

API Usuários

Introdução

A API de Usuários permite que você se cadastre como cliente ou lojista para utilizar na API de E-Commerce. Nela, é possível se cadastrar, editar seu usuário, logar no sistema, resetar sua senha, editar status e o tipo de permissão do seu usuário (cliente, lojista ou administrador), além de fazer uma pesquisa que retorna uma lista dos usuários cadastrados (pesquisa permitida somente por um perfil de administrador). A API Usuários usa protocolos HTTP para realizar as requisições, e se utiliza de objetos em JSON para tanto enviar informações como recebê-las. É implementado internamente em .NET 6 com Identity Framework para a criação dos usuários e é utilizado o MySQL como banco de dados, além disso, é utilizada a API ViaCep (<https://viacep.com.br/>) para seu funcionamento e está integrada à API E-Commerce.

Utilização

Ao criar o banco de dados do projeto pela primeira vez, automaticamente será criado uma conta de Administrador. As credenciais dessa conta poderão ser visualizadas diretamente no banco de dados (MySQL) ou no tópico Login deste documento, recomenda-se que a senha deste usuário seja redefinida o quanto antes. Para começar a utilizar a API, basta inserir a URL, selecionar o método desejado e inserir no corpo da requisição as informações pertinentes (em JSON). Note que, para realizar algumas ações, é necessário estar logado com um tipo de conta específico. Para saber quais usuários têm acesso ao método, verifique o tópico do método desejado.

URL padrão

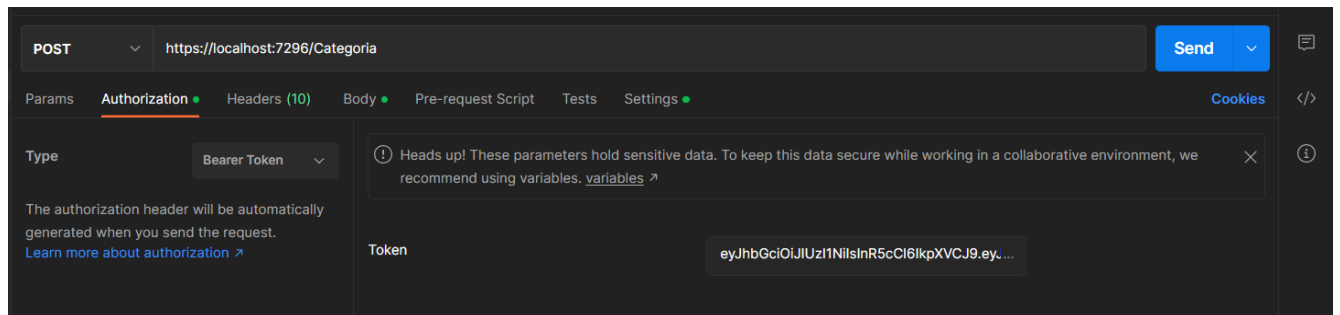
A URL padrão da API Usuários é:

`https://localhost:7221/{controller-desejada}`

Controller-desejada: Nome da controller a ser utilizada na requisição, podendo ser /cliente, /lojista, /solicita-reset, etc. Verifique o método para obter a informação completa.

Utilizando métodos que necessitam de login

Para validar o login em métodos que necessitem do mesmo, basta realizar a operação de login descrita neste documento. Copiar o Token retornado nesta requisição, ir ao método desejado que necessita de login e, na aba Authorization, selecionar o tipo Bearer Token e colar o mesmo no campo Token, como na imagem a seguir:



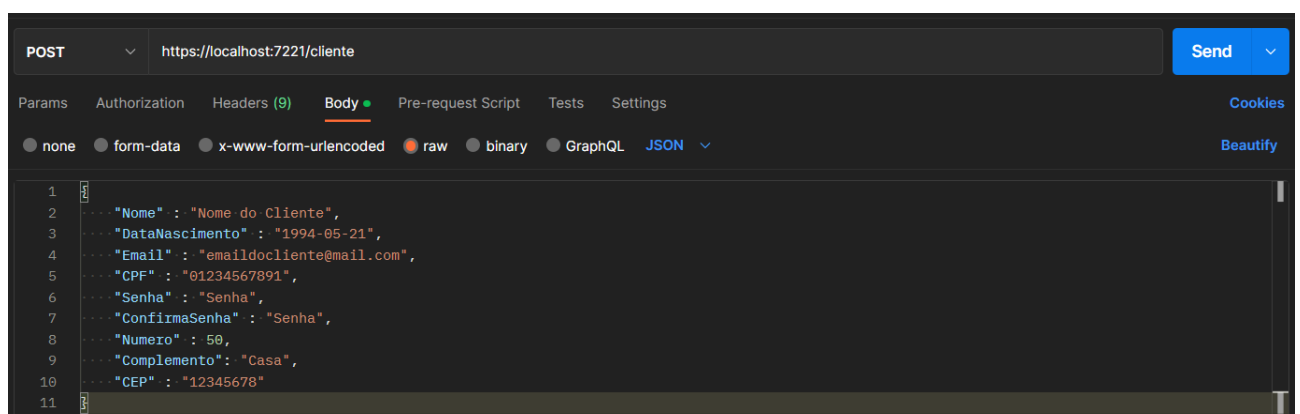
Atenção: os métodos exigem um tipo de usuário específico para a requisição, caso seja inserido um Token de usuário que não é válido, o método retornará com Status Code 403 - Forbidden. Caso o Token não seja inserido, o método retornará o Status Code 401 - Unauthorized.

Cadastro de Usuários

- CADASTRO DE CLIENTE - MÉTODO POST

URL: **https://localhost:7221/cliente**

Para cadastrar um novo cliente, é necessário enviar um objeto em JSON no Body da requisição, com alguns campos que são obrigatórios para o cadastro:



Ao enviar a requisição com as informações válidas, o método retornará um Status Code 201 - Created At Action, indicando que a criação ocorreu com sucesso. Além disso, retorna o objeto criado para visualização.

O cadastro de um cliente está condicionado a alguns critérios, caso um ou mais não sejam atendidos, o cadastro não será realizado e o usuário receberá o retorno de uma mensagem de erro especificando o mesmo junto com o Status Code 400 - Bad Request.

Os critérios para cadastrar um cliente com sucesso são:

- ☒ O campo Nome deve conter apenas letras do alfabeto, o único caractere especial permitido é o espaço. Além disso, pode conter no máximo 250 caracteres e não é possível cadastrar um nome que já esteja cadastrado;
- ☒ A Data de Nascimento deve ser anterior à data do cadastro. Além disso, o formato informado deve ser o seguinte: 'ano-mês-dia' (ex. 1990-01-20 para 20 de janeiro de 1990);
- ☒ O E-mail deve possuir um formato válido, ou seja email@email.com, além disso não é possível cadastrar um e-mail que já esteja cadastrado;
- ☒ O CPF deve ser válido, para casos de teste, recomenda-se a utilização de um gerador de CPFs válidos, deve ser informado os 11 números, sem pontuação ou hífen;
- ☒ A Senha deve possuir no mínimo 8 caracteres, entre eles: uma letra minúscula, uma letra maiúscula, um caractere especial e um número;
- ☒ O campo Confirma Senha deve ter seu valor idêntico ao campo senha;
- ☒ O campo CEP deve possuir um CEP válido e conter os 8 dígitos do mesmo, sem pontos ou hífen;
- ☒ O campo número deve ser um número maior que zero;
- ☒ O campo Complemento deve ser preenchido com o complemento do endereço.

- CADASTRO DE LOJISTA - MÉTODO POST

URL: <https://localhost:7221/lojista>

Para o cadastro de um lojista, é necessário logar com uma conta de Administrador ou Lojista. Para mais detalhes, verificar o tópico "Utilizando métodos que necessitam de login". Além do Token de acesso, para o cadastro do lojista, é necessário enviar um objeto em JSON, na aba Body do Postman, com as informações necessárias. Os campos obrigatórios para o cadastro de um lojista são os mesmos para o cadastro de um cliente, para mais detalhes, verificar o tópico Cadastro de Cliente.

Ao enviar a requisição com o Token de acesso e as informações válidas, o método retornará um Status Code 201 - Created At Action, indicando que a criação ocorreu com sucesso. Além disso, retorna o objeto criado para visualização.

O cadastro de um lojista está condicionado a alguns critérios, caso um ou mais não sejam atendidos, o cadastro não será realizado e o usuário receberá o retorno de uma mensagem de erro especificando o mesmo junto com o Status Code 400 - Bad Request. Além disso, caso a requisição seja feita com um Token de acesso de um usuário não permitido (cliente), o retorno será um Status Code 403 - Forbidden. Caso o Token não seja inserido, o método retornará o Status Code 401 - Unauthorized.

Os critérios para o cadastro de um lojista são os mesmo que o cadastro de cliente, para mais informações verificar o tópico de Cadastro de Cliente.

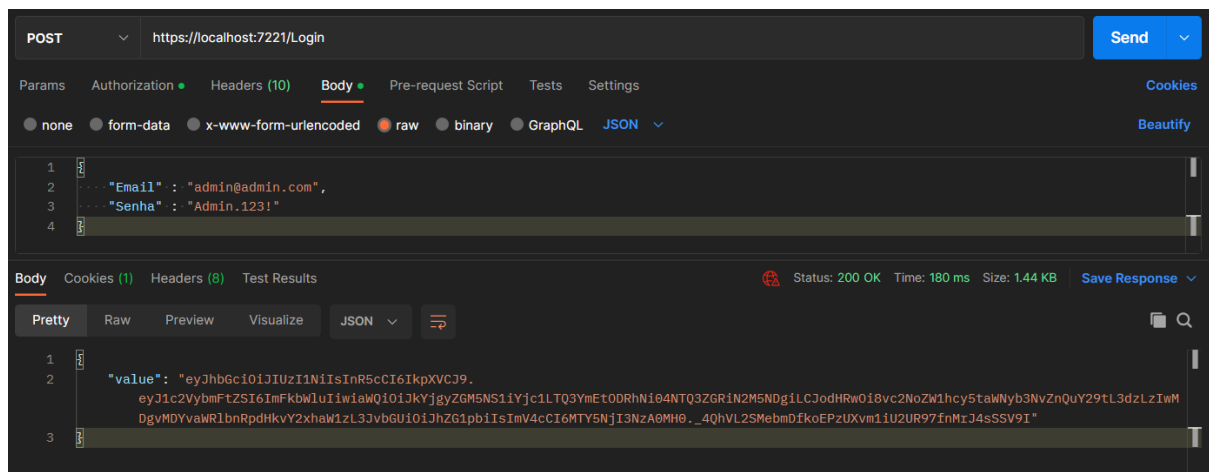
Login de Usuários

- LOGIN - MÉTODO POST

URL: `https://localhost:7221/Login`

Para efetuar o login no sistema, basta fazer a requisição enviando, via Body no Postman, os campos de E-mail com a conta que deseja logar e a Senha respectiva desta conta. A API Usuários possui um cadastro padrão de Administrador, com e-mail “admin@admin.com” e uma senha padrão “Admin.123!”. Recomenda-se que esta senha seja alterada para outra de sua preferência.

Exemplo de login:



Ao informar as credenciais válidas, a requisição retornará um objeto com um Token. Este Token será utilizado nas requisições que necessitam de login.

Neste Token está criptografado algumas informações do usuário, são elas:

- Nome do usuário;
- ID do usuário;
- Qual o perfil dele (cliente, lojista ou administrador);
- Em quanto tempo o Token irá expirar (por padrão, o Token é válido por 30 minutos, após esse período será necessário realizar a requisição de login novamente).

Caso seja informado uma senha inválida ou o e-mail não esteja cadastrado na base, o usuário obterá um retorno com Status Code 400 - Bad Request.

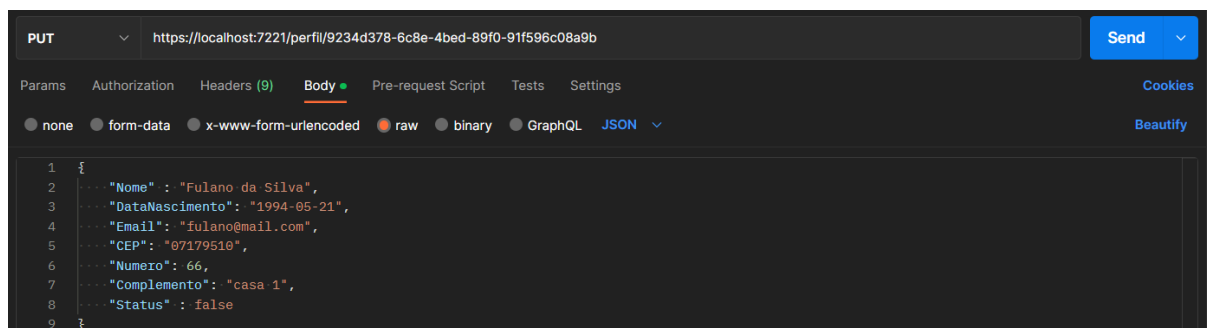
Edição de Usuários

- EDITAR PERFIL - MÉTODO PUT

URL: `https://localhost:7221/perfil/{id-do-usuário}`

Para editar o perfil de um usuário, basta informar na URL o ID do mesmo, além disso, enviar no Body da requisição os campos que são editáveis. Atenção: é necessário informar todos os campos para edição, se não deseja editar algum campo, basta manter a informação original, por exemplo, quero inativar o usuário, no Body eu passo todos os campos com as informações originais e modifico apenas o campo Status como 'false'.

Exemplo para edição:



Ao enviar a requisição com sucesso, a mesma retornará o Status Code 204 - No Content, indicando que a edição foi feita com sucesso.

Assim como o cadastro, a edição de usuários está condicionada a alguns critérios, que, caso não sejam atendidos, poderá resultar em Status Code 400 - Bad Request ou Status Code 404 - Not Found (nos casos onde o ID do usuário informado não foi encontrado).

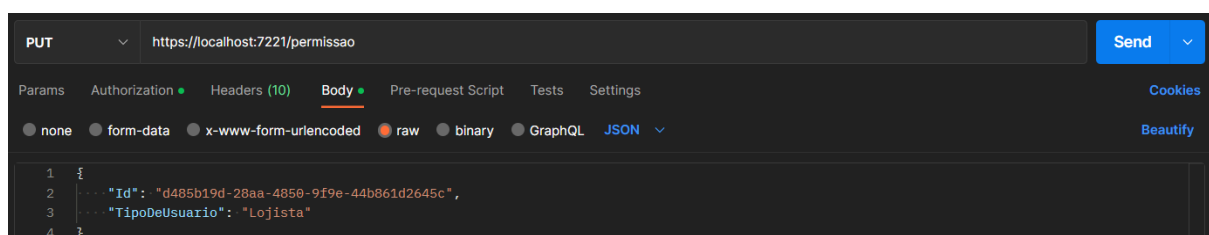
Os critérios para editar um usuário com sucesso são:

- ☒ O campo Nome deve ser único, possuir apenas letras do alfabeto e ter no máximo 250 caracteres;
- ☒ A Data de Nascimento deve ser anterior à data do cadastro. Além disso, o formato informado deve ser o seguinte: 'ano-mês-dia' (ex. 1990-01-20 para 20 de janeiro de 1990);
- ☒ O E-mail deve possuir um formato válido, ou seja email@email.com, além disso não é possível cadastrar um e-mail que já esteja cadastrado;
- ☒ O campo CEP deve possuir um CEP válido e conter os 8 dígitos do mesmo, sem pontos ou hífen;
- ☒ O campo Número deve ser um número maior que zero;
- ☒ O campo Complemento deve ser preenchido com o complemento do endereço.
- ☒ O campo Status deve ser definido como 'true' para Ativo ou 'false' para Inativo.

- *EDITAR PERMISSÃO DE USUÁRIO - MÉTODO PUT*

URL: <https://localhost:7221/permissoao>

Como Administrador, é possível editar a role de outros usuários, ou seja, o tipo daquele usuário. Na prática, é possível editar um usuário que seja do tipo Lojista para um Administrador, e também é possível editar um Administrador para Lojista. Para efetuar essa operação, é necessário estar logado com uma conta do tipo Administrador e enviar, via Body do Postman, o ID do usuário que deseja modificar a permissão e para qual tipo de usuário desejá-lo.



Note que, neste projeto, não é possível que um mesmo usuário seja do tipo Administrador **E** Lojista ao mesmo tempo, é possível modificar o tipo para um **OU** outro.

Ao enviar a requisição com sucesso, o retorno será um Status Code 204 - No Content.

Para efetuar a edição da permissão com sucesso, é necessário se atentar aos seguintes critérios:

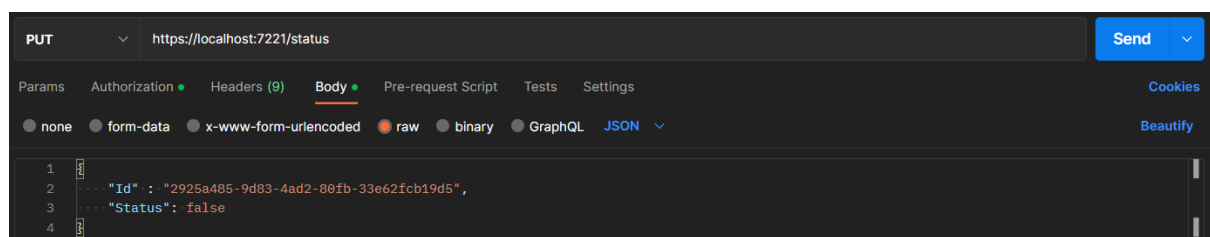
- ☒ Não é permitido modificar a permissão de um usuário que seja Cliente, ao se cadastrar como tal, não é possível alterá-lo;
- ☒ A modificação não será feita caso o campo Tipo de Usuário informado seja igual ao tipo atual daquele usuário (ex. usuário já é Lojista e quero alterar para Lojista);
- ☒ O campo Tipo de Usuário deve ser preenchido com uma das duas opções: "Lojista" ou "Admin".

Caso um ou mais dos critérios acima não sejam cumpridos, o usuário terá um retorno do tipo Status Code 400 - Bad Request. Além desses critérios, caso o usuário esteja logado com uma conta que não seja do tipo Administrador, a edição não será efetuada e caso o ID de usuário informado não seja válido, o retorno será um Status Code 404 - Not Found.

- *EDITAR STATUS DE USUÁRIO - MÉTODO PUT*

URL: `https://localhost:7221/status`

Como Administrador, é possível ativar ou inativar o perfil de outros usuários, independente do tipo dele, seja Cliente, Lojista ou Administrador. Para efetuar essa requisição é necessário estar logado com uma conta do tipo Administrador, além disso é necessário enviar no Body do Postman o ID do usuário que deseja alterar o status e o novo status dele.



Ao enviar a requisição com sucesso, o retorno será um Status Code 204 - No Content.

Caso o usuário já possua o Status informado, a edição não será realizada e o retorno será de um Status Code 400 - Bad Request. Além disso, caso o ID do usuário informado não esteja cadastrado na base de dados o retorno será um Status Code 404 - Not Found. E caso o usuário esteja logado com uma conta que não seja do tipo Administrador, a edição não será efetuada.

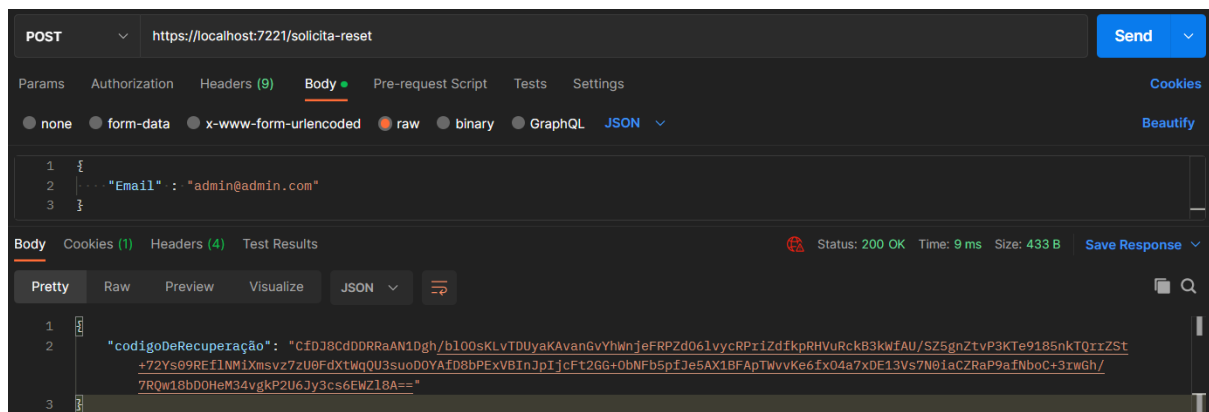
Recuperação de Senha

Se o usuário esquecer a senha cadastrada, é possível recuperá-la com duas requisições. Atenção: é necessário realizar as requisições na ordem descrita neste documento.

- SOLICITAR O RESET DE SENHA - MÉTODO POST

URL: <https://localhost:7221/solicita-reset>

O primeiro passo para recuperar a senha é informar qual o e-mail cadastrado na conta do usuário. Informe o E-mail no Body da requisição.



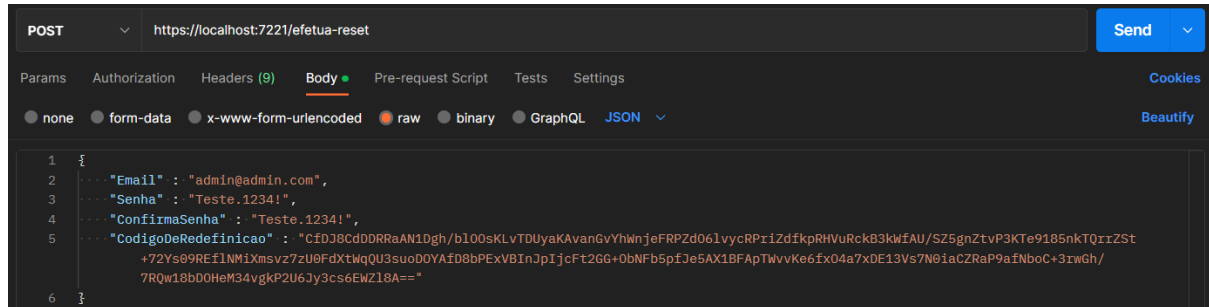
Caso o e-mail informado seja válido, a requisição retornará um Status Code 200 - OK junto a um objeto em JSON contendo o código de recuperação. Este código deve ser utilizado na próxima requisição Efetuar o Reset de Senha.

O retorno será um Status Code 404 - Not Found se o e-mail informado não estiver cadastrado na base de dados.

- EFETUAR O RESET DE SENHA - MÉTODO POST

URL: <https://localhost:7221/efetua-reset>

Para efetivamente resetar a senha, é preciso enviar no Body da requisição alguns campos obrigatórios.



Ao enviar a requisição com sucesso, o retorno será um Status Code 200 - OK junto de uma mensagem de “Senha redefinida com sucesso”.

Caso seja informado um e-mail inválido ou a senha não esteja em um formato válido, o retorno será de Status Code 404 - Not Found e Status Code 400 - Bad Request, respectivamente. Se o código de redefinição não for válido, ou se for solicitado um código com um e-mail e informar outro e-mail na requisição de efetuar o reset de senha, o retorno também será de Status Code 400 - Bad Request com uma mensagem de Token inválido.

Os critérios para redefinição de senha são:

- ☒ Informar um e-mail válido, deve ser o mesmo utilizado na requisição Solicitar o Reset de Senha;
- ☒ A senha deve conter no mínimo 8 caracteres, entre eles uma letra minúscula, uma letra maiúscula, um caractere especial e um número;
- ☒ O campo Confirma Senha deve ser igual ao campo Senha;
- ☒ O Código de Redefinição deve ser válido, ele é obtido ao enviar a requisição de Solicitar o Reset de Senha.

Pesquisa de Usuários

- PESQUISAR USUÁRIOS - MÉTODO GET

URL: <https://localhost:7221/pesquisa/>

Estando logado com uma conta do tipo Administrador, é possível fazer uma pesquisa de todos os usuários cadastrados na base de dados.

Esta pesquisa é feita através de filtros, são eles:

- Nome;
- CPF;
- E-mail;
- Status (filtra os usuários com o status selecionado).

Para a pesquisa, é necessário informar os filtros via URL da requisição, com o nome do filtro seguido de "=" e o valor a ser pesquisado, caso utilize mais de um filtro colocar "&" entre eles.

Alguns critérios devem ser atendidos, são eles:

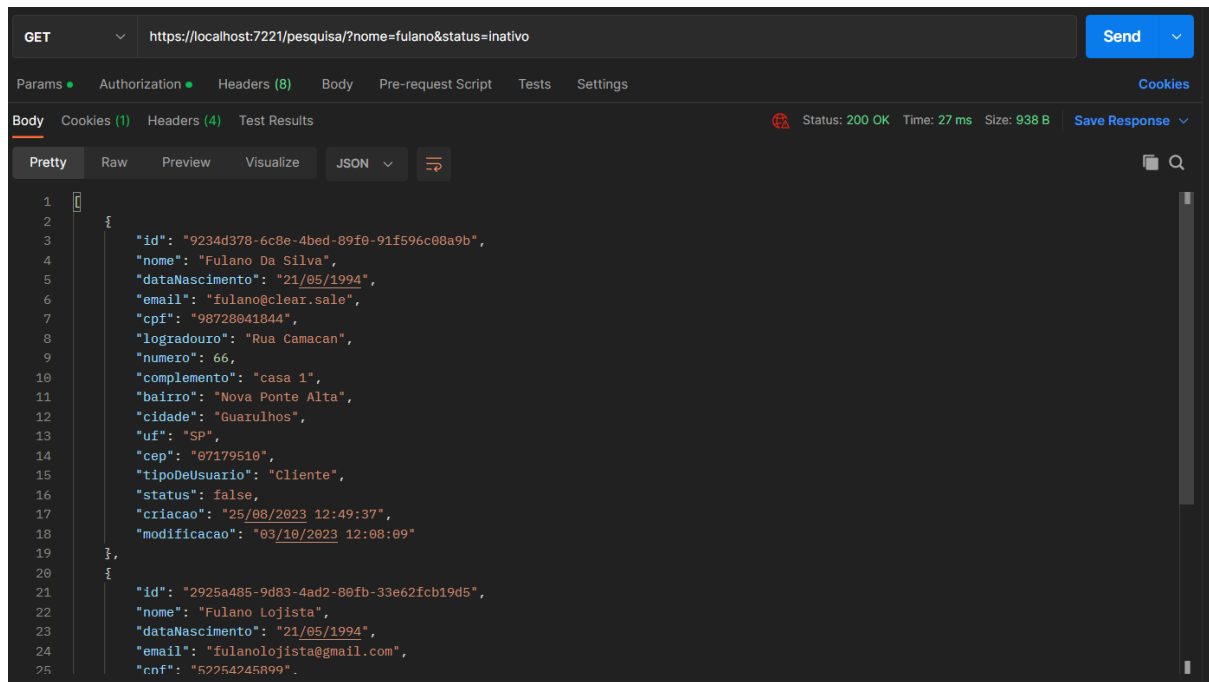
- ☒ Ao pesquisar pelo Nome, o campo deve conter no mínimo 3 e no máximo 250 caracteres;
- ☒ Ao pesquisar pelo CPF, o campo deve conter 11 dígitos, sem pontuação ou hífen;
- ☒ Ao pesquisar pelo E-mail, o mesmo deve possuir um formato válido;
- ☒ Ao pesquisar pelo Status, o campo deve ser preenchido com uma das três opções válidas: 'Ativo', 'Inativo' ou 'Todos'. Caso não seja informado, o status possui o valor padrão 'Todos'.

Os filtros podem ser combinados entre si, e o resultado da requisição será um Status Code 200 - OK junto de uma lista de objetos JSON com as informações resultantes da pesquisa.

Caso a pesquisa não retorne nenhum resultado, a requisição terá um Status Code 404 - Not Found. Se o valor informado em algum dos filtros não for válido, retornará o Status Code 404 - Bad Request com uma mensagem destacando o erro.

Também é possível fazer a pesquisa sem informar nenhum filtro, neste caso, o retorno será uma lista de todos os usuários cadastrados na base.

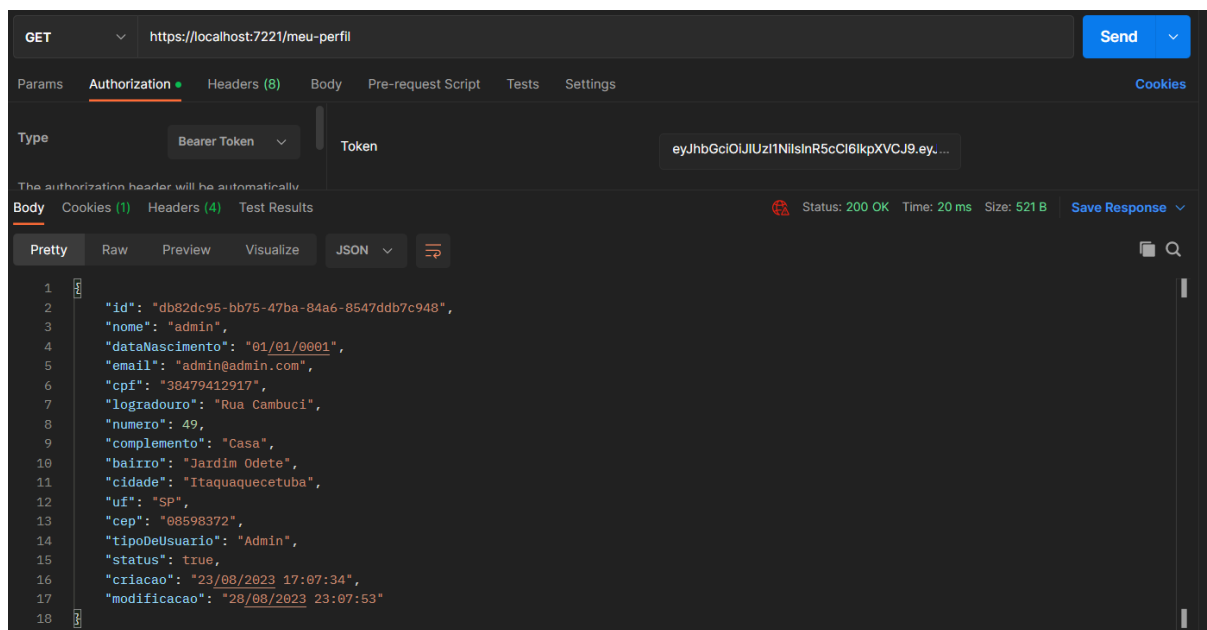
Exemplo de pesquisa com mais de um filtro:



- MEU PERFIL - MÉTODO GET

URL: `https://localhost:7221/meu-perfil`

Para o retorno da requisição Meu Perfil, é necessário informar somente o Token de login do usuário que deseja verificar o perfil. Neste caso, não é necessário passar nenhum parâmetro via URL ou Body da requisição.



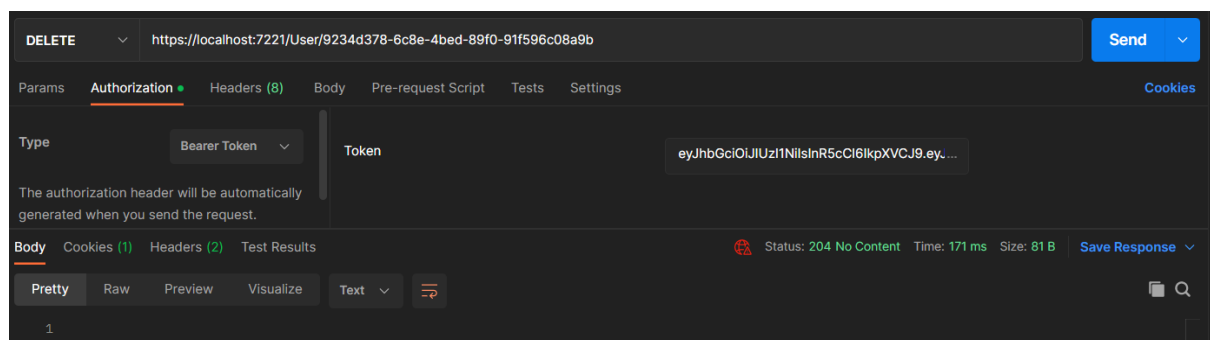
Caso o Token de login informado não seja válido ou não seja informado, o retorno será um Status Code 401 - Unauthorized.

Excluir Usuários

- MÉTODO DELETE

URL: `https://localhost:7221/User/`

Existem duas formas de se excluir um usuário: estando logado com o próprio usuário, ou estando logado com uma conta de Administrador. De qualquer forma, a requisição é feita da mesma maneira. Bastando informar o ID do usuário a ser excluído na URL do método.



Enviando a requisição corretamente, o retorno será um Status Code 204 - No Content, indicando que o usuário foi excluído com sucesso.

Caso o Token de login informado seja de um usuário que não seja o informado no ID ou não seja do tipo Administrador, o resultado será um Status Code 401 - Unauthorized com uma mensagem informando que não é permitido excluir o cadastro de outro usuário.