

# IPBeja

INSTITUTO POLITÉCNICO  
DE BEJA

Escola Superior de Tecnologia e Gestão  
Licenciatura em Engenharia Informática

## Proposta de Projeto

*David Capa / 22606*  
*Samuel Marto / 23763*

Beja, 2024



**INSTITUTO POLITÉCNICO DE BEJA**  
**Escola Superior de Tecnologia e Gestão**  
**Licenciatura em Engenharia Informática**

## **Proposta de Projeto**

David Capa / 22606  
Samuel Marto / 23763

Orientado por :  
Professor Luís Carlos Bruno, IPBeja



# Conteúdo

<b>Conteúdo</b>	<b>i</b>
<b>1 Enquadramento</b>	<b>1</b>
<b>2 Atores do Sistema</b>	<b>3</b>
<b>3 Casos de Uso</b>	<b>5</b>
<b>4 Entidade relação</b>	<b>11</b>
<b>5 Storyboard</b>	<b>13</b>
5.1 Fluxo 1 — visualização do artigo . . . . .	13
5.2 Fluxo 2 — submeter artigo . . . . .	14
5.3 Fluxo 3 — analisar artigo . . . . .	14
<b>6 WireFrames</b>	<b>17</b>
6.1 Página inicial . . . . .	17
6.2 Página — Visualizar artigo . . . . .	17
6.3 Página — Submeter artigo . . . . .	18
6.4 Página — avaliar artigo . . . . .	19
6.5 Página — visualizar estatísticas do artigo . . . . .	20
6.6 Página — login e registo . . . . .	21
<b>7 Melhorias efetuadas na análise e no desenho do sistema</b>	<b>23</b>
<b>8 Implementação</b>	<b>25</b>
8.1 Arquitetura do sistema . . . . .	25
8.2 Arquitetura do sistema . . . . .	25
8.3 Tecnologias usadas . . . . .	26
8.4 Desenvolvimento da API . . . . .	26
8.4.1 Especificação da interface . . . . .	26
8.4.2 Decisões de implementação . . . . .	26
8.4.3 Principais casos relevantes de programação . . . . .	27

## CONTEÚDO

---

8.5	Desenvolvimento da App frontend/MVC . . . . .	27
8.5.1	Decisões de implementação . . . . .	27
8.5.2	Principais casos relevantes de programação . . . . .	27
<b>9</b>	<b>Conclusão</b>	<b>29</b>
	<b>Bibliografia</b>	<b>31</b>

# Capítulo 1

## Enquadramento

Plataforma para Publicação Científica irá permitir que researcher's publiquem os seus artigos; Rever as pesquisas uns dos outros e obter análises sobre o desempenho do seu trabalho (por exemplo, visualizações, citações, downloads). A plataforma terá autenticação de utilizadores para researcher's e reviewer's. O projeto poderá abranger várias áreas científicas, e a parte de análise de dados e focar-se-á em acompanhar o impacto do artigo.

A plataforma contará com um sistema para **rever artigos**, permitindo que os researcher submetam os seus artigos, estes revistos por especialistas na sua área, assegurando a qualidade e credibilidade do conteúdo publicado. Cada artigo passará por um processo de análise, onde o researcher poderá receber críticas. Estes podem assumir o papel de reviewer para análise de outras submissões, promovendo um sentido de comunidade e de conhecimento.

A **análise de dados** será uma funcionalidade essencial da plataforma, oferecendo insights detalhados sobre o desempenho e impacto do artigo. O researcher poderá acompanhar métricas como o número de visualizações e downloads, entre outros.

O sistema de **autenticação de utilizadores** distinguirá entre os diferentes papéis: o **researcher** terá a capacidade de submeter, rever e acompanhar as análises dos seus próprios artigos. O **reviewer** poderá fornecer *feedback* sobre artigos submetidos por outros researcher's, contribuindo para o processo de revisão por pares.





## Capítulo 2

# Atores do Sistema

O sistema poderá ser utilizado por 3 tipos de atores diferentes:

- **Guest (Convidado):** pode pesquisar, visualizar e fazer download dos artigos publicados.
- **Researcher/Reviewer (Pesquisador/Revisor):** pode publicar artigos, rever artigos de outros researcher's e visualizar dados analíticos dos artigos que submeteu.

## Persona

**Nome:** Dra. Sofia Oliveira

**Idade:** 35 anos

**Profissão:** Bióloga Investigadora

**Área de Interesse:** Botânica Espacial, Fisiologia Vegetal em Condições Extremas, Efeitos da Micro gravidade

**Experiência:** Doutorado em Biologia Molecular e cinco anos de experiência em estudos de biologia espacial numa agência de investigação europeia

**Hobbies:** Leitura de artigos científicos, participação em fóruns sobre ciência e tecnologia espacial, jardinagem, escrita de artigos para revistas científicas

### **Biografia:**

A Dra. Sofia é uma investigadora apaixonada pela biologia espacial e pela compreensão de como ambientes extremos, como a micro gravidade, afetam o desenvolvimento e crescimento de organismos vivos. Ao longo da sua carreira, tem-se focado principalmente em entender como as plantas reagem à ausência de gravidade e como esses conhecimentos podem apoiar a agricultura espacial e a sustentabilidade de colónias espaciais. Para o seu novo projeto, está concentrada nos efeitos da micro gravidade sobre o crescimento das raízes e o metabolismo das plantas, pesquisando artigos e publicações para consolidar a sua revisão bibliográfica e identificar lacunas que possa explorar na sua investigação.

### **Objetivos:**

- Aceder rapidamente a artigos científicos relevantes e atualizados para aprofundar o seu conhecimento sobre a fisiologia vegetal em micro gravidade.
- Fazer o download de artigos que possam ser usados na sua investigação.
- Contribuir com a sua própria investigação submetendo artigos para revisão e publicação, à medida que avança nas fases iniciais do seu projeto.
- Acompanhar as métricas de visualização e citações dos seus artigos, para compreender o impacto da sua investigação na comunidade científica.

### **Cenário: Dra. Sofia a pesquisar artigos sobre micro gravidade**

A Dra. Sofia precisa de aprofundar o seu conhecimento sobre os efeitos da micro gravidade nas plantas, principalmente na estrutura das raízes e na capacidade de absorção de nutrientes. Após um dia de trabalho no laboratório, acede à plataforma de artigos para verificar novas publicações e pesquisar artigos que possam complementar a base teórica do seu estudo.

1. Ao abrir a plataforma, a Dra. Sofia utiliza a funcionalidade "Pesquisar Artigos" para procurar por palavras-chave como "micro gravidade", "crescimento de raízes" e "fisiologia vegetal".
2. O sistema exibe uma lista de artigos que correspondem aos critérios inseridos, e a Dra. Sofia começa a navegar pelos títulos e resumos.
3. Identifica alguns artigos que parecem relevantes e clica para visualizar detalhes de cada um, focando-se nas metodologias utilizadas e nas conclusões apresentadas.
4. Satisfeita com os resultados, a Dra. Sofia decide fazer o download de três artigos específicos, utilizando a opção de descarregamento em formato PDF.
5. Com o download concluído, ela retorna à interface de pesquisa para continuar a navegar, anotando outros artigos que pretende explorar numa outra ocasião.

#### **Resultado esperado:**

A Dra. Sofia reúne artigos suficientes para a sua revisão bibliográfica e pode prosseguir com segurança nas fases iniciais do seu estudo sobre os efeitos da micro gravidade no crescimento das plantas.

## Capítulo 3

# Casos de Uso

O sistema deve suportar os seguintes casos de uso:

### UC1: pesquisar artigos (guest)

- **Ator:** guest
- **Descrição:** O guest pode aceder à plataforma e pesquisar artigos publicados.
- **Pré-condição:** O guest pode aceder à plataforma sem autenticação.
- **Fluxo principal:**
  1. O guest acede à interface de pesquisa de artigos.
  2. O sistema exibe um campo de pesquisa.
  3. O guest insere palavras-chave ou filtros de pesquisa (por exemplo, categoria, data).
  4. O sistema processa a pesquisa e exibe uma lista de artigos que correspondem aos critérios.
  5. O guest pode navegar pela lista de resultados.
  6. O guest seleciona um artigo da lista para visualizar os detalhes.
- **Pós-condição:** O guest visualiza a lista de artigos filtrados e pode selecionar um artigo para mais informações.

### Casos Secundários de Erro:

- **Erro 1: pesquisa sem resultados**
  1. Se o guest inserir palavras-chave que não correspondem a nenhum artigo, o sistema exibe uma mensagem indicando que não foram encontrados resultados.

2. O guest pode tentar uma nova pesquisa ou ajustar os filtros.

- **Erro 2: falha na conexão**

1. Se ocorrer um problema de rede durante a pesquisa, o sistema exibe uma mensagem de erro informando sobre a falha de conexão.

2. O guest pode tentar novamente mais tarde.

## UC2: visualizar e fazer download de artigos (guest)

- **Ator:** guest

- **Descrição:** O guest pode visualizar o conteúdo detalhado de um artigo selecionado e, se desejar, fazer o download do mesmo.

- **Pré-condição:** O guest deve aceder à lista de artigos ou a uma pesquisa prévia e selecionado um artigo disponível.

- **Fluxo Principal:**

1. O guest seleciona um artigo da lista exibida ou realiza uma pesquisa e escolhe um artigo.

2. O sistema exibe o conteúdo completo do artigo, incluindo título, autor, resumo e corpo do texto.

3. O guest tem a opção de visualizar informações adicionais, como referências e artigos relacionados.

4. O sistema disponibiliza uma opção de download na interface de visualização do artigo.

5. O guest clica na opção de download.

6. O sistema confirma que o artigo está disponível para download.

7. O sistema gera o arquivo do artigo em formato pdf ou outro formato apropriado.

8. O sistema inicia o download do arquivo para o dispositivo do guest.

- **Pós-condição:** O guest visualiza o conteúdo completo do artigo e, se optar por fazer o download, o artigo é descarregado com sucesso. O guest recebe uma notificação de conclusão do download.

### Casos secundários de erro:

- **Erro 1: artigo não encontrado**

- 
1. Se o guest selecionar um artigo que não está mais disponível (por exemplo, removido), o sistema exibe uma mensagem de erro.
  2. O guest é redirecionado de volta à lista de artigos.

- **Erro 2: falha no download**

1. Se ocorrer uma falha durante o download, o sistema exibe uma mensagem indicando que o download não foi concluído.
2. O guest pode tentar iniciar o download novamente.

### UC3: submeter artigo (researcher)

- **Ator:** researcher

- **Descrição:** O researcher pode submeter um novo artigo para publicação.

- **Pré-condição:** O researcher deve estar autenticado e ter o perfil de researcher. As informações do researcher já estão associadas ao perfil na conta.

- **Fluxo Principal:**

1. O researcher acede à página de submissão de artigos.
2. O sistema exibe as informações associadas ao perfil do researcher (nome, e-mail, etc.).
3. O researcher preenche as informações do artigo: título, resumo, Palavras-chave, links e anexo do artigo.
4. O sistema valida os dados inseridos.
5. O sistema armazena o artigo como "Submetido".

- **Pós-condição:** O artigo fica disponível para revisão por outros reviewers.

### Casos secundários de erro:

- **Erro 1: dados inválidos**

1. Se o researcher inserir dados inválidos (por exemplo, título em branco), o sistema exibe uma mensagem de erro.
2. O researcher deve corrigir os dados antes de prosseguir com a submissão.

- **Erro 2: falha na submissão**

1. Se ocorrer um erro durante o processo de submissão (por exemplo, falha na base de dados), o sistema informa ao researcher sobre a falha.
2. O researcher pode tentar submeter o artigo novamente.

#### UC4: rever artigo (reviewer)

- **Ator:** reviewer
- **Descrição:** O reviewer pode rever os artigos submetidos por outros pesquisadores.
- **Pré-condição:** O utilizador deve estar autenticado e ter o perfil de researcher/reviewer. O artigo deve estar no estado "Submetido".
- **Fluxo Principal:**
  1. O reviewer acede à lista de artigos submetidos.
  2. Seleciona um artigo para rever.
  3. O reviewer dá feedback e aprova, rejeita ou solicita modificações.
  4. O sistema atualiza o estado do artigo com base na revisão.
- **Pós-condição:** O artigo é atualizado para o estado "Aprovado", "Rejeitado" ou "Em Revisão".

#### Casos Secundários de Erro:

- **Erro 1: artigo Não Encontrado**
  1. Se o reviewer tentar aceder a um artigo que não existe mais, o sistema exibe uma mensagem de erro.
  2. O reviewer é redirecionado para a lista de artigos disponíveis.
- **Erro 2: falha na atualização do estado**
  1. Se ocorrer uma falha ao tentar atualizar o estado do artigo, o sistema informa o reviewer sobre a falha.
  2. O reviewer pode tentar rever o artigo novamente mais tarde.

#### UC5: visualizar estatísticas de artigos (researcher)

- **Ator:** researcher
- **Descrição:** O researcher pode visualizar as estatísticas de visualizações e downloads dos artigos que submeteu.
- **Pré-condição:** O utilizador deve estar autenticado como researcher e ter pelo menos um artigo publicado.
- **Fluxo Principal:**
  1. O researcher acede à secção de estatísticas.

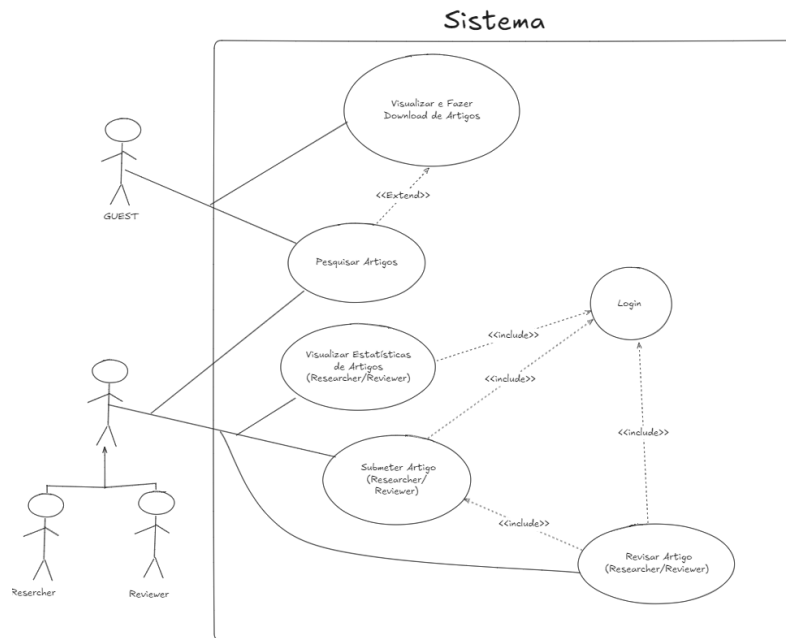
- 
2. Seleciona um artigo publicado.
  3. O sistema exibe as estatísticas: número de visualizações, downloads, tempo médio de leitura, etc.
- **Pós-condição:** As estatísticas são exibidas para o utilizador, permitindo que ele acompanhe o desempenho dos seus artigos.

#### Casos secundários de erro:

- **Erro 1: artigo sem estatísticas**
  1. Se o researcher selecionar um artigo que não possui estatísticas (por exemplo, um artigo recém-publicado), o sistema exibe uma mensagem informando que não há dados disponíveis.
- **Erro 2: falha na recolha de dados**
  1. Se ocorrer um erro ao tentar obter estatísticas (por exemplo, falha no servidor), o sistema informa o researcher sobre a falha.
  2. O researcher pode tentar visualizar as estatísticas mais tarde.

#### UML dos casos de uso

A figura 3.1 representa o sistema através de UML



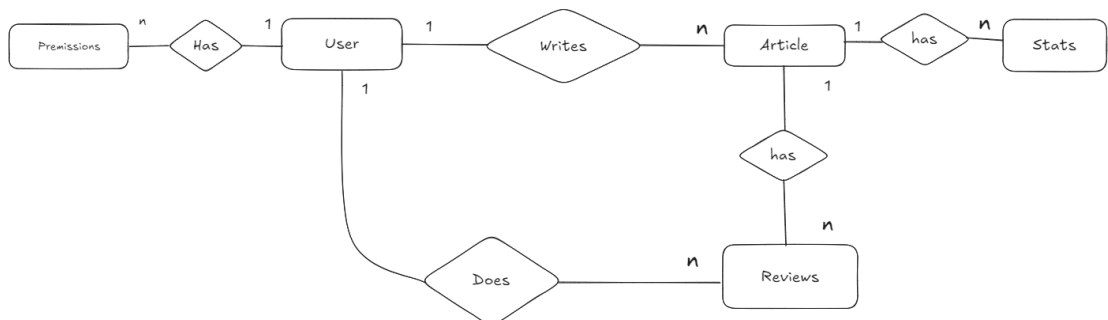
**Figura 3.1:** Diagrama de Caso de Uso



## Capítulo 4

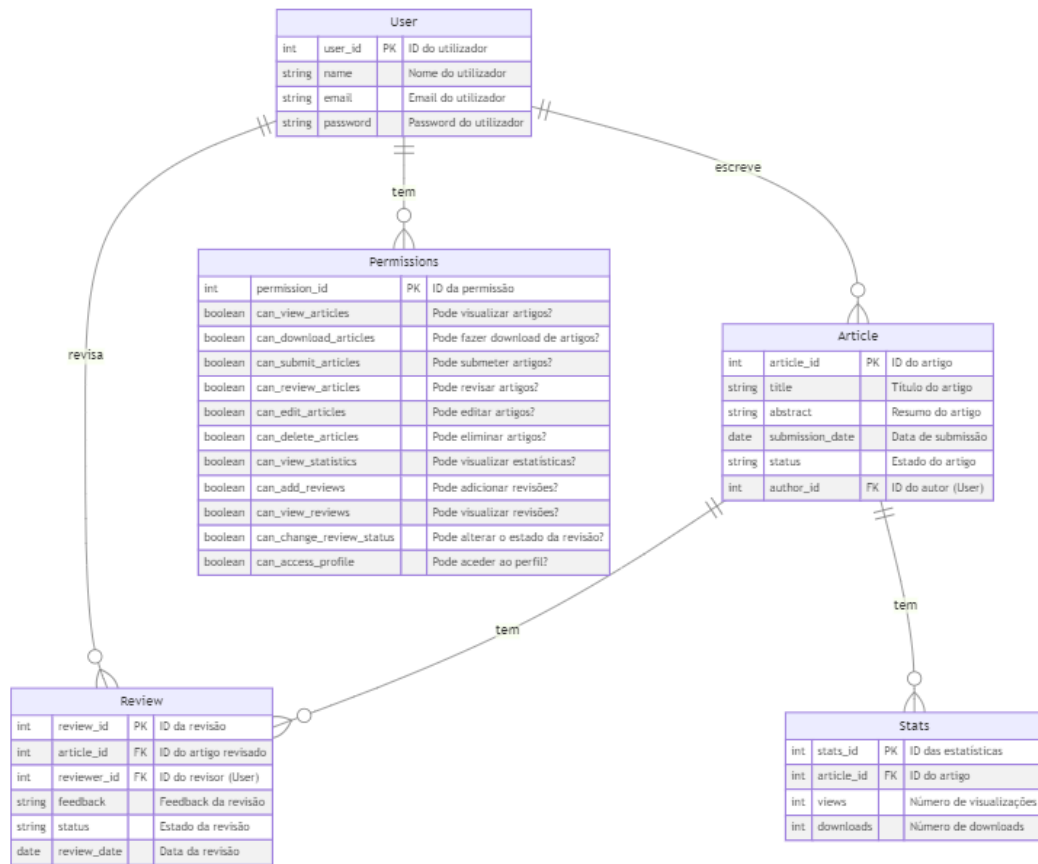
# Entidade relação

A figura 4.1 representa a entidade e relação da base de dados sendo que um utilizador escreve vários artigos. Um artigo tem varias reviews como também tem varias stats. O utilizador consoante as permissões realizam várias reviews.



**Figura 4.1:** Entidade relação

#### 4. ENTIDADE RELAÇÃO



**Figura 4.2:** Modelo físico da base de dados

## Capítulo 5

# Storyboard

### 5.1 Fluxo 1 — visualização do artigo

Na figura 5.1, O utilizador clica na barra de pesquisa e introduz os dados necessários para efetuar a pesquisa. Após introduzido os dados e submetido a pesquisa será redirecionado para uma página ao qual será exibido os artigos em formato de lista. Em seguida clicando num artigo será redirecionado para a página de visualização do artigo ao qual pode realizar download e ver outras informações do mesmo.



**Figura 5.1:** Visualização do artigo

## 5.2 Fluxo 2 — submeter artigo

Na figura 5.2, para submeter um artigo é necessário ter uma sessão iniciada com privilégio de researcher. Para ir à página de submeter um artigo é necessário clicar na foto de perfil representado a verde e em seguida selecionar no menu com a cor encarnada e clicar em "Submeter artigo". Após clicado será redirecionado para a página responsável por submeter o artigo. Após introduzir os dados necessários e clicar em "Submit" representado com a cor verde, será submetido o artigo. Após essa ação é necessário no modal de Alerta confirmar a ação.



**Figura 5.2:** Submeter artigo

## 5.3 Fluxo 3 — analisar artigo

Na figura 5.3, para analisar um artigo é necessário ter uma sessão iniciada com privilégio de reviewer. Para ir à página para analisar um artigo é necessário clicar na foto de perfil representado a verde e em seguida selecionar no menu com a cor encarnada e clicar em "Rever artigos". Após clicado será redirecionado para a página responsável por mostrar uma lista de artigos a espera para revisão. Quando escolhido um artigo para avaliar será redirecionado para a página contendo informações sobre o artigo, sendo possível adicionar comentários e aprovar (verde) ou rejeitar (encarnado) o artigo. Após clicar em aprovar ou rejeitar é demonstrado um modal de alerta para confirmar a ação.

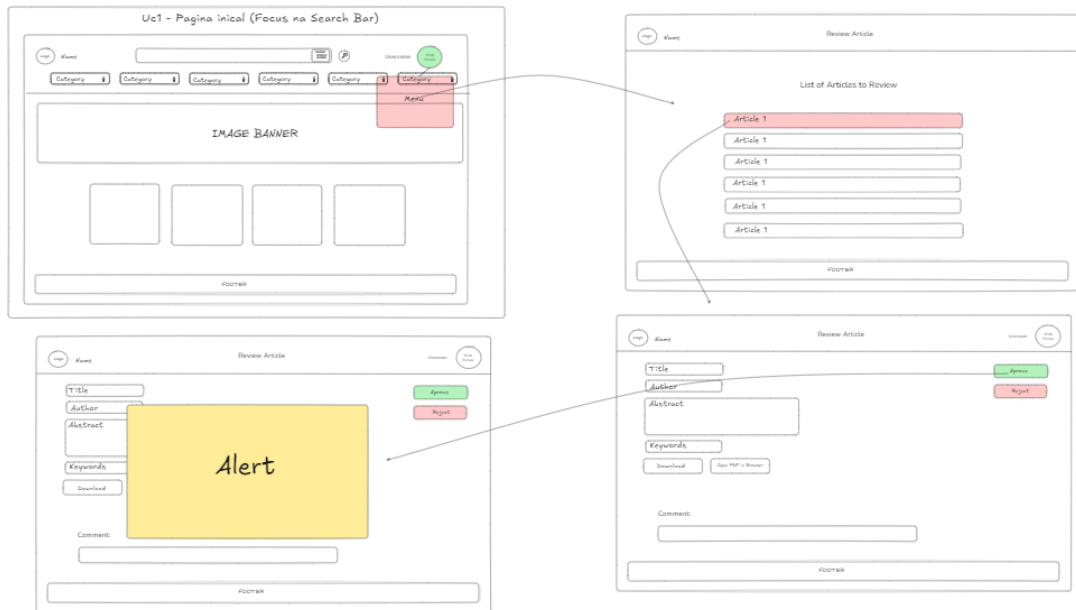


Figura 5.3: Analisar artigo

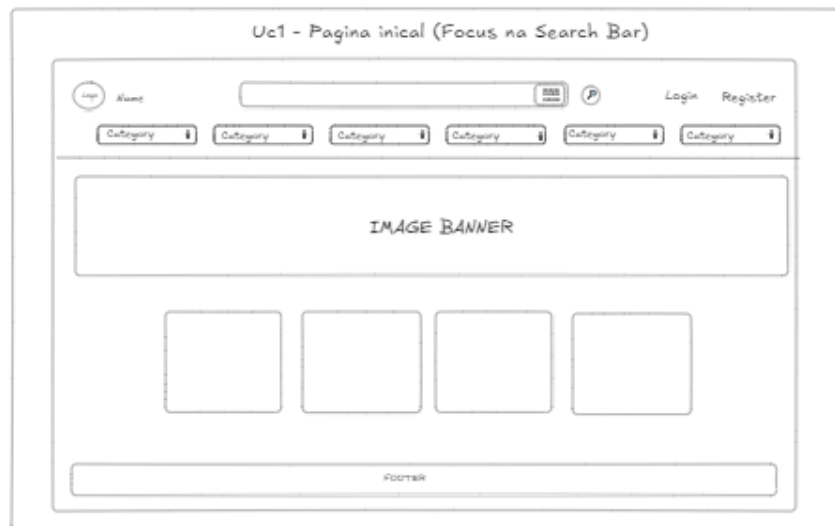


## Capítulo 6

# WireFrames

### 6.1 Página inicial

Na figura 6.1 encontra se representado a página inicial. Sendo o seu foco os filtros para pesquisar um determinado artigo. Na imagem conseguimos observar o que acontece ao clicar no botão da barra de pesquisa. O evento faz com que a barra de pesquisa aumente, conseguindo alterar alguns filtros como a categoria, o Autor, tipo de artigo entre outras categorias.



**Figura 6.1:** Página Inicial

### 6.2 Página — Visualizar artigo

Na figura 6.2 encontra se representado a visualização de determinado artigo. Ao clicar em qualquer artigo o utilizador será mandado para uma página deste género, em que estará os dados superficiais do artigo, como o título, a data de publicação, o autor e o abstract. Nesta

situação o utilizador tem a opção de fazer o download do artigo para o seu computador local, ou também tem a opção de visualizar o artigo no browser.



**Figura 6.2:** Página visualizar artigo

### 6.3 Página — Submeter artigo

Na figura 6.3 encontra se representado a página responsável pela submissão de um artigo. A informação sobre o autor já se encontra associada ao perfil que se encontra registado. Sendo necessário introduzir os dados relevantes do artigo como o seu title, abstract, keywords, submission type, topics e o artigo em formato pdf. Após preenchidos os dados serão possível submeter o artigo. Esta submissão será processada pelo back-end e atualizada na base de dados.



Uc3 - Submeter Artigo

Logo Name Username Profile Picture

Title:

Submission Type:

Abstract:

Topics:

Keywords:

Article File (PDF):  Choose File:

FOOTER

**Figura 6.3:** Página submeter artigo

## 6.4 Página — avaliar artigo

Na figura 6.4 encontra se representado a página responsável por avaliar os artigos, sendo que o primeiro wireframe listando os artigos submetidos e a espera de serem revistos. Após escolhido um artigo, ira para a página ao qual vai ter acesso ao artigo e todas as informações, sendo possível deixar comentários sobre o conteúdo do artigo. Após isso é possível Aceitar ou Rejeitar o artigo. Após selecionar uma destas opções será processado pelo back-end e atualizado na base de dados.

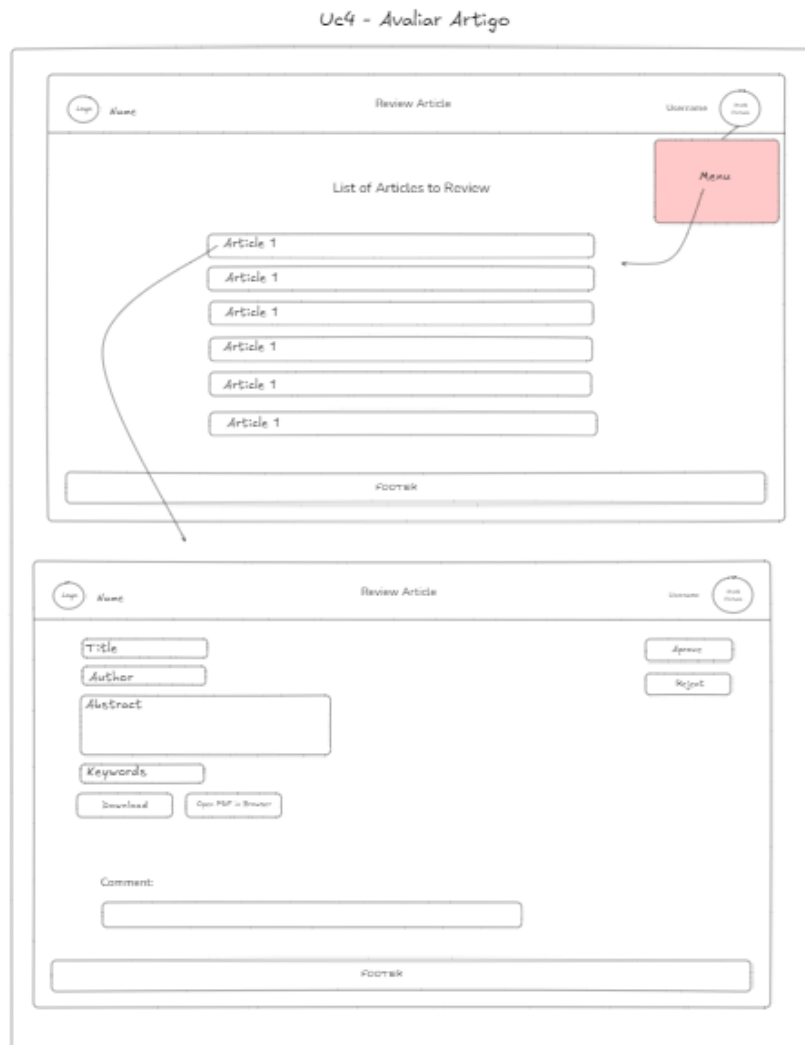
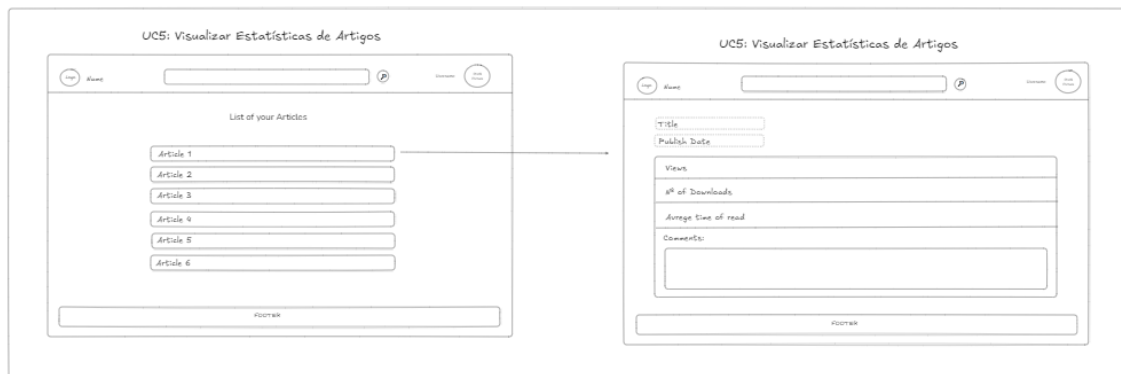


Figura 6.4: Página Avaliar artigo

## 6.5 Página — visualizar estatísticas do artigo

Na figura 6.5 encontra se representado a página responsável pela visualização da estatística de determinado artigo. Nem todos os utilizadores vão ter acesso a este tipo de informação. Apenas quem publicou o artigo vai ter acesso a este tipo de dados. Nestes dados estão incluídos o número de downloads, número de vezes que o artigo foi visto, tempo médio de leitura do artigo em questão, e comentários feitos por outros utilizadores.



**Figura 6.5:** Página Visualizar Estatísticas do artigo

## 6.6 Página — login e registo

Na figura 6.6 encontra se representado a página responsável pelo registo e login do utilizador. Esta pagina serve para o utilizador se registar e entrar na sua conta respetivamente. Esta ação não é completamente necessária para usar a aplicação web, visto que podemos visualizar os artigos não estando com a sessão iniciada.



**Figura 6.6:** Página login e registo



## Capítulo 7

# Melhorias efetuadas na análise e no desenho do sistema

As melhorias implementadas na fase de implementação, em comparação com a fase de desenho, destacam-se principalmente na maneira como os elementos (componentes) são apresentados, especialmente em termos de user interface (UI).

A estrutura da base de dados foi melhorada de forma a se adaptar às necessidades conforme o desenvolvimento.

Implementacao de PAginas Unauthorized



## Capítulo 8

# Implementação

Este capítulo, como o nome indica descreve todo o processo de implementação descrito quer na fase de análise, quer na fase de desenho. Com isto, primeiramente irei fazer uma abordagem ao desenvolvimento da API, esta na framework *Asp Net* e posteriormente a população dos dados como também decisões mais relevantes de código implementadas.

### 8.1 Arquitetura do sistema

### 8.2 Arquitetura do sistema

Para a implementação do projeto, foi necessário definir uma arquitetura para o sistema, visando a integração entre a aplicação e a API. A solução de arquitetura global encontrada baseia-se em três grandes características:

- **Models:** Os models consistem nas tabelas fundamentais, definindo assim as tabelas existentes na base de dados, bem como seus campos;
- **Controllers:** Os controllers realizam todas as operações CRUD necessárias na base de dados, sejam inserções de novos registros, atualizações ou apenas leituras. Além disso, os controllers retornam todos os pedidos HTTP para o sistema;
- **Views:** As views são responsáveis pelo encaminhamento das rotas aos pedidos necessários ao sistema.

Como pode ser observado na Figura ??, os três blocos das características descritas acima são representados. O "Client" corresponde ao nosso front-end, o "Controller" é responsável pelos pedidos realizados, e a "Base de Dados" é onde as operações ocorrem.

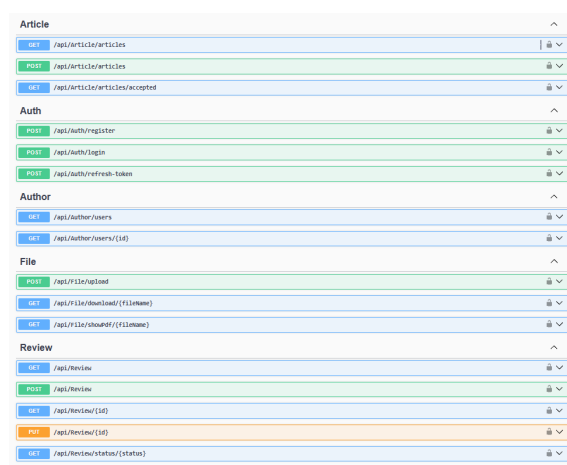
## 8.3 Tecnologias usadas

Para suporte à implementação do projeto, foi utilizada uma ferramenta que vem configurada inicialmente, designada *Swagger* permitindo interagir e fazer pedidos à API. Para além disso, foi utilizado o Asp.Net tanto no desenvolvimento da API, quanto do projeto em si.

## 8.4 Desenvolvimento da API

### 8.4.1 Especificação da interface

A API foi desenvolvida recorrendo a framework .NetCore com base na arquitetura REST. Com isto, para colmatar as necessidades dos casos de uso, esta é constituída por 16 endpoints (Fig. 4.2).



Article	
GET	/api/article/articles
POST	/api/article/articles
GET	/api/article/articles/accepted
Auth	
POST	/api/auth/register
POST	/api/auth/login
POST	/api/auth/refresh-token
Author	
GET	/api/author/users
GET	/api/author/users/{id}
File	
POST	/api/file/upload
GET	/api/file/download/{fileName}
GET	/api/file/showPDF/{fileName}
Review	
GET	/api/review
POST	/api/review
GET	/api/review/{id}
PUT	/api/review/{id}
GET	/api/review/status/{status}

Figura 8.1: Endpoints

### 8.4.2 Decisões de implementação

Esta aplicação seguiu a abordagem *code-first*, o que facilitou a realização de alterações nos campos ao longo do desenvolvimento. Para os modelos, definiu-se uma estrutura base dos dados que seriam armazenados nas tabelas da base de dados. Ao longo da aplicação, utilizou-se *DTO* (Data Transfer Object) para a abstração dos dados, permitindo utilizar apenas os campos necessários para realizar uma operação.

Para a gestão da sessão, foi utilizada a autenticação via *JWT* (JSON Web Token). Conforme a regra definida, certos controllers da API contêm restrições, sendo acessíveis apenas àqueles que estejam autenticados e possuam a role necessária para acessar esses métodos.

A visibilidade dos métodos foi gerida através de serviços e interfaces, com o objetivo de simplificar e encapsular métodos secundários, definidos como `private` quando são au-



xiliares a um processo necessário. Os controllers, por sua vez, têm a função de retornar os `HTTP Status` corretos conforme as circunstâncias. A validação dos campos é especialmente definida nos modelos, e essa validação é refletida para o restante da aplicação.

#### 8.4.3 Principais casos relevantes de programação

### 8.5 Desenvolvimento da App frontend/MVC

#### 8.5.1 Decisões de implementação

No que diz respeito à *App frontend/MVC*, optou-se pela abordagem de utilização de componentes devido à ocorrência de erros e à impossibilidade de exibir de forma dinâmica as informações para a interface do usuário (UI), já que a versão Razor não suporta o `RenderMode` através do servidor.

A programação das vistas tinha o exclusivo propósito de exibir as informações para a UI, sendo a lógica de negócios implementada em serviços. Esses serviços, assim como na API, contêm as regras de funcionamento dos casos de uso. Para a implementação e chamada de métodos, foram utilizadas interfaces, o que facilitou e organizou o desenvolvimento.

No contexto *MVC*, não foram implementadas grandes novidades, sendo a API responsável por todo o funcionamento da aplicação.

#### 8.5.2 Principais casos relevantes de programação

TokenManager Menu Reviews



## Capítulo 9

## Conclusão



# Bibliografia

- [1] UML - Casos de Uso - Conceitos (revisão), "*UML - Casos de Uso - Conceitos (revisão)*" - Sep 29, 2024

## BIBLIOGRAFIA

---

Documento elaborado com base no *template for final reports and dissertations* (Instituto Politécnico de Beja), disponível em <https://www.overleaf.com/project/5d936b9ea273390001434a37>, Version 1.0, 2023/01/25, Autor: João Paulo Barros, joao.barros@ipbeja.pt