**Plano de Testes da API REST Serverest**

Versão: 2.29.7

URL: https://compassuol.serverest.dev

# Apresentação

Este plano de testes visa validar funcionalidades críticas da API ServeRest (versão 2.29.7), garantindo a aderência às regras de negócio nos seguintes módulos: Usuários, Login, Produtos e Carrinhos.

# Objetivo

Assegurar que todos os endpoints funcionem corretamente de acordo com as especificações da API, com foco em:

- Validação de entradas (dados válidos e inválidos)

- Regras de negócio (e-mails únicos, login autorizado, produto não duplicado)

- Segurança e controle de acesso (token Bearer)

# Escopo

Módulos cobertos:

- Login: autenticação, token válido, falhas de login

- Usuários: CRUD, regras de validação de e-mail/senha, permissões

- Produtos: CRUD, nomes únicos, dependências com carrinhos

- Carrinhos: compra, deleção, uso de produtos válidos

# Ambiente de Testes

Hardware: Computadores

Software: Navegadores (Chrome), Postman (testes de API), Swagger (Serverest)

# Análise

O que será feito:

- Testes manuais e automatizados com Postman, baseados no Swagger e casos de uso reais.

- Criação de dados dinâmicos para evitar conflitos.

- Validação de mensagens de erro, status HTTP e integridade de dados.

Motivos:

- Garantir que o sistema impeça ações não autorizadas.

- Cobrir cenários comuns e alternativos (ex: duplicidade, dados inválidos).

- Atuar preventivamente contra falhas em produção.

# Mapa mental da aplicação

Diagrama

O conteúdo gerado por IA pode estar incorreto.

# Cenários de Teste

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Cenário** | **Ação (Passo)** | **Resultado Esperado** |
| TC01 | DELETE produto sem autenticação | Enviar DELETE /produtos/:id sem token | Acesso negado |
| TC02 | Buscar usuário com ID inexistente | Enviar GET /usuarios/:id\_inexistente | Retornar erro "usuário não encontrado" |
| TC03 | Criar usuário com senha curta | Enviar POST /usuarios com senha de 4 caracteres | Erro de validação de senha |
| TC04 | Produto não aceita valores float | Enviar POST /produtos com campo preco como 10.5 | Erro de validação: valor deve ser inteiro |
| TC05 | Criar usuário com e-mail do gmail | Enviar POST /usuarios com e-mail @gmail.com | Erro por provedor de e-mail não aceito |
| TC06 | Criar usuário com e-mail duplicado | Enviar POST /usuarios com e-mail já usado | Erro informando e-mail duplicado |
| TC07 | Criar usuário com dados válidos | Enviar POST /usuarios com dados válidos | Usuário criado com sucesso |
| TC08 | DELETE produto em carrinho | Enviar DELETE /produtos/:id | Erro informando dependência de carrinho |
| TC09 | PUT de produto com nome já existente | Enviar PUT /produtos/:id com nome já usado | Erro de nome duplicado |
| TC10 | PUT de produto com nome único | Enviar PUT /produtos/:id com nome válido | Produto atualizado com sucesso |
| TC11 | Criar produto sem autenticação | Enviar POST /produtos sem token | Acesso negado |
| TC12 | Criar produto com nome já utilizado | Enviar POST /produtos com nome repetido | Erro de nome duplicado |
| TC13 | Criar produto com nome único | Enviar POST /produtos com nome não utilizado | Produto criado com sucesso |
| TC14 | Login com senha incorreta | Enviar POST /login com senha incorreta | Não autorizado |
| TC15 | Login com e-mail não cadastrado | Enviar POST /login com e-mail inexistente | Não autorizado |
| TC16 | Login com dados válidos | Enviar POST /login com e-mail e senha corretos | Retorna token Bearer válido |
| TC17 | PUT com e-mail já usado por outro usuário | Enviar PUT /usuarios/:id com e-mail já usado | Erro de e-mail duplicado |
| TC18 | PUT com ID inexistente | Enviar PUT /usuarios/:id\_inexistente com dados válidos | Novo usuário criado |
| TC19 | Atualizar usuário existente com PUT | Enviar PUT /usuarios/:id com dados válidos | Dados atualizados com sucesso |
| TC20 | Criar usuário com e-mail inválido | Enviar POST /usuarios com e-mail mal formatado | Erro de validação de e-mail |
| TC21 | Criar usuário com e-mail do gmail | Enviar POST /usuarios com e-mail @gmail.com | Erro por provedor de e-mail não aceito |
| TC22 | Criar usuário com dados válidos | Enviar POST /usuarios com dados válidos | Usuário criado com sucesso |
| TC23 | Criar usuário com e-mail duplicado | Enviar POST /usuarios com e-mail já usado | Erro informando e-mail duplicado |

# Técnicas de Teste Aplicadas

|  |  |
| --- | --- |
| **ID** | **Técnica de Teste Aplicada** |
| TC01 | Teste de Autorização |
| TC02 | Classe de Valor Inválido |
| TC03 | Análise do Valor Limite (AVL) |
| TC04 | Particionamento de Equivalência (PE) |
| TC05 | Particionamento de Equivalência (PE) |
| TC06 | Classe de Valor Inválido |
| TC07 | Caminho Feliz / Fluxo Principal |
| TC08 | Teste de Condição de Erro |
| TC09 | Classe de Valor Inválido |
| TC10 | Caminho Feliz / Fluxo Principal |
| TC11 | Teste de Autorização |
| TC12 | Classe de Valor Inválido |
| TC13 | Caminho Feliz / Fluxo Principal |
| TC14 | Teste de Autenticação |
| TC15 | Teste de Autenticação |
| TC16 | Caminho Feliz / Fluxo Principal |
| TC17 | Classe de Valor Inválido |
| TC18 | Tabela de Decisão |
| TC19 | Caminho Feliz / Fluxo Principal |
| TC20 | Particionamento de Equivalência (PE) |
| TC21 | Particionamento de Equivalência (PE) |
| TC22 | Caminho Feliz / Fluxo Principal |
| TC23 | Classe de Valor Inválido |

# Priorização – Casos de Teste

|  |  |  |
| --- | --- | --- |
| **ID** | **Criticidade** | **Justificativa** |
| TC01 | Alta | Criação de usuário válida é essencial para acesso ao sistema. |
| TC02 | Alta | E-mails duplicados causam falhas graves na integridade dos dados. |
| TC03 | Média | Uso de provedores específicos pode ser uma regra de negócio opcional. |
| TC04 | Média | Mesmo caso do TC03; não impede o uso do sistema por completo. |
| TC05 | Alta | E-mails inválidos podem comprometer autenticação e notificações. |
| TC06 | Alta | Senhas curtas comprometem a segurança do sistema. |
| TC07 | Média | Regras de senha longa são importantes, mas com menor impacto. |
| TC08 | Alta | Atualização de dados é função crítica para manutenção de usuários. |
| TC09 | Média | Criar novo usuário via PUT com ID inexistente pode causar confusão. |
| TC10 | Alta | Evitar duplicidade de e-mails é fundamental para a unicidade dos dados. |
| TC11 | Média | A consulta por ID inválido não compromete o sistema, mas deve retornar erro adequado. |
| TC12 | Alta | Login é funcionalidade essencial para uso do sistema. |
| TC13 | Alta | Impede acesso indevido e garante segurança. |
| TC14 | Alta | Testar senha incorreta é crucial para controle de acesso. |
| TC15 | Média | A validade do token é importante, mas não impede uso imediato. |
| TC16 | Alta | Criar produtos é função essencial do sistema para administradores. |
| TC17 | Alta | Produtos duplicados afetam integridade e podem confundir clientes. |
| TC18 | Alta | Impedir ações sem autenticação protege o sistema de uso indevido. |
| TC19 | Alta | Atualizações válidas garantem a manutenção correta de produtos. |
| TC20 | Média | Criação involuntária via PUT deve ser tratada com atenção. |
| TC21 | Alta | Evita conflitos de nomes e problemas com controle de estoque. |
| TC22 | Alta | Remoção indevida de produto em uso pode causar erros em pedidos. |
| TC23 | Alta | Exigir autenticação para ações críticas é essencial para segurança. |

# Matriz de Risco

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Probabilidade** | **Impacto** | **Risco** |
| TC01 | Alta | Alto | Alto |
| TC02 | Alta | Alto | Alto |
| TC03 | Média | Médio | Médio |
| TC04 | Média | Médio | Médio |
| TC05 | Alta | Alto | Alto |
| TC06 | Alta | Alto | Alto |
| TC07 | Média | Médio | Médio |
| TC08 | Alta | Alto | Alto |
| TC09 | Média | Médio | Médio |
| TC10 | Alta | Alto | Alto |
| TC11 | Média | Médio | Médio |
| TC12 | Alta | Alto | Alto |
| TC13 | Alta | Alto | Alto |
| TC14 | Alta | Alto | Alto |
| TC15 | Média | Médio | Médio |
| TC16 | Alta | Alto | Alto |
| TC17 | Alta | Alto | Alto |
| TC18 | Alta | Alto | Alto |
| TC19 | Alta | Alto | Alto |
| TC20 | Média | Médio | Médio |
| TC21 | Alta | Alto | Alto |
| TC22 | Média | Alto | Alto |
| TC23 | Alta | Alto | Alto |

# Cobertura de Testes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Módulo** | **Total Casos** | **Passaram** | **Falharam** | **Cobertura (%)** |
| Usuário | 11 | 9 | 2 | 48% |
| Produto | 9 | 8 | 1 | 39% |
| Login | 3 | 3 | 0 | 13% |
| **Total Geral** | **23** | **20** | **3** | 100% |

|  |  |  |
| --- | --- | --- |
| **Resultado** | **Testes** | **%** |
| Passou | 20 | 87% |
| Falhou | 3 | 13% |
|  | 23 | 100% |

# Testes Candidatos à Automação

|  |  |
| --- | --- |
| **Caso de Teste** | **Por que automatizar?** |
| Criar usuário com dados válidos | Repetido com frequência |
| Criar usuário com e-mail já usado | Regra de negócio importante |
| Login com dados corretos | Usado em vários fluxos |
| Login com senha incorreta | Valida segurança básica |
| Criar produto com nome já existente | Previne duplicidade |
| Criar produto sem autenticação | Teste essencial de segurança |
| Excluir produto em carrinho | Regressivo e complexo |

# Principais alterações

Neste plano de testes da API ServeRest, foram realizadas melhorias para torná-lo mais claro e útil. As principais alterações incluem:

* Maior legibilidade do documento;
* Inclusão dos resultados dos testes executados;
* Adição dos endpoints utilizados em alguns testes;
* Foco no fluxo principal de compra, do cadastro à finalização.

Essas mudanças visam garantir mais objetividade, rastreabilidade e cobertura nos testes da API