

# AlbumFlow: Solução Web para Organização e Seleção de Fotografias

Maria Alice F. Teixeira  
UFERSA

Pau dos Ferros, Brasil

maria.teixeira@alunos.ufersa.edu.br

Felipe H. Santos-da-Silva  
UFERSA

Pau dos Ferros, Brasil

felipe.silva65229@alunos.ufersa.edu.br

Leonardo Inácio Guilherme Dantas  
UFERSA

Pau dos Ferros, Brasil

leonardo.dantas69361@alunos.ufersa.edu.br

**Resumo**—O processo de seleção de fotografias com clientes pode ser demorado e descentralizado devido ao grande volume de imagens envolvido. Fotógrafos frequentemente utilizam múltiplos canais de comunicação, como e-mails e aplicativos de mensagens, resultando em ineficiências e atrasos. O AlbumFlow é uma plataforma web de código aberto desenvolvida para otimizar esse fluxo de trabalho, oferecendo um ambiente centralizado para gerenciamento de álbuns, compartilhamento de imagens e facilitação da seleção pelos clientes. Implementado com Django (Python) no backend e uma interface baseada em HTML, CSS, JavaScript e Bootstrap no frontend, o sistema aprimora a experiência de escolha ao disponibilizar uma interface intuitiva que otimiza a colaboração entre fotógrafos e seus clientes.

**Index Terms**—Gerenciamento de Fotografias, Seleção de Imagens, Organização, Plataforma Web, Django, Next.js, Sistemas de Fotografia.

## I. INTRODUÇÃO

O processo de seleção de fotografias junto ao cliente pode ser demorado e descentralizado devido à abundância de imagens disponíveis. Muitos profissionais precisam recorrer a diferentes meios, como mensagens, e-mails ou reuniões presenciais, para compartilhar, revisar e confirmar a escolha das imagens. Esse fluxo de trabalho fragmentado reduz a eficiência e aumenta o risco de erros ou atrasos na entrega do produto final.

O AlbumFlow surge como uma solução para otimizar esse processo, oferecendo uma plataforma centralizada onde fotógrafos podem gerenciar álbuns e compartilhar imagens de maneira intuitiva e organizada.

Diversos estudos já abordaram a seleção de fotos. [4] desenvolveram o *MagicPhotobook*, uma ferramenta que automatiza a escolha de imagens filtrando o *background* e integrando ferramentas de edição. [5] investigaram a seleção de imagens em álbuns pessoais, comparando critérios subjetivos com métricas automáticas. [8] propuseram um classificador treinado em um conjunto de dados anotados para automatizar a seleção de fotos familiares. Já [2] exploraram métodos para identificar automaticamente imagens importantes em grandes coleções. Diferente dessas abordagens, nossa proposta não visa a automação da seleção, mas sim o desenvolvimento de uma plataforma centralizada que facilite a curadoria manual de fotografias.

Este trabalho está estruturado da seguinte forma: na Seção II, apresentamos a fundamentação teórica do estudo. Na Seção

III, detalhamos a modelagem e implementação do sistema. Por fim, na Seção IV, discutimos os resultados obtidos e apontamos direções para trabalhos futuros.

## II. FUNDAMENTAÇÃO TEÓRICA

Nesta seção, apresentamos os principais conceitos e tecnologias utilizadas no desenvolvimento do AlbumFlow, destacando sua relevância no contexto da engenharia de software.

O desenvolvimento do AlbumFlow seguiu os princípios fundamentais da engenharia de software estabelecidos por [9], garantindo o uso eficiente dos recursos disponíveis. Essa abordagem possibilitou benefícios como definição clara do processo de desenvolvimento, documentação adequada e consideração de aspectos de manutenção e escalabilidade do sistema.

### A. Arquitetura do Sistema

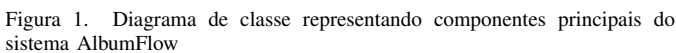
O AlbumFlow foi projetado com uma arquitetura de três camadas, garantindo modularidade, segurança e escalabilidade. A seguir, detalhamos os principais componentes tecnológicos utilizados:

- **Backend:** Utiliza Django Rest Framework (DRF) [3] para desenvolvimento de APIs REST, fornecendo suporte nativo para autenticação, serialização de dados e documentação automática das rotas.
- **Banco de Dados:** O SQLite3 [7] é adotado no ambiente de desenvolvimento por sua leveza e facilidade de configuração. Já o PostgreSQL [6] é utilizado na produção devido à robustez e suporte a operações complexas.
- **Frontend:** Desenvolvido com HTML, CSS e JavaScript, utilizando o framework Bootstrap [1] para responsividade e melhor experiência do usuário.
- **Métricas de Qualidade:** Aplicamos o SonarQube Community [10] para análise estática do código, garantindo padrões de qualidade, cobertura de testes e detecção de vulnerabilidades.

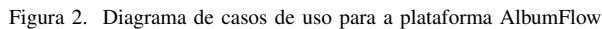
### B. Especificação Formal

A especificação formal dos requisitos foi realizada utilizando Z notation [11], permitindo uma definição rigorosa das restrições e funcionalidades do sistema.

O AlbumFlow é uma ferramenta web de código aberto (licença MIT), desenvolvida com Django REST Framework [3] no backend e HTML, CSS e JavaScript no frontend, utilizando Bootstrap para estilização.



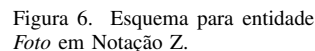
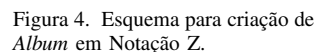
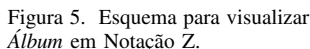
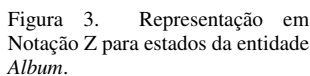
- **Álbum:** Representa a coleção de fotografias gerenciada pelos fotógrafos, contendo um identificador único, nome, link de acesso e metadados como datas de criação e atualização.
- **Foto:** Cada imagem pertence a um álbum específico e possui atributos como nome, URL, formato e tamanho.
- **Fotógrafo:** Está associado aos álbuns e fotos e herda atributos da interface Usuário, que define funcionalidades como login e gerenciamento de conta.
- **Tags:** Um álbum pode ter várias *tags* e uma *tag* pode estar relacionada a vários álbuns. Essa entidade permite classificar e filtrar as categorias das galerias a gosto do usuário.



sistema. O caso de uso central “Realizar Login” permite acesso às funcionalidades de gestão de álbuns e fotografias.

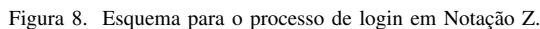
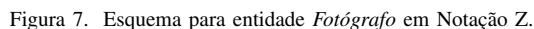
### Principais operações do sistema:

- **Gerenciamento de Álbuns:** Criar, editar, excluir e listar álbuns. A funcionalidade “Criar Álbum” permite adicionar fotos diretamente no momento da criação.
- **Gerenciamento de Usuário:** Possibilidade de visualizar e editar informações do fotógrafo.



As Fig. 3 e 4 ilustram esquemas para a entidade Álbum, representando seus principais atributos e o fluxo principal para a criação de um álbum no sistema.

A Fig. 5 representa o fluxo principal para visualizar as informações de um álbum no sistema. Os principais atributos da entidade Foto são ilustrados no esquema na Fig. 6



A Fig. 7 ilustra os atributos principais contidos no usuário fotógrafo, enquanto a Fig. 8 ilustra o fluxo principal para login de usuário.

#### IV. CONCLUSÃO E TRABALHOS FUTUROS

Neste trabalho, apresentamos a arquitetura e especificação das principais entidades do AlbumFlow, um sistema web para organização e seleção de fotografias. A plataforma foi projetada para otimizar o fluxo de trabalho dos fotógrafos, oferecendo um ambiente centralizado para a curadoria manual de imagens.

Para trabalhos futuros, pretendemos expandir o sistema incorporando funcionalidades voltadas para a experiência do cliente e do administrador, além de desenvolver uma versão funcional para validação em ambiente real.

#### REFERÊNCIAS

- [1] BOOTSTRAP. Bootstrap - The most popular HTML, CSS, and JS library, 2025. Accessed: 2025-03-26.
- [2] CERONI, A. *Personal Photo Management and Preservation*. Springer International Publishing, Cham, 2018, pp. 279–314.
- [3] CHRISTIE, T. Django rest framework, 2025. Acessado em: 24 mar. 2025.
- [4] GAO, Y., ATKINS, C. B., CHEATLE, P., XIAO, J., ZHANG, X., CHAO, H., WU, P., TRETTET, D., SLATTER, D., CARTER, A., PENNY, R., AND WILLIS, C. Magicphotobook: designer inspired, user perfected photo albums. In *Proceedings of the 17th ACM International Conference on Multimedia* (New York, NY, USA, 2009), MM '09, Association for Computing Machinery, p. 979–980.
- [5] KUZOVKIN, D., POULI, T., COZOT, R., LE MEUR, O., KERVEC, J., AND BOUATOUCH, K. Image selection in photo albums. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval* (New York, NY, USA, 2018), ICMR '18, Association for Computing Machinery, p. 397–404.
- [6] POSTGRESQL GLOBAL DEVELOPMENT GROUP. PostgreSQL Documentation, 2025. Accessed: 2025-03-26.
- [7] PYTHON DOCS. sqlite3 — DB-API 2.0 interface for SQLite databases, 2025. Accessed: 2025-03-26.
- [8] SHEN, S., YAMASAKI, T., SATO, M., AND KAJIWARA, K. Photo selection for family album using deep neural networks. In *Proceedings of the 2018 International Joint Workshop on Multimedia Artworks Analysis and Attractiveness Computing in Multimedia* (New York, NY, USA, 2018), MMArt&ACM'18, Association for Computing Machinery, p. 8–13.
- [9] SOMMERVILLE, I. *Engenharia de software*. Pearson Prentice Hall, 2011.
- [10] SONNARQUBE. SonarQube Community Build Documentation, 2025. Accessed: 2025-03-26.
- [11] SPIVEY, J. M. *The Z notation: a reference manual*. Prentice-Hall, Inc., USA, 1989.