

# Task Flow

Gerenciamento de Tarefas Simples, Rápido e Inteligente

Vandenberg Lucas | Nicollas Haniel | Myllena Medeiros

# Introdução e Objetivos



## O Problema

Desorganização e falta de clareza na gestão de projetos, afetando a produtividade da equipe e a visibilidade gerencial.



## Visão TaskFlow

Uma plataforma completa com UI intuitiva, insights poderosos para gerentes e controle robusto para administradores.



## Objetivo

Entregar valor essencial para as 3 personas (Administrador, Gerente, Colaborador) com foco nas funcionalidades "core".

# Metodologia BMAD (Adaptada)

## O que é BMAD?

---

Uma metodologia ágil focada em gerar valor de negócio através de um desenvolvimento iterativo, guiado por modelos e com colaboração contínua (Business-driven, Model-based, Agile, DevOps).

## Como aplicamos?

---

- ✓ Foco no valor de negócio desde o início.
- ✓ Desenvolvimento iterativo e incremental.
- ✓ Colaboração contínua e feedback (entre Agentes).
- ✓ Entrega de valor em ciclos curtos.

# Arquitetura da Solução (Visão Geral)



## Backend: API RESTful

- Spring Boot (Java 17)
- Web, JPA, Security, Validation
- Autenticação via JWT
- Lombok



## Database

- MySQL Database
- (Planejado: PostgreSQL)
- Validação de DTOs



## Frontend (Planejado)

- React.js
- Comunicação via API REST
- Visualização Kanban
- Dashboards

# Endpoints do projeto (Utilizando Swagger)

<b>comment-controller</b>			^
GET	/api/tasks/{taskId}/comments/{commentId}		🔒 ▼
PUT	/api/tasks/{taskId}/comments/{commentId}		🔒 ▼
DELETE	/api/tasks/{taskId}/comments/{commentId}		🔒 ▼
GET	/api/tasks/{taskId}/comments		🔒 ▼
POST	/api/tasks/{taskId}/comments		🔒 ▼
<b>task-controller</b>			^
PUT	/api/tasks/tasks/{taskId}		🔒 ▼
DELETE	/api/tasks/tasks/{taskId}		🔒 ▼
GET	/api/tasks/projects/{projectId}/tasks		🔒 ▼
POST	/api/tasks/projects/{projectId}/tasks		🔒 ▼
PATCH	/api/tasks/tasks/{taskId}/status		🔒 ▼
<b>project-controller</b>			^
GET	/api/projects		🔒 ▼
POST	/api/projects		🔒 ▼
POST	/api/projects/{projectId}/members		🔒 ▼
<b>auth-controller</b>			^
POST	/api/auth/register		🔒 ▼
POST	/api/auth/login		🔒 ▼
<b>admin-controller</b>			^
PATCH	/api/admin/users/{userId}/status		🔒 ▼
PATCH	/api/admin/users/{userId}/role		🔒 ▼
<b>user-controller</b>			^
GET	/api/users		🔒 ▼
GET	/api/users/{name}		🔒 ▼
<b>dashboard-controller</b>			^
GET	/api/dashboard/summary		🔒 ▼

# Funcionalidades: Autenticação e Perfis

## ✓ Implementado no MVP

---

- ✓ Cadastro de Usuários (AU-1)
- ✓ Login com e-mail e senha
- ✓ Níveis de Permissão (CA-2) (Admin, Manager, Collaborator)

## ✗ Fora do Escopo do MVP

---

- ✗ Login SSO (AU-2)
- ✗ Página de Perfil (AU-3)
- ✗ Recuperação de senha

## Funcionalidades: Autenticação e Perfis

Curl

```
curl -X 'POST' \
'http://localhost:8080/api/auth/register' \
-H 'accept: */*' \
-H 'Content-Type: application/json' \
-d '{
  "name": "Carlos Almeida",
  "email": "carlos@gmail.com",
  "password": "123456789"
}'
```

Request URL

**http://localhost:8080/api/auth/register**

### Server response

Code	Details
------	---------

201

### Response headers

```
cache-control: no-cache,no-store,max-age=0,must-revalidate
connection: keep-alive
content-length: 0
date: Fri,07 Nov 2025 16:50:40 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 0
```

## Responses

Code	Description
------	-------------

200 OK

## Curl

```
curl -X 'POST' \
  'http://localhost:8080/api/auth/login' \
  -H 'accept: */*' \
  -H 'Content-Type: application/json' \
  -d '{
    "email": "carlos@gmail.com",
    "password": "123456789"
  }'
```

Request URL

```
http://localhost:8080/api/auth/login
```

### Server response

Code	Details
------	---------

200

### References

Response body

[illegible]

### Response headers

```
cache-control: no-cache,no-store,max-age=0,must-revalidate
connection: keep-alive
content-type: application/json
date: Fri,07 Nov 2025 16:51:12 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 0
```

## Responses

Code	Description
------	-------------

200 OK

nks

to links

# Funcionalidades: Gestão de Projetos

## ✓ Implementado no MVP

---

- ✓ Criação de Projetos (GP-1 Parcial) (Apenas por Managers)
- ✓ Adição de Membros (GP-2 Parcial) (Apenas pelo Dono)
- ✓ Visualização de Projetos (GP-3)

## ✗ Fora do Escopo do MVP

---

- ✗ Renomear/Arquivar projetos (GP-1)
- ✗ Remover membros (GP-2)



# Funcionalidades: Gestão de Projetos (criando projeto)

**POST** /api/projects

**Parameters**

No parameters

**Request body** required

Edit Value Schema

```
{
  "name": "Projeto Accenture",
  "description": "Projeto para a apresentação final do treinamento"
}
```

201  
*Undocumented*

**Response body**

```
{
  "id": "5ee0f8a3-b168-40bb-a9f4-59967682dcc8",
  "name": "Projeto Accenture",
  "description": "Projeto para apresentação final do treinamento",
  "ownerId": "c477447b-bcb4-4a4f-acab-c33d8f5b350f"
}
```

**Response headers**

```
cache-control: no-cache,no-store,max-age=0,must-revalidate
connection: keep-alive
content-type: application/json
date: Fri,07 Nov 2025 16:57:04 GMT
expires: 0
keep-alive: timeout=60
pragma: no-cache
transfer-encoding: chunked
x-content-type-options: nosniff
x-frame-options: DENY
x-xss-protection: 0
```

**Responses**

Code	Description
200	OK

# Funcionalidades: Gestão de Projetos (adicionando membros ao projeto)

**POST** /api/projects/{projectId}/members

**Parameters**

Name	Description
<b>projectId</b> <small>required</small> string(\$uuid) (path)	5ee0f8a3-b168-40bb-a9f4-59967682dcc8

**Request body** required

Edit Value Schema

```
{
  "userId": "66f871fe-d47c-45c9-a083-83cae8c2fa62"
}
```

**Curl**

```
curl -X 'POST' \
  'http://localhost:8080/api/projects/5ee0f8a3-b168-40bb-a9f4-59967682dcc8/members' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiJ9.eyJyb2xlIjoiTUF0QUdFUlIsInVzZXJJZCI6ImM0N' \
  -H 'Content-Type: application/json' \
  -d '{
    "userId": "66f871fe-d47c-45c9-a083-83cae8c2fa62"
  }'
```

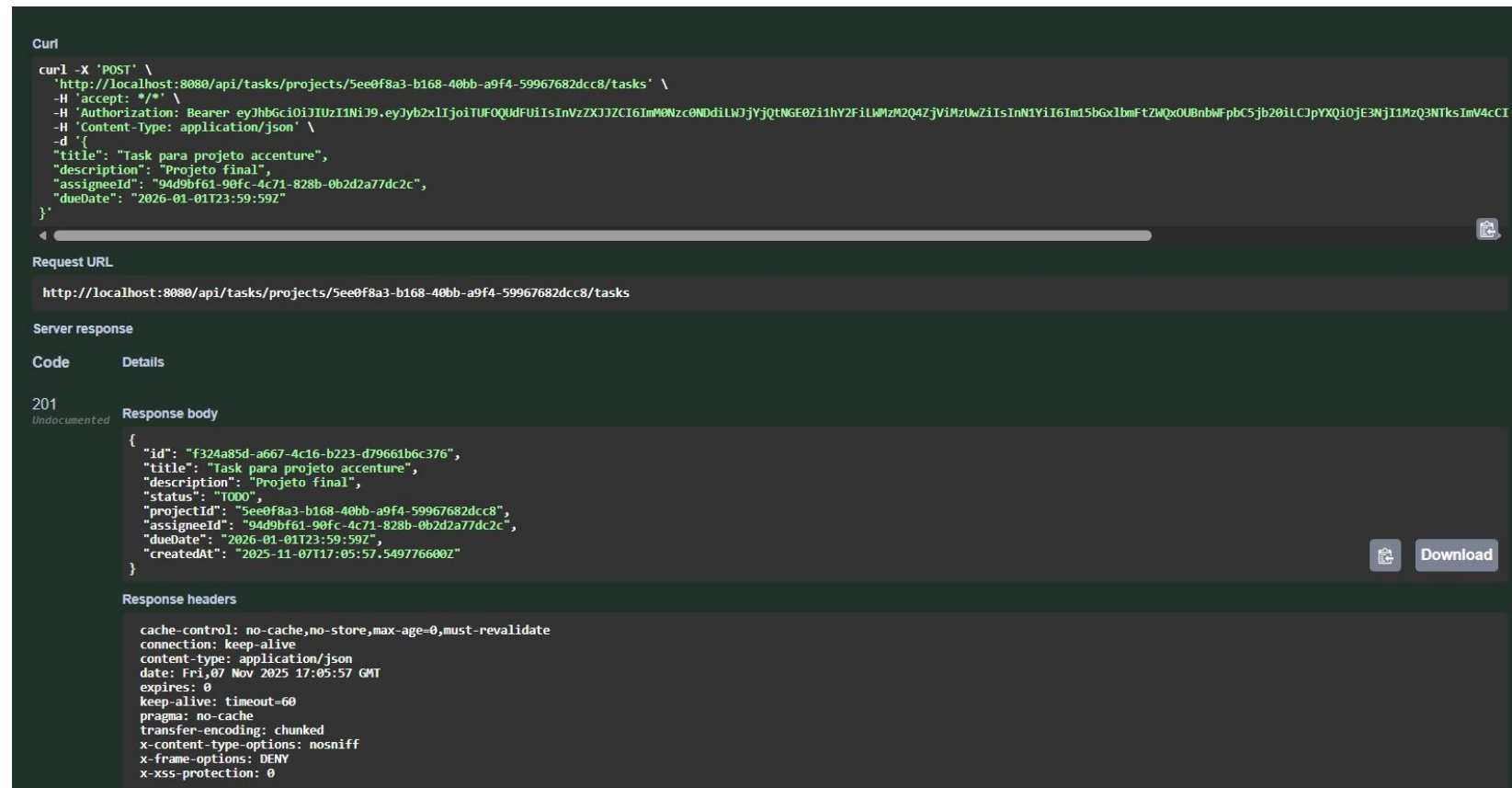
**Request URL**

```
http://localhost:8080/api/projects/5ee0f8a3-b168-40bb-a9f4-59967682dcc8/members
```

# Funcionalidades: Gestão de Tarefas (Core)

- ✓ Criação de Tarefas (GT-1): Título, descrição, responsável e data de entrega.
- ✓ Atualização de Status (GT-3): Lógica de backend para quadro Kanban (TODO, IN\_PROGRESS, DONE).
- ✓ Edição de Tarefas (GT-4): Atualização de todos os campos principais.
- ✓ Exclusão de Tarefas (GT-5): Endpoint de deleção implementado.
- ✓ Filtragem de Tarefas (GT-4): API para filtro por responsável e status.

## Funcionalidades: Gestão de Projetos (adicionando task a um projeto)



## Funcionalidades: Gestão de Tarefas (task por projeto)

GET

/api/tasks/projects/{projectId}/tasks

Parameters

Cancel

Name	Description
<b>projectId</b> <small>* required</small> string(\$uuid) (path)	5ee0f8a3-b168-40bb-a9f4-59967682dcc8
<b>status</b> string (query)	status
<b>assigneeId</b> string(\$uuid) (query)	assigneeId

Execute

Clear

Responses

[illegible]

## Funcionalidades: Gestão de Tarefas (Atualizando status)

PATCH /api/tasks/tasks/{taskId}/status	
Parameters	
Name	Description
taskId * required string(\$uuid) (path)	f324a85d-a667-4c16-b223-d79661b6c376
Request body required	
Edit Value	Schema
{ "status": "DONE" }	

[illegible]



## Funcionalidades: Gestão de Tarefas (Comentando Task)

<div>POST</div> <div>/api/tasks/{taskId}/comments</div>	
<div>Parameters</div>	
<div>Name</div>	<div>Description</div>
<div><div>taskId</div><div><div><div>* required</div></div></div><div>string(\$uuid)</div><div>(path)</div></div>	<div>f324a85d-a667-4c16-b223-d79661b6c376</div>
<div>Request body</div> <div>required</div>	
<div>Edit Value</div>	<div>Schema</div>
<div>{   "content": "Comentando na task do projeto accenture" }</div>	

```
Curl

curl -X 'POST' \
  'http://localhost:8080/api/tasks/f324a85d-a667-4c16-b223-d79661b6c376/comments' \
  -H 'accept: */*' \
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiJ9.eyJyb2xlIjoiTUF0QUdFUlIsInVzZXkiOiJlM0Nzc0NDdlWjJyYjQ0NGE0Zi1hY2FiZWwzM2Q4ZjVhZlwZiIsInN1YiI6Im15b6x1bmFtZWQxOUBnbWpC5jb20iLCJpYXQiOiJlM0NjI1MzQ3NTksImV4cCI6MTUzOTU0OTU0fQ.eyJ0eXAiOiJKV1QiLCJhbGciOiJIUzI1NiJ9' \
  -H 'Content-Type: application/json' \
  -d '{
    "content": "Comentando na task do projeto accenture"
  }'
```

Request URL

http://localhost:8080/api/tasks/f324a85d-a667-4c16-b223-d79661b6c376/comments

Server response

Code	Details
201	Response body

# Funcionalidades: Colaboração e Dashboard

## ✓ Implementado no MVP

---

- ✓ Comentários em Tarefas (CL-1)
- ✓ Dashboard do Usuário (DR-1 Parcial) (Resumo de tarefas)
- ✓ Dashboard do Projeto (DR-2 Parcial) (Resumo de tarefas)

## ✗ Fora do Escopo do MVP

---

- ✗ Anexar arquivos (CL-2)
- ✗ Notificações de menção (CL-3)
- ✗ Exportar Dashboard (DR-3)



# Funcionalidades: Configurações (Admin)

## ✔ Implementado no MVP

---

- ✔ Gerenciamento de Usuários (CA-1 Parcial) (Listar, desativar, promover)

## ✘ Fora do Escopo do MVP

---

- ✘ Convidar líder de projeto (CA-1)
- ✘ Logs de auditoria (CA-3)

## Funcionalidades: Configurações Admin (promovendo um colaborador)

PATCH

/api/admin/users/{userId}/role

Parameters

Name	Description
<b>userId</b> <small>* required</small>	
string(\$uuid) (path)	369c53e7-53dd-47bf-ac61-fbec0a30a37c

Request body required

Edit Value

Schema

```
{
  "role": "MANAGER"
}
```

The screenshot displays a REST client interface with the following sections:

- Curl**: A text area containing a curl command to patch a user's role.

```
curl -X 'PATCH' \  
  'http://localhost:8080/api/admin/users/369c53e7-53dd-47bf-ac61-fbec0a30a37c/role' \  
  -H 'accept: */*' \  
  -H 'Authorization: Bearer eyJhbGciOiJIUzI1NiJ9.eyJyb2xlIjoiaURNSU4iLCJ1c2VybmQoIjI2ZmJkYzRhOCC0b2ZmLnRmZG9tIjY0OzYhQ1ZGIzMTIwNGQoIjI3ZGIpbk80YXNrZmxvd55jb20iLCJpYXQiOiJlZ3NjI1MzQ3WDgsImV4cCI6MTc2MjE5fQ.' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "role": "MANAGER"  
  }'
```
- Request URL**: A text area showing the URL: `http://localhost:8080/api/admin/users/369c53e7-53dd-47bf-ac61-fbec0a30a37c/role`
- Server response**: A section for the response, currently empty.
- Code**: A dropdown menu showing the status code `200`.
- Details**: A section for response details, currently empty.
- Response headers**: A section for response headers, currently empty.

# Próximos Passos e Conclusão

## Próximos Passos

---

- Desenvolvimento do Frontend React (Kanban, Dashboards).
- Implementar funcionalidades restantes do MVP (SSO, Anexos, Notificações, Logs).
- Refinamento de UX/UI com base no feedback.

## Conclusão

---

O TaskFlow estabelece uma base sólida para o gerenciamento de tarefas.

Entregamos o valor essencial para as personas-chave e preparamos o terreno para futuras expansões com uma arquitetura de API robusta e testável.

# Obrigado!

Agradecemos a atenção de todos!