

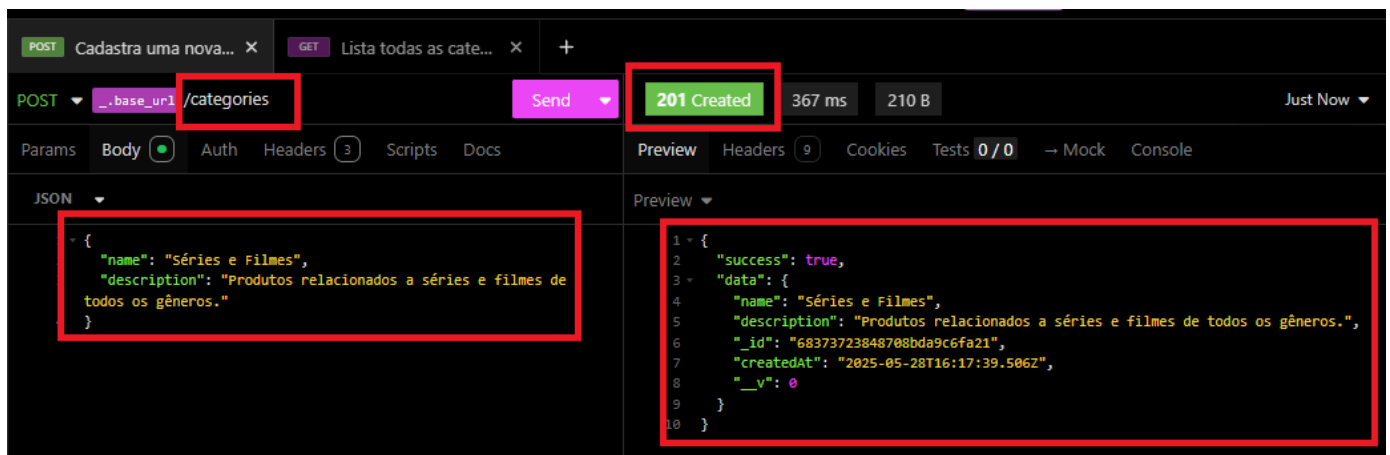
Categorias

User Story 11: Cadastro de uma nova categoria

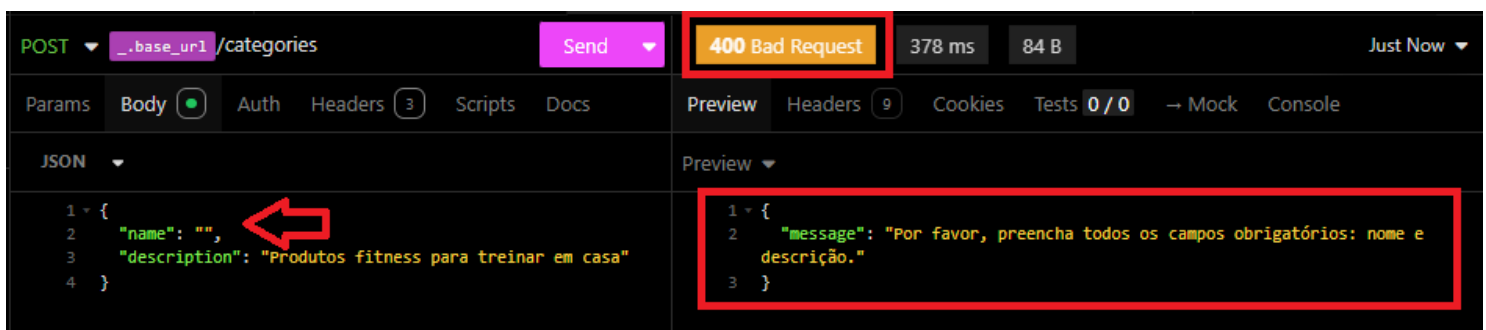
Como administrador, eu quero cadastrar uma nova categoria de produto **para que** os produtos possam ser organizados.

- **Critérios de Aceitação:**

- Ao realizar uma requisição **POST** para **/categorias** com um payload JSON válido (contendo o **nome** da categoria), a API **deve** retornar o status code **201 Created**.
- A resposta **deve** conter o objeto da categoria criada, incluindo um **id** gerado pela API.
- Ao tentar cadastrar uma categoria sem o campo **nome** no payload, a API **deve** retornar o status code **400 Bad Request** e uma mensagem indicando a ausência do campo: **{"message": "Por favor, preencha todos os campos obrigatórios: nome e descrição."}**.



The screenshot shows a REST client interface with a POST request to `_.base_url/categories`. The request body is a valid JSON object: `{ "name": "Séries e Filmes", "description": "Produtos relacionados a séries e filmes de todos os gêneros." }`. The response status is **201 Created** with a response body: `{ "success": true, "data": { "name": "Séries e Filmes", "description": "Produtos relacionados a séries e filmes de todos os gêneros.", "_id": "68373723848708bda9c6fa21", "createdAt": "2025-05-28T16:17:39.506z", "__v": 0 } } }`. Red boxes highlight the endpoint, the status code, and the request/response bodies.

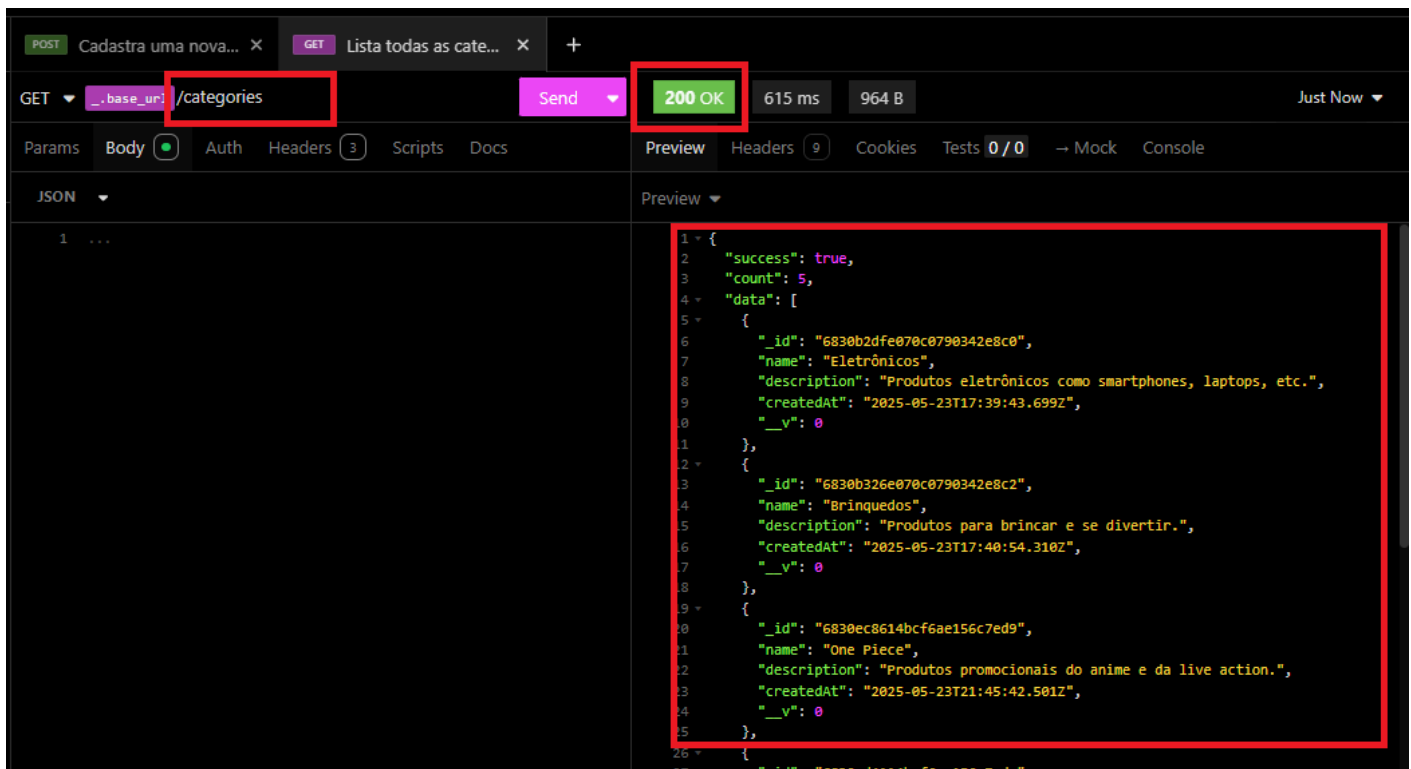


The screenshot shows a REST client interface with a POST request to `_.base_url/categories`. The request body is an invalid JSON object: `{ "name": "", "description": "Produtos fitness para treinar em casa" }`. A red arrow points to the empty `"name": ""` field. The response status is **400 Bad Request** with a response body: `{ "message": "Por favor, preencha todos os campos obrigatórios: nome e descrição." }`. Red boxes highlight the status code and the response body.

User Story 12: Listagem de todas as categorias

Como usuário, eu quero listar todas as categorias de produtos para que eu possa navegar pelos diferentes tipos de produtos.

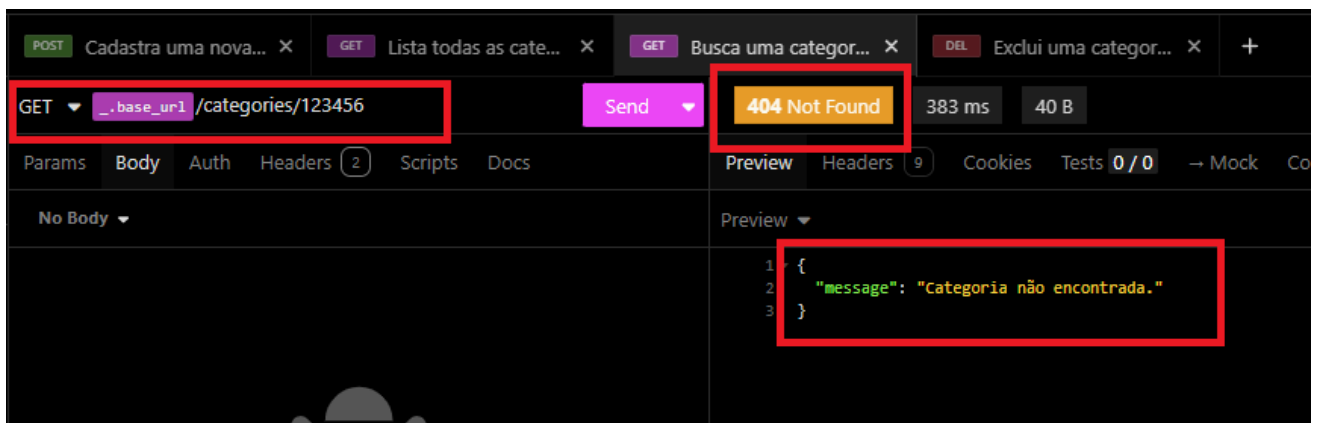
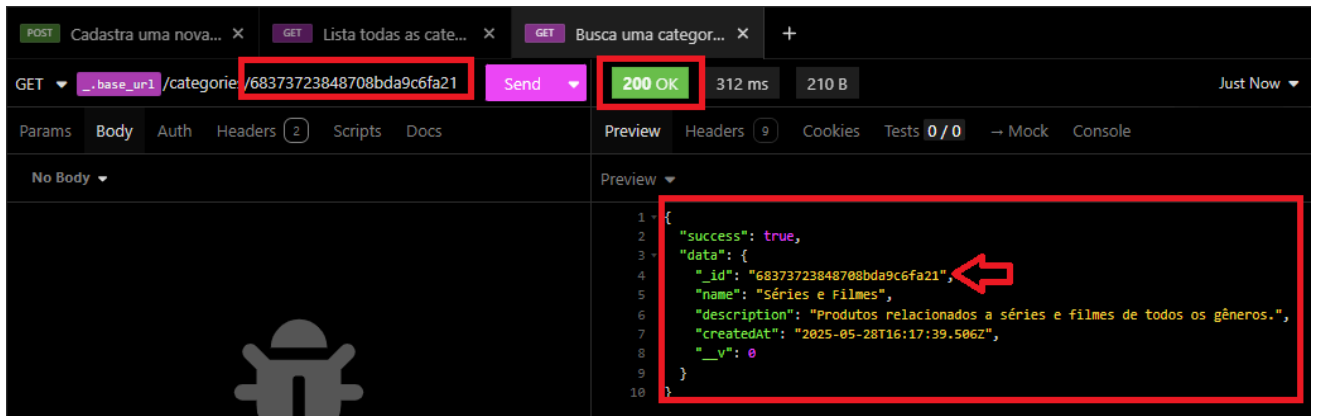
- **Critérios de Aceitação:**
 - Ao realizar uma requisição **GET** para `/categorias`, a API **deve** retornar o status code **200 OK**.
 - A resposta **deve** conter um array JSON com a lista de categorias.
 - Cada objeto de categoria no array **deve** conter os campos relevantes (a serem definidos ao testar a API, como `id` e `nome`).



User Story 13: Busca de categoria por ID

Como usuário, eu quero buscar uma categoria específica pelo seu ID para que eu possa obter detalhes sobre ela.

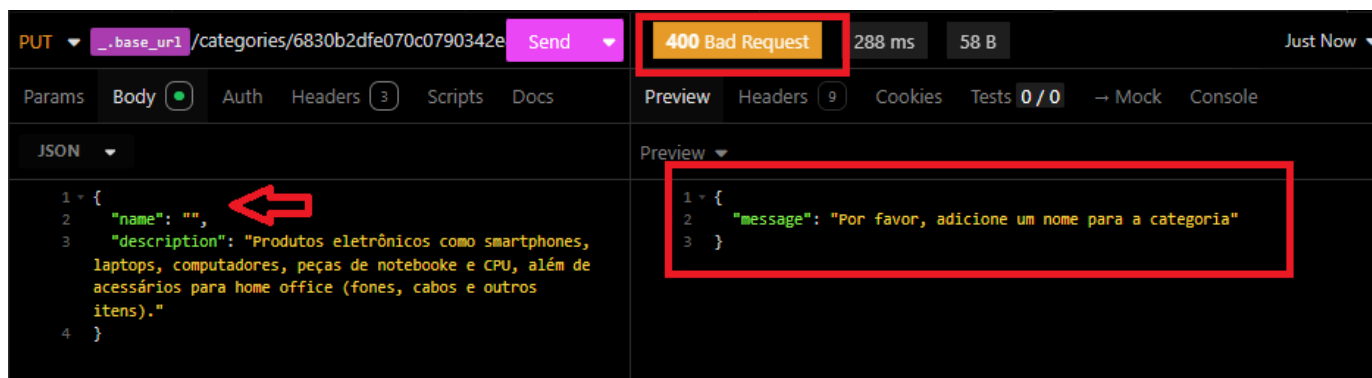
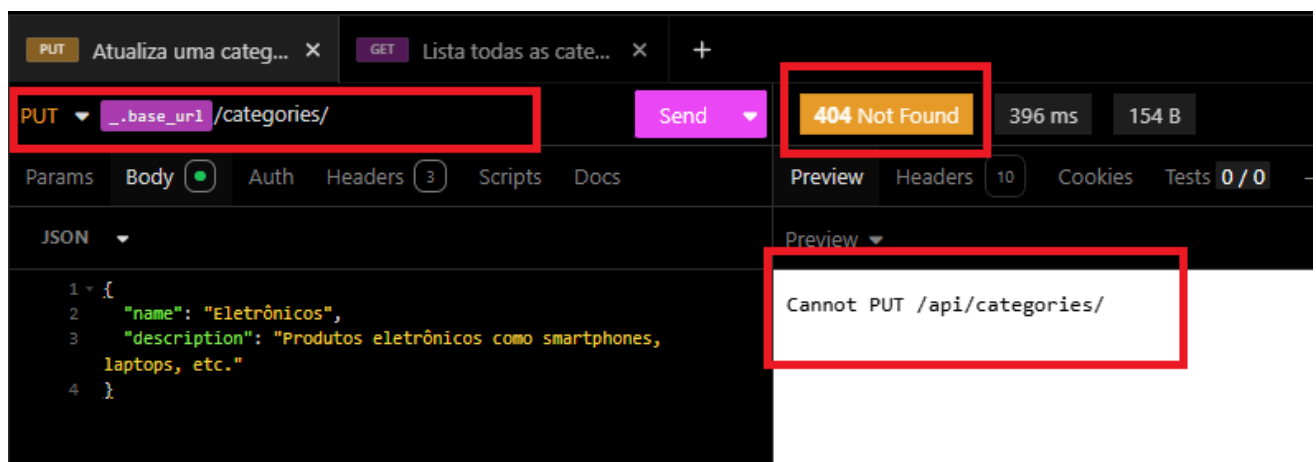
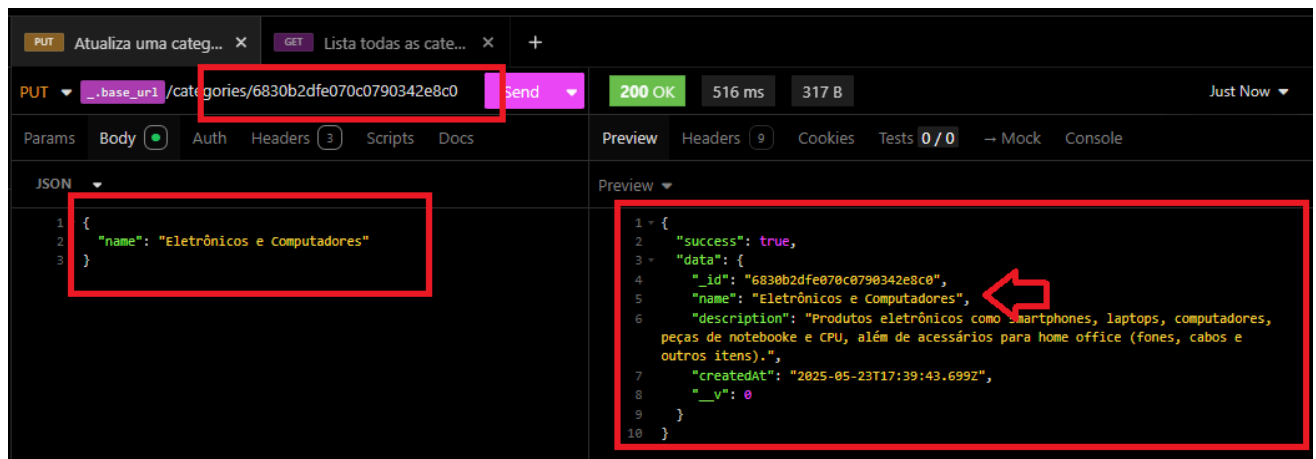
- **Critérios de Aceitação:**
 - **Dado** um ID de categoria existente, **ao** realizar uma requisição **GET** para `/categorias/{id}`, a API **deve** retornar o status code **200 OK**.
 - A resposta **deve** conter um objeto JSON com os detalhes da categoria solicitada.
 - **Dado** um ID de categoria **inexistente**, **ao** realizar uma requisição **GET** para `/categorias/{id}`, a API **deve** retornar o status code **404 Not Found**.
 - Para um ID inexistente, o corpo da resposta **deve** conter uma mensagem de erro informativa (a ser verificada ao testar, ex: `{"message": "Categoria não encontrada"}`).



User Story 14: Atualização de uma categoria existente

Como administrador, eu quero atualizar o nome de uma categoria existente para que a descrição esteja sempre atualizada.

- **Critérios de Aceitação:**
 - **Dado** um ID de categoria existente, **ao** realizar uma requisição **PUT** para `/categorias/{id}` com um payload JSON válido (contendo o campo `nome`), a API **deve** retornar o status code **200 OK**.
 - A resposta **deve** conter o objeto da categoria com o nome atualizado.
 - **Dado** um ID de categoria **inexistente**, **ao** realizar uma requisição **PUT** para `/categorias/{id}`, a API **deve** retornar o status code **404 Not Found**.
 - Ao tentar atualizar uma categoria sem o campo `nome` no payload, a API **deve** retornar o status code **400 Bad Request** e uma mensagem indicando a ausência do campo "nome" `{"message": "Por favor, adicione um nome para a categoria."}`



User Story 15: Exclusão de uma categoria

Como administrador, eu quero excluir uma categoria existente para que ela não seja mais utilizada para classificar produtos.

- **Critérios de Aceitação:**
 - Dado um ID de categoria existente, ao realizar uma requisição DELETE para `/categorias/{id}`, a API deve retornar o status code 200 OK com uma mensagem de sucesso.

- O corpo da resposta **deve** conter uma mensagem de sucesso `{"message": "Categoria removida com sucesso"}`.
- **Dado** um ID de categoria **inexistente**, **ao** realizar uma requisição **DELETE** para `/categorias/{id}`, a API **deve** retornar o status code **404 Not Found**.

A screenshot of a REST client interface. The top bar shows a **DELETE** request to `_.base_url/categorias/6830ec8614bcf6ae156`. The response status is **200 OK** with a response time of 56 ms and a body size of 60 B. The response body is displayed in the preview pane as a JSON object:

```
1 {
2   "success": true,
3   "message": "Categoria removida com sucesso."
4 }
```

A screenshot of a REST client interface. The top bar shows a **DELETE** request to `_.base_url/categorias/`. The response status is **404 Not Found** with a response time of 65 ms and a body size of 157 B. The response body is displayed in the preview pane as a plain text message: `Cannot DELETE /api/categories/`.