****

**CENTRO UNIVERSITÁRIO ASSIS GURGACZ**

**SANDRO MATHEUS RAMOS**

**TIPOS DE TESTES**

**DE SOFTWARE**

**CASCAVEL – PR**

**2021**

****

**CENTRO UNIVERSITÁRIO ASSIS GURGACZ**

**SANDRO MATHEUS RAMOS**

**TIPOS DE TESTES**

**DE SOFTWARE**

Trabalho apresentado na Disciplina de Projeto de Programação Web, do Curso de Engenharia de Software do Centro Universitário Assis Gurgacz.

Professor: Elaine Zanini

**CASCAVEL – PR**

**2021**

**Sumário**

1. Introdução

2. O que é o SWEBOK?

3. O que é o Teste De Software?

4. Fundamentos do Teste de Software

5. Níveis de Teste de Software

5.1 Teste de Unidade(Unitários)

5.2 Teste de Integração

5.3 Teste de Sistema

5.4 Outros Níveis de teste

6. Tipos de Teste de Software

6.1 Baseado na intuição e experiência do engenheiro de software:

6.2 Baseada na especificação técnica

6.3 Baseado No Código

6.4 Baseada em falhas

6.5 Baseada no uso

6.6 Baseada na natureza da aplicação

6.7 Seleção e combinação

7. Objetivos do Teste de Software

8. Conclusão

9. Referências

# 1. Introdução

Nas últimas décadas, a Engenharia de Software se subdividiu em diversas áreas, a quantidade de informação aumentou tanto que a constante especialização profissional se tornou algo comum para atingir o nível de excelência requerido pelo mercado, e profissionais que não se adaptam a essa constante evolução se tornam obsoletos em um curto período de tempo. O termo Engenharia de Software foi usado pela primeira como tema na conferência da OTAN no ano de 1968, evento esse que foi motivado pela crise no desenvolvimento de software. Nesta época, viu-se necessária a busca por métodos padronizados de desenvolvimento de software, visto que a grande maioria era construído sem padronização nenhuma. John Ronald Graham comentou: "construímos sistemas como os irmãos Wright construíam aviões - constrói-se de uma só vez, empurra-se para o despenhadeiro, deixa bater e começa tudo outra vez" (Naur & Randell, 1968).

No auxílio a esses profissionais e para atingir uma padronização foi criado o SWEBOK (Software Engineering Body of Knowledge), assim como o PMBOK(Project Management Body of Knowledge) ajuda os gerentes de projeto nas mais diversas áreas/disciplinas de um projeto, o SWEBOK ajuda os Engenheiros de Software nas mais diversas áreas da construção de um *software*, ele é um guia para o corpo de conhecimento em Engenharia de Software, patrocinado pelo IEEE (Institute of Electrical and Electronics Engineering). Se tratando um artigo relacionado a testes de software, veremos a visão do SWEBOK quanto a testes, suas definições e de como atingir a qualidade provida pelos testes na entrega de um software (SWEBOK, 2004).

A atividade de teste não é uma tarefa simples, ela exige um bom planejamento durante a execução para ser bem sucedida. E a falta de conhecimento dos programadores ou analistas sobre o processo de teste de software pode fazer com que muitos erros passem despercebidos, resultando em um software sem qualidade, e assim trazendo insatisfação aos clientes (ONEDAYTESTING, 2017).

# **2. O que é o SWEBOK?**

## 

## Não podemos falar de uma disciplina(testes) de Engenharia de Software sem citar o SWEBOK, pois nele estão compilados vários anos de estudos e pesquisa em busca melhores práticas nas diversas disciplinas que ele aborda. Podemos dizer que ele é um guia completo com todo conhecimento atual sobre a Engenharia de Software, ele é organizado de modo sistemático e estruturado, esses conhecimentos são divididos em quinze áreas de conhecimentos, que são elas conforme SWEBOK (2004):

* Requisitos de Software;
* Design de Software;
* Construção de Software;
* Teste de Software;
* Manutenção de Software;
* Gerenciamento de Configuração de Software;
* Gerenciamento de Engenharia de Software;
* Processos de Engenharia de Software;
* Modelos e Métodos da Engenharia de Software;
* Qualidade de Software;
* Práticas Profissionais em Engenharia de Software;
* Economia em Engenharia de Software;
* Fundamentos de Computação;
* Fundamentos Matemáticos;
* Fundamentos de Engenharia;

Como podemos ver Teste de Software é uma das disciplinas que são englobadas pelo SWEBOK. Nos próximos tópicos abordaremos os temas que contemplam esta disciplina a fim de termos um melhor entendimento e uma visão geral sobre testes, seus níveis, tipos e como aplicá-los da forma correta.

# **3. O que é Teste de Software?**

# 

Como estamos dando enfase nas ideias propostas no SWEBOK vermos primeiramente sua definição sobre o que são testes.

Segundo o SWEBOK, Teste de Software “é uma disciplina de engenharia de software que trata da avaliação da qualidade do produto e da identificação de defeitos e problemas. Consiste na verificação do comportamento de um programa em um conjunto finito de casos de teste”. Também não podemos deixar de citar Pressman (1995, p. 724) outro grande autor que define testes como “conformidade a requisitos funcionais e de desempenho explicitamente declarado, a padrões de desenvolvimento claramente documentados e a características implícitas que são esperadas de todo software profissionalmente desenvolvido”. Verificamos que ambos os autores ressaltam a importância de se verificar a se o software apresenta suas as características esperadas, e o modo de sanar tal dúvida é com um bom conjunto de testes.

Também temos de visualizar o software como elemento de sistema e os “custos” envolvidos associados às falhas de software são forças propulsoras para uma atividade de teste cuidadosa e bem planejada (PRESSMAN, 1995).

# **4. Fundamentos do Teste de Software**

# 

Neste tópico do SWEBOK ele nos apresenta os conceitos e terminologias que formam a base de conhecimento para compreensão do papel da manutenção de software, como por exemplo, a natureza, categorias, custos, evolução de software, entre outras (SWEBOK, 2004).

Quanto a terminologia, podemos verificar que esta é utilizada de acordo com a norma IEEE Standard 610.12-1990, que nos apresenta 3 termos relacionados a teste de software, que são elas:

* Falha (do inglês fault ): causa do mal funcionamento
* Defeito (do inglês failure ): resultado indesejado
* Erro (do inglês error ): uma ação humana que produz um resultado incorreto.

# 5. Níveis de Teste de Software

O Teste de Software deve sempre ser executado e em diferentes níveis durante o desenvolvimento do software e sua manutenção, tais níveis são o de Unidade, Integração e de Sistema. Cada um com seu aspecto e trazendo qualidade tanto para o nível micro quanto para o nivél macro do software. (SWEBOK, 2004).

## **5.1 Teste de Unidade(Unitários)**

Micro e pequenas empresas que estão na base do mercado, porém, com um balanço positivo sendo possível visualizar uma grande chance de crescimento.

## **5.2 Teste de Integração**

Micro e pequenas empresas que estão na base do mercado, porém, com um balanço positivo sendo possível visualizar uma grande chance de crescimento.

## **5.3 Teste de Sistema**

Micro e pequenas empresas que estão na base do mercado, porém, com um balanço positivo sendo possível visualizar uma grande chance de crescimento.

## **5.4 Outros Níveis de Teste**

# **6**. Tipos de Teste de Software

A CreateOne enfrente concorrentes com plataformas de fácil interação do usuário para o desenvolvimento como a Wix, Weebly, Webnode e One.com, que oferecem valores baixos para sua utilização.

# 6.1 Baseado na intuição e experiência do engenheiro de software

# 6.2 Baseada na especificação técnica

# 6.3 Baseado No Código

# 6.4 Baseada em falhas

# 6.5 Baseada no uso

# 6.6 Baseada na natureza da aplicação

# 6.7 Seleção e combinação

# **7. Objetivos do Teste de Software**

# 

Ao analisar o ambiente interno de uma empresa é possível determinar quais são as forças encontradas e que devem ser alavancadas e também quais são as fraquezas que devem ser minimizadas.

# 8. Conclusão

# 

Determinamos um padrão de qualidade sobre a confiabilidade em nossos serviços, determinada pelo portfólio de trabalhos realizados. Sendo este, próximo ao da concorrência.

# 9. Referências

# IEEE, **IEEE Standard Glossary of Software Engineering Terminology – IEEE Std 610.12.1990,** 1990, IEEE, New York, EUA.

# SWEBOK 2004, **Guide for the Software Engineering Body of Knowledge,** 2004 version, IEEE Computer Society, California, EUA.

# PRESSMAN, R.S. **Engenharia de software**. São Paulo: Makron Books. 1995.

# NAUR, RANDELL. **Software Engineering: Report on a Conference Sponsored by the NATO Science Committee. Technical report**, NATO, Garmisch, Germany.

ONE DAY TESTING. **TESTE DE SOFTWARE: INTRODUÇÃO, CONCEITOS BÁSICOS E TIPOS DE TESTES**. 2017. Disponível em: <https://blog.onedaytesting.com.br/teste-de-software/>. Acesso em: 20 agosto 2021.