

PLANO DE TESTE

Logical Forest

Compass - UOL

Challenge Sprint 6

Luann Lucas

JUN 22, 2022

1. Introdução

Esse plano de teste corresponde ao funcionamento da [ServeRest](#), que para consulta, inserção, alteração e exclusão dos dados, a API disponibiliza um conjunto de serviços REST em que o cliente (aplicação) envia uma requisição à uma URL (endpoint) para acessar os serviços da API e tem como retorno um objeto na forma de um JSON.

1.1. Objetivo

A API de simulação de e-commerce Serverest, baseia seu objetivo em um fluxo completo de compras.

1.2. Referências

[Teste de Software: Diferentes Tipos de Teste \(devmedia.com.br\)](#)

[Plano de Teste - Um Mapa Essencial para Teste de Software \(devmedia.com.br\)](#)

[Como testar APIs com Cypress – TALKING ABOUT TESTING](#)

[Automatizando testes. em APIs REST. com Cypress e NodeJS | by Alex Alves | Medium](#)

2. Requisitos para o Teste

A seguir itens (requisitos funcionais, suíte de casos de teste , candidatos a automação) que foram identificados como alvos do teste. O mapa mental (2.2) serve de auxílio para entender as funcionalidades da API e o que é prioritário para os testes (localizado à direita do mapa).

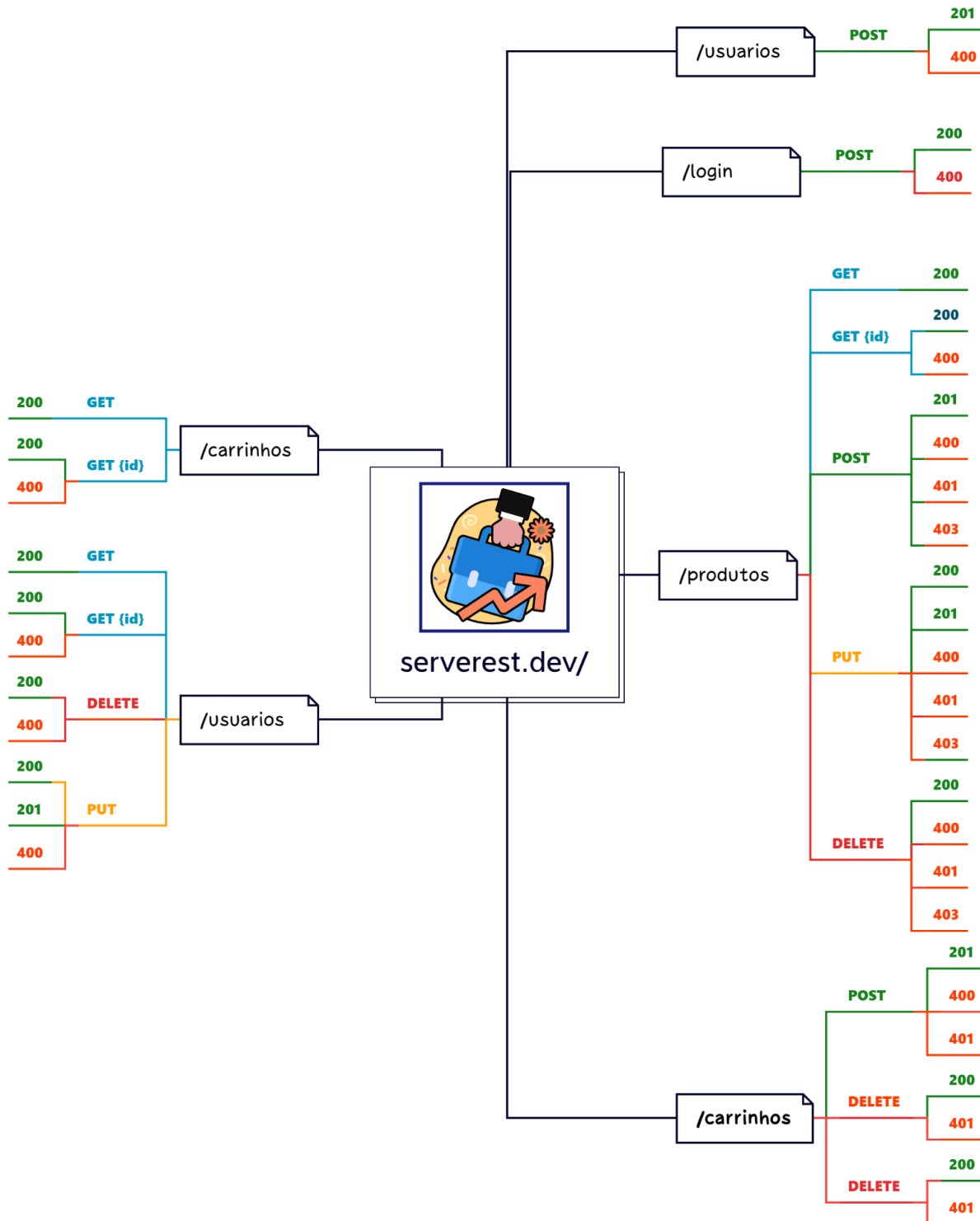
2.1. Requisitos funcionais

REQUISITOS	DESCRIÇÃO
RF001 - Usuários	O sistema deve permitir a consulta, inserção, alteração e exclusão dos dados dos usuários. Cada usuário cadastrado terá um email e senha para efetuar o login.
RF002 - Níveis de acesso	O sistema deve permitir inserção e alteração do nível de acesso (RF001). Os níveis possíveis são administrativo e padrão.

RF003 - Acesso	O sistema deve vetar a autenticação com login e senha em casos de consulta, inserção, exclusão e alteração de dados na rota dos usuários. E apenas consultas nas outras rotas.
RF003 - Login	O sistema deve permitir a autenticação de usuário e suas permissões por níveis de acesso (RF002) ao logar com email e senha cadastrados.
RF004 - Produtos	A nível administrativo (RF002) o sistema deve permitir a inserção, alteração e exclusão dos dados dos produtos. E em ambos os níveis a consulta.
RF005- Carrinhos	O sistema deve permitir a consulta, inserção e exclusão de carrinhos. A exclusão deve ser feita com a conclusão ou o cancelamento da compra.

2.2. Mapa mental

Será considerado prioritário as requisições que se encontram à direita desse mapa mental.



Presented with XMind

2.3. Suíte de casos de teste

CT-RU 001 - INSERÇÃO DE DADOS EM /USUÁRIOS

CASO DE TESTE (ID)	TÍTULO	MÉTODO	ROTA	RESPONSE E STATUS CODE
CT001	Cadastrar um novo usuário com sucesso	POST	/usuarios	{ 'Cadastro realizado com sucesso' } 201
CT002	Não deve cadastrar com email já existente	POST	/usuarios	{ 'Este email já está sendo usado' } 400
CT003	Não deve cadastrar usuário sem email	POST	/usuarios	Não documentado
CT004	Não deve cadastrar usuário sem nome	POST	/usuarios	Não documentado
CT005	Não deve cadastrar usuário sem senha	POST	/usuarios	Não documentado
CT006	Não deve cadastrar usuário sem informar administrador	POST	/usuarios	Não documentado
CT007	Não deve cadastrar usuário sem nenhuma informação passada	POST	/usuarios	Não documentado

CT-RU 002 - CONSULTA DE DADOS EM /USUÁRIOS

CT008	Deve buscar todos os usuários cadastrados	GET	/usuarios	required: ["quantidade", "usuarios"] 200
CT009	Deve buscar usuário por id	GET	/usuarios	required: ["nome", "email", "password", "administrador", "_id"] 200

CT-RU 003 - EXCLUSÃO DE DADOS EM /USUÁRIOS

CT010	Deve deletar usuário com sucesso	DELETE	/usuarios	{ "message": "Registro excluído com sucesso Nenhum registro excluído" } 200
CT011	Não deve deletar usuário com carrinho cadastrado	DELETE	/usuarios	required: ["Message": "Login realizado com sucesso", "_idCarrinho":] 400

CT-RU 004 - ALTERAÇÃO DE DADOS EM USUÁRIOS

CT012	Deve alterar dados de um usuário com sucesso	PUT	/usuarios	{ "message": "Registro alterado com sucesso" } 200
CT013	Não deve alterar dados de um usuário com email já cadastrado	PUT	/usuarios	{ "message": "Este email já está sendo usado" } 400
CT014	Deve cadastrar um novo usuário	PUT	/usuarios	required: ["Message": "Login realizado com sucesso",] 200

	se o id for inválido			<pre>"_id":] 201</pre>
--	----------------------	--	--	-------------------------

CT-RL 005 - INSERÇÃO DE DADOS EM /LOGIN

CT015	Deve logar com sucesso	POST	/login	<pre>required: ["Message": "Login realizado com sucesso", "authorization":]</pre> 200
CT016	Não deve logar com senha e/ou email inválido	POST	/login	<pre>{ "Email e/ou senha inválidos" }</pre> 400
CT017	Não deve logar com email inválido	POST	/login	<pre>{ "Email e/ou senha inválidos" }</pre> 400
CT018	Não deve logar com email em branco	POST	/login	Não documentado
CT019	Não deve logar com senha em branco	POST	/login	Não documentado

CT-RP 006 - INSERÇÃO DE DADOS EM /PRODUTOS

CT020	Deve cadastrar um produto com sucesso	POST	/produtos	<pre>{ Cadastro realizado com sucesso }</pre> 201
CT021	Não deve cadastrar produto com nome existente	POST	/produtos	<pre>{ Já existe produto com esse nome }</pre> 400
CT022	Não deve cadastrar produto com token adm inválido	POST	/produtos	<pre>{ Token de acesso ausente, inválido, expirado ou usuário do token não existe mais }</pre> 401

CT023	Não deve cadastrar com token adm vazio	POST	/produtos	{ Token de acesso ausente, inválido, expirado ou usuário do token não existe mais } 401
CT024	Não deve cadastrar produto sem usuário adm	POST	/produtos	Rota exclusiva para administradores 403

CT-RP 007 - CONSULTA DE DADOS EM /PRODUTOS

CT025	Deve buscar por todos os produtos	GET	/produtos	required: ["quantidade", "Produtos"] 200
CT026	Deve buscar produtos por id	GET	/produtos/{_id}	required: ["nome", "preco", "descricao", "quantidade", "_id"] 200

CT-RP 008 - EXCLUSÃO DE DADOS EM /PRODUTOS

CT027	Deve deletar um produto	DELETE	/produtos/{_id}	{ Registro excluído com sucesso } 201
CT028	Não deve deletar produto em carrinho	DELETE	/produtos/{_id}	{ Não é permitido excluir produto que faz parte de carrinho } 400
CT029	Não deve deletar produto com token inválido	DELETE	/produtos/{_id}	{ Token de acesso ausente, inválido, expirado ou usuário do token não existe mais } 401
CT030	Não deve deletar produto com token vazio	DELETE	/produtos/{_id}	{ Token de acesso ausente, inválido, expirado ou usuário do token não existe mais } 401

CT031	Não deve deletar produtos com usuário padrão	DELETE	/produtos/{id}	{ Rota exclusiva para administradores } 403
--------------	--	---------------	-----------------------	---

CT-RP 009 - ALTERAÇÃO DE DADOS EM /PRODUTOS

CT032	Deve atualizar nome de um produto	PUT	/produtos/{id}	{ Registro alterado com sucesso } 200
CT033	Não deve atualizar dados de produtos sem token de adm	PUT	/produtos/{id}	{ Já existe produto com esse nome } 400
CT034	Não deve atualizar dados de produtos com token vazio	PUT	/produtos/{id}	{ Token de acesso ausente, inválido, expirado ou usuário do token não existe mais } 401
CT035	Não deve atualizar dados de produtos com usuário padrão	PUT	/produtos/{id}	{ Rota exclusiva para administradores } 403

CT-RC 009 - INSERÇÃO DE DADOS EM /CARRINHOS

CT036	Cadastrar um carrinho	POST	/carrinhos	{ Cadastro realizado com sucesso } 201
--------------	-----------------------	-------------	-------------------	--

CT-RC 010 - EXCLUSÃO DE DADOS EM /CARRINHOS

CT039	Deve concluir a compra	DELETE	/carrinhos	{ Registro excluído com sucesso } 200
--------------	------------------------	---------------	-------------------	---

CT040	Deve concluir a compra	DELETE	/carrinhos	{ Registro excluído com sucesso. Estoque dos produtos reabastecido } 200
--------------	------------------------	---------------	-------------------	---

2.4. Candidatos a automação

GRUPO (ID)	TÍTULO	NÍVEL DE PRIORIDADE
CT-RU 001	INSERÇÃO DE DADOS EM /USUÁRIOS	ALTA
CT-RU 005	INSERÇÃO DE DADOS EM /LOGIN	ALTA
CT-RP 006	INSERÇÃO DE DADOS EM /PRODUTOS	ALTA
CT-RP 007	CONSULTA DE DADOS EM /PRODUTOS	MÉDIA
CT-RP 008	EXCLUSÃO DE DADOS EM /PRODUTOS	MÉDIA
CT-RP 009	ALTERAÇÃO DE DADOS EM /PRODUTOS	MÉDIA
CT-RC 009	-INSERÇÃO DE DADOS EM /CARRINHOS	ALTA
CT-RC 010	EXCLUSÃO DE DADOS EM /CARRINHOS	ALTA

3. Estratégia de teste

Serão desenvolvidos testes automatizados (2.4).

3.1. Escopo

Assume-se que o teste de unidade feito pelo time de desenvolvimento já forneceu uma cobertura extensiva do código fonte. A fim de minimizar o risco, aumentar a eficácia e comportamento do código neste primeiro momento.

Será então feito:

- Teste de integração: os testes de endpoints da API serão executados pelo time de qualidade, e serão verificados os status code esperados, respostas JSON para uma requisição..

3.2. Ferramentas

FERRAMENTA	FUNCIONALIDADE
CYPRESS	Desenvolvimento e execução de testes automatizados
NODE.JS	Ambiente de execução Javascript server-side
VS CODE	Desenvolvimento e edição de código
JAVASCRIPT	Linguagem que os testes foram desenvolvidos
GIT e GITHUB	Versionamento de código

3.3. Classificação de bugs

TIPO	SEVERIDADE	DESCRIÇÃO
1	Blocker	<ul style="list-style-type: none">• Bug que impede o fluxo de compra
2	Critical	Funciona pobremente e impede o uso <ul style="list-style-type: none">• Efeitos indesejados
3	Major	Sua funcionalidade principal não é afetada <ul style="list-style-type: none">• Mensagem de erro ou sucesso não é exibida• Input incomum causar efeitos indesejados, mas contornáveis
4	Minor	Quase nenhum impacto na funcionalidade, mas ainda é um erro <ul style="list-style-type: none">• Erro ortográfico• Response não documentada mas é algo que deve ser esperado

3.4. Definição de Pronto:

Será considerada pronta as funcionalidades que passarem pelas verificações descritas neste Plano de teste que não apresentarem bugs com a severidade acima de Menor.

4. Relatorio de bugs

ROTA /LOGIN - MÉTODO POST

ID	DESCRIÇÃO	ID-BUG	TIPO	RESPONSE
CT018	Não deve logar com email em branco	RB001	4	'email não pode ficar em branco' 400
CT017	Não deve logar com email inválido	RB002	4	email: "email deve ser um email válido" 400
CT019	Não deve logar com senha em branco	RB003	4	'password não pode ficar em branco' 400

ROTA /USUARIOS - MÉTODO POST

ID	DESCRIÇÃO	ID-BUG	TIPO	RESPONSE
CT003	Não deve cadastrar usuário sem email	RB004	4	'email é obrigatório' 400
CT004	Não deve cadastrar usuário sem nome	RB005	4	'nome é obrigatório' 400
CT005	Não deve cadastrar usuário sem senha	RB006	4	'password é obrigatório' 400
CT006	Não deve cadastrar usuário sem informar administrador	RB007	4	'administrador é obrigatório' 400
CT007	Não deve cadastrar usuário sem nenhuma informação passada	RB008	4	'email é obrigatório' 'nome é obrigatório' 'password é obrigatório' 'administrador é obrigatório' 400

