



# Plano de Teste

## Bootcamp Atlântico Avanti

**Professor:** Gutenberg  
**Monitor:** Niêdja Kaliene

**Equipe 7 - Caçadores do bug perdido:**  
Pablo Felipe  
José Ben Hur  
Anezia Beatriz

Outubro/2025

**Histórico de Revisões**

<b>Data</b>	<b>Versão</b>	<b>Descrição</b>	<b>Autor</b>
01/10/2025	1.0	Release Inicial	Pablo Felipe

## Índice

<b>1. Introdução.....</b>	<b>4</b>
1.1 Objetivos.....	4
1.2 Descrição da Aplicação .....	4
<b>2. Escopo de Teste .....</b>	<b>5</b>
<b>3. Tipos e Níveis de Teste .....</b>	<b>5</b>
3.1 Testes Funcionais .....	5
3.2 Teste de Usabilidade .....	5
3.3 Componentes .....	5
3.4 Integração .....	5
3.5 Testes de Compatibilidade: .....	6
3.6 Baseados na experiência .....	6
3.7 Acessibilidade.....	6
3.8 Exploratório .....	6
<b>4. Técnicas de Teste .....</b>	<b>6</b>
4.1 Caixa preta.....	6
<b>5. Ambiente de Teste.....</b>	<b>7</b>
<b>6. Papéis e Responsabilidades .....</b>	<b>7</b>
<b>7. Ferramentas .....</b>	<b>8</b>
<b>8. Métricas .....</b>	<b>8</b>
<b>9. Padrões .....</b>	<b>9</b>
9.1 Padrão de abertura e report de Bugs.....	9
9.2 Padrão de Especificação e Execução de Teste .....	10
<b>10. Escopo de Automação de Teste.....</b>	<b>10</b>
<b>11. Cronograma .....</b>	<b>12</b>

# 1. Introdução

## 1.1 Objetivos

O objetivo deste plano de testes é garantir a qualidade e o desempenho adequados da aplicação web "NopCommerce demo" (<https://demo.nopcommerce.com>), garantindo que as principais funcionalidades atendem aos requisitos funcionais e de usabilidade, oferecendo uma experiência eficiente e confiável para os usuários.

## 1.2 Descrição da Aplicação

O nopCommerce Demo é um site de demonstração de uma plataforma de e-commerce de código aberto. Ele oferece um ambiente prático para simular a jornada de um cliente em uma loja virtual, sendo ideal para o treinamento e desenvolvimento de testes funcionais e automatizados. É um recurso valioso tanto para iniciantes quanto para engenheiros de controle de qualidade avançados que desejam praticar em um ambiente de comércio eletrônico completo.

Esse Site é composto pela seguinte estrutura e funcionalidades:

1. **Home:** Página inicial que exibe produtos em destaque e as principais categorias da loja.
2. **Products (Produtos):** Apresenta uma vasta gama de produtos de e-commerce, organizados em categorias como Computadores, Eletrônicos, Vestuário, Livros, entre outros. Cada produto possui uma página de detalhes com imagens, preço, especificações e um botão para adicionar ao carrinho.
3. **Cart (Carrinho de Compras):** Permite que os usuários adicionem itens, ajustem quantidades e revisem os produtos selecionados antes de finalizar a compra.
4. **Signup / Login (Registo / Login):** Permite o registro de novas contas de usuário e a autenticação de usuários existentes.
5. **My Account (Minha conta): Painel** do usuário para gerenciar informações pessoais, endereços, visualizar histórico de pedidos e alterar o nome o sobrenome e senha.
6. **Checkout:** Processo de finalização de compra.

## 2. Escopo de Teste

Neste projeto, serão realizados testes funcionais, de usabilidade e de compatibilidade nas principais funcionalidades da aplicação, incluindo a Home, Register/Login, a navegação nos Produtos, o fluxo do Carrinho de Compras e o processo de Checkout. Será utilizada a técnica de teste de **caixa preta**, na qual não há acesso ao código-fonte, testando apenas a camada de apresentação (front-end) da aplicação.

## 3. Tipos e Níveis de Teste

A validação do sistema será feita por meio de testes funcionais na interface do usuário (UI), que verificarão o comportamento de cada página e a aderência do sistema como um todo aos requisitos.

### 3.1 Testes Funcionais

Os testes funcionais cobrirão todas as funcionalidades essenciais do sistema, validando interações entre componentes, interfaces e processos internos, garantindo que a aplicação funcione corretamente e de forma integrada.

### 3.2 Teste de Usabilidade

Serão realizados testes para avaliar a facilidade de uso e a experiência de navegação do usuário, analisando aspectos como responsividade, clareza de informações e usabilidade em diferentes dispositivos e resoluções.

### 3.3 Componentes

Este tipo de teste verifica módulos ou partes isoladas da aplicação, no caso elementos individuais que desempenham funções específicas.

**Observação:** os testes de componentes não serão realizados no projeto.

### 3.4 Integração

Os testes de integração têm como objetivo validar a comunicação entre os módulos e camadas do sistema, garantindo que os dados fluam corretamente entre banco de dados, backend e frontend. Também verificam se as informações armazenadas, processadas e exibidas estão consistentes em todo o sistema.

Observação: os testes de integração não serão realizados no projeto.

### **3.5 Testes de Compatibilidade**

Os testes de compatibilidade verificarão se a aplicação mantém o comportamento em diferentes navegadores (Google Chrome, Edge e FireFox) (Windows).  
Observação: devido ao tempo e escopo do projeto, este teste não será realizado.

### **3.6 Baseados na experiência**

Os testes baseados na experiência utilizam o conhecimento prévio e a intuição dos testadores para identificar possíveis falhas que não seriam detectadas com testes planejados.

### **3.7 Acessibilidade**

Esses testes garantem que o sistema seja utilizável por pessoas com diferentes tipos de deficiência, incluindo aquelas com limitações visuais, auditivas ou cognitivas.  
Observação: este tipo de teste não será realizado.

### **3.8 Exploratório**

Utilizam abordagem dinâmica onde o testador projeta casos de teste e os executa de forma simultânea, baseando-se no aprendizado e na intuição adquiridos durante a própria exploração do software.

## **4. Técnicas de Teste**

### **4.1 Caixa preta**

Esta técnica tem como objetivo definir um conjunto de condições de entrada capazes de validar todos os requisitos funcionais do software. Entre suas principais vantagens estão: reduzir a necessidade de testes extensivos, servir como guia na escolha de dados de entrada com alta probabilidade de identificar defeitos, dispensar o acesso ao código-fonte e manter a separação entre as visões do usuário e do desenvolvedor.

## 5. Ambiente de Teste

Os testes foram realizados em ambientes com as seguintes especificações:

Ambiente de Testes	
Browser utilizado:	Google Chrome
Versão do Browser:	140.0.7339.210 64 bits
Tipo do sistema operacional:	Windows
Versão do sistema operacional:	24H2
Ambiente:	Produção
Hardware:	Notebook LeNovoLOQ Core i5

## 6. Papéis e Responsabilidades

A equipe é formada por dois Analistas de QA e um Team Leader e um Engineer QA que tem as seguintes responsabilidades:

Responsável	Função	Responsabilidades
Pablo Felipe	Analista de QA	Planejar os Testes
José Ben Hur	Team Líder de QA	Projetar os Testes
Anezia Beatriz	Engineer QA	Implementar os Testes
Pablo Felipe	Analista de QA	Executar os Testes
José Ben Hur	Analista de QA	Avaliar os Testes

## 7. Ferramentas

Iremos utilizar as seguintes ferramentas auxiliares no desempenho de nossas funções:

Ferramentas Utilizadas	Finalidade
Captura de tela (ScrenPrint)	Utilizada para registrar evidências visuais durante a execução dos testes, facilitando a comprovação dos resultados e a análise de falhas.
Excel	Empregado para gerar gráficos, tabelas e métricas dos testes, permitindo uma visualização clara dos dados e auxiliando na análise dos resultados.
Word	Utilizado para documentar os casos de teste, detalhando cenários, passos, resultados esperados e obtidos, garantindo rastreabilidade e organização do processo.

## 8. Métricas

As métricas adotadas neste plano têm caráter **quantitativo** e têm como objetivo avaliar diferentes aspectos do processo de teste e da qualidade do software. Elas fornecem informações objetivas sobre **desempenho, eficácia e eficiência** das atividades executadas durante a validação do sistema.

**Estatísticas de Execução de Teste:** Número total de testes executados, passados e falhados.

**Taxa de Sucesso/Fracasso de Teste:** Proporção de testes bem-sucedidos em comparação com o total de testes executados.

Resultado dos testes	Observação
Total de cenários listados	Informar o número total de cenários criados.
Quantidade de testes bem-sucedidos	Indicar a quantidade e o percentual de testes que passaram.
Quantidade de testes com falha	Informar a quantidade e o percentual de testes que falharam.



## 9. Padrões

### 9.1 Padrão de abertura e report de Bugs

Os bugs serão categorizados de acordo com a gravidade.

**Crítico:** Impede o uso normal do site.

**Médio:** Impacta a funcionalidade, mas o usuário pode contornar o problema.

**Baixo:** Problemas de usabilidade ou estilísticos que não afetam a funcionalidade.

Abertura de Bugs	
<b>ID</b>	BugX
<b>Cenário</b>	Cenário em que o bug foi encontrado
<b>Título</b>	Nome do bug que será especificado.
<b>Responsável</b>	Nome do QA responsável pela descoberta do bug
<b>Data de Abertura</b>	Data de abertura do bug.
<b>Prioridade</b>	Prioridade com que o bug deve ser tratado
<b>Ambiente</b>	Forneça o ambiente em que ocorreu o Bug - Sistema operacional, navegador, versão e hardware
<b>Descrição</b>	Resultado que o sistema exibiu após os testes.
<b>Passo a passo</b>	Passo a passo realizado pelo testador para encontrar o bug.
<b>Dados de entrada</b>	Dados que foram fornecidos ao sistema.
<b>Resultado Esperado</b>	O que era esperado que o sistema retornasse após a ação.
<b>Resultado Observado</b>	Resultado que o sistema exibiu após os testes.
<b>Evidências</b>	Anexar capturas de tela ou arquivos para auxiliar na compreensão
<b>Status</b>	Aberto - Fechado - Reincidente

## 9.2 Padrão de Especificação e Execução de Teste

Seguiremos o seguinte padrão para cada caso de teste:

Especificação e Execução de Teste	
CT ID	Um número ou código único para identificar o teste
Descrição do CT	Uma breve descrição do que o teste está avaliando.
Prioridade	Alta - Média - Baixa
Autor	Responsável pela execução do teste
Ambiente	Informar em qual ambiente serão realizados os testes, como sistema operacional, navegadores e computadores
Pré Condições	O que é necessário para executar os testes
Dados de Entrada	Dados necessários para executar os testes
Passo a Passo	Passo a passo para reproduzir o teste
Resultado Esperado	O que esperamos que o sistema faça
Resultado Atual	O resultado real apresentado
Resultado Observado	Anotações de percepções que auxiliam no entendimento do resultado obtido ou na correção necessária
Status	Sucesso ou Falha

## 10. Escopo de Automação de Teste

Funcionalidade: Login	
ID	Cenários
001	Login com credenciais corretas
002	Login com credenciais incorretas

003	Logout do usuário
-----	-------------------

### Funcionalidade: Cadastro de usuário

ID	Cenários
001	Registro de novo usuário
002	Registro de usuário com e-mail já existente
003	Registro de usuário com e-mail inexistente
004	Registro de usuário com dados vazios
005	Registro de usuário com e-mail inválido
006	Registro de usuário com First Name inválido
007	Registro de usuário com Last Name inválido

### Funcionalidade: Produto

ID	Cenários
001	Visualização de Produtos
002	Detalhes do Produto
003	Comentários ao Produto
004	Pesquisa de Produtos
005	Adicionar ao carrinho
006	Adicionar produtos ao carrinho a partir de detalhes do produto
007	Adicionar quantidade de produto na aba detalhes do produto
008	Checar o produto no carrinho após o login

### Funcionalidade: Carrinho

ID	Cenários
001	Verificar a quantidade do produto no carrinho

002	Editar produto no carrinho
003	Remover produtos do carrinho
004	Verificar o carrinho após o login
005	Fazer check-out

## 11. Cronograma

Etapas	Data de Início	Data de Término
Planejar Testes	01/10/2025	03/10/2025
Projetar Testes	03/10/2025	05/10/2025
Implementar Testes	05/10/2025	07/10/2025
Executar Testes	07/10/2025	09/10/2025
Avaliar Testes	09/10/2025	16/10/2025