

Plano de Fluxo de Trabalho e Ciclo de Vida de um Bug

Introdução

Este documento descreve o plano de fluxo de trabalho e o ciclo de vida de um bug em um ambiente de desenvolvimento de software. Gerenciar bugs eficazmente é fundamental para manter a qualidade de um software, garantir a satisfação do cliente e melhorar a eficiência da equipe de desenvolvimento.

Plano de Fluxo de Trabalho

1. Identificação do Bug

O processo começa com a identificação de um bug. Isso pode ocorrer por meio de testes, relatórios de usuários, monitoramento de logs, ou outras fontes de feedback.

2. Registro do Bug

O bug é registrado em um sistema de rastreamento de problemas (como JIRA, GitHub Issues, ou Trello). O registro deve incluir informações detalhadas, como:

Descrição do bug.

Passos para reproduzir.

Capturas de tela ou anexos relevantes.

Prioridade e severidade.

Quem identificou o bug.

3. Atribuição do Bug

O bug é atribuído a um membro da equipe responsável por sua resolução. Isso pode ser um desenvolvedor, um tester ou outro membro da equipe, dependendo da complexidade e urgência do problema.

4. Resolução do Bug

O desenvolvedor designado começa a trabalhar na correção do bug. Isso envolve a identificação da causa raiz e a implementação de uma solução.

5. Teste da Correção

Após a correção, o bug é submetido a testes para garantir que a correção não introduza novos problemas e que o bug original tenha sido efetivamente resolvido.

6. Revisão e Validação

Um revisor, que pode ser o responsável pela atribuição do bug, revisa a correção e valida se ela resolve o problema de acordo com os critérios estabelecidos.

7. Testes Adicionais

Em alguns casos, testes adicionais podem ser necessários para garantir que a correção não afete outros aspectos do software.

8. Aprovação

Uma vez validada, a correção é aprovada e pronta para ser implantada.

9. Implantação

A correção é implantada no ambiente de produção ou na próxima versão do software.

10. Encerramento do Bug

O bug é fechado no sistema de rastreamento de problemas, e a documentação é atualizada com detalhes da correção.

Ciclo de Vida de um Bug

O ciclo de vida de um bug inclui os seguintes estados:

Aberto: O bug foi registrado e aguarda atribuição a um desenvolvedor.

Atribuído: O bug foi atribuído a um desenvolvedor para resolução.

Em Progresso: O desenvolvedor está trabalhando na correção do bug.

Aguardando Revisão: Após a correção, o bug aguarda revisão e validação.

Rejeitado: Se a correção não atender aos critérios, o bug é rejeitado e volta ao estado "Em Progresso" para mais trabalho.

Resolvido: O bug foi corrigido e validado.

Fechado: O bug foi implantado e aprovado. Não há mais trabalho a ser feito.

Conclusão

Um plano de fluxo de trabalho e um ciclo de vida de bug eficazes são essenciais para manter o software de alta qualidade, melhorar a comunicação da equipe e atender às expectativas dos usuários. Seguir um processo bem definido ajuda a garantir que os bugs sejam identificados, corrigidos e documentados de maneira consistente e eficiente.

Documentação de Teste

Introdução

Esta documentação de teste descreve dois casos de teste usando a técnica "Step-by-step". Os casos de teste visam verificar o funcionamento de um site de e-commerce (Caso de Teste 1) e o processo de login em um sistema (Caso de Teste 2).

Caso de Teste 1: Compra de Produto em um Site de E-commerce

Objetivo: Verificar se um usuário pode comprar um produto com sucesso em um site de e-commerce.

Passos do Teste

Acesse o site de e-commerce e faça login com suas credenciais.

Navegue até a página inicial.

Pesquise por um produto específico usando a barra de pesquisa.

Selecione o produto desejado na lista de resultados da pesquisa.

Adicione o produto ao carrinho.

Visualize o conteúdo do carrinho e verifique se o produto correto está listado.

Prossiga para o processo de checkout.

Forneça as informações de entrega, como endereço e método de envio.

Escolha um método de pagamento e forneça os detalhes do cartão.

Revise todas as informações e confirme a compra.

Verifique se a página de confirmação exibe uma mensagem de sucesso.

Verifique o e-mail de confirmação recebido após a compra.

Critérios de Aceitação

O usuário consegue concluir todas as etapas com sucesso.

A mensagem de confirmação é exibida.

O e-mail de confirmação é recebido e contém as informações corretas da compra.

Caso de Teste 2: Processo de Login em um Sistema

Objetivo: Validar se o processo de login em um sistema é eficaz.

Passos do Teste

Acesse a página de login do sistema.

Insira um nome de usuário válido.

Insira uma senha válida.

Clique no botão "Login".

Verifique se o sistema redireciona você para a página inicial.

Faça logout da conta.

Tente fazer login com um nome de usuário inválido.

Verifique se o sistema exibe uma mensagem de erro apropriada.

Tente fazer login com uma senha incorreta.

Verifique se o sistema exibe outra mensagem de erro apropriada.

Tente fazer login com campos em branco.

Verifique se o sistema não permite o login e exibe mensagens de erro adequadas.

Critérios de Aceitação

O usuário pode fazer login com sucesso usando credenciais válidas.

O sistema trata credenciais inválidas e em branco corretamente, exibindo mensagens de erro apropriadas.

O redirecionamento para a página inicial após o login é bem-sucedido.

Conclusão

Estes casos de teste "Step-by-step" fornecem uma estrutura detalhada e sequencial para validar o comportamento de um site de e-commerce e o processo de login em um sistema. Eles garantem que as funcionalidades sejam testadas de maneira abrangente e que os critérios de aceitação sejam atendidos. A execução desses testes é essencial para garantir a qualidade do software e a satisfação do usuário.

Teste BDD

Introdução

Esta documentação de teste BDD (Behavior-Driven Development) descreve dois casos de teste que seguem a abordagem BDD. Os casos de teste foram desenvolvidos para verificar o processo de reserva de voos em um sistema de reservas online (Caso de Teste 1) e a funcionalidade de adicionar tarefas a uma lista em um aplicativo de lista de tarefas (Caso de Teste 2).

Caso de Teste 1: Reserva de Voos

Objetivo: Verificar o processo de reserva de voos em um sistema de reservas online.

Cenário: Reserva de Voo Básica

Dado que um usuário acessa a página de reserva de voos

Quando o usuário seleciona um destino e uma data de viagem

E o usuário clica no botão "Pesquisar Voos"

Então o sistema deve exibir uma lista de voos disponíveis

E o usuário deve ser capaz de selecionar um voo e continuar com a reserva

Critérios de Aceitação:

O usuário pode acessar a página de reserva de voos.

O sistema exibe uma lista de voos disponíveis com base no destino e na data selecionados.

O usuário pode selecionar um voo e prosseguir com a reserva.

Caso de Teste 2: Aplicativo de Lista de Tarefas

Objetivo: Verificar a funcionalidade de adicionar tarefas a uma lista em um aplicativo de lista de tarefas.

Cenário: Adição de Tarefas à Lista

Dado que o usuário está na página inicial do aplicativo de lista de tarefas

Quando o usuário insere o nome de uma nova tarefa no campo de entrada

E o usuário pressiona a tecla "Enter" ou clica no botão "Adicionar"

Então a nova tarefa deve ser exibida na lista de tarefas

E o campo de entrada deve ser limpo para adicionar outra tarefa

Critérios de Aceitação:

O usuário pode adicionar uma nova tarefa à lista.

A nova tarefa é exibida na lista de tarefas.

O campo de entrada é limpo após a adição da tarefa.

Conclusão

Estes casos de teste BDD descrevem o comportamento esperado de um sistema de reservas online e de um aplicativo de lista de tarefas de uma forma clara e legível. A abordagem BDD ajuda a criar casos de teste que são compreensíveis para todas as partes interessadas, facilitando a colaboração entre desenvolvedores, testadores e partes interessadas não técnicas. A execução destes testes é fundamental para garantir que os sistemas atendam aos requisitos e às expectativas dos usuários.