

DESENVOLVIMENTO DE UM SISTEMA WEB PARA ARQUIVAMENTO E ACESSO A TRABALHOS ACADÊMICOS

DEVELOPMENT OF A WEB SYSTEM FOR ARCHIVING AND ACCESSING ACADEMIC WORKS

Rafael Duailibe Lima Privado¹

Edilson Carlos Silva Lima²

Jonathan Araújo Queiroz³

Elda Regina de Sena Caridade⁴

RESUMO: A ausência de divulgação e acesso às pesquisas desenvolvidas em instituições de ensino superior limita o progresso acadêmico, dificultando que alunos encontrem trabalhos realizados por curso, área de conhecimento e docentes/discentes, um aspecto essencial para a continuidade das pesquisas. Este artigo apresenta um estudo de caso com o desenvolvimento de uma plataforma chamada GUARANI, que busca minimizar a inacessibilidade aos registros dessas pesquisas. O sistema foi implementado utilizando Node.js para a API, Bootstrap para o front-end e MySQL como banco de dados. A metodologia científica adotada foi de pesquisa exploratória com abordagem qualitativa, envolvendo a entrevista de 54 alunos dos cursos de Engenharia de Computação e Direito de uma instituição privada no Maranhão. Os resultados indicaram a aprovação do modelo de divulgação das pesquisas proposto pela plataforma.

Palavras-chave: Node.js. API. Plataforma Web.

ABSTRACT: The lack of dissemination and access to research developed in higher education institutions limits academic progress, making it difficult for students to find work carried out by course, area of knowledge and faculty/students, an essential aspect for the continuity of research. This article presents a case study with the development of a platform called GUARANI, which seeks to minimize the inaccessibility of records of these researches. The system was implemented using Node.js for the API, Bootstrap for the front-end and MySQL as the database. The scientific methodology adopted was exploratory research with a qualitative approach, involving the interview of 54 students of the Computer Engineering and Law courses of a private institution in Maranhão. The results indicated the approval of the research dissemination model proposed by the platform.

36

Keywords: Node.js. API. Web Platform.

I INTRODUÇÃO

A divulgação de pesquisas acadêmicas é essencial para promover o avanço científico e inspirar novos trabalhos dentro das instituições de ensino superior. Contudo, muitas vezes, os alunos não têm acesso adequado aos trabalhos desenvolvidos por discentes e docentes, como artigos, dissertações, teses e Trabalhos de Conclusão de Curso (TCC). Essa lacuna prejudica a

¹Formado em Engenharia de Computação na Universidade CEUMA,

²Orientador, professor da Universidade CEUMA,

³Coorientador, professor da Universidade CEUMA,

⁴Coorientadora, professora da Universidade CEUMA,

continuidade das pesquisas e limita a utilização dessas produções como referências para novos estudos. Além disso, a falta de uma plataforma acessível que organize essas informações por ano, *qualis*, área e subárea de conhecimento dificultam ainda mais o acesso e a visibilidade dessas produções acadêmicas.

Diante desse contexto, este artigo apresenta o desenvolvimento da plataforma GUARANI, projetada para solucionar a falta de acessibilidade às pesquisas realizadas na universidade. A GUARANI organiza e divulga os trabalhos acadêmicos por critérios como ano, área e subárea de conhecimento, tipo de pesquisa e orientadores. Essa ferramenta visa, assim, fornecer uma base de dados que sirva como inspiração e referência para alunos e pesquisadores.

O principal objetivo deste estudo é desenvolver a plataforma Guarani e avaliar se ela atende às necessidades de divulgação de pesquisas na universidade. A pesquisa justifica-se pela relevância de promover a continuidade e a visibilidade das produções científicas e tecnológicas realizadas, contribuindo para a sociedade e para a disseminação do conhecimento.

Para alcançar os objetivos propostos, a metodologia aplicada inclui um estudo de caso para o desenvolvimento da plataforma, utilizando Node.js para a API, Bootstrap para o front-end e MySQL como banco de dados. Na etapa de análise de resultados, foi adotada uma pesquisa exploratória de abordagem qualitativa, com entrevistas realizadas com 54 estudantes dos cursos de Engenharia de Computação e Direito de uma instituição privada no Maranhão.

Por meio dessa abordagem, o presente artigo busca evidenciar as contribuições da GUARANI para a solução de um problema prático e a promoção do acesso ao conhecimento dentro do ambiente acadêmico.

2 TRABALHOS RELACIONADOS

O presente capítulo discute três trabalhos que serviram como referências relacionadas de grande importância para o desenvolvimento do Projeto Guarani. Cada trabalho apresenta alguma relação com o tema de gestão e arquivamento de documentos acadêmicos, trazendo contribuições úteis, enquanto suas limitações justificam a necessidade do presente estudo.

2.1 Desenvolvimento de uma API para uma Plataforma de pesquisa de Trabalhos Científico

No artigo de Sousa (2023), descreve o desenvolvimento de uma solução voltada para o registro e acesso rápido a dados acadêmicos, com ênfase na área de engenharia elétrica. Este sistema propõe a digitalização e centralização dos registros, possibilitando uma gestão mais eficiente e acessível desses dados. A vantagem principal desse trabalho reside na centralização das informações acadêmicas, facilitando o acesso para os usuários e promovendo uma melhor gestão

do conhecimento. No entanto, uma das limitações é a falta de integração com outras plataformas e sistemas de gestão, o que pode dificultar sua aplicação em contextos variados.

Esse artigo serviu de referência direta para o Projeto Guarani, oferecendo uma base para o desenvolvimento de uma plataforma que centraliza e organiza trabalhos acadêmicos. Além disso, o artigo de Sousa (2023) sugere a incorporação futura de bibliotecas digitais, o que poderia ampliar o acesso aos registros acadêmicos e aumentar o alcance da plataforma.

2.2 O Uso do Vue.js no Desenvolvimento de uma Plataforma Web

No artigo de Bezerra (2023), apresenta o desenvolvimento de um sistema de gestão de documentos acadêmicos voltado para a engenharia de software. Esse sistema introduz uma inovação no uso de tecnologias de banco de dados, facilitando o armazenamento e a recuperação de dados acadêmicos, com ênfase na segurança e acessibilidade. Uma das principais vantagens do trabalho é a adoção de um banco de dados relacional, que garante a organização estruturada dos dados.

Contudo, o sistema possui limitações, como a falta de flexibilidade na personalização dos dados, não permitindo uma categorização específica para diferentes áreas de conhecimento. Como trabalhos futuros, o artigo sugere a ampliação do escopo para atender outros setores acadêmicos e a implementação de mecanismos avançados de busca e filtragem. Esse estudo foi uma referência direta para o desenvolvimento do Projeto Guarani, que também busca centralizar e organizar trabalhos acadêmicos em uma plataforma acessível e eficiente.

2.3 Depositório Digital de Trabalhos Acadêmicos

Por fim, o artigo de Vrana (2010), examina a relevância dos repositórios digitais no contexto educacional e como sua integração pode potencialmente revolucionar o acesso à informação e facilitar o aprendizado em ambientes acadêmicos. Esse estudo, publicado no *International Journal of Emerging Technology in Learning*, destaca que a falta de sistemas organizados para o armazenamento de trabalhos acadêmicos é um problema recorrente em diversas instituições, evidenciando que essa lacuna não se limita à Universidade Ceuma. Essa análise reforça a importância e a aplicabilidade do Projeto Guarani, mostrando que a iniciativa de criação de uma plataforma própria para gerenciar e arquivar trabalhos acadêmicos é uma resposta eficiente a um problema que afeta não só a Universidade Ceuma, mas também muitas outras instituições de ensino superior.

Entre os benefícios mencionados por Vrana (2010), destaca-se uma abordagem inovadora que aproxima várias plataformas digitais dos usuários, proporcionando acesso a vastos acervos de dados acadêmicos que, de outra forma, seriam difíceis de acessar. Contudo, uma das limitações apontadas no artigo é a dificuldade de integração com os sistemas já existentes nas instituições de ensino. Como desenvolvimento futuro, o estudo propõe interfaces de integração mais amigáveis, adaptáveis às diferentes necessidades institucionais.

2.4 Diferencial da plataforma GUARANI

A singularidade do Projeto Guarani, em comparação com os trabalhos mencionados, está em sua capacidade de resolver um problema específico identificado em uma instituição de ensino superior no estado do Maranhão, onde a ausência de um acervo de trabalhos acadêmicos. Enquanto os artigos Sousa (2023) e Bezerra (2023) focam em sistemas gerais de arquivamento e gestão de dados acadêmicos, o Guarani propõe-se a ser uma solução especialmente personalizada, relevante para gerenciar e organizar esses trabalhos de forma direcionada às necessidades específicas de uma universidade concreta.

Diferentemente dos mencionados, o Guarani também demanda que o sistema seja acessível e fácil de usar para estudantes e professores. A facilidade de uso e a simplicidade da interface são considerações fundamentais neste design, e o sistema deve ser escalável e flexível para futuras expansões e adaptações a outras áreas do conhecimento e exigências institucionais.

Enquanto o artigo Vrana (2010) aborda a integração de repositórios digitais, o Projeto Guarani nasce com a premissa de ser um repositório completo e acessível, voltado à preservação e consulta de trabalhos acadêmicos — aspectos que nos outros artigos ainda estão em desenvolvimento.

3 MÉTODOS

Neste capítulo, apresentaremos os métodos utilizados no desenvolvimento do projeto GUARANI. No item 3.1, abordaremos o desenvolvimento do back-end, enquanto no item 3.2 trataremos do front-end. Posteriormente, detalharemos o método de pesquisa adotado para a análise qualitativa da receptividade do sistema junto aos usuários.

Com o objetivo de promover o avanço científico e inspirar novos trabalhos acadêmicos, foi desenvolvido o estudo de caso denominado GUARANI. A iniciativa busca facilitar o acesso a produções acadêmicas, como artigos, dissertações, teses e Trabalhos de Conclusão de Curso (TCC). Este estudo surgiu da necessidade de tornar mais acessíveis as pesquisas realizadas dentro

da instituição, beneficiando tanto a comunidade acadêmica quanto o público em geral, ao mesmo tempo que incentiva a continuidade dos estudos e o desenvolvimento de novas pesquisas.

Além disso, o método adotado para avaliar a eficácia e aceitação da plataforma foi uma pesquisa exploratória de caráter qualitativo. Essa pesquisa foi realizada por meio de entrevistas com 54 dos cursos de Engenharia de Computação e Direito de uma instituição privada no Maranhão. Os estudantes compartilharam suas percepções sobre plataformas acadêmicas e sobre como o GUARANI poderia melhorar sua experiência no acesso a trabalhos acadêmicos.

3.1 Desenvolvimento do Back-End

O back-end do Projeto Guarani é fundamental para o gerenciamento e armazenamento seguro dos dados acadêmicos, garantindo integridade, escalabilidade e eficiência. Este sistema foi estruturado com tecnologias que suportam a comunicação entre o cliente (interface do usuário) e o banco de dados, por meio de uma API RESTful desenvolvida em Node.js. A seguir, detalharemos o papel da API, os métodos HTTP utilizados, o banco de dados MySQL e a integração entre os componentes do sistema.

3.1.1 API RESTful e Verbos HTTP

40

Uma API (Application Programming Interface) é um conjunto de definições e protocolos usados no desenvolvimento e integração de aplicativos segundo RedHat (2020). Uma API às vezes é descrita como um contrato entre um provedor e um consumidor de informações, estabelecendo o que o consumidor precisa (chamadas) e o que o produtor precisa (respostas). Por exemplo, um design de API para um serviço meteorológico pode especificar que o usuário forneça um CEP e o produtor responda em duas partes, com a primeira parte contendo a temperatura mais alta e a segunda parte contendo a temperatura mais baixa.

Em outras palavras, ao interagir com um computador ou sistema para recuperar informações ou executar uma função, a API ajudará a comunicar o que você deseja ao sistema para que o sistema entenda e execute sua solicitação.

Pense nas APIs como um mediador entre os usuários ou clientes e os recursos ou serviços web que eles querem obter. As APIs também servem para que organizações compartilhem recursos e informações e, ao mesmo tempo, mantenham a segurança, o controle e a obrigatoriedade de autenticação, pois permitem determinar quem tem acesso e o que pode ser acessado. (REDHAT, 2020)

Outra vantagem de usar a API é que você não precisa saber todos os detalhes sobre armazenamento em cache, como os recursos são recuperados ou de onde eles vêm. Uma API trabalha com um conjunto definido de regras que explicam como computadores ou aplicativos se

comunicam entre si. (Normalmente, essas regras são documentadas na especificação da API). Uma API utiliza requisições HTTP para fazer as requisições básicas necessárias para a manipulação dos dados (figura 1).

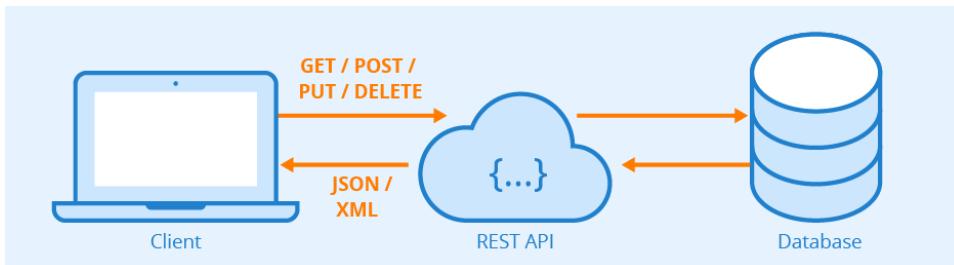


Figura 1. imagem representativa das conexões do api com banco de dados e cliente.

Fonte: SOMDAY, 2021

No Projeto Guarani, a API RESTful segue os princípios definidos por Roy Fielding (FIELDING, 2000), que estabelecem um padrão leve e escalável para manipulação de dados através de requisições HTTP. Esse modelo de API é amplamente utilizado por sua eficiência em aplicações web.

Os métodos HTTP (ou verbos) usados na API incluem:

- **GET:** Recupera dados do servidor, como uma lista de trabalhos acadêmicos ou detalhes de um trabalho específico.
- **POST:** Envia dados para o servidor, como o registro de um novo trabalho
- **PUT:** Atualiza informações de registros existentes, por exemplo, a edição de um trabalho acadêmico.
- **DELETE:** Exclui registros, permitindo a remoção de trabalhos do sistema quando necessário.

41

Essa estrutura de API permite ao Projeto Guarani lidar de forma ágil e estruturada com operações de CRUD (Create, Read, Update, Delete), garantindo a integridade dos dados e a simplicidade na comunicação entre o cliente e o banco de dados.

3.1.2 Banco de Dados MySQL

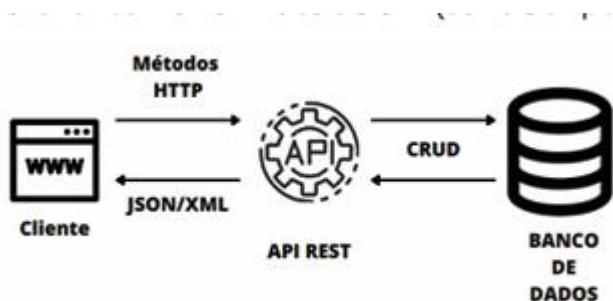
Para o armazenamento dos dados acadêmicos, foi utilizado o banco de dados relacional MySQL. De acordo com Silberschatz (2020), bancos de dados relacionais são essenciais para organizar e persistir dados de maneira estruturada, possibilitando fácil manipulação e consulta. O MySQL foi escolhido por ser uma solução de código aberto que oferece a robustez e a escalabilidade necessárias para projetos de médio e grande porte, como o Guarani (DATE, 2019).

Além disso, o MySQL Workbench foi empregado como uma interface gráfica para facilitar a administração e consulta dos dados, tornando o gerenciamento do banco de dados mais acessível (MARTINS, 2018).

3.1.3 Integração entre a API e o Banco de Dados

A integração entre o banco de dados MySQL e a API RESTful foi desenvolvida para assegurar uma comunicação eficiente e segura entre o front-end e o back-end do sistema. A API recebe requisições do cliente e, em seguida, interage com o banco de dados para buscar, armazenar ou atualizar informações acadêmicas. Os dados são retornados ao cliente no formato JSON (JavaScript Object Notation), um padrão leve que facilita a manipulação e o transporte de dados pela rede.

A arquitetura de integração é apresentada na Figura 2, onde a API atua como intermediária entre o cliente e o banco de dados, processando as requisições HTTP e transformando-as em operações de CRUD.



42

Figura 2. Arquitetura do modelo da API.

Fonte: Sousa, 2023.

Essa integração permite que a API RESTful atue como uma interface entre clientes (como front-end) e o banco de dados, garantindo uma comunicação eficiente e segura.

3.1.4 Node.js e Express.js

A API do Projeto Guarani foi construída utilizando Node.js, uma plataforma que permite a execução de JavaScript no lado do servidor. Node.js possui um modelo de execução assíncrono, o que possibilita manipular dados em tempo real com alta performance — uma característica essencial para sistemas de APIs (TILK, 2019).

Para complementar, foi utilizado o framework Express.js, que simplifica o desenvolvimento de servidores web e facilita a gestão de rotas. Com isso, o fluxo de dados entre o cliente e o banco de dados se torna mais fluido, melhorando a experiência do usuário ao navegar pela plataforma (YOUNG, 2020).

3.2 Front-end

O front-end da plataforma GUARANI foi projetado para oferecer uma interface acessível e intuitiva, permitindo aos usuários uma experiência fluida ao navegar pelas funcionalidades de pesquisa e cadastro de trabalhos acadêmicos. Para o desenvolvimento da interface visual, foi utilizado o framework Bootstrap, que combina HTML, CSS e JavaScript, garantindo um design responsivo, moderno e consistente, alinhado às melhores práticas de usabilidade.

Bootstrap é um framework front-end amplamente utilizado para o desenvolvimento de interfaces web responsivas e modernas. Criado para facilitar o design de sites e aplicativos, o Bootstrap oferece uma biblioteca de componentes pré-construídos, como botões, formulários, menus de navegação e muito mais, que ajudam a padronizar e acelerar o processo de desenvolvimento. Ele é baseado em HTML, CSS e JavaScript, que, em conjunto, permitem a criação de layouts consistentes e interativos (BOOTSTRAP, 2021).

1. HTML (HyperText Markup Language): No Bootstrap, o HTML fornece a estrutura básica para os elementos do site, como tabelas, formulários e botões. Através de classes predefinidas, como `btn` para botões ou `container` para organização de layouts, o HTML trabalha como a base que conecta o CSS e o JavaScript do framework.

2. CSS (Cascading Style Sheets): O CSS no Bootstrap define o estilo visual dos elementos HTML. Ele é responsável por cores, tipografia, margens, espaçamentos e ajustes para tornar os sites visualmente atraentes. Um dos grandes diferenciais do Bootstrap é o sistema de **grades responsivas**, que facilita a criação de layouts que se adaptam automaticamente a diferentes tamanhos de tela, como desktops, tablets e smartphones.

3. JavaScript: O JavaScript no Bootstrap adiciona interatividade e comportamento dinâmico aos elementos da interface. Com base em bibliotecas como o jQuery, ele possibilita a criação de componentes interativos, como sliders, modais, menus suspensos e validações de formulários. Esses recursos tornam os sites mais envolventes e práticos para os usuários.

O principal objetivo do Bootstrap é simplificar o processo de desenvolvimento front-end, permitindo que desenvolvedores criem interfaces responsivas de forma rápida e com um visual padronizado, alinhado às melhores práticas de design web.

Com o uso combinado dessas ferramentas e tecnologias, o front-end da Plataforma Guarani oferece aos usuários uma interface visualmente agradável, intuitiva e prática para acesso e gestão dos trabalhos acadêmicos, contribuindo diretamente para a melhoria do acesso à informação (NIELSEN, 2021).

4 RESULTADOS

A implementação do estudo de caso o projeto GUARANI surgiu como solução para a falta de acesso e divulgação dos trabalhos acadêmicos desenvolvidos em uma instituição de ensino superior, abordando uma lacuna crítica na promoção e continuidade da produção científica. Por meio do desenvolvimento de um sistema integrado, composto por uma API robusta em Node.js no *back-end* e um front-end responsivo utilizando Bootstrap, foi possível criar uma plataforma acessível e intuitiva. Essa solução organiza e disponibiliza dados acadêmicos de forma estruturada, permitindo o acesso por critérios como ano, qualis, área e subárea de conhecimento. Com isso, o projeto não apenas facilita a busca e a utilização de artigos, dissertações, teses e TCCs como referências, mas também fomenta a visibilidade das produções acadêmicas e o avanço científico dentro da instituição.

Alguns resultados do sistema, incluindo o código de implementação de funcionalidades principais e capturas de tela da interface da plataforma:

1. **Exclusão de Trabalhos Acadêmicos:** Na figura 3, temos um exemplo de código da rota de exclusão de trabalhos acadêmicos na API do Guarani. Utilizando o método HTTP **DELETE**, esta rota permite que registros sejam excluídos com base no identificador (ID) fornecido. Em caso de sucesso, a API retorna uma mensagem confirmando a exclusão; em caso de erro, uma mensagem de erro é retornada.

44

```
// Rota para excluir um trabalho acadêmico
app.delete('/api/trabalhos/:id', (req, res) => {
  const { id } = req.params;

  const query = 'DELETE FROM trabalhos WHERE id = ?';

  db.query(query, [id], (err, result) => {
    if (err) {
      console.error('Erro ao excluir o trabalho:', err);
      return res.status(500).json({ message: 'Erro ao excluir o trabalho.' });
    }

    res.json({ message: 'Trabalho acadêmico excluído com sucesso!' });
  });
});
```

Figura 3. Rota para excluir um trabalho acadêmico na API do Projeto Guarani.
Fonte: Autoral, 2024.

2. **Consulta de Usuários:** Outra funcionalidade implementada foi a consulta de todos os usuários registrados no sistema. A rota utiliza o método HTTP **GET** para recuperar todos os



registros da tabela de usuários, exibindo os resultados em formato JSON. Esse recurso é essencial para a administração do sistema (figura 4).

```
// Rota para obter todos os usuários
app.get('/api/usuarios', (req, res) => {
  const query = 'SELECT * FROM usuarios';

  db.query(query, (err, results) => {
    if (err) {
      console.error('Erro ao buscar usuários:', err);
      return res.status(500).json({ message: 'Erro ao buscar usuários.' });
    }

    res.json(results);
  });
});
```

Figura 4. Rota para obter todos os usuários na API do Projeto Guarani.

Fonte: Autoral, 2024.

3. **Interface Principal da Plataforma Guarani:** A interface da plataforma Guarani foi desenvolvida com foco em acessibilidade e usabilidade, permitindo que alunos e professores naveguem facilmente por categorias como monografias, artigos, dissertações e teses. Na figura 5, uma captura de tela da página inicial da plataforma, onde o usuário pode navegar pelas diferentes seções.



Figura 5. Interface principal da plataforma Guarani
Fonte: Autoral, 2024.

4. **Página de Login:** A página de login permite que os usuários façam autenticação para acessar funcionalidades restritas, como o cadastro e edição de trabalhos acadêmicos. A interface é simples e direta, garantindo que o acesso ao sistema seja rápido e seguro (Figura 6).

Figura 6. Página de login da plataforma Guarani

Fonte: Autoral, 2024.

5. **Cadastro de Trabalhos Acadêmicos:** Na plataforma Guarani, a funcionalidade de cadastro de trabalhos acadêmicos, permite que os usuários insiram informações detalhadas sobre cada trabalho, incluindo título, área de conhecimento, curso, autores e resumo. Na figura 7, a tela de cadastro de trabalhos acadêmicos.

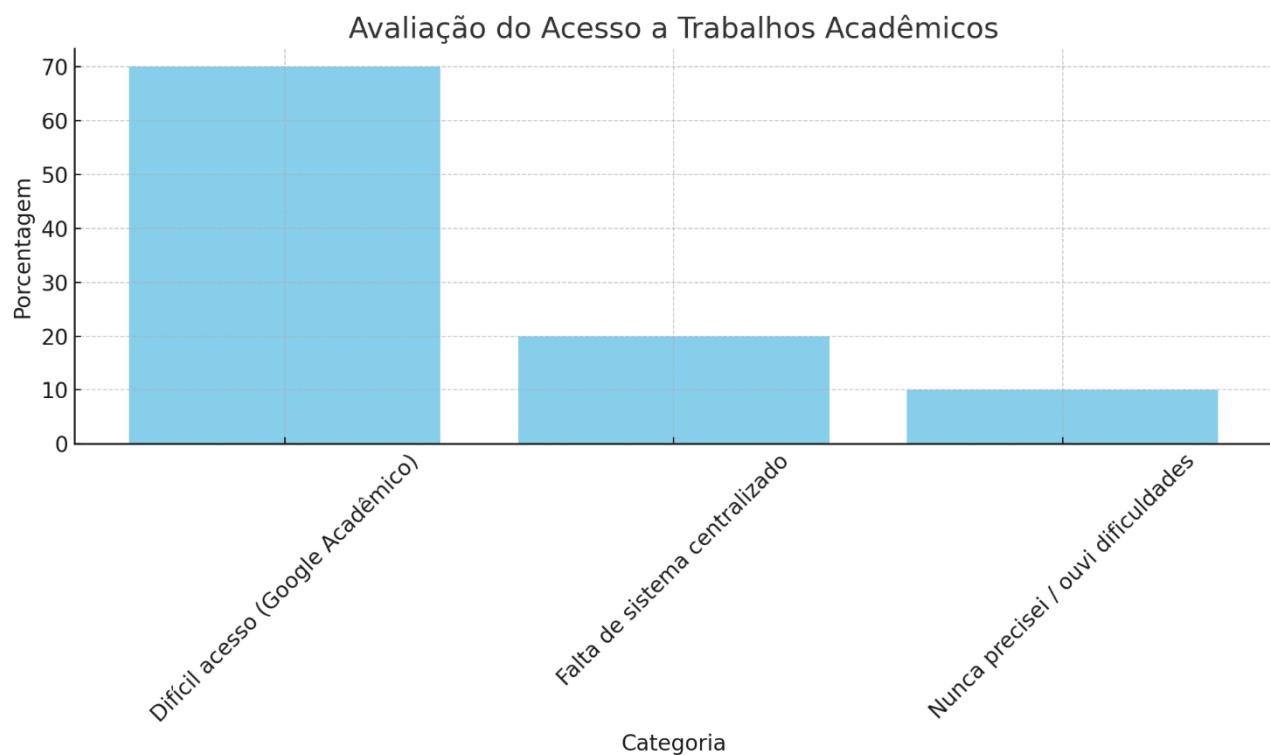
Figura 7. Tela de cadastro de trabalho acadêmico.

Fonte: Autoral, 2024.

Com o objetivo de avaliar a eficácia e a aceitação do sistema, foi conduzida uma pesquisa exploratória de abordagem qualitativa com 54 estudantes dos cursos de Engenharia de Computação e Direito em uma instituição privada no estado do Maranhão. Os resultados dessa investigação são apresentados a seguir, destacando a recepção positiva dos alunos e os benefícios proporcionados pelo GUARANI na resolução do problema de acesso limitado aos trabalhos acadêmicos.

No **Gráfico 1**, observa-se que uma parte significativa dos estudantes relatou dificuldade em acessar trabalhos acadêmicos na instituição. Cerca de 70% dos participantes indicaram que enfrentam desafios com o acesso a materiais acadêmicos, especialmente devido à dependência de fontes externas, como o Google Acadêmico. Esse dado evidencia uma lacuna significativa nas plataformas institucionais, o que dificulta o uso dos recursos da própria universidade.

Gráfico 1. Avaliação do Acesso a Trabalhos Acadêmicos.



Fonte: Autoral, 2024.

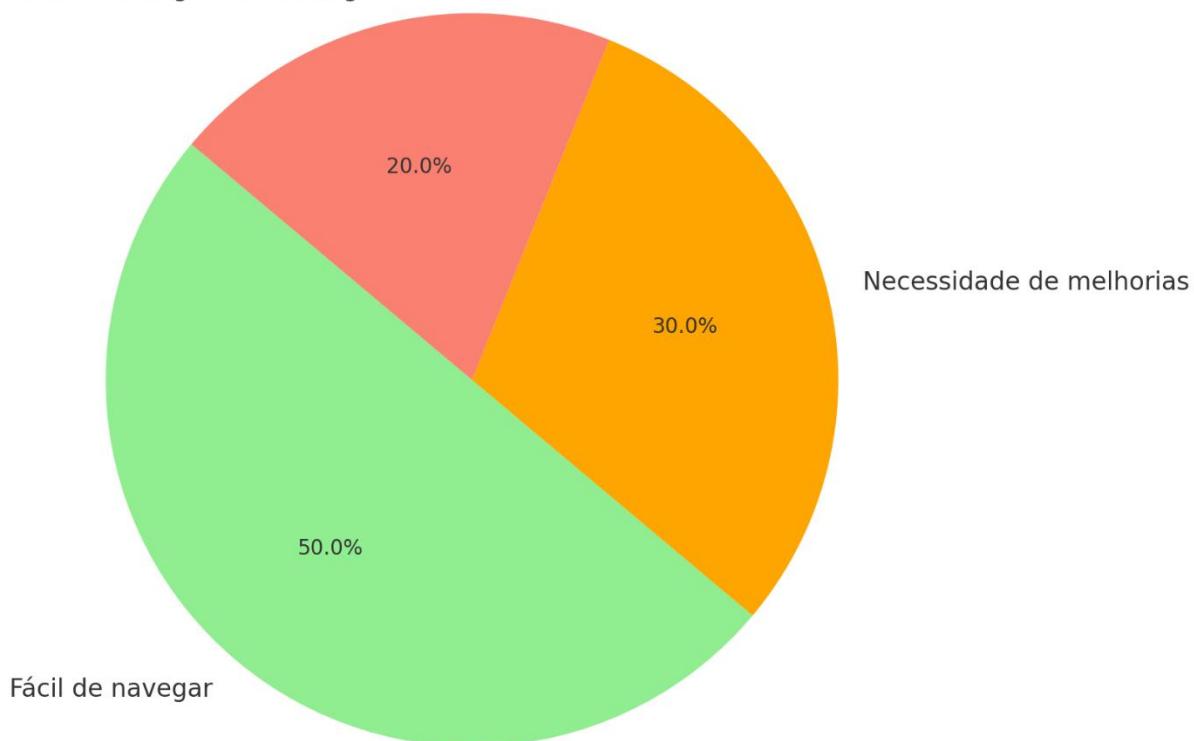
Essa falta de acesso representa um desafio na pesquisa acadêmica, limitando os alunos a buscar fontes externas para complementar seu conhecimento, como mencionado por aqueles que recorrem ao Google Acadêmico em vez de usar o acervo da própria universidade.

No Gráfico 2, destaca-se que a maioria dos estudantes considerou o site Guarani fácil de navegar, embora alguns tenham sugerido melhorias. Entre as sugestões estão ajustes no design e a organização de funcionalidades para melhorar a experiência de uso, como o redirecionamento de certas funções, como o cadastro de trabalhos, diretamente para a página inicial, tornando o acesso mais intuitivo.

Gráfico 2. Avaliação do Site Guarani.

Experiência com o Site Guarani

Problemas de carregamento/design

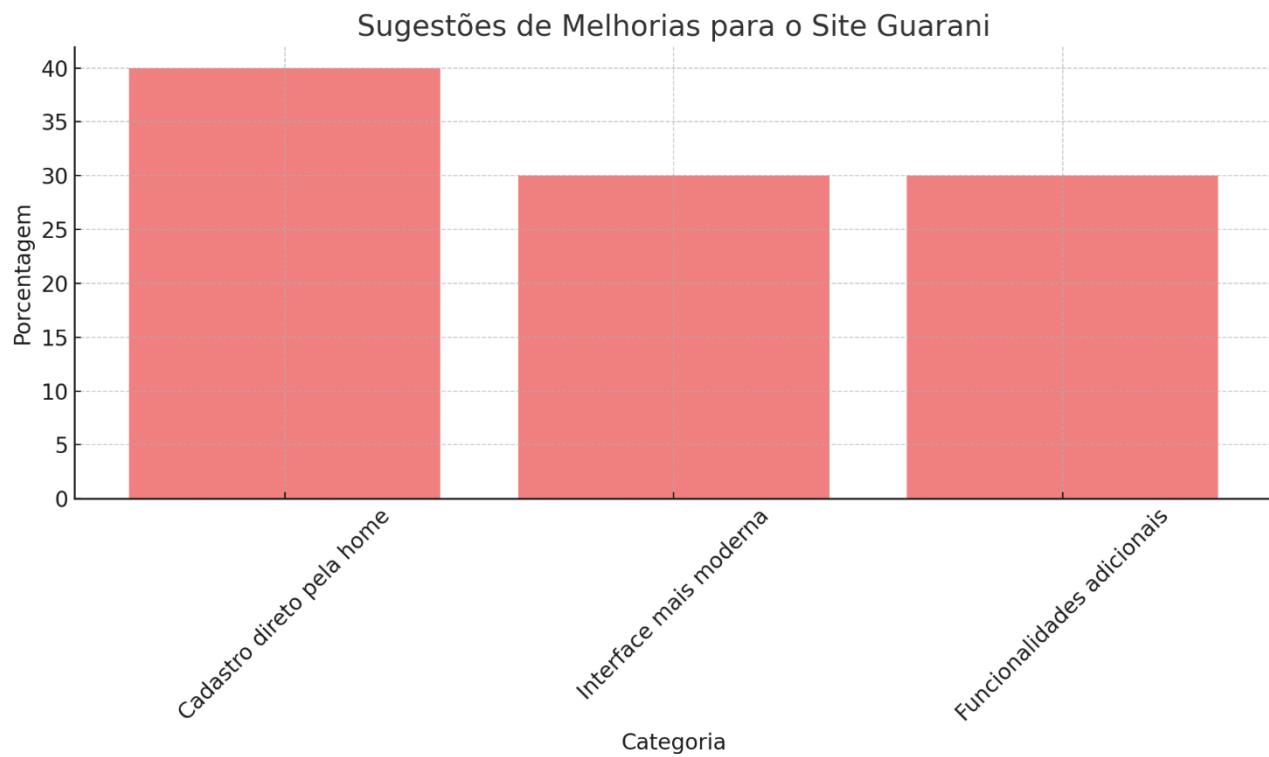


48

Fonte: Autoral, 2024.

No Gráfico 3, foram identificadas as principais sugestões de melhorias para o site Guarani. Entre as recomendações, destacam-se a inclusão de funcionalidades adicionais, aprimoramentos na estética visual e a simplificação do processo de autenticação. Essas sugestões indicam que, embora o Guarani ofereça uma boa base funcional, ainda há potencial para expandir suas funcionalidades e otimizar a experiência do usuário.

Gráfico 3. Sugestões de Melhoria para o Site Guarani.



Fonte: Autoral, 2024.

49

De modo geral, os resultados indicam uma recepção positiva do sistema GUARANI entre os estudantes, que destacaram a importância de uma plataforma centralizada e acessível para o armazenamento e consulta de trabalhos acadêmicos. Embora tenha sido bem avaliado em termos de naveabilidade, o *feedback* dos alunos sugere que aprimoramentos na interface e na funcionalidade do site podem torná-lo ainda mais eficiente e satisfatório. Essa análise qualitativa inicial revela que o GUARANI cumpre seu propósito de facilitar o acesso a produções acadêmicas e aponta caminhos para evoluções futuras, alinhadas às expectativas dos usuários.

5 DISCUSSÕES

O desenvolvimento do sistema GUARANI teve como base um estudo de caso específico, com o objetivo de criar uma solução prática e acessível para a centralização de trabalhos acadêmicos em uma instituição de ensino superior. Para isso, utilizamos metodologias como a criação de uma API RESTful para comunicação entre o front-end e o banco de dados, que foi implementado em MySQL. A escolha das tecnologias, como Node.js para o back-end e Bootstrap para o front-end, permitiu a construção de uma plataforma escalável e de fácil uso, alinhada com as demandas dos usuários.

Além da implementação técnica, aplicamos uma pesquisa exploratória de caráter qualitativo para avaliar a recepção da plataforma entre os estudantes. Entrevistamos 54 alunos dos cursos de Engenharia de Computação e Direito para identificar a eficácia do sistema e as dificuldades enfrentadas pelos usuários no acesso aos trabalhos acadêmicos antes da implantação do GUARANI. Os resultados mostraram que a maioria dos estudantes valoriza o acesso centralizado a esses materiais e demonstraram interesse em expandir o uso da plataforma, apesar de apontarem melhorias que poderiam ser feitas.

Ao comparar o GUARANI com outros sistemas abordados na revisão de literatura, como os estudos de Sousa (2023), Bezerra (2023) e Vrana (2010), percebe-se que o projeto apresenta um diferencial importante: ele foi projetado especificamente para atender às necessidades de uma instituição localizada no Maranhão. Enquanto os trabalhos anteriores abordam sistemas de arquivamento e gestão de dados acadêmicos de forma mais genérica, o GUARANI foi desenvolvido para lidar com as peculiaridades locais, o que incluiu o envolvimento direto dos usuários durante o desenvolvimento e a avaliação.

Apesar dos resultados promissores, o estudo possui algumas limitações. Primeiramente, a pesquisa qualitativa foi realizada com um número limitado de participantes (54 estudantes), o que pode não representar completamente a diversidade de usuários que a plataforma poderia alcançar. Além disso, a plataforma GUARANI ainda está em estágio inicial de desenvolvimento, o que significa que funcionalidades adicionais, como filtros mais avançados e integração com outras bases de dados, não foram implementadas nesta versão.

Futuras pesquisas podem explorar maneiras de expandir a adoção e o impacto do GUARANI, incluindo a possibilidade de integrar a plataforma com outras bases de dados acadêmicas, o desenvolvimento de funcionalidades adicionais que aprimorem a experiência do usuário e uma avaliação mais abrangente com diferentes grupos de usuários. Essas melhorias poderiam consolidar o GUARANI como uma ferramenta essencial no ambiente acadêmico, oferecendo um suporte ainda mais completo para o acesso e organização de trabalhos acadêmicos.

6 CONCLUSÃO

O Projeto GUARANI, desenvolvido neste trabalho, apresentou uma solução inovadora para centralizar, gerenciar e divulgar produções acadêmicas em uma instituição de ensino superior. A plataforma digital permitiu o cadastro e a consulta de monografias, artigos, dissertações e teses, promovendo maior visibilidade e acesso organizado às produções científicas da instituição. Com uma interface intuitiva e responsiva, desenvolvida em Bootstrap, e

funcionalidades robustas sustentadas por uma API em Node.js e banco de dados MySQL, o sistema foi projetado para atender às demandas da comunidade acadêmica de forma eficiente e segura.

A infraestrutura técnica do GUARANI destacou-se pela flexibilidade e compatibilidade com diferentes dispositivos, ampliando seu alcance e acessibilidade. Recursos como busca avançada com filtros e autenticação de usuários foram implementados para garantir uma experiência de uso prática e confiável. Os resultados obtidos reforçam o impacto positivo da plataforma na solução do problema de acesso limitado às produções acadêmicas, incentivando a continuidade das pesquisas dentro da instituição.

Como perspectivas futuras, propõe-se a implementação de funcionalidades adicionais para aprimorar a usabilidade e a segurança do sistema. Entre elas, destacam-se um sistema de notificações para informar os usuários sobre novos trabalhos cadastrados, um mecanismo de troca de mensagens para facilitar a comunicação acadêmica entre discentes e orientadores, e melhorias na segurança, como verificação por e-mail e autenticação em duas etapas. Essas evoluções visam consolidar o GUARANI como uma ferramenta indispensável para o avanço científico e tecnológico no ambiente acadêmico.

REFERÊNCIAS

51

1. BEZERRA, I. M. **O uso do Vue.js no desenvolvimento de uma plataforma web para a disponibilização de trabalhos científicos.** Criando Soluções Tecnológicas com a Engenharia de Computação. Editora Pascal, 2023.
2. BOOTSTRAP. **Responsive Web Development with Bootstrap.** 2021.
3. DATE, C.J. **Introdução a Sistemas de Bancos de Dados.** 8^a ed. Pearson, 2019.
4. ELOQUENT JAVASCRIPT. **A Modern Introduction to Programming.** 2022.
5. FIELDING, R. T. **Architectural Styles and the Design of Network-based Software Architectures.** University of California, 2000.
6. MARTINS, J. **Aplicações Práticas de MySQL em Sistemas Web.** Tech Publisher, 2018.
7. NIELSEN, J. **Usabilidade na Web.** 2021.
8. REDHAT, Editor. **O que é API REST?** RedHat, 2020. Disponível em: <https://www.redhat.com/pt-br/topics/api/what-is-a-rest-api>. Acesso em: nov-2024.
9. SILBERSCHATZ, A. **Sistemas de Banco de Dados.** 6^a ed. Pearson, 2020.



10. SOUSA, L. S. **Implementação de uma API REST usando o Spring para um sistema de divulgação acadêmica e tutoria. Criando Soluções Tecnológicas com a Engenharia de Computação.** Editora Pascal, 2023.
11. TILK, G. **Aprendendo Node.js: Desenvolvimento de Aplicações Web com JavaScript no Servidor.** Editora Code, 2019.
12. VRANA, R. **Digital Repositories and Possibilities of Their Integration into Higher Education.** International Journal of Emerging Technology in Learning, 2010.
13. W3C. **HTML5 & CSS3 Overview.** 2019.
14. YOUNG, B. **Express.js para Iniciantes.** Developer's Handbook, 2020.