

# **Plano de testes: E-commerce**

## **1. Introdução**

Este documento apresenta o planejamento das atividades de testes do sistema Exemplo o qual será utilizado como base para as atividades de acompanhamento, revisão, verificação e validação do projeto, desde seu início até sua conclusão, a fim de garantir a análise comparativa do resultado real versus planejado. Desta forma, ações corretivas e preventivas poderão ser tomadas, sempre que resultados reais desviarem significativamente do planejado.

O escopo de desenvolvimento deste projeto consiste em um e-commerce que possibilita a venda de produtos, mercadorias ou até mesmo serviços por qualquer tipo de usuário que possua um cadastro no software.

## **2. Requisitos a serem testados**

A lista a seguir identifica os itens (casos de uso, requisitos funcionais, requisitos não funcionais) que foram identificados como alvos do teste. Essa lista representa o que deverá ser testado.

### **2.1 Teste de Integridade dos Dados e do Banco de Dados**

- Verificar acesso ao Banco de Dados do Catálogo de Produtos.
- Verificar acesso ao Banco de Dados através da ação de Login na plataforma.

### **2.2 Teste de Interface com o Usuário**

- Verificar a facilidade de navegação utilizando um conjunto de amostras de telas.

### **2.3 Teste de Desempenho**

- Verificar o tempo de resposta para acesso ao Catálogo de Produtos.
- Verificar o tempo de resposta para login.

### **2.4 Teste de Segurança e Controle de Acesso**

- Verificar o login a partir de um PC local.
- Verificar a segurança de login por meio de mecanismos de nome de usuário ou e-mail e senha.

### 3. Estratégia do Teste

A Estratégia de Teste apresenta a abordagem recomendada para o teste dos aplicativos de software. As principais considerações para a estratégia de teste são as técnicas a serem utilizadas e o critério para saber quando o teste está concluído.

#### 3.1 Teste de Integridade dos Dados e do Banco de Dados

Os bancos de dados e os processos de banco de dados devem ser testados como sistemas separados. Esses sistemas devem ser testados sem os aplicativos (como a interface para os dados). É necessário executar pesquisas adicionais referentes ao DBMS a fim de identificar as ferramentas / técnicas que poderão existir para suportar os testes identificados a seguir.

**Objetivo do teste:** Assegurar que os processos e métodos de acesso ao Banco de Dados funcionem corretamente e sem corrupção de dados.

**Técnicas:**

- Chamar cada processo e método de acesso a banco de dados, propagando cada um com dados válidos e inválidos (ou pedidos de dados).
- Inspecionar o banco de dados para assegurar que os dados foram preenchidos conforme planejado e que todos os eventos do banco de dados ocorreram adequadamente ou revisar os dados retornados para assegurar que os dados corretos foram recuperados (pelas razões corretas)

**Crítérios de conclusão:** Todos os processos e métodos de acesso ao banco de dados funcionam conforme projetado e sem nenhuma corrupção de dados.

**Considerações especiais:**

- Os processos devem ser chamados manualmente.
- Bancos de dados pequenos ou de tamanho mínimo (número limitado de registros) devem ser utilizados para aumentar a visibilidade de quaisquer eventos não aceitáveis.

#### 3.2 Teste de Interface com o Usuário

O teste da Interface com o Usuário verifica a interação de um usuário com o software. A meta do Teste de UI é assegurar que a Interface com o Usuário forneça ao usuário o acesso e a navegação adequados por meio das funções dos aplicativos.

**Objetivo do teste:** Verificar se os objetos e características da janela, tais como menus, tamanho, posição, estado e foco estão em conformidade com os padrões.

**Técnicas:** Criar / modificar testes para cada janela a fim de verificar a navegação adequada e os estados de objeto para cada janela e objeto do aplicativo.

**Crítérios de conclusão:** verificação com êxito de cada janela permanecer consistente com a versão de benchmark ou dentro do padrão aceitável

**Considerações especiais:** nem todas as propriedades de objetos personalizados e de terceiros podem ser acessadas.

### 3.3 Teste de Desempenho

O teste de desempenho mede tempos de resposta, taxas de transação e outros requisitos sensíveis ao tempo. A meta do teste de desempenho é verificar e validar se os requisitos de desempenho foram alcançados. O teste de desempenho normalmente é executado várias vezes, utilizando cargas diferentes no sistema.

**Objetivo do teste:** validar o Tempo de Resposta do Sistema para funções de negócios ou transações designadas sob as duas condições a seguir:

- Volume normal previsto;
- Volume de pior caso previsto.

**Técnicas:**

- Modificar arquivos de dados (a fim de aumentar o número de transações) ou modificar scripts a fim de aumentar o número de iterações ocorrido em cada transação.
- Os scripts devem ser executados em uma máquina (o melhor é avaliar o desempenho de um único usuário, uma única transação) e repetidos com vários clientes (virtuais ou reais, consulte as considerações especiais a seguir).

**CrITÉrios de conclusão:**

- **Transação única / usuário único:** conclusão com êxito dos scripts de teste sem nenhum defeito e na alocação de tempo esperada / requerida (por transação)
- **Várias transações / vários usuários:** conclusão com êxito dos scripts de teste sem nenhum defeito e dentro de alocação de tempo aceitável.

**Considerações especiais:** O teste de desempenho deverá ser executado em uma máquina dedicada ou em um período de tempo dedicado. Isso permitirá o controle total e a medição exata.

### 3.4 Teste de Segurança e Controle de Acesso

O Teste de Segurança e de Controle de Acesso tem como foco duas áreas principais de segurança:

- Segurança do aplicativo, incluindo o acesso aos Dados ou às Funções de Negócios.
- Segurança do sistema, incluindo acesso remoto ao sistema.

A segurança do aplicativo assegura que, com base na segurança desejada, os usuários têm restrição a funções específicas ou estão limitados aos dados que estão disponíveis a eles. Por exemplo, todos têm permissão para inserir dados e criar novas contas, mas apenas os gerentes poderão excluí-los. Se houver segurança no nível dos dados, o teste assegura que o usuário "tipo" um pode consultar todas as informações do cliente, incluindo dados financeiros; no entanto, o usuário dois consulta apenas os dados demográficos para o mesmo cliente.

A segurança do sistema assegura que apenas os usuários, para os quais o acesso ao sistema foi concedido, sejam capazes de acessar as informações pessoais de cada conta.

**Objetivo do teste:**

- **Segurança de Função / Dados:** verificar se o usuário pode acessar apenas as funções / dados para os quais seu tipo de usuário tenha recebido permissão.
- **Segurança do Sistema:** verificar se apenas os usuários com acesso ao sistema e aplicativo(s) têm permissão para acessá-los.

**Técnicas:**

- Identificar e listar cada tipo de usuário e as funções ou dados para os quais cada tipo tem permissão.
- Criar testes para cada tipo de usuário e verificar a permissão criando transações específicas para cada tipo de usuário.
- Modificar o tipo de usuário e executar novamente os testes para os mesmos usuários. Em cada caso, verificar se as funções ou dados adicionais estão corretamente disponíveis ou se têm seu acesso negado.
- Acesso ao Sistema.

**Critérios de conclusão:** para cada tipo de usuário conhecido, a função e/ou dados apropriados estão disponíveis e todas as transações funcionam como esperado.

**Considerações especiais:** o acesso às funções do sistema deve ser revisado / discutido com o administrador do sistema. Talvez esse teste não seja necessário, pois pode ser uma função de administração da rede ou do sistema.

**4. Cronograma de Atividades**

A seguir será listado a ordem de execução das atividades a serem desenvolvidas neste projeto:

1. Planejamento e definição do projeto a ser desenvolvido;
2. Planejamento e definição do escopo de testes a ser realizado;
3. Desenvolvimento do software;
4. Aplicação dos testes planejados;
5. Aplicação dos ajustes identificados nos testes;
6. Encerramento do projeto.