



[PLANO DE TESTES]

[Bootcamp Escola Atlântico Avanti]

SISTEMA: SWAG LABS

EQUIPE: Bugzilla

COLABORADORES:

Emily Carla

Fernanda Tiyomi

Leandro Freire

Isadora Carvalho

João Gilberto

Histórico de Revisões

Data	Versão	Descrição	Autor
18/09/2025	0.1	Estruturação Inicial	Isadora Carvalho
22/09/2025	1.0	Finalização e aprovação do Plano de Testes	Equipe Bugzilla

Índice

1. Introdução.....	4
1.1. Objetivo.....	4
1.2. Descrição da aplicação.....	4
2. Escopo Geral.....	6
3. Tipos e Níveis de Teste.....	6
3.1. Testes Funcionais.....	6
3.2. Teste de Usabilidade.....	7
3.3. Testes de Compatibilidade.....	7
4. Técnicas de Teste.....	7
4.1. Testes de Caixa Preta.....	7
5. Ambiente de Teste.....	7
6. Papéis e Responsabilidades.....	7
7. Ferramentas.....	8
8. Métricas.....	8
9. Padrões.....	10
9.1. Padrão de Especificação e Execução de Teste.....	10
9.2. Padrão de Abertura e Report de Bugs.....	10
10. Escopo de Teste.....	11
11. Cronograma.....	13

1. Introdução

O presente Plano de Testes tem como objetivo garantir a qualidade e o bom funcionamento do sistema Swag Labs (<https://www.saucedemo.com/v1/>), avaliando suas funcionalidades principais: autenticação de usuários, visualização de produtos e fluxo completo de compras. A abordagem de testes adotada considera critérios funcionais, usabilidade e experiência do usuário, assegurando que a aplicação esteja de acordo com padrões de qualidade e confiabilidade esperados.

1.1. Objetivo

O objetivo deste plano é realizar uma avaliação minuciosa das funcionalidades do Swag Labs, que incluem login, exibição e navegação de produtos, manipulação de itens no carrinho de compras, checkout e finalização de compras. A principal meta é verificar se todas as funcionalidades estão operando corretamente e de maneira consistente, sem erros críticos ou falhas, além de garantir que a interface seja amigável e intuitiva para o usuário.

1.2. Descrição da aplicação

O Swag Labs é um site de e-commerce fictício criado como ambiente seguro para testes automatizados e treinamento de profissionais de Quality Assurance (QA). A aplicação simula uma loja online com login, catálogo de produtos, carrinho e processo de finalização de compra, mas sem qualquer transação real envolvida.

Seu uso é amplamente recomendado para profissionais e estudantes de qualidade de software praticarem testes manuais e automatizados em um sistema completo, porém controlado.

Esse sistema é composto pelas seguintes estruturas e funcionalidades:

1. Página inicial (Home):

- Exibe uma lista de produtos com imagens, descrições, preços e a opção de adicionar ao carrinho.

- Produtos são organizados de forma simples, facilitando a navegação e a interação.
- Funcionalidade de filtragem e ordenação dos produtos por preço e popularidade.

2. Página de Login:

- Permite ao usuário realizar o login com credenciais pré-definidas (usuários padrões como "standard_user", "locked_out_user", "problem_user", "performance_glitch_user").
- Realiza uma verificação de credenciais e exibe mensagens de erro para dados inválidos.

3. Carrinho de Compras:

- Exibe os itens selecionados para compra, permitindo visualizar o total da compra.
- O usuário pode adicionar ou remover itens, além de ajustar a quantidade dos produtos.

4. Página de Checkout:

- O usuário deve fornecer informações de envio, como nome, sobrenome e código postal/endereço.
- Após o preenchimento dos dados, é possível revisar o pedido antes de finalizar a compra.
- Exibe uma simulação de pagamento com um botão para concluir a compra.

5. Processo de Finalização de Compras:

- Após a confirmação das informações de pagamento, o usuário é redirecionado para uma tela de resumo da compra, com uma mensagem de agradecimento.

6. Página de Erro (para usuários bloqueados):

- Caso um usuário com credenciais inválidas ou bloqueadas tente acessar o sistema, a aplicação exibe uma mensagem de erro específica.

7. Menu de Navegação:

- Um menu lateral que oferece acesso a páginas como "Home", "Carrinho", "Checkout" e "Logout", permitindo fácil navegação pela plataforma.

8. Funções de Responsividade:

- A aplicação é otimizada para diferentes dispositivos e tamanhos de tela, com um design responsivo para garantir uma boa experiência de usuário tanto em desktops quanto em dispositivos móveis.

2. Escopo Geral

No presente projeto, serão conduzidos testes funcionais, de usabilidade e de compatibilidade, com foco nas principais funcionalidades do sistema. Para a execução desses testes, será adotada a abordagem de teste do tipo Caixa Preta, na qual a análise é realizada sem o acesso ao código-fonte da aplicação, limitando-se à interface do usuário (camada front-end). Essa técnica permite verificar o comportamento do sistema com base nas entradas fornecidas e nas saídas observadas, garantindo que o software atenda aos requisitos especificados sob a perspectiva do usuário final.

3. Tipos e Níveis de Teste

Os testes a serem aplicados nesse projeto abrangem os seguintes tipos e níveis, cada um com objetivos específicos.

3.1. Testes Funcionais

Visam validar se as funcionalidades do sistema funcionam de acordo com os requisitos definidos anteriormente.

3.2. Teste de Usabilidade

Possuem como objetivo avaliar a facilidade de uso e experiência do usuário.

3.3. Testes de Compatibilidade

Objetivam verificar se o sistema funciona de forma correta em diferentes dispositivos, sistemas operacionais e navegadores.

4. Técnicas de Teste

4.1. Testes de Caixa Preta

Essa técnica consiste em uma abordagem de teste baseada nos requisitos e comportamentos esperados do sistema, sem levar em consideração sua estrutura interna ou implementação do código. O testador, atuando como um usuário final, fornece entradas ao sistema e observa as saídas geradas, a fim de verificar se estão em conformidade com o comportamento especificado.

5. Ambiente de Teste

Os testes foram realizados em ambientes com as seguintes características e especificações.

Ambiente de Teste		
Browser	Google Chrome	
Versão do Browser	141.0.7390.55	
Tipo de SO	Windows	
Versão do SO	Windows 11	Windows 10
Ambiente	Produção	

6. Papéis e Responsabilidades

A equipe é formada por um Team Leader, três analistas QA e um Engenheiro de qualidade que possuem as seguintes responsabilidades:

Responsável	Função	Responsabilidade
Isadora Carvalho	Líder QA	Planejar os testes e liderar a equipe
Emily Carla	Analista QA	Projetar os testes
Fernanda Tiayomi	Engenheiro QA	Implementar os testes
Leandro Freire	Analista QA automatizado	Executar os testes
João Gilberto	Analista QA	Avaliar os testes

7. Ferramentas

Serão utilizadas as seguintes ferramentas auxiliares no desempenho de nossas funções.

Ferramentas Utilizadas	
Organização	Trello
Documentação	Google Docs, Google Sheets
Evidências	Screencast

8. Métricas

As métricas adotadas pela equipe tem como objetivo avaliar a efetividade do processo de testes, mensurar a qualidade do sistema testado e fornecer indicadores quantitativos para a tomada de decisão.

Foram selecionadas métricas simples e objetivas, adequadas ao escopo e ao nível de maturidade do projeto, permitindo uma análise clara da execução e dos resultados obtidos.

As métricas definidas para este trabalho são quantitativas, pois utilizam dados numéricos coletados durante a execução dos testes para gerar indicadores mensuráveis de desempenho e qualidade.

A seguir, estão descritos os tipos de métricas que serão utilizados:

- **Métrica de Produtividade de Teste**

(Quantidade de cenários estimados × Quantidade de cenários executados)

Essa métrica avalia o nível de cumprimento do planejamento de testes. Ela permite verificar se todos os casos definidos no plano foram efetivamente executados dentro do cronograma, identificando eventuais atrasos ou desvios no processo.

Um alto percentual de execução indica boa gestão de tempo e eficiência da equipe.

- **Métrica de Qualidade do Sistema**

(Quantidade de cenários manuais × Quantidade de bugs encontrados)

Essa métrica mede a proporção de falhas detectadas em relação ao total de testes realizados. Seu objetivo é mensurar o nível de estabilidade e confiabilidade do sistema.

Uma taxa de falhas baixa demonstra que o sistema apresenta comportamento estável; já uma taxa alta pode indicar necessidade de correções estruturais ou revisões de requisitos.

- **Métrica de Severidade dos Defeitos**

(Quantidade de bugs × Criticidade dos bugs)

Essa métrica analisa a gravidade e o impacto das falhas encontradas durante a execução dos testes. Ao classificar os defeitos como Críticos, Altos, Médios ou Baixos, é possível identificar quais problemas demandam prioridade de correção e quais têm impacto reduzido no funcionamento geral.

Ela fornece uma visão clara sobre a saúde do produto e a urgência das ações corretivas.

Resultados		Observações
Total de cenários estimados	%	Listar o percentual de cenários de testes totais
Total de cenários testados	%	Listar o percentual de testes realizados
Total de testes de sucesso	%	Listar o percentual de testes que obtiveram sucesso
Total de testes sem sucesso	%	Listar o percentual de testes que falharam

9. Padrões

9.1. Padrão de Especificação e Execução de Teste

Seguiremos o seguinte padrão para cada caso de teste

Especificação e Execução de Teste	
ID	Um código único para identificar o teste
Autor (a)	Responsável pela execução do teste
Descrição do CT	Uma breve descrição do que será testado
Ambiente	Informar em qual ambiente os testes serão realizados, como o SO, navegadores e computadores
Pré Condição	O que é necessário para realizar 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 Encontrado	O resultado real apresentado
Prioridade	Alta - Média - Baixa

9.2. Padrão de Abertura e Report de Bugs

Os bugs serão categorizados de acordo com sua gravidade, conforme descrito abaixo.

Abertura de Bugs	
ID	Um código único para identificar o Bug
Responsável	Responsável por descobrir o Bug
Data de abertura	Data em que o Bug foi encontrado
Cenário	Cenário em que o Bug foi encontrado
Descrição	Descrição do Bug
Passo a passo	Detalhamento do Bug, incluindo passos para reprodução
Ambiente	Informar em qual ambiente ocorreu o Bug, como o SO, navegadores e computadores
Prioridade	Baixa - Média - Alta
Severidade	Menor - Maior - Crítica
Evidências	Anexo de capturas de tela para ajudar na compreensão
Status	Aberto - Fechado

10. Escopo de Teste

ID	Cenários
CT001	Realizar login com credenciais válidas (usuário: standard_user)
CT002	Tentar login com um usuário bloqueado (usuário: locked_out_user)
CT003	Tentar login com um usuário que apresenta problemas de interface (usuário: problem_user)

CT004	Tentar login com um usuário que causa lentidão (usuário: performance_glitch_user)
CT005	Tentar login com usuário válido e senha inválida
CT006	Tentar login com usuário inválido e senha válida
CT007	Tentar login com os campos 'usuário' e 'senha' em branco
CT008	Realizar logout com sucesso a partir do menu
CT009	Tentar acessar páginas restritas usando o botão "voltar" do navegador após o logout
CT0010	Verificar se a página de produtos carrega e exibe corretamente todos os itens, com imagem, nome, descrição e preço
CT0011	Navegar para a página de detalhes de um produto e verificar se as informações são exibidas corretamente
CT0012	Adicionar e remover um item do carrinho a partir da página de listagem de produtos
CT0013	Adicionar e remover um item do carrinho a partir da página de detalhes do produto
CT0014	Ordenar os produtos por nome (A-Z e Z-A) e verificar se a ordenação está correta
CT0015	Ordenar os produtos por preço (menor para o maior e maior para o menor) e verificar se a ordenação está correta
CT0016	Remover um item do carrinho e verificar se ele foi removido
CT0017	Clicar em "Continuar Comprando" e verificar se o usuário é

	redirecionado para a página de produtos sem esvaziar o carrinho
CT0018	Prosseguir para o checkout e preencher todos os campos do formulário com dados válidos
CT0019	Tentar prosseguir no checkout com todos os campos em branco
CT0020	Tentar finalizar a compra utilizando somente dados inválidos nos campos obrigatórios.
CT0021	Tentar prosseguir no checkout deixando cada um dos campos obrigatórios ("First Name", "Last Name", "Postal Code") em branco, um de cada vez
CT0022	Tentar prosseguir no checkout com o carrinho vazio
CT0023	Verificar se a página de resumo (overview) exibe corretamente as informações da compra (produtos, valores, total)
CT0024	Finalizar a compra e verificar se o usuário é redirecionado para a página de agradecimento
CT0025	Cancelar a compra na página de resumo e verificar se o usuário retorna à página de produtos

11. Cronograma

Etapas	Início	Fim	Responsável
Reunião inicial de alinhamento	17/09/2025	17/09/2025	Equipe Bugzilla
Elaboração do Plano de Testes	18/09/2025	22/09/2025	Isadora Carvalho
Definição dos casos de teste	23/09/2025	28/09/2025	Emily Carla
Revisão dos casos de teste e ajustes no Plano	29/09/2025	30/09/2025	Equipe Bugzilla

Execução e registro dos testes	30/09/2025	06/10/2025	Equipe Bugzilla
Consolidação do Relatório de Testes	07/10/2025	10/10/2025	Isadora Carvalho
Registro de falhas no Relatório de Bugs	11/10/2025	13/10/2025	Equipe Bugzilla
Revisão de documentos	14/10/2025	14/10/2025	Isadora Carvalho
Ajustes Finais e Entrega	15/10/2025	18/10/2025	Isadora Carvalho