

1 Testes do Microserviço de Eventos

Para validar o funcionamento do microserviço de eventos, foram realizados testes utilizando o Postman, sempre autenticados com um token JWT. Abaixo são apresentados os passos realizados e os resultados obtidos.

1.1 Criação de Evento

Foi criado um evento utilizando a rota `/api/events`. A resposta confirmou que o evento foi criado com sucesso.

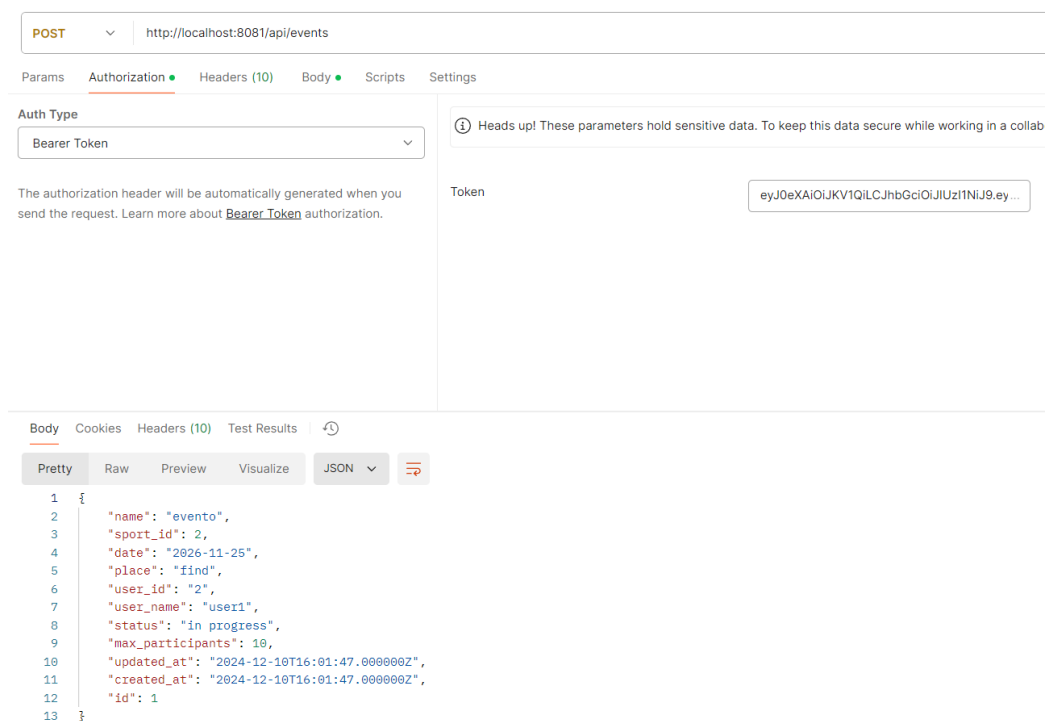


Figura 1: Teste de Criação de Evento

1.2 Atualização de Evento

O evento criado anteriormente foi atualizado utilizando a rota `/api/events/id`. A resposta confirmou a atualização com sucesso.

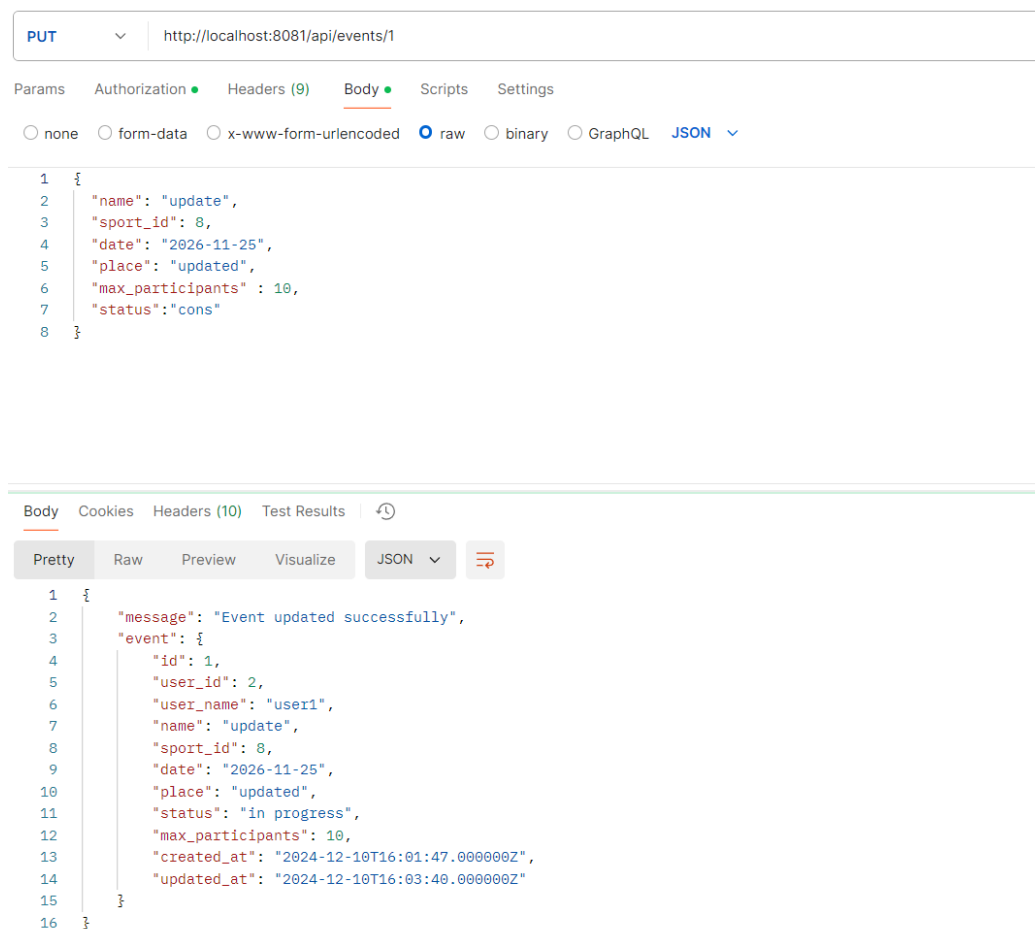


Figura 2: Teste de Atualização de Evento

1.3 Participação em Evento

Um segundo utilizador juntou-se ao evento criado anteriormente utilizando a rota `/api/events/id/join`. A resposta confirmou a participação no evento.

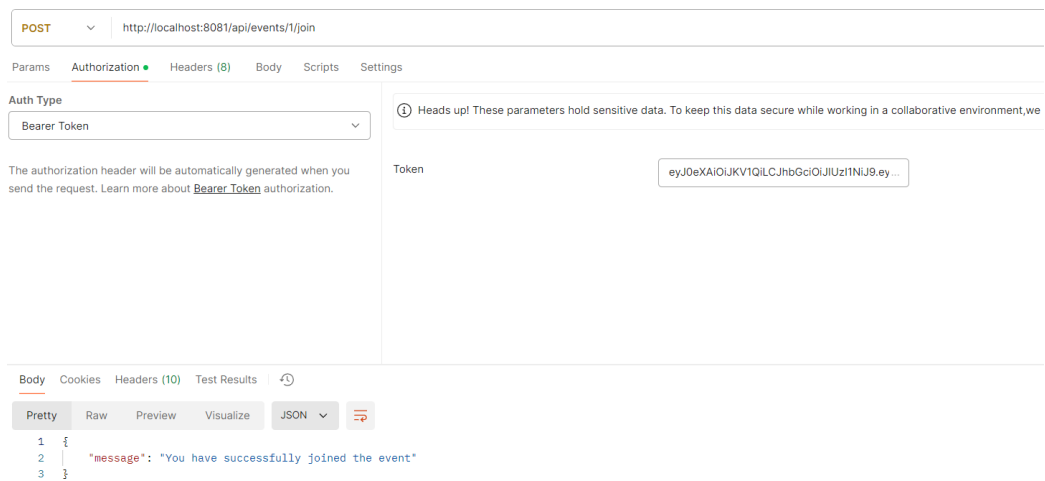


Figura 3: Teste de Participação em Evento

1.4 Listagem de Participantes do Evento

Foi utilizada a rota `/api/events/id/participants` para listar os participantes do evento. A resposta mostrou os utilizadores que estavam associados ao evento.

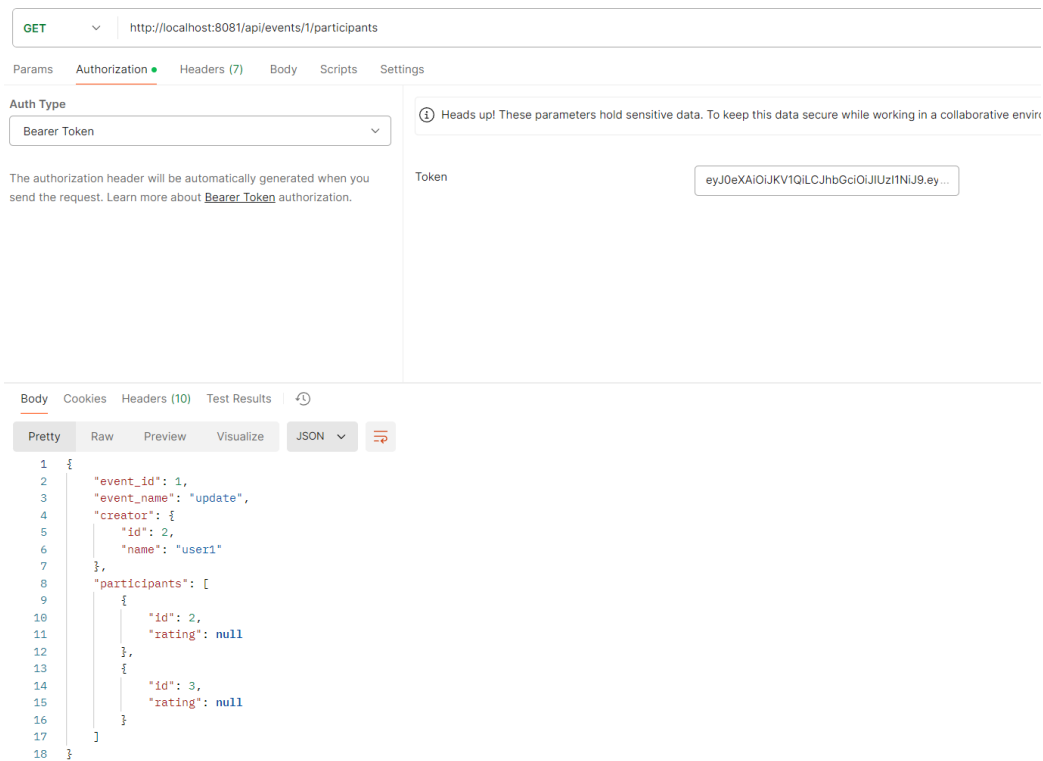


Figura 4: Teste de Listagem de Participantes do Evento

1.5 Eventos do Utilizador

O segundo utilizador criou outro evento e, em seguida, utilizou a rota `/api/events/mine` para listar os eventos aos quais pertencia. A resposta mostrou os dois eventos associados a este utilizador.

GET http://localhost:8081/api/events/mine

Params Authorization Headers (7) Body Scripts Settings

Auth Type: Bearer Token

Heads up! These

Body Cookies Headers (10) Test Results

Pretty Raw Preview Visualize JSON

```
3      "id": 1,
4      "name": "update",
5      "sport": "Andebol",
6      "date": "2026-11-25",
7      "place": "updated",
8      "status": "in progress",
9      "max_participants": 10,
10     "creator": {
11       "id": 2,
12       "name": "user1"
13     },
14     "participants": [
15       {
16         "id": 2,
17         "name": "user1",
18         "rating": null
19       },
20       {
21         "id": 3,
22         "name": "user2",
23         "rating": null
24       }
25     ],
26   },
27   {
28     "id": 2,
29     "name": "evento",
30     "sport": "Futsal",
31     "date": "2026-11-25",
32     "place": "find",
33     "status": "in progress",
34     "max_participants": 10,
35     "creator": {
36       "id": 3,
37       "name": "user2"
38     },
39     "participants": [
40       {
41         "id": 3,
```

Figura 5: Listagem de Eventos do Utilizador Após Criar Novo Evento

1.6 Eliminação de Evento

Um dos eventos foi eliminado, pelo criador, utilizando a rota `/api/events/id`. A resposta confirmou que o evento foi eliminado com sucesso.

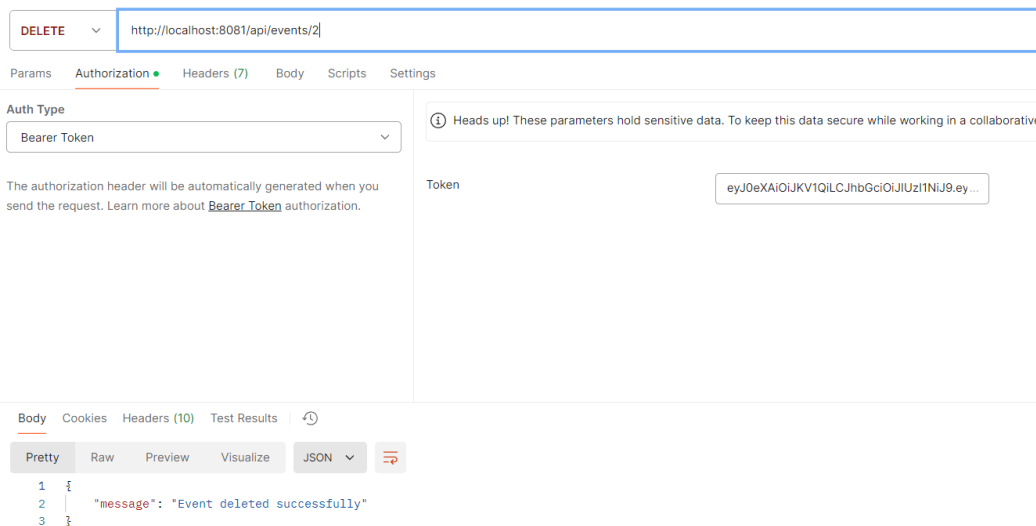


Figura 6: Teste de Eliminação de Evento

1.7 Verificação dos Eventos Após Eliminação

Foi novamente utilizada a rota `/api/events/mine` para listar os eventos do utilizador. Como o evento eliminado não estava mais presente, a resposta mostrou apenas o evento restante.

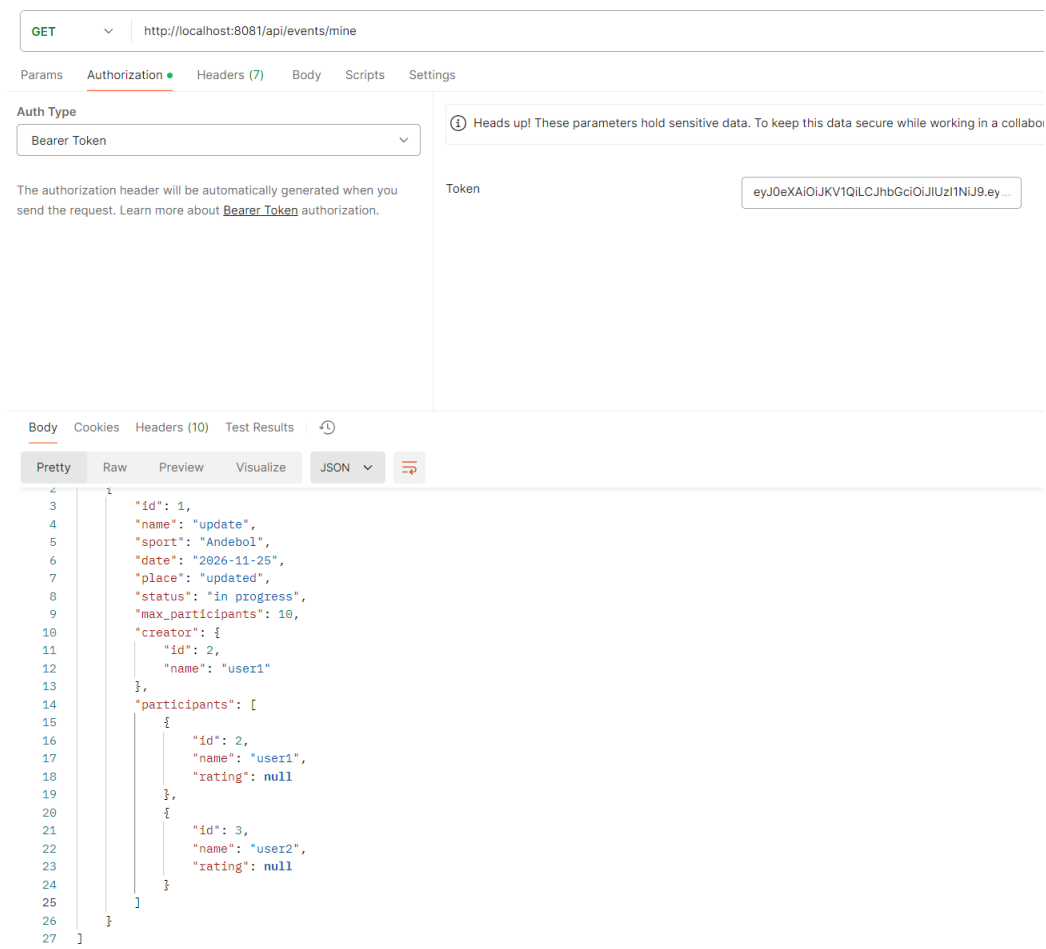


Figura 7: Listagem de Eventos do Utilizador Após Eliminação de Evento

1.8 Saída de Evento

O segundo utilizador saiu do evento restante utilizando a rota `/api/events/id/leave`. A resposta confirmou a saída com sucesso.

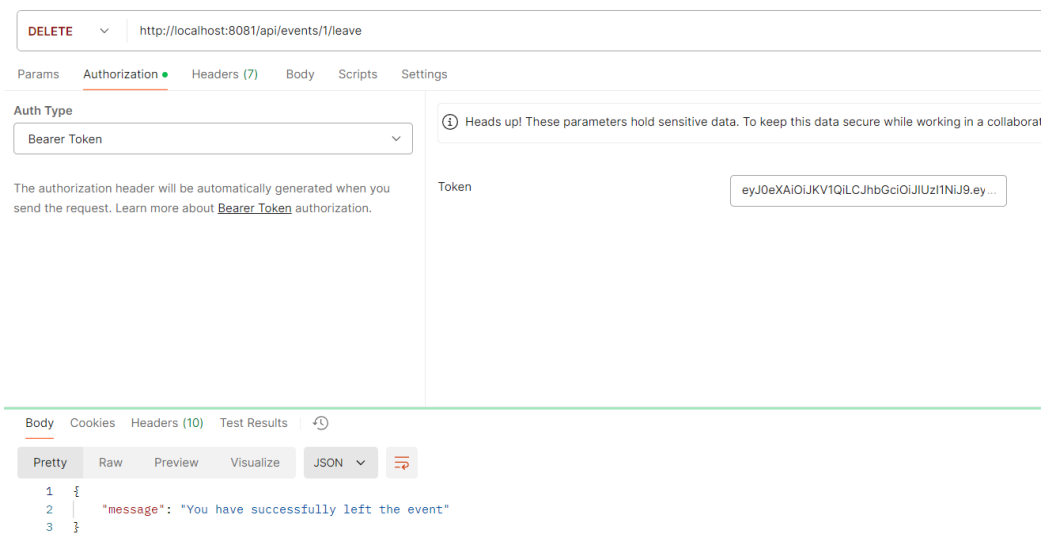


Figura 8: Teste de Saída de Evento

1.9 Verificação de Eventos Após Saída

Por fim, foi novamente utilizada a rota `/api/events/mine` para verificar os eventos do utilizador. Como não havia mais eventos associados, a resposta foi vazia.

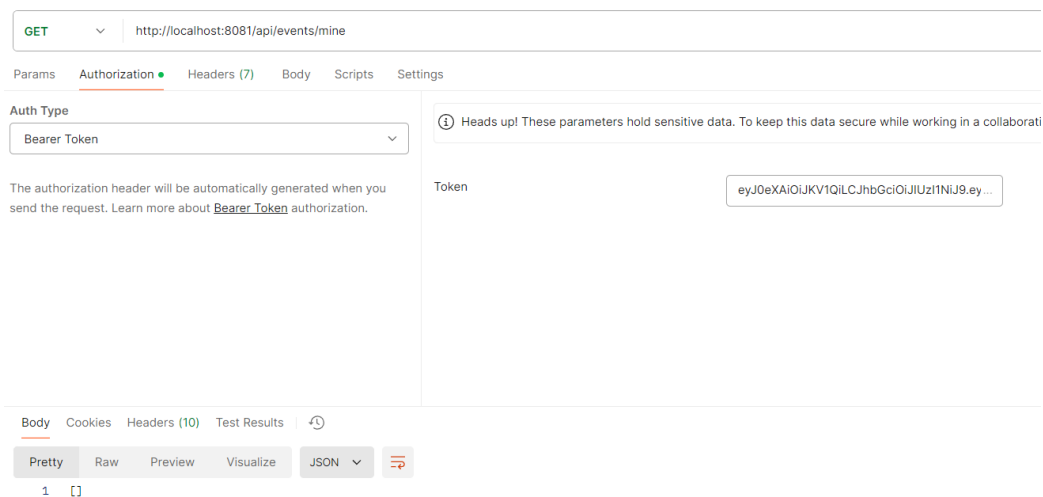


Figura 9: Listagem de Eventos Após Saída

1.10 Procura de Eventos

Foram realizados testes para procurar eventos com base em diferentes parâmetros utilizando a rota `/api/events/search`.

1.10.1 Procura por Nome

Foi utilizado o parâmetro `name` para procurar eventos com o nome especificado. A resposta confirmou o sucesso da operação.

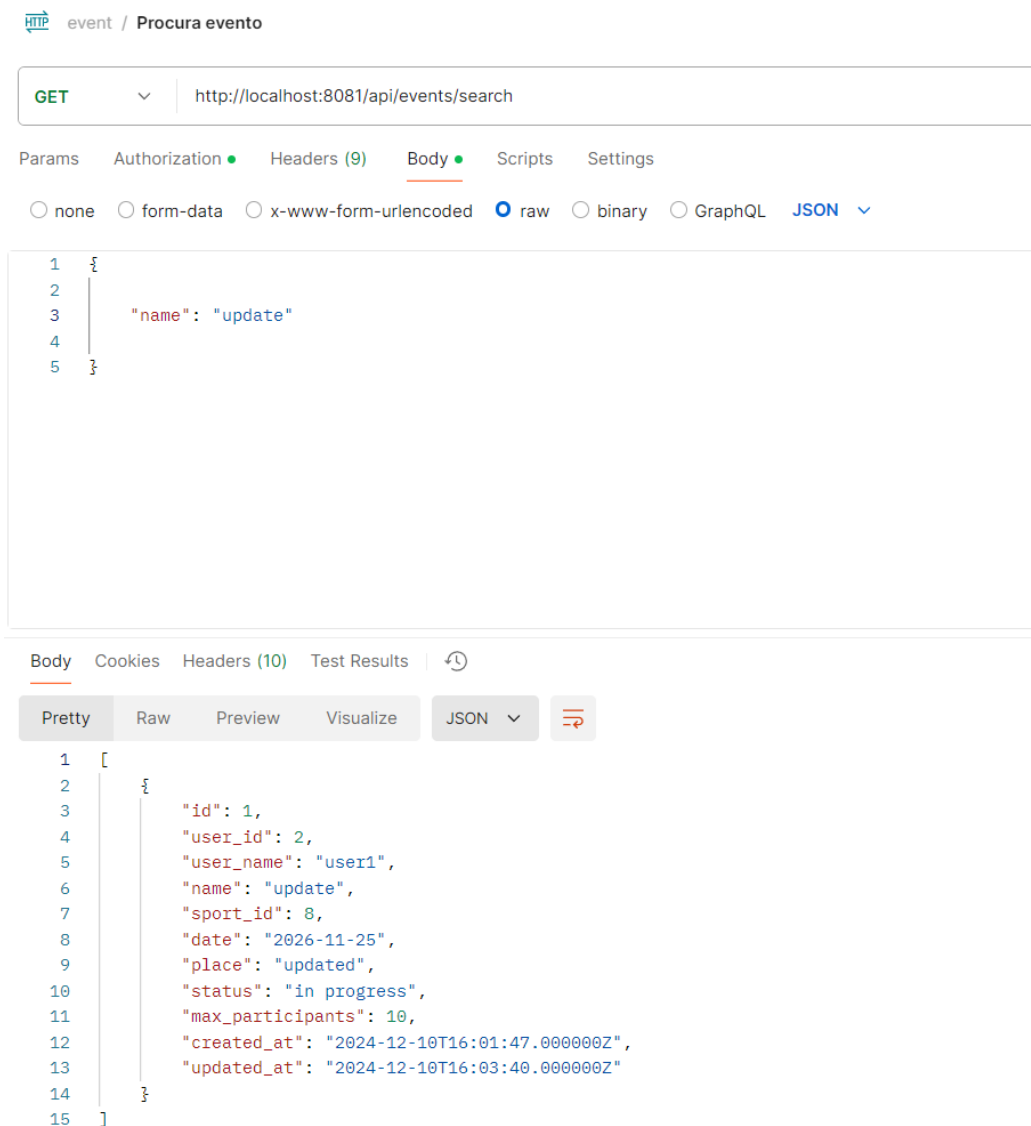


Figura 10: Procura de Evento por Nome

1.10.2 Procura por Desporto

Foi utilizado o parâmetro `sport_id` para procurar eventos associados a um desporto específico.

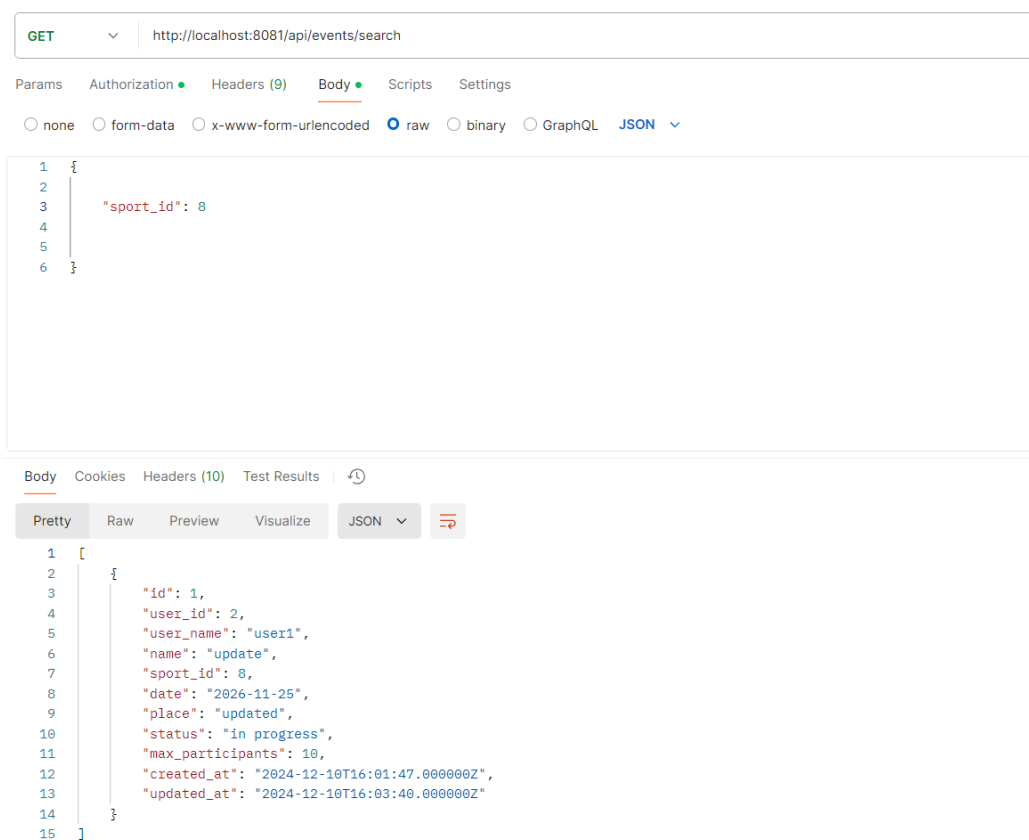


Figura 11: Procura de Evento por Desporto

1.10.3 Procura por Data

Foi utilizado o parâmetro `date` para procurar eventos realizados em uma data específica.

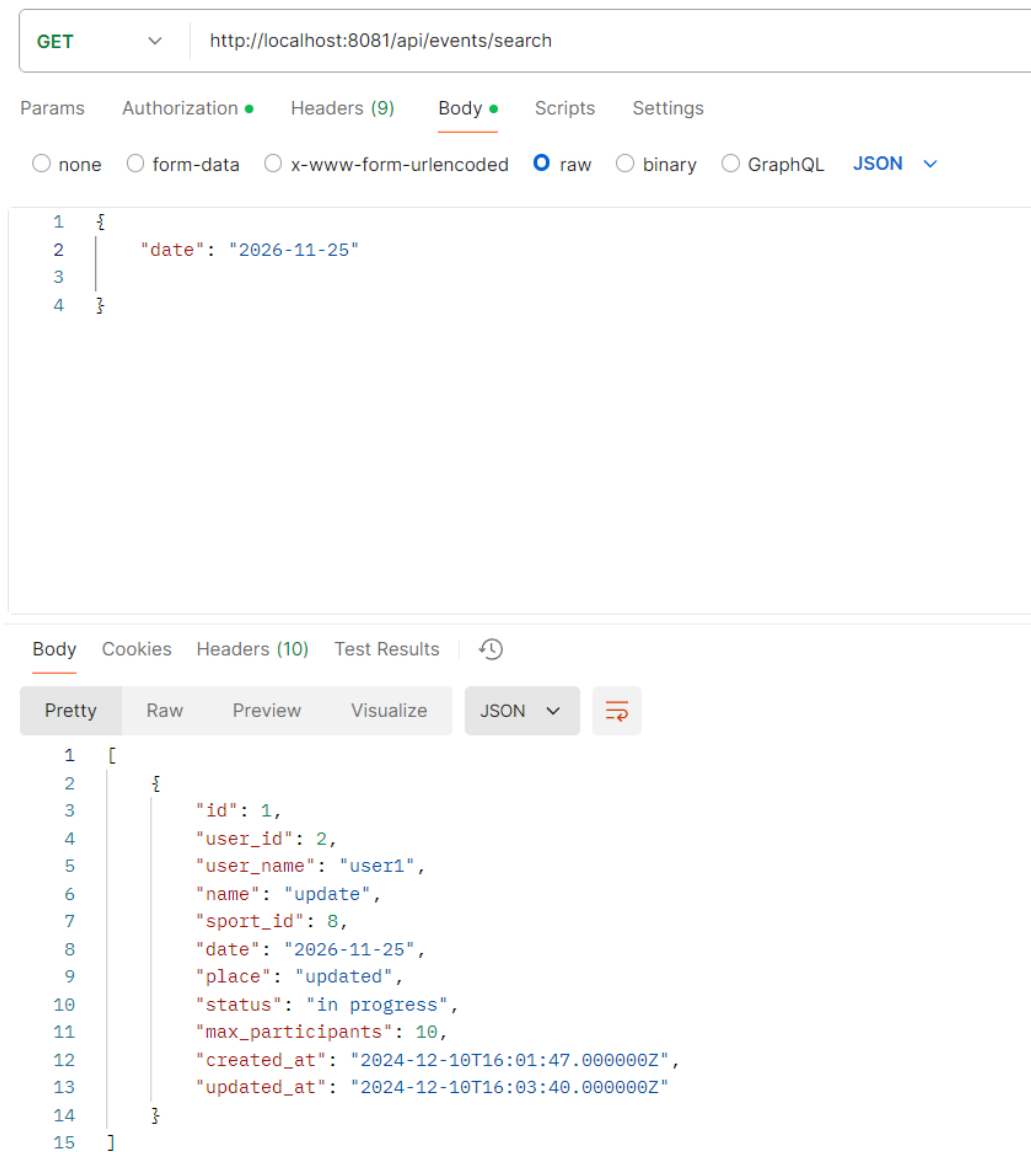


Figura 12: Procura de Evento por Data

1.10.4 Procura por Local

Foi utilizado o parâmetro `place` para procurar eventos num local específico.

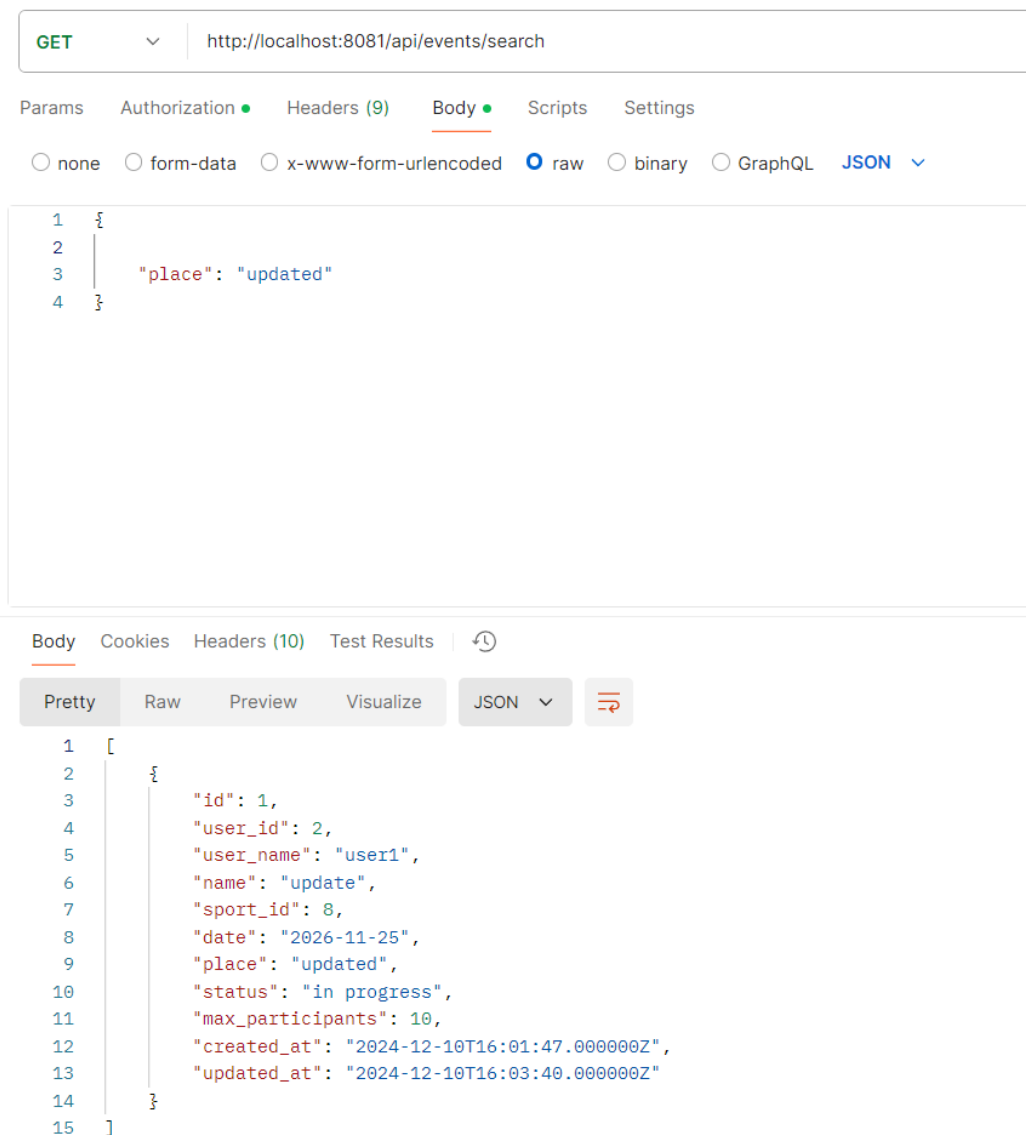


Figura 13: Procura de Evento por Local

1.10.5 Procura por ID

Foi utilizado o parâmetro `id` para procurar um evento com o ID específico.

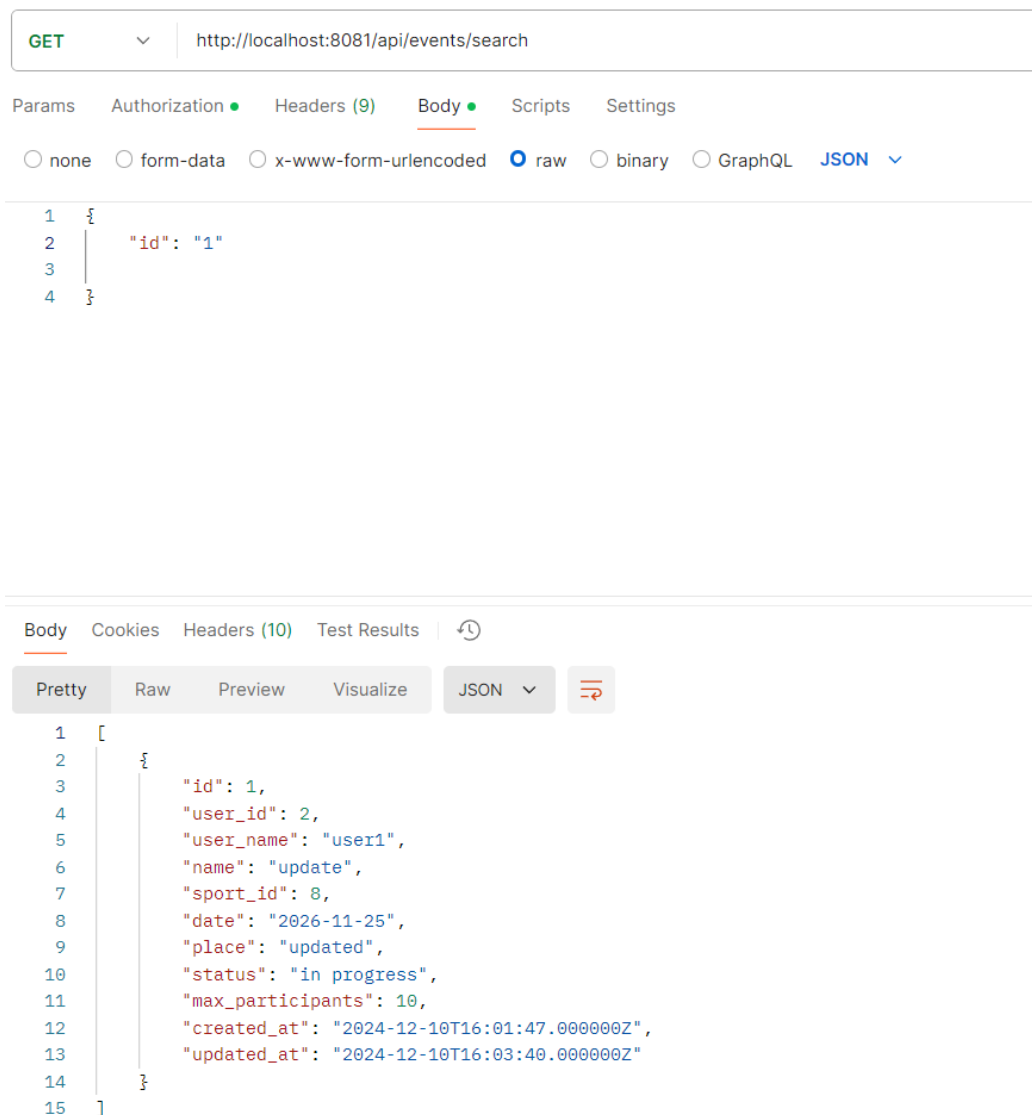


Figura 14: Procura de Evento por ID

1.11 Expulsão de Participante

Um segundo utilizador foi adicionado a um evento para testar a funcionalidade de expulsão de participantes através da rota `/api/events/event_id/participants/user_id`.

1.11.1 Estado Inicial

Antes da expulsão, foi utilizado o endpoint `/api/events/mine` para listar os eventos do utilizador. O utilizador estava listado como participante.

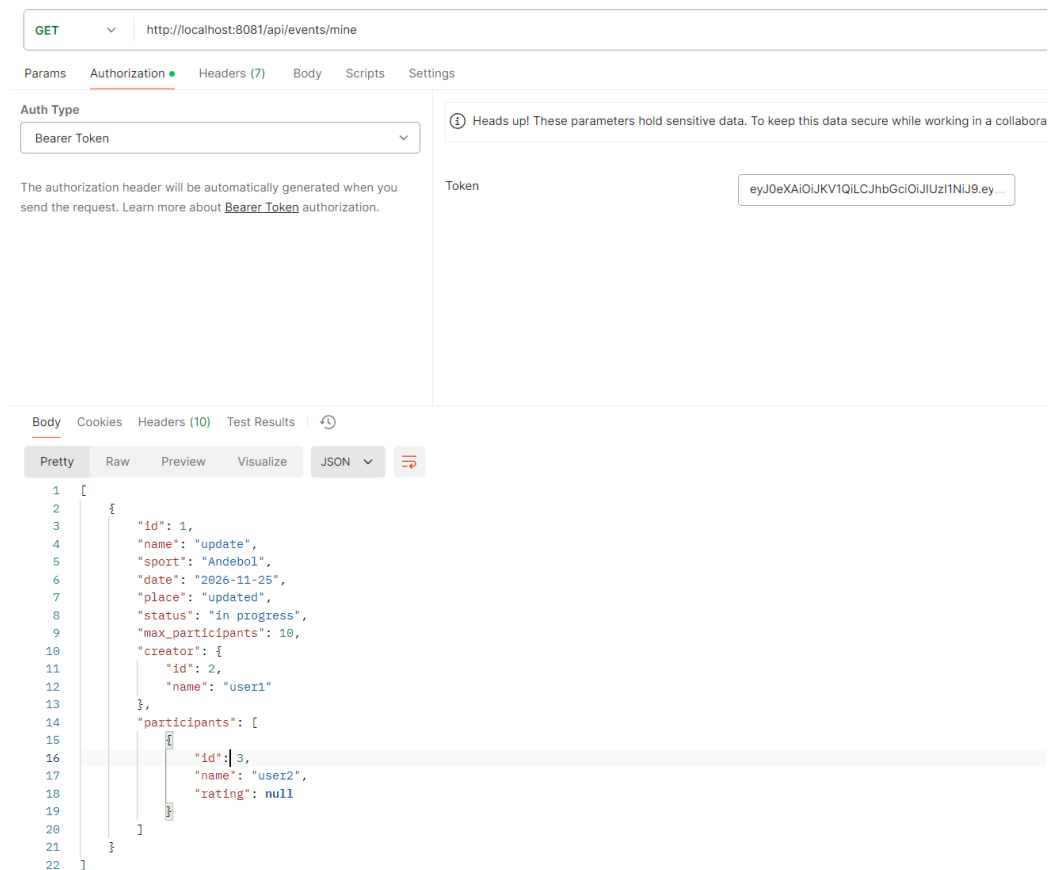


Figura 15: Estado Inicial Antes da Expulsão

1.11.2 Expulsão do Participante

O participante foi removido do evento utilizando o endpoint de expulsão usando o token do criador. A operação foi concluída com sucesso.

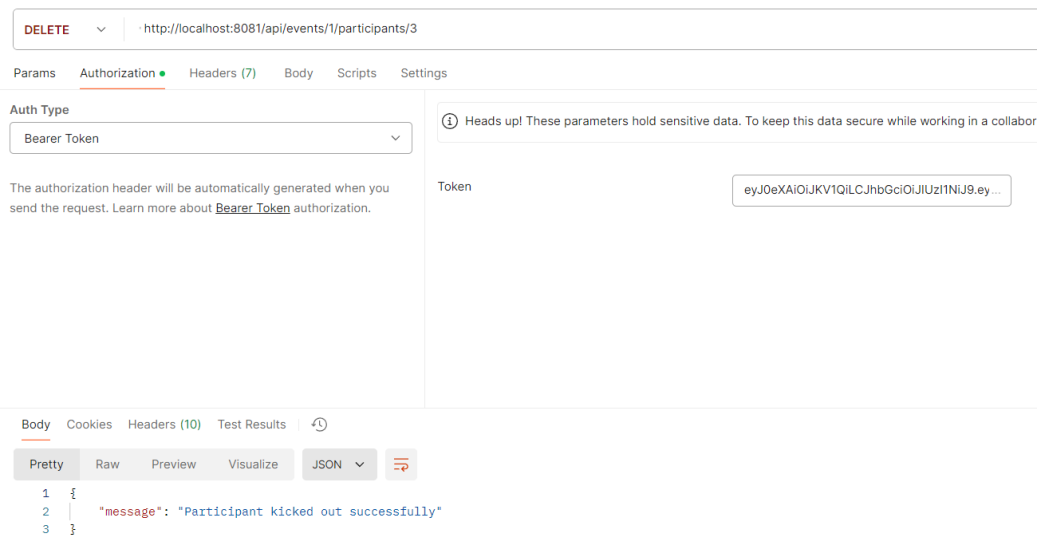


Figura 16: Expulsão do Participante

1.11.3 Estado Após a Expulsão

Após a expulsão, foi novamente utilizado o endpoint `/api/events/mine` para verificar se o utilizador expulsado se mantinha no evento. O utilizador não estava a participar no evento.

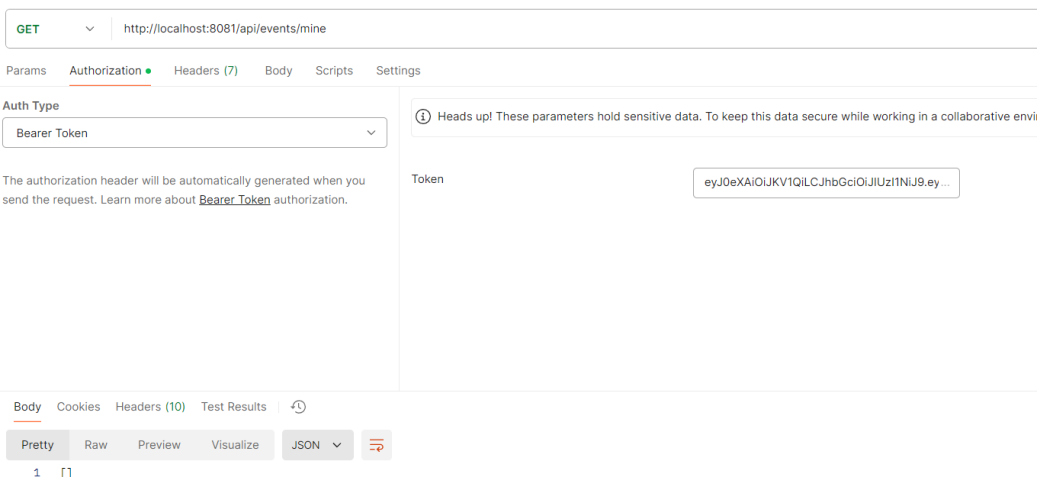


Figura 17: Estado Após a Expulsão

1.12 Listagem de Eventos (Admin)

Foi utilizado o endpoint `/api/admin/events` para listar todos os eventos no sistema. A operação foi realizada com sucesso, retornando todos os eventos disponíveis.

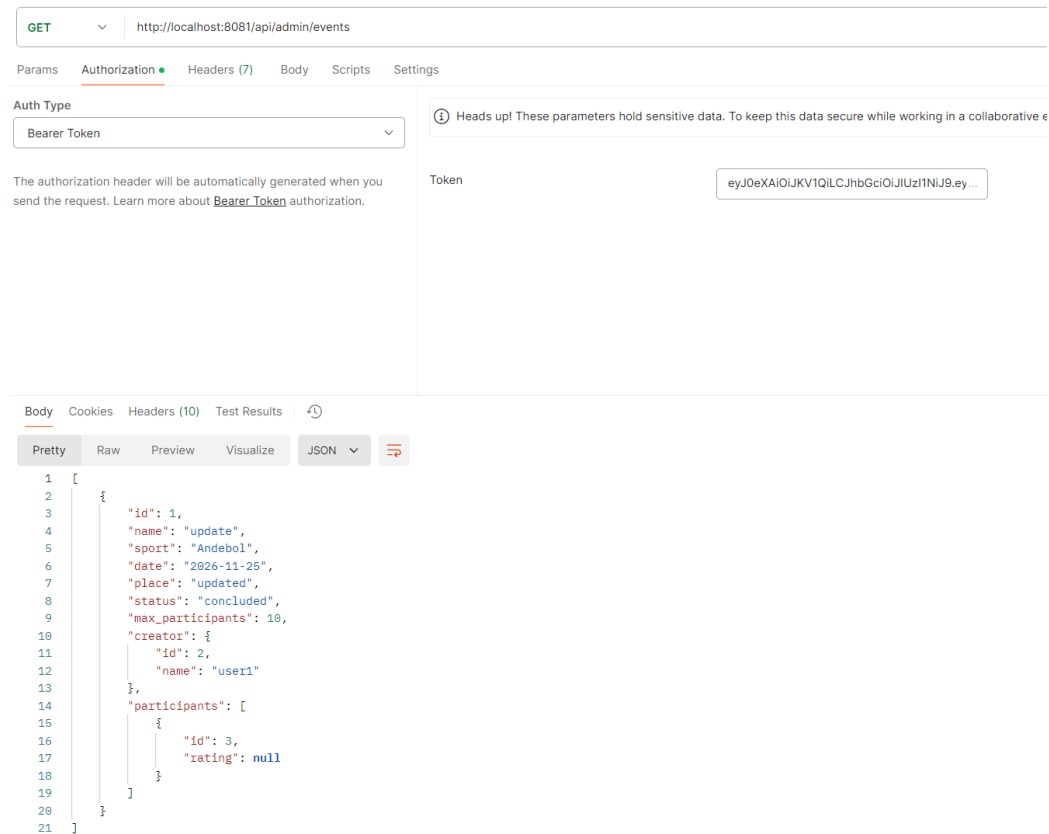


Figura 18: Listagem de Eventos (Admin)

1.13 Conclusão de Evento

Foi testada a funcionalidade de conclusão de um evento utilizando a rota `/api/admin/events/id/conclude`. O evento foi concluído com sucesso.

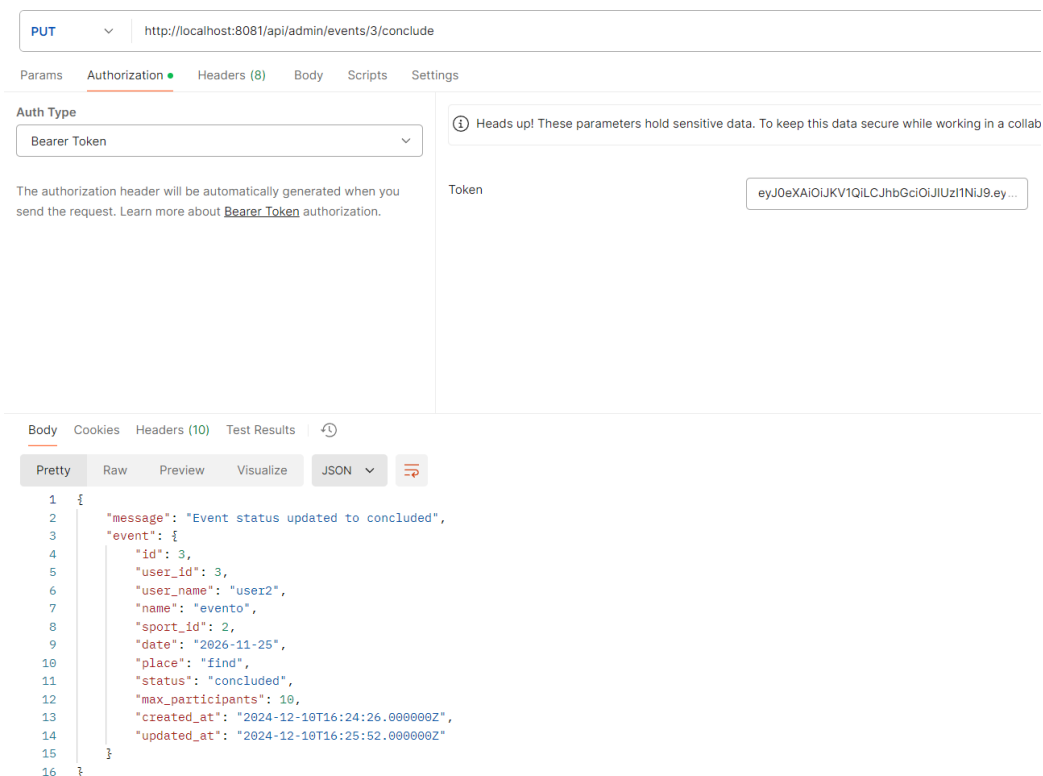


Figura 19: Conclusão de Evento

1.14 Avaliação de Participantes

Após a conclusão do evento, foi realizado um teste para avaliar um participante utilizando a rota `/api/events/event_id/rate`. A avaliação foi registrada com sucesso.

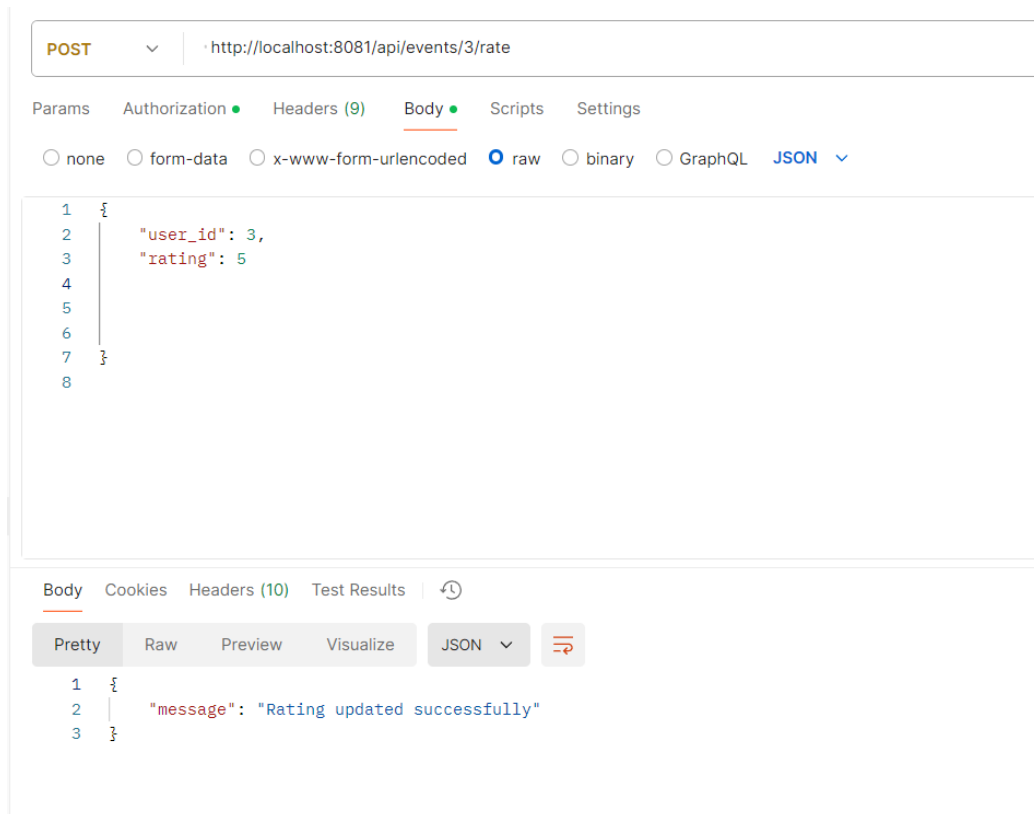


Figura 20: Avaliação de Participante

1.15 Verificação de Avaliação pelo Administrador

Foi utilizada a rota `/api/admin/events` para listar todos os eventos e confirmar a inclusão da avaliação no evento. O participante avaliado estava devidamente registrado com a avaliação atribuída.

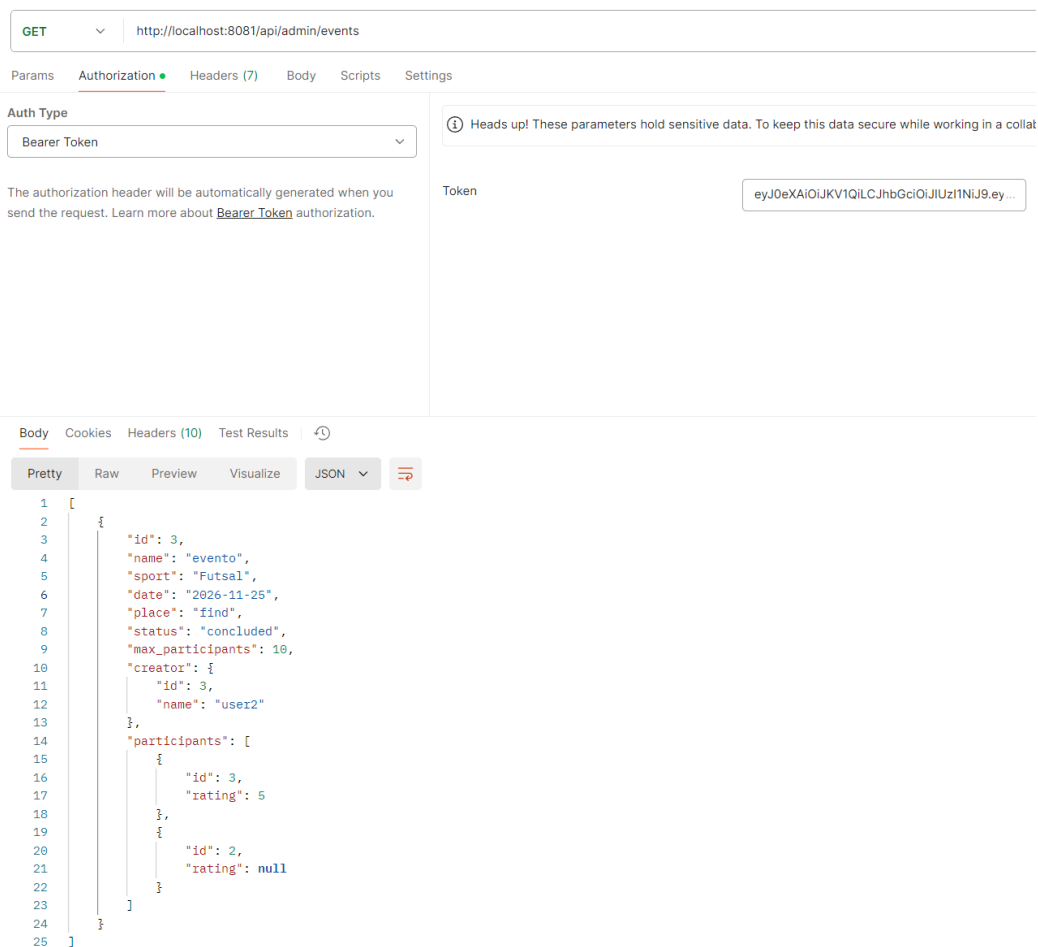


Figura 21: Verificação de Avaliação pelo Administrador

1.16 Eliminação de Evento pelo Administrador

Um evento foi eliminado utilizando a rota `/api/admin/events/id`. A resposta confirmou que o evento foi eliminado com sucesso.

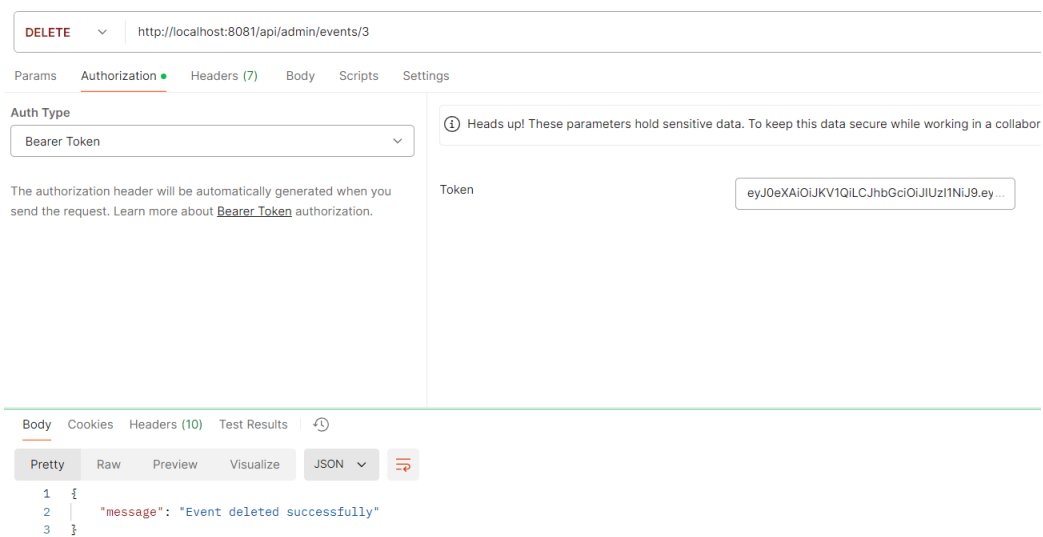


Figura 22: Eliminação de Evento pelo Administrador

1.17 Verificação de Eventos Após Eliminação pelo Administrador

Após a eliminação, foi novamente utilizada a rota `/api/admin/events` para listar todos os eventos e confirmar que o evento eliminado não estava mais listado.

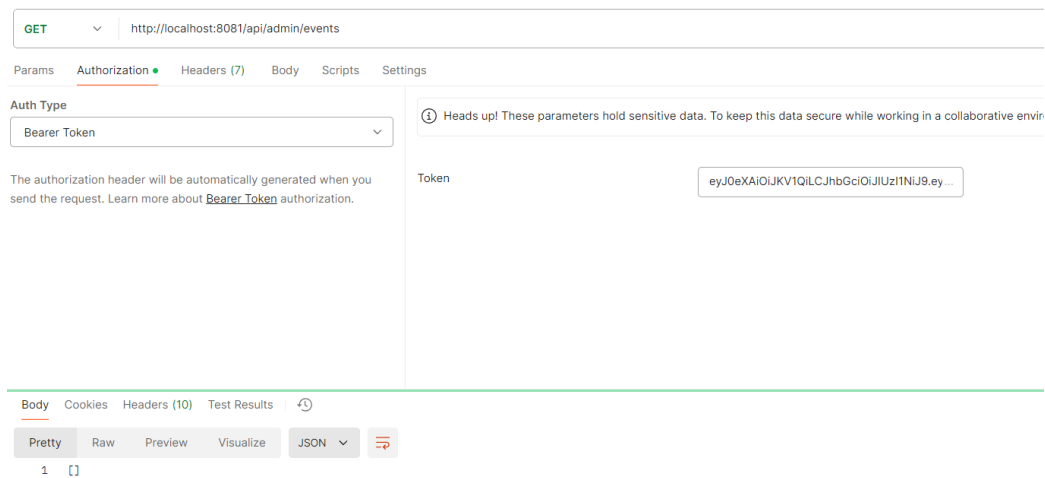


Figura 23: Verificação de Eventos Após Eliminação pelo Administrador

1.18 Resumo Atualizado dos Resultados

Os testes realizados confirmaram que o microserviço de eventos funciona corretamente, validando as seguintes funcionalidades:

- Criação, atualização e eliminação de eventos.
- Participação de utilizadores em eventos e listagem de participantes.
- Listagem e procura de eventos com base em diferentes critérios.
- Expulsão de participantes de um evento.
- Conclusão de eventos pelo administrador.
- Avaliação de participantes após a conclusão de eventos.
- Eliminação de eventos pelo administrador.

Todos os testes foram realizados utilizando um token JWT para garantir a autenticação e autorização em cada endpoint testado.