

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 realizar antes de entregar o produto, visto que o mesmo reduz o máximo de erros até chegar ao usuário final. Mas a grande questão é que a maior parte das empresas não realizam testes.

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.

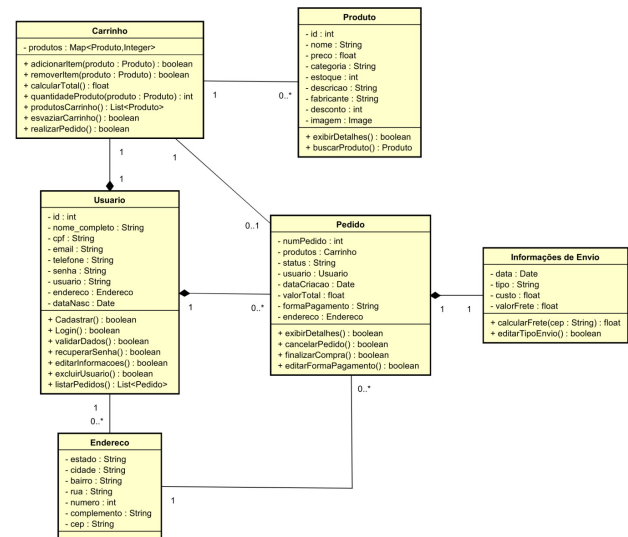


Figura 1. Diagrama de UML com os métodos e atributos do projeto.

III. REQUISITOS DE USUÁRIO

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

Recuperação de senha: O sistema deve permitir que os usuários recuperem senhas perdidas por meio de processos de redefinição seguros. A capacidade de recuperar senhas perdidas é essencial para garantir que os usuários não sejam bloqueados de suas contas. O sistema deve fornecer um processo de redefinição de senha seguro e fácil de usar.

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

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.

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.

Editar forma de pagamento: O usuário pode editar qual será a forma de pagamento do pedido que está sendo realizado no momento.

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

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.

Calcular frete: O sistema pode calcular o frete de entrega de um produto.

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 utilizado 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 A 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.