

Planejamento de Teste – ServeRest

Squad: String Testers (QA)

Site avaliado:  [Front - ServeRest](#)

Bruno, Amilly, Caio, Eduardo, Ana Livia

21 de Agosto de 2025

Bloco 1 - Bruno:

Estilo de teste: **Null, Zero, Vazio;**

1. Objetivo: Validar o comportamento do sistema **ServeRest** nas funcionalidades de **Login** e **Cadastro de Produtos**, garantindo que as validações de obrigatoriedade, formato e consistência sejam aplicadas corretamente. Com o estilo Null, nos campos numéricos colocamos 0, campo de texto colocaríamos vazio e espaço, com os demais testes.

2. Escopo:

- **Funcionalidade Login**
- **Funcionalidade Cadastro de Produtos**

3. Cenários de Teste:

Login

1. Campos vazios

- **Ação:** Deixar e-mail e senha em branco.
- **Resultado Esperado:** Sistema deve exibir mensagem: “E-mail e senha são obrigatórios”.

2. Formato inválido de e-mail

- **Ação:** Inserir caracteres especiais sem `@.com`.
- **Resultado Esperado:** Sistema deve exibir mensagem: “Formato de e-mail inválido”.

3. E-mail inexistente

- **Ação:** Inserir e-mail não cadastrado com qualquer senha.
- **Resultado Esperado:** Sistema deve exibir mensagem: “Usuário ou senha inválidos”.

4. Senha incorreta para e-mail válido

- **Ação:** Inserir e-mail válido cadastrado + senha incorreta.
 - **Resultado Esperado:** Sistema deve exibir mensagem genérica: “Usuário ou senha inválidos”.
-

5. Campos vazios

- **Ação:** Tentar cadastrar produto com nome, preço, descrição e quantidade em branco.
- **Resultado Esperado:** Sistema deve exibir mensagem de obrigatoriedade em todos os campos.

6. Nome inválido

- **Ação:** Cadastrar produto com nome **-2**.
- **Resultado Esperado:** Sistema **não deve permitir**, mas atualmente **permite**.
- **Observação:** Defeito identificado.

7. Preço negativo

- **Ação:** Cadastrar produto com preço **-10**.
- **Resultado Esperado:** Sistema deve recusar e exibir mensagem “Preço deve ser positivo”.

8. Quantidade igual a 0

- **Ação:** Cadastrar produto com quantidade **0**.
- **Resultado Esperado:** Sistema deve recusar e exibir mensagem “Quantidade deve ser maior que 0”.

9. Cadastro válido

- **Ação:** Cadastrar produto com nome válido, preço positivo, descrição válida e quantidade maior que 0.
- **Resultado Esperado:** Sistema deve permitir o cadastro com sucesso.

4. Riscos Identificados

- Inconsistência na validação do campo **Nome do Produto** (permite valores inválidos como **-2**).
- Mensagem genérica no **Login com senha incorreta** pode dificultar a experiência do usuário, embora seja prática de segurança.

5. Critérios de Sucesso

- Todas as regras de obrigatoriedade, formato e consistência devem estar alinhadas com as boas práticas.
- Nenhum cadastro de produto inválido deve ser permitido.
- Sistema deve apresentar mensagens claras e consistentes.

Bloco 2 - Eduardo:

Estilo de teste: **Teste de Responsividade;**

Identificação:

- Funcionalidade: Layout e usabilidade em diferentes dispositivos
- Campos testados: Página de login, página principal, formulário de cadastros, listagem de produtos e página principal de produtos.
- Técnica utilizada: Teste exploratório de responsividade em múltiplos tamanhos de tela (de 360x640 até 1920x1080).

Objetivo do Teste:

Verificar a adaptação da interface em diferentes resoluções, garantindo acessibilidade, legibilidade e usabilidade em dispositivos móveis, tablets e desktops.

Cenários Executados

Tela Testada / Página	Resultado Esperado	Resultado Obtido	Status
Tela de Login e Cadastro	Caixa do menu de login deve se ajustar ao centro da tela.	Caixa não diminui, laterais ultrapassam a borda da tela.	Possível melhoria
Página 'Bem-vindo (Adm)'	Layout ajustado, textos legíveis, botões alinhados, sem sobreposição.	A baixo de 991px de largura o menu superior some e não é possível abrir na lateral. Palavras começando a se quebrar. Otimizado para smartphone, mas menu continua com problema	Possível melhoria
Lista de Usuários e Lista de Produtos	Layout ajustado, textos legíveis, botões	a partir de 700px de largura borda roxa aparece na lateral direita e	Possível melhoria

	alinhados, sem sobreposição.	aumenta quanto menor a largura. Palavras quebram. Problema com menu persiste.	
Produtos	Layout ajustado, imagens e textos alinhados, botões proporcionais.	A partir de 1099px de largura caixas de produtos perdem formatação (texto e imagem desalinhados). Botões de adicionar ficam desalinhados. Problema do menu permanece. Não otimizado para smartphone	Urgente

Observações

- Em telas pequenas (smartphones), alguns formulários precisam de ajuste para não sobrepor outros elementos.
- Em telas grandes, o sistema poderia aproveitar melhor o espaço disponível (ex.: grid mais distribuído para produtos).
- O menu deve ser corrigido com urgência, pois o mesmo problema se repete em várias páginas.

Conclusão

- O sistema é parcialmente responsivo, com melhor desempenho em tablets e desktops.
- Pontos a corrigir: ajustes nos formulários em dispositivos móveis e melhor aproveitamento de tela em resoluções maiores.
- Recomendação: aplicar melhorias de CSS responsivo (media queries) e revisar proporções em diferentes breakpoints.

- Corrigir problemas críticos que impedem a navegação adequada na plataforma (ex.: menu em smartphones).

Bloco 3- Ana Livia:

Estilo de teste: **Goldilocks**

Identificação

- Funcionalidade: Cadastro de Produtos
- Campo testado: Descrição do Produto
- Técnica utilizada: Teste exploratório com heurística Goldilocks

Objetivo do teste:

Verificar o comportamento do sistema ao inserir diferentes valores no campo descrição, validando restrições e limites.

Cenários executados:

Cenário	Entrada	Resultado Esperado	Resultado Esperado	Status
Muito pouco	Descrição pequena	Sistema deve impedir cadastro	Cadastro aceito sem restrição	Possível Falha
No Ponto	Texto curto e ideal	Cadastro deve ser aceito.	Cadastro aceito	Sucesso
Muito	Texto com ~1000 palavras	Sistema deveria limitar tamanho (esperado).	Cadastro aceito sem restrição	Possível Falha

Observações

- O sistema não possui validação de limite de caracteres no campo descrição.
- Isso pode permitir entradas excessivamente grandes ou pequenas, resultando em problemas de desempenho, armazenamento desnecessário ou impacto na experiência do usuário.

Conclusão

- O campo está validado para não aceitar valores vazios, o que é positivo.
- Porém, não há limite de caracteres configurado, o que representa um risco funcional e técnico.
- Recomendação: definir um limite de caracteres adequado para garantir consistência e performance.

Bloco 4 - Caio:

Estilo de teste: **CRUD**

Identificação

Funcionalidade: Cadastro de Produtos, Buscas Produtos, Atualizar Produto e Deletar Produtos

Técnica utilizada: Teste exploratório com heurística CRUD

Objetivo do teste:

Verificar o comportamento do sistema ao Criar, Buscar, Atualizar e Deletar

Teste de CRUD:

Métodos	Objetivos
Criar	Criar um produto no severest
Buscar	Buscar esse produto
Atualizar	Atualizar o produto
Deletar	Deletar o produto

Observação

Com base na descrição fornecida, o teste exploratório com a heurística CRUD (Criar, Buscar, Atualizar, Deletar) foi aplicado para avaliar a funcionalidade de gerenciamento de produtos em um sistema. Abaixo estão as observações gerais sobre o processo e os pontos a serem considerados.

Criar (Create): Comportamento esperado: O sistema deve permitir o cadastro de um novo produto com informações válidas (como nome, descrição, preço, etc.) e retornar uma confirmação de sucesso (ex.: mensagem de sucesso ou código HTTP 201).

Buscar (Read): Comportamento esperado: O sistema deve localizar o produto criado com base em critérios de busca (ex.: ID, nome, ou outros filtros) e exibir suas informações corretamente.

Atualizar (Update): Comportamento esperado: O sistema deve permitir a edição de

informações de um produto existente (ex.: alterar preço ou descrição) e confirmar a atualização com sucesso.

Deletar (Delete): Comportamento esperado: O sistema deve permitir a exclusão de um produto específico e confirmar a operação com sucesso.

Conclusão

O teste exploratório com a heurística CRUD é uma abordagem prática para validar as funcionalidades principais de um sistema de cadastro de produtos. As operações de Criar, Buscar, Atualizar e Deletar foram testadas com o objetivo de verificar o comportamento do sistema em cenários típicos de uso.

Bloco 5 - Amilly:

Estilo de teste: **Bookmark**

Funcionalidade Avaliada

Proteção de rotas internas (ex.: cadastro de produto, listagem de produtos)

Técnica Utilizada: Teste exploratório com foco em navegação direta por URL

Objetivo do Teste: Verificar se o sistema bloqueia ou redireciona acessos diretos a páginas protegidas, simulando o uso de bookmarks fora do fluxo normal de navegação.

Métodos e Objetivos

- Sem login:

Ao acessar diretamente a URL /cadastroprodutos em aba anônima, o sistema redirecionou corretamente para a tela de login.

- Com usuário comum:

Após login com um usuário sem permissão de administrador, o sistema não bloqueou o acesso à tela de cadastro de produto.

- Com administrador:

Ao acessar diretamente a mesma URL após login como administrador, o sistema permitiu o acesso normalmente.

Conclusão:

O teste exploratório com foco em bookmark revelou falhas importantes no controle de acesso do sistema. Ao acessar diretamente a rota `/cadastrarprodutos`, utilizando um usuário comum (sem permissão de administrador), o sistema permitiu o acesso à tela protegida, sem apresentar bloqueio ou redirecionamento.

Esse comportamento indica que o front-end do ServeRest **não está validando corretamente o tipo de usuário** ao lidar com acessos diretos via URL. A ausência de restrição por perfil

representa um risco de segurança e quebra o fluxo esperado da aplicação, sendo um ponto crítico a ser corrigido.

Conclusão Geral da Squad

- O sistema apresenta **validações corretas em alguns pontos**, mas ainda possui **inconsistências de obrigatoriedade**.
- Há risco funcional no campo **Descrição** (falta limite de caracteres).
- O fluxo **CRUD** não funciona adequadamente.
- O sistema é **parcialmente responsivo**, com falhas em telas pequenas e grandes, especialmente no **menu**.

Recomendações Gerais:

- Melhorar consistência das mensagens de erro.
- Definir limites claros para campos de entrada.
- Garantir integridade em todas as operações CRUD.
- Corrigir problemas críticos de responsividade.
- O sistema de autenticação e níveis de permissão, prevenindo acesso indevidos e garantindo controle adequado das rotas protegidas.