



DOCUMENTAÇÃO

Desafio de Desenvolvimento Full Stack

Candidato: Brendo Gomes Santana



DESAFIO DE DESENVOLVIMENTO FULL STACK

Sumário

1. Pré-requisitos	2
2. Instalar as dependências	2
3. Executar a Aplicação	2
4. ROTAS API	2
4.1. POST - /tarefas	3
4.2. GET - /tarefas	4
4.3. GET - /tarefas/<id>	4
4.4. PUT - /tarefas/<id>	5
4.5. DELETE - /tarefas/<id>	5



DESAFIO DE DESENVOLVIMENTO FULL STACK

1. Pré-requisitos

Antes de começar, certifique-se de ter os seguintes itens instalados:

python - <https://www.python.org/downloads/>

2. Instalar as dependências

Instale as dependências necessárias utilizando o pip:

- ✓ pip install sqlite3
- ✓ pip install flask
- ✓ pip install flask_cors

3. Executar a Aplicação

Agora você pode executar a aplicação.

Abra o terminal, navegue até o diretório do projeto e execute o comando abaixo

(figura 1):

```
cd caminho/para/sua/pasta/do/projeto  
python app.py
```

Figura 1: terminal

Substitua caminho/para/sua/pasta/do/projeto pelo caminho real onde o projeto está salvo no seu computador.

Isso irá iniciar o servidor localmente. A aplicação estará disponível em (figura 2):

```
cd caminho/para/sua/pasta/do/projeto  
python app.py
```

Figura 2: url

4. ROTAS API

Esta API foi desenvolvida para gerenciar um sistema de tarefas, permitindo a criação, consulta, atualização e exclusão de registros no banco de dados. Abaixo, você encontrará a descrição completa de cada rota disponível, o método HTTP utilizado, os dados que devem ser enviados (quando necessário), e o tipo de resposta esperada.



DESAFIO DE DESENVOLVIMENTO FULL STACK

4.1. POST - /tarefas

Esta rota é usada para cadastrar uma nova tarefa no sistema. O cliente deve enviar um corpo JSON (figura 3) contendo as informações necessárias, como nome e comentário.

Importante: O campo `created_at` será adicionado automaticamente pelo sistema com a data e hora da criação.

```
{
  "nome": "Exemplo de tarefa",
  "comentario": "Descrição ou observação da tarefa"
}
```

Figura 3: Corpo da Requisição (JSON)

Caso a requisição seja bem-sucedida, ela retornará uma resposta (Figura 4).

```
{
  "id": 1,
  "nome": "Exemplo de tarefa",
  "comentario": "Descrição ou observação da tarefa",
  "created_at": "2025-05-14 14:35:00"
}
```

Figura 4: Resposta esperada (JSON) – Criado com sucesso

Erros Possíveis:

400 - Campos "nome" e "comentario" são obrigatórios.



DESAFIO DE DESENVOLVIMENTO FULL STACK

4.2. GET - /tarefas

Retorna uma lista com todas as tarefas cadastradas em Ordem ASC (Figura 5).

```
[
  {
    "id": 1,
    "nome": "Tarefa 1",
    "comentario": "Comentário da tarefa 1",
    "created_at": "2025-05-14 14:35:00"
  },
  {
    "id": 2,
    "nome": "Tarefa 2",
    "comentario": "Comentário da tarefa 2",
    "created_at": "2025-05-14 15:10:42"
  }
]
```

Figura 5: Corpo da Requisição (JSON) – lista de tarefas

4.3. GET - /tarefas/<id>

Retorna os detalhes de uma tarefa com base no ID informado.

Caso a requisição seja bem-sucedida, ela retornará uma resposta (Figura 6).

```
{
  "id": 1,
  "nome": "Exemplo de tarefa",
  "comentario": "Descrição ou observação da tarefa",
  "created_at": "2025-05-14 14:35:00"
}
```

Figura 6: Resposta esperada (JSON) – Buscar uma tarefa específica

Erros Possíveis:

404 – Tarefa não encontrada



DESAFIO DE DESENVOLVIMENTO FULL STACK

4.4. PUT - /tarefas/<id>

Atualiza os dados de uma tarefa existente. O ID é informado na URL, e o corpo deve conter os novos dados (figura 7).

```
{  
  "nome": "Exemplo de tarefa",  
  "comentario": "Descrição ou observação da tarefa"  
}
```

Figura 7: Corpo da Requisição (JSON) – Atualizar tarefa

Caso a requisição seja bem-sucedida, ela retornará uma resposta (Figura 8).

```
{  
  "message": "Tarefa atualizada com sucesso"  
}
```

Figura 8: Resposta esperada (JSON) – Atualizar com sucesso

Erros Possíveis:

400 - Campos "nome" e "comentario" são obrigatórios.

4.5. DELETE - /tarefas/<id>

Remove uma tarefa do sistema com base no ID informado.

Caso a requisição seja bem-sucedida, ela retornará uma resposta (Figura 9).

```
{  
  "message": "Tarefa deletada com sucesso"  
}
```

Figura 9: Resposta esperada (JSON) – Deletado com sucesso