

FACULDADE VENDA NOVA DO IMIGRANTE

LEONARDO PAIM MAGALHÃES

**IMPORTÂNCIA DO USO DE METODOLOGIA NA REALIZAÇÃO
DE TESTES EM UMA EQUIPE DE TESTES DE SOFTWARE**

CAXIAS DO SUL

2023

FACULDADE VENDA NOVA DO IMIGRANTE

LEONARDO PAIM MAGALHÃES

**IMPORTÂNCIA DO USO DE METODOLOGIA NA REALIZAÇÃO
DE TESTES EM UMA EQUIPE DE TESTES DE SOFTWARE**

Trabalho de conclusão de curso apresentado como requisito parcial à obtenção do título especialista em Gestão de Qualidade de Software.

CAXIAS DO SUL

2023

IMPORTÂNCIA DO USO DE METODOLOGIA NA REALIZAÇÃO DE TESTES EM UMA EQUIPE DE TESTES DE SOFTWARE

Leonardo Paim Magalhães¹,

Declaro que sou autor(a)¹ deste Trabalho de Conclusão de Curso. Declaro também que o mesmo foi por mim elaborado e integralmente redigido, não tendo sido copiado ou extraído, seja parcial ou integralmente, de forma ilícita de nenhuma fonte além daquelas públicas consultadas e corretamente referenciadas ao longo do trabalho ou daqueles cujos dados resultaram de investigações empíricas por mim realizadas para fins de produção deste trabalho.

Assim, declaro, demonstrando minha plena consciência dos seus efeitos civis, penais e administrativos, e assumindo total responsabilidade caso se configure o crime de plágio ou violação aos direitos autorais. (Consulte a 3ª Cláusula, § 4º, do Contrato de Prestação de Serviços).

RESUMO- O presente artigo aborda assuntos referentes a importância dos métodos e processos utilizados para garantir a qualidade na atividade de testes no desenvolvimento de software. O objetivo é contextualizar a realização de testes e mostrar através de revisão bibliográfica a relevância que se pode ter quando utilizada corretamente pela equipe de testes uma metodologia que direcione a realização de testes de software. A demanda por qualidade tem motivado a comunidade de software para o desenvolvimento e aplicação de modelos para garantir a qualidade de software, neste artigo são descritos conceitos de qualidade de software com base nos autores pesquisados que fundamentam a importância da utilização da metodologia correta e os riscos que a empresa pode enfrentar caso não aplique corretamente essa metodologia durante o desenvolvimento de softwares para obtenção de maior confiabilidade dos seus produtos de software.

PALAVRAS-CHAVE: Software. Testes. Qualidade. Metodologia. Desenvolvimento.

¹ E-mail do autor: leonardopm@msn.com

1 INTRODUÇÃO

Atualmente a questão da qualidade de software tornou-se uma necessidade para que empresas (especialmente do ramo de desenvolvimento de software) possam conquistar e fidelizar seus clientes. Cada vez mais são exigidas destas empresas garantias dos processos de qualidade de software adotados pela equipe de testes e demais envolvidos nesta etapa.

Considerando a importância que a garantia da qualidade de software tem tido nas organizações que desenvolvem software, observa-se a crescente utilização de modelos e metodologias que auxiliam estas organizações a alcançar melhoria nos processos que envolvem os testes de software. Importante ressaltar ainda que a realização de testes não consiste simplesmente na geração e execução dos testes, mas envolve também questões de planejamento, gerenciamento e análise dos resultados.

Os processos e normas para se ter um software confiável tornam indispensável no processo de desenvolvimento de software o uso de métodos, técnicas e ferramentas que permitam a realização da atividade de testes, aumentando a qualidade do produto de software. Entende-se com isso a importância da utilização de metodologias nesta atividade.

Dentre muitos modelos de gerenciamento da qualidade de software, alguns dos mais conhecidos são o *Capability Maturity Model* (CMM), o modelo de Melhoria do Processo de Software Brasileiro (MPS.BR), ou ainda normas da *International Organization for Standardization* (ISO). Este trabalho não entra no detalhe sobre os modelos, mas aborda a importância do uso de um modelo como os citados ou semelhante.

Neste sentido, este artigo tem como objetivo geral demonstrar a importância da utilização de uma metodologia durante a realização dos testes de software no ciclo de desenvolvimento de software buscando alcançar qualidade do produto de software. No decorrer serão apresentados os fatores importantes para adoção de metodologia, seus benefícios, e ainda os malefícios de não se seguir uma metodologia que oriente a realização dos testes pela equipe responsável.

Destaca-se que, visto a importância que os softwares têm no mundo contemporâneo, podendo ser utilizado em diversas atividades críticas, é necessária a devida importância por parte dos profissionais responsáveis quanto à busca contínua pela qualidade no produto de software. Deste modo é de extrema relevância a conscientização das empresas onde há o desenvolvimento de software quanto a produção de softwares com cada vez mais qualidade.

No decorrer deste artigo será apresentada uma revisão bibliográfica buscando apresentar pontos relevantes com base em pesquisa bibliográfica realizada. Para tanto serão considerados outros trabalhos e publicações que tenham como foco o processo de realização de testes e seus impactos positivos e negativos em um contexto onde a garantia de qualidade de software se torna progressivamente indispensável.

O restante do artigo está organizado da seguinte maneira. A seção 2, de Desenvolvimento, apresenta os conceitos e ressalta os pontos relevantes ao tema referentes a metodologia nos testes de software. A seção 3, Conclusão, apresenta o fechamento deste artigo com os resultados obtidos neste estudo. Por fim, na seção 4, Referências, o referencial bibliográfico utilizado na fundamentação teórica deste trabalho.

2 DESENVOLVIMENTO

Neste capítulo serão apresentados os conceitos relacionados ao tema de acordo com a pesquisa bibliográfica realizada. Essa pesquisa por referencial bibliográfico buscou como fontes artigos científicos e publicações relevantes sobre o tema abordado.

Para a construção deste trabalho são utilizados materiais e conhecimentos obtidos por meio de referenciais teóricos pesquisados através de leituras seletivas, com foco em justificar a importância do uso de metodologia para garantir que a qualidade de software na realização de testes durante o desenvolvimento de software seja mantida no maior nível possível.

Considerando a questão abordada neste artigo, já introduzindo o assunto levantado, Cauzzi (2015, p. 17) faz um destaque sobre a crescente expansão da utilização de sistemas informatizados na maioria das atividades das pessoas,

fazendo com que a qualidade de software deixe de ser um diferencial dos produtos e passa a ser uma exigência.

Conforme descrito por Correia, Ferreira e Pires (2021, p. 2), algumas empresas se encontram presas a antigos paradigmas que impedem (ou atrasam) a adoção de abordagens modernas de qualidade de software que surgiram com os avanços na área de desenvolvimento de software. Os mesmos autores ainda complementam que, se falando de um “cenário no qual são envolvidas diversas variáveis durante o processo de desenvolvimento de sistema, percebe-se que a complexidade deste tende a aumentar proporcionalmente”.

No decorrer deste capítulo serão explorados os conceitos necessários para entendimento do assunto abordado.

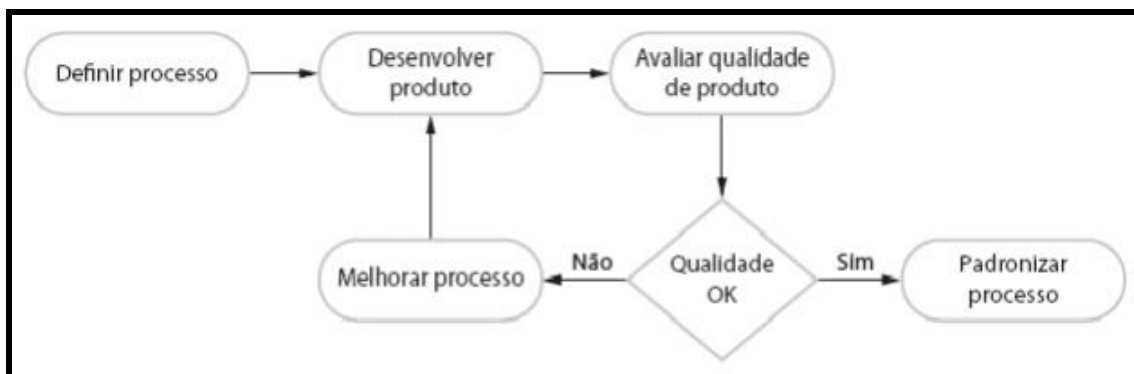
2.1 Qualidade de Software

A qualidade de software trata-se de uma das áreas da engenharia de software que objetiva garantir a qualidade do produto de software através do processo de desenvolvimento definido (PERUCCI; CAMPOS, 2016, apud DA SILVA, 2022, p. 21). Complementando, Cauzzi (2015, p. 18) considera que, se a expectativa é construir softwares de alta qualidade, é dever da equipe ser capaz de criar um projeto que seja adequado ao problema e, ao mesmo tempo, apresente características que levem a um software com as dimensões e fatores de qualidade.

Segundo Sambo (2017, apud DA SILVA, 2022, p. 21), a qualidade de software é um processo metódico, focado nos artefatos e em todas as etapas desenvolvidas, que busca garantir a conformidade de produtos e processos e processos, evitando ou eliminando possíveis defeitos. Para Martins (2012, p. 25, apud KALB, 2014, p. 26) é importante ainda destacar que a qualidade de software é um conceito complexo, que não pode ser definido de maneira simples, mas que classicamente significa que o produto desenvolvido deve cumprir sua especificação.

Segundo Pinto (2011, p. 3) existem empresas em todo o mundo que se distinguem pela qualidade dos produtos que vendem e dos serviços que prestam, aos quais se aliam fatores como elevado grau de competitividade, eficiência de processos e satisfação crescente dos clientes. A Figura 1 apresenta um modelo de Sommerville (2011, apud SILVA; DALLILO, 2019, p. 3) sobre a representação das etapas que constroem a qualidade, se baseando em processos.

Figura 1 – Qualidade e suas etapas



Fonte: SOMMERVILLE (2011, apud SILVA; DALLILO, 2019, p. 3)

Conforme Pressman e Maxim (2016, apud DA SILVA, 2022, p. 22) o processo de garantia de software envolve uma série de atividades a serem seguidas, dentre elas, a atividade de Testes de Software.

2.2 Testes de software

Pressman e Maxim (2016, apud DA SILVA, 2022, p. 22) definem os testes de software como “atividade de controle de qualidade, com objetivo de encontrar erros”. Os autores afirmam ainda que é necessário que na qualidade de software seja garantido que os testes sejam devidamente planejados e conduzidos de forma correta, com maior probabilidade possível de atingir seu principal objetivo.

Conforme Souza e Gasparotto (2013, p. 2), na década de 70 a execução de testes nos softwares era feita pelos próprios desenvolvedores de sistemas, a atividade era vista como uma tarefa secundárias, sem muita importância, feita apenas se o prazo de entrega e custo do produto permitisse. Ainda conforme as autoras, com o passar do tempo, devido à concorrência existente no mercado e ao aumento da complexidade dos sistemas, o nível de exigência por qualidade aumentou e com isso a necessidade de mais testes.

Para Molinari (2012, apud SOUZA; GASPAROTTO, 2013, p. 2), a atividade de testes é uma das etapas do ciclo de desenvolvimento de software e tem o objetivo de relatar possíveis defeitos existentes no sistema para que sejam solucionados. Nesta fase verifica-se se o comportamento do sistema está de acordo com o especificado nos requisitos levantados, e com isso diagnosticar o grau de

qualidade do sistema, reduzindo a probabilidade de ocorrência de erros quando o sistema estiver em produção.

Complementando, no ponto de vista de Neto (2007, p. 2) o conceito de testes de software pode ser compreendido através de uma visão intuitiva ou mesmo de uma maneira formal, sendo que existem várias definições para esse conceito. O autor afirma ainda que de uma forma simples, testar software significa verificar através de uma execução controlada se o seu comportamento corre de acordo com o que foi especificado.

Ainda conforme Neto (2007, p. 2), o objetivo principal da tarefa de execução de testes de software é revelar o número máximo de falhar dispondo do mínimo de esforço, ou seja, mostrar aos que desenvolvem se os resultados estão ou não de acordo com os padrões estabelecidos.

Sobre os recursos aplicados a testes, Konscianski e Soares (2006, apud CAUZZI, 2015 p. 34) entendem que os esforços para os testes de software devem ser compensatórios, ou seja, deve haver um balanceamento entre o tempo e o custo de testes em um software e a quantidade de defeitos encontrados. Os recursos alocados para testes (pessoas, tempo e ferramentas), segundo os autores, fazem parte da definição do projeto e não devem ser decididos na realização de testes em si. Pode ser utilizada como base para a definição dos recursos o histórico de experiências anteriores da empresa.

Considerando uma questão mais técnica sobre testes, para Stephens (2015, apud ARUANA, 2017, p. 20) afirma que a facilidade de corrigir erros está diretamente ligada com a ideia de encontrá-los previamente e corrigi-los, podendo ter apoio com a utilização dos níveis de testes. O mesmo autor define os testes como possuindo 6 níveis: Teste de unidade, Teste de Integração, Testes Automatizados, Teste de Interface do Componente, Teste de Sistema e Teste de Aceitação.

Para Souza e Gasparotto (2013, p. 13), no contexto da realização de testes de software, após a elaboração dos artefatos e liberação dos casos de uso para os testes, inicia-se a ação da equipe de testes, que percorre os cenários descritos nos casos de testes e registra os casos onde ocorram erros.

2.3 Equipe de Testes

Não há uma regra sobre as funções que podem compor uma equipe de testes, sendo que há variações de acordo com cada organização ou projeto. Rodrigues e Estrela (2012, apud RODRIGUES, 2013, p. 5) afirmam que as empresas de desenvolvimento de software têm necessidades próprias, de acordo com suas características de trabalho, tipos de projetos desenvolvidos, clientes, dentre outras.

Geralmente são encontradas nas empresas de desenvolvimento de software, ou em empresas de outros ramos que tem desenvolvimento de software interno, as funções de Gerente de Testes, Analista de testes e Testador sendo que as atribuições de cada função também podem sofrer variações em cada situação.

O Gerente de Testes, segundo Aruana (2017, p. 47) pode ser responsável por organizar e distribuir a equipe de acordo com as demandas que chegam para realização de testes de software, sendo responsável por definir o que será executado e cada priorização. Além disso, avaliar os resultados obtidos nas execuções e o rendimento dos testadores e analistas de testes. O autor ressalta ainda que o gerente de testes está no mesmo patamar do gerente de projetos, logo a opinião dos dois tem o mesmo peso nos momentos de decisão.

Conforme a definição de Feller (2016, p. 2), o Analista de Testes tem por função principal gerenciar o planejamento de testes, o que envolve demandas como receber e analisar demandas de teste; elaborar documentação; participar de reuniões de projeto; buscar estudar ferramentas para a melhoria do trabalho da equipe; propor e executar processos e metodologias de trabalho, além de também ser um executor de testes ou acompanhar a execução dos testes, criando relatórios de testes, sendo resultado positivo ou negativo.

Já o Testador, também conforme a definição de Feller (2016, p. 2), tem como papel a execução de testes e geração de relatórios de testes (sucesso ou erro), mas também pode buscar e estudar ferramentas que ajudem no processo de trabalho, propor melhorias sempre que achar necessário, além de possuir liberdade de testar itens que não foram contemplados no plano de testes, mas que ache importante passarem por uma avaliação mais criteriosa.

Sobre os papéis envolvidos na realização de testes, Aruana (2017, p. 29) faz uma ressalva. O autor explica que durante o desenvolvimento de software é normal

que desenvolvedores e testadores tenham entendimentos diferentes do que representa uma falha, ou que uma documentação esteja um pouco confusa e possa gerar dupla interpretação.

A tarefa de testes exige uma organização antes do seu início, conforme sugerido por Perry (2006, apud ARUANA, 2017, p. 24). Na organização são definidas questões relacionadas ao tempo e recursos necessários para que o processo seja concluído com sucesso. Segundo o autor a organização desse processo pode passar cinco passos: Definir gerente de testes; Definir o escopo do teste; Definir a equipe de testes; Verificar a documentação; e Validar a Estimativa do teste.

Kalb (2014, p. 28) abre ainda a possibilidade de se ter uma organização independente para realização dos testes de software, com testadores e consultores de testes externos, que podem trazer uma imparcialidade muito necessária para os processos de testes de melhor qualidade.

Com a organização dos testes de software, conforme ARUANA (2017, p. 26), o processo de testes é iniciado e ele é basicamente definido em três partes: planejamento, execução e avaliação dos resultados. Esse processo visa garantir a qualidade do produto de software.

2.4 Garantia de Qualidade de Software

Para entender o conceito, o Instituto de Engenheiros Eletrônicos e Eletricistas (2010, apud RODRIGUES 2013, p. 4) define que a garantia da qualidade é o conjunto de atividades planejadas para avaliar os processos pelos quais os produtos são desenvolvidos ou fabricados; o controle da qualidade, por sua vez, é o conjunto de atividades planejadas para avaliar a qualidade dos produtos desenvolvidos ou fabricados. Rodrigues (2013) conclui então que a garantia da qualidade de software está focada na qualidade do processo, enquanto que o controle da qualidade está relacionado à qualidade produto.

Segundo Andrade (2015, apud DA SILVA, 2022, p. 21), na garantia de qualidade do software, seus atributos devem ser satisfeitos de forma a atender as necessidades dos envolvidos. Estes atributos são definidos de acordo com sua relevância para o software em questão: tecnologia utilizada, características

específicas dos projetos, necessidades da organização e do usuário e domínio da aplicação.

Sobre controle de qualidade, Cauzzi (2015, p. 19) complementa que para que se garanta a qualidade de software deve existir um controle de qualidade. Para Bartiè (2002, apud CAUZZI, 2015, p. 19), o controle de qualidade se concentra no monitoramento e desempenho dos resultados do projeto para determinar se ele está atendendo aos padrões de qualidade no processo de desenvolvimento.

Conforme dito por Souza e Gasparotto (2013, p. 3), devido ao aumento da exigência por maior qualidade, várias normas, metodologias ou modelos (ágil e *Rational Unified Process* (RUP), por exemplo), e órgãos reguladores foram criados, como CMMI, MPS.BR e ISO.

2.5 Metodologias de Testes de Software

Para Sommerville (2004, p. 141, apud KALB, 2014, p. 28), aumentou o interesse por parte das organizações que tem desenvolvimento de software pelo avanço de suas técnicas. O avanço da metodologia significa compreender os métodos existentes e poder alterá-los. Com esse objetivo obtido, o abatimento dos valores e do período pode se tornar a fundamental meta do avanço.

Uma das abordagens geralmente utilizadas na realização de testes buscando a garantia da qualidade de software é referente a estratégias de testes. Conforme Cauzzi (2015, p. 43) a estratégia adotada determina o modo como irá se estabelecer o teste e como serão conduzidos os procedimentos de testes visando minimizar esforços e ampliar as chances de detecção de erros que estão inseridos no software. Da Silva (2022, p. 16) complementa que na aplicação de engenharia de software estão envolvidos os processos de software por meio dos quais os desenvolvedores são capazes de avaliar e aplicar essas estratégias que possam contribuir com a qualidade do sistema.

Ainda sobre estratégias de testes, Silva e Dallilo (2019) enfatizam que as estratégias de testes são importantes porque são capazes de otimizar o tempo, ou seja, evitam que ele seja desperdiçado assim como o esforço que poderia ser dispensado e evidencia a possibilidade da descoberta de erros que podem ser solucionados logo nos testes, almejando conseguir uma maior agilidade e controle nos procedimentos operacionais durante a execução dos testes. Souza e Gasparotto

(2013) afirmam que nas estratégias de testes devem ser analisados métodos, níveis e tipos de testes que serão adotados, e que devem ser considerados diversos fatores como: o porte e importância do software, os requisitos, os prazos assumidos, os riscos do projeto, entre outros.

Independente da metodologia adotada pela empresa ou projeto, os autores pesquisados concordam sobre a necessidade de uma forte documentação (que podem ser chamados de artefatos) para garantia de qualidade de software. Para Cauzzi (2015, p. 47) todo projeto de software que passa por um processo de qualidade deve ser bem documentado. O autor cita que a principal norma para documentação de testes é a IEEE 829, do Instituto de Engenheiros Eletrônicos e Eletricistas, onde as formas de documentação propostas são Casos de Testes e Planos de Testes. Complementando, Neto (2007) apresenta ainda como elementos essenciais: Critério de Testes, Critério de Cobertura dos Testes, Critério de Adequação de Casos de Teste e Critério de geração de Casos de Teste.

Outro ponto importante na garantia de qualidade de software é o uso de métricas para medir essa qualidade. Cauzzi (2015, p. 49) afirma que as métricas de software são essenciais para a avaliação do processo de desenvolvimento. Nessa linha, Rios e Filho (2013, apud CAUZZI, 2015, p. 50) o uso de métricas é a etapa que leva ao controle e, eventualmente a melhoria. Se não se mede, não se pode entender o processo. Se não se entende o processo não se pode controlá-lo. Se não pode controlá-lo, não pode aperfeiçoá-lo nem gerenciá-lo.

Bartiè (2002, apud CAUZZI, 2015, p. 51) afirma que a qualidade do software determina o quanto esse software está próximo dos requisitos e será através dos defeitos encontrados em cada uma das fases que será possível determinar a distância entre o patamar desejado de qualidade o que está sendo realizado. O autor afirma ainda que será através das análises dos resultados destes indicadores que serão tomadas importantes decisões como a finalização da etapa, a implantação definitiva do produto no cliente ou a negociação por aumento de prazo ou recursos para o projeto.

Para Kalb (2014, p. 30) é por meio do avanço dos processos de software que se visa chegar a um acréscimo de qualidade e a uma obra final que atenda o mercado em geral. Dentre esses avanços, uma opção é a utilização de exemplos de processos para ter como referência, como os modelos de qualidade de software. O autor apresenta como exemplos: o Modelo CMMI (*Capability Maturity Model*

Integration) com a sua vertente para desenvolvimento CMMI-DEV; o modelo ISO/IEC 9126, que trata da avaliação do produto de software do ponto de vista de suas características de qualidade (atributos de Funcionalidade, Confiabilidade, Usabilidade, Eficiência, Manutenibilidade e Portabilidade) e; o modelo MPS, que acredita-se que seja compatível com os padrões de qualidade aceitos internacionalmente e a sua capacidade existente nos modelos de avanços de processos já disponíveis, ficando em acordo com as principais abordagens internacionais para definição, avaliação e melhoria de processos de software.

2.6 Importância do Uso de Metodologias

Pinto (2011, p. 5) afirma que estudos relacionam a questão da qualidade com a obtenção de vantagens competitivas para as organizações, ao nível de satisfação dos clientes, da inovação ou da eficiência dos processos.

Bartiè (2002, apud CAUZZI, 2015 p. 18) e Koscianski e Soares (2006, apud CAUZZI, 2015, p. 18) afirmam que qualidade de software é um processo sistemático que focaliza todas as etapas e artefatos produzidos, dependendo do correto emprego das boas metodologias pelos colaboradores, com o objetivo de garantir a conformidade de processos e produtos, prevenindo e eliminando defeitos.

Pressman e Maxim (2016, apud DA SILVA, 2022, p. 22), dentre a série de atividades a serem seguidas para garantia da qualidade de software, apresentam a Educação como fator primordial para o aperfeiçoamento dos envolvidos no processo de desenvolvimento de software, sendo a garantia de qualidade de software a líder no processo de aperfeiçoamento do software, sendo um proponente fundamental e patrocinador de programas educacionais.

Sommerville (2011, apud SILVA; DALLILO, 2019, p. 2) afirma que os padrões são muito importantes para gerenciar e garantir a qualidade de software, selecionando e definindo quais os padrões que devem ser aplicados durante o processo de desenvolvimento (e testes). O autor corrobora ainda que, como parte do processo de gestão de qualidade, devem ser escolhidas ferramentas e métodos para suportar o uso desses padrões.

Conforme Cauzzi (2015, p. 51), visto que são necessários muitos esforços para alcançar o nível de qualidade de software esperado, torna-se fundamental a existência de indicadores que possibilitem avaliar o volume de erros detectados em

cada etapa. Conforme o autor, medir a eficiência da verificação é importante, pois as atividades de verificação são trabalhosas e introduzem custos e esforços adicionais no processo e por isso os testes de eficiência tem sua importância. Conclui-se então que há custos para a organização que não aplica corretamente testes de software.

2.7 Custos da Não Utilização de Metodologias em Testes

Para Molinari (2012, apud SOUZA; GASPAROTTO, 2013, p. 5), existem empresas que negligenciam a atividade de testes devido ao custo e tempo exigidos. Muitas vezes os testes ainda são executados por desenvolvedores ou outro profissional que tenha conhecimento de regras de negócio do sistema. Também é comum que a fase seja abreviada para que o prazo e custo sejam alcançados, porém, pode-se então entregar ao cliente um software com defeitos não revelados. Tal atitude pode acarretar algo custoso à empresa visto a necessidade de solucionar defeitos encontrados após a entrega, assim como a queda de credibilidade e satisfação do cliente.

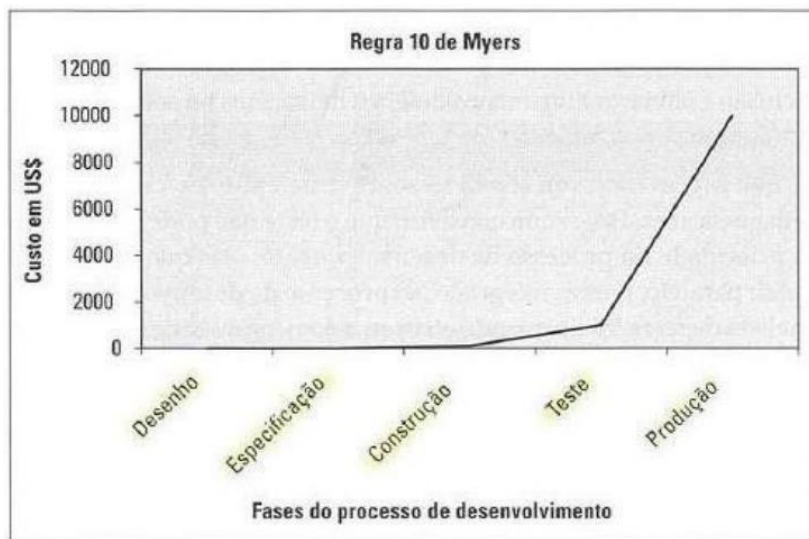
Souza e Gasparotto (2013, p. 4) deixam claro que o teste de software não é uma atividade barata e fácil, e que normalmente o valor gasto em testes varia entre 30% a 40% do total do projeto, dependendo das técnicas de testes que foram utilizadas e da tolerância a falhas exigida. Porém, caso não sejam realizados os testes seguindo procedimentos bem desenhados de acordo com o processo organizacional, esse custo provavelmente fica muito maior.

Conforme Kalb (2014, p. 28), quando os modelos de negócios e os mercados mudam mais rápido do que as aplicações que os suportam pode ser desenvolvida, o teste de software é muitas vezes o primeiro a ser cortado do orçamento ou cronograma, apesar dos efeitos que os defeitos podem causar. O autor afirma que mesmo um pequeno número de defeitos pode ter um impacto na empresa, clientes e parceiros. Estima-se que um defeito de software encontrado e os custos pós-produção tenha de 100 vezes mais do que se o defeito foi encontrado na fase de concepção.

Souza e Gasparotto (2013, p. 3) explicam que, além de colaborar para a obtenção de um produto com qualidade, a atividade de testes tem o papel fundamental na redução de custos com possíveis reparos ao sistema. Segundo elas, quanto mais cedo descoberto e corrigido um erro, menor é o seu custo para o

projeto, ou seja, defeitos encontrados nas fases iniciais do desenvolvimento do software são mais baratos de serem corrigidos do que aqueles encontrados quando o software já está em produção. A figura 2 apresenta um gráfico sobre essa lógica.

Figura 2 – Custo dos defeitos com o passar das fases do projeto



Fonte: BASTOS *et al.* (2017, apud SOUZA; GASTAPAROTTO, 2013, p. 4)

Segundo Silva e Dallilo (2019, p. 9) a introdução de softwares com qualidade no mercado mostra que a realização de testes é um processo importante e essencial na produção e desenvolvimento de um bom sistema. Os autores destacam ainda que a falta de um processo de testes planejado poderá acarretar na demora da entrega ao cliente, encarecimento do custo do produto, frustrações e outros problemas, tanto para a empresa responsável como para o cliente.

3 CONCLUSÃO

O objetivo do presente artigo é apresentar a importância de metodologia no processo de garantia de qualidade de software, não apenas no contexto de desenvolvimento, mas também nos benefícios que uma empresa que desenvolve softwares alcança quando aplica métodos para qualidade de software, como a confiança do cliente interno ou externo que recebe o produto com mais credibilidade, melhorando a imagem da empresa no mercado.

Conforme os autores pesquisados, não seria adequado se os testes fossem executados sem a existência de técnicas. Muito provavelmente a grande parte deles seria deficiente, desorganizada e sem propósitos bem definidos, assim a qualidade do produto de software ficaria comprometida.

O não investimento na atividade de testes pode trazer complicações financeiras e a falta de qualidade. Um sistema além de um produto é uma ferramenta de trabalho indispensável nas organizações, logo, deve ser confiável, não ter inconsistências nos dados, perdas de informações, entradas de dados incompletos, entre outros.

Apesar dos custos envolvidos com o gerenciamento da qualidade de software, é importante que tais métodos sejam introduzidos, a fim de que uma empresa sobreviva por um longo tempo e possa estar sempre atualizada, com um nível de produção de software de alta qualidade, satisfazendo sempre as necessidades de seus clientes.

Respondendo ao proposto neste artigo, é necessário, portanto que os testes sejam executados por profissionais capacitados usando a metodologia apropriada para o contexto da empresa ou projeto. Diferentes técnicas podem ser utilizadas e diferentes requisitos de qualidade devem ser garantidos, seja em metodologias tradicionais ou até em metodologias ágeis.

4 REFERÊNCIAS

ARUANA, J. G. S. **Estratégias para realizar testes funcionais de interface com o usuário – visão de uma equipe de testes**. 2017. 90 f. Monografia (Graduação em Engenharia de Software) – Universidade de Brasília, Brasília.

CAUZZI, E. J. **Proposta de plano de garantia da qualidade de software para o laboratório de criação e aplicação de software**. 2015. 141 f. Trabalho de Conclusão de Curso (Bacharelado em Sistemas de Informação) – Universidade de Caxias do Sul, Caxias do Sul, 2015.

CORREIA, J. M. G.; FERREIRA, J. V. C.; PIRES, D. F. **Importância dos Testes para a Qualidade do Software**. 2021. 20 f. Artigo (Departamento de Computação) – Centro Universitário Municipal de Franca, Franca. 2021.

DA SILVA, L. G. **Análise das Atividades de Teste de Software quanto a sua Aplicabilidade e Importância em Empresas Privadas**. 2022. 55 f. Trabalho de Conclusão de Curso (Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas) – Instituto Federal de Educação, Ciência e Tecnologia do Rio Grande do Norte, Nova Cruz.

FELLER, N. J. **Disseminando a Cultura de Teste e Qualidade de Software no CPD-UFRGS**. 2016. 4 f. Artigo (Centro de Processamento de Dados) – Universidade Federal do Rio Grande do Sul. Disponível em: <https://lume.ufrgs.br/handle/10183/142255>. Acesso em: 3 mai. 2023.

KALB, D. F. **APLICABILIDADE DA QUALIDADE DE SOFTWARE**: estudo de caso com nível G do MPS.BR em uma empresa de pequeno porte. 2014. 74 f. Trabalho de Conclusão de Curso (Curso de Ciência da Computação do Departamento de Ciências Exatas e Engenharia) – Universidade Regional do Noroeste do Estado do Rio Grande do Sul, Ijuí.

NETO, A. C. D. **Introdução a teste de software**. Engenharia de Software Magazine, v. 1, p. 22, 2007. Disponível em: https://www.researchgate.net/profile/Arilo-Neto/publication/266356473_Introducao_a_Testes_de_Software/links/5554ee6408ae6fd2d821ba3a/Introducao-a-Testes-de-Software.pdf. Acesso em: 7 mai. 2023.

PINTO, J. C.; PINTO, A. L. **A importância da certificação de sistemas de gestão da qualidade em Portugal**. Revista de Gestão dos Países de Língua Portuguesa, v. 10, n. 1-2, p. 48-61, 2011. Disponível em: <https://scielo.pt/pdf/rpbg/v10n1-2/v10n1-2a06.pdf>. Acesso em: 12 mai. 2023.

RODRIGUES, N. N. **Praticando Qualidade de Software: Ensinando e Aprendendo seus Valores através de Ambiente Real**. 2013. 12 f. Artigo (Instituto

Federal de Educação, Ciência e Tecnologia da Paraíba), João Pessoa. Disponível em: <https://sol.sbc.org.br/index.php/sbsi/article/view/5713>. Acessado em 8 mai. 2023.

SILVA, M. L. M.; DALLILO, F. D. **Uma visão geral sobre automação de testes**. Revista Científica Multidisciplinar Núcleo do Conhecimento. 2019. Disponível em: <https://www.nucleodoconhecimento.com.br/tecnologia/automacao-de-testes>. Acessado em 13 mai. 2023.

SOUZA, K. P. S.; GASPAROTTO, A. M. S. **A Importância da Atividade de Teste no Desenvolvimento de Software**. *In*: Encontro Nacional de Engenharia de Produção, 33, 2013, Salvador. Disponível em: https://abepro.org.br/biblioteca/enegep2013_TN_STO_177_007_23030.pdf. Acesso em: 3 mai. 2023.