Issues e melhorias identificados

CT-003 POST/usuarios - Mensagem de erro para e-mail inválido não documentada no Swagger

Descrição Detalhada:

Ao realizar um cadastro de usuário utilizando um e-mail inválido (e.g., "joaoteste.com.br"), a API retorna um erro 400 Bad Request, com a seguinte mensagem: "email": "email deve ser um email válido"

Contudo, essa mensagem de erro não está documentada no Swagger No Swagger, os códigos de resposta esperados são: 201 - Cadastro realizado com sucesso e 400 - E-mail já cadastrado

Passo a Passo para Reproduzir:

1. Abra o Postman .
2. Envie uma requisição POST /usuarios/
3. Inserir Payload com nome,email,password, administrador
4. Clique em **Send** para enviar a requisição.

Resultado Esperado:

- Status Code: 400 Bad Request
- Mensagem de Erro: padronizada e documentada no Swagger, indicando que o e-mail é inválido.

Resultado Atual:

- Status Code: 400 Bad Request
- Mensagem de Erro: "email": "email deve ser um email válido"

Gravidade:

- **Baixa**.

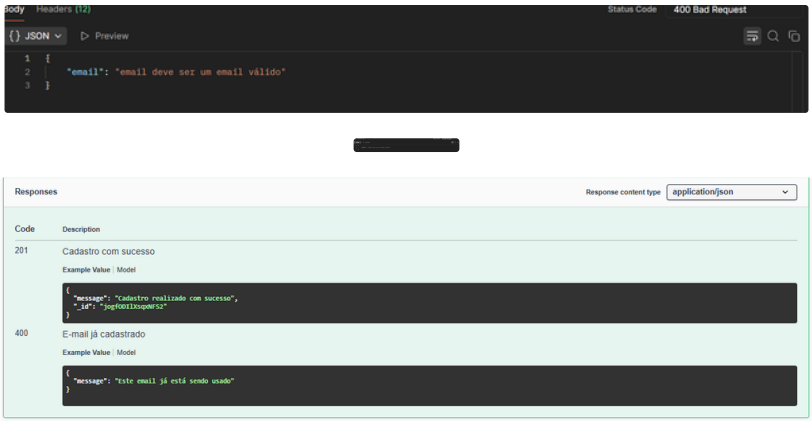
Prioridade:

- **Media**

Ambiente onde ocorre:

- **URL:** https://compassuol.serverest.dev/usuarios/
- **Versão:** 1.0
- **Ambiente:** Desenvolvimento e Homologação

Anexo:



Nota Adicional:

Recomenda-se a atualização da documentação Swagger para refletir as mensagens de erro reais retornadas pela API

CT-004 POST/usuarios -Cadastro permitido com domínio de e-mail bloqueado (Gmail, Hotmail)

Descrição Detalhada:

Ao realizar um cadastro de usuário utilizando um e-mail com domínio bloqueado (e.g., "joao@gmail.com"), a API retorna um status 201 Created, indicando que o cadastro foi realizado com sucesso, mesmo quando o domínio está explicitamente bloqueado.
Conforme as regras de negócio, não deveria ser permitido o cadastro de e-mails com domínios bloqueados, como **Gmail** e **Hotmail**.

Passo a Passo para Reproduzir:

1. Abra o Postman
2. Envie uma requisição **POST**/usuarios: nome,email,password,administrador
3. Verifique o retorno da API.

Resultado Esperado:

- 400 Bad Request
- Mensagem: "Domínio de e-mail não permitido"
- O cadastro não deve ser realizado com domínios de e-mail bloqueados (Gmail, Hotmail)

Resultado Atual:

- Status Code: 201 Created
- Mensagem: "Cadastro realizado com sucesso."

Gravidade:

- **Media**

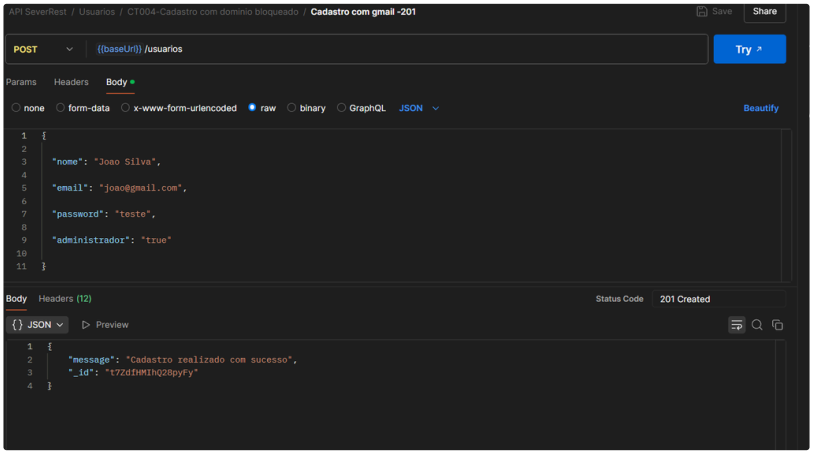
Prioridade:

- **Alta**

Ambiente onde ocorre:

- **URL:** https://compassuol.serverest.dev/usuarios
- **Versão:** 1.0
- **Ambiente:** Desenvolvimento e Homologação

Anexo:



CT-005 POST/usuarios - Mensagens de erro para campos obrigatórios não documentadas no Swagger

Descrição Detalhada:

Ao realizar um cadastro de usuário sem informar os campos obrigatórios (nome, email, password e administrador), a API retorna um erro 400 Bad Request, com a seguinte mensagem:

"nome não pode ficar em branco",

"email": "email não pode ficar em branco",

"password": "password não pode ficar em branco",

"administrador": "administrador deve ser 'true' ou 'false'"

Contudo, essas mensagens de erro não estão documentadas no Swagger, No Swagger, os códigos de resposta esperados são:

201 - Cadastro realizado com sucesso

400 - E-mail já cadastrado

Passo a Passo para Reproduzir:

1. Abra o **Postman** o
2. Envie uma requisição POST /usuarios
3. Inserir os seguintes payload: nome,email,password,administrador
4. Clique em Send para enviar a requisição.

Resultado Esperado:

- **Status Code**: 400 Bad Request
- Mensagem:

  padronizadas e documentadas no Swagger, indicando que os campos obrigatórios não foram informados.

Resultado Atual:

- **S**tatus Code: 400 Bad Request
- **Mensagem:**

  "nome não pode ficar em branco",

  "email": "email não pode ficar em branco",

  "password": "password não pode ficar em branco",

  "administrador": "administrador deve ser 'true' ou 'false'"

  Essas mensagens não estão documentadas no Swagger, gerando falta de alinhamento entre a documentação e o comportamento da API

Gravidade:

- **Alta** -

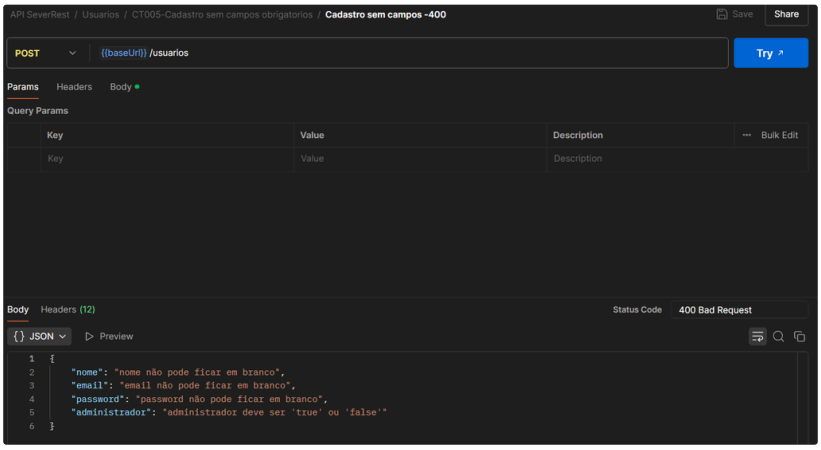
Prioridade:

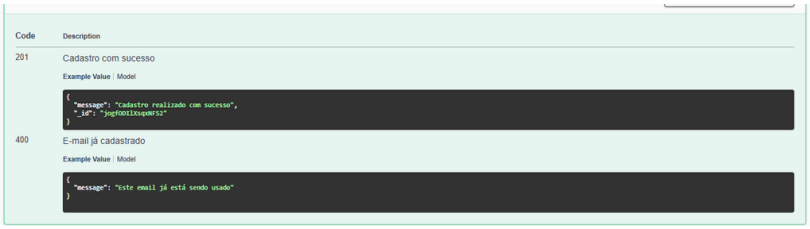
- **Alta**

Ambiente onde ocorre:

- **URL:** https://compassuol.serverest.dev/usuarios/
- **Versão:** 1.0
- **Ambiente:** Desenvolvimento e Homologação

Anexo:





Nota Adicional:

Recomenda-se a atualização da documentação Swagger para refletir as mensagens de erro reais retornadas pela API

CT-006- POST/usuarios - Cadastro permitido com senha fora do limite permitido (menor que 5 ou maior que 10 caracteres)

Descrição Detalhada:

Ao realizar um cadastro de usuário utilizando uma senha com menos de 5 caracteres ou mais de 10 caracteres, a API permite o cadastro com sucesso, retornando um 201 Created com a seguinte mensagem:
"Cadastro realizado com sucesso",

Contudo, o esperado é que a API retorne um erro 400 Bad Request, indicando que a senha deve ter entre 5 e 10 caracteres

Passo a Passo para Reproduzir:

1. Abra o **Postman**
2. Envie uma requisição **POST** /usuarios.
3. enviar um payload: "password": "123"
4. Clique em **Send** para enviar a requisição.

Resultado Esperado:

- **Status Code:** 400 Bad Request
- **Mensagem de Erro:**

  A senha deve ter entre 5 e 10 caracteres

Resultado Atual:

- Status Code: 201 Created
- **Mensagem de Erro:**

  Cadastro realizado com sucesso

Gravidade:

- **Média**

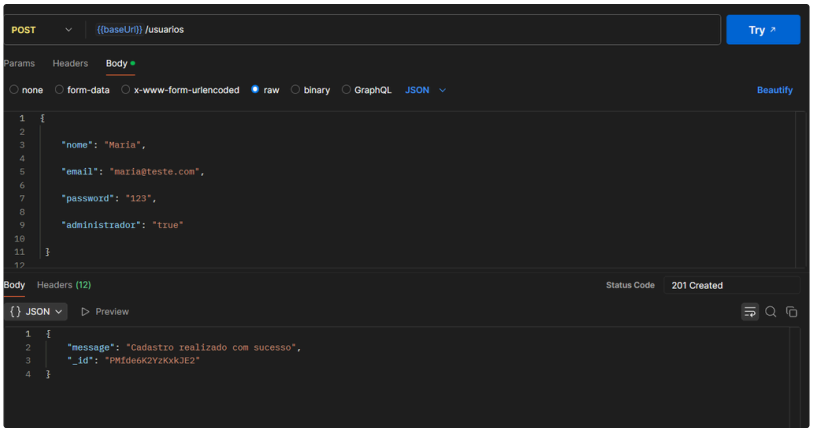
Prioridade:

- **Alta**

Ambiente onde ocorre:

- **URL:** https://compassuol.serverest.dev/usuarios
- **Versão:** 1.0
- **Ambiente:** Desenvolvimento e Homologação

Anexo:



Relação com Outros Defeitos ou Requisitos:

- Validação de tamanho de senha não está sendo respeitada
- Divergência entre comportamento da API e a especificação do teste

Nota Adicional:

Recomenda-se revisar a lógica de validação de tamanho de senha para garantir que o cadastro seja recusado quando estiver fora do intervalo de 5 a 10 caracteres

CT-013 - GET/Usuarios - Listagem de usuários sem autenticação permitida pela API

Descrição Detalhada:

Ao realizar uma requisição para listar os usuários sem enviar um token de autenticação, a API retorna um status 200 OK e lista todos os usuários cadastrados no sistema.
Este comportamento é crítico, pois permite acesso a dados sensíveis sem qualquer tipo de validação, violando princípios de segurança e proteção de dados.

Passo a Passo para Reproduzir:

1. Abra o Postman
2. Envie uma requisição GET/usuarios
3. Não incluir o cabeçalho de autenticação (Authorization Token)
4. Clique em Send para enviar a requisição.

Resultado Esperado:

- Nao autorizado e mensagem indicando que a autenticação é necessária para listar os usuários cadastrados.

Resultado Atual:

Status code: 200 OK

exibe a lista completa de usuários, inclusive com informações sensíveis como nome, email, senha e permissões de administrador

Gravidade:

- Alta

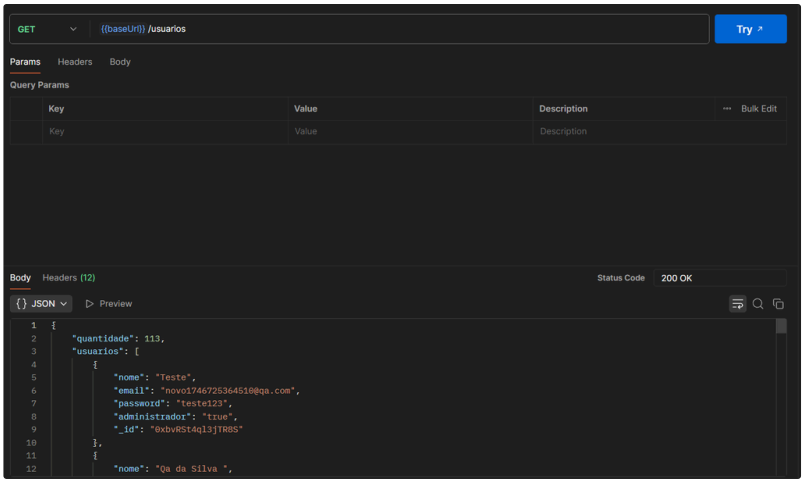
Prioridade:

- Critica

Ambiente onde ocorre:

- URL: https://compassuol.serverest.dev/usuario
- Versão: 1.0
- Ambiente: Desenvolvimento e Homologação

Anexo:



Relação com Outros Defeitos ou Requisitos:

- Exposição de dados sensíveis sem autenticação.
- Violações de segurança e LGPD

Nota Adicional:

Recomenda-se revisar a política de autenticação da API para endpoints de listagem de usuários.l.

CT-021- POST /login - Mensagens de erro para login com campos vazios não documentadas no Swagger

Descrição Detalhada:

Ao realizar uma tentativa de login sem preencher os campos obrigatórios (email e password), a API retorna um erro 400 Bad Request, com a seguinte mensagem:

"email não pode ficar em branco",

"password": "password não pode ficar em branco"

Contudo, essas mensagens de erro não estão documentadas no Swagger,No Swagger, os códigos de resposta esperados são:

200 - Login realizado com sucesso

401 - E-mail e/ou senha inválidos

Passo a Passo para Reproduzir:

1. Abra o Postman
2. Envie uma requisição POST /login:
3. Enviar um payload com
   "email": "",
   "password": ""
4. Clique em Send para enviar a requisição.

Resultado Esperado:

- Status Code: 400 Bad Request
- com mensagens padronizadas e documentadas no Swagger, indicando que os campos obrigatórios não foram informados.

Resultado Atual:

Status code: 400 Bad Request

Mensagem: "email": "email não pode ficar em branco",

"password": "password não pode ficar em branco"

Essas mensagens não estão documentadas no Swagger, gerando falta de alinhamento entre a documentação e o comportamento da API.

Gravidade:

- Baixa

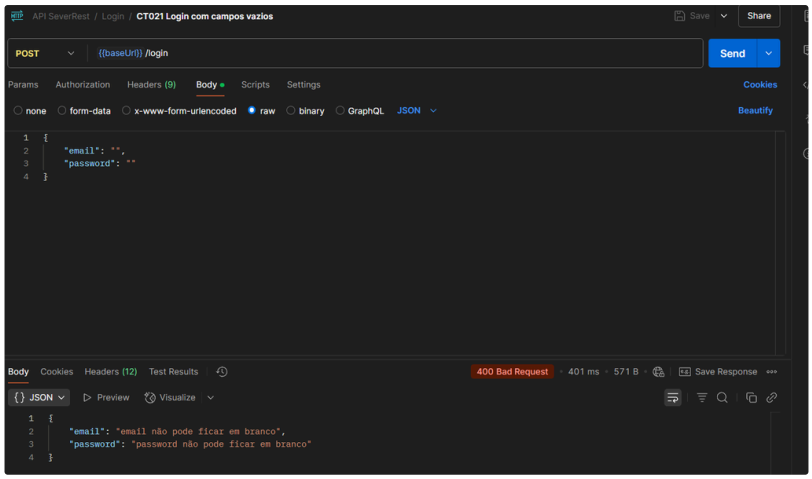
Prioridade:

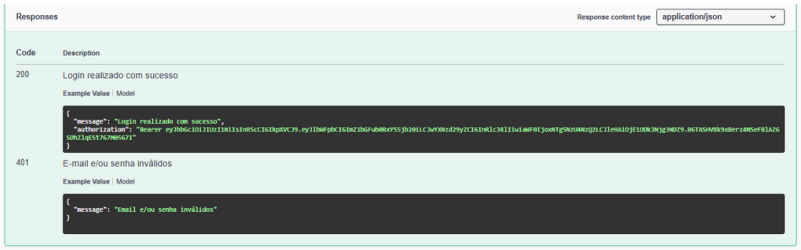
- Media

Ambiente onde ocorre:

- URL: https://compassuol.serverest.dev/login
- Versão: 1.0
- Ambiente: Desenvolvimento e Homologação

Anexo:





Nota Adicional:

Recomenda-se a atualização da documentação Swagger para refletir as mensagens de erro reais retornadas pela API

CT-022- POST/login-Mensagens de erro para login com e-mail mal formatado não documentadas no Swagger

Ao realizar uma tentativa de login utilizando um e-mail mal formatado (e.g., sem o símbolo @), a API retorna um erro 400 Bad Request, com a seguinte mensagem: "email": "email deve ser um email válido"

Contudo, essa mensagem de erro não está documentada no Swagger, No Swagger, os códigos de resposta esperados são:

200 - Login realizado com sucesso

401 - E-mail e/ou senha inválidos

Passos para Reproduzir

1. Abra o Postman
2. Envie uma requisição POST/login
3. Enviar o seguinte payload:
   {

   "email": "Anateste.com.br",

   "password": "teste"

   }
4. Clique em Send.

Resultado Atual

- Status code 400 Bad Request com mensagens padronizadas e documentadas no Swagger, indicando que o formato do e-mail é inválido.

Resultado Esperado

- A API retorna um erro 400 Bad Request com a seguinte mensagem:

  "email": "email deve ser um email válido"

  Essa mensagem não está documentada no Swagger, gerando falta de alinhamento entre a documentação e o comportamento da API.

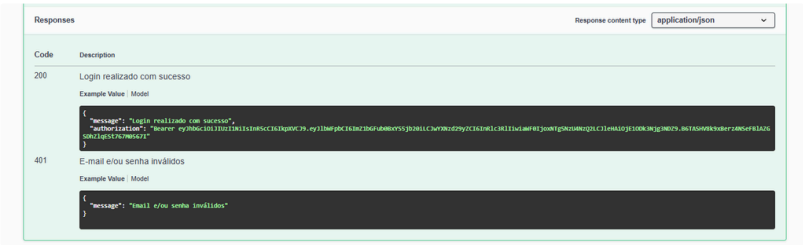
Ambiente onde ocorre:

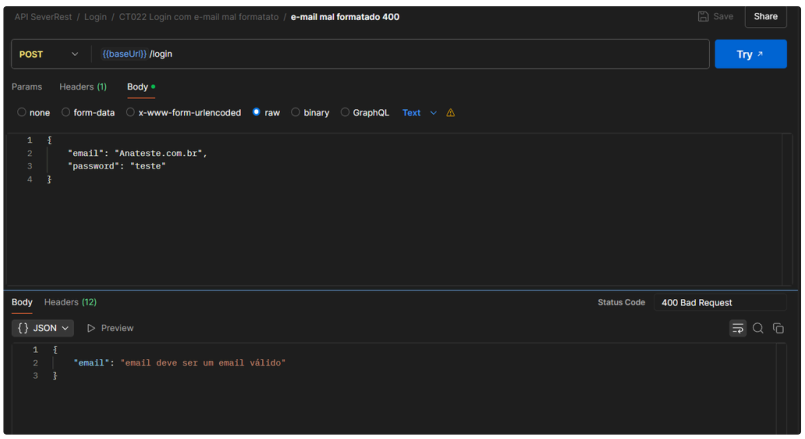
- Ambiente de teste: ServeRest /login
- API Version: 1.0

Gravidade: Baixa

Prioridade: Media

Anexo





Nota Adicional:

- Recomenda-se a atualização da documentação Swagger para refletir as mensagens de erro reais retornadas pela API

CT-028- POST/produto- Mensagem de erro para preço negativo não documentada no Swagger

Descrição

Ao realizar um cadastro de produto utilizando um valor negativo para o campo preco, a API retorna um erro 400 Bad Request, com a seguinte mensagem: "preco": "preço deve ser um número positivo"

Contudo, essa mensagem de erro não está documentada no Swagger, No Swagger, os códigos de resposta esperados são:

201 - Cadastro realizado com sucesso

400 - Já existe produto com esse nome

401 - Token ausente, inválido ou expirado

403 - Rota exclusiva para administradores

Passos para Reproduzir

1. Abra o Postman
2. Envie uma requisição POST/produto
3. Enviar o seguinte payload:
   {

   "nome": "Celular Samsung",

"preco": -1500.00,

"descricao": "Smartphone Samsung Galaxy",

"quantidade": 10

}

4. Clique em Send.

Resultado Atual

- A API deve retornar um erro 400 Bad Request com mensagens padronizadas e documentadas no Swagger, indicando que o preço deve ser um valor positivo.

Resultado Esperado

- A API retorna um erro 400 Bad Request com a seguinte mensagem:

  "preco": "preço deve ser um número positivo"

  Essa mensagem não está documentada no Swagger, gerando falta de alinhamento entre a documentação e o comportamento da API.

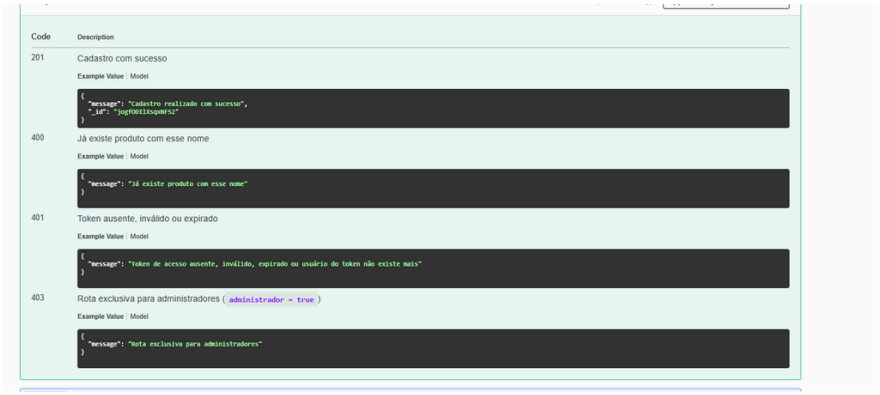
Ambiente onde ocorre:

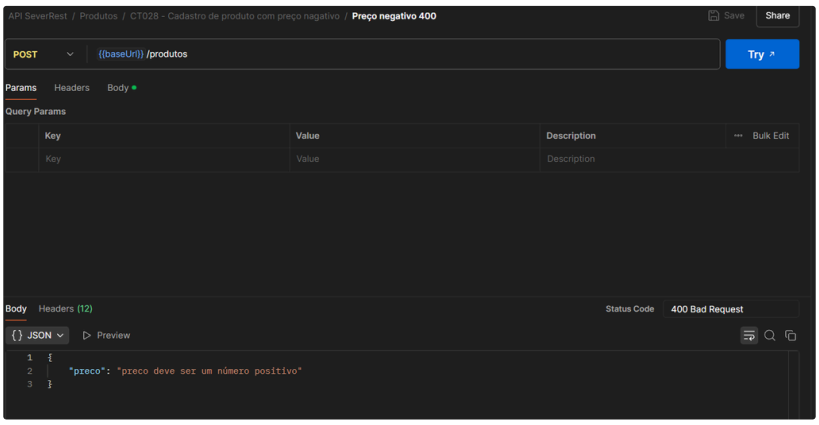
- Ambiente de teste: ServeRest /produtos
- API Version: 1.0

Gravidade: Media

Prioridade: Media

Anexo





Nota Adicional:

- Recomenda-se a atualização da documentação Swagger para refletir as mensagens de erro reais retornadas pela API

CT-037- GET/produtos- Listagem de produtos permitida sem autenticação

Descrição

Ao realizar uma requisição para listar os produtos sem enviar um token de autenticação (JWT), a API retorna um status `200 OK` e lista todos os produtos cadastrados no sistema.

Esse comportamento é um risco de segurança, pois permite a visualização de informações sem autenticação, o que deveria ser restrito por segurança.

Passos para Reproduzir

1. Abra o Postman
2. Envie uma requisição GET/produtos
3. Clique em Send.

Resultado Esperado

A API deve retornar um erro 401 Unauthorized ou 403 Forbidden, indicando que a autenticação é necessária para listar os produtos cadastrados

Resultado Atual

- A API retorna um status `200 OK` e exibe a lista completa de produtos, incluindo nome, preço, descrição e quantidade, mesmo sem um token de autenticação.

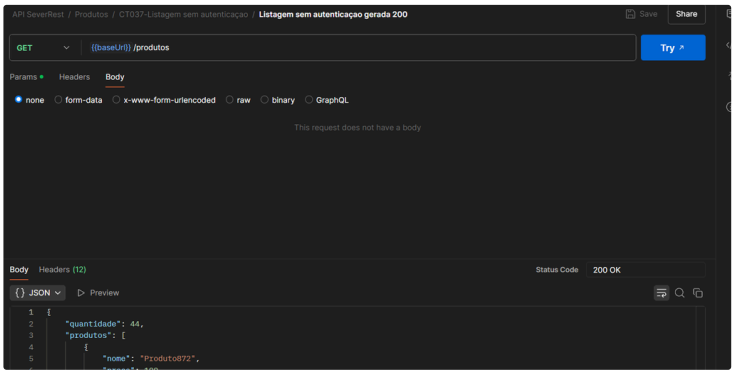
Ambiente onde ocorre:

- Ambiente de teste: ServeRest /produtos
- API Version: 1.0

Gravidade: Media

Prioridade: Alta

Anexo



Relação com Outros Defeitos ou Requisitos:

- Exposição de dados sensíveis sem autenticação.
- Divergência entre comportamento da API e a especificação do teste

Nota Adicional:

- Recomenda-se revisar a política de autenticação da API para endpoints de listagem de produtos.

CT-054- DELETE/carrinho- Mensagem de cancelamento de compra divergente do Swagger3

Ao realizar o cancelamento de uma compra com carrinho válido, a API retorna um status 200 OK com a seguinte mensagem: “Registro excluído com sucesso. Estoque dos produtos reabastecido"

Contudo, essa mensagem não está documentada no Swagger. De acordo com a documentação, o retorno esperado seria:

200 - Registro excluído com sucesso | Não foi encontrado carrinho para esse usuário

A mensagem apresentada na API está correta em seu conteúdo, mas divergente da documentação oficial.

Passos para Reproduzir

1. Abra o Postman
2. Envie uma requisição DELETE/carrinhos/cancelar-compra
3. Clique em Send.

Resultado Esperado

A API deve retornar um status `200 OK` com uma mensagem padronizada e documentada no Swagger, indicando o cancelamento da compra e a reposição do estoque

Resultado Atual

- A API retorna um status `200 OK` com a seguinte mensagem:
  "message": "Registro excluído com sucesso. Estoque dos produtos reabastecido"
  Essa mensagem não está documentada no Swagger, gerando falta de alinhamento entre a documentação e o comportamento da API

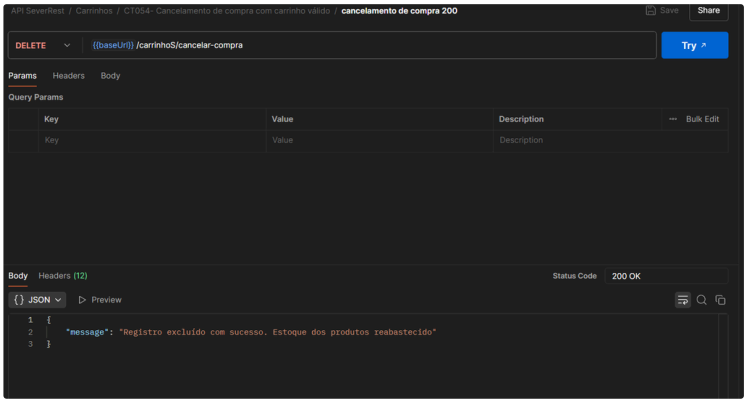
Ambiente onde ocorre:

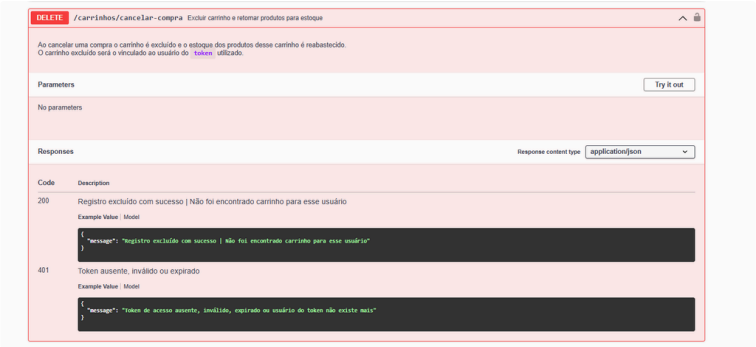
- Ambiente de teste: ServeRest /carrinho
- API Version: 1.0

Gravidade: Media

Prioridade: Media

Anexo





Relação com Outros Defeitos ou Requisitos:

- **Divergência entre comportamento da API e a especificação do Swagger**
- **Inconsistência na mensagem de sucesso documentada**

Nota Adicional:

- Recomenda-se a atualização da documentação Swagger para refletir a mensagem real retornada pela API ou ajustar o retorno para ficar alinhado com a documentação