

Quality First: Unleashing the Power of Software Testing

Francisco Renan Leite da Costa
Departamento
Organização
Pau dos Ferros, Brasil
francisco.costa57942@alunos.ufersa.edu.br

Jonathan Ryan da Silva Paiva
Departamento
Organização
Pau dos Ferros, Brasil
Jonathasryann@gmail.com

Vinicius Anacleto de Almeida
Departamento
Organização
Pau dos Ferros, Brasil
0009-0001-6251-407X

Resumo—Os testes de software desempenham um papel importante no processo de desenvolvimento de software. Eles ajudam a garantir que o software seja de alta qualidade e funcione conforme o esperado. O presente artigo tem como finalidade mostrar a importância do teste de software, que está diretamente ligado à qualidade de software. Testar aplicações possui um papel importante na qualidade do produto final. O teste é uma tarefa que toda empresa de desenvolvimento precisa

Index Terms—Teste de Software, Qualidade, Produto, Desenvolvimento.

I. INTRODUÇÃO

Com os avanços tecnológicos o mundo ficou cada vez mais influenciado pela tecnologia e a eficiência da informática, a demanda por software de qualidade só cresce a cada ano. Hoje, o conceito de qualidade tem importância fundamental para alavancar a competitividade das empresas. Conceituar qualidade de fato é uma tarefa complexa, mas ela pode ser vista como um método gerencial que através de procedimentos disseminados por toda a organização, busca garantir um produto final que satisfaça às expectativas do cliente, dentro daquilo que foi acordado inicialmente. Portanto, o presente projeto é baseado através do desenvolvimento de um sistema de comércio eletrônico permitindo que empresas vendam seus produtos ou serviços on-line. Os sistemas de e-commerce são ferramentas poderosas que podem ajudar empresas a crescer e prosperar. De acordo com as análises, o projeto terá o modelo conforme está definido na figura 1.

II. DESENVOLVIMENTO

A definição de requisitos é uma etapa fundamental no processo de desenvolvimento de sistemas de software. Ela assegura que o sistema seja projetado para atender às necessidades dos usuários e alcançar os objetivos de negócio estabelecidos.

III. REQUISITOS DE USUÁRIO

Cadastro do usuário: O sistema deve permitir que os usuários se cadastrem com seus dados pessoais, incluindo

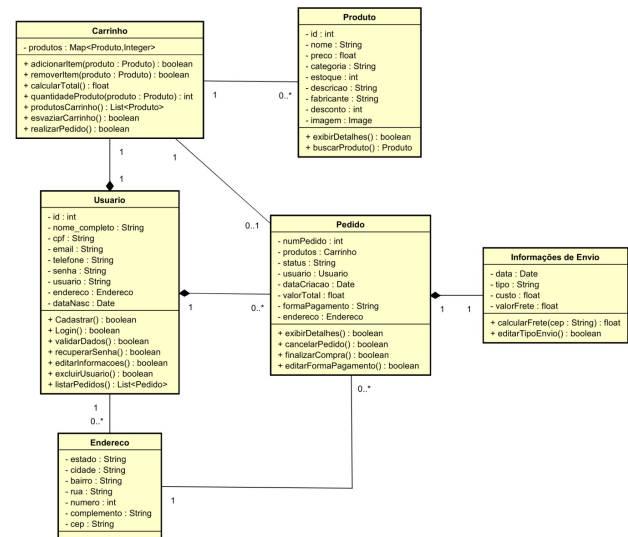


Figura 1. Diagrama de UML com os métodos e atributos do projeto.

nome, CPF, endereço, e-mail, número de telefone (opcional) e data de nascimento, senha e nome de usuário.

Fazer login: O usuário pode acessar o sistema a partir dos seus dados previamente cadastrados no sistema.

Validar dados: Os campos de dados devem ser validados para garantir que as informações sejam corretas e completas.

Editar informações do usuário: O usuário pode editar as suas informações cadastradas no sistema.

Armazenamento de dados: Os dados do usuário devem ser armazenados em um banco de dados seguro.

Endereço do usuário: O usuário pode ter mais de um endereço, como também adicionar e excluir endereços.

Excluir usuário: O usuário pode excluir conta do sistema.

IV. REQUISITOS DE CATÁLOGO DE PRODUTO

Dados do produto: O catálogo de produtos deve conter informações verdadeiras e atualizadas sobre os produtos, incluindo nome, descrição, preço, imagens e disponibilidade.

Buscar produto: O usuário deve ser capaz de filtrar e buscar produtos por nome, categoria, preço e outros critérios. O sistema irá exibir o produto selecionado e indicar o seu nome, preço, categoria, descrição, fabricante, quantidade disponível no estoque e se possui algum desconto no momento. **Listar produto:** O usuário deve ser capaz de listar todos os produtos disponíveis.

V. REQUISITOS DE CARRINHO DE COMPRAS

Dados do carrinho: O carrinho de compras deve conter uma lista de todos os produtos que o usuário deseja comprar.

Visualizar carrinho: Para cada produto, o carrinho de compras deve exibir o preço unitário, quantidade disponível, nome e valor total das compras.

Atualizar carrinho: O usuário deve ser capaz de atualizar o carrinho de compras, adicionando, removendo ou alterando a quantidade de produtos.

Esvaziar carrinho: O usuário pode esvaziar completamente o carrinho. **Inicializar carrinho:** O usuário ele inicializa o carrinho. **Esvaziar carrinho:** O usuário pode esvaziar completamente o carrinho.

VI. REQUISITOS DE PEDIDO

Realizar pedido: O usuário pode realizar o seu pedido de todos os produtos do seu carrinho. Após a realização desse pedido vai ser gerado as informações do mesmo onde terá, o número do pedido, os produtos que serão comprados, qual usuário efetuou o pedido, a data da criação desse pedido, o seu valor total e o status “em andamento”. Além disso, o usuário deverá inserir qual forma de pagamento será efetuada.

Listar pedidos: O usuário pode listar todos os pedidos que foram feitos por ele. O sistema irá exibir uma lista com esses pedidos e suas respectivas informações.

Formas de pagamento: O sistema deve oferecer diferentes formas de pagamento.

Exibir detalhes do pedido: O usuário pode selecionar um pedido e o sistema irá exibir todos os seus detalhes, pagamento.

Cancelar pedido: O usuário pode cancelar um pedido que não foi finalizado.

Métodos de entrega: O sistema deve permitir que os usuários escolham entre diferentes métodos de entrega, com estimativas de tempo e custo. Isso deve incluir opções como entrega padrão, entrega expressa e retirada na loja. A capacidade de escolher entre diferentes métodos de entrega é importante para atender às necessidades dos usuários. O sistema deve oferecer uma variedade de opções para que os usuários possam escolher a que melhor se adapta às suas necessidades.

Finalizar compra: O usuário pode finalizar a sua compra do pedido que foi efetuado, o sistema irá exibir todas as informações do pedido para que o usuário consiga revisá-las e realizar a compra. Esses requisitos são essenciais para garantir

que o sistema seja seguro, acessível e conveniente para os usuários.

VII. TESTES A SEREM EXECUTADOS

O sistema será submetido ao ciclo de vida de teste completo, conforme o padrão de projeto de software de qualidade. Este ciclo inclui as seguintes fases:

Teste de unidade: Verificado o comportamento de unidades individuais como as funções, classes ou módulos, irão ser projetados para garantir que cada unidade atenda aos seus requisitos funcionais e não apresente erros.

Teste de integração: Verificar a interação entre unidades individuais, para garantir que as unidades possam se comunicar e cooperar entre si para atingir os objetivos do sistema.

Teste de validação: Verificar se o sistema atende aos requisitos do usuário, garantindo que o sistema funcione de acordo com as expectativas dos usuários.

Teste de aceitação: O sistema atende às necessidades da organização que o está desenvolvendo ou usando.

Teste de sistema: Analisar o comportamento global do sistema, ou seja, garantir que o sistema funcione conforme o esperado, levando em consideração todos os seus componentes e interações.

Em caso de novas versões do sistema, o teste de regressão. Ademais, os casos de teste estarão concentrados nas fases de unidade e integração.

VIII. TESTES CAIXA PRETA

Teste de classe de equivalência: Teste de classe de equivalência: esse teste irá abordar os testes de entradas válidas e inválidas, e suas respectivas saídas. Em auxílio desse teste poderá ser utilizadas técnicas de grafo de causa/efeito e/ou tabela de decisão, tendo como objetivo organizar e facilitar a visualização de como ocorre o fluxo desses testes.

IX. TESTES CAIXA BRANCA

Teste de fluxo de controle: Nesse teste iremos verificar todas as estruturas de controle de um módulo do código do sistema, como laços de repetição, condições, decisões e chamadas de função. Os testes serão executados com critérios o mais próximo possível do critério Todos-Caminhos, tendo como objetivo que a maior quantidade possível de comandos e caminhos sejam testados. Para esse teste será utilizado um grafo para esse fluxo de controle que facilitará o entendimento e a análise do teste.

Teste de fluxo de dados: O teste de fluxo de dados irá verificar o uso dos dados presentes no código, ele irá detalhar a definição, o uso e a destruição de cada variável, e com isso indicar se há algum erro ou uso incorreto da variável. Esse teste também será utilizado um grafo, o qual é bem semelhante ao grafo do fluxo de controle, tendo em vista que irá mostrar o fluxo de dados das variáveis por meio de um módulo.

Implementando os testes de fluxo de controle e de dados no sistema de e-commerce verificaremos a estrutura de controle e o uso dos dados em cada módulo do nosso sistema, ou seja, será efetuado testes nas classes do usuário, do produto, do carrinho e do pedido, e em seus respectivos métodos.

X. CONSIDERAÇÕES FINAIS

Em um mundo em constante evolução impulsionado pela tecnologia, não podemos subestimar a importância dos testes de software. Neste artigo, exploramos os conceitos fundamentais, as práticas recomendadas e as tendências emergentes no campo dos testes de software. Constatamos que, à medida que os sistemas de software se tornam cada vez mais complexos e os prazos se tornam mais apertados, a qualidade do software desempenha um papel crítico no impacto direto na satisfação do usuário e no sucesso dos projetos.

Os testes de software não são apenas uma fase em um ciclo de desenvolvimento, mas sim um processo contínuo que requer dedicação e melhoria contínua. Conforme novas tecnologias e abordagens de desenvolvimento surgem, os profissionais de teste de software devem ajustar-se e aprimorar suas habilidades para lidar com os desafios em constante evolução.

Dessa forma, conseguimos obter os seguintes resultados para os testes implementados.

Caso de teste 1: Cadastro do cliente no sistema. Este caso de teste se concentra no cenário em que um cliente deseja se cadastrar no sistema. As pré-condições não impõem restrições, ou seja, o cliente pode se cadastrar sem problemas. As regras estipulam que o CPF, o e-mail e o usuário do cliente não devem estar previamente registrados no sistema, o que garante que não haja duplicações de informações. O resultado esperado é que, se todas essas condições forem atendidas, o cliente seja cadastrado com sucesso.

Caso de Teste 2: Cadastro do Cliente no Sistema Neste outro caso de teste, não há requisitos específicos a serem cumpridos. No entanto, é necessário que o CPF, e-mail ou nome de usuário do cliente já estejam registrados no sistema. Isso significa que o sistema precisa ser capaz de identificar duplicatas e evitar o registro repetido. Conforme indicado pela ação, espera-se que o sistema exiba uma mensagem de erro para informar ao cliente que o cadastro não foi concluído com sucesso. Portanto, o resultado desejado é que o cliente não seja registrado no sistema.

Caso de Teste 3: Cadastro do Cliente no Sistema Neste caso de teste, as pré-condições são novamente abertas, não impondo nenhuma condição específica antes do teste. As regras indicam que o cliente deve digitar seus dados corretamente. Caso o cliente digite informações de forma incorreta, o sistema deve identificar o erro e emitir uma mensagem de erro. Isso serve para garantir a integridade dos dados no sistema e a precisão das informações. O resultado esperado é que o cliente não seja cadastrado no sistema se algum dado for inserido incorretamente.

Estes casos de teste são úteis para garantir que o processo de cadastro de clientes no sistema seja robusto e capaz de lidar

com diversas situações, incluindo a prevenção de duplicações e a validação de dados corretos.

Caso de Teste 4: Login do cliente no sistema Para que este caso de teste seja executado, é necessário que haja usuários devidamente cadastrados no sistema. Essa é uma etapa essencial para a verificação do processo de login. Como regras, os dados de usuário e senha inseridos devem ser corretos e correspondentes aos registros existentes no sistema. Após a inserção dos dados de login, o sistema deve permitir o acesso ao cliente. O resultado esperado para esse caso de teste é que o cliente seja capaz de efetuar o login com sucesso e tenha acesso ao sistema.

Caso de Teste 5: Login do cliente no sistema Assim como no caso anterior, é necessário que haja usuários cadastrados no sistema para a realização deste teste. Como regra, os dados de usuário e/ou senha inseridos estão incorretos, ou seja, não correspondem aos registros existentes. Quando os dados incorretos são inseridos, o sistema deve responder emitindo uma mensagem de erro. Neste caso, o cliente não deve obter acesso ao sistema devido à inserção de informações de login incorretas. O resultado esperado é que o sistema emita uma mensagem de erro.

Caso de Teste 6: Excluir Cliente O cliente deve ter realizado o login com sucesso, o que implica que ele já tem uma conta ativa no sistema, deve navegar até o menu de seu perfil, indicando a intenção de exclusão da conta. A regra é o cliente seleciona a opção de excluir usuário, manifestando a vontade de encerrar sua conta. O sistema deve efetuar a exclusão do usuário com sucesso, removendo-o do banco de dados do sistema. Após a exclusão, o usuário é redirecionado para a tela de login.

Neste caso de teste, o resultado esperado é que o cliente não possua mais uma conta ativa no sistema após a conclusão do processo de exclusão. Ele será redirecionado para a tela de login e, a partir desse momento, sua conta não estará mais disponível no sistema.

Esses casos de teste visam garantir a funcionalidade adequada do processo de login, bem como a capacidade de exclusão de contas de clientes de forma segura e controlada. O primeiro caso abrange um cenário em que o login é bem-sucedido, o segundo trata de situações em que os dados estão incorretos e o terceiro avalia a exclusão da conta de um cliente. Estas etapas são essenciais para a segurança e usabilidade do sistema.

Caso de teste 7: Listar pedidos do Cliente O cliente deve ter realizado o login com sucesso, o que indica que ele já possui uma conta no sistema. O cliente deve navegar até o menu de seu perfil, demonstrando a intenção de visualizar seus pedidos. O cliente deve ter pelo menos um ou mais pedidos registrados no banco de dados. O cliente deve selecionar a opção "Listar Pedidos" no menu de seu perfil. O sistema efetua uma busca no banco de dados, recuperando todos os pedidos relacionados a esse cliente. O resultado esperado é que o cliente seja capaz de visualizar todos os pedidos que foram realizados por ele. Esse caso de teste avalia a funcionalidade de exibir os pedidos de um cliente, o que é crucial para que

ele acompanhe suas compras e histórico de pedidos.

Caso de teste 8: Listar pedidos do Cliente O cliente deve ter efetuado o login com sucesso. O cliente deve navegar até o menu de seu perfil. O cliente não possui nenhum pedido registrado no banco de dados. O cliente deve selecionar a opção "Listar Pedidos" no menu de seu perfil. O sistema tenta buscar pedidos relacionados a esse cliente no banco de dados. Nesse cenário, o resultado esperado é que o cliente não visualize nenhum pedido, já que ele não possui pedidos registrados no sistema. O sistema deve, no entanto, emitir uma mensagem de erro apropriada para informar ao cliente que não há pedidos a serem exibidos.

Caso de teste 9: Adicionar Endereço do Cliente O cliente deve ter efetuado o login com sucesso, navegar até o menu de seu perfil. Regra o cliente deve selecionar a opção "Adicionar Endereço." E inserir todas as informações corretamente. Dessa forma, O sistema efetua o cadastro do endereço do cliente no banco de dados. O resultado esperado é que o endereço do cliente seja adicionado com sucesso. Isso implica que todas as informações do endereço foram registradas corretamente no sistema e estarão disponíveis para uso futuro, como para a entrega de pedidos.

Caso de teste 10: Adicionar Endereço do Cliente O cliente deve ter efetuado o login com sucesso, navegar até o menu de seu perfil. Regra o cliente deve selecionar a opção "Adicionar Endereço." E inserir todas as informações incorretas. É esperado o sistema emitir uma informação de erro. Nesse caso, o resultado esperado é que o sistema emita uma mensagem de erro e o endereço do cliente não seja adicionado com sucesso. Isso ocorre porque as informações do endereço inseridas foram incorretas, o que pode prejudicar a entrega de pedidos futuros ou causar problemas de localização.

Esses casos de teste servem para verificar a funcionalidade do sistema em relação ao gerenciamento de pedidos e endereços dos clientes, garantindo que os dados sejam armazenados e exibidos corretamente, e que o sistema seja capaz de lidar com cenários em que não há informações disponíveis ou em que ocorram erros de entrada de dados.

A manipulação dos dados fornecidos pelo usuário é uma parte crucial do funcionamento do nosso sistema e, portanto, passa por uma série de verificações rigorosas para garantir a qualidade e a validade das informações. A seguir, são apresentados os diferentes testes aos quais os dados do usuário são submetidos:

A. Testes do Nome:

Não Pode Ser Nulo: É fundamental que o campo de nome seja preenchido, e a presença de dados é obrigatória.

Tamanho Adequado: O nome deve possuir no mínimo 4 caracteres e não pode exceder 100 caracteres, assegurando um equilíbrio entre informações relevantes e evitando entradas excessivamente longas.

Ausência de Símbolos: Caracteres simbólicos não são permitidos no campo de nome.

Ausência de Números: Números não devem estar presentes no nome.

B. Testes do CPF:

Não Pode Ser Nulo: O CPF deve ser fornecido e não pode ser deixado em branco.

Validação por API: A validação do CPF é realizada por meio de uma API dedicada. Se essa validação não for bem-sucedida, o CPF será considerado inválido.

C. Testes do Email

Não Pode Ser Nulo: O campo de e-mail deve conter informações e não pode ser omitido.

Validação por API: A validação do e-mail é executada por meio de uma API especializada. Se a validação não for concluída com sucesso, o e-mail será considerado inválido.

D. Testes do Telefone

Validação por API: A validação do número de telefone é conduzida por uma API específica. Caso essa validação não seja aprovada, o telefone será considerado inválido.

E. Testes da senha:

Não Pode Ser Nula: A senha deve ser fornecida e não pode ser deixada em branco.

Tamanho Adequado: A senha deve conter no mínimo 8 caracteres e não pode exceder 40 caracteres.

F. Testes do Usuário:

Não Pode Ser Nulo: O campo de usuário não pode estar vazio e deve conter informações.

Tamanho Adequado: O nome de usuário deve possuir pelo menos 4 caracteres e não mais que 100 caracteres.

Ausência de Símbolos Caracteres simbólicos não são permitidos no nome de usuário.

G. Testes da Data de Nascimento:

Não Pode Ser Nula: A data de nascimento deve ser fornecida e não pode ser omitida.

Data no Passado: A data de nascimento não pode estar no futuro.

Restrição de Idade: A data de nascimento não pode ser superior a 110 anos atrás, e deve ser, no mínimo, há 18 anos, garantindo que o usuário tenha pelo menos 18 anos de idade.

Ademais, existe muitos outros casos de testes que foram testados e que estarão disponíveis no repositório. Dessa forma vocês podem acessar e conhecer todos os testes implementados. Portanto, podemos demonstrar o quanto estamos contentes com este trabalho, pois ele contribui para nosso entendimento de como o mundo dos testes funcionam e, esperançosamente, a dos leitores futuros, sobre uma área da tecnologia que pode não ser muito popular no mercado atual, mas que continua a apresentar um crescimento significativo em termos de demanda e importância.

XI. REFERÊNCIAS

DELAMARO, M. Introdução ao Teste de Software SPILLNER, A.; LINZ, T.; SCHAEFER, H. Software Testing Foundations.

GONÇALVES, P. F.; et al. Testes de Software e Gerência de Configuração

JORGENSEN, P. C. Software Testing: A Craftsman's Approach.