

TESTE DE SOFTWARE: UM ESTUDO DE CASO

Marina Aparecida da Silva¹, Douglas da Silva², Jaciara Silva Carosia³

¹ Discente do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas /
marina.silva30@fatec.sp.gov.br

² Discente do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas /
douglas.silva132@fatec.sp.gov.br

³ Docente do Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas/
jaciara.carosia@fatec.sp.gov.br

RESUMO

O teste de software é um processo fundamental no ciclo de desenvolvimento de software. A atividade de teste e qualidade de software, nos dias de hoje, é indispensável para que as empresas tenham um produto que seja competitivo no mercado. A execução de teste de software tem o objetivo de encontrar falhas adicionadas no decorrer do processo de desenvolvimento. Acredita-se que os gastos com correção de falhas aumentam com o avanço das fases, mesmo assim, em algumas empresas a fase de teste ainda não é realizada em tempo hábil para atividade ou os testes são realizados pelos próprios desenvolvedores do sistema. Este artigo tem como objetivo principal apresentar alguns conceitos de teste de software. Através de um estudo de caso realizado em empresa de desenvolvimento de software, pode-se constatar a importância de executar testes por uma equipe qualificada e bem estruturada, tal resultado foi levantado.

Palavras-chave: Teste de software; Desenvolvimento de software; Qualidade.

1 INTRODUÇÃO

A engenharia de software tem evoluído significativamente nos últimos anos explorando novas técnicas, critérios, métodos e ferramentas para a produção de software, em decorrência da crescente utilização de sistemas computacionais em praticamente todas as áreas da atividade humana, provocando um aumento significativo na demanda por qualidade e produtividade (JORGE; BEZERRA; COUTINHO; MONTEIRO; ANDRADE, 2015).

O processo de desenvolvimento de software abrange uma série de passos nos quais, apesar das técnicas, métodos e ferramentas empregados, erros no produto ainda podem ocorrer com uma certa frequência (BARBOSA; MALDONADO; VINCENZI; DELAMARO; SOUZA; JINO, 2004). Na generalidade, a engenharia de

software oferece toda a estrutura para o desenvolvimento e manutenção do software, sendo o teste de software uma de suas subáreas de conhecimento. A área de teste de software tem se destacado pela grande importância no desenvolvimento de sistemas que atendem a requisitos de qualidade (SILVA; MÜLLER; BERNARDI; 2011).

O teste de software tem por objetivo analisar um programa com o propósito de descobrir erros e defeitos, podemos considerar que o teste de software é apto a diminuir a probabilidade de ocorrência de defeitos no produto, a fim de minimizar os riscos para a empresa que o fornece, garantindo assim a qualidade do mesmo (SANTOS, 2016).

O objetivo principal do presente artigo é contribuir com a divulgação de conceitos de teste de software e da importância dessa atividade para a melhoria da qualidade do produto. O objetivo específico é mostrar como a atividade é feita na prática através de um estudo de caso realizado em empresa desenvolvedora de software da cidade de Mococa-SP.

2 REFERENCIAL TEÓRICO

Teste de Software

A atividade de teste de software corresponde uma análise dinâmica do produto e é uma atividade pertinente para a identificação e eliminação de erros persistentes (BARBOSA; MALDONADO; VINCENZI; et al, 2004). O teste de software, bem concretizado, não deve ser feito na intenção de provar que o software funciona. A finalidade fundamental do teste é simplesmente analisar um programa pretendendo descobrir erros e defeitos, convertendo-se assim em uma atividade investigativa, envolvendo ações durante todo o ciclo de desenvolvimento, com o objetivo de descobrir seus problemas (ARCANJO JÚNIOR, 2011).

Inicialmente precisamos entender a diferença entre Defeitos, Erros e Falhas. O defeito está intimamente relacionado a algo que está implementado de forma incorreta, sendo que já faz parte do software. O erro é a manifestação concreta de um defeito, ou seja, é a diferença do valor obtido com o valor esperado. Já a falha, é um comportamento do software diferente do que está sendo esperado, em outras

palavras, é um resultado errado, que decorre de um defeito (SANTOS, 2016). A figura a seguir ilustra esses conceitos.

Figura 1 - Diferença entre Defeito x Erro x Falha.



Fonte: DIAS NETO, 2015, p.55.

Teste de software é visto como um processo fundamental para atingir um software de boa qualidade (CRUZ; JINO; CRESPO; ARGOLLO, 2006). Segundo Tosetto e Bellini (2008), qualidade de software é a conformidade a requisitos funcionais e de desempenho sem restrições declarados, com padrões de desenvolvimento nitidamente documentados e a características implícitas que são esperadas de todo software desenvolvido, ou seja, para que se tenha uma produção de software com qualidade superior, é necessário que haja um processo formal e padronizado de desenvolvimento.

A etapa de testes de software é muito desafiadora para o engenheiro de software, é nesta fase que se cria uma série de casos que têm por objetivo assolar o software que ele mesmo construiu. Diante disso, a atividade de teste é um passo do processo de engenharia de software que, na maioria das vezes, é visto como destrutivo, ao invés de construtivo (BUSCH; LODI; BORBA; 2007). No processo de testes encontramos dois objetivos diferentes. O primeiro objetivo corresponde em mostrar ao desenvolvedor e ao cliente que o programa atende aos requisitos. O segundo objetivo corresponde em descrever os casos em que o sistema pode se comportar de maneira distinta das especificações (SALOMÃO, 2016).

A fase de testagem deve passar por uma etapa em que acontece a entrada de dados e arquivos especialmente montados e, intencionalmente, devem ser inseridos todos os tipos de erros possíveis, todas as operações inválidas e todas as exceções

possíveis. Todas as incapacidades e todos os erros devem ser corrigidos. A simulação deve ser repetidas inúmeras vezes até que funcione corretamente (BUSCH; LODI; BORBA; 2007).

Testes Manuais ou Testes Automatizados

Testes manuais compreendem no desenvolvedor ou até mesmo o próprio usuário final, utilizar o sistema com o intuito de encontrar irregularidades no funcionamento do software. É considerado o tipo de teste menos eficiente, pois dificilmente uma pessoa conseguiria testar incansavelmente um sistema de maneira que não restassem nenhuma possibilidade de ocorrências de falhas (LEAL; 2009).

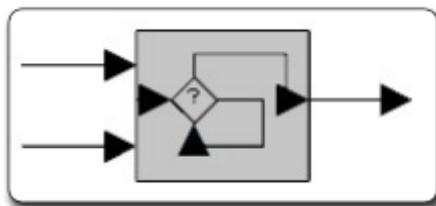
Entretanto, ao recorrer aos testes manuais, a possibilidade de encontrar problemas reais é bem grande, algo que os testes automatizados não conseguem verificar, já que não estão preparados para validar esse tipo de comportamento. Os testes manuais suportam mudanças, uma vez que é mais rápido mudar o paradigma da execução manual (LOPES; 2021).

Os testes automatizados utilizam aplicativos específicos que são capazes de testar incansavelmente um software através de scripts de testes pré-configurados. Não representam 100% de cobertura, visto que algumas falhas lógicas acontecem somente através de combinações específicas de entrada de dados (LEAL; 2009).

Os testes são executados de maneira rápida e eficaz, se tornando um processo rentável a longo prazo. O resultado de cada teste pode ser validado por qualquer pessoa, uma vez que está visível a todos, algo que não está acessível nos testes manuais (LOPES, 2021).

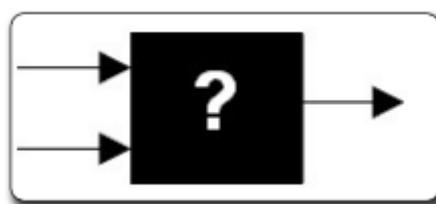
Teste Estrutural (Caixa Branca) e Teste Funcional (Caixa Preta)

O teste estrutural ou teste de caixa branca tem como objetivo testar a estrutura interna do software, permitindo que cada caminho lógico seja testado pelo menos uma vez (FRANZEN; BELLINI, 2005). Nesta técnica é avaliado o comportamento interno do sistema, ou seja, o código-fonte, é recomendada nos níveis de Teste de Unidade e Teste de Integração, testes estes que ficam a cargo dos desenvolvedores do código-fonte, que detêm maior conhecimento do mesmo (SOUZA; GASPAROTTO, 2013), como ilustra a figura a seguir.

Figura 2 - Técnica de Teste Estrutural.

Fonte: DIAS NETO, 2015, p.57.

O teste funcional ou teste de caixa preta o testador não tem acesso ao código-fonte do programa. O objetivo é realizar o teste sobre as diversas funcionalidades e verificar se os resultados obtidos estão de acordo com o esperado (BUSCH; LODI; BORBA, 2007), são testes conduzidos na interface do software, sem preocupação com a estrutura interna do mesmo (ARCANJO JÚNIOR, 2011), conforme mostra a figura 3.

Figura 3 - Técnica de Teste Funcional

Fonte: DIAS NETO, 2015, p. 57.

O quadro a seguir descreve sobre as principais diferenças entre teste estrutural (ou caixa branca) e teste funcional (ou caixa preta).

Quadro 1 - Diferença entre os testes de Caixa Branca e Caixa Preta.

Tipo de teste	Teste Caixa Branca	Teste Caixa Preta
Visibilidade do testador	Tem visibilidade do código e escreve casos de teste baseado no código;	Não têm visibilidade do código escreve os casos de teste baseados em possíveis entradas e saídas e das funcionalidades documentado nas especificações e/ou requisitos;

O que revela um caso de teste que falha?	Um problema (falha);	Um sintoma de um problema (falha);
Controlado?	Sim – O caso de teste ajuda a identificar as linhas específicas do código envolvido;	Não – pode ser difícil encontrar a causa do fracasso no teste.

Fonte: ARCANJO JÚNIOR, 2011, p. 15.

Verificação e Validação

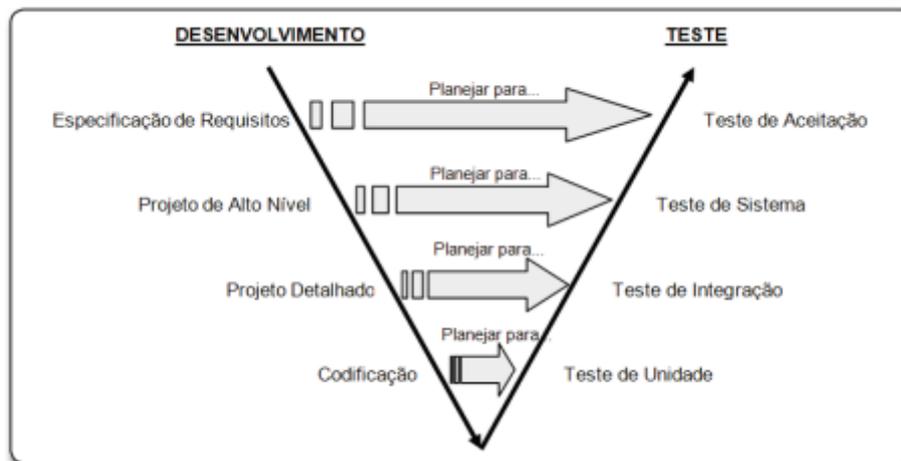
Dentre os possíveis defeitos inseridos durante o desenvolvimento do projeto sejam encontrados o mais rápido possível, é importante que as atividades de verificação e validação sejam executadas (SOUZA; GASPARATTO, 2013).

O objetivo do processo de verificação é checar se o software atende as suas especificações funcionais e não funcionais (SALOMÃO, 2016), ou seja, é garantir que o sistema está sendo desenvolvido de maneira correta, seguindo todas as normas e metodologias adotadas para a realização do projeto (SOUZA; GASPARATTO, 2013). Enquanto o processo de validação tem como objetivo garantir que o software desenvolvido para o cliente atenda às suas expectativas (SALOMÃO, 2016), e tenha sido construído conforme descrito nos requisitos, sendo necessário navegar dentro do sistema para detectar possíveis diferenças entre o sistema e os artefatos do cliente (SOUZA; GASPARATTO, 2013).

Os testes de verificação e validação são complementares e necessitam ser executados no decorrer de todo o ciclo de vida do processo de teste. Apesar de serem vistos como independentes, são responsáveis por reforçar o processo de constatação de erros, o que aumenta a garantia de bom funcionamento do produto, como resultado a eficiência do software (ARCANJO JÚNIOR, 2011).

O modelo em V, ilustrado na figura a seguir, é um dos mais aceitos atualmente, pois ele ressalta as atividades de verificação e validação com o objetivo de encontrar defeitos gerados durante o desenvolvimento do software e diminuir os riscos do projeto. Este modelo auxilia o entendimento de que a atividade de teste não deve ser realizada apenas no final do projeto e sim em todo o desenvolvimento do projeto (SOUZA; GASPARATTO, 2013).

Figura 4 - Modelo V descrevendo o paralelismo entre as atividades de desenvolvimento e teste de software.



Fonte: DIAS NETO, 2015, p. 56.

Tipos de Testes

Segundo Salomão (2016) os sistemas de softwares comerciais devem passar por esses três estágios de testagens. São eles:

1. Testes em Desenvolvimento: Os testes são feitos durante o desenvolvimento do software, a fim de encontrar erros e falhas e os desenvolvedores podem estar envolvidos neste processo.
2. Teste de release: Nesta etapa uma equipe de testes independente testa a versão completa do sistema antes de entregá-lo aos usuários, está também verifica se o software atende a todos os requisitos especificados.
3. Teste de Usuário: Os usuários testam o sistema determinando se o produto está apto ou não para ser comercializado.

Níveis de Teste de Software

O planejamento dos testes deve ocorrer em diferentes níveis, os principais níveis de teste de software são:

- Teste de Unidade: tem por objetivo investigar a menor unidade do projeto, procurando ocasionar falhas provocadas por falhas lógicas e de implementação nos módulos separadamente (DIAS NETO, 2015), ou seja, o teste de unidade executa as funções e procedimentos mais específicos do software (SANTOS, 2016).

- É o tipo de teste mais importante, já que ele testa se um algoritmo faz o que realmente deveria ser feito e garante que o código encapsulado por uma unidade deve executar o efeito colateral esperado (BERNARDO; KON, 2008).
- Teste de Integração: pretende provocar falhas associadas às interfaces entre os módulos quando integrados para construir a estrutura do software (DIAS NETO, 2015), é a fase para a construção da estrutura do programa e encontrar erros associados a interface (LUFT, 2012), ou seja, as funções e os procedimentos são fixados para serem realizados juntos, buscando a descoberta de erros entre eles (SANTOS, 2016).
- Teste de Sistema: é realizado após a integração do sistema e pretende identificar os erros das funções, onde verifica se elas estão de acordo com as especificações dos requisitos, apresentados antes do início do desenvolvimento (SANTOS, 2016). Dessa forma, os testes são executados nos mesmos ambientes, com as mesmas circunstâncias e com os mesmos dados de entrada que o usuário utilizaria no dia a dia de manuseio do software (DIAS NETO, 2015).
- Teste de Aceitação: é a etapa em que o teste é conduzido por usuários finais do sistema, onde simulam operações de rotinas do sistema a fim de verificar se seu comportamento está de acordo com o exigido (DIAS NETO, 2015).
- Teste de Regressão: envolve aplicar, a cada nova versão do software ou a cada ciclo (DIAS NETO, 2015), segmentos já testados após a realização de uma mudança em outra parte do software, que visam garantir a integridade do software após a realização dessas modificações (SALOMÃO, 2016), minimizando assim os efeitos colaterais.

3 METODOLOGIA

A pesquisa baseia-se pela análise qualitativa dos dados. Assim, não se objetiva, a representatividade numérica, mas sim uma maior aproximação entre a problemática elencada pela pesquisa, a fim de formular uma compreensão mais

profunda sobre teste de software. A adoção da abordagem da pesquisa qualitativa apresenta métodos com o intuito de explicar o motivo pelo qual ocorrem os fenômenos que estão sendo estudados (POUPART et al., 2008).

Trata-se de uma pesquisa descritiva, através da exposição de conceitos e fatos sobre teste de software. Foi realizado um levantamento bibliográfico sobre o tema, que permitiu embasar a execução de um estudo de caso. Através das leituras realizadas foi elaborado um questionário de teste de software, a fim de verificar métodos de testes existentes nas empresas de software de TI.

O questionário foi elaborado utilizando o Microsoft Word 2010 e estruturado em três partes, sendo que a primeira parte contém um quadro de identificação da empresa, a segunda é composta por seis questões destinadas a delinear as características gerais da empresa e a terceira parte com 14 questões específicas de testes, e para a realização da pesquisa, foi encaminhado este questionário por e-mail para 3 empresas de desenvolvimento de software da cidade de Mococa – SP.

A coleta de dados da pesquisa durou aproximadamente 3 semanas e destas 3 empresas, 1 empresa respondeu ao questionário.

Primeira parte:

NOME DA EMPRESA	
ENDEREÇO	
TELEFONE	
EMAIL	

Segunda parte:

- 1) Tempo da empresa no mercado?
- 2) Quantidade de funcionários?
- 3) Quantidades de projetos que podem ser desenvolvidos simultaneamente?
- 4) Quantidade de funcionários exclusivos para teste?
- 5) Qual o nível de investimento em testes e qualidade?
- 6) Quais as principais linguagem de desenvolvimento?

Terceira parte:

- 1) Frequência de execução de testes?
- 2) Presença de equipes de testes?
- 3) Frequência em que os testes são reexecutados?
- 4) Realização de testes automatizados?
- 5) Caracterização da gestão de testes?
- 6) Qual fase é encontrada a maior número de falhas no software?
- 7) Quais são os tipos de abordagens que têm sido utilizadas para auxiliar o teste de software?
- 8) Quais são as fases de teste de software que mais têm sido contempladas?
- 9) Quais são as linguagens utilizadas para auxiliar no teste de software?
- 10) Percentual de esforço do projeto dedicado a testes?
- 11) Investimento que a empresa estaria disposta a fazer em ferramentas de software?
- 12) Documentos de testes elaborados pela empresa?
- 13) Frequência de geração de relatórios de qualidade?
- 14) Papel da atividade de teste para a empresa?

4 RESULTADOS E DISCUSSÃO

Por meio das respostas obtidas realizou-se a análise e caracterização da empresa.

A empresa está no mercado de desenvolvimento de software há 5 anos, possui 3 funcionários, tendo a capacidade de desenvolver 2 projetos simultaneamente.

A empresa realiza a atividade de teste desde que começou a atuar no mercado de desenvolvimento de softwares, realizando teste de caixa branca, teste de caixa-preta, teste de sistema e teste de integração.

Os responsáveis por realizar os testes são os próprios desenvolvedores do sistema e uma empresa de auditoria contratada para auxiliar na parte fiscal e tributária, sendo que a execução dos testes é iniciada na fase de implementação do software.

A maior incidência de erros está aplicada no processo de desenvolvimento do software na parte de validações fiscais, erros de código, interpretação da análise dos

requisitos e de funcionalidade, que são os mais comuns de serem encontrados nos sistemas desenvolvidos.

As dificuldades que a empresa encontra para a realização dos testes é a falta de profissionais especializados na área de testes, a dificuldade de implantar um processo de teste, o desconhecimento de técnicas de testes adequadas e o fator mais agravante, ser um processo muito caro.

A empresa realiza os testes manualmente e de maneira automatizada. Segundo Luft (2012) as principais vantagens em se utilizar os testes automatizados em comparação com os testes feitos manualmente é a eliminação do trabalho repetitivo de inserir os dados e observar os resultados, a melhoria na qualidade do processo e do produto, além do aumento da produtividade e a diminuição do custo destinados as atividades de testes.

A atividade de teste não é documentada, sendo documentada somente os testes que são realizados pela empresa de auditoria. A documentação é uma parte de extrema importância quando o assunto é atividade de testes. Por meio dela é possível ter um controle do que está sendo analisado, o que já foi analisado e o que ainda falta analisar, além de ser uma forma de comunicação entre as equipes de testes e fornece ao cliente evidências da qualidade do software (LUFT, 2012).

O plano de testes é um documento que caracteriza o planejamento para a execução do teste. É o documento principal para que os gerentes de projeto de testes consigam gerenciar e controlar todos os projetos de testagem de software. É por meio deste documento que são descritas equipes de execução, atividades, recursos, cronogramas, estratégias de testes, abordagens e outros (ARCANJO, 2011). A não construção de um plano de testes implica em vários problemas, sendo eles: o próprio desenvolvedor executar a atividade de teste, falta de entendimento dos riscos reais do projeto, não identificar todas as possibilidades de testes possíveis antes de iniciá-los, falta de envolvimento dos usuários, alterações emergentes sendo liberadas sem a realização dos testes, entre outros. Por isso é fundamental a existência desse documento para que seja realizado um bom gerenciamento dos testes.

Segundo Salomão (2016), é indicado que o percentual de esforço para o teste de um projeto varie no intervalo em torno de 30 a 40%. O percentual de esforço da empresa que é projetado para a atividades de testes é de 30%. Os testes apresentam

grande contribuição, havendo uma significativa diminuição dos custos e dos erros que são encontrados após a implantação da atividade de testes.

Estão relacionados a seguir algumas limitações importantes desta pesquisa:

- Acesso apenas a um entrevistado da empresa;
- Recusa de duas empresas em participarem desta pesquisa;
- Grande subjetividade a estudos de casos apoiados por entrevistas;
- Impossibilidade de observar diretamente as atividades de testes de softwares realizados na empresa.

5 CONSIDERAÇÕES FINAIS

A execução do teste de software é uma das atividades mais importantes, entretanto uma das mais difíceis no processo de desenvolvimento de software, pois envolve uma qualidade significativa dos recursos de um projeto. O autodomínio e o alto custo, aliado a esta atividade dependem principalmente da criticidade da aplicação a ser desenvolvida.

Este trabalho apresentou os diferentes tipos de testes que podem ser realizados, além de apresentar a importância da fase de testes para que se entregue e obtenha um software de boa qualidade. Entretanto, para que isso ocorra é necessário planejamento, envolvimento e capacitação dos profissionais e tempo hábil para que a execução dos testes seja completa.

As dificuldades encontradas para a execução deste trabalho de conclusão de curso, foi encontrar empresas que estivessem dispostas a responder o questionário elaborado e ao pouco material bibliográfico encontrado que abordasse a temática de testes automatizados.

REFERÊNCIAS

ARCANJO JÚNIOR, A. PROCESSO DE TESTE DE SOFTWARE: Uma descrição com a perspectiva da qualidade de software. Belo Horizonte. 2011. 40 p. **Monografia** (Departamento de Ciência da Computação) – Universidade Federal de Minas Gerais.

Disponível em: https://repositorio.ufmg.br/bitstream/1843/BUOS-94NM37/1/antonio_arcanjo.pdf. Acesso em: 10 Out. 2021.

BARBOSA, E. F.; MALDONADO, J. C.; VINCENZI, A. M. R.; et al. **Introdução ao Teste de Software** (versão 2004-01). [S.l: s.n.], 2004.

BERNARDO, P. C; KON, F. A Importância dos Testes Automatizados. **Engenharia de Software Magazine**, v.1, n.3, p. 54-57, 2008. Disponível em: <https://www.ime.usp.br/~kon/papers/EngSoftMagazine-IntroducaoTestes.pdf>. Acesso em: 27 nov. 2021.

BUSCH, A. V.; LODI, R. A.; BORBA, S. F. P. Metodologia de desenvolvimento e aplicação de testes para qualidade de software. **Rev. Ciên. Empresariais da UNIPAR**, Umuarama, v. 8, n. 1 e 2, p. 125-139, jan./dez. 2007.

CRUZ, J. L.; JINO, M.; CRESPO, A. N.; ARGOLLO, M. Suporte automatizado à rastreabilidade em um processo de teste de software baseado em documentação. SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE (SBQS), 5., 2006, Vila Velha. **Anais [...]**. Porto Alegre: Sociedade Brasileira de Computação, 2006. p. 278-292. Disponível em: <https://doi.org/10.5753/sbqs.2006.15615>. Acesso em: 27 nov. 2021.

DIAS NETO, A. C. Introdução a Teste de Software. **Engenharia de Software Magazine**. Edição 1, p. 54-59, Maio/2015. Disponível em: https://edisciplinas.usp.br/pluginfile.php/3503764/mod_resource/content/3/Introducao_a_Testes_de_Software.pdf. Acesso em: 28 Out. 2021.

POUPART, Jean et al. **A pesquisa qualitativa: enfoques epistemológicos e metodológicos**. Tradução: Ana Cristina Nasser. Petrópolis: Vozes. 2008.

FRANZEN, M. B.; BELLINI, C. G. P. Arte ou prática em teste de software? **REAd - Revista Eletrônica de Administração**. Edição 45 V. 11, n. 3, p. 1-25, mai/jun 2005. Disponível em: <https://seer.ufrgs.br/read/article/view/40653/25859>. Acesso em: 25 Out. 2021.

JORGE, F. F.; BEZERRA, C. I. M.; COUTINHO, E. F.; MONTEIRO, J. M.; ANDRADE, R. M. C. A Evolução do Jogo iTestLearning para o Ensino das Atividades de Execução de Testes de Software. **Nuevas Ideas en Informática Educativa TISE 2015**, p. 295 – 305. Disponível em: <http://www.tise.cl/volumen11/TISE2015/295-305.pdf>. Acesso em: 08 Nov. 2021.

LEAL, I. Requisitos de Metodologias de Teste de software para processos ágeis. **Departamento de Ciência da Computação – Universidade Federal de Minas Gerais**, 2009. Disponível em: <https://homepages.dcc.ufmg.br/~rodolfo/dcc823-1-09/Entrega2Pos/igor2.pdf>. Acesso em: 15 out. 2021.

LOPES, C. S. Automação de testes e o impacto no desenvolvimento de software. Portugal, 2021. 102 p. **Dissertação de Mestrado (Instituto Politécnico de Viseu) –**

Escola Superior de Tecnologia e Gestão de Viseu. Disponível em: <http://hdl.handle.net/10400.19/6719>. Acesso em: 27 nov.2021.

LUFT, C. C. Teste de Software: uma necessidade das empresas. Santa Rosa, 2012. 90 p. **Dissertação de Mestrado (Departamento de Ciências Exatas e Engenharias) – Universidade Regional do Noroeste do Estado do Rio Grande do Sul.** Disponível em: <https://bibliodigital.unijui.edu.br:8443/xmlui/bitstream/handle/123456789/1307/TCC%20-%20Cristiane%20Carline%20Luft.pdf?sequence=1&isAllowed=y>. Acesso em: 25 Nov. 2021.

SALOMÃO, R. G. Análise da Relevância de Teste de Regressão para o Mercado de Desenvolvimento de Software do Triângulo Mineiro. Uberlândia, 2016. 33 p. **Monografia** (Faculdade de Ciência da Computação) – Universidade Federal de Uberlândia – UFU. Disponível em: <https://repositorio.ufu.br/bitstream/123456789/19594/1/An%C3%A1liseRelev%C3%A2nciaTestes.pdf>. Acesso em: 08 Nov. 2021.

SANTOS, D. B. Implantação de teste de software em empresa de pequeno porte: Um estudo de caso. Marília, 2016. 78 p. **Monografia** (Bacharelado em Sistema de Informação) – Centro Universitário Eurípides de Marília – UNIVEM. Disponível em: <https://aberto.univem.edu.br/bitstream/handle/11077/1577/Monografia%20Daniel%20Botter%20dos%20Santos.pdf?sequence=1&isAllowed=y>. Acesso em: 15 Out. 2021.

SILVA, T. G.; MÜLLER, F. M.; BERNARDI, G. Panorama do Ensino de Engenharia de Software em Cursos de Graduação Focado em Teste de Software: Uma Proposta de Aprendizagem Baseada em Jogos. **Novas Tecnologias na Educação**, V. 9, n 2, p. 1 -10, dezembro, 2011. Disponível em: <https://seer.ufrgs.br/renote/article/view/25115>. Acesso em: 15 Nov. 2021.

SOUZA, K. P.; GASPAROTTO, A. M. S.; A importância da atividade de teste do desenvolvimento de software. **XXXIII ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO** A Gestão dos Processos de Produção e as Parcerias Globais para o Desenvolvimento Sustentável dos Sistemas Produtivos Salvador, BA, Brasil, 08 a 11 de outubro de 2013.

TOSETTO, M.; BELLINI, C. G. P. Gestão sociotécnica do teste de software em projetos de sistemas de informação. **Revista de Gestão da Tecnologia e Sistemas de Informação**. Vol. 5, No. 2, p. 325-346, 2008. Disponível em: <https://www.scielo.br/j/jistm/a/xMpw9Y8PWgkSg6C7Zdd9Lkh/?lang=pt&format=pdf>. Acesso em: 27 nov. 2021.