

Report de issues encontrados na API da PetStore

→ **RAFAEL SOARES**

[BUG-1] - PUT /username/{username} - É possível editar um usuário sem autenticação, apesar da documentação exigir.

Descrição:

Segundo a documentação Swagger da API, o endpoint /user/{username} “só pode ser feito por um usuário logado”. No entanto, ao testar a API, foi possível atualizar qualquer usuário apenas fornecendo o nome na URL e um corpo JSON válido, sem enviar qualquer forma de autenticação. Ainda está ausente o erro 401 Unauthorized na documentação Swagger.

Passos Para Reproduzir:

1. Acesse o endpoint PUT: <https://petstore3.swagger.io/api/v3/user/{username}> com um username válido.
2. Envie uma requisição PUT com um body válido.

RESULTADO ESPERADO:

A API deveria retornar o erro 401 Unauthorized se o usuário não estiver autenticado.

RESULTADO ATUAL:

A API aceita a atualização mesmo sem autenticação, contrariando o que está escrito na documentação.

[BUG-2] - PUT /username/{username} - Campos ausentes no corpo da requisição causam perda de dados do usuário

Descrição:

Ao atualizar um usuário com o PUT /user/{username}, se um ou mais campos estiverem ausentes no corpo da requisição, esses campos são sobrescritos com valores vazios, o que leva a perda de dados.

Dados do usuário testado:

```
1 {  
2   "id": 10,  
3   "username": "theUser",  
4   "firstName": "John",  
5   "lastName": "James",  
6   "email": "john@email.com",  
7   "password": "12345",  
8   "phone": "12345",  
9   "userStatus": 1  
10 }
```

Passos Para Reproduzir:

1. Acesse o endpoint PUT: <https://petstore3.swagger.io/api/v3/user/{username}> com um username válido.
2. Envie uma requisição PUT com um body com campos ausentes de dados.

Requisição PUT realizada onde o campo Email não foi enviado.

```
1 {  
2   "id": 10,  
3   "username": "theUser",  
4   "lastName": "James after the test",  
5   "nt": "James",  
6   "password": "123445",  
7   "userStatus": 1  
8 }
```

RESULTADO ESPERADO:

A API deveria retornar um erro ou rejeitar a requisição, informando que campos obrigatórios estão faltando. Alternativamente, poderia ser mantido os dados não preenchidos.

RESULTADO ATUAL:

A API retorna o status 200 e os campos ausentes são totalmente apagados, resultando em perda de dados do usuário.

Fazendo GET do usuário é possível observar a ausência do Email.

```
1 {  
2   "id": 10,  
3   "username": "theUser",  
4   "lastName": "James after the test",  
5   "password": "123445",  
6   "userStatus": 1  
7 }
```

[BUG-3] - GET /pet/findByTags - Retorna status 200 OK para tags inválidas ou inexistentes.

Descrição:

Ao enviar uma requisição com tags inexistentes ou inválidas, a API retorna 200 OK com body vazio, quando o esperado seria um status 400 Bad Request ou 404 Not Found para indicar que os dados solicitados não existem.

Passos Para Reproduzir:

1. Acesse o endpoint: GET https://petstore3.swagger.io/api/v3/pet/findByTags?tags=tag_inexistente
2. Insira uma tag inexistente ou inválida.

RESULTADO ESPERADO:

Status Code: 400 Bad Request (para tags inválidas) ou 404 Not Found (se a tag não existir).
Body com mensagem explicativa (ex: {"error": "Tag não encontrada"}).

RESULTADO ATUAL:

A API retorna o status 200 indicando que a requisição foi um sucesso e retorna o seguinte body:

