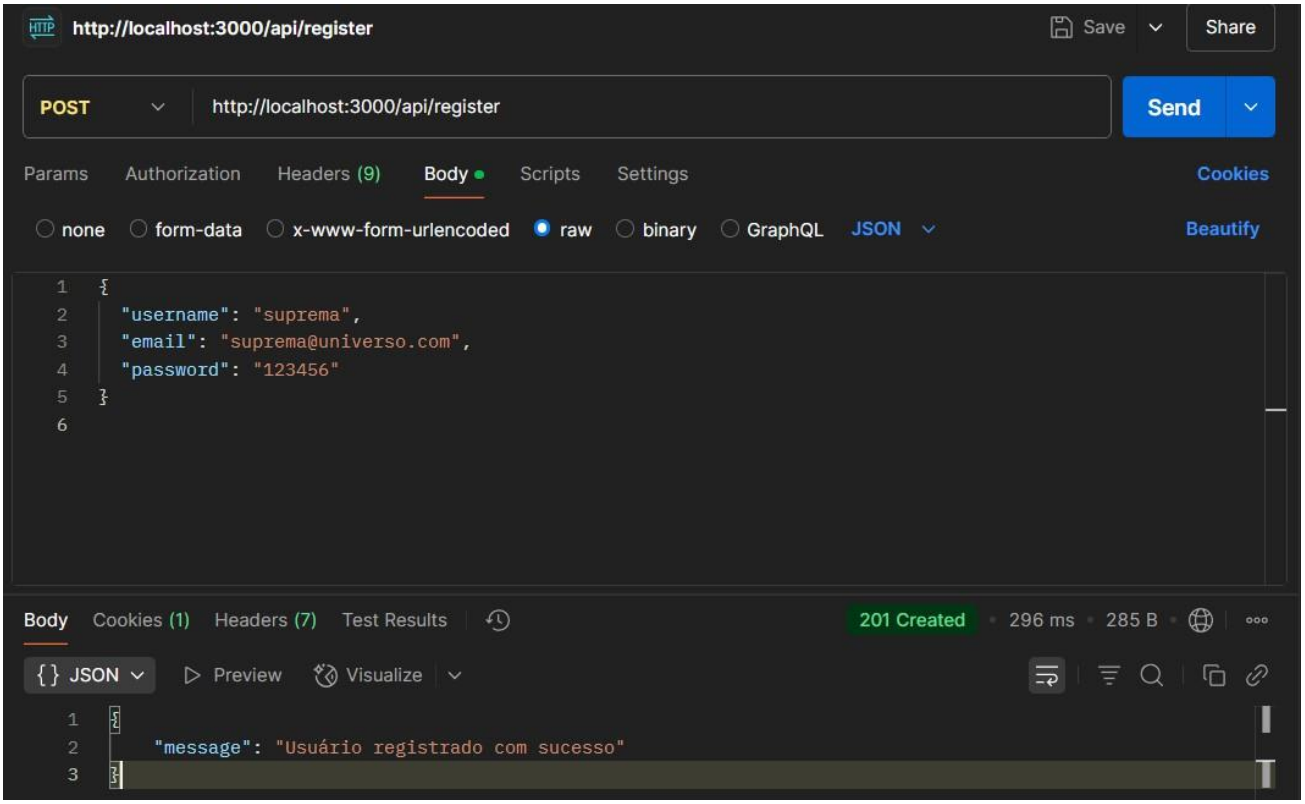
Pentesting (parte 3)

Pâmela Baron, Dereck Conink, Maria A. Giuliari e Mariele Vieira

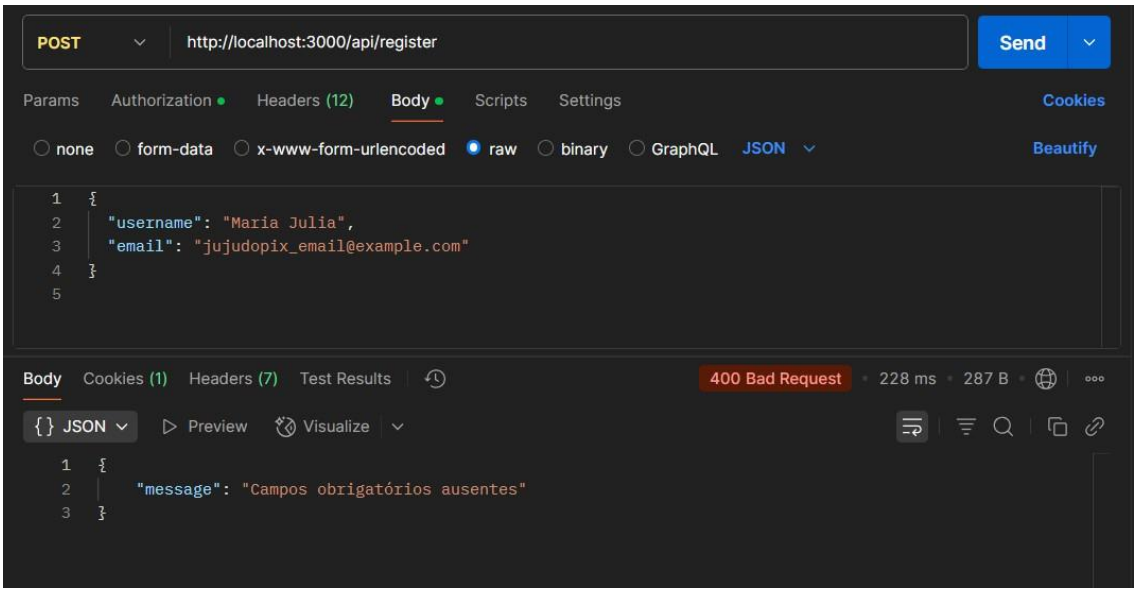
ROTA DE REGISTRO

1) Rota de Registro:
POST /register. Rota para registrar novos usuários. Os campos obrigatórios são username, email e password. A senha deve ser criptografada antes de ser armazenada no banco de dados.

- Método: POST
- URL: http://localhost:3000/api/register
- Teste no Postman:

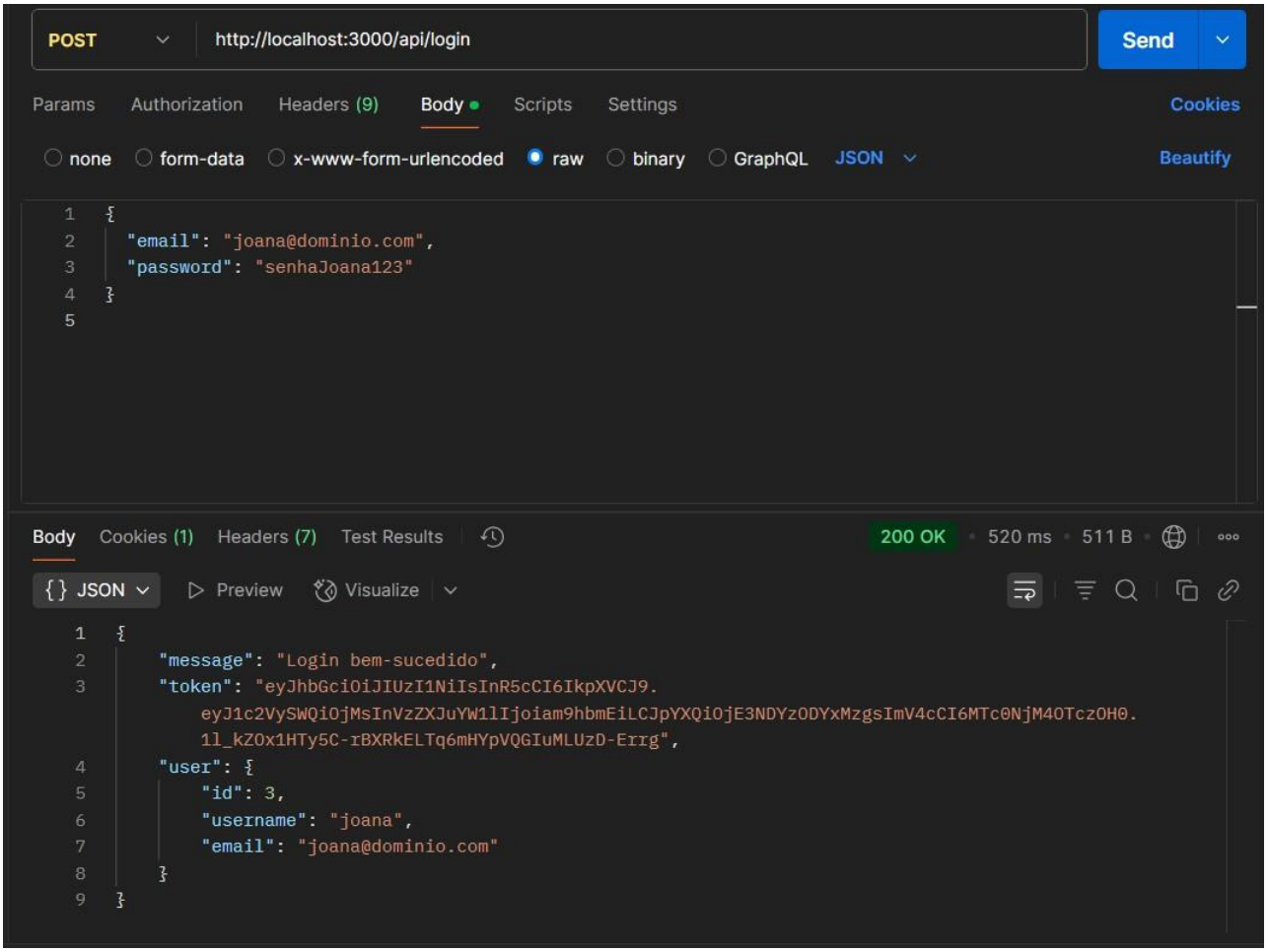


Caso falte algum campo obrigatório:



2) Rota de Login:
POST /login. Rota para login de usuários. O sistema deve validar as credenciais e, em caso de sucesso, retornar um token JWT que será usado para autenticação nas próximas requisições.

- Método: POST
- URL: http://localhost:3000/api/login
- Teste no Postman:



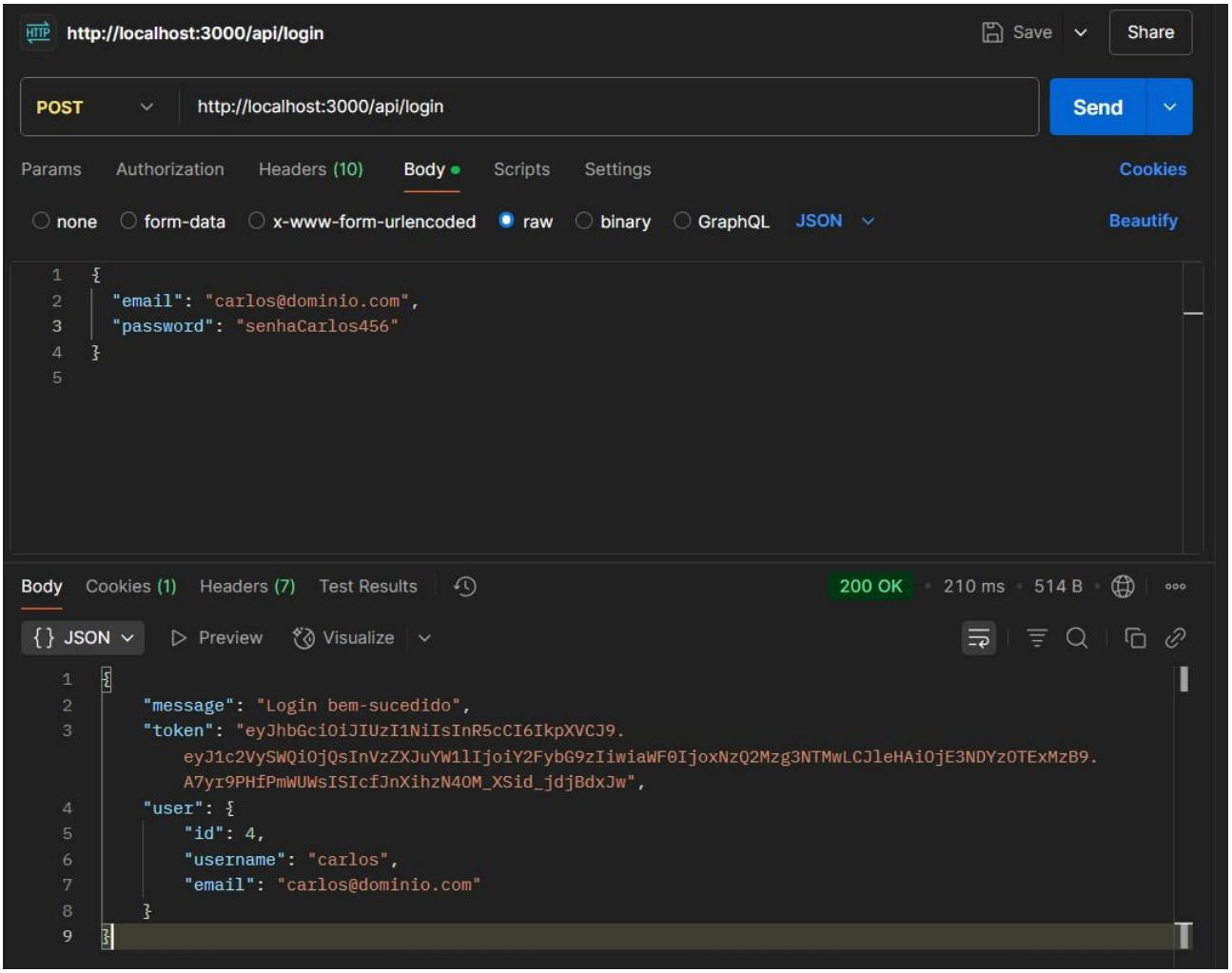
1) Rotas CRUD para Usuários:
GET /users: Retorna uma lista de todos os usuários (disponível para qualquer usuário autenticado).
PUT /users: Atualiza os dados de um usuário com base em seu id (apenas usuários autenticados).
DELETE /users: Exclui um usuário com base em seu id (apenas usuários autenticados).

3)

ROTAS CRUD PARA USUÁRIOS

Pré requisitos:

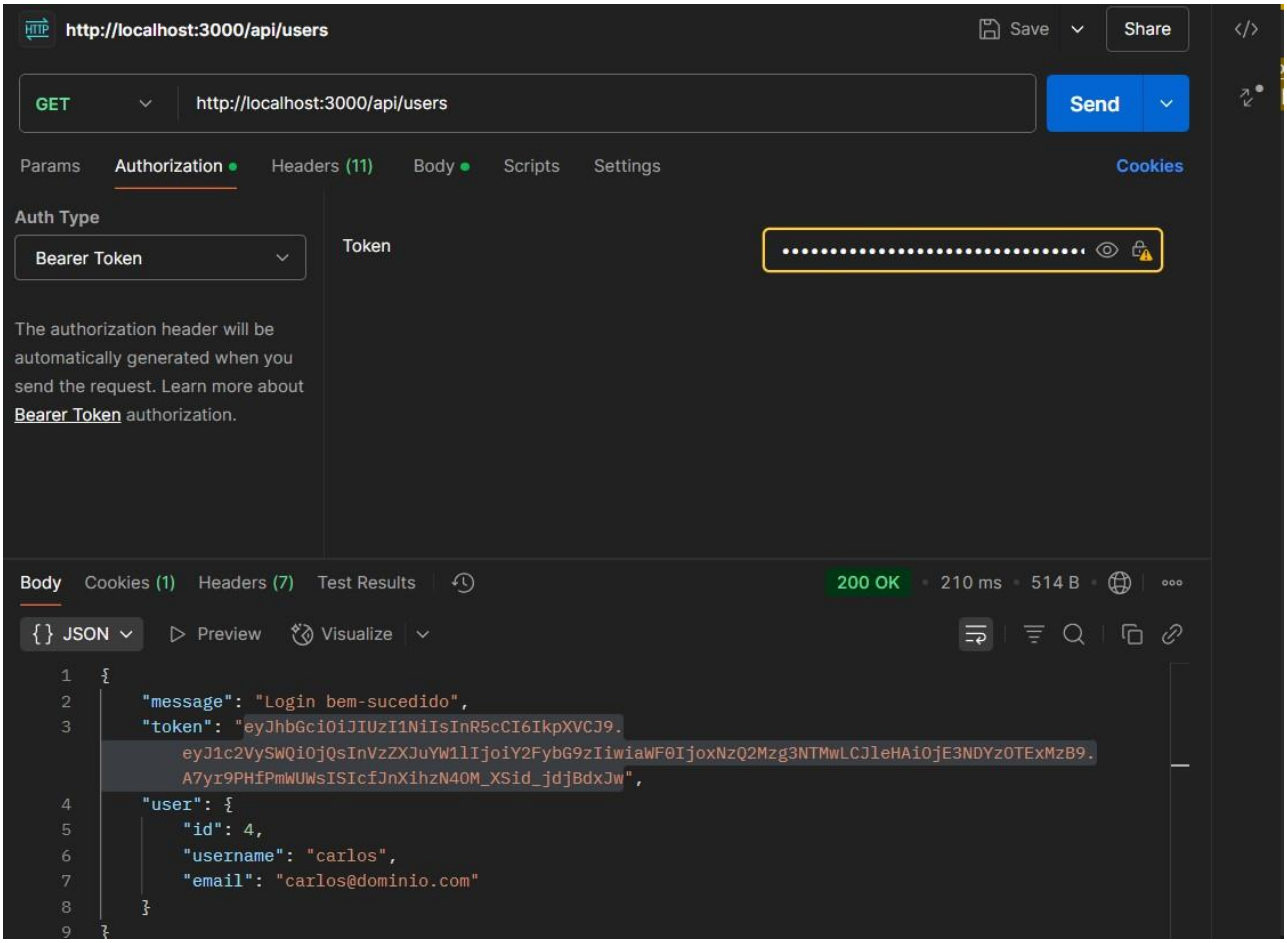
- Estar cadastrado e logado no sistema com um usuário válido.
- Fazer uma requisição POST para /api/login com email e senha válidos.
- Copiar o token JWT retornado.
- Utilizar o token em requisições protegidas.

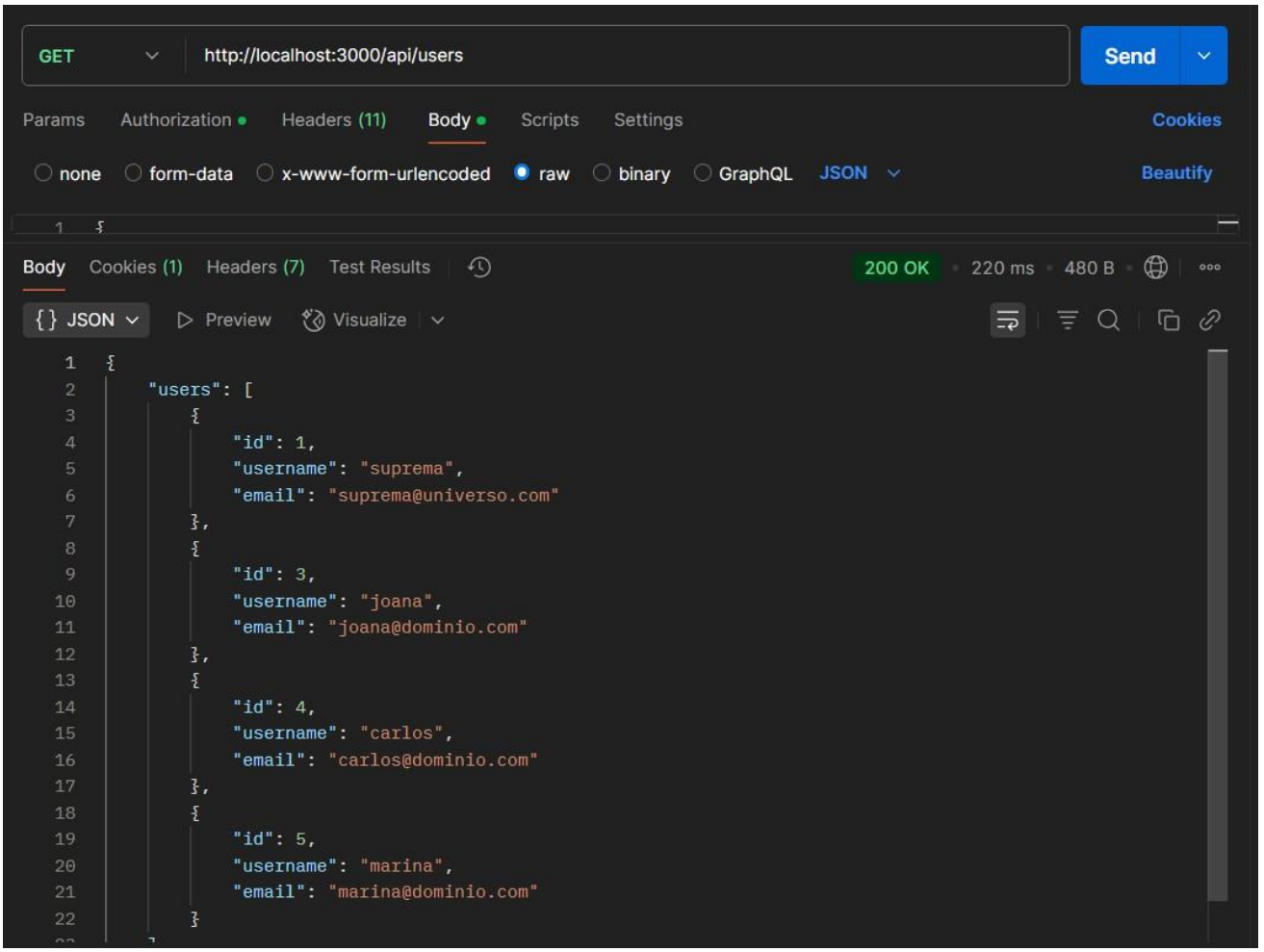


1. GET /api/users – Listar usuários

Configuração no Postman:

- Método: GET.
- URL: http://localhost:3000/api/users
- Headers:
    - Content-Type: application/json
    - Authorization: Bearer <O TONKEN DO CARLOS>

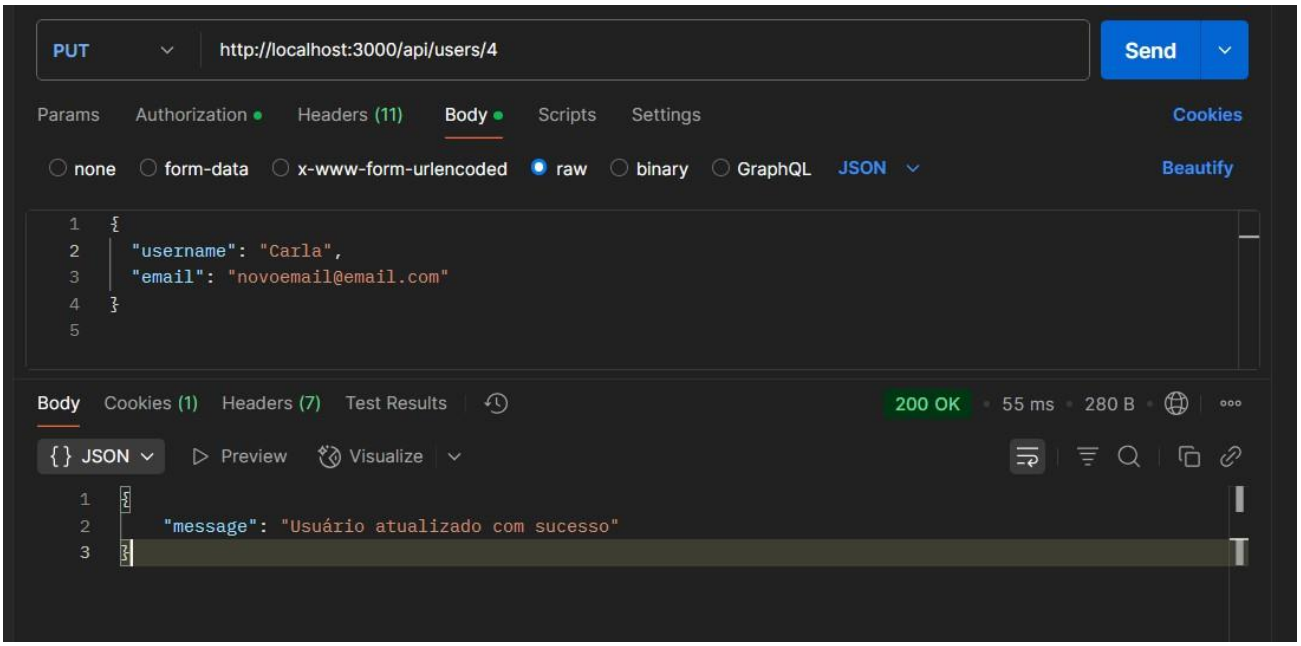




2. PUT /users: Atualiza os dados de um usuário com base em seu id (apenas usuários autenticados).

Configuração no Postman:

- Método: PUT.
- URL: http://localhost:3000/api/users/



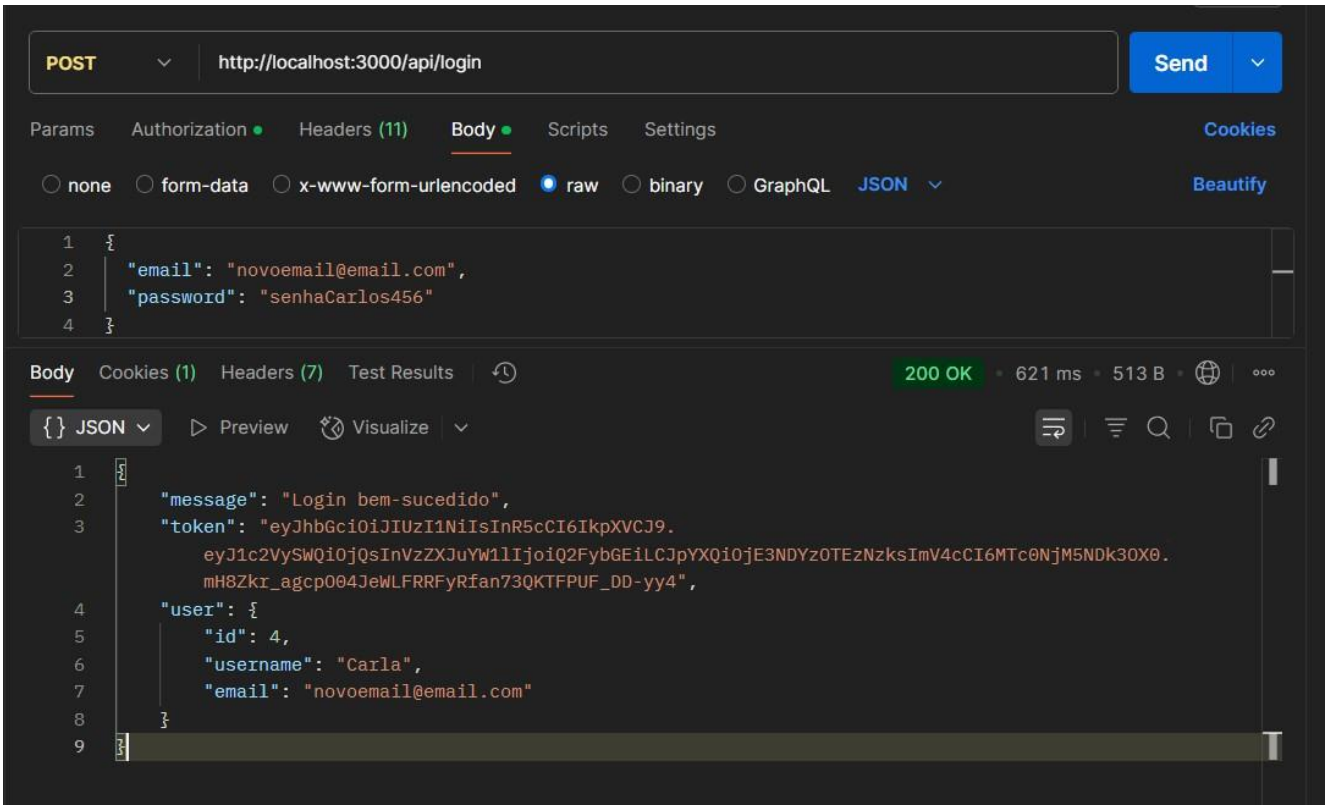
3. DELETE /users: Exclui um usuário com base em seu id (apenas usuários autenticados).

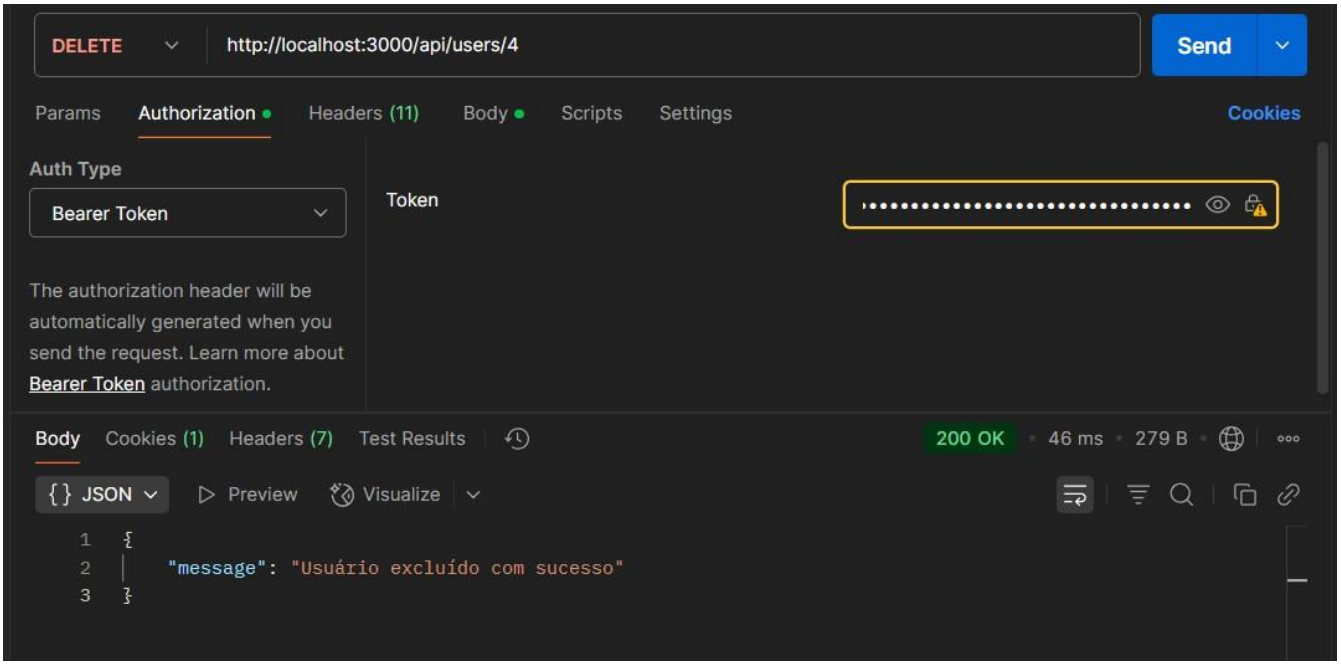
Pré requisitos:

- Já estar autenticado (ou seja, já ter feito login e ter um token JWT)
- O ID que será usado no DELETE deve ser o seu próprio ID de usuário (o mesmo contido no token).

Configuração no Postman:

- Método: DELETE.
- URL: http://localhost:3000/api/login


POST
http://localhost:3000/api/login
Send
Params
Authorization
Headers (11)
Body
Scripts
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"email": "novoemail@email.com",
"password": "senhaCarlos456"
}
Body
Cookies (1)
Headers (7)
Test Results
200 OK
621 ms
513 B
{} JSON
Preview
Visualize
{
"message": "Login bem-sucedido",
"token": "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
eyJ1c2VySWQiOjQsInVzZXJuYW1lIjoiQ2FybGEiLCJpYXQiOjE3NDYzOTEzNzksImV4cCI6MTc0NjM5NDk3OX0.
mH8Zkr_agcpO04JeWLFRRFyRfan73QKTFPUF_DD-yy4",
"user": {
"id": 4,
"username": "Carla",
"email": "novoemail@email.com"
}
}


DELETE
http://localhost:3000/api/users/4
Send
Params
Authorization
Headers (11)
Body
Scripts
Settings
Cookies
Auth Type
Bearer Token
Token
The authorization header will be automatically generated when you send the request. Learn more about Bearer Token authorization.
Body
Cookies (1)
Headers (7)
Test Results
200 OK
46 ms
279 B
{} JSON
Preview
Visualize
{
"message": "Usuário excluído com sucesso"
}

4) Proteção de Rotas:
As rotas PUT e DELETE devem ser protegidas, ou seja, somente usuários com um token JWT válido devem ter acesso.

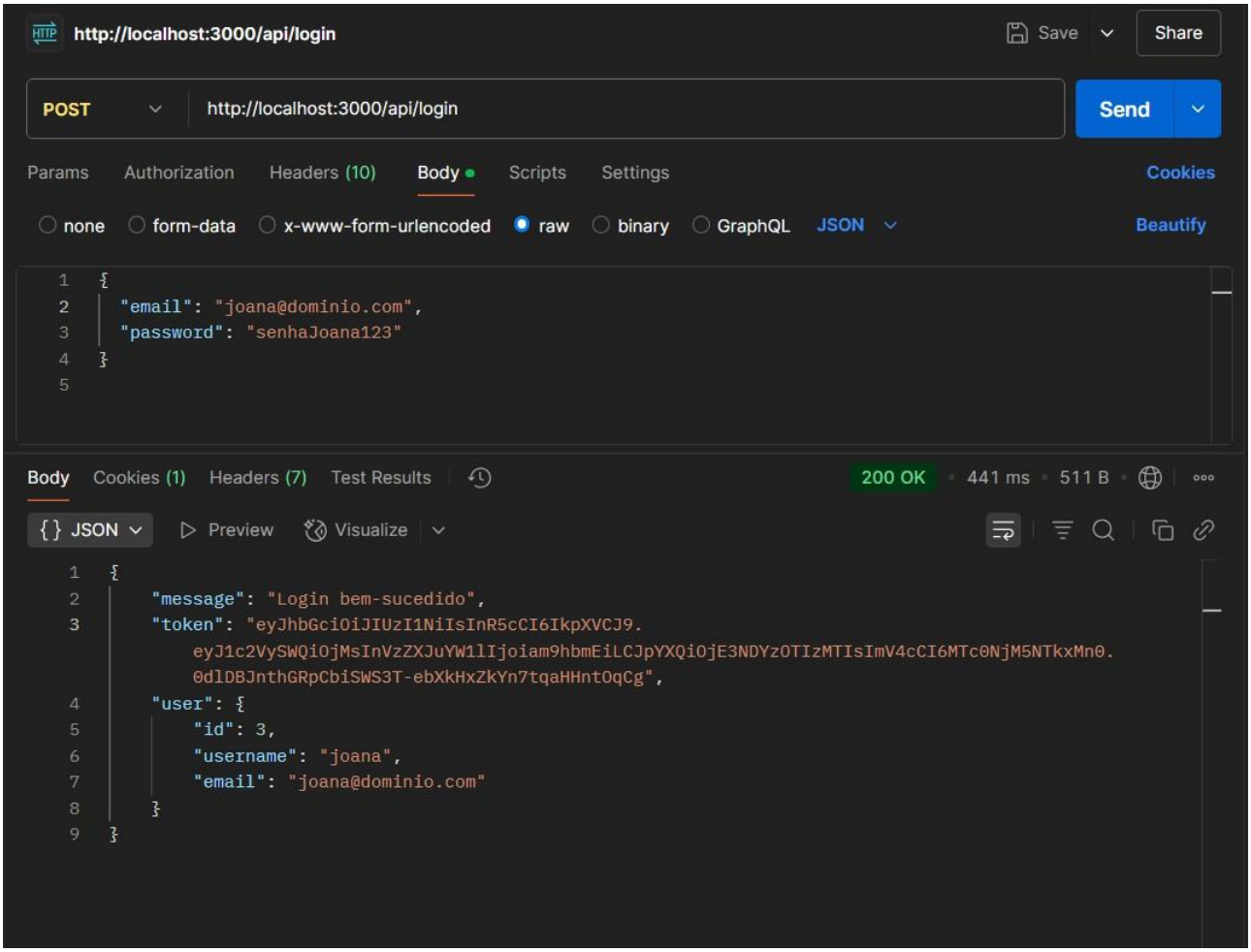
Passo a Passo (Rota PUT):

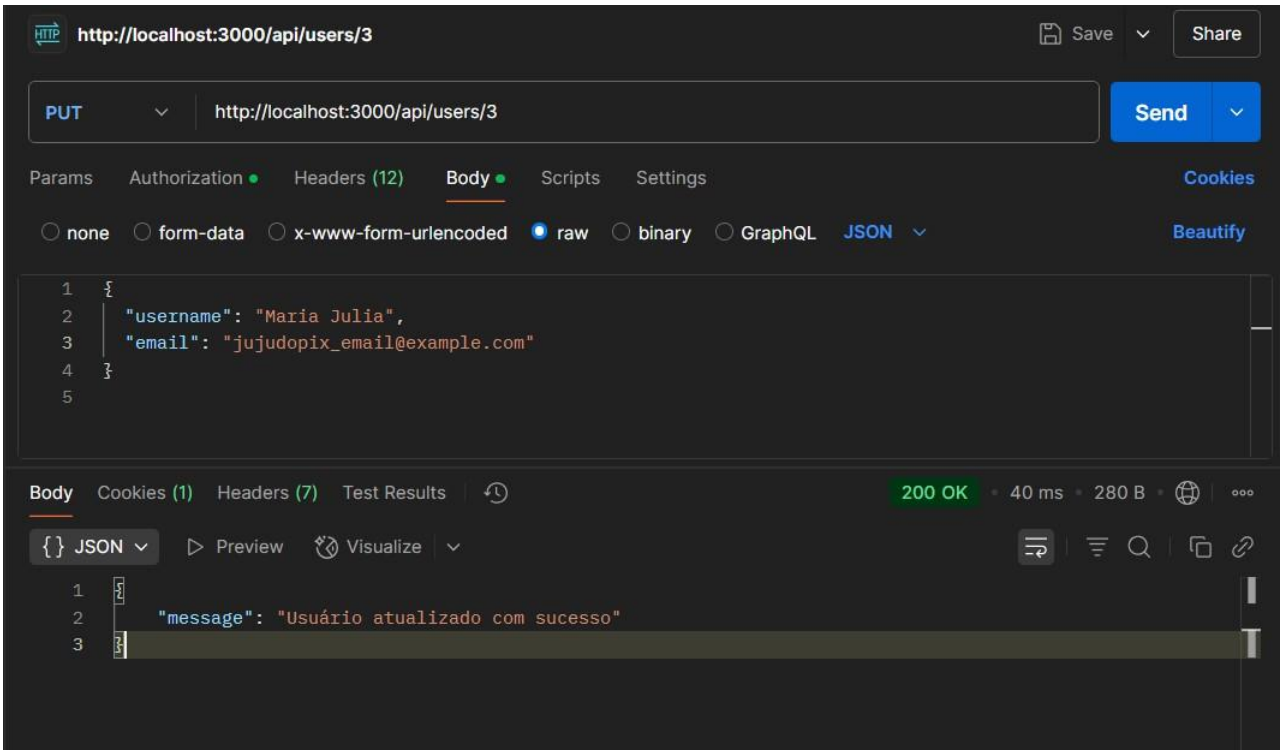
1. Realizar Login e obter o Token JWT.
2. Abrir o Postman.

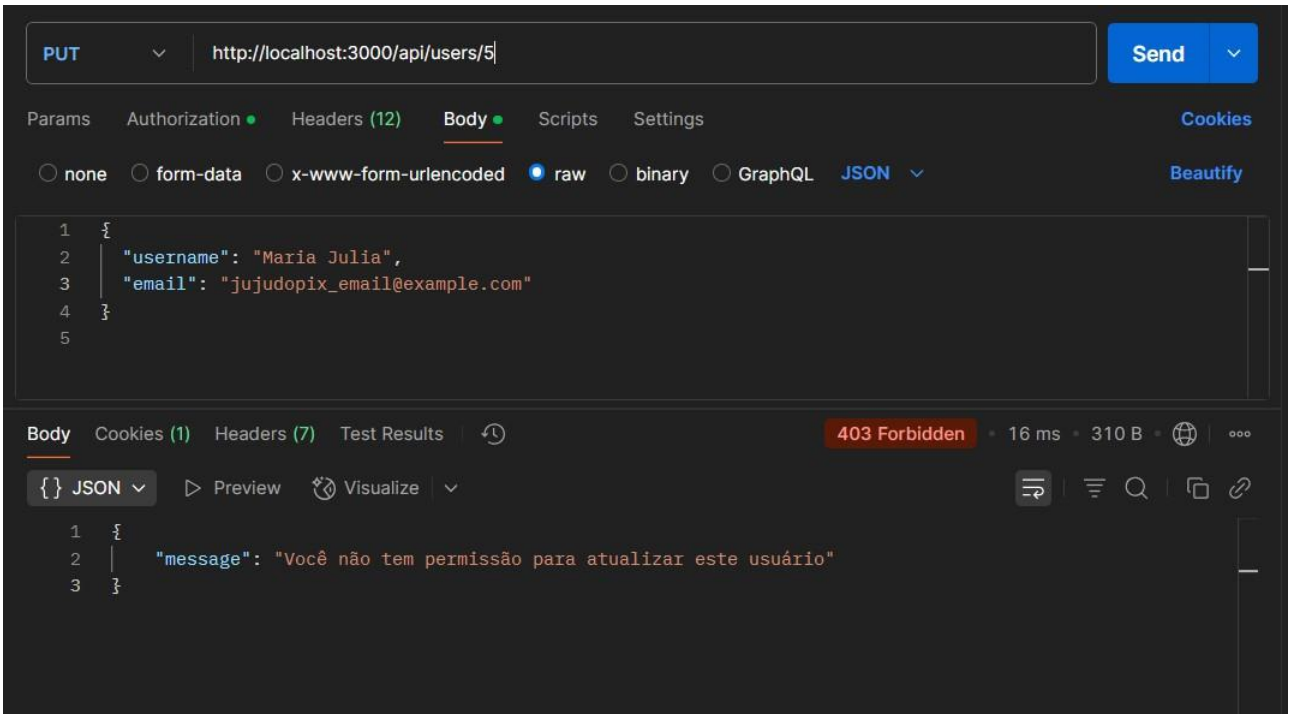
Criar uma nova requisição do tipo POST. Configuração no

Postman:

- Método: POST.
- URL: http://localhost:3000/api/login







Passo a Passo (Rota DELETE):

1. Realizar Login e obter o Token JWT.
2. Abrir o Postman.
3. Criar uma nova requisição do tipo DELETE.

Configuração no Postman:

- Método: DELETE.
- URL: http://localhost:3000/api/users/