



## **GESTÃO DA QUALIDADE DE SOFTWARE: Aprimoramento no Processo de Acompanhamento e Melhoria de Requisitos.<sup>1</sup>**

Victor Almeida Cunha<sup>2</sup>

Débora Rodrigues Stefanello<sup>3</sup>

### **RESUMO**

O processo de melhoria de requisitos de um software é altamente relevante para o crescimento de uma organização, pois além de envolver a preocupação com os produtos, esse tipo de processo deve se fazer presente em todas as fases de um projeto e desse modo, garantir a eficiência dos produtos a serem desenvolvidos. Com tal finalidade, as organizações passam a fazer uso de padrões de qualidade presentes no mercado, padrões estes que consistem em técnicas e boas práticas já utilizadas e comprovadas por outras equipes, que permite que os padrões de qualidade desejados sejam obtidos no final do processo de desenvolvimento e implantação.

**Palavras-chaves:** Qualidade, Padrões de Qualidade, Melhoria de Requisitos.

### **1 INTRODUÇÃO**

Com a evolução da sociedade, houve um grande avanço em diversas áreas que compõem o crescimento da humanidade, e uma delas foi a área da tecnologia. Um ramo a se destacar foi no desenvolvimento de *software*, no qual há diversas linguagens de programação para se fazer a estruturação. Com esse avanço, ficou claro a importância da tecnologia para o auxílio de diversos tipos de funcionalidades, desse modo, auxiliar em ações de rotinas e processos de gerenciamento empresarial.

A qualidade de software está diretamente ligada a capacidade do mesmo de atender as necessidades do usuário, seja uma pessoa ou uma empresa. Não está contida apenas nos quesitos funcionais do software, mas também nos métodos e processos envolvidos na sua criação (SALGUEIRO; ARTE, 2005).

---

<sup>1</sup> *Paper* apresentado à disciplina de Gestão da Qualidade de Software, do Centro Universitário UNDB.

<sup>2</sup> Aluno do 8º período do curso Sistemas de Informação da UNDB

<sup>3</sup> Professora Mestra.

Visto o aumento gradativo da importância dos softwares na sobrevivência das organizações da atualidade, é vital que o processo de desenvolvimento de tais ferramentas seja bem planejado e executado, pois o resultado terá impactos diretos no funcionamento e no futuro da organização.

A gestão de qualidade visa produzir um sistema que além de atender as necessidades do cliente, apresente uma boa funcionalidade, isto é, com nível baixo de erros. Entretanto, apenas apresentar uma baixa quantidade de erros não é por si só característica definidora de qualidade. Juntamente com tal quesito, um software de qualidade deve facilitar os processos de melhoria contínua, isto é, a constante evolução do sistema e facilidade de resolução de problemas que possam surgir (SOUZA; MONTEIRO, 2009).

A metodologia aplicada para a elaboração deste trabalho acadêmico é baseada em uma pesquisa bibliográfica e explorando artigos com relação ao tema do aprimoramento no processo de acompanhamento e melhoria de requisitos, tendo em vista os mesmos com objetividade clara. No quesito de natureza técnica da pesquisa, destaca-se que será uma pesquisa constituída principalmente de artigos científicos encontrados em blogs, sites acadêmicos contidos na internet.

O trabalho se fez em divisão de tópicos para que seja fácil o entendimento sobre os estudos da gestão da qualidade de software, assim analisando a importância de se melhorar os processos de melhoria de requisitos para que se tenha um software de qualidade de acordo com as normas assim padronizadas para tal finalidade.

## **2 GESTÃO DA QUALIDADE DE SOFTWARE**

### **2.1 GARANTIA DE QUALIDADE PARA MELHORIA DE REQUISITOS**

A fim de se obter garantia que o software a ser criado irá apresentar todas as características que definem um produto de qualidade, se faz necessário seguir uma série de métodos e processos que já se provaram eficientes para tal finalidade. É a partir dessa ideia que surgem os padrões de qualidade. Esses padrões são diretrizes nacionais e internacionais que explicam e ditam a forma com a qual o desenvolvimento em geral deve acontecer. Tais padrões estão geralmente presentes e organizados em manuais, os quais devem ser metodicamente seguidos pela equipe de desenvolvimento, com o intuito de obter resultados com padrões de qualidade esperados (SOUZA; MONTEIRO, 2009).

A utilização de padrões de qualidade oferece uma garantia de que o produto criado irá atender as necessidades dos clientes, pois são constituídos de técnicas e procedimentos já utilizados e comprovados em outros projetos, que passaram por testes, mudanças e evoluções até se tornarem padrões reconhecidos pela comunidade. Existem vários padrões de qualidade de excelência reconhecidos mundialmente, dentre eles pode-se citar: CMMI (Capability Maturity Model Integration – Modelo Integrado de Maturidade da Capacitação), MPS.BR (Melhoria de Processo de Software Brasileiro) e os da ISO (Organização Internacional de Padronização) no qual se destaca a ISO 9001 (SOUZA; MONTEIRO, 2009).

## **2.2 QUALIDADE DE PROCESSO DE DESENVOLVIMENTO DE SOFTWARE**

Os padrões de qualidade consistem das melhores e mais apropriadas práticas para o desenvolvimento de software. Para se chegar no conhecimento que se tem atualmente, foi necessário o aprimoramento de tais práticas, proporcionado através de experiências baseadas em tentativas e erros. Desta forma, os padrões de qualidade permitem que a equipe evite a repetição de erros cometidos no passado, possibilitando maior segurança na construção de softwares.

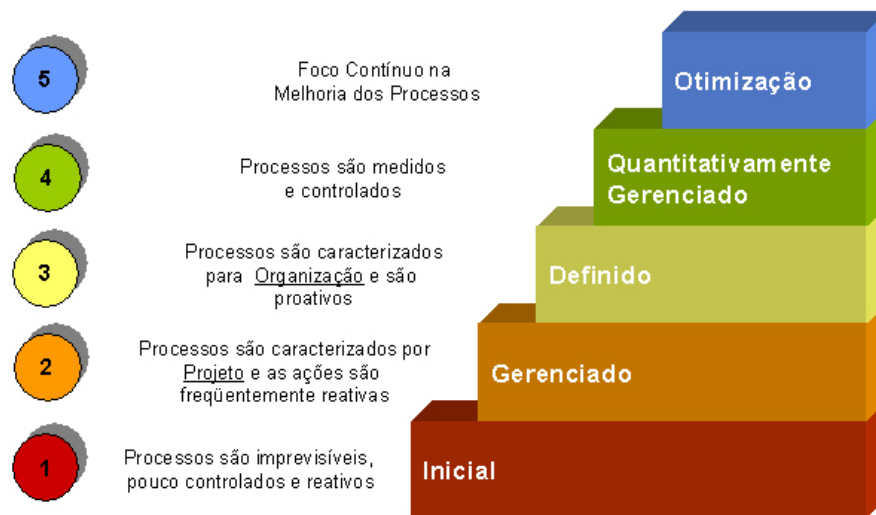
A qualidade do software a ser desenvolvido está diretamente conectada a qualidade dos processos utilizados para sua criação. Desta forma, é possível aumentar a qualidade do produto através do reforçamento dos processos utilizados. Adicionalmente, é possível fazer uso de processos e práticas as quais já são reconhecidas como benéficas para o processo de desenvolvimento. Tais práticas são contidas em chamados modelos de qualidade, que oferecem formas de melhoria e avaliação de processos de desenvolvimento.

Um exemplo a se destacar é o CMMI (Modelo Integrado de Capacitação e Maturidade) que é um modelo para melhoria de processos baseado em níveis de maturidade a ser utilizado no desenvolvimento de produtos e serviços. Outro exemplo é o MPS.BR (Melhoria de Processo de Software Brasileiro), onde segundo Souza e Monteiro (2009) consiste, assim como o CMMI, em um modelo baseado em níveis de maturidade. O MPS.BR por sua vez, visa a implantação do modelo de forma gradual em empresas de pequeno e médio porte. Outro modelo a se destacar é a ISO (Organização Internacional de Padronização) que rege regras para padronizar e assim melhorar processos.

### **2.2.1 MODELO CMMI**

O CMMI (Modelo Integrado de Capacitação e Maturidade) é um modelo para melhoria de processos baseado em níveis de maturidade a ser utilizado no desenvolvimento de produtos e serviços. Esse modelo consiste nas melhores práticas que envolvem desenvolvimento e atividades desde a concepção até a entrega e manutenção do software. O CMMI apresenta cinco níveis de maturidade, que correspondem a capacidade da organização em realizar projetos de grande porte. Assim, quanto maior for o nível de maturidade dos processos, maior será a qualidade obtida no produto final a ser desenvolvido (MONTEIRO; SOUZA, 2009).

Figura 1 – Os cinco níveis de maturidade do CMMI



Fonte: Devmedia (2012)

A seguir constam os níveis de maturidade presentes no CMMI, segundo Monteiro e Souza (2009):

- **Nível 1 (Inicial):** aqui constam organizações que não contam com nenhum tipo de certificação, consequentemente apresentando projetos de desenvolvimento com baixa qualidade de planejamento;
- **Nível 2 (Gerenciado):** neste nível já passa a ser observado organização entre os processos realizados, envolvendo gestão, medição de desempenho e maior facilidade de tomadas de decisão e mudanças;
- **Nível 3 (Definido):** neste nível já é possível observar uma padronização dos processos por parte da organização, permitindo o gerenciamento e administração de projetos;

- Nível 4 (Gerenciado Quantitativamente): acontece um gerenciamento mais detalhado dos processos, onde a organização é capaz de dizer exatamente os pontos mais vitais para o desempenho dos processos;

- Nível 5 (Otimização): Incorpora os processos de melhoria contínua ao padrão de desenvolvimento da empresa. Empresas neste nível de maturidade possui maior facilidade em identificar as causas em seus problemas de desenvolvimento.

Segundo Salgueiro e Arte (2005), os principais objetivos do CMMI são: redução de duplicidades e retrabalho, melhoria de clareza e entendimento, estabelecimento de regras de construção uniformes.

### **2.2.2 MODELO MPS.BR**

O MPS.BR (Melhoria de Processo de Software Brasileiro), segundo Monteiro e Souza (2009) consiste, assim como o CMMI, em um modelo baseado em níveis de maturidade. O MPS.BR por sua vez, visa a implantação do modelo de forma gradual em empresas de pequeno e médio porte. Devido a sua semelhança com o CMMI, os níveis de maturidade poder ser representados de acordo com sua relação entre os modelos da forma que se observa nos itens a seguir:

- Os níveis gerenciado e parcialmente gerenciado, são os níveis mais superficiais do modelo e correspondem ao nível 2 do modelo CMMI
- Os níveis parcialmente definidos, largamente definido e definido correspondem ao nível 3 do modelo CMMI
- O nível gerenciado quantitativamente corresponde ao nível 4 do CMMI
- O nível em otimização corresponde ao nível 5 do CMMI

O MPS.BR foi criado com o intuito de adequar o CMMI para um paradigma voltado ao perfil cultural das empresas, na medida em que também busca definir um modelo de melhoria e avaliação compatível com os padrões de qualidade que são mais utilizados no mercado internacional (MONTEIRO; SOUZA, 2009).

## **2.3 NORMALIZAÇÃO ISO**

A ISO (International Standards Organization - Organização Internacional de Normalização) é uma entidade não governamental responsável por estabelecer normas internacionais. As normas ISO expressam o objetivo de elaborar normas, partindo de um

consenso global, definindo assim regras, diretrizes e características para produtos, documentos e atividades em geral (SALGUEIRO; ARTE, 2005)

As normas ISO tornaram-se bastante populares no Brasil na década de 90 juntamente com a explosão dos ideais de Tecnologias da Informação como ferramenta vital para o funcionamento das organizações. Entretanto, essas normas são utilizadas em processos e produtos de diversas áreas, não apenas do setor de TI.

Segundo Monteiro e Souza (2009) o objetivo das normas ISO é desenvolver um padrão de atividades e práticas que possam ser utilizados em qualquer país. Os autores continuam afirmando que, as normas existem buscando a normalização de produtos e serviços, mantendo a satisfação dos clientes como principal objetivo. Adicionam também os autores, que essas normas se aplicam a diversos ramos de atividade, podendo ser adotadas por qualquer tipo de organização, e acabam por se mostrar como uma boa ferramenta responsável por melhorar a imagem da empresa, gerando um bom apelo de marketing.

### **3 CONTROLE DE QUALIDADE PARA MELHORIA DE REQUISITOS**

O controle de qualidade de software visa avaliar e auxiliar na garantia de qualidade, avaliando o desempenho dos processos que estão sendo utilizados, com o intuito de fornecer ao cliente um produto que atenda suas expectativas.

Salgueiro e Arte (2005) afirmam que os objetivos principais do controle de qualidade são:

1. Planejar a qualidade pelos clientes no qual implica num esforço de localizar o cliente, saber de suas necessidades, traduzir essas necessidades em características mensuráveis, de forma que seja possível gerenciar todo processo;
2. Manter a qualidade desejada pelo cliente onde são cumpridos os padrões previamente acordados;
3. Melhorar a qualidade desejada pelo cliente onde trata de localizar os resultados indesejáveis e utilizar um método de solução de problemas com a finalidade de melhorá-los (SALGUEIRO; ARTE, 2005, p. 14).

Ou seja, para obter melhorias no processo de acompanhamento de melhorias de requisitos, é necessário o monitoramento dos resultados, para que seja possível identificar se os mesmos se encontram nos padrões de qualidade e assim possibilitando eliminar possíveis resultados negativos, melhorando esse processo.

De acordo Oliveira (2014), o controle de qualidade atua diretamente ligado com a garantia de qualidade, este último sendo um termo mais generalizado, abrangendo outros conceitos, que incluem o controle de qualidade. O controle de qualidade em si, exibe no arsenal técnicas como testes de software, revisões e inspeções, todas utilizadas com o objetivo de garantir que o produto final obtido no fim do processo de desenvolvimento atenderá as expectativas e aos requisitos especificados na fase de planejamento.

É de grande importância que os produtos possam ser devidamente avaliados, isto é, apresentar características que possam de fato ser mensuradas, proporcionando para a equipe de desenvolvimento um mecanismo de feedback. Através dessas informações, é possível que a equipe tenha noção de como o projeto está sendo realizado, e se necessário realizar ajustes nos processos de desenvolvimento ou no software em si (OLIVEIRA, 2014).

Algumas técnicas são frequentemente empregadas com o objetivo de manter o controle de qualidade em um processo de desenvolvimento de software, tais como testes de software, refatorações e revisões.

As revisões, consistem em reuniões entre as partes interessadas de um projeto, onde são realizados comentários, discussões e aprovações acerca do produto ou produtos que estão sendo desenvolvidos. As revisões podem tratar da análise e discussão de diferentes etapas do desenvolvimento, como por exemplo revisão de código ou revisão de planejamento (OLIVEIRA, 2014).

Uma outra técnica frequentemente utilizada na área de desenvolvimento de software é a refatoração. A refatoração consiste em alterações na estrutura interna de um software, a fim de trazer melhorias para o seu funcionamento, sem que o comportamento externo do produto seja comprometido. A refatoração é uma boa prática recomendada em um processo de desenvolvimento de software, pois auxilia na estrutura do código, facilitando o entendimento dos membros da equipe de desenvolvimento, assim como também trazendo melhorias na performance do software, tanto em termos de processamento como em termos de utilização de memória. Adicionalmente, este processo ajuda na identificação e correção de erros no código, assim como diminui os custos de manutenção em geral (OLIVEIRA, 2014).

Por fim, fala-se dos testes de software que é uma das atividades de maior importância para averiguar de fato o funcionamento do produto, assim como o cumprimento de seus requisitos. Desta forma, os testes de software buscam descobrir se o produto criado é capaz de realizar seu propósito, exibindo as funcionalidades que se esperam dele de acordo com as especificações. É improvável que um software na fase de testes esteja completamente

desprovido de erros, entretanto, os testes visam amenizar o problema, aumentando assim a confiabilidade no sistema por parte dos clientes (OLIVEIRA, 2014).

#### **4 MÉTRICAS DE QUALIDADE PARA ACOMPANHAR E MELHORAR O PROCESSO DE REQUISITOS**

Nos processos de garantia de qualidade de software, além de um bom planejamento de gestão é importante que a equipe possa avaliar o andamento do trabalho e averiguar se o mesmo condiz com o planejado. Para tal finalidade, existem as métricas de qualidade. Segundo Oliveira (2014), métricas de qualidade são variáveis mensuráveis que indicam se os critérios de qualidade definidos estão sendo alcançados. O autor complementa que, as métricas podem ser classificadas em: métricas de produto, processo ou de projeto, e afirma também que as métricas de qualidade estão diretamente ligadas com as métricas de software, que envolvem aspectos como quantidade de erros, robustez do software, performance e outros aspectos.

De acordo com Oliveira (2014), entre as métricas de qualidade, algumas se destacam como excelentes indicadores para a equipe de desenvolvimento, como por exemplo manutenibilidade e usabilidade.

A manutenibilidade indica a capacidade e facilidade que um produto de software pode ser modificado, seja por necessidades de mudanças nos requisitos, correção de erros ou mudanças no ambiente. A manutenibilidade não pode ser diretamente mensurada, por esse motivo são utilizadas métricas indiretas com o objetivo de avaliá-la. Com isso em mente, é feito a avaliação do tempo médio para mudanças, que consiste no tempo médio gasto nos processos de mudança, desde a requisição da mudança, planejamento, implementação (codificação), testes até a liberação final para o usuário. Em seguida, são feitas comparações do tempo médio para mudanças, com produtos de área semelhante, assim como mudanças e alterações semelhantes, para de fato determinar o grau de manutenibilidade do produto em questão (OLIVEIRA, 2014).

Segundo Oliveira (2014), outra métrica de grande importância para indicar a qualidade do software é a usabilidade. A usabilidade é indispensável para um produto de software, uma vez que indica a forma e a facilidade com que usuários podem fazer uso do software analisado. O autor segue afirmando que, por mais que um software apresente todas as funcionalidades especificadas, se o mesmo não apresentar boa usabilidade, não é de fácil utilização, estará predestinado a falhar como produto. Um software com baixo nível de



usabilidade, muito provavelmente será responsável por uma drástica redução de produtividade no ambiente organizacional.

## **5 CONSIDERAÇÕES FINAIS**

Pode-se concluir que para se tenha um controle no acompanhamento e melhoria de requisitos dos processos, boas práticas e ferramentas de controle devem ser colocadas em prática. Além disso, observou-se que a qualidade dos processos utilizados no desenvolvimento de um software, terão impactos diretos na qualidade do software em si, logo, se faz necessário que atenção seja direcionada ao planejamento de qualidade de processos. Adicionalmente, atenção especial também deve ser atribuída ao controle de qualidade, pois através de tal é possível averiguar o andamento do projeto assim como o cumprimento das especificações estabelecidas no início do mesmo, desse modo evitando possíveis retrabalhos e garantindo de fato, a qualidade do produto de software que venha a ser desenvolvido.

## REFERÊNCIAS

DEV MEDIA. **Maturidade em Desenvolvimento de Software - Revista Engenharia de Software Magazine 52**. 2012. Disponível em: <<https://www.devmedia.com.br/maturidade-em-desenvolvimento-de-software-revista-engenharia-de-software-magazine-52/25977>>. Acesso em: 02 nov. 2019.

GIL, Antônio Carlos. **Como elaborar projetos de pesquisa**. 4. ed. São Paulo: Atlas S.a., 2002.

OLIVEIRA, Bruno Henrique. **Qualidade de software no desenvolvimento com métodos ágeis**. 2014. Disponível em: <<http://www.teses.usp.br/teses/disponiveis/55/55134/tde-12082014-165244/pt-br.php>> Acesso em: 30 de out. 2019.

SALGUEIRO, Luiz Fernando; ARTE, Marcus Vinícius do. **MODELOS DE QUALIDADE DE SOFTWARE**. 2005. 48 f. TCC (Graduação) - Curso de Sistemas de Computação, Universidade Federal do Estado do Rio de Janeiro, Rio de Janeiro, 2005. Disponível em: <<http://bsi.uniriotec.br/tcc/textos/200508SalgueiroMarcus.pdf>>. Acesso em: 27 out. 2019.

SOUZA, André Coura Costa Lopes de; MONTEIRO, Rafael Ferreira. **GESTÃO DA QUALIDADE DE SOFTWARE: Garantia da Qualidade Total**. 2009. 57 f. Monografia (Especialização) - Curso de Ciência da Computação, Universidade Fumec, Belo Horizonte, 2009. Disponível em: <[http://professores.dcc.ufla.br/~terra/publications\\_files/students/2009\\_fumec\\_souza\\_e\\_monteiro.pdf](http://professores.dcc.ufla.br/~terra/publications_files/students/2009_fumec_souza_e_monteiro.pdf)>. Acesso em: 28 out. 2019.