

Para o desenvolvimento *front-end* do sistema, foi empregado o *framework* Bootstrap, uma ferramenta de código aberto utilizada para facilitar a estilização de páginas Web e agilizar a criação de sites responsivos. O *framework* disponibiliza coleções de códigos, componentes e recursos reutilizáveis de HTML, CSS e JavaScript, os quais potencializam a criação de páginas e facilitam a adaptação das interfaces aos diferentes tamanhos de tela (Alura, 2021). Para o controle de versionamento, o gerenciamento de código-fonte e a colaboração entre os desenvolvedores, foram utilizadas as ferramentas Git e GitHub. De acordo com Chacon e Straub (2014), o Git é um sistema de controle de versão que registra as alterações em arquivos ao longo do tempo, permitindo recuperar versões específicas.

Complementarmente, o GitHub consiste em um serviço de hospedagem Web para repositórios Git que apoia o trabalho de desenvolvedores e analistas de sistemas. A plataforma hospeda o repositório principal, que contém o código-fonte do sistema, permitindo que os membros da equipe realizem cópias para seus próprios computadores, efetuem as atualizações necessárias e enviem as modificações de volta ao repositório central (Microsoft, 2024). Para o gerenciamento do ambiente de *containers*, foi utilizado o *Docker Desktop*, uma aplicação que permite criar, compartilhar e executar aplicativos e microsserviços em *containers* a partir de uma interface gráfica de usuário (Docker Inc., 2026). A utilização desse aplicativo facilitou o gerenciamento dos *containers* responsáveis pela execução da aplicação Web e do banco de dados, contribuindo para a organização e a padronização do ambiente de desenvolvimento.

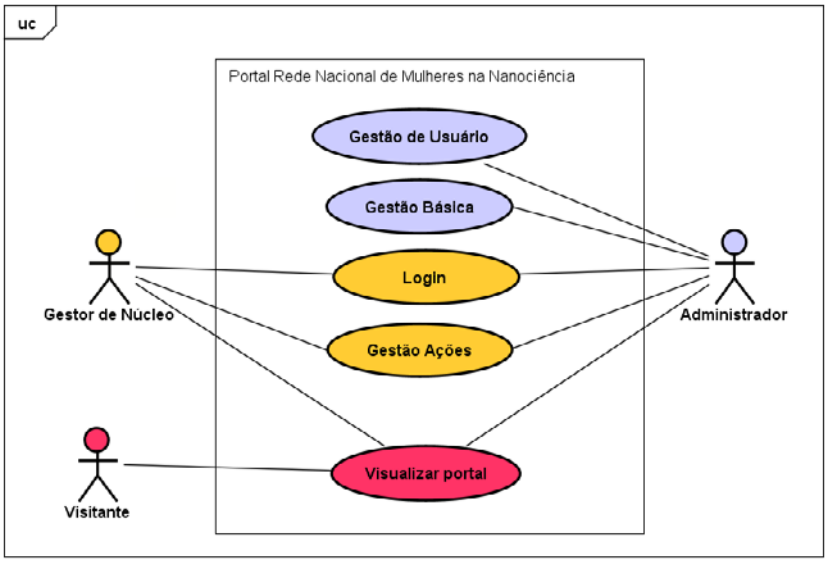
3 METODOLOGIA

Para a escrita do artigo, utilizou-se a pesquisa de revisão bibliográfica. Em relação ao desenvolvimento deste projeto de software, optou-se pela metodologia ágil Scrum, apoiada pelo método Kanban. As ferramentas tecnológicas empregadas foram: Python, Django, Bootstrap, GitHub e Docker. Os encontros para a construção do sistema ocorreram semanalmente, utilizando-se o ambiente de desenvolvimento Visual Studio Code, associado ao repositório de versionamento de código no GitHub. O responsável técnico pelo Laboratório de Práticas da Computação da Universidade Franciscana atuou como *Scrum Master*; as bolsistas, como equipe de desenvolvimento (*Development Team)*; e a coordenadora da Rede Nacional de Mulheres na Nanociência, como cliente ou *Product Owner*. A modelagem do sistema concentrou-se nos aspectos funcionais, sendo fundamentada no *Product Backlog* e nos diagramas da *Unified Modeling Language* (UML). Destaca-se que, com o objetivo de garantir a clareza textual, a coesão normativa e o rigor da norma da língua portuguesa, utilizou-se a ferramenta de Inteligência Artificial generativa Gemini (desenvolvida pela Google). A tecnologia foi empregada exclusivamente como suporte na revisão gramatical dos parágrafos autorais que compõem este trabalho, sem interferência na produção intelectual ou no conteúdo científico apresentado.

A caracterização de um sistema de *software* pressupõe a descrição de suas dimensões estruturais e funcionais. A dimensão estrutural abrange a arquitetura estática da aplicação, englobando o esquema do banco de dados e o diagrama de classe, por exemplo. Em contrapartida, a dimensão funcional correlaciona-se com o comportamento dinâmico e com os requisitos funcionais ou não funcionais esperados para a plataforma.

Segundo Sommerville (2011), os casos de uso são eficazes para identificar como os usuários interagem com o sistema. A Figura 1 representa o diagrama de casos de uso desenvolvido para o sistema apresentado neste artigo. No diagrama, estão presentes três atores: o visitante, o administrador e o gestor de núcleo, que representam os usuários do sistema e as interações que estes podem realizar de acordo com os seus níveis de permissão. O visitante pode apenas visualizar as informações disponíveis na interface pública do portal, ao passo que o administrador e o gestor de núcleo podem autenticar-se no sistema de gerenciamento para executar ações operacionais. O gestor de núcleo é responsável exclusivamente por administrar as ações do seu próprio núcleo, tais como eventos, notícias e atividades. Por sua vez, o administrador possui acesso a todas as funcionalidades do sistema, sendo responsável pela gestão dos demais usuários e pelo gerenciamento básico do portal, o qual contempla produtos educacionais, participantes, premiações e instituições.

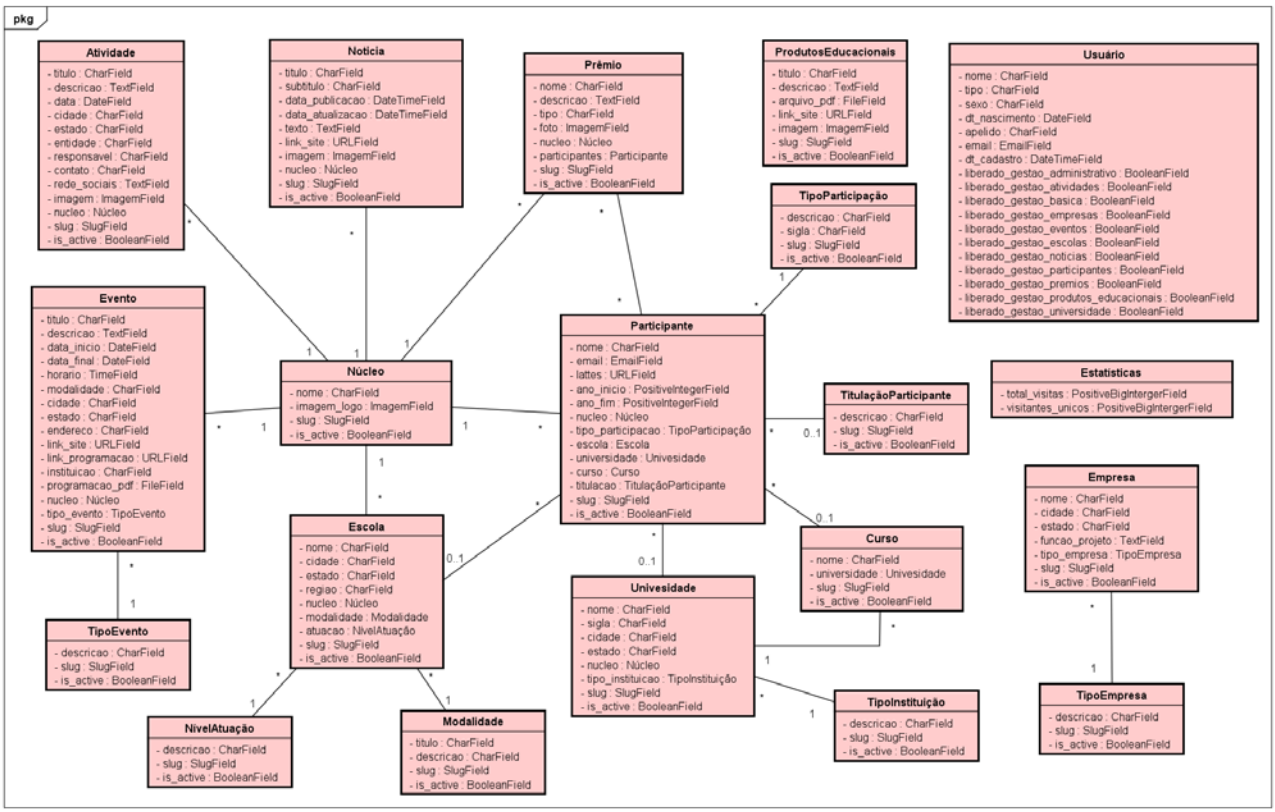
Figura 1 - Diagrama de Caso de Uso.



Fonte**:** Construção dos Autores.

Os diagramas de classes representam a estrutura de um sistema ao exibir suas entidades de dados, atributos e associações (Sommerville, 2011). Seus componentes principais são as classes, que modelam os objetos essenciais do *software*, e as relações que definem como essas classes interagem em diferentes níveis de detalhamento.

Figura 2 - Diagrama customizado de Classe.



Fonte**:** Construção dos Autores.

A Figura 2 apresenta o diagrama de classes customizado para o portal do projeto, ilustrando a perspectiva de seu aspecto estrutural. O modelo contempla as entidades e os seus respectivos atributos; entre elas, destacam-se as classes Núcleo e Participante, as quais detêm relacionamentos com a maioria das demais estruturas, configurando-se como elementos centrais da aplicação. Adicionalmente, entidades de categorização são utilizadas para apoiar a organização interna das outras classes. Os relacionamentos e as multiplicidades representados no diagrama contribuem para a otimização da estrutura de dados e para a implementação das funcionalidades requeridas no sistema.

A apresentação e a respectiva análise desses dois diagramas cumprem o propósito de ilustrar as dimensões funcionais e estruturais do sistema proposto. Enquanto a modelagem de casos de uso delimita o comportamento dinâmico da plataforma, o diagrama de classes mapeia a arquitetura estática dos dados, evidenciando as entidades, os atributos e os módulos que viabilizam o funcionamento do CMS conforme os requisitos estabelecidos.

4 RESULTADOS E DISCUSSÕES

Esta seção apresenta os resultados e as principais funcionalidades do Sistema de Gerenciamento de Conteúdo do Portal da Rede Nacional de Mulheres na Nanociência. A implementação do sistema centralizou a gestão de informações e materiais, otimizando a organização e a manutenção dos conteúdos. Além disso, o CMS estruturou e integrou diferentes tipos de dados à interface pública, fortalecendo a divulgação das ações e atividades desenvolvidas pela Rede.

A interface pública do sistema está estruturada em diversas páginas, as quais são destinadas à divulgação da rede, das atividades e das produções vinculadas ao projeto. Entre as páginas, destaca-se a de *Webinar*, destinada aos encontros virtuais com especialistas convidados, em que o visitante poderá acessar as informações referentes ao encontro, bem como assisti-lo por meio da incorporação de vídeos disponibilizados no YouTube. Há também a página do Simpósio, que reúne informações (como local e site oficial) sobre os eventos promovidos pela rede. Há uma seção de notícias destinada à divulgação de reportagens e de ações que envolvam a rede ou suas participantes.

O portal também disponibiliza páginas destinadas à listagem de instituições (escolas, universidades e empresas) e de participantes (bolsistas, pesquisadoras e voluntárias) vinculadas ao projeto. Além disso, há uma página com fotos e informações de premiações recebidas por integrantes da rede, outra de produtos educacionais, que fornece acesso e divulga materiais didáticos, e, por fim, as páginas institucionais, que apresentam informações acerca da rede, como objetivos, composição dos núcleos e do comitê gestor. As

Figuras 3 e 4 apresentam a tela inicial do portal e a página com informações sobre a rede, respectivamente.

Figura 3 - Imagem da tela inicial (*home page*) do portal.



Fonte**:** Construção dos Autores.

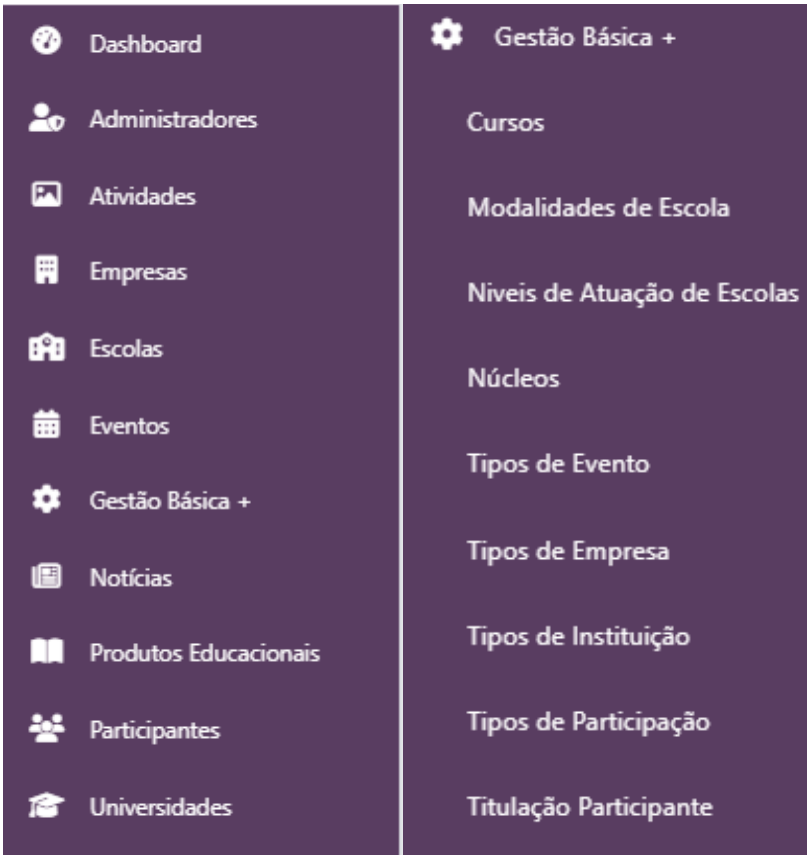
Figura 4 - Imagem da página que contém informações sobre a rede.



Fonte**:** Construção dos Autores.

O Sistema de Gerenciamento de Conteúdo do portal disponibiliza diversas funcionalidades. Entre elas, estão o gerenciamento de escolas, universidades e empresas que fazem parte do projeto; o cadastro de bolsistas, pesquisadoras e voluntárias vinculadas à rede; o registro de premiações recebidas pelas participantes; e a divulgação de notícias e eventos. Dentre as principais funcionalidades do sistema, destacam-se o registro das atividades promovidas pela rede, a gestão dos produtos educacionais desenvolvidos pelas participantes e o gerenciamento de usuários e suas permissões. A Figura 5 lista todas as funcionalidades de gerenciamento presentes no portal.

Figura 5 - Imagem da visão de um administrador do gerenciamento.



Fonte**:** Construção dos Autores.

O registro das atividades da rede é de extrema importância para o acompanhamento das ações promovidas nos diferentes núcleos. Esse registro é feito por meio do fornecimento de dados como a descrição da atividade, o local, a data e o núcleo em que ela foi realizada, além de fotos tiradas ao longo do desenvolvimento da ação. Para quem visita o portal, esses registros podem ser acessados na página da galeria (Figura 6), a qual está dividida de acordo com os núcleos e também conta com uma seção que exibe as atividades praticadas em conjunto por mais de uma região. Por questões de privacidade, para este trabalho, as fotos registradas durante as atividades foram substituídas pelas logomarcas dos núcleos.

Figura 6 - Imagem da página da galeria.

Fonte**:** Construção dos Autores.

A gestão dos produtos educacionais (Figura 7) é um dos pontos principais do sistema, pois, dentre os objetivos que a rede busca alcançar, estão o maior contato das meninas com materiais didáticos relacionados à nanociência e o acesso mais igualitário a esses conteúdos em todo o país. Para que isso ocorra, o projeto busca incentivar o desenvolvimento de jogos e atividades didáticas pelas próprias alunas e professoras integrantes da Rede Nacional de Mulheres na Nanociência. Portanto, gerenciar, divulgar e permitir o acesso a esses produtos educacionais (por meio de *download* ou de direcionamento para sites externos) é fundamental para o portal (Figura 8).

Figura 7 - Imagem da gestão de produtos educacionais



Fonte**:** Construção dos Autores.

Figura 8 - Imagem da página que exibe os produtos educacionais.



Fonte**:** Construção dos Autores.

O gerenciamento de usuários que acessam o CMS e de suas respectivas permissões é realizado pelos administradores do sistema. Esses administradores possuem acesso a todas as funcionalidades disponíveis e são responsáveis pelo cadastro dos demais usuários. No projeto atual, existem três usuários com esse perfil: as duas desenvolvedoras do sistema e a coordenadora geral da Rede Nacional de Mulheres na Nanociência. Os demais usuários recebem permissões restritas às funcionalidades que irão gerenciar (Figura 9). Por exemplo, cada núcleo terá uma bolsista responsável por registrar atividades, cadastrar eventos e divulgar notícias relacionadas à sua região; logo, essa bolsista terá acesso apenas às funcionalidades de atividades, eventos e notícias.

Figura 9 - Gestão de permissões de usuários.



Fonte**:** Construção dos Autores.

5 CONSIDERAÇÕES

Neste artigo, foram abordadas as fundamentações teóricas acerca de Sistemas de Gerenciamento de Conteúdo, a utilização da Computação em Nuvem por meio de aplicações Web e do Docker, bem como a metodologia de trabalho e os recursos tecnológicos empregados no desenvolvimento do CMS para o portal da Rede Nacional de Mulheres na Nanociência. Ao longo do estudo, detalhou-se a etapa de modelagem mediante a elaboração de diagramas que representam tanto as interações dos usuários com o sistema quanto a arquitetura estática de suas entidades e respectivos relacionamentos. Por fim, apresentaram-se os resultados práticos obtidos, descrevendo-se as principais funcionalidades e as interfaces do portal e de sua área administrativa. Como produto principal deste projeto, a plataforma da Rede Nacional de Mulheres na Nanociência encontra-se desenvolvida e homologada para acesso público. Com a disponibilização desse ambiente virtual, o objetivo geral desta pesquisa foi plenamente alcançado: estabelecer uma plataforma centralizada para a difusão de informações, objetivos institucionais e materiais didáticos produzidos pela Rede. Adicionalmente, o sistema viabiliza o registro otimizado de atividades, eventos e notícias que alimentam o portal, potencializando a comunicação e a projeção do grupo junto à comunidade científica e à sociedade.

Para trabalhos futuros, propõe-se o estabelecimento de uma rotina de manutenção preventiva e de atualização contínua do código-fonte, assegurando a estabilidade, o desempenho e a segurança da aplicação. Outrossim, sugere-se a implementação progressiva de novas funcionalidades e o refinamento dos módulos existentes, alinhando a plataforma às demandas emergentes e à evolução natural das atividades da Rede.

REFERÊNCIAS

ALURA. Bootstrap: o que é, como usar o framework Bootstrap no HTML e criar sites responsivos. **Alura**, 20 jul. 2021. Disponível em: https://www.alura.com.br/artigos/bootstrap. Acesso em: 16 ago. 2026.

BRASILEIRO, Ítalo B.; SHINTAKU, M.; MACÊDO, D. J.; COSTA, L. R.; MOURA, R. dos S. de; SCHIESSL, I. T.; SANTOS, M. C. de C. dos; MENÊSES, R. da V. de. Plugin de integração do WordPress com Tematres. **Anais do Workshop de Informação, Dados e Tecnologia - WIDaT**, *[S. l.]*, v. 6, 2023. DOI 10.22477/vi.widat.37. Disponível em: https://widat.ibict.br/index.php/widat2023/article/view/37. Acesso em: 16 ago. 2026.

CHACON, Scott; STRAUB, Ben. **Pro Git**. 2. ed. *[S. l.]*: Apress, 2014. Disponível em: https://git-scm.com/book/en/v2/Getting-Started-What-is-Git%3F. Acesso em: 16 ago. 2026.

DJANGO SOFTWARE FOUNDATION. Documentação do Django. **Django**, 2026. Disponível em: https://docs.djangoproject.com/pt-br/6.1/. Acesso em: 16 ago. 2026.

DOCKER INC. What is Docker? **Docker Docs**, 2026. Disponível em: https://docs.docker.com/get-started/docker-overview/. Acesso em: 16 ago. 2026.

ESFÉRICA TECNOLOGIA. **Computação em nuvem**. Versão 1.0. 2020. Disponível em: https://esferica.com.br/wp-content/uploads/2020/12/ebook-Cloud-Computing-v1.0-Capa.pdf. Acesso em: 16 ago. 2026.

IBM. What is a Content Management System (CMS)? **IBM Think Topics**, 2025. Disponível em: https://www.ibm.com/think/topics/content-management-system. Acesso em: 18 ago. 2026.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA. **Estatísticas de gênero**: indicadores sociais das mulheres no Brasil. 3. ed. Rio de Janeiro: IBGE, 2024. Disponível em: https://agenciadenoticias.ibge.gov.br/media/com_mediaibge/arquivos/9ac298aaf1203418036ae00bf1272e92.pdf. Acesso em: 11 ago. 2026.

KANBAN UNIVERSITY. The Kanban Method. **Kanban University**, 2026. Disponível em: https://kanban.university/kanban-guide/. Acesso em: 21 ago. 2026.

KOLBE JÚNIOR, Armando. **Computação em nuvem**. 1. ed. São Paulo: Contentus, 2020. E-book. Disponível em: https://plataforma.bvirtual.com.br. Acesso em: 16 ago 2026.

MICROSOFT. Conceitos básicos do Git e do GitHub para a documentação do Microsoft Learn. **Microsoft Learn**, 2024. Disponível em: https://learn.microsoft.com/pt-br/contribute/content/git-github-fundamentals. Acesso em: 16 ago. 2026.

NAIDEKA, N.; SANTOSA, Y. H.; SOARESA, P.; HELLINGERA, R.; HACKA, T.; ORTH, E. S. Mulheres cientistas na química brasileira. **Química Nova**, v. 43, n. 6, p. 823-836, 2020. DOI 10.21577/0100-4042.20170556. Disponível em: https://www.scielo.br/j/qn/a/L8mY73ZKdMk6SVntnN8hLHt/?lang=pt. Acesso em: 10 ago. 2026.

PYTHON SOFTWARE FOUNDATION. About Python. **Python.org**, 2026. Disponível em: https://www.python.org/about/. Acesso em: 16 ago. 2026.

SCHWABER, Ken; SUTHERLAND, Jeff. **Guia do Scrum**: um guia definitivo para o Scrum: as regras do jogo. 2013. Disponível em: https://scrumguides.org/docs/scrumguide/v1/Scrum-Guide-Portuguese-BR.pdf. Acesso em: 15 ago. 2026.

SILVA, J. R. *et al.* O impacto de Sistemas de Gerenciamento de Conteúdo (SGC/CMS) na governança de portais institucionais. In: **Anais do Congresso de Tecnologia da Informação**, v. 14, n. 2, p. 112-125, 2024.

SILVA, U. F.; FERREIRA, D. J.; AMBRÓSIO, A. P. L.; OLIVEIRA, J. L dos S. Problemas enfrentados por alunas de graduação em ciência da computação: uma revisão sistemática. **Educação e Pesquisa**, São Paulo, v. 48, e236643, 2022. DOI 10.1590/S1678-4634202248236643por. Disponível em: https://www.scielo.br/j/ep/a/M4W8kqJcTxBcCQhjpYsyMpR/?lang=pt. Acesso em: 11 ago. 2026.

SOMMERVILLE, I. **Engenharia de software**. 9. ed. São Paulo: Pearson, 2011. E-book. Disponível em: https://plataforma.bvirtual.com.br. Acesso em: 15 ago 2026.

SOUTO, Daniela Cruz; SOUTO, Renata Cruz. Importância das iniciativas de inserção de meninas e mulheres na área de STEM no Brasil. **Revista Ibero-Americana de Humanidades, Ciências e Educação**, *[S. l.]*, v. 8, n. 10, p. 4319-4333, 2022. DOI 10.51891/rease.v8i10.7478. Disponível em: https://periodicorease.pro.br/rease/article/view/7478. Acesso em: 10 ago. 2026.

VULNERABILIDADES E SEGURANÇA NO CÓDIGO GERADO POR INTELIGÊNCIA ARTIFICIAL

VULNERABILITIES AND SECURITY IN ARTIFICIAL INTELLIGENCE GENERATED CODE

Yasmin Machado Flôres[1], Bruno Tubino Franco[2], Guilherme Gonsalves Carlin[3] e Leonardo Teixeira Gomes[4]

RESUMO

A utilização de Inteligência Artificial Generativa no desenvolvimento de software vem modificando a forma como programadores produzem, analisam e corrigem código. Ferramentas baseadas em Grandes Modelos de Linguagem são capazes de gerar programas a partir de descrições em linguagem natural, proporcionando maior automação no processo de desenvolvimento. Entretanto, a utilização dessas tecnologias também apresenta riscos relacionados à introdução de vulnerabilidades, à confiança dos desenvolvedores nos resultados produzidos e à possibilidade de manipulação dos dados utilizados no treinamento dos modelos. Este trabalho tem como objetivo analisar os riscos de segurança associados ao código gerado por Inteligência Artificial, identificando as principais vulnerabilidades, os fatores que contribuem para sua ocorrência e as estratégias propostas pela literatura para sua mitigação. A pesquisa consiste em uma revisão bibliográfica fundamentada em oito trabalhos científicos que investigam diferentes dimensões da segurança de código gerado por Inteligência Artificial. São analisados estudos experimentais sobre código gerado, estudos envolvendo usuários, técnicas de controle da geração, ataques de envenenamento de dados, engenharia de *prompts* e *benchmarks* de avaliação. A análise demonstra que modelos de Inteligência Artificial podem produzir código funcionalmente adequado, porém vulnerável, e que a utilização dessas ferramentas pode influenciar também a percepção dos desenvolvedores sobre a segurança de suas próprias implementações. Conclui-se que a geração automatizada de código não oferece garantia

1 Acadêmica do curso de Ciência da Computação. Universidade Franciscana. E-mail: y.machado@ufn.edu.br. ORCID: https://orcid.org/0009-0002-7241-4379
2 Acadêmico do curso de Ciência da Computação. Universidade Franciscana. E-mail: tubino.bruno@ufn.edu.br. ORCID: https://orcid.org/0009-0004-0824-6088
3 Acadêmico do curso de Ciência da Computação. Universidade Franciscana. E-mail: guilherme.cgonsalves@ufn.edu.br. ORCID: https://orcid.org/0009-0007-2784-1343
4 Acadêmico do curso de Ciência da Computação. Universidade Franciscana. E-mail: leonardo.tgomes@ufn.edu.br. ORCID: https://orcid.org/0009-0001-9649-613X

automática de segurança, sendo necessária a combinação entre assistência por Inteligência Artificial, conhecimento do desenvolvedor e mecanismos sistemáticos de validação.

Palavras-chave: modelos de linguagem; desenvolvimento de software; engenharia de prompts; ferramentas de programação.

ABSTRACT

The use of Generative Artificial Intelligence in software development has changed the way programmers produce, analyze, and correct code. Tools based on Large Language Models are capable of generating programs from natural language descriptions, providing increased automation throughout the software development process. However, the use of these technologies also introduces risks related to vulnerability generation, developers' trust in generated outputs, and the possibility of manipulating data used to train the models. This study aims to analyze the security risks associated with Artificial Intelligence generated code, identifying the main vulnerabilities, the factors contributing to their occurrence, and the strategies proposed in the literature for their mitigation. The research consists of a literature review based on eight scientific studies investigating different dimensions of the security of Artificial Intelligence generated code. Experimental studies on generated code, user studies, generation control techniques, data poisoning[5] attacks, prompt engineering techniques, and security evaluation benchmarks are analyzed. The analysis demonstrates that Artificial Intelligence models can produce functionally adequate but vulnerable code, and that the use of these tools can also influence developers' perception of the security of their own implementations. It is concluded that automated code generation does not provide an automatic security guarantee, requiring a combination of Artificial Intelligence assistance, developer knowledge, and systematic validation.

Keywords: *language models; software development; prompt engineering; programming tools.*

1 INTRODUÇÃO

A Inteligência Artificial Generativa passou a ocupar espaço significativo no desenvolvimento de software por meio de ferramentas capazes de produzir código a partir de descrições realizadas em linguagem natural. Grandes Modelos de Linguagem, conhecidos como *Large Language Models* ou LLMs, podem auxiliar na implementação de funcionalidades, complementação de código, correção de erros e produção de diferentes componentes de sistemas computacionais.

A capacidade de geração automática modifica determinadas etapas do desenvolvimento tradicional de software. Em vez de escrever manualmente cada instrução, o desenvolvedor

5 Ataque que consiste na inserção ou manipulação de dados maliciosos no conjunto utilizado para treinar um modelo, com o objetivo de influenciar seu comportamento.

pode descrever uma necessidade e utilizar a resposta de um modelo como ponto de partida para a implementação. Essa característica pode ampliar a capacidade de produção do programador, mas também cria uma nova relação de dependência entre o desenvolvedor e o sistema de Inteligência Artificial.

Entre as principais questões decorrentes dessa transformação encontra-se a segurança do código produzido. Um modelo pode gerar uma implementação que atende à funcionalidade solicitada e, ao mesmo tempo, introduzir uma vulnerabilidade que não seja imediatamente percebida pelo usuário. Pearce *et al.* (2021) realizaram uma das primeiras análises sistemáticas desse problema utilizando o GitHub Copilot. Os pesquisadores construíram 89 cenários relacionados a vulnerabilidades conhecidas e utilizaram a ferramenta para produzir 1.689 programas. Aproximadamente 40% dos programas gerados apresentaram alguma vulnerabilidade. O resultado demonstra que a capacidade de produzir código funcional não implica necessariamente a produção de código seguro.

A questão também envolve o comportamento dos desenvolvedores. Perry *et al.* (2022) realizaram um estudo com usuários de um assistente de programação baseado no modelo Codex[6][7] e observaram que os participantes que tinham acesso à ferramenta produziram código significativamente menos seguro do que aqueles que não tinham acesso. Além disso, os participantes que utilizaram a Inteligência Artificial apresentaram maior probabilidade de acreditar que haviam produzido código seguro. O problema, portanto, não está restrito à capacidade técnica do modelo; existe também uma dimensão relacionada à forma como os resultados são interpretados e utilizados.

Fu *et al.* (2023) ampliaram a investigação para códigos associados a projetos do GitHub. Ao analisarem 733 *snippets*[8] produzidos por ferramentas de geração de código, os autores encontraram problemas de segurança em 29,5% dos *snippets* Python e em 24,2% dos *snippets* JavaScript. Os problemas estavam distribuídos em 43 categorias da *Common Weakness Enumeration*[9], incluindo vulnerabilidades como *Cross-Site Scripting*[10] e uso de valores insuficientemente aleatórios.

Além da geração de código inseguro, existe uma ameaça relacionada ao próprio processo de treinamento dos modelos. Improta (2024) discute ataques de envenenamento

6 Modelo de geração de código desenvolvido pela OpenAI, baseado na arquitetura GPT e utilizado como assistente de programação.

7 Modelo de geração de código desenvolvido pela OpenAI, baseado na arquitetura GPT, utilizado como base de assistentes de programação como o GitHub Copilot.

8 Pequenos trechos de código-fonte, geralmente utilizados como exemplos ou fragmentos reaproveitáveis.

9 Catálogo de categorias de vulnerabilidades e fraquezas de software, mantido pela organização MITRE e utilizado como referência para classificação de falhas de segurança.

10 Tipo de vulnerabilidade que permite a um atacante injetar scripts maliciosos em páginas web visualizadas por outros usuários.

de dados, nos quais amostras maliciosas podem ser inseridas nos dados utilizados para treinamento de modelos de geração de código. Dessa forma, um atacante pode buscar influenciar o comportamento futuro do sistema sem necessariamente interferir diretamente na interação entre o desenvolvedor e a ferramenta.

Diante desse cenário, este trabalho busca responder ao seguinte problema de pesquisa: quais são os principais riscos de segurança associados ao código gerado por Inteligência Artificial e quais estratégias podem ser utilizadas para reduzir esses riscos?

O objetivo geral é analisar os riscos de segurança relacionados ao código produzido por Grandes Modelos de Linguagem. Como objetivos específicos, busca-se identificar a ocorrência de vulnerabilidades em código gerado por Inteligência Artificial, analisar a influência da ferramenta sobre o comportamento dos desenvolvedores, investigar riscos relacionados aos dados de treinamento e apresentar técnicas e métodos propostos para aumentar a segurança do código gerado. Para alcançar esses objetivos, é realizada uma revisão bibliográfica baseada em oito trabalhos científicos selecionados como núcleo da pesquisa: Pearce *et al.* (2021), Perry *et al.* (2022), Fu *et al.* (2023), He e Vechev (2023), Improta (2024), Bruni *et al.* (2025), SECODEPLT (2025) e A.S.E. (2025).

2 GERAÇÃO DE CÓDIGO POR INTELIGÊNCIA ARTIFICIAL

Os Grandes Modelos de Linguagem são sistemas capazes de identificar padrões presentes em grandes volumes de dados e produzir novas sequências de texto a partir de determinadas entradas. No desenvolvimento de software, essa capacidade permite gerar código a partir de descrições fornecidas pelo programador.

A geração de código apresenta uma característica importante para a discussão de segurança. O modelo não executa necessariamente uma análise completa das consequências de segurança de cada implementação que produz; seu comportamento é resultado dos padrões aprendidos durante o treinamento e das informações presentes na solicitação recebida. He e Vechev (2023) destacam a necessidade de considerar propriedades de segurança durante a geração de código, formulando o problema da geração controlada, no qual o modelo precisa ser orientado a produzir código que apresente determinada propriedade de segurança sem perder sua capacidade funcional.

Essa questão evidencia uma diferença fundamental entre correção funcional e segurança: um programa pode executar corretamente uma determinada função e ainda possuir uma vulnerabilidade. Essa distinção é fundamental para compreender os resultados apresentados pelos demais estudos, já que a avaliação de uma ferramenta de geração de código não deve considerar apenas se o código produzido funciona,

mas também se sua implementação é resistente a situações que podem comprometer a segurança do sistema.

3 VULNERABILIDADES EM CÓDIGO GERADO POR IA

3.1 O ESTUDO DE PEARCE *ET AL.*

Pearce *et al.* (2021) investigaram a segurança das contribuições produzidas pelo GitHub Copilot por meio de cenários construídos a partir de vulnerabilidades conhecidas.

Foram elaborados 89 cenários e produzidos 1.689 programas. A análise mostrou que aproximadamente 40% das gerações apresentaram alguma vulnerabilidade. O resultado representa uma evidência relevante porque demonstra que o problema pode ocorrer de maneira recorrente: a vulnerabilidade não precisa ser resultado de uma falha excepcional do sistema, e, dependendo da tarefa apresentada ao modelo, uma parcela significativa das respostas pode apresentar problemas.

Esse resultado estabelece uma das principais bases para a discussão deste trabalho: a geração automática de código não deve ser interpretada como geração automática de código seguro.

3.2 EVIDÊNCIAS EM PROJETOS DO GITHUB

Fu *et al.* (2023) investigaram uma situação mais próxima do desenvolvimento realizado em projetos públicos. O estudo analisou *snippets* produzidos por ferramentas como GitHub Copilot, CodeWhisperer e Codeium presentes em projetos do GitHub.

Foram identificados 733 *snippets*. Entre eles, 29,5% dos códigos Python e 24,2% dos códigos JavaScript apresentaram problemas de segurança. As vulnerabilidades estavam distribuídas em 43 categorias CWE, entre elas a CWE-330, relacionada ao uso de valores insuficientemente aleatórios, a CWE-94, relacionada ao controle inadequado da geração de código, e a CWE-79, relacionada a *Cross-Site Scripting*.

Os pesquisadores também avaliaram a capacidade do Copilot Chat de corrigir vulnerabilidades quando recebia alertas provenientes de ferramentas de análise estática: nesse cenário, até 55,5% dos problemas puderam ser corrigidos. Esse resultado é relevante porque apresenta uma característica dupla da Inteligência Artificial - a mesma tecnologia capaz de introduzir vulnerabilidades também pode auxiliar na identificação e correção de parte delas.

4 A RELAÇÃO ENTRE SEGURANÇA E COMPORTAMENTO DO DESENVOLVEDOR

A análise de segurança não pode considerar somente o código produzido pelo modelo. A forma como os desenvolvedores utilizam a ferramenta também influencia o resultado.

Perry *et al.* (2022) realizaram um estudo de usuários para avaliar como a disponibilidade de um assistente de programação influencia a produção de código relacionado a tarefas de segurança. Os participantes que tiveram acesso ao assistente baseado no Codex produziram código menos seguro do que os participantes sem assistência. Entretanto, os usuários da ferramenta também apresentaram maior probabilidade de acreditar que haviam produzido código seguro.

Esse resultado é especialmente importante porque indica a existência de uma possível diferença entre segurança real e percepção de segurança. Uma ferramenta pode produzir uma resposta tecnicamente convincente, fazendo com que o desenvolvedor atribua maior confiança ao código; se essa confiança não for acompanhada por uma revisão adequada, uma vulnerabilidade pode permanecer no sistema.

Os pesquisadores também observaram que usuários que confiavam menos na Inteligência Artificial e que interagiam mais cuidadosamente com a linguagem e o formato dos *prompts* apresentaram menos vulnerabilidades. Dessa forma, a utilização segura de ferramentas de geração de código depende também do comportamento do desenvolvedor: o sistema deve ser compreendido como uma ferramenta de assistência, e não como uma autoridade capaz de garantir a segurança do código produzido.

5 ATAQUES CONTRA OS DADOS DE TREINAMENTO

Os estudos analisados apresentam ainda uma dimensão diferente do risco: a possibilidade de manipulação dos dados utilizados para treinar os modelos. Improta (2024) discute o problema de *data poisoning*, ou envenenamento de dados, direcionado a modelos utilizados para geração de código.

Grandes modelos podem ser treinados com volumes extensos de informações coletadas de fontes disponíveis na Internet. Essa característica pode criar uma superfície de ataque, pois um adversário pode tentar inserir amostras especialmente construídas nos dados utilizados durante o treinamento. O trabalho apresenta um ataque de envenenamento capaz de fazer com que modelos de geração produzam código vulnerável, e a preocupação é ampliada pelo fato de que uma alteração maliciosa pode ser construída de maneira a manter características esperadas do código, dificultando sua identificação.

Esse tipo de ameaça modifica a compreensão do problema. A vulnerabilidade não precisa surgir somente no momento em que o desenvolvedor solicita uma solução ao

modelo; ela pode estar relacionada ao próprio conhecimento incorporado ao modelo durante o treinamento. Consequentemente, a segurança de um sistema de geração de código depende não apenas da qualidade da arquitetura e do *prompt*, mas também da confiabilidade dos dados utilizados em seu desenvolvimento.

6 ESTRATÉGIAS PARA AUMENTAR A SEGURANÇA

6.1 CONTROLE DA GERAÇÃO

He e Vechev (2023) apresentam uma estratégia denominada SVEN, desenvolvida para controlar propriedades de segurança durante a geração de código.

O método utiliza vetores específicos para orientar a geração do modelo sem modificar diretamente seus pesos, permitindo que o modelo seja direcionado para produzir código associado a uma determinada propriedade de segurança. Nos experimentos, um modelo CodeGen com 2,7 bilhões de parâmetros produziu código seguro em 59,1% dos casos; com a aplicação do SVEN para segurança, essa proporção aumentou para 92,3%.

Um resultado importante é que o método manteve desempenho próximo ao modelo original em relação à correção funcional. Isso demonstra que segurança e funcionalidade não precisam necessariamente ser tratadas como propriedades incompatíveis, sendo possível desenvolver mecanismos que orientem a geração para determinados requisitos de segurança.

6.2 ENGENHARIA DE *PROMPTS*

Bruni *et al.* (2025) investigaram uma estratégia diferente, baseada na forma como os *prompts* são formulados.

O trabalho avaliou modelos GPT e diferentes técnicas de engenharia de *prompts* voltadas à geração segura de código. Os resultados indicaram que a utilização de instruções explicitamente direcionadas à segurança pode reduzir a ocorrência de vulnerabilidades em determinadas condições. Os autores também investigaram *prompting* iterativo, no qual o modelo participa de um processo de identificação e correção de vulnerabilidades.

Essa abordagem é relevante por representar uma forma de mitigação que não depende necessariamente da alteração da arquitetura ou dos parâmetros do modelo - o *prompt* pode funcionar como uma camada adicional de orientação. Entretanto, os resultados não significam que a engenharia de *prompts* seja suficiente para garantir segurança; ela deve ser compreendida como uma estratégia complementar dentro de um processo maior de validação.

7 A NECESSIDADE DE *BENCHMARKS* DE SEGURANÇA

Os trabalhos mais recentes analisados demonstram que avaliar a segurança de modelos de geração de código é uma tarefa cada vez mais complexa. O SECODEPLT (2025) foi desenvolvido com o objetivo de fornecer um *benchmark* unificado para avaliar os riscos e capacidades de segurança de sistemas de geração de código.

O *benchmark* possui mais de 5,9 mil amostras distribuídas em 44 categorias de risco baseadas na CWE, contemplando diferentes linguagens de programação. O estudo avalia diferentes capacidades dos modelos, incluindo geração de código seguro, identificação de vulnerabilidades e geração de correções.

Uma característica importante do SECODEPLT é a preocupação com os métodos de avaliação: os autores consideram que avaliações baseadas exclusivamente em regras estáticas ou em julgamentos de modelos de linguagem podem apresentar limitações, motivo pelo qual o *benchmark* incorpora mecanismos de avaliação dinâmica. Essa característica demonstra uma mudança importante na pesquisa - não basta perguntar se um modelo consegue gerar código, é necessário estabelecer métodos capazes de avaliar sistematicamente quão seguro é o código produzido.

8 AVALIAÇÃO EM NÍVEL DE REPOSITÓRIO

O A.S.E. (2025) amplia essa discussão ao avaliar segurança em nível de repositório. Uma limitação de muitos *benchmarks* é a utilização de pequenos trechos de código isolados; em projetos reais, entretanto, uma vulnerabilidade pode depender da interação entre diferentes arquivos, funções, dependências e componentes.

O A.S.E. foi desenvolvido para considerar esse contexto. O *benchmark* utiliza repositórios reais do GitHub associados a vulnerabilidades documentadas por CVEs[11]. A base possui 120 instâncias, sendo 40 conjuntos originais e 80 variantes modificadas semanticamente e estruturalmente.

O *benchmark* considera quatro categorias de vulnerabilidades em aplicações web: *SQL Injection*[12], *Path Traversal*[13], *Cross-Site Scripting* e *Command Injection*[14].

11Common Vulnerabilities and Exposures: sistema de identificação padronizada de vulnerabilidades de segurança conhecidas, mantido publicamente como referência para a comunidade de segurança da informação.
12Vulnerabilidade que ocorre quando entradas controladas pelo usuário são interpretadas como comandos SQL pelo sistema.
13Vulnerabilidade que permite acessar arquivos ou diretórios fora do caminho autorizado pela aplicação.
14Vulnerabilidade que permite inserir comandos de sistema operacional em uma aplicação de modo que sejam executados pelo sistema.

A avaliação demonstra que modelos avançados ainda apresentam dificuldades quando precisam produzir código seguro considerando o contexto de um repositório completo. Esse resultado é importante para a discussão porque demonstra que bons resultados em pequenos exemplos não significam necessariamente que um modelo esteja preparado para desenvolver ou modificar sistemas completos de maneira segura.

9 DISCUSSÃO

Os oito trabalhos analisados podem ser organizados em diferentes dimensões do mesmo problema. A Tabela 1 sintetiza o foco, a amostra e o resultado principal de cada estudo utilizado como núcleo desta revisão.

Pearce *et al.* (2021) estabelecem uma evidência experimental de que ferramentas de geração de código podem produzir vulnerabilidades. Fu *et al.* (2023) ampliam essa constatação ao analisar *snippets* associados a projetos do GitHub, demonstrando que problemas semelhantes podem ser observados em um contexto mais próximo do desenvolvimento real.

Perry *et al.* (2022) acrescentam o fator humano: o risco não está somente na resposta produzida pelo modelo, mas também na maneira como o desenvolvedor interpreta essa resposta. A maior confiança observada entre os usuários do assistente é particularmente relevante porque pode reduzir a percepção da necessidade de revisão. Improta (2024) amplia a superfície do problema para o treinamento, demonstrando que a segurança precisa ser considerada desde a origem das informações utilizadas pelo modelo.

Tabela 1 - Síntese dos estudos analisados.

Estudo	Foco	Amostra	Resultado principal
Pearce *et al.* (2021)	Segurança do GitHub Copilot	89 cenários / 1.689 programas	Aproximadamente 40% dos programas gerados apresentaram vulnerabilidade
Perry *et al.* (2022)	Comportamento do desenvolvedor	Estudo com usuários (Codex)	Código menos seguro produzido por usuários assistidos, com maior confiança indevida
Fu *et al.* (2023)	Snippets em projetos GitHub	733 *snippets*	29,5% (Python) e 24,2% (JavaScript) com problemas, em 43 categorias CWE
He e Vechev (2023)	Controle da geração (SVEN)	CodeGen com 2,7 bilhões de parâmetros	Código seguro elevado de 59,1% para 92,3%
Improta (2024)	Envenenamento de dados de treinamento	Ataque experimental de *data poisoning*	Demonstra viabilidade de ataques que induzem geração de código vulnerável
Bruni *et al.* (2025)	Engenharia de *prompts*	Modelos GPT	Prompts direcionados à segurança reduzem a ocorrência de vulnerabilidades

Estudo	Foco	Amostra	Resultado principal
SECODEPLT (2025)	Benchmark de segurança	5,9 mil amostras / 44 categorias CWE	Avaliação dinâmica de múltiplas capacidades de segurança
A.S.E. (2025)	Benchmark em nível de repositório	120 instâncias (40 originais + 80 variantes)	Modelos avançados ainda apresentam dificuldades no contexto de repositório completo

Fonte: Construção dos autores, com base em Pearce *et al.* (2021), Perry *et al.* (2022), Fu *et al.* (2023), He e Vechev (2023), Improta (2024), Bruni *et al.* (2025), SECODEPLT (2025) e A.S.E. (2025).

Por outro lado, He e Vechev (2023) demonstram que os modelos podem ser direcionados para aumentar a segurança da geração, e Bruni *et al.* (2025) apresentam outra possibilidade por meio da engenharia de *prompts*. Para tornar essa comparação mais objetiva, a Tabela 2 reúne os principais percentuais reportados pelos estudos empíricos, incluindo o efeito da mitigação proposta por He e Vechev (2023).

Tabela 2 - Percentuais de vulnerabilidade e efeito da mitigação (SVEN).

Estudo / Condição	Percentual observado
Pearce *et al.* (2021) - programas com vulnerabilidade (Copilot)	≈ 40%
Fu *et al.* (2023) - *snippets* Python com vulnerabilidade	29,5%
Fu *et al.* (2023) - *snippets* JavaScript com vulnerabilidade	24,2%
Fu *et al.* (2023) - vulnerabilidades corrigidas pelo Copilot Chat	até 55,5%
He e Vechev (2023) - código seguro sem SVEN (CodeGen 2,7B)	59,1%
He e Vechev (2023) - código seguro com SVEN (CodeGen 2,7B)	92,3%

Fonte: Construção dos autores, com base em Pearce *et al.* (2021), Fu *et al.* (2023) e He e Vechev (2023).

Os números reunidos na Tabela 2 evidenciam dois pontos centrais da revisão: de um lado, uma parcela relevante do código gerado por diferentes ferramentas apresenta vulnerabilidades; de outro, técnicas de mitigação como o SVEN são capazes de reduzir esse problema de forma expressiva, sem comprometer a correção funcional do modelo. Esses resultados indicam que o problema não deve ser analisado de maneira binária - o resultado depende do modelo, do contexto, da tarefa, do *prompt*, do treinamento e do processo de validação utilizado posteriormente.

Os trabalhos SECODEPLT (2025) e A.S.E. (2025) reforçam essa conclusão ao demonstrar a necessidade de avaliações mais abrangentes. Existe, portanto, uma evolução metodológica na área: os primeiros estudos buscaram identificar se modelos produziam código vulnerável, e estudos posteriores passaram a investigar o comportamento dos usuários, mecanismos de mitigação, ataques contra o treinamento e *benchmarks* capazes de avaliar diferentes dimensões da segurança.

Essa evolução é importante porque evidencia que segurança não é uma propriedade automática da geração de código. Ela precisa ser explicitamente considerada durante o processo.

10. VANTAGENS E LIMITAÇÕES DA UTILIZAÇÃO DE IA

Os oito artigos selecionados concentram-se principalmente nos aspectos de segurança e, por isso, não permitem estabelecer uma estimativa abrangente de economia financeira ou de tempo de desenvolvimento. Entretanto, os próprios estudos demonstram que as ferramentas são capazes de produzir código funcional e auxiliar em determinadas atividades de programação.

Fu *et al.* (2023), por exemplo, demonstraram que o Copilot Chat conseguiu corrigir até 55,5% dos problemas de segurança analisados quando recebeu informações provenientes de ferramentas de análise estática. He e Vechev (2023) também demonstraram que é possível aumentar significativamente a proporção de código seguro sem eliminar a capacidade funcional do modelo.

Esses resultados indicam que a Inteligência Artificial possui valor potencial como ferramenta de assistência, especialmente quando integrada a mecanismos de avaliação. Entretanto, os mesmos estudos demonstram limitações: a existência de vulnerabilidades em uma parcela significativa das gerações, o aumento da confiança observado no estudo de Perry *et al.* (2022) e os riscos associados ao treinamento demonstrados por Improta (2024) impedem que a ferramenta seja considerada uma substituta da revisão humana e dos mecanismos tradicionais de segurança.

Portanto, com base exclusivamente no núcleo científico selecionado, não é possível afirmar que a Inteligência Artificial necessariamente economiza determinado percentual de tempo ou dinheiro no desenvolvimento. O que pode ser sustentado é que ela oferece capacidade de assistência e automação, enquanto sua utilização também introduz novos requisitos de validação e segurança.

11 CONCLUSÃO

A análise dos oito trabalhos científicos selecionados demonstra que o uso de Inteligência Artificial para geração de código apresenta benefícios potenciais, mas também riscos concretos de segurança.

Pearce *et al.* (2021) demonstraram que aproximadamente 40% das gerações avaliadas em seu estudo apresentaram vulnerabilidades. Fu *et al.* (2023) encontraram problemas de segurança em 29,5% dos *snippets* Python e 24,2% dos *snippets* JavaScript analisados em projetos do GitHub. Esses resultados demonstram que código funcionalmente adequado não deve ser automaticamente considerado seguro. O estudo de Perry *et al.* (2022) acrescenta uma dimensão comportamental ao problema ao demonstrar que usuários assistidos

por Inteligência Artificial podem produzir código menos seguro e, simultaneamente, apresentar maior confiança na segurança de suas soluções.

Improta (2024) amplia a discussão ao apresentar a possibilidade de ataques de envenenamento de dados, indicando que a segurança dos modelos também depende da integridade dos dados utilizados durante seu treinamento. Apesar desses riscos, os trabalhos analisados também demonstram possibilidades de mitigação: He e Vechev (2023) apresentaram o SVEN, capaz de aumentar significativamente a proporção de código seguro produzido por determinado modelo, e Bruni *et al.* (2025) investigaram técnicas de engenharia de *prompts* voltadas à redução de vulnerabilidades.

Os trabalhos SECODEPLT (2025) e A.S.E. (2025) demonstram ainda que a avaliação da segurança precisa acompanhar a evolução dos modelos. *Benchmarks* mais amplos e avaliações em nível de repositório são necessários para compreender o comportamento dos sistemas em situações mais próximas do desenvolvimento real.

Dessa forma, a principal conclusão deste trabalho é que a geração de código por Inteligência Artificial não deve ser confundida com geração de código seguro. A Inteligência Artificial pode atuar como uma ferramenta de assistência, podendo produzir código, auxiliar na identificação de problemas e participar da correção de vulnerabilidades; entretanto, a segurança precisa ser tratada como uma propriedade específica, submetida a avaliação e validação. O uso responsável dessas ferramentas depende, portanto, da combinação entre capacidade de geração, conhecimento do desenvolvedor, revisão do código e mecanismos sistemáticos de análise. A IA pode fazer parte de um processo de desenvolvimento seguro, mas os resultados apresentados pelos trabalhos analisados não permitem tratá-la como uma garantia automática de segurança.

REFERÊNCIAS

PEARCE, Hammond; AHMAD, Baleegh; TAN, Benjamin; DOLAN-GAVITT, Brendan; KARRI, Ramesh. **Asleep at the Keyboard? Assessing the Security of GitHub Copilot's Code Contributions.** 2021. Disponível em: https://arxiv.org/abs/2108.09293. Acesso em: 29 ago. 2026.

PERRY, Neil; SRIVASTAVA, Megha; KUMAR, Deepak; BONEH, Dan. **Do Users Write More Insecure Code with AI Assistants?** 2022. Disponível em: https://arxiv.org/abs/2211.03622. Acesso em: 29 ago. 2026.

FU, Yujia; LIANG, Peng; TAHIR, Amjed; LI, Zengyang; SHAHIN, Mojtaba; YU, Jiaxin; CHEN, Jinfu. **Security Weaknesses of Copilot-Generated Code in GitHub Projects: An Empirical Study.** 2023. Disponível em: https://arxiv.org/abs/2310.02059. Acesso em: 29 ago. 2026.

HE, Jingxuan; VECHEV, Martin. Large Language Models for Code: Security Hardening and Adversarial Testing. In: ACM SIGSAC CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY, 2023. **Proceedings [...].** New York: ACM, 2023. DOI: 10.1145/3576915.3623175. Disponível em: https://doi.org/10.1145/3576915.3623175. Acesso em: 29 ago. 2026.

IMPROTA, Cristina. **Poisoning Programs by Un-Repairing Code: Security Concerns of AI-generated Code.** 2024. Disponível em: https://arxiv.org/abs/2403.06675. Acesso em: 29 ago. 2026.

BRUNI, Marc; GABRIELLI, Fabio; GHAFARI, Mohammad; KROPP, Martin. **Benchmarking Prompt Engineering Techniques for Secure Code Generation with GPT Models.** 2025. Disponível em: https://arxiv.org/abs/2502.06039. Acesso em: 29 ago. 2026.

NIE, Yuzhou *et al.* SECODEPLT: A Unified Benchmark for Evaluating the Security Risks and Capabilities of Code GenAI. In: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS, 38., 2025. **Proceedings [...].** 2025.

LIAN, Keke *et al.* **A.S.E: A Repository-Level Benchmark for Evaluating Security in AI-Generated Code.** 2025. Disponível em: https://arxiv.org/abs/2508.18106. Acesso em: 29 ago. 2026.

SIRC
SIMPÓSIO DE INFORMÁTICA
Universidade Franciscana

15 a 16
set | 2026

Interconectando

códigos dados inteligência inovação

O XVII Simpósio de Informática (SIRC 2026) reúne trabalhos e reflexões desenvolvidos a partir do tema "Interconectando dados, códigos, inteligência e inovação". O evento promove o intercâmbio de conhecimentos entre estudantes, pesquisadores e profissionais, incentivando a discussão sobre tendências, tecnologias e práticas emergentes em Computação. A publicação registra contribuições acadêmicas que evidenciam a pesquisa, a inovação e a colaboração como elementos fundamentais para o avanço científico e tecnológico.

CIÊNCIA DA COMPUTAÇÃO
JOGOS DIGITAIS
SISTEMAS DE INFORMAÇÃO
editora UFN
UFN Universidade Franciscana