



Entrega intermédia - Endpoints

Licenciatura em Engenharia Informática
Desenvolvimento de Aplicações Empresariais

Duarte Pereira (2222123)
José Fernandes (2211054)
Rafael Coelho (2241997)
Tatiana Frade (2024113212)

Leiria, dezembro de 2025

Índice

EP01 - Criar publicação	3
EP02 - Corrigir resumo gerado por IA numa publicação	4
EP03 - Atribuir rating a uma publicação	5
EP04 - Comentar publicação	6
EP06 - Consultar lista de publicações com visibilidade pública	7
EP07 - Consultar lista de publicações submetidas pelo próprio utilizador	9
EP08 - Consultar histórico de edições das publicações submetidas pelo próprio utilizador	12
EP09 - Pesquisar publicações	14
EP10 - Ordenar lista de publicações	15
EP11 - Subscrever uma tag	17
EP12 - Consultar o próprio histórico de atividade	18
EP13 - Editar dados pessoais	19
EP14 - Alterar palavra-passe	20
EP15 - Recuperar palavra-passe através do email	21
EP16 - Criar uma nova tag	22
EP17 - Eliminar tag (Hard delete)	23
EP18 - Desassociar uma ou mais tags de uma publicação (Soft delete)	24
EP19 - Ocultar ou mostrar comentários de uma publicação	25
EP20 - Ocultar ou mostrar publicação	26
EP21 - Consultar informação oculta (tags, comentários, publicações)	27
EP22 - Criar um novo utilizador	28
EP23 - Editar os dados de um utilizador	29
EP24 - Remover utilizador (Hard delete)	30
EP25 - Ativar utilizador	31
EP26 - Desativar utilizador (Soft delete)	32
EP27 - Alterar o papel (role) de um utilizador	33
EP28 - Consulta o histórico de atividade de qualquer utilizador	34
EP29 - Login / Autenticação do utilizador	35
EP30 - Logout / Terminar sessão do utilizador	36
EP31 - Consultar todos os utilizadores	37
EP32 - Consultar todos as Tags	39

Plataforma de Gestão de Publicações Científicas

EP01 - Criar publicação

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode criar/fazer upload de uma **publicação**, enviando um ficheiro PDF ou ZIP. O sistema gera automaticamente o campo “summary” com base no conteúdo do ficheiro. O autor da publicação é o utilizador autenticado que efetua o pedido.

POST	/api/posts
Headers	Content-Type: multipart/form-data Authorization: Bearer <token_colaborador_responsavel_admin>
Body	title: Modelo CNN para deteção de tumores scientific_area: Modelo Tumores is_visible: false file: modelo_tumores.pdf

Resposta JSON (**201 Created**)

```
{
  "id": 1,
  "title": "Modelo CNN para deteção de tumores",
  "scientific_area": "Modelo Tumores",
  "summary": "Resumo gerado automaticamente no momento da criação",
  "is_visible": false,
  "file_url": "https://dominio.com/api/posts/1/file",
  "author": {
    "id": 7,
    "name": "Joana B"
  },
  "average_rating": 0,
  "ratings_count": 0,
  "tags": [],
  "created_at": "2025-10-30T14:22:05Z"
}
```

Possíveis códigos de resposta:

- **201 Created** — Publicação criada com sucesso
- **400 Bad Request** — Campos obrigatórios ausentes ou inválidos

- **401 Unauthorized** — Token inválido expirado, ou ausente
- **409 Conflict** — Já existe uma publicação com o mesmo título.

EP02 - Corrigir resumo gerado por IA numa publicação

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode editar o resumo (summary) gerado automaticamente por IA, mas apenas de publicações da sua própria autoria.

PATCH	/api/posts/{post_id}/summary
Header	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	{ "summary": "resumo corrigido pelo autor" }

Resposta JSON (200 OK)

```
{
  "id": 1,
  "title": "Modelo CNN para deteção de tumores",
  "scientific_area": "Modelo Tumores",
  "summary": "Resumo corrigido pelo autor.",
  "is_visible": false,
  "file_url": "https://dominio.com/api/posts/1/file",
  "author": {
    "id": 7,
    "name": "Joana B"
  },
  "average_rating": 0,
  "ratings_count": 0,
  "tags": [],
  "updated_at": "2025-10-31T10:22:40Z"
}
```

Possíveis códigos de resposta:

- **200 OK** — Resumo atualizado com sucesso
- **400 Bad Request** — Campo summary vazio ou inválido
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — Utilizador não é o autor da publicação
- **404 Not Found** — Publicação não encontrada (post_id inexistente)

EP03 - Atribuir rating a uma publicação

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode atribuir uma classificação (rating, entre 1 a 5) a uma **publicação** pública (is_visible = true), exceto às publicações da sua autoria.

POST	/api/posts/{post_id}/ratings
Header	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "rating": 5 }</pre> <p>Valores permitidos para rating: 1,2,3,4,5</p>

Resposta JSON (200 OK)

```
{  
  "post_id": 1,  
  "title": "Modelo CNN para deteção de tumores",  
  "scientific_area": "Modelo Tumores",  
  "summary": "Resumo corrigido pelo autor.",  
  "is_visible": false,  
  "file_url": "https://dominio.com/api/posts/1/file",  
  "author": {  
    "id": 7,  
    "name": "Joana B"  
  },  
  "rating_given": 5,  
  "average_rating": 3.8,  
  "ratings_count": 12,  
  "tags": [  
    { "id": 1, "name": "Deep Learning" },  
    { "id": 4, "name": "Imagem Médica" }  
  ],  
  "updated_at": "2025-11-01T14:55:29Z"  
}
```

Possíveis códigos de resposta:

- **200 OK** — Rating atribuído com sucesso à publicação
- **400 Bad Request** — Rating inválido (fora do intervalo permitido)
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — Publicação não têm visibilidade pública ou o utilizador tenta avaliar a sua própria publicação
- **404 Not Found** — Publicação não encontrada (post_id inexistente)

EP04 - Comentar publicação

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode adicionar um comentário a uma publicação. Comentários podem ser feitos apenas em publicações com visibilidade pública (is_visible = true). Apenas o autor pode comentar na sua publicação sem que esta esteja visível para o público.

POST	/api/posts/{post_id}/comments
Header	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "comment": "Bastante relevante para o projeto X!" }</pre>

Resposta JSON (200 OK)

```
{
  "comment_id": 1,
  "post_id": 1,
  "author": {
    "id": 7,
    "name": "Joana B"
  },
  "comment": "Bastante relevante para o projeto X!",
  "created_at": "2025-11-01T15:14:08Z",
  "post": {
    "id": 1,
    "title": "Modelo CNN para deteção de tumores",
    "scientific_area": "Modelo Tumores",
    "summary": "Resumo corrigido pelo autor.",
    "is_visible": true,
    "file_url": "https://dominio.com/api/posts/1/file",
    "average_rating": 3.9,
    "ratings_count": 14,
    "tags": [
      { "id": 1, "name": "Deep Learning" },
      { "id": 4, "name": "Imagem Médica" }
    ]
  }
}
```

<pre>], "updated_at": "2025-11-01T15:14:08Z" } } </pre>
<p>Possíveis códigos de resposta:</p> <ul style="list-style-type: none"> • 200 OK — Comentário adicionado com sucesso • 400 Bad Request — Comentário inválido ou vazio • 401 Unauthorized — Token inválido expirado, ou ausente • 403 Forbidden — Publicação não é pública e o utilizador não é o autor • 404 Not Found — Publicação não encontrada (post_id inexistente)

EP06 - Consultar lista de publicações com visibilidade pública	
O utilizador autenticado com o papel Colaborador, Responsável ou Administrador ou Visitante pode consultar a lista de todas as publicações com visibilidade pública (is_visible = true).	
GET	/api/posts
Header	(vazio)
Query params	<div> <div>?search=CNN</div> <div>→ filtrar por título ou área científica</div> </div> <div> <div>?tag=2</div> <div>→ filtrar por tag associada</div> </div> <div> <div>?page=2</div> <div>→ página de listagem (default = 1)</div> </div> <div> <div>?limit=20</div> <div>→ número de resultados por página (default = 10)</div> </div>
Body	(vazio)

Resposta JSON (200 OK)

```
[
  {
    "id": 101,
    "title": "Artigo 1",
    "scientific_area": "Ciência de Dados",
    "summary": "Resumo opcional da publicação.",
    "is_visible": true,
    "file_url": "https://dominio.com/api/posts/101/file",
    "author": {
      "id": 7,
      "name": "Joana B"
    },
    "average_rating": 4.5,
    "ratings_count": 10,
    "tags": [
      { "id": 1, "name": "Projeto X" },
      { "id": 2, "name": "LLM" }
    ],
    "created_at": "2025-10-30T12:10:45Z",
    "updated_at": "2025-10-30T12:10:45Z"
  },
  {
    "id": 102,
    "title": "Novo Material Compósito",
    "scientific_area": "Ciência dos Materiais",
    "summary": "Resumo do trabalho apresentado.",
    "is_visible": true,
    "file_url": "https://dominio.com/api/posts/102/file",
    "author": {
      "id": 8,
      "name": "Manuel C"
    },
    "average_rating": 3.8,
    "ratings_count": 7,
    "tags": [
      { "id": 2, "name": "LLM" }
    ],
    "created_at": "2025-10-30T14:05:12Z",
    "updated_at": "2025-10-30T14:05:12Z"
  }
]
```

Possíveis códigos de resposta:

- **200 OK** — Lista de publicações públicas retornada com sucesso
- **400 Bad Request** — Parâmetros de pesquisa ou paginação inválidos (ex: page = -1, limit = 0, tag não numérico)

- **401 Unauthorized** — Token inválido expirado, ou ausente

EP07 - Consultar lista de publicações submetidas pelo próprio utilizador

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode consultar a lista de publicações que ele próprio submeteu (incluindo as não visíveis para todos)

GET	/api/users/me/posts	
Header	Authorization: Bearer <token_colaborador_responsavel_admin>	
Query params	?page=1	→ paginação (default = 1)
	&limit=10	→ número de registos por página (default = 10)
	&is_visible=true	→ filtrar apenas visíveis (true/false)
	&tag=2	→ filtrar por tag específica (ID numérico)

Body	(vazio)
<p>Resposta JSON (200 OK)</p> <pre>[{ "id": 101, "title": "Artigo 1", "scientific_area": "Ciência de Dados", "summary": "Resumo opcional da publicação.", "is_visible": true, "file_url": "https://dominio.com/api/posts/101/file", "author": { "id": 1, "name": "Joana B" }, "average_rating": 4.5, "ratings_count": 10, "tags": [{ "id": 1, "name": "Projeto X" }, { "id": 2, "name": "LLM" }], "created_at": "2025-10-30T12:10:45Z", "updated_at": "2025-10-30T12:10:45Z" }, { "id": 102, "title": "Novo Material Compósito", "scientific_area": "Ciência dos Materiais", "summary": "Resumo do trabalho apresentado.", "is_visible": true, "file_url": "https://dominio.com/api/posts/102/file", "author": { "id": 1, "name": "Joana B" }, "average_rating": 3.8, "ratings_count": 7, "tags": [{ "id": 2, "name": "LLM" }], "created_at": "2025-10-30T14:05:12Z", "updated_at": "2025-10-30T14:05:12Z" }]</pre>	
<p>Possíveis códigos de resposta:</p> <ul style="list-style-type: none"> ● 200 OK — Lista de publicações retornada com sucesso ● 400 Bad Request — Parâmetros de paginação ou filtros inválidos (ex: page = -1, limit = 0, tag não numérico) 	

EP08 - Consultar histórico de edições das publicações submetidas pelo próprio utilizador

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode consultar o histórico de edições de uma publicação da qual é autor, incluindo as não visíveis para todos.

GET	/api/posts/{post_id}/history
Header	Authorization: Bearer <token_colaborador_responsavel_admin>
Query params	?page=1 → paginação (default = 1) &limit=10 → número de registos por página (default = 10)
Body	(vazio)

Resposta JSON (200 OK)

```
[
  {
    "edit_id": 1,
    "post_id": 101,
    "edited_at": "2025-10-31T10:22:40Z",
    "edited_by": {
      "id": 1,
      "name": "Joana B"
    },
    "changes": {
      "summary": "Resumo corrigido pelo autor."
    }
  },
  {
    "edit_id": 2,
    "post_id": 101,
    "edited_at": "2025-11-01T09:15:22Z",
    "edited_by": {
      "id": 1,
      "name": "Joana B"
    },
    "changes": {
      "title": "Artigo 1 - Versão Revista",
      "tags": [
        { "id": 2, "name": "LLM" }
      ]
    }
  }
]
```

Possíveis códigos de resposta:

- **200 OK** — Histórico de edições retornado com sucesso
- **400 Bad Request** — Parâmetros de paginação inválidos (ex: page = -1, limit = 0)
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não é o autor da publicação, por isso não pode consultar o histórico
- **404 Not Found** — Publicação não existe (post_id inexistente)

EP09 - Pesquisar publicações

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode pesquisar publicações através de múltiplos critérios (título, autor, área científica, tags, intervalo de datas, etc). É possível combinar vários filtros em simultâneo. A pesquisa retorna apenas publicações com visibilidade pública (`is_visible = true`).

POST	/api/posts/search
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "title": "CNN", "author_id": 7, "scientific_area": "Modelo Tumores", "tags": [1, 4], "date_from": "2025-10-01", "date_to": "2025-11-01" }</pre>

Resposta JSON (200 OK)

```
[
  {
    "id": 1,
    "title": "Modelo CNN para deteção de tumores",
    "scientific_area": "Modelo Tumores",
    "summary": "Modelo desenvolvido para classificação de imagens...",
    "is_visible": true,
    "file_url": "https://dominio.com/api/posts/1/file",
    "author": {
      "id": 7,
      "name": "Joana B"
    },
    "average_rating": 3.9,
    "ratings_count": 14,
    "tags": [
      { "id": 1, "name": "Deep Learning" },
      { "id": 4, "name": "Imagem Médica" }
    ],
    "created_at": "2025-10-30T14:22:05Z",
    "updated_at": "2025-11-02T10:21:18Z"
  }
]
```

Possíveis códigos de resposta:

- **200 OK** — Lista de publicações filtrada retornada com sucesso
- **400 Bad Request** — Parâmetros de pesquisa inválidos (ex: tags não é array, data_from > date_to)
- **401 Unauthorized** — Token inválido expirado, ou ausente

EP10 - Ordenar lista de publicações

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode ordenar a lista de publicações públicas por 3 critérios: número de comentários (comments_count), rating médio (average_rating) e número de ratings (ratings_count).

POST	/api/posts/sort
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "sort_by": "average_rating", "order": "desc" }</pre> <p>Valores permitidos para o sort_by: average_rating, comments_count, ratings_count. Para o order: asc e desc</p>

Resposta JSON (lista de publicações ordenadas) (200 OK)

```
[
  {
    "id": 1,
    "title": "Modelo CNN para deteção de tumores",
    "scientific_area": "Modelo Tumores",
    "summary": "Resumo opcional da publicação.",
    "is_visible": true,
    "file_url": "https://dominio.com/api/posts/1/file",
    "author": {
      "id": 7,
      "name": "Joana B"
    },
    "average_rating": 3.9,
    "ratings_count": 14,
    "comments_count": 5,
    "tags": [
```

```
{ "id": 1, "name": "Deep Learning" },  
{ "id": 4, "name": "Imagem Médica" }  
],  
  
"created_at": "2025-10-30T14:22:05Z",  
"updated_at": "2025-11-02T10:21:18Z"  
}  
]
```

Possíveis códigos de resposta:

- **200 OK** — Lista de publicações ordenada retornada com sucesso
- **400 Bad Request** — Parâmetros inválidos (ex.: sort_by não permitido, ou order diferente de “asc” ou “desc”)
- **401 Unauthorized** — Token inválido expirado, ou ausente

EP11 - Subscriver uma tag

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode subscriver uma tag. Subscriver uma tag significa que o utilizador manifesta interesse em receber ou acompanhar publicações que tenham essa tag associada.

POST	/api/tags/{tag_id}/subscribe
Header	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
{
  "message": "Tag subscrita com sucesso",
  "tag": {
    "id": 3,
    "name": "Teste"
  },
  "user": {
    "id": 7,
    "name": "Joana B"
  }
}
```

Possíveis códigos de resposta:

- **200 OK** — Tag subscrita com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **404 Not Found** — Tag não encontrada
- **409 Conflict** — Tag já subscrita pelo utilizador

EP12 - Consultar o próprio histórico de atividade

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode consultar o seu próprio histórico de atividades (ex: uploads de publicações, edições realizadas, atribuição de tags, etc).

GET	/api/users/{id}/activity
Header	Authorization: Bearer <token_colaborador_responsavel_admin>
Query params	?page=2 → paginação (default = 1) ?limit=20 → número de resultados por página (default = 10)
Body	(vazio)

Resposta JSON (200 OK)

```
{
  "activities": [
    {
      "id": 123,
      "type": "upload",
      "title": "Modelo CNN para deteção de tumores",
      "date": "2025-11-01T10:15:00Z"
    },
    {
      "id": 124,
      "type": "edit",
      "title": "Modelo CNN revisto",
      "date": "2025-11-02T14:30:00Z"
    }
  ],
  "page": 1,
  "total_pages": 5,
  "total_activities": 25
}
```

Possíveis códigos de resposta:

- **200 OK** — Histórico retornado com sucesso

- **401 Unauthorized** — Token inválido expirado, ou ausente

EP13 - Editar dados pessoais

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode editar os seus próprios dados pessoais (ex: nome, email, morada, telefone, etc)

PATCH	/api/users/me
--------------	---------------

Header	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
---------------	---

Body	<pre>{ "email": "novoemail@exemplo.com", "name": "João Silva" }</pre>
-------------	---

Resposta JSON (200 OK)

```
{
  "id": 456,
  "name": "João Silva",
  "email": "novoemail@exemplo.com",
  "role": "Colaborador",
  "updated_at": "2025-11-07T12:00:00Z"
}
```

Possíveis códigos de resposta:

- **200 OK** — Dados pessoais atualizados com sucesso
- **400 Bad Request** — Dados inválidos (ex.: email mal formatado)
- **401 Unauthorized** — Token inválido expirado, ou ausente

EP14 - Alterar palavra-passe

O utilizador autenticado com o papel **Colaborador, Responsável ou Administrador** pode alterar a sua própria palavra-passe fornecendo a password atual e a nova password.

PATCH	/api/users/me/password
Header	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "current_password": "senha_atual123", "new_password": "nova_senha456" }</pre>
Resposta JSON (200 OK) <pre>{ "message": "Palavra-passe alterada com sucesso", "updated_at": "2025-11-07T12:30:00Z" }</pre>	
Possíveis códigos de resposta: <ul style="list-style-type: none">● 200 OK — Palavra-passe alterada com sucesso● 400 Bad Request — Dados inválidos (ex.: nova password fraca ou campos em falta, password atual incorreta)● 401 Unauthorized — Token inválido expirado, ou ausente	

EP15 - Recuperar palavra-passe através do email

O utilizador que não consegue iniciar sessão solicita a redefinição da palavra-passe, indicando o seu endereço de email. Se o email corresponder a uma conta válida, é enviado um email com instruções.

POST	/api/users/password/forgot
Header	Content-Type: application/json
Body	<pre>{ "email": "usuario@exemplo.com" }</pre>
Resposta JSON (200 OK) <pre>{ "message": "Se existir uma conta associada a este email, enviámos instruções para redefinir a palavra-passe." }</pre>	
Possíveis códigos de resposta: <ul style="list-style-type: none">• 200 OK — Pedido registado com sucesso (não expõe se o e-mail existe para segurança)• 400 Bad Request — Email inválido (ex:mal formatado)	

EP16 - Criar uma nova tag

O utilizador autenticado com o papel **Responsável** ou **Administrador** cria uma nova tag no sistema. As tags servem para categorizar e facilitar a pesquisa de publicações.

POST	/api/tags
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "name": "Machine Learning" }</pre>

Resposta JSON (**201 Created**)

```
{  
  "message": "Tag criada com sucesso",  
  "tag": {  
    "id": 14,  
    "name": "Machine Learning"  
  }  
}
```

Possíveis códigos de resposta:

- **201 Created** — Tag criada com sucesso
- **400 Bad Request** — Nome da tag inválido (ex:vazio ou acima do tamanho permitido)
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não possui o papel Responsável ou Administrador, logo não tem permissão para criar tags.
- **409 Conflict** — Já existe uma tag com esse nome no sistema.

EP17 - Eliminar tag (Hard delete)

O utilizador autenticado com o papel **Responsável ou Administrador** pode eliminar uma tag do sistema. Se a tag estiver associada a publicações, a eliminação deve ser impedida.

DELETE	/api/tags/{tag_id}
Headers	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
{
  "message": "Tag eliminada com sucesso"
}
```

Possíveis códigos de resposta:

- **200 OK** — Tag eliminada com sucesso
- **400 Bad Request** — A tag não pode ser eliminada (ex.: está em uso e a política não permite remoção)
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não possui o papel Responsável ou Administrador, logo não pode eliminar tags.
- **404 Not Found** — Não existe uma tag com o ID fornecido.

EP18 - Desassociar uma ou mais tags de uma publicação (Soft delete)

O utilizador autenticado com o papel **Responsável** ou **Administrador** pode remover/desassociar uma ou mais tags previamente associadas a uma publicação.

DELETE	/api/posts/{post_id}/tags
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "tags": [3, 7] }</pre>

Resposta JSON (200 OK)

```
{  
  "message": "Tags removidas com sucesso",  
  "post_id": 101,  
  "removed_tags": [3, 7]  
}
```

Possíveis códigos de resposta:

- **200 OK** — Tags removidas da publicação com sucesso
- **400 Bad Request** — Lista de tags inválida ou vazia, ou contém tags que não estão associadas à publicação
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador autenticado não possui o papel Responsável ou Administrador, logo não pode desassociar tags de publicações
- **404 Not Found** — Publicação não encontrada. (post_id inexistente)

EP19 - Ocultar ou mostrar comentários de uma publicação

O utilizador autenticado com o papel **Responsável ou Administrador** pode alterar a visibilidade dos comentários associados a uma publicação.

PUT	/api/posts/{post_id}/comments/visibility
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "visible": false }</pre> Valores permitidos para visible: true e false

Resposta JSON (200 OK)

```
{
  "message": "Visibilidade dos comentários atualizada com sucesso",
  "post_id": 101,
  "visible": false,
  "updated_at": "2025-11-07T17:10:00Z"
}
```

Possíveis códigos de resposta:

- **200 OK** — Visibilidade dos comentários atualizada
- **400 Bad Request** — Campo visible inválido ou ausente
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O usuário não tem o papel Responsável ou Administrador logo não pode alterar a visibilidade dos comentários de uma publicação.

- **404 Not Found** — Publicação não encontrada (post_id inexistente).

EP20 - Ocultar ou mostrar publicação

O utilizador autenticado com o papel **Responsável** ou **Administrador** pode alterar a visibilidade de uma publicação no sistema.

PUT	/api/posts/{post_id}/visibility
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "visible": false }</pre> <p>Valores permitidos para visible: true e false</p>

Resposta JSON (200 OK)

```
{  
  "message": "Visibilidade da publicação atualizada com sucesso",  
  "post_id": 101,  
  "visible": false,  
  "updated_at": "2025-11-07T17:10:00Z"  
}
```

Possíveis códigos de resposta:

- **200 OK** — Visibilidade da publicação atualizada com sucesso
- **400 Bad Request** — Campo visible inválido ou ausente
- **401 Unauthorized** — Token inválido expirado, ou ausente

- **403 Forbidden** — O Utilizador não tem papel Responsável ou Administrador, logo não pode alterar a visibilidade de uma publicação.
- **404 Not Found** — Publicação não encontrada, (post_id inexistente)

EP21 - Consultar informação oculta (tags, comentários, publicações)

O utilizador autenticado com o papel **Responsável ou Administrador** pode visualizar conteúdo que está marcado como oculto (ex:publicações, comentários,tags). É possível filtrar pelo tipo de entidade oculta usando query params.

GET	/api/hidden						
Headers	Authorization: Bearer <token_colaborador_responsavel_admin>						
Query Params	<table> <tr> <td>?type=posts</td><td>→ listar publicações ocultas</td></tr> <tr> <td>?type=comments</td><td>→ listar comentários ocultos</td></tr> <tr> <td>?type=tags</td><td>→ listar tags ocultas/arquivadas</td></tr> </table>	?type=posts	→ listar publicações ocultas	?type=comments	→ listar comentários ocultos	?type=tags	→ listar tags ocultas/arquivadas
?type=posts	→ listar publicações ocultas						
?type=comments	→ listar comentários ocultos						
?type=tags	→ listar tags ocultas/arquivadas						
Body	(vazio)						

Resposta JSON (200 OK)

```
{
  "type": "posts",
  "results": [
    {
```

<pre> "post_id": 101, "title": "Modelo CNN para detecção de tumores", "author": "João Silva", "visible": false, "date_hidden": "2025-11-05T12:10:00Z" }, { "post_id": 98, "title": "Estudo sobre regressão logística", "author": "Maria Ferreira", "visible": false, "date_hidden": "2025-11-02T09:32:00Z" }], "total": 2 } </pre>
<p>Possíveis códigos de resposta:</p> <ul style="list-style-type: none"> ● 200 OK — Informação oculta listada com sucesso ● 400 Bad Request — Tipo (type) inválido ou não suportado ● 401 Unauthorized — Token inválido expirado, ou ausente ● 403 Forbidden — O utilizador não tem papel Responsável ou Administrador, logo não têm permissão para visualizar conteúdo oculto ● 404 Not Found — Nenhum conteúdo oculto encontrado

EP22 - Criar um novo utilizador	
<p>O utilizador autenticado com o papel Administrador pode criar um novo utilizador no sistema, definindo o seu papel (Colaborador, Responsável ou Administrador) e dados pessoais iniciais.</p>	
POST	/api/users
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre> { "name": "João Silva", "email": "joao.silva@example.com", "password": "PasswordForte123", "role": "Colaborador" } </pre>

	} Valores permitidos para role: Colaborador, Responsável e Administrador
Resposta JSON (201 Created) <pre>{ "message": "Utilizador criado com sucesso", "user_id": 152, "name": "João Silva", "email": "joao.silva@example.com", "role": "Colaborador" }</pre>	
Possíveis códigos de resposta: <ul style="list-style-type: none"> ● 201 Created — Utilizador criado com sucesso ● 400 Bad Request — Dados inválidos (ex.: email mal formatado, password fraca ou campos obrigatórios ausentes) ● 401 Unauthorized — Token inválido expirado, ou ausente ● 403 Forbidden — O Utilizador autenticado não possui papel Administrador, logo não pode criar contas de outros utilizadores. ● 409 Conflict — Já existe um utilizador com o mesmo email no sistema. 	

EP23 - Editar os dados de um utilizador	
O utilizador autenticado com o papel Administrador pode editar os dados de qualquer utilizador.	
PUT	/api/users/{user_id}
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	<pre>{ "name": "João Silva", "email": "joao.silva@newmail.com", "role": "Responsável" }</pre> Valores permitidos para role: Colaborador, Responsável e Administrador
Resposta JSON (200 OK)	

```
{
  "message": "Dados atualizados com sucesso",
  "user_id": 152,
  "name": "João Silva",
  "email": "joao.silva@newmail.com",
  "role": "Responsável"
}
```

Possíveis códigos de resposta:

- **200 OK** — Dados atualizados com sucesso
- **400 Bad Request** — Dados inválidos (ex.: email mal formatado, campos obrigatórios ausentes)
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não tem o papel Administrador, logo não têm permissões para editar os dados de um utilizador.
- **404 Not Found** — Utilizador não encontrado, (user_id inexistente)

EP24 - Remover utilizador (Hard delete)

O utilizador autenticado com o papel **Administrador** pode eliminar um utilizador do sistema.

DELETE	/api/users/{user_id}
Headers	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
{
  "message": "Utilizador eliminado do sistema"
}
```

Possíveis códigos de resposta:

- **200 OK** — Utilizador removido com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não tem o papel Administrador, logo não tem permissão para remover utilizadores.
- **404 Not Found** — Utilizador não encontrado (user_id inexistente)

EP25 - Ativar utilizador

O utilizador autenticado com o papel **Administrador** pode reativar um utilizador previamente desativado (soft delete). Este endpoint restaura o acesso do utilizador ao sistema.

PUT	/api/users/{user_id}/activate
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)
Resposta JSON (200 OK)	

```
{
  "message": "Utilizador ativado com sucesso",
  "user_id": 152,
  "status": "active"
}
```

Possíveis códigos de resposta:

- **200 OK** — Utilizador ativado com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não possui o papel Administrador, logo não tem permissões para ativar utilizadores.
- **404 Not Found** — Utilizador não encontrado, (user_id inexistente)
- **409 Conflict** — Utilizador já se encontra ativo

EP26 - Desativar utilizador (Soft delete)

O utilizador autenticado com o papel **Administrador** pode desativar utilizadores. Enquanto está desativado, um utilizador não pode fazer login nem interagir com o sistema, mas os seus dados permanecem preservados.

PUT	/api/users/{user_id}/deactivate
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
{
  "message": "Utilizador desativado com sucesso",
  "user_id": 152,
  "status": "inactive"
}
```

Possíveis códigos de resposta:

- **200 OK** — Utilizador desativado com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não possui o papel Administrador, logo não tem permissões para desativar outros utilizadores
- **404 Not Found** — Utilizador não encontrado, (user_id inexistente)
- **409 Conflict** — Utilizador já se encontra desativado

EP27 - Alterar o papel (role) de um utilizador

O utilizador autenticado com o papel **Administrador** pode alterar o papel de outro utilizador, definindo se este é **Colaborador**, **Responsável** ou **Administrador**.

PUT	/api/users/{user_id}/role
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>

Body	<pre>{ "role": "Responsável" }</pre> <p>Valores permitidos para role: Colaborador, Responsável e Administrador</p>
Resposta JSON (200 OK) <pre>{ "message": "Papel do utilizador atualizado com sucesso", "user_id": 152, "new_role": "Responsável" }</pre>	
Possíveis códigos de resposta: <ul style="list-style-type: none"> • 200 OK — Papel atualizado com sucesso • 400 Bad Request — Papel inválido ou não permitido • 401 Unauthorized — Token inválido expirado, ou ausente • 403 Forbidden — O utilizador não possui o papel de Administrador, logo não tem permissões para alterar o role de outros utilizadores. • 404 Not Found — Utilizador não encontrado, (user_id inexistente) 	

EP28 - Consulta o histórico de atividade de qualquer utilizador

O utilizador autenticado com o papel **Administrador** pode consultar o histórico de atividade de qualquer utilizador (ex:uploads, edições, comentários, ratings,etc)

GET	/api/users/{user_id}/activity
------------	-------------------------------

Headers	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)
Resposta JSON (200 OK) <pre>{ "activities": [{ "id": 123, "type": "upload", "title": "Publicação sobre Redes Neurais", "date": "2025-11-01T10:15:00Z" }, { "id": 124, "type": "edit", "title": "Edição do dataset MNIST", "date": "2025-11-02T14:30:00Z" }], "page": 1, "total_pages": 3, "total_activities": 42 }</pre>	
Possíveis códigos de resposta: <ul style="list-style-type: none"> • 200 OK — Histórico retornado com sucesso • 400 Bad Request — Parâmetros inválidos (ex:número de página negativa) • 401 Unauthorized — Token inválido expirado, ou ausente • 403 Forbidden — O utilizador não possui o papel de Administrador, logo não tem permissões para consultar o histórico de atividades de outros utilizadores • 404 Not Found — Utilizador não encontrado, (user_id inexistente) 	

EP29 - Login / Autenticação do utilizador

O utilizador fornece o email e password para autenticar-se no sistema e receber um token de acesso ao sistema.

POST	/api/auth/login
Headers	Content-Type: application/json
Body	<pre>{ "email" : "mail@ipleiria.pt " "password" : "passwordLegit" }</pre>
Resposta JSON (200 OK) <pre>{ "message": "Autenticação bem-sucedida", "token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9...", "user": { "id": 7, "name": "Joana B", "role": "Colaborador" } }</pre>	
Possíveis códigos de resposta: <ul style="list-style-type: none"> • 200 OK — Login efetuado com sucesso • 400 Bad Request — Campos obrigatórios ausentes ou inválidos (má formatação do email) • 401 Unauthorized — Credenciais inválidas • 429 Too Many Requests — Tentativas de login falhas excessivas 	

EP30 - Logout / Terminar sessão do utilizador

O utilizador autenticado encerra a sessão atual, invalidando o token de acesso

POST	/api/auth/logout
Headers	Content-Type: application/json Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)
Resposta JSON (200 OK) <pre>{ "message": "Logout efetuado com sucesso" }</pre>	
Possíveis códigos de resposta: <ul style="list-style-type: none"> • 200 OK — Logout efetuado com sucesso • 401 Unauthorized — Token Inválido, expirado ou ausente 	

O utilizador autenticado com o papel **Administrador** pode visualizar todos os utilizadores do sistema.

GET	/api/users/
Headers	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
[
  {
    "id": 1,
    "name": "Joana B",
    "email": "joana@exemplo.com",
    "role": "Colaborador",
    "status": "active"
  },
  {
    "id": 2,
    "name": "Tio Patinhas",
    "email": "patinhas@exemplo.com",
    "role": "Responsável",
    "status": "active"
  }, ...
]
```

Possíveis códigos de resposta:

- **200 OK** — Utilizadores retornados com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não é Administrador, logo não tem permissões para consultar os utilizadores

EP32 - Consultar todos as Tags

O utilizador autenticado com o papel **Responsável** ou **Administrador** pode visualizar todas as Tags do sistema, sendo estas ocultas ou não.

GET	/api/tags/
Headers	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
[
  {
    "id": 2,
    "name": "Machine Learning"
  },
  {
    "id": 3,
    "name": "Inteligência Artificial",
  }, ...
]
```

Possíveis códigos de resposta:

- **200 OK** — Lista de tags retornada com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não é Responsável ou Administrador, logo não tem permissões para consultar os tags.

EP33 - Obter comentários de uma publicação

O utilizador autenticado com o papel **Responsável** ou **Administrador** pode visualizar os comentários de uma certa publicação

GET	/api/posts/{post_id}/comments
Headers	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
{
  "message": "Mensagem 1",
  "post_id": 101,
  "visible": false,
  "updated_at": "2025-11-07T17:10:00Z"
}

{
  "message": "Mensagem 2",
  "post_id": 101,
  "visible": false,
  "updated_at": "2025-11-010T18:10:00Z"
}
```

Possíveis códigos de resposta:

- **200 OK** — Comentários retornados com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — O utilizador não é Responsável ou Administrador, logo não tem permissões para consultar os comentários.

EP34 - Consultar detalhes de uma publicação

O utilizador autenticado pode consultar os detalhes de uma publicação específica.

Se a publicação tiver visibilidade pública (`is_visible = true`), qualquer utilizador autenticado pode consultar.

Se a publicação estiver oculta (`is_visible = false`), apenas o **autor** da mesma ou utilizadores com papel **Responsável** ou **Administrador** podem consultar os detalhes.

GET	/api/posts/{post_id}
Headers	Authorization: Bearer <token_colaborador_responsavel_admin>
Body	(vazio)

Resposta JSON (200 OK)

```
{
  "id": 101,
  "title": "Modelo CNN para deteção de tumores",
  "scientific_area": "Modelo Tumores",
  "summary": "Resumo completo da publicação gerado por IA e revisto pelo autor...",
  "is_visible": true,
  "file_url": "pi/posts/101/file",
  "author": {
    "id": 7,
    "name": "Joana B"
  },
  "average_rating": 4.5,
  "ratings_count": 10,
  "comments_count": 5,
  "tags": [
    { "id": 1, "name": "Deep Learning" },
    { "id": 4, "name": "Imagem Médica" }
  ],
  "created_at": "2025-10-30T14:22:05Z",
```

```
"updated_at": "2025-11-02T10:21:18Z"  
}
```

Possíveis códigos de resposta:

- **200 OK** — Detalhes da publicação recebidos com sucesso
- **401 Unauthorized** — Token inválido expirado, ou ausente
- **403 Forbidden** — A publicação é privada (**is_visible = false**) e o utilizador não é o autor, nem possui papel de Responsável ou Administrador.