**Glossário de Testes de Software**

**Bug:** Um defeito, erro ou falha no software que causa comportamento inesperado ou não desejado.

**Caso de teste:** Uma especificação detalhada das condições, ações e resultados esperados para um teste específico.

**Critério de cobertura dos testes:** Permitem a identificação de partes do programa que devem ser executadas para garantir a qualidade do software e indicar quando o mesmo foi suficientemente testado.

**DDD:** (Domain-Driven Design) Trata-se de uma abordagem de design de software de forma disciplinada, abordando uma série de conceitos e técnicas sempre com o foco no domínio do software.

**Etapa Green do TDD:** Adicionamos uma nova funcionalidade ao sistema, fazendo o teste passar

**Etapa Red do TDD:** Escreva um teste que não passe

**Etapa Refactor do TDD:** Refatorar o código com a nova funcionalidade

**Fluxo de trabalho de teste:** Uma sequência de atividades e processos que compõem o ciclo de vida dos testes, desde o planejamento até a execução e relatórios.

**Inspeção simples :** detecção de defeitos na lógica e no código de maneira informal

Testes estáticos oferecem uma maneira de melhorar a produtividade e qualidade do desenvolvimento de software, através das ferramentas de análise, revisando e inspecionando suas variações.

**Qualidade de software:** Totalidade de características de um software que concebe a capacidade de satisfazer necessidades explícitas e implícitas.

**TDD:** (Test Driven Development) Desenvolvimento voltado para a realização de testes automatizados com base em ciclos curtos de duração e consistência na qualidade do código.

**Técnicas de testes**: Técnicas são procedimentos técnicos e gerenciais que ajudam a avaliação e a melhoria do processo. As técnicas mais utilizadas são: teste de caixa branca e teste de caixa preta.

**Testes automatizados:** Utilizam aplicativos específicos que são capazes de testar incansavelmente um software através de scripts de testes pré-configurados.

**Teste de aceitação:** busca verificar se o que foi implementado atende corretamente o que o cliente esperava; validar o ponto de vista do cliente.

**Teste de caixa branca:** Um tipo de teste em que o testador tem conhecimento interno da estrutura e implementação do sistema sendo testado.

**Teste de caixa cinza:** Um tipo de teste que combina elementos do teste de caixa preta e teste de caixa branca, em que o testador possui um conhecimento parcial da estrutura interna do sistema.

**Teste de caixa preta:** Um tipo de teste em que o testador não tem conhecimento interno da estrutura ou funcionamento do sistema sendo testado, focando no comportamento externo.

**Teste de carga:** O teste de carga verifica qual o limite de usuários simultâneos ou transações antes de ficar fora do ar, lento ou com problemas de carregamento dos dados.

**Teste de configuração**: verifica a operação do sistema em diferentes configurações de software e hardware.

**Teste de integração:** Visa provocar falhas associadas às interfaces entre os módulos quando esses são integrados para construir a estrutura do software que foi estabelecida na fase de projeto.

**Teste de interface**: Os testes de interfaces ocorrem quando módulos ou subsistemas são integrados para criar sistemas maiores.

**Teste de performance:** Medição e verificação da capacidade e desempenho do sistema de software em relação aos requisitos especificados.

**Teste de qualidade de software:** É uma atividade realizada durante o processo de desenvolvimento de software para avaliar a conformidade do produto final com os padrões de qualidade estabelecidos.

**Teste de release:** Uma equipe de testes independente testa a versão completa do sistema antes de entregá-lo aos usuários, também verifica se o software atende a todos os requisitos especificados.

**Teste de segurança:** Tipo de teste de software que identifica vulnerabilidades e avalia a segurança do sistema em relação a possíveis ameaças.

**Testes de sistema:** testa como é a interação entre os elementos do sistema como hardware, software e banco de dados. Não se vê apenas o software isoladamente, mas sim o sistema como um todo, como se o usuário final estivesse utilizando o sistema.

**Testes de unidade**: São códigos e interfaces testados separadamente do sistema, em blocos, ou seja, testado funções, classes, trechos de códigos.

**Teste de usabilidade:** usado para determinar o grau com o qual a interface é

fácil de usar para o usuário, detectando problemas na integração com o

mesmo.

**Teste de software:** o teste de software é um processo controlado, que verifica a qualidade e integridade do software, por meio da identificação de erros, cuja incidência é assumida, desde o início.

**Teste funcional:** é uma técnica utilizada para se projetarem casos de testes na qual o programa ou sistema é considerado uma caixa preta e, para testá-lo, são fornecidas entradas e avaliadas as saídas geradas para verificar se estão em conformidade com os objetivos especificados.

**Teste regressivo**: processo no qual envolve re-testar segmentos já analisados, diante de mudanças no software.

**Testes aleatórios** : Entradas dos casos são aleatórias dentro de uma margem. Eficácia considerada baixa. Conhecidos como testes de stress.

**Testes exaustivos**: Exploram as possibilidades de entrada e saída, e demandam muito tempo.

**Unidade:** menor trecho de código que pode ser testado.

**Verificações dinâmicas**: ocorrem em testes de software. O sistema é executado com dados de teste e seu comportamento operacional é observado.