

A detailed black and white line drawing of a circuit board, featuring various components like resistors, capacitors, and integrated circuits connected by a network of lines.

1

TEXTO BASE

QUALIDADE DE SOFTWARE

Texto base

1.1

Introdução ao Teste de Software

Marco Túlio Jeunon

Resumo

A busca por software, cada dia se torna mais presente no nosso cotidiano, em todas as nossas atividades e áreas, porém como satisfazer as necessidades de qualidade dos mesmos? Nesta aula o objetivo é introduzir o aluno no contexto de teste de software, fazendo com que entenda a necessidade da sua realização estruturada, bem como conhecer as principais atividades relacionadas a ele, assim será capaz de atingir os objetivos de qualidade requeridos.

1.1. O que é teste de Software?

Teste de *software* é parte do processo de desenvolvimento de *software*, que tem como objetivo encontrar falhas ou defeitos, antes que o produto de software seja implantado em produção ou seja tenta garantir a qualidade com base nos requisitos esperados pelo solicitante, tanto os funcionais quanto os não funcionais.

1.2. Por que é necessário testar?

Cada vez mais softwares estão sendo necessários a nossa vida cotidiana, tanto para empresas quanto para pessoas e em todas as áreas. O software tem uma natureza muito complexa, pois além de terem os aspectos funcionais, temos os não funcionais, que são as restrições estabelecidas para o seu desenvolvimento. O dissabor no encontro de falhas e defeitos em um software é uma experiência que em alguma medida já nos deparamos. A falha o software poderá gerar muitos problemas e gerar graves consequências. Portanto devemos nos preocupar na realização estruturada dos testes.

1.3. Principais consequências da falta de teste

A falta do teste ou a sua baixa cobertura na execução podem levar a graves consequências, dentre as quais podemos destacar:

- **Perdas Financeiras**, tanto pelo lado do contratante do software, quanto para o fornecedor do mesmo;
- **Perda de Produtividade**, pois um software com problemas, pode paralisar algumas atividades durante um longo período, ocasionando perda de horas e horas de trabalho ou acesso de clientes, que não se consegue recuperar posteriormente;
- **Perda de Qualidade**, ou seja para se ter qualidade devemos atender aos requisitos ou até mesmo superá-los, uma dos requisitos mais desejáveis do software é que ele esteja disponível no tempo certo e na hora certa para que cumpra a sua missão e seja utilizado.
- **Perda de Confiança**, uma das conquistas mais importantes de um fornecedor de software, é que ele passe confiança ao contratante. Essa relação ganha ganha faz com que vários negócios possam ser realizados, porém quando se perde a confiança se perde o cliente, que certamente custou para conseguir;
- **Lesões corporais ou morte**, dependendo do tipo de aplicação sendo desenvolvida, podemos ter risco de lesões e ou até morte, em virtude da falta de teste.
- **Desastres tecnológicos.** podem acontecer.

1.4. Principais causas de falhas no software

As principais causas de falhas no software, são muitas porém destacamos :

- Produto de software escrito e desenvolvido por pessoas;
- Pressão do tempo ou seja prazo requerido para desenvolvimento menor do que seria necessário para se garantir a qualidade do mesmo;
- Complexidade do software;
- Falha na comunicação, entre os envolvidos;
- Estabilidade dos requisitos, ou seja muita alteração nos requisitos funcionais e não funcionais, ocasionando muitas manutenções na aplicação;
- Processo de desenvolvimento imaturo, sem se ter maturidade e processo de desenvolvimento.

1.5. Devemos testar tudo?

Normalmente não se tem tempo e recursos suficientes para se realize uma cobertura de 100% dos testes necessários, em virtude disso devemos priorizar os testes levando primeiramente em conta os riscos. Em segundo lugar devemos testar as restrições do projeto.

1.6. Principais atividades de teste

Dentre as principais atividades de teste, destacamos :

- Planejamento e controle

Nessa atividade devemos elaborar o Plano de Testes, que de acordo com o padrão IEE 829 (Padrão para documentação de teste de software), deverá conter:

- Identificador do plano de testes;
- Introdução (sumário)
- Itens de teste (versões de programas, meio onde estão armazenados, acesso a bibliotecas, ambientes, dentre outros)
- Módulos ou *features* do software (projeto de teste para cada módulo ou combinação de módulos) a serem testados;
- Módulos que não serão testados;
- Abordagem de teste (métricas, metodologias, ferramentas);
- Critério de identificação de defeito;
- Critérios de interrupção e finalização dos testes;
- Documentação de teste;
- Identificação dos testes que serão realizados e como serão executados;
- Necessidade de equipamentos ou software;
- Responsabilidade da equipe envolvida;
- Definição da equipe de teste e das suas necessidades;
- Cronograma de atividades de teste;
- Riscos e contingências;
- Critérios de aprovação dos testes pelas áreas envolvidas.

- Seleção das condições de teste;

Identificação das condições de teste e a priorização deles junto a área gestora ou usuários.

- Modelagem das condições de teste;

Elaboração dos casos de teste, de acordo com a IEEE 829 deverá conter:

- Identificador da especificação do caso de teste;
- Especificação de entrada;
- Especificação de saída;
- Necessidade de ambiente;
- Requerimentos de procedimentos especiais;
- Dependência de outro caso de teste.

- Execução dos testes;

Efetiva execução dos casos de teste e sua efetiva coleta de evidências de realização.

- Verificação dos resultados;

Avaliar de acordo com critérios estabelecidos, se o resultado está de acordo ou não e sua sinalização.

- Avaliação dos critérios de conclusão;
Avaliar de acordo com critérios estabelecidos no Plano de Teste, se os testes atingiram ou não os critérios de conclusão;
- Atividades de Encerramento;
Reunir e documentar as atividades realizadas, bem como registrar sua execução na base histórica e lições aprendidas, se for o caso.

1.7. Referências

PEZZE, M.& YOUNG, M. Teste e Análise de Software: processos, princípios e técnicas. Porto Alegre: Bookman, 2008. ISBN: 978-85-778-0262-3.

BARTIE, A. Garantia da Qualidade de Software. Rio de Janeiro: Elsevier, 2002. ISBN: 978-85-352-1124-5.

DELAMARO, M. E.; MALDONADO, J.C. & JINI, M. Introdução ao teste de software, Rio de Janeiro: Elsevier, 2007. ISBN: 978-85-352-2634-8.

HUMBLER, J. & FARLEY, D. **Entrega contínua**. Porto Alegre: Bookman, 2014.
<https://integrada.minhabiblioteca.com.br/#!/books/9788582601044>