

# Arquivo das *Newsletters* da U.Porto 2.0

Relatório Final do Projeto Integrador

Duarte Filipe Lopes  
Miguel Ângelo Nogueira  
Simão Queirós Rodrigues  
Tomás Agante Martins



Licenciatura em Engenharia Informática e Computação

**Tutor na U.Porto:** Prof. Pedro Brandão  
**Proponente:** Eng. Rodolfo Matos

Porto, 20 de Junho de 2024

# Conteúdo

<b>1</b>	<b>Introdução</b>	<b>2</b>
1.1	Enquadramento . . . . .	2
1.2	Objetivos e resultados esperados . . . . .	2
1.3	Estrutura do relatório . . . . .	3
<b>2</b>	<b>Metodologia utilizada e principais atividades desenvolvidas</b>	<b>4</b>
2.1	Metodologia utilizada . . . . .	4
2.2	Intervenientes, papéis e responsabilidades . . . . .	4
2.3	Atividades desenvolvidas . . . . .	5
<b>3</b>	<b>Desenvolvimento da Solução</b>	<b>7</b>
3.1	Requisitos . . . . .	7
3.2	Arquitetura e Tecnologias . . . . .	8
3.3	Solução Desenvolvida . . . . .	10
3.4	Validação . . . . .	12
<b>4</b>	<b>Conclusões</b>	<b>14</b>
4.1	Resultados alcançados . . . . .	14
4.2	Lições aprendidas . . . . .	15
4.3	Trabalho futuro . . . . .	15

# 1 Introdução

## 1.1 Enquadramento

O objetivo deste relatório é detalhar todo o trabalho realizado durante o projeto, incluindo o que foi planeado, o que foi executado e o que não foi possível concluir. Serão abordados o processo de trabalho, o funcionamento e as tecnicidades do sistema desenvolvido.

O projeto foi realizado para a UP Digital, cuja responsabilidade é assegurar às entidades constitutivas (EC) da U.Porto, unidades orgânicas e serviços autónomos, um serviço padrão no domínio das TIC. Isso inclui um catálogo de serviços e respetivo apoio técnico e consultoria, no âmbito do desenvolvimento, gestão e operação dos sistemas de informação, centros de dados, infraestruturas tecnológicas de redes de dados e voz, infraestruturas e serviços da U.Porto, e aplicações informáticas de suporte à atividade da Universidade. Por solicitação de uma EC, poderão ser contratualizados serviços adicionais, mediante acordo interorgânico, incluindo a alocação de recursos humanos sob coordenação da UP Digital. A UP Digital também pode rentabilizar os seus meios humanos e técnicos através de projetos de prestação de serviços, cujos encargos serão suportados pelas entidades solicitantes.

O problema específico abordado pelo projeto era a preservação do legado digital da Universidade do Porto, começando pelas newsletters publicadas por todas as entidades pertencentes à mesma. O objetivo futuro seria escalar o conceito de arquivador desenvolvido para incluir todas as páginas digitais da Universidade do Porto, arquivando snapshots temporários como a Wayback Machine.

## 1.2 Objetivos e resultados esperados

Os objetivos deste projeto eram:

- Tornar o catálogo existente ([mkt.up.pt/timeline/](http://mkt.up.pt/timeline/)) mais responsivo, seguindo abordagens semelhantes às do [arquivo.pt](http://arquivo.pt) e do Internet Archive, eliminando a necessidade de manter ligações às publicações originais.
- Melhorar a acessibilidade das newsletters recolhidas.
- Realizar a indexação e catalogação de todas as newsletters, de forma a que esses conteúdos sejam úteis no futuro.

Os resultados esperados incluíam:

- Sistema de arquivo com interface para interação, pesquisa e filtros de resultados.
- Integração com Apache Solr para otimização das buscas.
- Vista em linha temporal para melhor visualização histórica.
- Interface customizada, adaptada às necessidades dos utilizadores.

### 1.3 Estrutura do relatório

Este relatório está estruturado em 4 secções:

1. **Introdução:** Apresenta o objetivo do relatório, o contexto organizacional, o problema abordado e a motivação para o trabalho realizado, os objetivos e os resultados esperados, e a estrutura do relatório.
2. **Principais atividades desenvolvidas:** Descreve as atividades principais desenvolvidas durante o projeto, incluindo a metodologia utilizada, intervenientes, papéis e responsabilidades, e atividades desenvolvidas.
3. **Desenvolvimento da solução:** Explica o desenvolvimento da solução, incluindo os requisitos, arquitetura e tecnologias utilizadas, a solução desenvolvida e a sua validação.
4. **Conclusão:** Conclui o relatório e relata os resultados alcançados e as lições aprendidas.

## 2 Metodologia utilizada e principais atividades desenvolvidas

### 2.1 Metodologia utilizada

Adotamos uma abordagem de desenvolvimento iterativo com sprints semanais e reuniões de progresso. Esta metodologia permitiu-nos receber feedback contínuo e fazer os ajustes necessários à implementação do código à medida que o projeto evoluía. O projeto durou 16 semanas, com sprints de uma semana cada.

Além dessas reuniões, usamos ferramentas como GitHub para organização eficiente do código, alocação de tarefas e para que o nosso tutor e cliente avaliassem e comentassem o nosso progresso. Utilizamos também o Trello para a gestão de tarefas.

Para a comunicação da equipa utilizamos o Discord, que também serviu como plataforma para sessões de desenvolvimento de código em conjunto.

Adicionalmente, mantivemos encontros regulares com o nosso tutor e proponente da UP Digital. Essas reuniões proporcionaram oportunidades valiosas para obter perspetivas sobre as expectativas e percepções das mesmas, bem como para apresentar protótipos intermediários para feedback e validação.

### 2.2 Intervenientes, papéis e responsabilidades

Este projeto contou com o contributo de várias pessoas, todo ele essencial para o desenvolvimento e sucesso do projeto.

- Acompanhamento:
  - **Proponente:** Engenheiro Rodolfo Matos
  - **Tutor:** Professor Pedro Brandão

Responsáveis por supervisionar o projeto, fornecer orientação e feedback, garantindo que o projeto se alinhava com os objetivos esperados.

- Equipa do projeto:
  - **Duarte Filipe Campos Barbosa Lopes:** Desenvolvimento frontend e responsável pela interface de utilizador.

- **Miguel Ângelo Aguiar e Nogueira:** Desenvolvimento backend e gestão de infraestrutura.
- **Simão Queirós Rodrigues:** Integração com a API e otimização de busca.
- **Tomás Manuel de Moura Duarte Agante Martins:** Análise de dados e responsável pela indexação e catalogação das newsletters.

Todos os membros da equipa estiveram ativamente envolvidos na realização e desenvolvimento do projeto e contribuíram para o progresso contínuo e sucesso do projeto.

## 2.3 Atividades desenvolvidas

Ao longo do desenvolvimento do projeto, foram realizadas diversas atividades, incluindo planeamento do projeto, desenvolvimento de código, reuniões de equipa e reuniões com o cliente e tutor.

- Pesquisa inicial sobre a tecnologia a utilizar, a ferramenta [mkt.up.pt/timeline](https://mkt.up.pt/timeline), compreendendo as necessidades do projeto e identificando recursos e funcionalidades.
- Desenvolvimento de mockups de acordo com os requisitos esperados e com as especificações das componentes corretas.
- Implementação do código de acordo com os mockups e os requisitos anteriormente estudados, desenvolvimento de uma ferramenta de mirror e scrape das newsletters e criação da interface web para o nosso projeto.
- Apresentação do projeto ao tutor e proponente pontualmente, recolhendo feedback de pontos a melhorar. Discussão de perguntas e preocupações que surgiram, assim como dúvidas sobre as entregas.

O principal desafio encontrado foi a integração com Apache Solr, devido à complexidade da indexação e otimização de buscas.

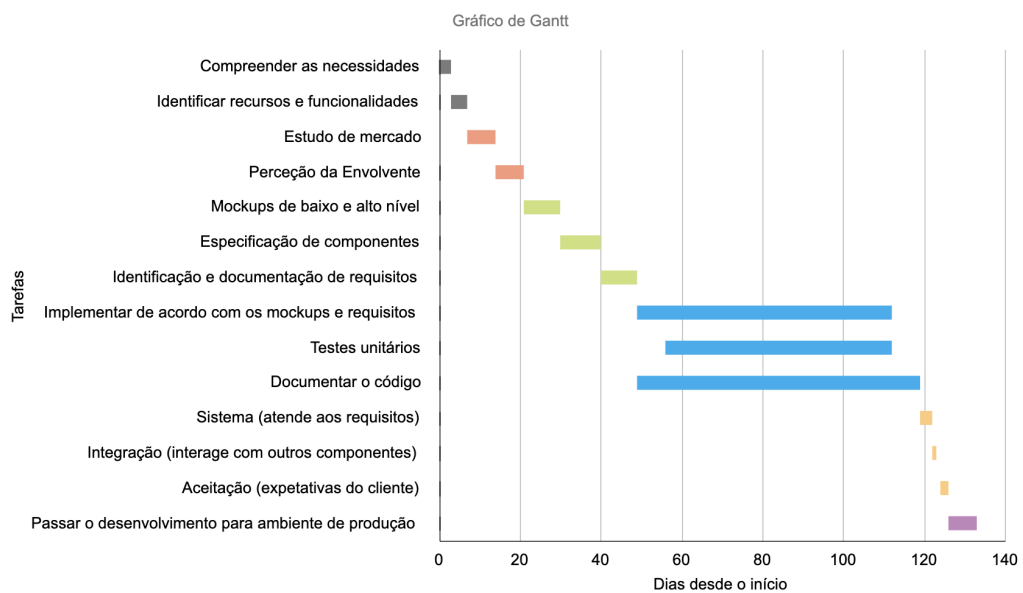


Figura 1: Fases do Projeto

## 3 Desenvolvimento da Solução

### 3.1 Requisitos

Os requisitos funcionais do projeto foram identificados com base nas necessidades dos clientes e nas especificações de mercado, visto que funcionalidades como pesquisa de conteúdo, organização de informação e filtragem de resultados são essenciais para sistemas de pesquisa de alta qualidade. Os principais requisitos funcionais incluem:

- **Pesquisa de Conteúdo:** Permitir que os utilizadores pesquisem dentro do conteúdo das newsletters.
- **Organização de Informação:** Catalogar e organizar a informação obtida das newsletters de forma estruturada.
- **Filtragem de Resultados:** Oferecer a capacidade de filtrar os resultados das pesquisas por diferentes critérios, tais como data, entidade publicadora e tipo de publicação.

Os requisitos não funcionais foram estabelecidos para garantir que o sistema não só cumprisse as suas funções principais, mas também fosse eficiente, seguro, fácil de usar e legalmente compatível. Entre os principais requisitos não funcionais, destacam-se:

- **Tempo de Resposta:** O sistema deve ser rápido e eficiente, proporcionando tempos de resposta baixos para as consultas dos utilizadores.
- **Escalabilidade:** A solução deve ser capaz de lidar com um volume crescente de dados e utilizadores.
- **Segurança:** Utilização de HTTPS para garantir a segurança na transmissão de dados.
- **Usabilidade:** A interface deve ser amigável e intuitiva, utilizando HTML semântico para melhorar a acessibilidade e garantindo um design responsivo que se adapte a diferentes dispositivos.
- **Conformidade Legal:** O sistema deve estar em conformidade com o RGPD, permitindo que os utilizadores solicitem a exclusão de dados pessoais que os mencionem.



Foram identificadas algumas restrições específicas que moldaram o desenvolvimento do projeto:

- **Tempo:** O projeto tinha um prazo limite de entrega até 16 de Junho de 2024.
- **Tecnológicas:** A única restrição tecnológica foi a necessidade de utilização do Bootstrap para a interface web.
- **Integração:** Não havia restrições de integração com outros sistemas.

### 3.2 Arquitetura e Tecnologias

A arquitetura do sistema foi organizada de forma modular, utilizando uma abordagem em camadas. Esta abordagem permitiu uma melhor gestão do desenvolvimento e facilitou a manutenção e a escalabilidade da solução. A arquitetura foi dividida nas seguintes camadas:

- **Mirroring:** Um script em Python utilizando `wget` foi desenvolvido para realizar o mirroring do Mautic da Universidade do Porto e de todas as suas newsletters.
- **Organização de Dados:** Após o mirroring, os dados são organizados e estruturados. O script processa as informações obtidas, organiza a estrutura das pastas e cria JSONs contendo resumos das newsletters, fazendo scraping das mesmas.
- **API:** Uma API foi desenvolvida em Flask para conectar os dados organizados ao sistema de pesquisa.
- **Interface Web:** A interface web foi desenvolvida utilizando Bootstrap, HTML, CSS e JavaScript.

As tecnologias utilizadas consistiram na linguagem Python e na stack web HTML/CSS/JavaScript. A escolha destas tecnologias baseou-se na sua facilidade de uso, rapidez de desenvolvimento e capacidade de atender aos requisitos do projeto de maneira eficiente.

Estava idealizado utilizar a plataforma de busca de código aberto Apache Solr, mas não conseguimos a sua inclusão na arquitetura do projeto dentro do prazo estabelecido. Apesar disso, com vista à escalabilidade do arquivo,

o Solr continua a fazer parte do plano de trabalhos, para a continuação do mesmo.

Os diagramas técnicos elaborados incluem:

- **Diagrama de Pipeline:** Este diagrama detalha o fluxo de processamento dos dados, desde o mirroring até a disponibilização dos mesmos na interface web.

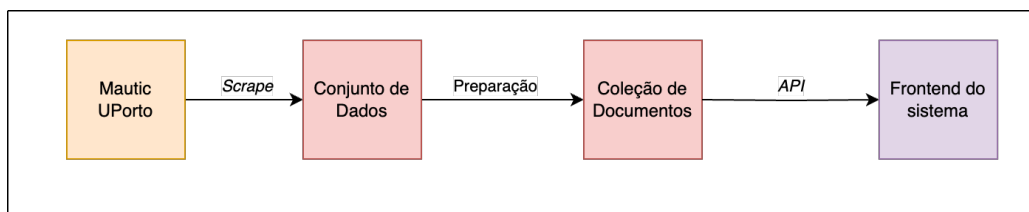


Figura 2: Diagrama de Pipeline

- **Diagrama de Estrutura de Documentos:** Este diagrama mostra a organização das pastas que armazenam as newsletters.

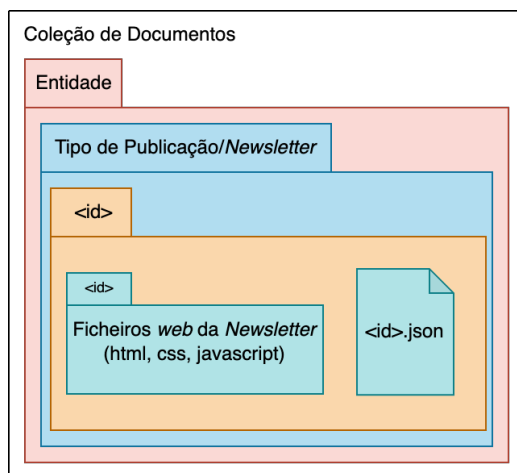


Figura 3: Coleção de Documentos

Durante o desenvolvimento, foram encontradas algumas dificuldades técnicas significativas, sendo a que mais se destacou:

- **Exibição de Newsletters:** Houve desafios na exibição correta das newsletters, devido a problemas com os diretórios carregados e à falta de imagens que não foram inicialmente descarregadas.
- **Resolução:** A solução para estes problemas envolveu a modificação do script que usa o `wget` para garantir o descarregamento correto das imagens e ajustes na API para assegurar a exibição completa e correta das newsletters.

### 3.3 Solução Desenvolvida

A solução desenvolvida consiste numa interface reimaginada, com a possibilidade de novos fluxos de interação. Agora, é possível pesquisar sobre o conteúdo das newsletters e aplicar diversos filtros sobre as mesmas. A interface do utilizador foi projetada para ser simples e acessível, com as seguintes características principais:

- **Página Inicial:** A página inicial exibe, em ordem decrescente de data de publicação, cards representando as newsletters. À medida que o utilizador rola a página, mais newsletters são carregadas dinamicamente.

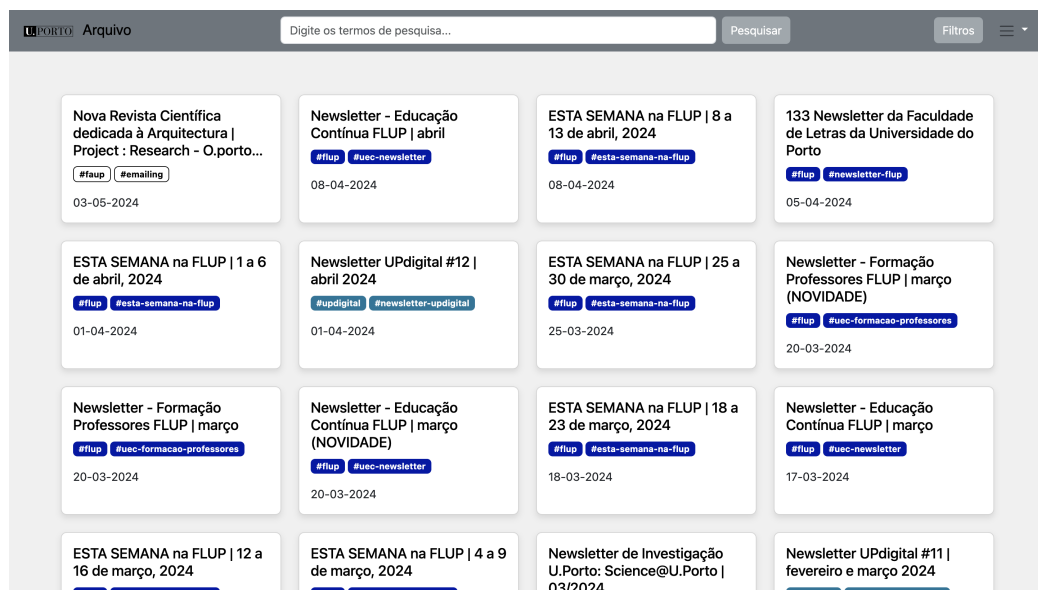


Figura 4: Página Inicial

- **Visualização de Newsletters:** Ao clicar em um card, o utilizador é redirecionado para uma página dedicada à respectiva newsletter, onde pode visualizar o seu conteúdo completo.

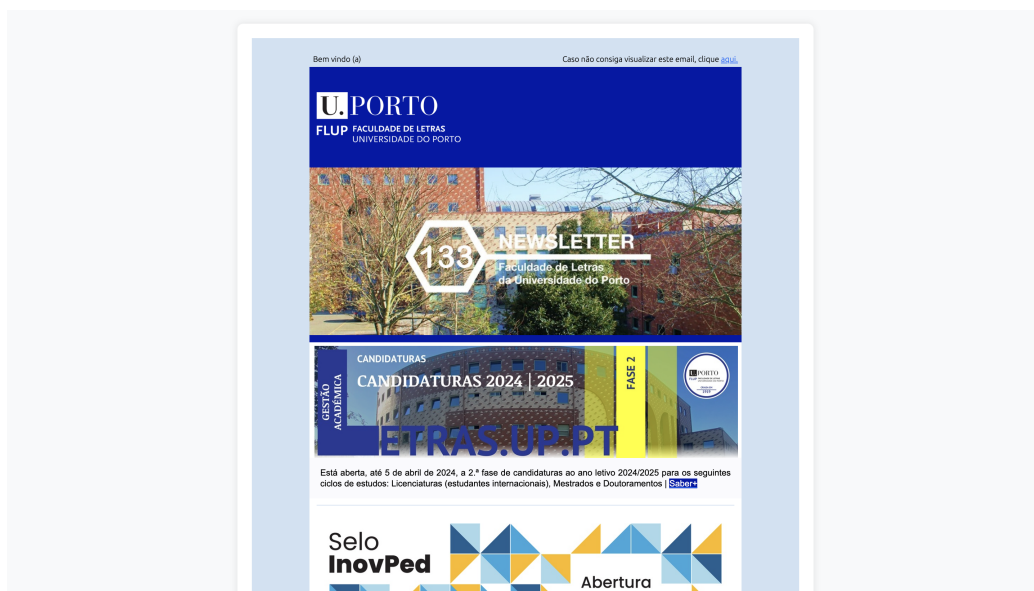


Figura 5: Visualização de uma Newsletter

- **Pesquisa e Filtragem:** A interface permite a pesquisa textual dentro das newsletters e oferece opções de filtragem por data de publicação, entidade publicadora e tipo de publicação.

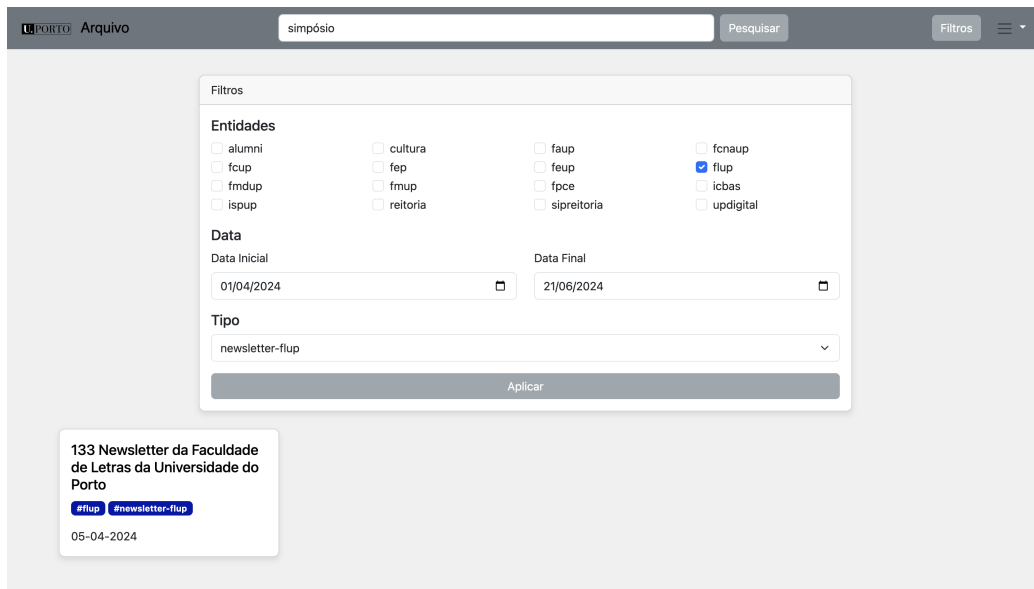


Figura 6: Pesquisa e Filtros

Os dados reais, obtidos através de mirroring e scraping (na parte de tratamento de informação), são utilizados para ilustrar as funcionalidades da aplicação.

### 3.4 Validação

Para validar a solução desenvolvida, foram realizadas várias ações de testes:

- **Testes de Integração Manuais:** Testes manuais foram conduzidos para verificar a correta integração dos diferentes componentes do sistema. Estes testes envolveram a verificação manual da obtenção e exibição dos dados, bem como o funcionamento correto das funcionalidades de pesquisa e filtragem.
- **Resultados dos Testes:** Os resultados dos testes foram positivos, confirmando que as newsletters são corretamente obtidas e exibidas, e que as funcionalidades de pesquisa e filtragem operam conforme o esperado.

O feedback do proponente do projeto foi coletado para avaliar a satisfação com a solução desenvolvida:

- **Feedback do Proponente:** O proponente do projeto expressou satisfação com a solução, destacando a importância do trabalho realizado e a funcionalidade do sistema.
- **Conclusões:** A conclusão a partir do feedback foi que, embora a solução já ofereça funcionalidades importantes e operacionais, ainda há espaço para melhorias e expansões futuras. O projeto foi considerado um sucesso inicial, com a possibilidade de desenvolvimento contínuo para aprimorar e adicionar novas funcionalidades.

## 4 Conclusões

### 4.1 Resultados alcançados

O projeto desenvolvido atingiu com sucesso os objetivos inicialmente estabelecidos, fornecendo uma solução funcional e eficiente para a pesquisa, organização e filtragem de newsletters. Os principais resultados alcançados incluem:

- **Desenvolvimento de uma ferramenta de mirroring e scraping:** Capaz de coletar dados do Mautic da Universidade do Porto e organizar as newsletters em uma estrutura lógica e acessível.
- **Implementação de uma API:** Utilizando Flask para conectar os dados organizados a um sistema de pesquisa.
- **Criação de uma interface web:** Utilizando Bootstrap, HTML, CSS e JavaScript, oferecendo uma experiência de usuário amigável e responsiva.
- **Funcionalidades de pesquisa e filtragem:** Permitem aos usuários localizar e acessar newsletters de forma eficiente e intuitiva.

Estas contribuições não só atendem aos requisitos funcionais e não funcionais do projeto, mas também superam as limitações da solução anterior, proporcionando uma melhoria significativa na gestão e acessibilidade das newsletters.

No caso de trabalho em grupo, as contribuições individuais foram as seguintes:

- **Duarte (25%):** Participou um bocado de todas as etapas do desenvolvimento, desde o mirroring até à implementação da interface web.
- **Miguel (25%):** Contribuiu um bocado em todas as áreas do projeto, ajudando na resolução de problemas e na integração das funcionalidades, sobretudo a parte do tratamento de informação em conjunto com o Duarte.
- **Simão (25%):** Envolvido em todos os aspectos do projeto, desde a concepção até a implementação, sobretudo interface web e API.

- **Tomás (25%):** Trabalhou em conjunto com os outros membros em todas as fases do projeto, garantindo a coesão e a qualidade do trabalho, sobretudo em conjunto com o Simão.

## 4.2 Lições aprendidas

Durante o desenvolvimento deste projeto, foram aprendidas várias lições importantes, alinhadas com os objetivos de aprendizagem:

- **Trabalho em Equipe:** A coordenação e a divisão de tarefas foram essenciais para o sucesso do projeto. Aprendemos a importância de uma comunicação eficaz e da colaboração entre os membros da equipa.
- **Gestão de Tempo:** Trabalhar com um prazo definido ensinou-nos a importância de planear e priorizar tarefas de forma eficiente.
- **Resolução de Problemas:** As dificuldades técnicas encontradas exigiram soluções criativas e uma abordagem sistemática para resolver problemas.
- **Conhecimentos Técnicos:** O uso de tecnologias como Python, Flask e Bootstrap aprimorou nossas habilidades técnicas e nos proporcionou uma experiência prática valiosa.

## 4.3 Trabalho futuro

Apesar dos sucessos alcançados, o projeto apresenta algumas limitações que podem ser abordadas em trabalhos futuros, sendo algumas delas:

- **Integração com Apache Solr:** A inclusão do Solr na arquitetura do projeto não foi concluída dentro do prazo estabelecido. A integração do Solr é uma melhoria prevista para aumentar a escalabilidade e eficiência do sistema de pesquisa.
- **Melhoria da Interface de Usuário:** Embora a interface seja funcional e amigável, há espaço para aprimoramentos no design e na usabilidade, proporcionando uma experiência de utilizador ainda melhor. Além disso, existia também o plano que não foi concebido, de ter uma vista em linha temporal, pelo que ficará para trabalho futuro.



- **Automatização de Testes:** Implementar um conjunto de testes automatizados para garantir a robustez e a confiabilidade do sistema, reduzindo a necessidade de testes manuais extensivos.
- **Expansão das Funcionalidades:** Adicionar novas funcionalidades, como a possibilidade de personalizar filtros de pesquisa e fornecer análises detalhadas das newsletters.

Em suma, o projeto atingiu os objetivos principais e forneceu uma base sólida para futuras melhorias e expansões, demonstrando a capacidade de aplicar conhecimentos técnicos em um contexto prático e colaborativo.

## Referências

- [1] Apache Solr. *Apache Solr Reference Guide*. Disponível em: <https://solr.apache.org/guide/solr/latest/index.html>.
- [2] Bootstrap. *Introduction to Bootstrap 5.3*. Disponível em: <https://getbootstrap.com/docs/5.3/getting-started/introduction/>.
- [3] Arquivo.pt. *Sobre o Arquivo.pt*. Disponível em: <https://sobre.arquivo.pt/pt/>.
- [4] Mozilla Developer Network. *JavaScript Reference*. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference>.
- [5] Flask. *Flask Documentation*. Disponível em: <https://flask.palletsprojects.com/en/3.0.x/>.
- [6] Universidade do Porto. *Mautic UPorto Timeline*. Disponível em: <https://mkt.up.pt/timeline/>.
- [7] Knight Lab. *Timeline JS*. Disponível em: <https://timeline.knightlab.com/#help>.
- [8] Mozilla Developer Network. *HTML Reference*. Disponível em: <https://developer.mozilla.org/en-US/docs/Web/HTML/Reference>.
- [9] Docker. *Docker Documentation*. Disponível em: <https://docs.docker.com/>.