**Passo-a-Passo de um Ambiente de Testes de Software**

1. **Cenário:**

- Outsourcing de teste de desenvolvimento para a Área da Aviação:

- Equipes geograficamente espalhadas: Ucrânia, Paquistão, Malta, Londres e outros países;

- Requer agilidade diária das equipes de desenvolvimento e teste;

1. **Processo de Desenvolvimento e Teste:**

- Engenheiros de QA recebem requisitos junto aos desenvolvedores;

- Desenvolvedores programam e criam testes unitários para cobrir o código;

- Engenheiros de QA preparam ciclos de teste para a funcionalidade;

- Recomendação de cobertura mínima: 50% para projetos com interface gráfica e 80% para projetos sem interface.

1. **Execução de Testes:**

- Testes Unitários são executados automaticamente durante a compilação;

- Após a programação, a equipe de teste realiza os testes funcionais;

- Estratégias de teste utilizadas:

- Testes de Integração;

- Testes de Sistemas;

- Testes de Desempenho;

- Testes de Aceitação.

1. **Testes de Integração e Regressão:**

- Testes de Integração garantem o funcionamento da nova funcionalidade com outros sistemas;

- Testes de Regressão verificam se as funcionalidades existentes não foram afetadas.

1. **Documentação e Aprovação:**

- Equipe de QA realiza a aprovação final (sign-off) da versão;

- Documentos são criados, detalhando as builds, testes e defeitos encontrados;

- Especificações técnicas e documentação para o usuário final são preparadas.

1. **Desafios na Área de Qualidade de Software:**

- Tempo limitado para todos os profissionais envolvidos;

- Selecionar e priorizar casos de teste críticos, considerando o conhecimento do negócio do cliente;

- Estimar casos de teste e utilizar ferramentas adequadas para auxiliar nessa tarefa.

1. **Desafios Clássicos de Teste:**

- Seleção, priorização, estimativa e design de casos de teste;

- Habilidades técnicas e conhecimento do negócio são necessários;

- Técnicas Funcionais são amplamente utilizadas, assim como Testes Unitários, de Sistemas, de Integração e de Desempenho;

**Detalhes sobre os Tipos de Testes:**

1. **Testes Unitários:**

- São testes realizados em pequenas unidades de código, geralmente, funções ou métodos individuais;

- O objetivo é verificar se cada unidade de código funciona corretamente e isoladamente;

- São executados pelos desenvolvedores, para garantir que o código esteja correto antes de ser integrado ao sistema;

- Geralmente, são automatizados e verificam a lógica interna do código.

1. **Testes Funcionais:**

- São testes realizados para verificar se o software atende aos requisitos funcionais especificados;

- Centram-se no comportamento esperado do sistema e nas suas funcionalidades.

- São executados pelos engenheiros de QA para validar se o software está funcionando corretamente, de acordo com os requisitos do cliente.

1. **Testes de Integração:**

- São testes realizados para verificar se as diferentes partes do sistema funcionam corretamente, quando integradas;

- O objetivo é detectar problemas de comunicação, interoperabilidade e integração entre os componentes do software.

- Podem ser executados após os testes unitários e antes dos testes de sistemas;

- Garantem que as diferentes partes do sistema trabalhem em conjunto sem erros.

1. **Testes de Sistemas:**

- São testes realizados para verificar se o sistema como um todo atende aos requisitos e funcionalidades especificadas;

- Testam a interação entre os componentes do sistema e como eles se comportam em diferentes cenários;

- Geralmente, são realizados após os testes de integração e antes dos testes de desempenho e aceitação.

1. **Testes de Desempenho:**

- São testes realizados para avaliar o desempenho e a capacidade de resposta do sistema em condições de carga e estresse;

- Verificam se o sistema é capaz de lidar com um grande número de usuários, transações ou volume de dados sem degradar seu desempenho;

- Podem incluir testes de carga, testes de estresse e testes de escalabilidade.

1. **Testes de Aceitação:**

- São testes realizados para verificar se o software atende aos critérios de aceitação definidos pelo cliente;

- Geralmente são realizados em colaboração com o usuário final ou representantes do cliente;

- O objetivo é garantir que o software atenda às expectativas e necessidades do cliente antes de ser implantado em produção.

1. **Testes de Regressão:**

- São testes realizados para verificar se as alterações ou correções realizadas no software não introduziram novos defeitos ou afetaram funcionalidades existentes;

- São executados após modificações no sistema, seja por meio de atualizações, correções de bugs ou novas implementações;

- Garantem que as funcionalidades previamente testadas e aprovadas continuem funcionando corretamente após as alterações realizadas.

1. **Relato de Defeitos encontrados nos testes:**

Ao encontrar defeitos durante os testes, é importante reportá-los de forma adequada para que possam ser corrigidos pelos desenvolvedores. Alguns pontos relevantes no relato de defeitos incluem:

- Identificação clara do defeito: Descrever de forma detalhada o problema encontrado, incluindo a localização no software, passos para reproduzir o defeito e qualquer mensagem de erro ou comportamento inesperado;

- Priorização e classificação: Classificar a severidade do defeito com base no impacto e na urgência, indicando se é um problema crítico, importante ou menor;

- Contexto e ambiente: Informar informações relevantes sobre o ambiente de teste, como sistema operacional, versão do software, configurações específicas, dados de entrada utilizados e qualquer outra informação que possa ajudar na reprodução do defeito;

- Anexar evidências: Incluir capturas de tela, registros de erros, arquivos de log ou qualquer outra evidência que possa ajudar os desenvolvedores a entender e resolver o defeito;

- Informar passos de reprodução: Fornecer instruções claras e detalhadas sobre como reproduzir o defeito, para que os desenvolvedores possam verificar e corrigir o problema;

- Acompanhamento e comunicação: Manter uma comunicação aberta e contínua com os desenvolvedores, informando sobre o status dos defeitos relatados, respondendo a possíveis perguntas e fornecendo feedback adicional, se necessário.

Ao seguir essas diretrizes, o relato de defeitos encontrados durante os testes contribuirá para a melhoria da qualidade do software, permitindo que os desenvolvedores identifiquem e corrijam os problemas de forma eficiente.

A importância do processo de software bem definido é para garantir o sucesso do produto e entregar valor ao cliente.