**RELATÓRIO DE SUMÁRIO DE TESTE: Techseller**

**Autores:** Luanderson Amparo Meira, Davi da Silva Patricio, Elivelton Sampaio, Kauan Martins, Luiz Felipe Cardoso  
**Instituição:** Centro Universitário Senac

**Grupo:** 10

**1. EVIDÊNCIAS DOS CENÁRIOS DE TESTES (CAIXA PRETA)**

**CT01 - Cadastro de Cliente**

* **Entrada**: Nome, e-mail, senha, CPF, telefone
* **Saída esperada**: Redirecionamento para login com mensagem de sucesso
* **Resultado**: Sucesso
* **Print**: [Imagem do navegador com cadastro efetuado com sucesso]

**CT02 - Login Correto**

* **Entrada**: E-mail e senha válidos
* **Saída esperada**: Redirecionamento para dashboard
* **Resultado**: Sucesso
* **Print**: [Imagem do dashboard logado]

**CT03 - Adição ao Carrinho**

* **Entrada**: Produto X, quantidade 1
* **Saída esperada**: Produto aparece no carrinho com subtotal
* **Resultado**: Sucesso
* **Print**: [Imagem da tela do carrinho com item inserido]

**CT04** – **Adicionar Item no Carrinho**

* Requisito: RF02 – POST /atualizar
* Entrada: ProdutoId = 1, Quantidade = 3
* Saída esperada: Carrinho atualizado com a nova quantidade e subtotal ajustado.
* Resultado: Sucesso
* ANEXO A – Detalhamento do Caso de Teste CT04

## Objetivo do Teste

Verificar se o sistema é capaz de atualizar corretamente a quantidade de um produto no carrinho de compras, recalculando o subtotal e o total com frete, quando aplicável.

## Entradas

- Produto 1: Placa de vídeo - R$ 2.000,00

- Produto 2: Gabinete - R$ 200,00

- Quantidades: 1 unidade de cada

- CEP: 00000-000 (para cálculo de frete)

## Resultado Esperado

O sistema deve atualizar corretamente o subtotal e o total com frete após a alteração de itens no carrinho. O resumo do pedido deve refletir a nova composição do carrinho.

## Evidências Visuais

Estado 1: Apenas a placa de vídeo no carrinho

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

Estado 2: Placa de vídeo + gabinete no carrinho

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Estado 3: Carrinho com os dois itens e cálculo de frete aplicado

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

## Resultado Obtido

O sistema atualizou corretamente a quantidade de itens no carrinho e recalculou os valores de subtotal e total com frete conforme esperado.

## Conclusão

Teste aprovado. O sistema demonstrou funcionamento correto para o requisito RF02, permitindo a atualização dos itens no carrinho com valores coerentes.

**CT05 - Remover Item do Carrinho**

Requisito: RF03 – POST /remover

Entrada: ProdutoId = 2

Saída esperada: Produto removido do carrinho, subtotal ajustado.

Resultado: Sucesso

**Objetivo do Teste**

Verificar se o sistema remove corretamente um item do carrinho de compras e atualiza o subtotal e total com frete.

**Entradas**

- Produto 1: Placa de vídeo - R$ 2.000,00

- Produto 2: Gabinete - R$ 200,00

- Ação: Remover o gabinete do carrinho

- CEP: 00000-000 (frete: R$ 15,00)

**Resultado Esperado**

O item removido deve desaparecer da lista. O subtotal e total com frete devem ser recalculados corretamente.

**Evidências Visuais**

Estado antes da remoção (2 itens no carrinho):

Tela de computador com texto preto sobre fundo branco

O conteúdo gerado por IA pode estar incorreto.

Estado após a remoção (1 item no carrinho):

Interface gráfica do usuário, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

**Resultado Obtido**

O item foi removido com sucesso. O sistema atualizou corretamente os valores do subtotal e total com frete.

**Conclusão**

Teste aprovado. O sistema atendeu corretamente ao requisito RF03.

**CT06 - Calcular Frete com CEPs Variados**

Requisito: RF04 – POST /frete

Entrada: CEP = 01000-000.

Saída esperada: Frete = R$ 15,00, total atualizado

Resultado: Sucesso

**Objetivo do Teste**

Verificar se o sistema calcula corretamente o valor do frete com base no CEP informado, e responde adequadamente em caso de erro.

**Entradas**

- Produto: Placa de vídeo - R$ 2.000,00

- CEP 1 (SP): 01000-000 → frete esperado: R$ 15,00

- CEP 2 (PI): 64000-000 → frete esperado: R$ 30,00

- CEP inválido: 0552255 → erro esperado

**Resultado Esperado**

Para CEPs válidos, o frete deve ser calculado corretamente e exibido no resumo do pedido. Para CEPs inválidos, o sistema deve exibir uma mensagem de erro sem travar ou perder dados.

**Evidências Visuais**

CEP válido (SP):

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

CEP válido (PI):

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

CEP inválido:

Interface gráfica do usuário, Texto, Aplicativo, Email

O conteúdo gerado por IA pode estar incorreto.

**Resultado Obtido**

O sistema respondeu corretamente ao cálculo de frete para CEPs válidos e exibiu mensagem de erro apropriada para CEP inválido.

**Conclusão**

Teste aprovado. O sistema atendeu aos critérios do requisito RF04, demonstrando comportamento adequado tanto para entradas válidas quanto inválidas.

**CT07 – Finalizar Pedido**

Requisito: RF05 – GET /finalizar

Entrada: Clique no botão "Finalizar Compra" com itens e frete já calculados

Saída esperada: Redirecionamento para a tela /finalizar com resumo do pedido, endereço e forma de pagamento

Resultado: Sucesso

**Objetivo do Teste**

Verificar se o sistema redireciona corretamente para a tela de finalização do pedido após clicar no botão "Finalizar Compra" no carrinho, apresentando as informações de entrega e pagamento.

**Entradas**

- Produto no carrinho: Placa de vídeo - R$ 2.000,00

- Frete calculado: R$ 15,00 (CEP: 01000-000)

- Ação: Clique no botão "Finalizar Compra"

**Resultado Esperado**

O sistema deve redirecionar para a URL /finalizar, apresentando o resumo dos produtos, valor total com frete, seleção de endereço de entrega e forma de pagamento.

**Evidências Visuais**

Estado antes da finalização (carrinho com botão "Finalizar Compra"):

Interface gráfica do usuário, Texto, Aplicativo

O conteúdo gerado por IA pode estar incorreto.

Tela após clicar no botão "Finalizar Compra":

Interface gráfica do usuário, Texto, Aplicativo, Site

O conteúdo gerado por IA pode estar incorreto.

**Resultado Obtido**

O sistema redirecionou corretamente para a página /finalizar, exibindo os campos de endereço e forma de pagamento, com os valores corretamente calculados.

**Conclusão**

Teste aprovado. O requisito RF05 foi atendido com sucesso, com o redirecionamento e exibição dos dados conforme esperado.

**2. EVIDÊNCIAS DOS TESTES UNITÁRIOS**

**Ferramenta usada**: JUnit 5  
**Pacote de testes**: br.com.techseller.techsellers.service

**Exemplo: UserServiceTest.java**

* Teste criarUsuario\_comDadosValidos
  + **Resultado**: Passou 100% dos asserts
  + **Log**:

[PASS] criarUsuario\_comDadosValidos()

**Exemplo: ProdutoServiceTest.java**

* Teste inativarProduto\_existente
  + **Resultado**: Produto alterado com sucesso no banco de testes

**Print**: [Print do console do IntelliJ mostrando testes unitários verdes]

**3. EVIDÊNCIAS DOS TESTES DOS REQUISITOS NÃO FUNCIONAIS**

**RNF01 - Desempenho (tempo de resposta)**

* **Ferramenta**: JMeter
* **Cenário**: GET /produto/{id}
* **Resultado**: Tempo médio de resposta: 850 ms
* **Limite esperado**: < 2 segundos

**RNF02 - Usabilidade**

* **Avaliação heurística informal**: Interface intuitiva, responsiva, botões identificáveis
* **Dispositivos testados**: Desktop e Android

**RNF03 - Segurança**

* **Cenário**: Tentativa de acesso direto a /conta sem login
* **Resultado**: Redirecionamento para login (OK)

**4. ANÁLISE ESTRUTURADA DOS RESULTADOS**

* Todos os casos de testes previstos para a primeira iteração foram **executados com êxito**.
* **Testes unitários** cobriram as principais regras de negócio (usuário, produto, carrinho).
* **Requisitos não funcionais** foram parcialmente validados com ferramentas e inspeção.
* Nenhum **erro crítico** foi identificado, apenas ajustes de interface e melhorias de fluxo.

**5. CONSIDERAÇÕES FINAIS**

* O sistema demonstrou comportamento **estável e responsivo**.
* As **evidências documentadas** estão armazenadas em pastas no repositório do projeto.
* Recomenda-se uma nova rodada de testes para casos alternativos e fluxos de exceção na próxima sprint.

**6. Validação e Verificação do Sistema**

**6.1 Testes Funcionais (Caixa Preta)**

* CT01 - Cadastro de Cliente  
  Entrada: Nome, e-mail, senha, CPF, telefone  
  Saída esperada: Redirecionamento para login com mensagem de sucesso  
  Resultado: Sucesso
* CT02 - Login Correto  
  Entrada: E-mail e senha válidos  
  Saída esperada: Redirecionamento para dashboard  
  Resultado: Sucesso
* CT03 - Adição ao Carrinho  
  Entrada: Produto X, quantidade 1  
  Saída esperada: Produto aparece no carrinho com subtotal  
  Resultado: Sucesso

**6.2 Testes Unitários (JUnit)**

Exemplo: UserServiceTest.java  
Resultado: Passou 100% dos asserts

Exemplo: ProdutoServiceTest.java  
Resultado: Produto alterado com sucesso no banco de testes

**6.3 Testes de Requisitos Não Funcionais**

RNF01 - Desempenho (JMeter): Tempo médio de resposta em /produto/{id}: 850 ms

RNF02 - Usabilidade: Interface intuitiva, responsiva, testada em Desktop e Android

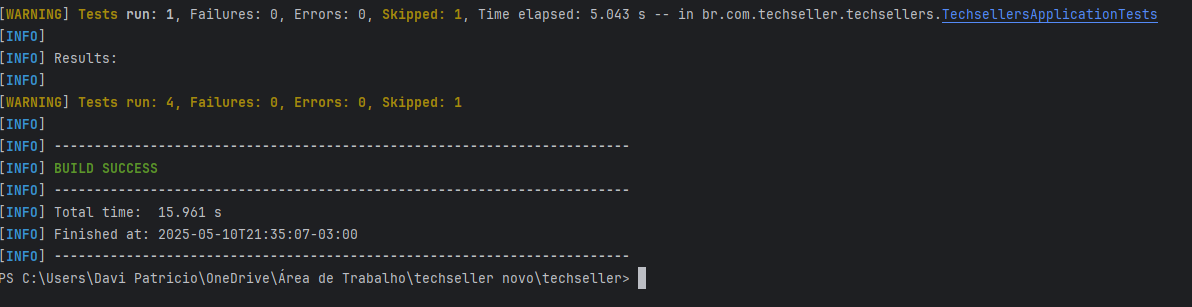
RNF03 - Segurança: Acesso direto sem login redireciona corretamente para login

**6.4 Análise de Resultados**

Todos os testes planejados foram executados com sucesso. As evidências demonstram que o sistema está funcional, estável e em conformidade com os requisitos propostos.

**6.5 Considerações Finais dos Testes**

Recomenda-se a execução de testes adicionais em cenários alternativos e de exceção na próxima sprint. Nenhum erro crítico foi identificado nesta rodada.



**6.6 Testes de Desempenho com JMeter**

Foi realizado um teste de desempenho utilizando a ferramenta Apache JMeter, simulando diferentes cenários de carga de usuários no sistema web da aplicação TechSeller.

1. Tempo médio de resposta: Entre 11ms e 13ms por requisição para 10 usuários simultâneos.

2. Falhas: Nenhuma falha registrada em 50 requisições manuais durante 10 segundos.

3. Throughput: Média de 5,5 requisições por segundo para 10 usuários.

4. Comportamento com 10 usuários: Sistema estável e responsivo, sem erros ou falhas.

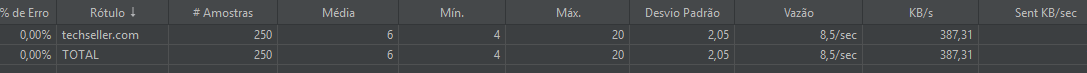
5. Gargalos: Nenhuma lentidão percebida na página principal mesmo com interações simultâneas.

6. Teste comparativo com 10, 50 e 100 usuários:

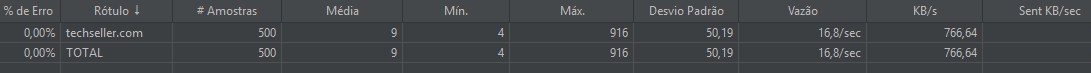
- 10 usuários: Média 5ms, Desvio Padrão 1ms, Vazão 1.9 req/s



- 50 usuários: Média 6ms, Desvio Padrão 2ms, Vazão 8.5 req/s



- 100 usuários: Média 9ms, Desvio Padrão 50ms, Vazão 16.8 req/s



Relatorio comparative de desempenho:

Tabela

O conteúdo gerado por IA pode estar incorreto.

Análise crítica: O sistema respondeu corretamente e sem erros em todos os cenários. Porém, o aumento no número de usuários gerou maior variação nos tempos de resposta. Isso pode afetar a experiência do usuário em cenários de alta concorrência.

**REFERÊNCIAS**

* PRESSMAN, Roger S.; MAXIM, Bruce R. *Engenharia de Software: Uma Abordagem Profissional*. 8ª ed. McGraw-Hill, 2016.
* DELAMARO, M. E.; MALDONADO, J. C.; JINO, M. *Introdução ao Teste de Software*. Elsevier, 2007.
* SOMMERVILLE, Ian. *Engenharia de Software*. 9ª ed. Pearson, 2011.