

REPOSITÓRIO DIGITAL DE TRABALHOS DE CONCLUSÃO DE CURSO

Enzo Leis Musto Macedo, Renan Cordeiro Santo, Thiago Bitencourt Sancho

¹Colégio UNIVAP - Unidade Centro R. Paraibuna, 75 - Jardim São Dimas - 12245-020 - São José dos Campos-SP, Brasil, enzoleis12@gmail.com, thiagosancho2006@gmail.com, rcordeiro945@gmail.com.

Resumo

O projeto "Repositório Digital de Trabalhos de Conclusão de Cursos" tem como objetivo criar uma plataforma web para armazenar e preservar Trabalhos de Conclusão de Curso (TCCs) em diferentes formatos, como arquivos .ZIP e PDFs. Utilizando tecnologias como HTML, CSS, Vue.js, Node.js, Express e *MongoDB*, o sistema oferece uma interface intuitiva e reativa, além de garantir a integridade e acessibilidade dos documentos acadêmicos. O repositório aborda desafios relacionados à performance e escalabilidade, com soluções focadas na segurança dos dados e na busca eficiente. Futuras melhorias podem incluir a implementação de um *Progressive Web App* (PWA) e a integração com outras plataformas acadêmicas. O projeto contribui para a preservação do patrimônio acadêmico e promove o acesso ao conhecimento de forma inclusiva e acessível.

Palavras-chave: Repositório digital. Trabalhos de Conclusão de Curso. Preservação acadêmica. *MongoDB*. Vue.js

Curso: Ensino Médio, Técnico em Informática

Introdução

Em tempos em que a tecnologia se torna cada vez mais indispensável nas diversas áreas do conhecimento, a criação de soluções inovadoras para a preservação e disseminação da informação é essencial para o avanço acadêmico e científico (Gomez, 2015).

Repositórios digitais, que reúnem, preservam e disponibilizam conteúdo produzido pelas instituições, desempenham um papel crucial nesse cenário, oferecendo acesso confiável à informação científica (Queiroz, et al.).

Na área da Informática, a disseminação do conhecimento é fundamental para promover a inovação. Trabalhos de Conclusão de Curso (TCCs) são uma fonte valiosa de aprendizado e pesquisa, representando o resultado do esforço acadêmico dos alunos e contribuindo significativamente para o acervo intelectual das instituições. Com isso, torna-se imprescindível criar meios eficientes para armazenar e disponibilizar esses trabalhos, garantindo que o conhecimento gerado seja acessível e possa inspirar futuras gerações.

Diante desse contexto, o projeto "Repositório Digital de Trabalhos de Conclusão de Cursos" foi desenvolvido com o objetivo de criar uma aplicação web capaz de armazenar TCCs de diversos cursos. Esses trabalhos serão preservados em arquivos .ZIP, que conterão o código-fonte, documentação e monografias em formato PDF, garantindo a integridade e acessibilidade do acervo.

Para o desenvolvimento desta aplicação, foram utilizadas tecnologias modernas como HTML, CSS e o framework Vue.js. O banco de dados *MongoDB* foi escolhido por sua flexibilidade e capacidade de lidar com grandes volumes de dados, enquanto Node.js e Express foram empregados para garantir uma comunicação eficiente entre o *front-end* e o *back-end*.

Com o crescimento contínuo do repositório, desafios relacionados à performance e escalabilidade do sistema se tornam evidentes. Futuros desenvolvimentos poderão focar na implementação de ferramentas de monitoramento mais robustas para identificar gargalos de desempenho e na adoção de técnicas de escalabilidade horizontal, como o *sharding* em *MongoDB*, visando manter a eficiência do sistema à medida que o volume de dados aumenta.

Em suma, este projeto não só contribui para a preservação do patrimônio acadêmico, mas também exemplifica como as tecnologias digitais podem ser usadas para promover o conhecimento de forma inclusiva e acessível. A partir do "Repositório Digital de Trabalhos de Conclusão de Cursos", espera-se que outras instituições sigam o mesmo caminho, reconhecendo a importância de integrar novas tecnologias ao processo de ensino e aprendizado, preparando os cidadãos para os desafios do mundo contemporâneo.

Metodologia

A implementação do repositório digital visa melhorar o acesso aos TCCs, beneficiando futuros trabalhos dos alunos. Para garantir que as necessidades fossem atendidas, consultas com coordenadores de cursos foram realizadas. O *MongoDB* foi escolhido como banco de dados devido à sua flexibilidade e escalabilidade, permitindo gerenciar grandes volumes de dados heterogêneos de forma eficiente.

O processamento de dados foi realizado com *JavaScript*, utilizando *Node.js* e *Express* para gerenciar requisições ao banco de dados, garantindo uma interação fluida entre front-end e back-end. O desenvolvimento do sistema empregou *HTML*, *CSS* e o *framework Vue.js* para criar interfaces de usuário reativas e performáticas.

No que tange à armazenagem de dados, optou-se pelo uso do *MongoDB* como banco de dados principal, devido à sua natureza não relacional. Essa escolha se justifica pela sua capacidade de oferecer maior flexibilidade aos dados com requisitos variáveis (Kyle, 2011).

O *MongoDB* é amplamente reconhecido por sua escalabilidade e desempenho robusto, sendo uma escolha ideal para aplicações que demandam armazenamento eficiente de grandes volumes de dados heterogêneos (Chodorow, 2013).

Para facilitar a interação com o *MongoDB*, utilizamos o *Mongoose*, uma biblioteca que simplifica o uso do banco de dados, fornecendo um ambiente de modelagem de dados com esquemas e validação, o que facilita o desenvolvimento e a manutenção do código (Holmes, 2016).

O desenvolvimento do sistema segue o padrão de arquitetura MVC (*Model-View-Controller*), que organiza o código de maneira a separar as responsabilidades de manipulação de dados, interface de usuário e lógica de controle. Esse padrão proporciona uma estrutura modular, facilitando a manutenção e a escalabilidade da aplicação (Fowler, 2003).

A aplicação foi projetada para oferecer uma *RestAPI*, que permite a comunicação entre o frontend e o backend de maneira eficiente e segura. A *RestAPI* foi implementada seguindo os princípios *RESTful*, utilizando *HTTP* como protocolo de comunicação e garantindo a interoperabilidade entre diferentes sistemas (Richardson & Ruby, 2007).

Para garantir a segurança do sistema, foi utilizada autenticação baseada em *tokens*, onde o usuário, ao realizar o login, recebe um *token* de acesso que é utilizado para autorizar as ações subsequentes. Essa abordagem não apenas melhora a segurança, mas também facilita o controle de acesso e a implementação de funcionalidades como sessões expiráveis e autenticação em múltiplos dispositivos (Stallings, 2011).

Para funcionalidades relacionadas ao envio de e-mails, como recuperação de senha e notificações, foi utilizada a biblioteca *Nodemailer*, que permite a integração com diversos serviços de e-mail e oferece uma interface simples e eficaz para o envio de mensagens (Rehman, 2020).

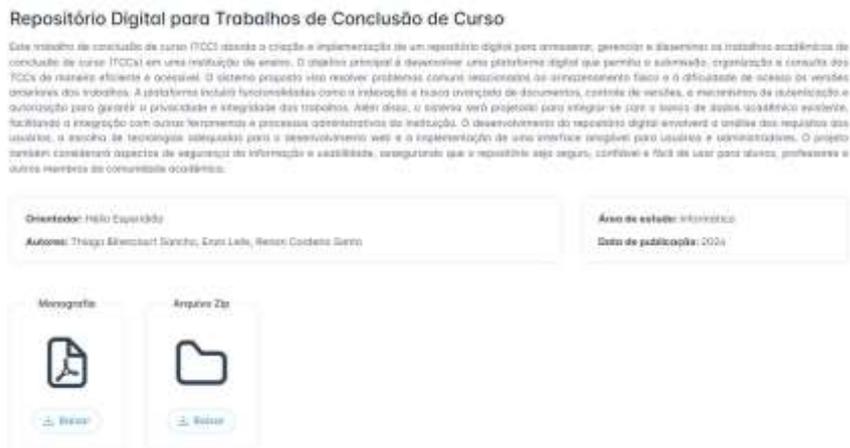
Resultados

O desenvolvimento do Repositório Digital de TCCs resultou em uma plataforma web funcional e acessível, que armazena TCCs em formato .ZIP e monografias em PDF, preservando o acervo acadêmico. A interface em *Vue.js* oferece navegação intuitiva e responsiva. *MongoDB* gerenciou grandes volumes de documentos com eficácia, enquanto *Node.js* e *Express* asseguraram uma comunicação fluida entre front-end e back-end.

A Figura 1 mostra a interface do sistema que exibe o resumo do trabalho, com autores, orientador, área de estudo, data de publicação e opções para download da monografia em PDF e do arquivo ZIP.



Figura 1 – Tela referente ao trabalho de conclusão de curso



Fonte: Os autores

O painel administrativo pode ser visualizado na figura 2, que faz o controle da aprovação e publicação dos TCCs definindo o tema, líder, orientador, curso, status, membros e ações de visualizar, editar e excluir.

Figura 2 – Painel administrativo de TCC's

Grupos	Tema	Líder	Orientador	Curso	Status	Membros	Ações
<input type="checkbox"/>	Repositório Digital para Trabalhos de Conclusão de Curso	Thiago Silveira de Sencho	Hélio Esperidião	Informática	Aprovado	>	  

Fonte: Os autores

Discussão

A criação de um sistema de busca eficaz foi um dos principais desafios do projeto, superado com o uso de metadados detalhados para uma busca precisa e filtragem eficiente (Keith, 2010). A segurança dos dados e a privacidade dos autores dos TCCs também foram prioridades. Medidas como criptografia e autenticação de usuários foram rompidas para proteger informações sensíveis, conforme recomendado por Chodorow (2013) e Goodman (2018). Esses aspectos foram essenciais para a confiabilidade do repositório, ressaltando a importância de práticas de segurança estruturadas (Severance, 2019).

Conclusão

O projeto "Repositório Digital de Trabalhos de Conclusão de Cursos" conseguiu atingir seu objetivo principal ao criar um sistema funcional e acessível, que não apenas preserva o acervo acadêmico da instituição, mas também o torna disponível para futuras pesquisas e projetos. Esse resultado reflete o sucesso do projeto em democratizar o acesso à produção científica, cumprindo o propósito de preservar e disseminar o conhecimento de forma inclusiva.

No entanto, há espaço para melhorias futuras que podem aumentar a eficiência, acessibilidade e escalabilidade do sistema. A implementação de um Progressive Web App (PWA) poderia melhorar a navegação em dispositivos móveis, oferecendo uma experiência mais intuitiva e acessível. Além disso, personalizações avançadas da interface, como a possibilidade de ajustar temas e tornar a experiência de usuário mais adaptável, podem contribuir para uma usabilidade aprimorada.

Outra melhoria significativa seria a adição de tutoriais interativos, que ajudariam novos usuários a se familiarizar mais rapidamente com o sistema, tornando a curva de aprendizado menos íngreme. Além disso, a integração do repositório com outras plataformas acadêmicas, como bases de dados científicas e sistemas de gestão de aprendizado (LMS), permitiria uma maior conectividade e visibilidade dos trabalhos armazenados, facilitando o acesso por um público mais amplo.

Para garantir a integridade dos arquivos no repositório, seria útil criar um sistema de verificação técnica e formal, além de implementar um processo de validação acadêmica com avaliação por banca e revisão

dos orientadores para assegurar a qualidade dos TCCs. Com o crescimento do repositório, a performance e escalabilidade serão desafios. Desenvolvimentos futuros podem focar em ferramentas de monitoramento para identificar gargalos e em técnicas de escalabilidade, como sharding em MongoDB, para manter a eficiência.

Em resumo, o projeto preserva o patrimônio acadêmico e exemplifica como tecnologias digitais podem tornar o conhecimento mais acessível, inspirando outras instituições a integrar novas tecnologias no ensino.

Agradecimentos

As nossas famílias e amigos pelo incentivo e apoio incondicional. À Fundação ValeParaibana de Ensino pela oportunidade e ao professor Hélio Esperidião pela orientação e confiança. Aos nossos companheiros, essenciais na superação dos desafios desta jornada.

Referências

Banqueiro, Kyle. *MongoDB em ação* .

Chodorow, Kristina. *MongoDB: O guia definitivo* .

Creswell, John W.Poth, Cheryl N. *Investigação qualitativa e desenho de pesquisa: escolhendo entre cinco abordagens* .

Holmes, Alex. *Node.js 8 do jeito certo* .

Keith, Jeremy. *Scripting DOM: Web design com JavaScript e o modelo de objeto de documento* .

Konk, Pedro Teixeira. *Node.js profissional: construindo software escalável baseado em JavaScript* .W Rauschmayer, Axel. *Falando JavaScript* .

Severance, Charles. *JavaScript: Desenvolvimento web para iniciantes*. Cria

Tilkov, Stefan; Vinoski, Steve. *Node.js: Usando JavaScript para construir programas de rede de alto desempenho* .

Zakas, Nicholas C. *JavaScript para desenvolvedores web* . Wiley Publishing.

Chodorow, Kristina. *MongoDB: o guia definitivo*. O'Rei

Goodman, Danny. *Bíblia JavaScript* . Wiley,

Keith, Jeremy. *Scripting DOM: Web design com JavaScript e o modelo de objeto de documento* .

Severance, Charles. *JavaScript: Desenvolvimento web para iniciantes*. Cr

Freecodecamp. As vantagens e desvantagens do JavaScript. Disponível em: <https://www.freecodecamp.org/news/the-advantages-and-disadvantages-of-javascript/> . Acesso em: 2024.

Tecnospeed. Devo usar o Node.js?. Disponível em: <https://blog.tecnospeed.com.br/devo-usar-node-js/#:~:text=com%20o%20Node.-,Por%20que%20e%20onde%20usar%20Node.,o%20c%C3%B3digo%20JavaScript%20da%20aplica%C3%A7%C3%A3o> . Acesso em: 2024.