



MINISTÉRIO DA EDUCAÇÃO  
INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DE RONDÔNIA  
CAMPUS VILHENA  
PÓS-GRADUAÇÃO LATO SENSU EM DESENVOLVIMENTO WEB

**DESENVOLVIMENTO DE UMA PLATAFORMA PARA O NIMPI**

**DEIVID FRANCIS DE OLIVEIRA  
EZIO LEMES  
ITAMAR ALVES RIBEIRO  
WANDREUS MÜHL DOURADO**

VILHENA

2025

## **DESENVOLVIMENTO DE UMA PLATAFORMA PARA O NIMPI**

**DEIVID FRANCIS DE OLIVEIRA  
EZIO LEMES  
ITAMAR ALVES RIBEIRO  
WANDREUS MÜHL DOURADO**

Produto apresentado ao Instituto Federal de Educação, Ciência e Tecnologia de Rondônia, como requisito avaliativo para conclusão do curso de Pós-graduação em Desenvolvimento WEB, sob a orientação do Professor Marco Antônio Augusto de Andrade.

Ficha catalográfica elaborada pelo Sistema Gerador de Ficha Catalográfica do IFRO.

Desenvolvimento de uma plataforma para o NIMPI / Deivid Francis de Oliveira... [ et al ] / Vilhena, 2025.

19f. : il.

Orientador(a): Prof. Me. Marco Antônio Augusto de Andrade

Trabalho de Conclusão de Curso (Pós-Graduação Lato Sensu em Desenvolvimento Web) – Instituto Federal de Educação, Ciência e Tecnologia de Rondônia - IFRO, Vilhena, 2025.

1. Lemes, Ezio 2. Ribeiro, Itamar Alves. 3. Dourado, Wandreus Mühl. I. Andrade, Marco Antonio Augusto de (orient.). II. Instituto Federal de Educação, Ciência e Tecnologia de Rondônia - IFRO. III. Título.

**Bibliotecário(a) Responsável:** Rosilene Maria do Couto Marques, CRB-11/321

## DESENVOLVIMENTO DE UMA PLATAFORMA PARA O NIMPI

**Abstract.** This article details the development project of an old newspaper search platform - NIMP designed by the Software Factory (FSLab) of the Federal Institute of Rondônia. This project unfolded during the postgraduate course in Web Development offered by IFRO - Campus Vilhena. The NIMP platform is divided as follows: Front-end, which makes up the visual part of the platform developed using React.js. Back-end, developed a Programming Interface (API) using express and connected to a document-oriented database mongodb. Infrastructure, creates docker images of the application for deployment. Third Party Services, communication with OCR service already available through FSLab.

**Resumo.** Este artigo discrimina o projeto de desenvolvimento de uma plataforma de buscas de jornais antigos - NIMP idealizada pela Fábrica de Software (FSLab) do Instituto Federal de Rondônia. O desdobramento deste projeto se deu durante a pós-graduação em Desenvolvimento Web ofertado pelo IFRO - Campus Vilhena. A plataforma NIMP está dividida da seguinte maneira: Front-end, qual compõe a parte visual da plataforma desenvolvida utilizando React.js. Back-end, desenvolvida uma Interface de Programação (API) utilizando express e conectado a uma base de dados orientada a documentos mongodb. Infraestrutura, cria imagens docker da aplicação para deploy. Serviços de Terceiros, comunicação com serviço de OCR já disponível pelo FSLab.

### 1. INTRODUÇÃO

É notável que os registros históricos sempre foram uma prática do ser humano, desde os dias atuais onde tudo pode ser registrado pela câmera de um celular até a milhares de anos atrás através de pinturas rústicas em paredes de cavernas. O registro histórico é praticado por inúmeros motivos, seja para registrar um momento especial em família, um caso inusitado que aconteceu na cidade ou para comprovar/documentar que certos compromissos foram honrados.

Com o avanço tecnológico ficou mais fácil e prático ter registros históricos, pois pode facilmente se ter um backup de segurança, facilmente ser disponibilizado através da internet e não se degrada com o tempo como os registros físicos.

A plataforma NIMPI tem como objetivo servir de ferramentas onde jornais antigos possam ser digitalizados e apresentados, desta maneira, qualquer pessoa poderá acessar de qualquer lugar e ter acesso a estes registros de forma prática, podendo, assim, perpetuar ainda mais aquele registro e ser alcançado por muito mais pessoas.

# 1 Cronograma

Para estabelecer e definirmos o cronograma do projeto, utilizamos um gráfico de Gantt, uma ferramenta que nos permitiu antever a implementação do código do projeto e avaliar o tempo necessário para o desenvolvimento do sistema. A representação visual correspondente está ilustrada na figura abaixo:

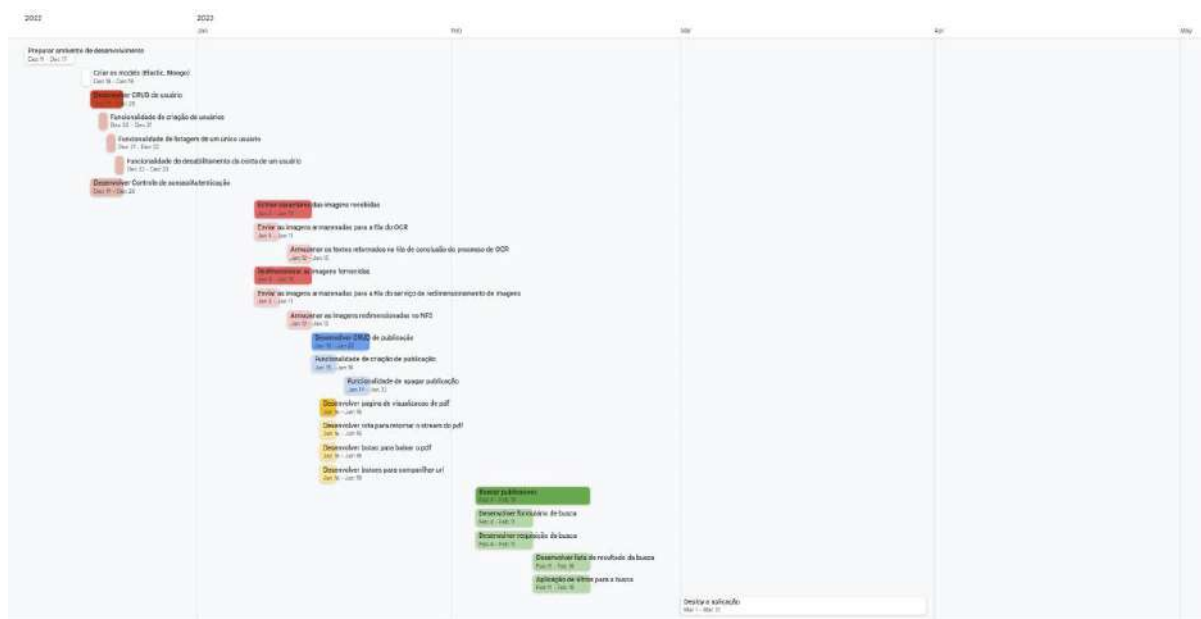


Figura 1. Cronograma do projeto

## 2 PRODUTO

### 2.1 Arquitetura da aplicação

A arquitetura da aplicação NIMPI, também conhecida como arquitetura de software, delinea a estrutura organizacional e funcional do sistema que suporta a plataforma. Ela define as interconexões entre componentes como interface do usuário, banco de dados e serviços, projetada para atender requisitos específicos do ambiente de consulta de documentos históricos. A abordagem C4 Model foi escolhida para garantir usabilidade, desempenho e segurança.

### 2.2 C4 Model

A abordagem C4 Model ajuda a entender de forma sistemática e clara a representação da arquitetura de software de forma escalável e compreensível. Essa abordagem, possibilita às equipes comunicarem de forma eficaz a estrutura e seus sistemas (Brown, 2018).

O Modelo C4 é dividido em 5 partes: Contexto, Contêineres, Componentes e Código. A seguir, vamos apresentar as três primeiras partes do Modelo C4 no contexto da aplicação MOOC.

### 2.2.1 Contexto

A camada de contexto (figura 2) oferece uma visão abrangente da plataforma NIMPI, identificando suas interações com atores externos, como usuários comuns e administradores. Nesta camada, também observamos a integração com o serviço OCR, responsável por transformar imagens em texto, assim como com o serviço de Processamento de Imagens, que trabalha com documentos em PDF, convertendo-os em imagens. Essa representação visual destaca as fronteiras e limites da plataforma, proporcionando uma visão geral do sistema.

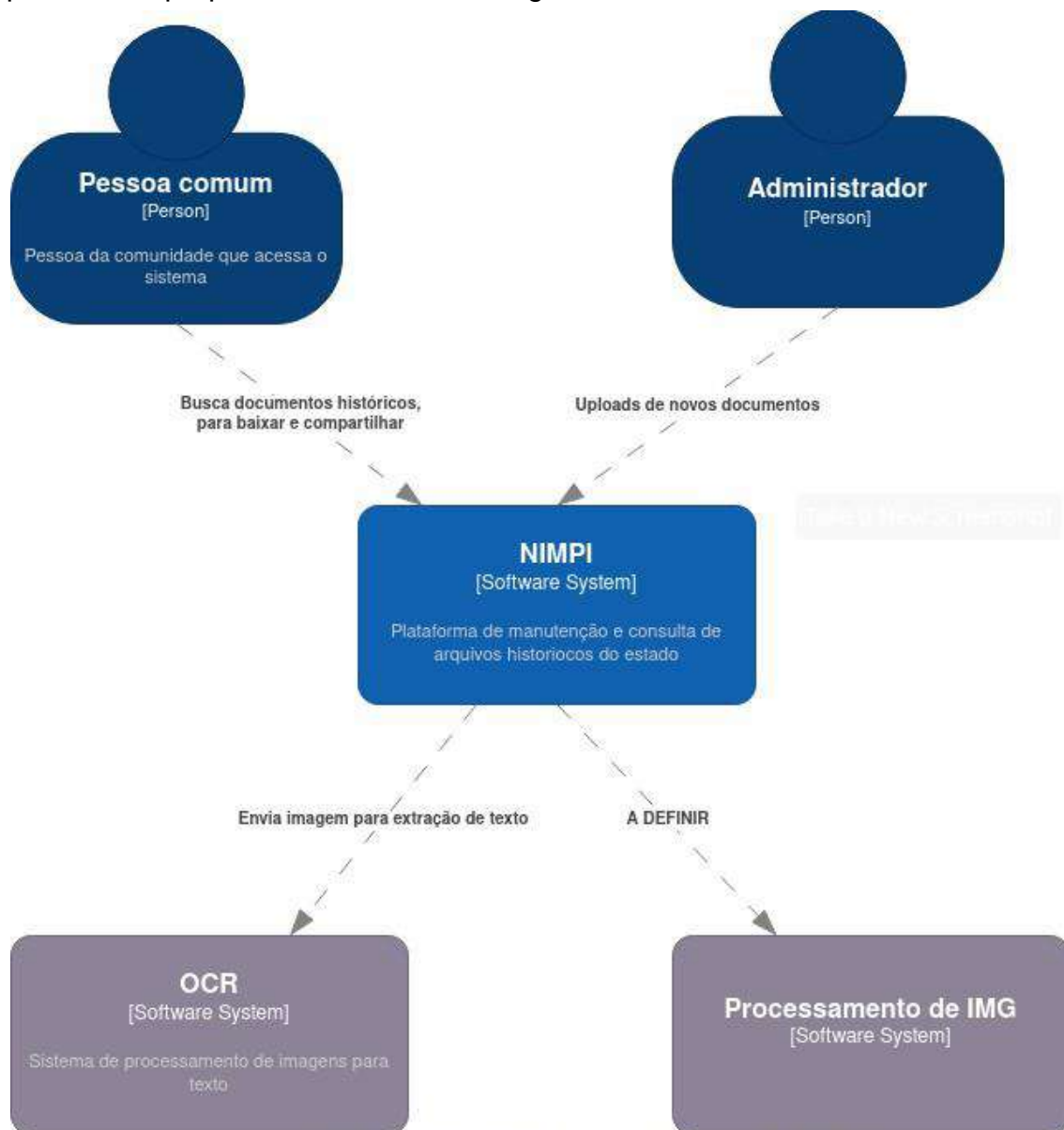


Figura 2. Exemplo Modelo C4 na visão de contexto

### 2.2.2 Contêineres

Na camada de Contêineres (Figura 3), os principais componentes da plataforma NIMP são identificados, englobando aplicativos web, bancos de dados, serviços de comunicação e módulos de processamento. A representação visual desta camada descreve de que maneira esses contêineres interagem entre si para proporcionar as funcionalidades essenciais da plataforma.

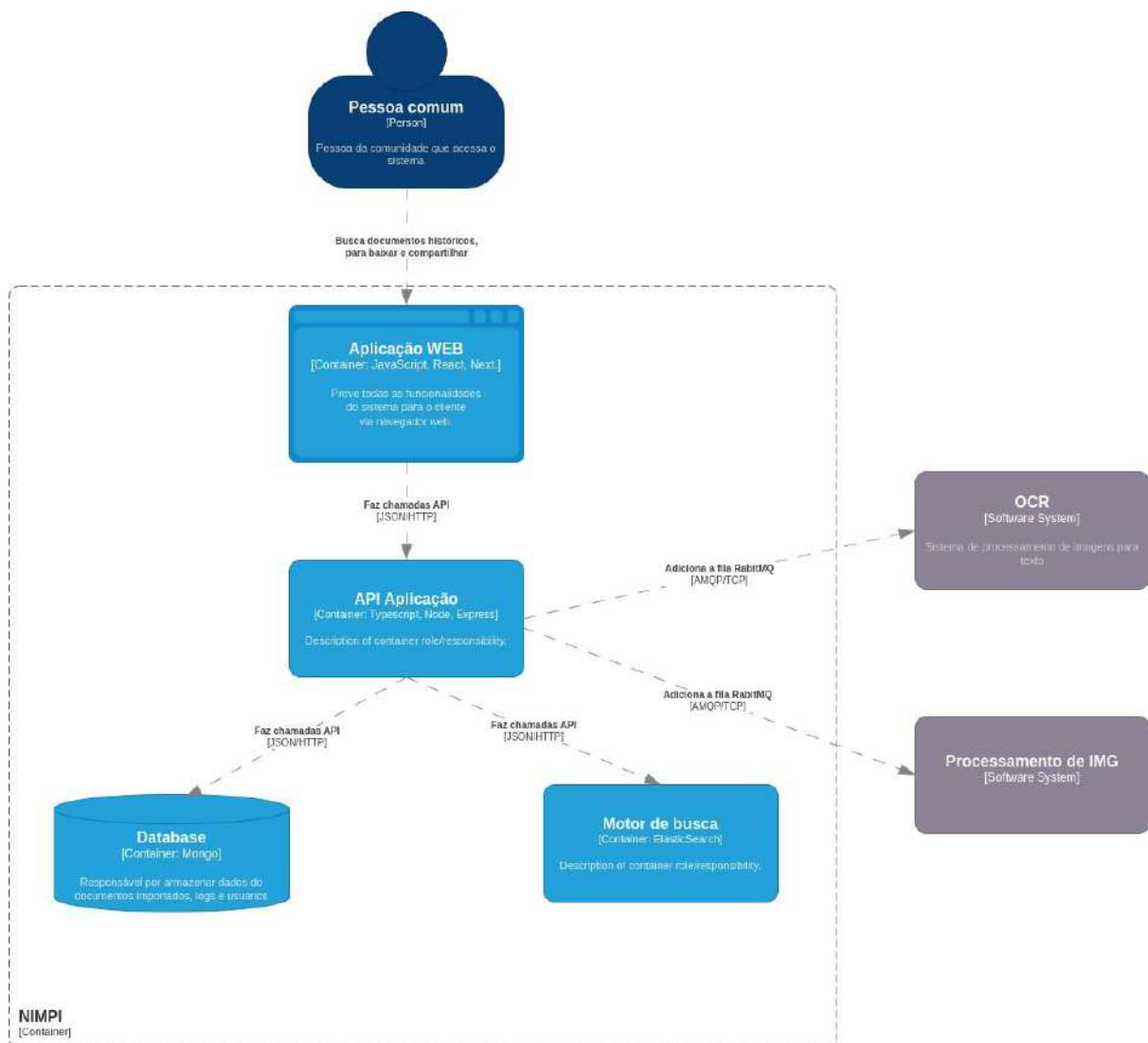


Figura 3. Exemplo Modelo C4 na visão de Contêineres

## 2.2.3 Componentes

Fase de "Componentes" (Figura 4) do Modelo C4, a arquitetura da aplicação NIMP é apresentada em detalhes mais aprofundados, oferecendo uma visão ampliada dos principais elementos funcionais que compõem cada contêiner previamente identificado. Nesse estágio, a ênfase recai sobre os módulos e componentes específicos que desempenham papéis essenciais na operação da plataforma de consulta de arquivos históricos.

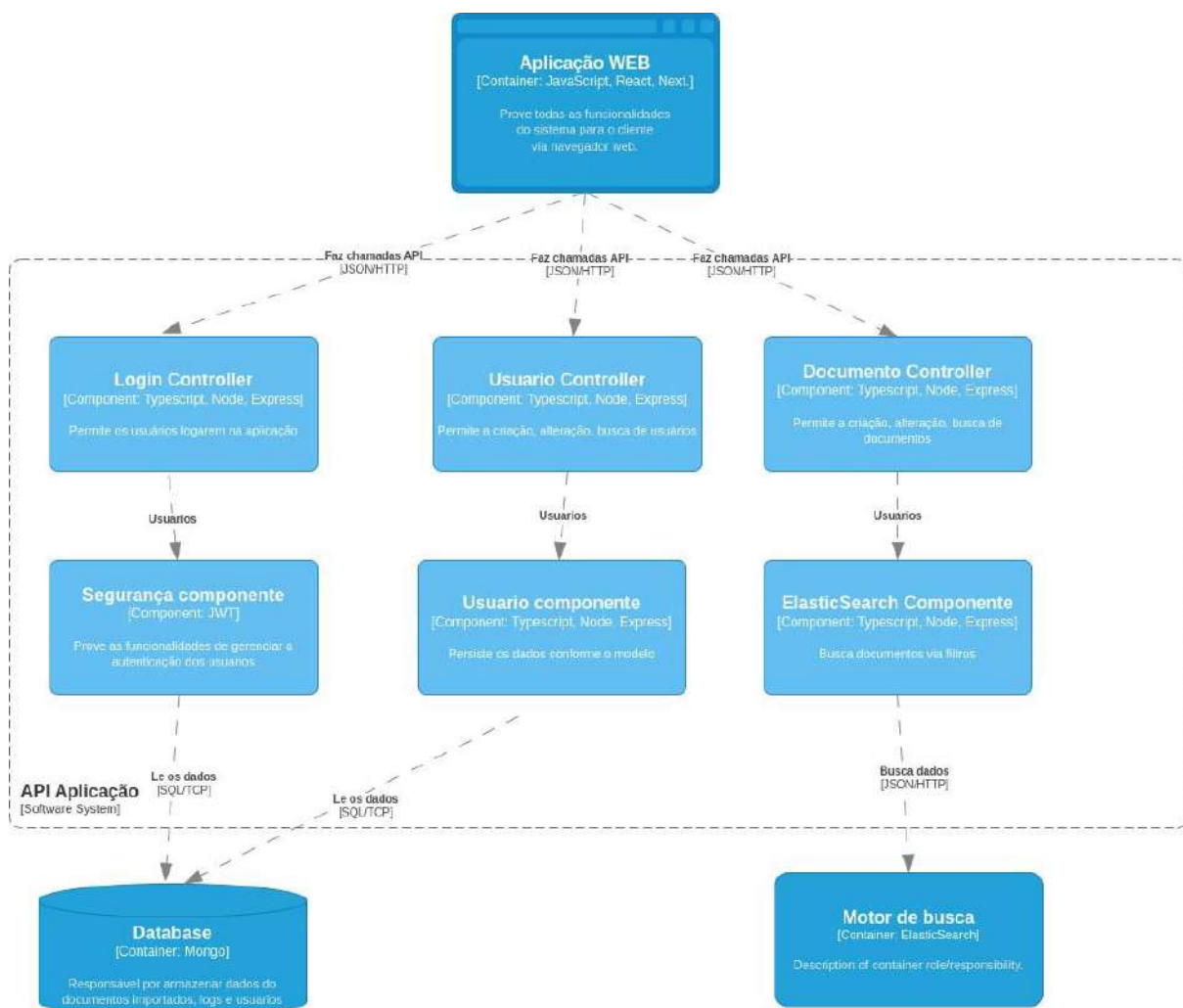


Figura 4. Exemplo Modelo C4 na visão de Componentes

## 2.3 Screenshots

Abaixo, você encontrará capturas de tela do Front-end que fornecem uma visão geral da experiência proporcionada pela plataforma de busca do NIMPI do IFRO (Instituto Federal de Educação, Ciência e Tecnologia de Rondônia).

A figura 5 demonstra a página inicial da aplicação, Onde usuário pode realizar a busca através do nome do jornal.



Figura 5. Página inicial da aplicação

A figura 6 apresenta a mesma tela com filtros avançados que permitem ao usuário refinar sua busca de maneira mais precisa. Estes filtros incluem opções como ano, fonte, ordem, título e conteúdo

Buscar no acervo digital

Ex: Marechal Rondon

Busca avançada

Ano	Fonte	Ordenar por
1983 <input type="button" value="▼"/>	Jornal Alta Madeira <input type="button" value="▼"/>	Fonte [crescente] <input type="button" value="▼"/>
<input checked="" type="checkbox"/> Títulos	<input checked="" type="checkbox"/> Conteúdo	



Figura 6. Página inicial da aplicação com filtros avançados

A figura 7 mostra a tela de resultados, onde os usuários podem visualizar uma prévia do conteúdo, incluindo a capa e o título dos itens encontrados. Além disso, eles têm a opção de realizar uma nova busca utilizando o mesmo formulário localizado no topo da página. Essa funcionalidade proporciona uma experiência contínua e intuitiva de pesquisa, permitindo aos usuários revisar rapidamente os resultados obtidos e realizar ajustes adicionais conforme necessário.

Ex: Marechal Rondon

Busca avançada ▾

25 resultados encontrados:

Thumbnail 1: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

Thumbnail 2: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

Thumbnail 3: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

Thumbnail 4: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

Thumbnail 5: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

Thumbnail 6: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

Thumbnail 7: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

Thumbnail 8: Jornal Alto Madeira - Edição de 1915 [Visualizar](#)

1 2 3 4

Figura 7. Página com os resultados

A Figura 8 apresenta a página de detalhes de um jornal, onde o PDF completo é renderizado para leitura do usuário. Nessa página, há opções para exibir metadados, fornecendo informações adicionais sobre o documento, e também a opção de baixar o arquivo PDF. Essa funcionalidade oferece aos usuários acesso conveniente ao conteúdo completo do jornal, juntamente com recursos adicionais para visualização e download.



**FSLab**  
Laboratório de Fábricas de Software

IFRO - Instituto Federal de Rondônia  
Av. Lauro Sodré, 6500 - Censipam - Aeroporto, Porto Velho - RO, 76805-260  
Fone/Fax: (69) 2182-9600

  
**INSTITUTO FEDERAL**  
 Rondônia  
 Campus  
 Anápolis

Figura 8. Página com os detalhes ao clicar

A Figura 9 exibe a página de envio de imagens para o acervo, mostrando a parte administrativa onde é possível fazer o upload de um novo jornal para o acervo utilizando a estratégia de "Drag & Drop" (arrastar e soltar). Essa interface simplificada permite que os administradores carreguem facilmente novos jornais no sistema, arrastando o arquivo desejado para a área designada, proporcionando uma experiência intuitiva e eficiente.

Enviar imagens para o acervo



Figura 9. Upload do arquivo do jornal para o acervo

A Figura 10 ilustra a página de pré-visualização de upload de jornal para o acervo, onde o administrador pode fornecer informações adicionais, como título e ano do jornal, para aprimorar os metadados antes do envio. Além disso, nesta

página, há a opção de remover o jornal, caso necessário. Essa funcionalidade permite que os administradores revisem e ajustem os detalhes do jornal antes de efetuar o upload, garantindo uma organização precisa e completa do acervo.

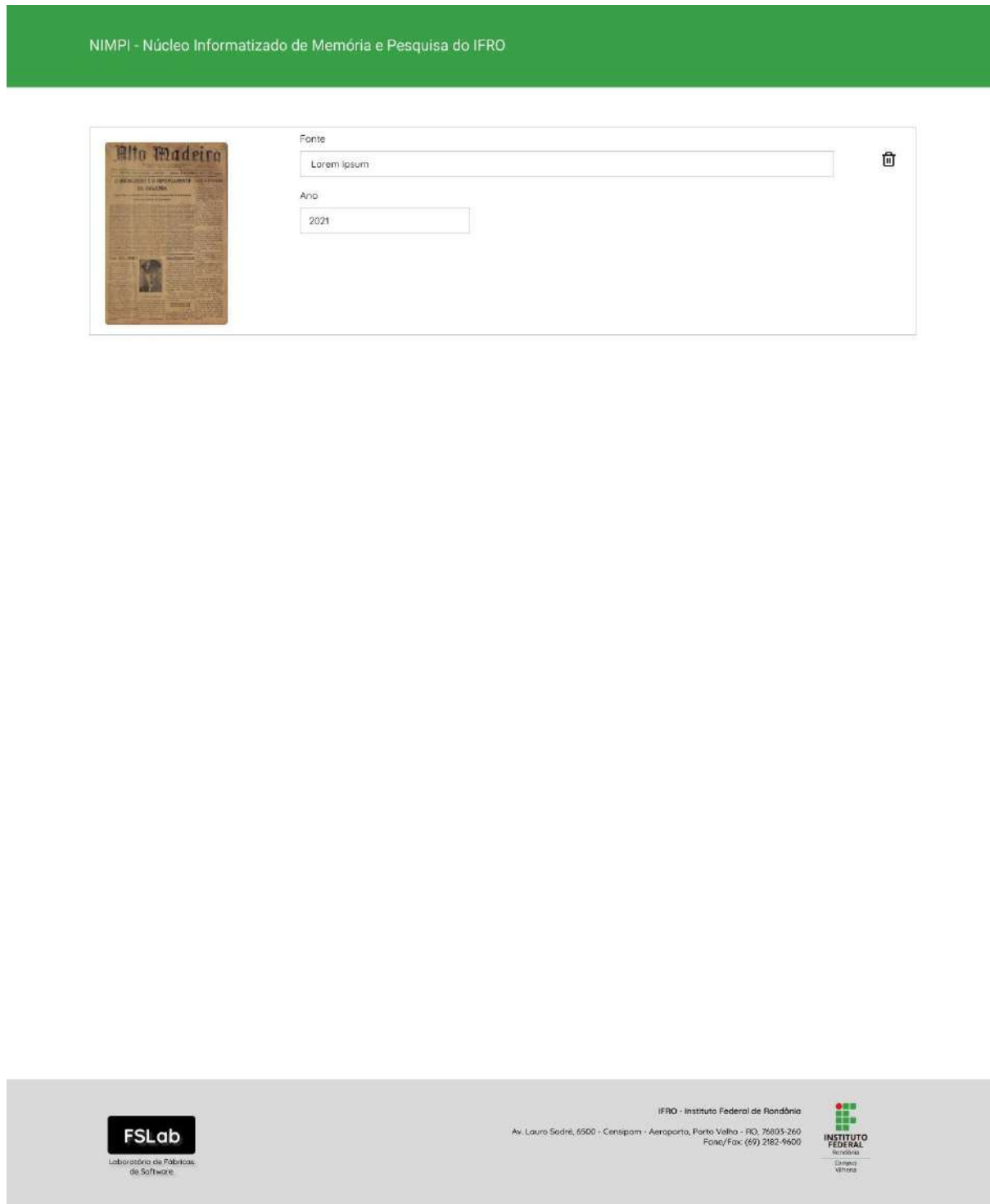


Figura 10. Página de pré visualização de upload

### 3 Documentação da API

A Figura 11 mostra uma lista de endpoints relacionados a jornais, incluindo operações para pesquisar, detalhar, cadastrar e enviar jornais para a fila de processamento de imagens.



Figura 11. Lista de endpoints de jornal

A Figura 12 exibe a documentação do endpoint para pesquisar um jornal por diversos parâmetros. Inclui detalhes sobre o método HTTP, o caminho do endpoint, os parâmetros de consulta aceitos e a estrutura da resposta esperada.

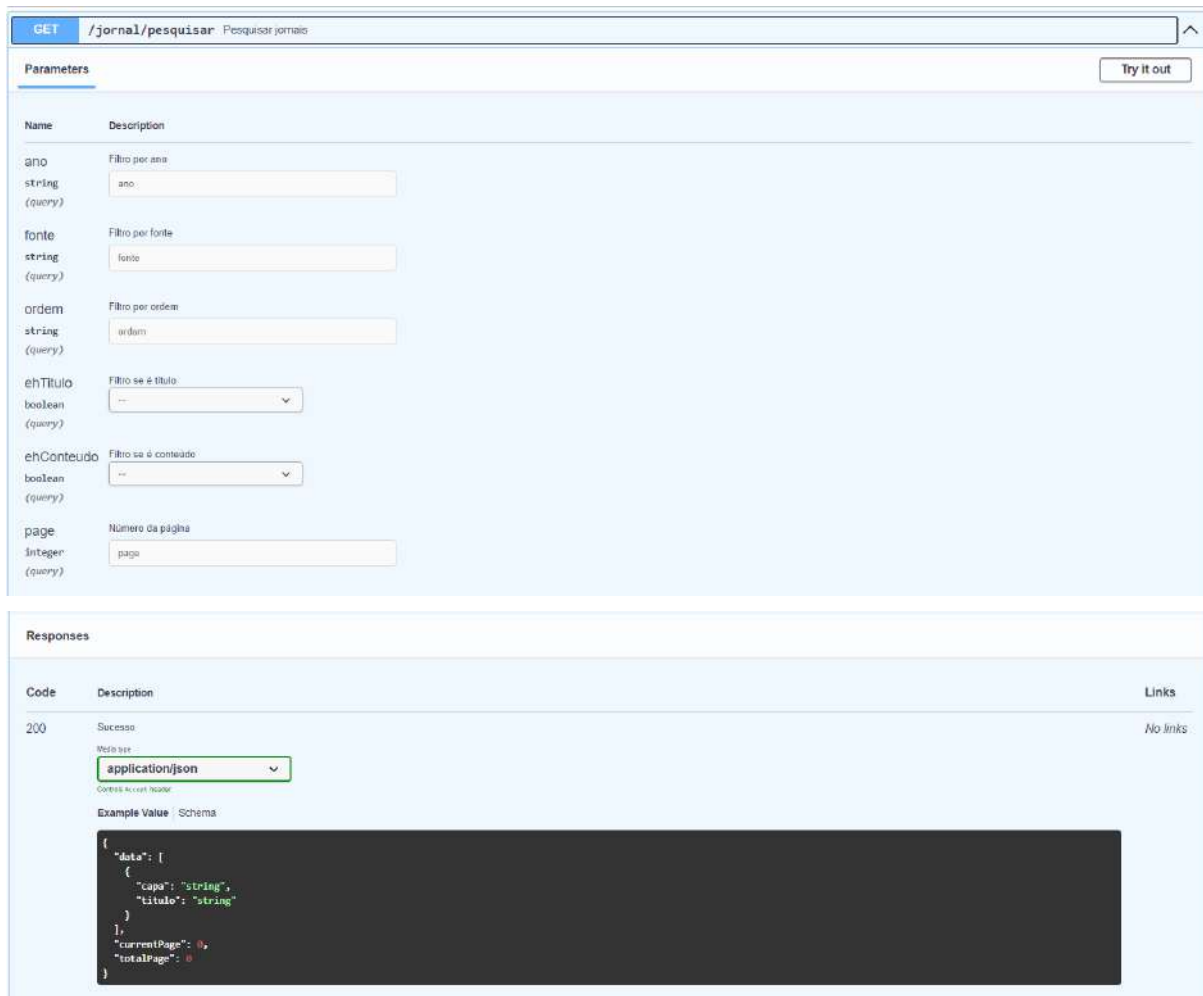


Figura 12. Endpoint de pesquisar jornal

A Figura 13 mostra a documentação do endpoint para obter detalhes de um jornal específico por id. Inclui informações sobre o método HTTP, o caminho do endpoint, os parâmetros necessários e a estrutura da resposta.

The screenshot displays the API documentation for the endpoint `GET /jornal/{id}`. The interface includes a "Parameters" section with a table listing a required parameter `id` of type `string` (path). Below this, the "Responses" section shows a `200` status code for "Sucesso" (Success). The response content type is set to `application/json`. An "Example Value" section shows the following JSON schema:

```
{
  "titulo": "string",
  "arquivo01": "string",
  "metadados": {
    "titulo": "string",
    "ano": "string",
    "fonte": "string"
  }
}
```

Figura 13. Endpoint detalhar jornal

A Figura 14 apresenta a documentação do endpoint para cadastrar um novo jornal. Detalha o método HTTP qual é um post, o caminho do endpoint, os parâmetros do corpo da requisição e a estrutura da resposta de sucesso.

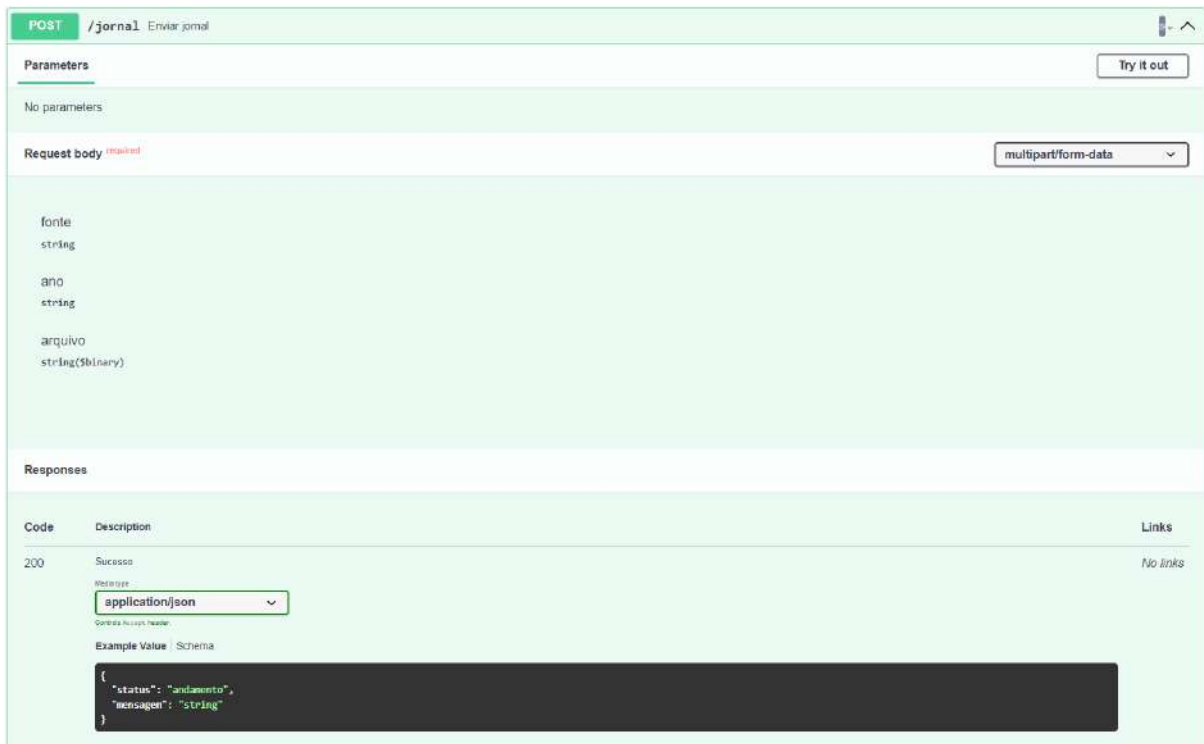


Figura 14. Endpoint de cadastrar jornal

A Figura 15 exibe a documentação do endpoint para enviar um jornal para a fila de processamento de imagens utilizando RabbitMQ. Mostra o método HTTP, o caminho do endpoint, os parâmetros do corpo da requisição e a resposta esperada.

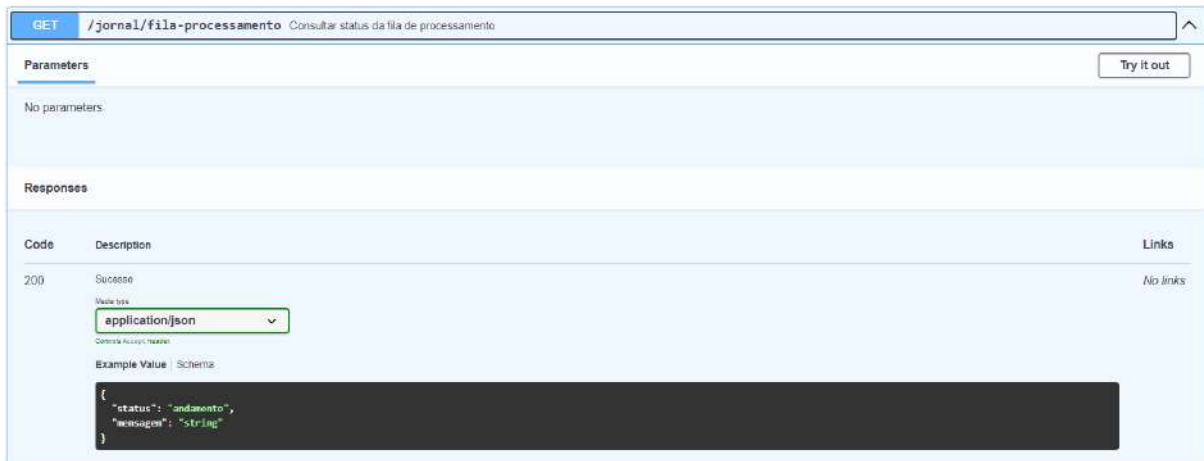


Figura 15. Endpoint da fila de processamento

A Figura 16 mostra os endpoints relacionados ao gerenciamento de contas de usuário. Inclui operações para criação, autenticação, atualização e exclusão de contas.

**Conta** Endpoints relacionados à gestão de contas de usuário ^






POST	/entrar	Entrar (Login)	  
POST	/recuperar-senha	Recuperar Senha	
POST	/criar-conta	Criar Conta	

Figura 16. Endpoints de gerenciamento de contas

## 4 Conclusão

O projeto que foi detalhado trouxe a oportunidade do grupo pôr em prática os conhecimentos adquiridos durante a jornada de aprendizado na pós-graduação de desenvolvimento Web, além de também conceder a oportunidade de poder trabalhar em equipe, onde o grupo demonstrou um ótimo engajamento e foco para cumprir cada etapa estabelecida sempre focando no resultado final. Cada integrante assumiu e cumpriu seu papel e objetivo no projeto seguindo normas e metodologias aplicadas.

A aplicação foi desenvolvida utilizando arquitetura, linguagem de programação e frameworks atuais, os quais proporcionam entregar uma solução de extrema qualidade e de fácil manutenção.

Em resumo, a equipe demonstrou estar preparada para o mercado de trabalho, sendo capaz de cumprir as demandas que os desafios que as empresas propõem.

## 5 Referências

SOUSA, Rosilene Paiva Marinho. **Reflexões Sobre a Digitalização de Documentos e Utilização de Documentos Digitais na Preservação da Memória Histórica**. Disponível em:

<[http://www.liber.ufpe.br/ctcm2011/anais/anais\\_ctcm/40\\_reflexoes\\_digitaliza.pdf](http://www.liber.ufpe.br/ctcm2011/anais/anais_ctcm/40_reflexoes_digitaliza.pdf)>.

Acesso: 15 set 2023.

AMOASEI, Juliana. **ORM com NodeJS: API com Sequelize e MySQL**. Disponível

em:<<https://www.alura.com.br/conteudo/orm-nodejs-api-sequelize-mysql#:~:text=O%20Sequelize%20abstrai%20os%20comandos,que%20est%C3%A1%20escrita%20em%20JavaScript.>>>. Acesso: 01 fev. 2024.

Brown, Simon. **O modelo C4 de documentação para Arquitetura de Software**

**2018**. Disponível em: <<https://www.infoq.com/br/articles/C4-architecture-model/>>.

Acesso em 12 dez 2023.