

MiniDMAIC: Uma Abordagem para Análise e Resolução de Causas em Projetos de Desenvolvimento de Software

Carla Ilane M. Bezerra^{1,2}, Ciro Carneiro Coelho²
, Fca. Márcia G. S. Gonçalves¹, Carlo Giovano S. Pires², Gabriela Telles², Adriano Bessa Albuquerque¹

¹Mestrado em Informática Aplicada, Universidade de Fortaleza
Av. Washington Soares 1321, 60811-341 - Fortaleza – CE, Brasil

²Instituto Atlântico, Rua Chico Lemos, 946, 60822-780 - Fortaleza – CE, Brasil
{carlailane, cgiovano, gabi}@atlantico.com.br, ccoelho@gmail.com,
marcia@hotmail.com, adriano.ba@terra.com.br.

Abstract: *Handling problems and defects in software development projects is still a difficult matter in many organizations. The problems analysis, when performed, usually do not focus on the problem sources and root causes, leading to bad decisions that do not solve the problem. It is also hard for organizations to implement a Causal Analysis and Resolution (CAR) process in software projects, as described in the CMMI level 5 because projects usually have to deal with very limited resources. This paper presents the MiniDMAIC, an approach for defects and problems causal analysis and resolution based on Six Sigma's DMAIC methodology and the Causal Analysis and Resolution process area from CMMI.*

Resumo: *O tratamento dos problemas e defeitos encontrados nos projetos de software ainda é deficiente na maioria das organizações. As análises em geral não focam suficientemente no problema e suas possíveis origens, levando a tomadas decisões erradas que acabam por não resolver o problema. Também é difícil a implementação de um processo de Análise e Resolução de Causas (CAR) em projetos, conforme prescrito pelo nível 5 do CMMI, devido aos recursos limitados com os quais os mesmos têm que trabalhar. Este trabalho propõe o MiniDMAIC, uma abordagem para análise e resolução de causas de defeitos e problemas em projetos de desenvolvimento de software com base na metodologia DMAIC do Six Sigma e na área de processo de CAR do CMMI.*

1 Introdução

A busca incessante por produtos e serviços cada vez melhores leva muitas organizações a pesquisar e implantar várias técnicas, ferramentas e normas capazes de alavancar a qualidade de seus produtos. Dentro do contexto da busca por excelência, surge como solução para diminuir custo e melhorar a qualidade a implementação de iniciativas tais como o CMMI (2006) e o *Six Sigma* [Tayntor 2003].

Um dos aspectos fundamentais da melhoria da qualidade é a análise e resolução de problemas. Para isso, pode ser utilizado um método formal de resolução de problemas, que pode trazer inúmeros benefícios, tais como [Banas Qualidade 2007]:

- Evitar que os solucionadores de problemas passem direto para a conclusão;
- Garantir a análise da causa-raiz;

- Desmistificar o processo de solução de problemas;
- Estabelecer ferramentas analíticas a usar e determinar quando utilizá-las.

Nesse contexto, a utilização de ferramentas da metodologia *Six Sigma*, como o DMAIC, tem se destacado. Ao contrário de outras metodologias para resolução de problemas, que focam apenas na eliminação do problema em si, a metodologia DMAIC [Rath and Strong 2005] empregada pelo *Six Sigma* abrange desde a seleção dos problemas que merecem um tratamento mais aprofundado até o controle dos resultados obtidos no decorrer do tempo.

A metodologia DMAIC mostra passo a passo como tais problemas devem ser tratados, agrupando as principais ferramentas da qualidade, ao mesmo tempo em que cria uma rotina padronizada na solução de problemas com uma comprovada eficiência na aplicação em organizações de software.

Apesar de adequados para o nível organizacional, os métodos formais de resolução de problemas podem ser inviáveis no nível projetos. Um dos grandes desafios enfrentados por empresas que almejam o nível 5 do CMMI é justamente a implantação da área de processo de Análise e Resolução de Causas (*Causal Analysis and Resolution – CAR*) no contexto de projetos de software, uma vez que estes geralmente possuem recursos bastante limitados. Com isso, são tomadas apenas ações imediatas para resolução dos problemas e, na maioria das vezes, os mesmos voltam a acontecer.

Apesar de todos os benefícios do uso da metodologia *Six Sigma* em conjunto com o CMMI, a implementação da área de processo Análise e Resolução de Causas em projetos de software muitas vezes torna-se inviável, pelos seguintes motivos:

- Projetos DMAIC têm duração entre 3 a 6 meses. No entanto, projetos requerem rapidez na resolução de seus problemas, não podendo esperar muito tempo;
- Devido à grande necessidade da utilização de ferramentas estatísticas, o DMAIC pode tornar-se excessivamente caro; a economia obtida pode ser menor do que o custo para alcançar melhorias, e os projetos geralmente possuem recursos limitados.
- O nível de qualificação da equipe DMAIC é bastante rigoroso; no entanto, no contexto de projetos de desenvolvimento de software, outros atributos como domínio do negócio e de gerência de projetos podem trazer maiores resultados do que o fato de ter uma equipe com grande conhecimento em estatística.

Diante deste contexto, este trabalho visa à elaboração de uma metodologia baseada no DMAIC do *Six Sigma*, denominada MiniDMAIC, para tratar a área de processo de Análise e Resolução de Causas do CMMI em projetos de desenvolvimento de software, procurando reduzir as desvantagens acima descritas de uso do DMAIC. Também visa mostrar a aplicação da metodologia em projetos de desenvolvimento de software em uma organização a partir de uma ferramenta *workflow*, onde foi implementada as práticas do MiniDMAIC.

Este trabalho está organizado em cinco seções, além desta introdução. Na seção 2, apresentamos o embasamento teórico referente ao *Six Sigma* e, mais detalhadamente, à metodologia DMAIC. Na seção 3, abordamos o CMMI e a área de processo de Análise e Resolução de Causas pertencente ao nível 5 de maturidade. Na seção 4, apresentamos a abordagem proposta, denominada MiniDMAIC. Aspectos referentes à

utilização do MiniDMAIC em projetos reais, bem como os resultados obtidos, são apresentados na seção 5. Por fim, na seção 6, apresentamos as considerações finais e limitações da metodologia proposta.

2 O *Six Sigma* e a Metodologia DMAIC

O *Six Sigma* é um método que se concentra na diminuição ou eliminação da incidência de erros, defeitos e falhas em um processo. A metodologia *Six Sigma* visa também reduzir a variabilidade do processo, podendo ser aplicada na maioria dos setores da atividade econômica [Smith 2000].

Alcançar o *Six Sigma* significa reduzir defeitos, erros e falhas¹ a zero e atingir a quase perfeição no desempenho dos processos. A metodologia associa um rigoroso enfoque estatístico a um arsenal de ferramentas que são empregadas com o objetivo de caracterizar as fontes da variabilidade para demonstrar como esse conhecimento pode controlar e aperfeiçoar os resultados do processo [Watson 2001].

A metodologia *Six Sigma* procura definir as causas óbvias e não óbvias que afetam o processo de forma a eliminá-las, ou melhorá-las e controlá-las [Rotondaro 2002].

O *Six Sigma* apresenta algumas metodologias para o tratamento de problemas e melhorias, tais como DMAIC (*Define, Measure, Analyse, Improve and Control*), DCOV (*Define, Characterize, Optimize, Verify*) e DFSS (*Design For Six Sigma*). Neste trabalho, será utilizada a metodologia DMAIC.

A metodologia DMAIC foi criada de forma mais padronizada pela *General Electric* e, de acordo com Tayntor (2003), é a mais utilizada nas empresas que implementam o *Six Sigma*, e também a mais indicada para o desenvolvimento de software.

A metodologia DMAIC é constituída por cinco fases: definir, medir, analisar, melhorar e controlar. Na fase “**definir**”, é necessário identificar o problema e então definir as oportunidades existentes para solucioná-lo de acordo com os requisitos dos clientes. Na fase “**medir**”, deve-se verificar a situação atual através de medições quantitativas do desempenho, para que as decisões posteriores sejam baseadas em fatos. Na fase “**analisar**”, deve-se determinar as causas do desempenho encontrado e analisar as oportunidades existentes. Após ter feito essa análise é possível perceber pontos para melhoria do desempenho, assim as melhorias podem ser implantadas na fase “**melhorar**”. Na fase “**controlar**”, deve-se assegurar a melhoria, através de um controle do desempenho do processo implantado.

Na Tabela 1 são descritos os objetivos das fases do modelo DMAIC de acordo com Tayntor (2003).

¹ Na metodologia *Six Sigma*, os defeitos, os erros e as falhas são qualquer desvio de uma característica que gere insatisfação ao cliente [Blauth 2003].

Tabela 1. Objetivos definidos por Tayntor (2003) para o modelo DMAIC

O Modelo DMAIC	
Fase	Objetivos
Definir	Identificar o problema e os clientes; definir e priorizar os requisitos do cliente; definir o processo atual.
Medir	Confirmar e quantificar o problema; medir as diversas etapas do processo atual; rever e esclarecer o escopo do problema, se necessário; definir os resultados desejados
Analisar	Determinar a causa raiz do problema; propor soluções
Melhorar	Priorizar soluções; desenvolver e implementar as soluções que trarão maiores benefícios
Controlar	Medir as melhorias; comunicar e comemorar o sucesso; assegurar que as melhorias no processo serão implantadas

Pande (2001) ressalta que não se pode utilizar o DMAIC para qualquer melhoria. Um projeto de melhoria *Six Sigma*, segundo o autor, deve ter três qualificações:

- Há uma lacuna entre o desempenho atual e o desempenho necessário/desejado;
- A causa do problema não é claramente compreendida;
- A solução não é predeterminada, nem é a solução ótima aparente.

Além disso, devem ser observados os critérios de viabilidade, tais como: os recursos necessários, a habilidade disponível, a complexidade, a probabilidade de sucesso e o apoio ou engajamento da equipe.

3 O CMMI e a Análise e Resolução de Causas

O *Capability Maturity Model Integration* (CMMI) [Chrissis 2006] é um modelo de maturidade de desenvolvimento de produtos desenvolvido pelo *Software Engineering Institute* (SEI), que está cada vez mais sendo adotado nas empresas, uma vez que esse modelo busca orientar as organizações na implementação de melhorias contínuas em seu processo de desenvolvimento.

3.1 O Nível 5 de Maturidade

O foco do nível 5 de maturidade está na melhoria contínua dos processos. Enquanto o nível 4 foca nas causas especiais de variação no processo da organização, o nível 5 tenta encontrar as causas comuns e tratá-las, resultando em inúmeras melhorias, que são introduzidas de forma disciplinada. Medições são usadas para selecionar as melhorias e estimar os custos e benefícios em atender as melhorias propostas. As mesmas medições podem ser usadas para justificar esforços para futuras melhorias [Kulpa 2003].

O nível 5 do CMMI é composto por duas áreas de processo: Inovação e Desenvolvimento Organizacional (*Organizational Innovation and Deployment – OID*) e

Análise e Resolução de Causas (*Causal Analysis and Resolution – CAR*), sendo esta última o foco deste trabalho.

O objetivo da PA Análise e Resolução de Causas (*Causal Analysis and Resolution – CAR*) é identificar causas de defeitos e outros problemas e tomar ações para prevenir a ocorrência deles no futuro.

A Tabela 2 apresenta o relacionamento dos objetivos específicos (SG) com suas respectivas práticas específicas (SP) para essa área de processo.

Tabela 2. Análise e Resolução de Causas no CMMI [Chrissis 2006]

SG 1	Determinar as Causas dos Defeitos	
	SP 1.1	Selecionar os dados para análise dos defeitos
	SP 1.2	Analisar as causas
SG 2	Tratar as Causas dos Defeitos	
	SP 2.1	Implementar as ações propostas
	SP 2.2	Avaliar os efeitos das mudanças
	SP 2.3	Registrar os dados

4 MiniDMAIC

O MiniDMAIC é uma estratégia que visa à simplificação do modelo DMAIC com o intuito de tratar as causas e resolução de problemas em projetos de desenvolvimento de software de forma mais prática, rápida, com menos riscos e custos, prevenindo recorrências futuras, implantando melhorias no processo de desenvolvimento e assim, aumentando cada vez mais a satisfação dos seus clientes.

As principais características do MiniDMAIC são:

- Curta duração (variando de 1 a 6 semanas);
- Requer conhecimento básico de estatística;
- Associada a riscos;
- Custo baixo;
- Específico para projetos de desenvolvimento de software.

Os problemas que merecem um tratamento mais criterioso através da aplicação da metodologia MiniDMAIC podem ser definidos a nível organizacional (Ex.: limites de controle, quantidade de defeitos, etc.) e serem refinados durante o planejamento do projeto. Deve estar bem claro para a equipe do projeto a diferença entre os problemas que requerem apenas ações simples e imediatas, e os problemas que requerem o tratamento definido na metodologia MiniDMAIC.

Exemplos de problemas de projetos que merecem o tratamento através da abordagem MiniDMAIC são:

- Projeto fora de controle, podendo ser considerados indicadores críticos para o cliente e desvios das metas organizacionais (Ex.: produtividade, desvio na entrega, densidade de defeitos, etc.)
- Problemas cuja causa raiz seja duvidosa;
- Problemas relacionados a critérios de aceitação do projeto pelo cliente.

Como apoio à metodologia, podem ser utilizadas ferramentas como planilhas eletrônicas, ferramentas de gestão de projetos, entre outras.

4.1 A Fase “Definir”

A fase *definir* consiste na definição do problema, fontes e metas, além da formação da equipe, conforme podemos ver na Tabela 3.

Tabela 3. Passos da fase “Definir”

Definir	Passo 1 - Definir o Problema	Deve ser definido o problema que será tratado para que esteja clara a sua importância e sejam definidos os seus objetivos. É importante descrever qual o impacto ou conseqüências do problema sobre o projeto. Esta descrição deve estar focando somente nos sintomas e não em causas ou soluções.
	Passo 2 - Determinar a Fonte do Problema	Esta etapa deve mostrar qual foi a fonte que mostrou a ocorrência do problema. Exemplos de fontes de problemas em projetos de desenvolvimento de software são: <ul style="list-style-type: none">• Resultados de pesquisas de satisfação com o cliente;• Reclamações do cliente;• Indicadores do projeto;• Resultados de estudos de <i>benchmarking</i>;• Resultados de avaliações da qualidade.
	Passo 3 - Definir as Metas	Nesta etapa são definidas as metas que devem ser atingidas com a execução da metodologia MiniDMAIC para tratamento do problema em questão. As metas devem ser definidas de forma quantitativa, podendo ser através de alguma medição que define o problema a ser atacado e também do cálculo do retorno de investimento (ROI – <i>return of investment</i>) planejado.
	Passo 4 - Formar a Equipe	Em uma ação MiniDMAIC não há necessidade de se ter <i>Black Belts</i> ² na liderança. Por serem simples e diretamente relacionadas ao projeto, é necessário apenas um conhecimento básico sobre <i>Six Sigma</i> e treinamento na metodologia MiniDMAIC. O mais relevante é o entendimento relacionado ao negócio do projeto e técnicas de gestão, sendo, portanto, indicado que o Gerente do Projeto lidere o MiniDMAIC. O tamanho da equipe MiniDMAIC pode variar de acordo com a necessidade do problema. Em algumas situações, podemos ter apenas o gerente do projeto e um membro da equipe contando com a participação de outras pessoas apenas em determinados passos, como apoio de um líder <i>Green Belt</i> ³ (principalmente na fase medir e analisar).

4.1 A Fase “Medir”

A fase *medir* consiste na análise das medições (existentes ou a definir) relacionadas ao problema, bem como o cálculo do nível sigma atual, conforme podemos ver na Tabela 4.

² *Black Belts*: Colaboradores que trabalham aplicando os conceitos e as ferramentas do *Six Sigma* em projetos. Sua formação depende de um treinamento intensivo em estatística e em técnicas para solução de problemas. Eles normalmente atuam identificando em diferentes processos, oportunidades a serem exploradas por meio de projetos [Harry e Schroeder, 2000].

³ *Green Belts*: Colaboradores em diferentes partes da organização que aplicam o *Six Sigma* no seu dia-a-dia. Sua formação envolve um treinamento menos intensivo que o dos *Black Belts* [Harry e Schroeder, 2000].

Tabela 4. Passos da fase “Medir”

Medir	Passo 1 - Planejar Medições	Nesta etapa deve ser analisado se existe a necessidade de uma nova medição que forneça maiores evidências para o problema em questão. Em grande parte das situações, as medições já estarão sendo conduzidas de acordo com o processo definido que atende à PA de Medição e Análise (<i>Measurement and Analysis – MA</i>).
	Passo 2 - Medir Situação Atual	As medições selecionadas no passo anterior devem ser utilizadas para medir a situação atual do projeto. Posteriormente, essas mesmas medições serão utilizadas para aferir a melhoria obtida.

4.2 A Fase “Analisar”

A fase *analisar* consiste na definição da categoria do problema, determinação de suas causas e possíveis ações para tratá-las, e, avaliação dos riscos associados às causas raiz do problema, conforme podemos ver na Tabela 5.

Tabela 5. Passos da fase “Analisar”

Analisar	Passo 1 - Determinar as Causas do Problema	Este é um dos passos mais importantes do MiniDMAIC, uma vez que seu propósito é encontrar a causa raiz do problema. Caso esta atividade não seja feita corretamente, o resultado do MiniDMAIC pode ser comprometido, pois todas as atividades seguintes terão como base o resultado deste passo. Sendo assim, é importante que participem desta etapa as pessoas que possuem conhecimento a respeito do problema e que possam contribuir com informações sobre suas causas. Exemplos de técnicas para determinar as causas do problema são: <i>brainstorming</i> , cinco porquês, diagrama de causa e efeito [Ishikawa, 1998], gráfico de pareto [Juran,1991], entre outros.
	Passo 2 – Definir Ações Candidatas	Neste passo devem ser levantadas as possíveis ações que venham a tratar o problema em questão. Este passo será complementado na próxima fase onde as ações serão priorizadas.

4.3 A Fase “Melhorar”

A fase *melhorar* consiste na priorização e aprovação das ações propostas, elaboração e execução do plano de ação, bem como o acompanhamento dos resultados obtidos, conforme podemos ver na Tabela 6.

Tabela 6. Passos da fase “Melhorar”

Melhorar	Passo 1 - Priorizar as Ações Propostas	A priorização das ações propostas deve ser feita de acordo com o processo definido para a área de Análise e Tomada de Decisão (<i>Decision Analysis and Resolution – DAR</i>), que define as diretrizes para seleção das soluções propostas. Caso seja identificada apenas uma ação para tratamento do problema, deve ser feita uma análise de viabilidade de execução da mesma.
	Passo 2 - Obter Aprovação	Deve ser realizada uma avaliação de impacto referente à implementação das ações priorizadas, resumindo seus efeitos positivos e negativos. Essas informações devem ser submetidas à aprovação da gerência sênior.
	Passo 3 - Elaborar e Executar Plano de Ação	Um plano de ação para execução das ações priorizadas e aprovadas deve ser elaborado. Este plano deve conter as seguintes informações: <ul style="list-style-type: none"> • Tarefas a serem executadas; • Responsável pela execução da tarefa; • Esforço necessário para execução da tarefa; • Prazo para conclusão da tarefa. A execução do plano de ação se dá através da implementação das suas tarefas.

Passo 4 - Acompanhar Ações	Neste passo as tarefas devem ser monitoradas para que se tenha uma visibilidade do avanço do MiniDMAIC. Estes resultados devem ser acompanhados de acordo com o processo de Monitoramento e Controle do Projeto (<i>Project Monitoring and Control – PMC</i>).
----------------------------	--

4.4 A Fase “Controlar”

A fase *controlar* consiste na avaliação dos resultados obtidos e divulgação dos resultados e lições aprendidas obtidos, conforme podemos ver na Tabela 7.

Tabela 7. Passos da fase “Controlar”

Controlar	Passo 1 - Avaliar Resultados	Na avaliação dos resultados obtidos deve ser feita uma análise se a meta estabelecida na fase “Definir” foi atingida, bem como uma análise do custo-benefício do tratamento do problema utilizando a metodologia MiniDMAIC. O custo benefício do projeto pode ser medido através do cálculo do retorno de investimento (ROI – <i>return of investment</i>).
	Passo 2 - Divulgar os Principais Resultados e Lições Aprendidas	Ao final da execução do MiniDMAIC os resultados obtidos devem ser compartilhados com toda a organização através de um repositório organizacional acessível a todos os projetos. O compartilhamento destas informações pode servir de insumo para o tratamento de problemas similares em outros projetos. A forma de divulgação deve seguir o processo definido para a PA de Foco no Processo Organizacional (<i>Organizational Processes Focus – OPF</i>), que define como serão compartilhadas as lições aprendidas pela organização. Caso sejam identificadas possíveis melhorias nos processos organizacionais, as mesmas devem ser encaminhadas para o Grupo de Engenharia de Processos (<i>Engineering Process Group – EPG</i>) para que possam ser analisadas e devidamente tratadas.

4.5 MiniDMAIC x DMAIC

A abordagem proposta tem como base os passos definidos por Tayntor (2003) para o modelo DMAIC no âmbito do desenvolvimento de software. Alguns passos foram suprimidos da metodologia e outros foram adaptados para tornar mais fácil a execução do processo nos projetos de desenvolvimento de software. Foi tomado o cuidado para que a essência do modelo não fosse alterada, conforme os objetivos que Tayntor (2003) define na Tabela 1.

A grande diferença entre o MiniDMAIC e o DMAIC é que o DMAIC através de uma análise e resolução das causas para o problema definido tem como objetivo maior a melhoria de um processo que já existe e é executado na organização. O MiniDMAIC apenas trata as causas a nível do projeto e tem como objetivo a prevenção e tratamento dos problemas definidos através da análise e resolução da(s) causa(s) raiz(es) destes problemas, não modificando necessariamente o processo organizacional.

Além disso, o DMAIC exige a comprovação estatística das causas dos problemas e das melhorias alcançadas, o que não é necessário no MiniDMAIC, que levanta e prioriza as causas por meio de ferramentas mais simples, como Diagrama de *Ishikawa* e Gráfico de Pareto, e analisa as melhorias obtidas observando o progresso dos indicadores do projeto.

4.6 MiniDMAIC x CAR

Para um melhor entendimento do relacionamento do MiniDMAIC com a PA de Análise e Resolução de Causas (CAR), foi realizado um mapeamento entre os passos do

MiniDMAIC com as práticas específicas de CAR como podemos ver na Tabela 8. Vale ressaltar que tal relacionamento não implica em estar cobrindo toda a prática, uma vez que a mesma não está atrelada apenas a projetos, mas também possui subpráticas no nível organizacional.

Tabela 8. Relacionamento dos Passos do MiniDMAIC com as Práticas Específicas de CAR

Fase	MiniDMAIC (Passos)	CAR (Práticas Específicas)	Observações
Definir	Passo 1 - Definir o Problema	-	Relacionado à PA <i>Quantitative Project Management – QPM</i>
	Passo 2 - Determinar a Fonte do Problema	-	
	Passo 3 - Definir as Metas	-	Relacionado à PA <i>Quantitative Project Management – QPM</i>
	Passo 4 - Formar a Equipe	-	Relacionado à GP 2.7 - Identificar e Envolver os <i>Stakeholders</i> Relevantes
Medir	Passo 1 – Planejar Medições	SP 1.1 - Selecionar os dados de defeitos para análise	
	Passo 2 – Medir Situação Atual	SP 1.1 - Selecionar os dados de defeitos para análise	
Analisar	Passo 1 - Determinar as Causas do Problema	SP 1.2 - Analisar as Causas	
	Passo 2 – Definir Ações Candidatas	SP 1.2 - Analisar as Causas	
Melhorar	Passo 1 - Priorizar as Ações Propostas	SP 1.2 - Analisar as Causas	
		SP 2.1 - Implementar as Ações Propostas	
	Passo 2 - Obter Aprovação	-	Relacionado à GP 2.10 - Revisar o Status com o Gerente de Alto Nível
	Passo 3 - Elaborar e Executar Plano de Ação	SP 2.1 - Implementar as Ações Propostas	
	Passo 4 - Acompanhar Ações	-	Relacionado à PA <i>Project Monitoring and Control – PMC</i>
Controlar	Passo 1 - Avaliar Resultados	SP 2.2 - Avaliar os Efeitos das Mudanças	
	Passo 2 - Divulgar os Principais Resultados e Lições Aprendidas	SP 2.3 - Registrar os Dados	

5 MiniDMAIC na Prática

A metodologia MiniDMAIC foi aplicada no contexto de projetos do Instituto Atlântico para utilização de análise de causas e defeitos dentro dos projetos de desenvolvimento de software. O Instituto Atlântico é uma instituição de pesquisa e desenvolvimento de médio porte localizada em Fortaleza-Ceará e possui nível 3 de maturidade no modelo CMMI-SW e norma ISO 9001:2000. Visando a melhoria contínua dos seus processos, o

Atlântico encontra-se em processo de implantação dos níveis 4 e 5 de maturidade do modelo CMMI com o apoio da metodologia *Six Sigma*.

Para disparar um MiniDMAIC dentro dos projetos da organização foram adotados os seguintes critérios:

- Defeitos recorrentes;
- Número elevado de defeitos em revisões técnicas (quando a causa é desconhecida);
- Número elevado de defeitos classificados como critical e blocker em testes sistêmicos (quando a causa é desconhecida);
- Número elevado de defeitos encontrados pelo cliente (quando a causa é desconhecida);
- Projeto fora dos limites de especificação das *baselines*;
- Projeto fora dos limites de controle das *baselines*.

Todos os passos do MiniDMAIC foram implementados na ferramenta Jira⁴, uma ferramenta comercial de gerenciamento de *workflow* que pode ser facilmente customizada. Segue abaixo, na Figura 1, um exemplo da tela de criação de uma ação MiniDMAIC no Jira.

The screenshot shows the 'Criar Pendência' (Create Issue) form in Jira. The form is titled 'Criar Pendência' and has a subtitle 'Passo 2 de 2: Escolha os detalhes da pendência..'. The form contains the following fields and options:

- Projeto:** Teste
- Tipo de Pendência:** MiniDMAIC
- * Resumo:** Text input field
- Descrição:** Text area
- Organizational Performance Baseline:** Text input field
- Goals:** Text area
- MiniDMAIC goals:** Text area
- Affected Process:** Dropdown menu with options: Nenhum, Análise e Projeto, Análise e Tomada de Decisão, Comercial, Compras
- * Responsável:** Dropdown menu with option: - Automático - and a 'Delegar para mim' button
- Participants:** Text input field with a user icon
- Data para Ficar Pronto:** Text input field with a calendar icon

At the bottom of the form, there are two buttons: 'Criar' and 'Cancelar'.

Figura 1. Tela do Jira implementando o MiniDMAIC

Na Figura 1 podemos observar as informações necessárias para que seja dado início a um MiniDMAIC:

⁴ Site do Jira: www.atlassian.com/software/jira/

- **Project:** este campo apresenta o nome do projeto e é preenchido automaticamente, de acordo com o projeto em questão;
- **Issue Type:** para iniciar uma ação MiniDMAIC, deve ser selecionada a opção “MiniDMAIC”;
- **Summary:** este campo deve conter o problema a ser tratado.
- **Description:** este campo deve conter uma descrição detalhada do problema, citando desde quando o mesmo existe e qual o seu impacto ou conseqüências sobre o projeto. Esta descrição deve estar focando somente nos sintomas e não em causas ou soluções;
- **Organizational Performance Baseline:** este campo deve mostrar qual é a *baseline* de desempenho organizacional relacionada ao problema que está sendo tratado. A *baseline* de desempenho organizacional servirá de parâmetro para as metas de melhoria que serão estabelecidas posteriormente;
- **Goals:** este campo deverá conter a metas e objetivos a serem atingidos com a execução da ação MiniDMAIC. Estas metas devem ser quantificáveis e viáveis;
- **Affected Process:** neste campo devem ser selecionados os processos que serão afetados com a execução da ação MiniDMAIC. Podem ser selecionados um ou mais processos;
- **Assign To:** este campo deve conter o nome do responsável que estará conduzindo a ação MiniDMAIC. Esta função geralmente será conduzida pelo Coordenador do Projeto. Caso seja julgado necessário, pode ser designado um colaborador com formação de *Green Belt* (o QA do projeto, por exemplo) para apoiar o Coordenador;
- **Participants:** este campo deve listar as pessoas que participarão da ação MiniDMAIC, além do responsável pela sua condução. Podem ser selecionadas pessoas da equipe do projeto que estejam relacionadas direta ou indiretamente no problema a ser tratado, além de pessoas de fora do projeto que detenham conhecimento e possam colaborar na condução da ação MiniDMAIC;
- **Due Date:** Este campo deve conter o prazo máximo para término da ação. Esse período geralmente varia de 1 a 6 semanas, de acordo com a complexidade do problema e o tempo necessário para se obter os resultados esperados.

Posteriormente, a ação MiniDMAIC será completada com outras informações, a medida que as fases do MiniDMAIC forem executadas na ferramenta Jira. Essas informações incluem as análises de causa realizadas, as ações de melhoria, os resultados obtidos e as lições aprendidas durante o MiniDMAIC.

Até o momento, já foram realizadas várias ações MiniDMAIC em diversos projetos de software da organização, apresentando ótimos resultados. O tempo de execução dos MiniDMAICs nos projetos levou em média de 4 a 6 semanas, executado por uma equipe com uma média de 3 a 8 pessoas. Participaram da execução dos projetos os seguintes papéis: líder *Green Belt* disponibilizado pela equipe de qualidade da

organização, coordenador do projeto, líder de requisitos, analista de teste e desenvolvedor.

As principais ferramentas que têm sido utilizadas para a análise de causas são o Diagrama de *Ishikawa* e sessões de *Brainstorming* com as equipes do projeto, para levantamento das potenciais causas do problema, Gráfico de Pareto e métodos de análise e tomada de decisão, para classificação e priorização das causas a serem atacadas e das ações a serem implementadas. A Tabela 9 descreve algumas dessas ações.

Tabela 9. Resultados dos MiniDMAICs executados em projetos de desenvolvimento de software

Problemas	Causas Priorizadas	Ações Tomadas	Resultados Obtidos
MiniDMAIC 1 - Produtividade do projeto muito baixa para atingir os limites de especificação da organização	<ul style="list-style-type: none"> - Estimativas muito otimistas - Falta de experiência em algumas tecnologias utilizadas no projeto - Problemas na comunicação interna 	<ul style="list-style-type: none"> - Detalhar mais as estimativas, para tentar avaliar melhor os impactos - Fazer a implementação de uma parte das soluções técnicas para ter uma idéia melhor das dificuldades - Medir <i>refactoring</i> em outros UCs e incluir como estimativa <i>bottom-up</i> - Incluir margem de erro nas estimativas <i>bottom-up</i> - Avaliar o impacto antes da adoção de novas tecnologias e alocar horas para auto-estudo quando uma nova tecnologia for adotada - Realizar reuniões técnicas mensais, para difundir conhecimento - Utilizar um wiki para documentar informações do projeto - Registrar pendências na ferramenta de controle de problemas - Enviar atas de discussões técnicas com o cliente para a lista do projeto - Enviar para o cliente, por email, atas com o resultado de acordos e conversas - Cobrar aprovação dos casos usos 	Produtividade melhorou 27% no projeto.
MiniDMAIC 2 - Alto número de defeitos em um dos projetos da organização	<ul style="list-style-type: none"> - Não realização de pré-teste - Paralelismo entre A&P e codificação - Testes unitários não são eficazes - Correções da revisão de código feitas após a liberação para testes 	<ul style="list-style-type: none"> - Planejar o pré-teste nos cronogramas das iterações seguintes - Incluir folga no planejamento para evitar paralelismo de A&P e codificação - Alocar toda a equipe na elaboração de testes unitários - Só autorizar liberação para testes após a correção dos problemas de Revisão Técnica de código 	Diminuição da densidade de defeitos em 96%.

<p>MiniDMAIC 3 - Baixa produtividade de engenharia do projeto</p>	<ul style="list-style-type: none"> - Complexidade do projeto, que envolve duas plataformas bem diferentes: .NET e Java - Pouca experiência da equipe na tecnologia .NET - Deficiências no Sistema Operacional atual, exigindo vários <i>refactorings</i> para fazer a aplicação executar de maneira aceitável em relação a desempenho e memória - Com a utilização de tecnologias pouco maduras, ocorreram problemas ainda não identificados, onde foi gasto parte do esforço de codificação pesquisando soluções para os problemas e para implementar soluções alternativas 	<ul style="list-style-type: none"> - Realização de treinamentos técnicos - Aumentar o acompanhamento diário do Líder Técnico - Implementar ações de motivação da equipe 	<p>Produtividade de Engenharia melhorada em 29%.</p>
---	--	--	--

6 Considerações Finais e Limitações

Por décadas, as organizações têm utilizado uma infinidade de ferramentas e metodologias diferentes para o tratamento dos problemas, todas elas com diferentes graus de eficácia. No entanto, nenhuma delas visa tratar problemas relacionados especificamente a projetos de desenvolvimento de software. A metodologia MiniDMAIC, proposta neste trabalho, tem o intuito de suprir tal necessidade.

Além disso, espera-se que a utilização do MiniDMAIC possa vir a tornar a implantação da área de processo de Análise e Resolução de Causas do CMMI menos custosa e menos complexa, tornando viável a sua adoção no âmbito dos projetos de desenvolvimento de software.

Assim, pode-se auxiliar as empresas a alcançar níveis de maturidade mais elevados, aumentando a satisfação de seus clientes e reduzindo variações de processos em sua busca pela excelência operacional.

A utilização do MiniDMAIC em projetos reais tem apresentado resultados animadores. Como principal fator para o sucesso dessas ações, temos a busca pela causa-raiz dos problemas.

A utilização de ferramentas de análise de causa, bem como de ferramentas estatísticas implementadas como parte do nível 4 do CMMI, permitem uma análise mais consistente dos problemas e permitem que suas causas reais, e não apenas seus

sintomas, sejam resolvidas. Assim, o problema é eliminado e não volta a ocorrer, trazendo resultados mais duradouros.

6.1 Limitações do Trabalho

O MiniDMAIC foi elaborado no contexto de organizações que possuem pelo menos o nível 3 de maturidade do CMMI. Isso não impede que ele seja aplicado em organizações que não tenham esse nível de maturidade. Porém, alguns passos desta metodologia podem não ter o resultado esperado, devendo ser adaptados para a realidade da organização.

Caso a organização possua o nível 4 do CMMI, as ações MiniDMAIC terão resultados ainda melhores, uma vez que diversos parâmetros para medir os resultados dos projetos já estarão definidos, bem como a utilização de ferramentas de análise estatística já será prática comum na organização.

Além disso, a sua utilização não vem a atender toda a PA de Análise e Resolução de Causas, uma vez que não é intenção do MiniDMAIC tratar os problemas organizacionais. Para estes casos, é recomendado o uso do DMAIC.

Referências

- Banas Qualidade. (2007) “Melhoria contínua - Soluções de Problemas”, Quality News. São Paulo. Disponível em <<http://www.estatbrasil.com.br/PgQtN20030003.htm>>. Acesso em: 22 fev, 2007.
- Blauth, Regis. (2003) “Seis Sigma: uma estratégia para melhorar resultados”, Revista FAE Business, nº 5.
- Chrissis, Mary B.; Konrad, Mike; Shrum, Sandy. (2006) “CMMI: Guidelines for Process Integration and Product Improvement”, 2nd edition, Boston, Addison Wesley.
- Harry D., Schroeder, R. (2000) Six Sigma: the breakthrough management strategy revolutionizing the World’s Top Corporation. New York: Doubleday.
- Ishikawa, K. (1998) Controle de Qualidade Total – À Maneira Japonesa. Rio de Janeiro, Campus, 6^a edição.
- Juran, J. M. (1991) Controle de Qualidade Handbook. J. M. Juran, Frank M. Gryna - São Paulo - Makron, McGraw-Hill.
- Kulpa, Margaret K.; Johnson, Kent A. (2003) “Interpreting the CMMI: a process improvement approach” Florida, Auerbach.
- Pande, S. (2001) “Estratégia *Six Sigma*: como a GE, a Motorola e outras grandes empresas estão aguçando seu desempenho”. Rio de Janeiro, Qualitymark.
- Rath and Strong. (2005) “*Six Sigma*/DMAIC Road Map”, 2nd edition.
- Rotondaro, G. R.; Ramos, A. W.; Ribeiro, C. O.; Miyake, D. I.; Nakano, D.; Laurindo, F. J. B.; Ho, L. L.; Carvalho, M. M.; Braz, A. A.; Balestrassi, P. P. (2002) “Seis Sigma: Estratégia Gerencial para Melhoria de Processos, Produtos e Serviços”, São Paulo, Atlas.
- Smith, B.; Adams, E. (2000) “LeanSigma: advanced quality”, Proc. 54th Annual Quality Congress of the American Society for Quality, Indianapolis, Indiana.

Tayntor, Christine B. (2003) "*Six Sigma* Software Development", Florida, Auerbach.

Watson, G. H. (2001) "Cycles of learning: observations of Jack Welch", ASQ
Publication.