



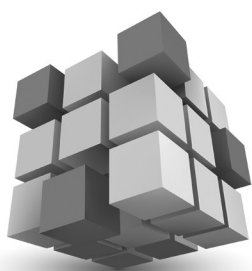
X WORKSHOP ANUAL DO MPS

# WAMPS 2014

Campinas-SP, 1 a 3 de dezembro de 2014







X WORKSHOP ANUAL DO MPS

# WAMPS 2014

Campinas-SP, 1 a 3 de dezembro de 2014



FICHA CATALOGRÁFICA ELABORADA PELO  
Sistemas de Bibliotecas da UNICAMP / Setor de Catalogação  
Bibliotecária: Priscila Gomes Cruz

W892a

Workshop Anual do MPS, 10., Campinas, SP, 2014.  
Anais do X WAMPS 2014, realizado em Campinas, de 01 a 03 de dezembro de 2014; organizadores: Marcos Kalinowski, Monalessa Perini Barcellos, Cristina Angela Filipak Machado. -- Campinas, SP : Associação para Promoção da Excelência do Software Brasileiro - Softex, 2014.  
250 p.

1. Software – Congressos. 2. Programas de computador– Congressos.

I. Kalinowski, Marcos. II. Barcellos, Monalessa Perini. III. Machado, Cristina Angela Filipak.

IV. Título.

CDD – 001.6425  
- 001.642

ISBN 978-85-99334-80-5

Índices para Catálogo Sistemático

1. Software - Congressos - 001.6425
2. Programas de computador – Congressos - 001.642

### **Softex - Associação para Promoção da Excelência do Software Brasileiro**

Criada em dezembro de 1996, a sociedade Softex, ou simplesmente Softex, é uma Organização da Sociedade Civil de Interesse Público (OSCIP) sediada em Brasília e Campinas, SP, Brasil.

#### **Missão da Softex**

Ampliar a competitividade das empresas brasileiras de software e serviços de TI e a sua participação nos mercados nacional e internacional, promovendo o desenvolvimento do Brasil.

O sistema Softex, por sua vez, tem abrangência nacional. É formado pela Sociedade Softex e por agentes regionais, aos quais se vinculam mais de 2.000 empresas com atividades em software e serviços de TI.

#### **Presidente**

Rubén Arnoldo Soto Delgado

#### **Vice-Presidente Executivo**

Ney Gilberto Leal

#### **Diretoria Executiva**

Glaucia Critter Chiliatto – Diretoria de Operações

Fabian Appel Petrait – Controller – Diretoria Softex

#### **Gerência da Qualidade**

Nelson Henrique Franco de Oliveira

Elidiane Teixeira Barroso

Cleide Gonçalves da Silva

Rafael Barreto Oliveira

Dentre as atividades da Softex no âmbito da Gerência da Qualidade, pelos resultados alcançados desde dezembro de 2003, destaca-se o Programa MPS.BR - Melhoria de Processo do Software Brasileiro.

#### **Programa MPS.BR - Melhoria de Processo do Software Brasileiro**

Kival Chaves Weber - Coordenador Executivo

# Sumário

Apresentação	11
Organização do WAMPS 2014	13
Programação do WAMPS 2014	14
<b>1 - Palestrantes convidados</b>	
1.1 – “Do we really re-use our knowledge (or not), Luigi Buglione (Université du Québec, Canada)	18
1.2 – “Governança e Gestão Estratégica de Dados – Novas Fronteiras e Maturidade”, Carlos Barbieri (FUMSOFT)	20
1.3 - “Criatividade e Inovação”, Allan Costa (FGV)	22
1.4 – “O lado H do Sucesso”, Dines Schaffer (CRAMI CAMPINAS)	24
1.5 – “O modelo MPS-RH”, Renato Luiz Della Volpe – ASR Consultoria e Assessoria em Qualidade	26
<b>2 – Artigos técnicos selecionados</b>	
2.1 - “Mapeamento para Implantação Conjunta dos Modelos MR-MPS-SW e CERTICS”, Larissa Araujo – (COPPE / UFRJ), Ana Regina Rocha – (COPPE / UFRJ), Gleison Santos – (UNIRIO)	28
2.2 – “Explorando Efeitos da Adoção do MPS-SW no Desempenho das Organizações”, Matheus Rabello – (UFJF), Victor Stroele – (UFJF)	40
2.3 – “Uma Análise da Influência do Programa MPS.BR em Artigos Técnicos Publicados no WAMPS”, Diego Cruz – (UNIRIO), Raphael Freire – (UNIRIO), Eliezer Dutra – (UNIRIO), Gleison Santos – (UNIRIO)	52
2.4 – “Uma Investigação sobre a Gerência de Projetos de Desenvolvimento de Software em Órgãos do Governo do Ceará com Base no MPS-SW”, Marum Simão Filho – (Faculdade 7 de Setembro), Rômulo Frota Santos – (UNIFOR), Adriano Albuquerque – (UNIFOR)	64
2.5 – “Mapeamento Sistemático sobre Métricas no Contexto de Métodos Ágeis aplicadas a Teste de Software”, Thaynã Mota – (UFAM), Arilo Dias Neto – (UFAM)	76
2.6 – “Processos de Apoio Gerencial Integrados ao Processo de Teste de Software”, Jeanne Trovão – (UFAM), Arilo Dias Neto – (UFAM)	88
2.7 – “Indicações de Abordagens para Rastreabilidade de Requisitos no contexto do MR-MPS-SW por meio de uma Revisão Sistemática da Literatura”, Paulo Malcher – (UFPA), Sandro Ronaldo Bezerra Oliveira – (UFPA)	100

### **3 – Artigos selecionados sobre relatos de experiência**

- 3.1 – “Experiência de Desenvolvimento e Utilização do Método de Avaliação CERTICS”, **114**  
Sonia T. Maintinguer – (Facti), Clenio F. Salviano - (CTI Renato Archer), Weslei Marinho – (Facti), Marcia Martinez – (Facti), Maysa Dias Rezende – (Facti), Adalberto N. Crespo – (Facti), Eduardo Paulo de Souza – (Facti), Davi C. Silva – (Facti), Alan Raldi – (Facti), Giancarlo N. Stefanuto – (Facti), Angela M. Alves - (CTI Renato Archer)
- 3.2 – “Relato de Experiência da Certificação do software PRIME Saúde da ECO Sistemas na CERTICS”, **122**  
Elizabeth Mocny – (ECO Sistemas), Larissa Araujo – (ECO Sistemas), Mariano Montoni – (ProMove Soluções), Analia irigoyen – (ProMove Soluções)
- 3.3 – “Relato da experiência de implementação do modelo CERTICS em uma empresa que foi avaliada de acordo com o modelo de referência MPS-SW nível G”, **130**  
Allan Moura – (Fumsoft), Breno Duarte (Visual Sistemas), Charles Alvarenga – (Fumsoft), Paulo Lana (Visual Sistemas)
- 3.4 – “Encontrando o Equilíbrio entre a Metodologia Scrum na Fábrica JAVA e o modelo MPS.br- SW - nível F”, **140**  
Nanachara Silva – (Datacooper Software), Paulo Antiquera (Datacooper Software), Edenilson Burity (SENAI Londrina)
- 3.5 – “SKAM - Um Processo usando Scrum e Kanban para Customização de Software em Dispositivos Móveis”, **150**  
John Sousa – (UFC), Alberto Monteiro – (UFC), Andrea Oliveira – (UFC), Frederico Silva – (UFC), Ismayle Santos – (UFC), Reinaldo Braga – (UFC), Rossana Andrade – (UFC),
- 3.6 – “Identificação de Fatores que Influenciam a Produtividade em Projetos de Software”, **158**  
Reinaldo Cabral – (Inform Sistemas), Rafael Morais – (Inform Sistemas)
- 3.7 – “Una Experiencia de Implementación y Evaluación Conjunta CMMI-DEV y CMMI-SVC Nivel 5 con MPS-SW Nivel A en Sofrecom Argentina”, **168**  
Valeria Chiuki – (Sofrecom Argentina), Viviana Rubinstein – (Liveware), Jorge Boria – (Liveware), Andres Rubinstein – (Liveware), Andrea Baglietto – (Sofrecom Francia), Silvia Andino – (ESCAMPI S.A.), Ana Regina Rocha – (COPPE / UFRJ)
- 3.8 – “Implantação do MR-MPS-SV na Instituição Avaliadora COPPE-UFRJ”, **178**  
Ana Regina Rocha – (COPPE/UFRJ), Gleison Santos – (UNIRIO), Cristina Cerdeiral – (COPPE/UFRJ), Marcelo Schots – (COPPE/UFRJ), Natália Chaves Lessa Schots – (COPPE/UFRJ), Taísa Gonçalves – (COPPE/UFRJ), Elaine Nunes – (COPPE/UFRJ), Tayana Conte – (UFAM), Reinaldo Cabral – (UFAL). Mylene Cabral (COPPE/UFRJ)
- 3.9 – “Avaliação do Processo de Medição em Gerência de Incidentes e Gerência de Continuidade e Disponibilidade à Luz do MR-MPS-SV”, **186**  
Bianca Trinkenreich – (UNIRIO), Gleison Santos – (UNIRIO)
- 3.10 – “Relato de Implementação da Gestão de Portfólio como Ferramenta de Execução da Estratégia”, **196**  
Karina Klein Hartmann – (SoftDesign), Osmar André Mezetti Pedroso – (SoftDesign)



3.11 – “Um Relato de Experiência da Implantação de Soluções de Rastreabilidade Bidirecional em Empresas de Salvador”, Glauco Carneiro – (UNIFACS), João Werther Filho – (Desenvolva), Cristiane Carneiro – (UNIFACS), Vinicius Santos – (Atena Tecnologia), Renato Pinto Vilela – (Medicware), Martin Noel – (Union) **206**

#### **4 – Artigos selecionados sobre ferramentas**

4.1 – “CERTICSys para avaliações de processo da CERTICS e de outros métodos baseados na Norma ISO/IEC 15504”, Alan Raldi – (Facti), Davi Silva – (Facti), Clenio Salviano – (CTI Renato Archer), Angela Alves – (CTI Renato Archer) **216**

4.2 – “Spider-QA: Uma Ferramenta de Apoio ao Processo de Garantia da Qualidade”, Iuri Raiol – (UFPA), Luiz Otávio Lima – (UFPA), Sandro Ronaldo Bezerra – (UFPA) **224**

4.3 – “MTControl: Ferramenta de Apoio à Gestão de Testes de Aplicativos Móveis Baseada nas Diretrizes do AQuA”, Juliana Nascimento – (UFAM), Jonathas dos Santos – (UFAM), Arilo Dias Neto – (UFAM) **234**

4.4 – “RAFTool - Ferramenta de Filtragem de Métodos, Classes e Pacotes com Medições Incomuns de Métricas de Software”, Tarcísio G. S. Filó – (UFMG), Mariza A. S. Bigonha – (UFMG), Kecia Aline Marques Ferreira – (CEFET-MG) **242**



# Apresentação

É com grande satisfação que, em nome do Comitê de Programa e da Comissão Organizadora, saudamos os participantes do X Workshop Anual do MPS (WAMPS 2014). O WAMPS é um evento anual, realizado pela Softex, que tem por objetivo reunir os representantes da Indústria, Governo, Academia, Equipe Técnica do Modelo (ETM), Fórum de Credenciamento e Controle (FCC), BID e países latino-americanos envolvidos e interessados no aprimoramento de processos de software por meio tanto do Modelo MPS quanto do Programa MPS.BR – Melhoria de Processo do Software Brasileiro.

Desde a edição de 2009, o Workshop Anual do MPS passou a promover, de forma integrada, o Workshop de Avaliadores MPS, o Workshop de Implementadores MPS, o Workshop de Empresas MPS.BR e o Workshop de IOGE MPS.BR.

A integração que resultou no Workshop Anual do MPS teve êxito em seu objetivo de compartilhar as experiências, aumentar a sinergia entre os grupos e usufruir da maturidade adquirida ao longo de dez anos de trabalho intenso com o Modelo de Referência para Software (MR-MPS-SW), o Modelo de Referência para Serviços (MR-MPS-SV) e o Programa MPS.BR. Este evento anual permite que os diferentes colaboradores e usuários possam compartilhar suas experiências e adquirir informações atualizadas sobre o Modelo MPS e o Programa MPS.BR. Além disso, o evento tem se consolidado como um importante fórum de discussão de trabalhos relacionados à melhoria de processos de uma forma geral. Outro ponto importante proporcionado pelo WAMPS é a constante aproximação entre Academia, Indústria e Governo, dentro da filosofia inerente ao Programa MPS.BR.

Com o apoio da SBC (Sociedade Brasileira de Computação), o WAMPS promove uma chamada para trabalhos técnicos, relatos de experiências e ferramentas envolvendo Engenharia de Software e que estejam relacionados ou sejam aplicáveis ao contexto de iniciativas de melhoria de processos de software e serviços. Neste ano foram aceitos trabalhos em três trilhas: Trabalhos Técnicos, Relatos de Experiências e Sessão de Ferramentas.

Para as trilhas de trabalhos técnicos e de relatos de experiência, são incentivadas submissões que descrevam propostas de métodos e/ou técnicas de Engenharia de Software que possam ser empregados no contexto de iniciativas de melhoria de processos de software e de serviços, em especial envolvendo os Modelos do MPS; relatos de métodos e/ou técnicas de Engenharia de Software já empregados no contexto de iniciativas de melhoria de processos de software e de serviços; relatos de experiências e lições aprendidas relacionados à utilização (implementação e/ou avaliação) dos Modelos do MPS; relatos de experiências e lições aprendidas na utilização de outros modelos de qualidade em conjunto com os Modelos do MPS; relatos de experiências e lições aprendidas na utilização (implementação e/ou avaliação) de outros modelos de qualidade que possam ser aplicáveis também aos Modelos do MPS; e estudos relatando resultados sobre melhoria de processo de software e serviços no contexto dos Modelos do MPS ou aplicáveis a esse contexto.

No WAMPS 2014 serão apresentados sete artigos técnicos e onze relatos de experiência, selecionados por um Comitê de Programa composto por pesquisadores e profissionais especialistas no Modelo MPS.

Desde 2010, o WAMPS tem propiciado, por meio da Sessão de Ferramentas, discussões de experiências e soluções automatizadas para apoiar a implementação e avaliação do MR-MPS em suas mais diversas necessidades e manifestações. Participam da Sessão de Ferramentas empresas que desenvolveram e/ou adaptaram ferramentas para apoio aos processos do MPS, bem como membros da comunidade acadêmica que realizaram pesquisas envolvendo o desenvolvimento ou utilização de ferramentas de apoio à implementação e/ou avaliação MPS. Este ano serão apresentados quatro artigos sobre ferramentas.

Além de seções técnicas para apresentação dos artigos, a programação do WAMPS 2014 conta com três minicursos técnicos, ministrados por especialistas nos temas:

- “Implementação dos Processos de Engenharia (MPS-SW nível D) com Métodos Ágeis”, ministrado por Analia Irigoyen (Promove Soluções)
- “Implementação Multimodelo de Software e Serviços (MPS-SW, MPS-SV, MPT e CERTICS)”, ministrado por Ana Regina Rocha (COPPE/UFRJ) e Larissa Araújo (ECO Sistemas)
- “Design Thinking”, ministrado por Verena Petitinga

Neste ano, em sua décima edição, o WAMPS conta com a presença de quatro palestras nacionais: “Governança e Gestão Estratégica de Dados - Novas Fronteiras e Maturidade”, apresentada por Carlos Barbieri (FUMSOFT); “Criatividade e Inovação”, ministrada por Allan Costa (CBN); “O Lado H do Sucesso”, ministrada por Dines Schäffer (CRAMI CAMPINAS); e “O Modelo MPS-RH”, ministrada por Renato Luiz Della Volpe (ASR).

Adicionalmente, como convidado internacional, o WAMPS 2014 traz o professor, pesquisador e consultor Luigi Buglione (Engineering Ingegneria Informatica SpA – Itália & Université du Québec - Canada), com a palestra “Do we really re-use our knowledge (or not)?”.

A programação do WAMPS 2014 conta, também, com dois painéis para discutir alguns aspectos específicos sobre o Modelo MPS e o Programa MPS.BR:

- “Resultados da Melhoria de Processos de Software no Brasil com base no modelo MPS-SW (Software)”, tendo como painelistas Marcello Thiry (Univale), Guilherme H. Travassos (COPPE/UFRJ) e Virgínia Duarte (Softex) e como moderador Kival C. Weber (Softex).
- “Experiências de Avaliações MPS.BR fora do Brasil”, tendo como painelistas Ana Regina Rocha (COPPE/UFRJ) e Cristina Filipak Machado (QualityFocus & CELEPAR) e como moderador Marcos Kalinowski (UFF).

Durante o período do workshop, são oferecidos cursos oficiais e são realizadas a reunião do CGP Conselho de Gestão do Programa MPS.BR e a reunião de Coordenadores de II, IA e IOGE.

Gostaríamos de agradecer a todos que contribuíram para a realização do WAMPS 2014. Somos imensamente gratos aos palestrantes convidados, aos instrutores de minicursos, aos membros e revisores do Comitê de Programa e a todos os autores que submeteram trabalhos. Todo o nosso reconhecimento vai também para Nelson Franco e sua equipe, cujo empenho tornou o WAMPS 2014 possível.

Desejamos a todos um ótimo workshop!

Campinas, novembro de 2014

**Marcos Kalinowski (UFF)**

Coordenador Científico do WAMPS 2014 –X Workshop Anual do MPS

**Monalessa Perini Barcellos (UFES)**

Coordenador da Trilha de Relatos de Experiência do WAMPS 2014

**Rodrigo Quites Reis (UFPA)**

Coordenador da Sessão de Ferramentas do WAMPS 2014

**Nelson Franco (Softex)**

**Kival Weber (Softex/MPS.BR)**

**Cristina Filipak Machado (QualityFocus e CELEPAR)**

**Ana Regina Rocha (COPPE/UFRJ)**

Coordenação Geral do WAMPS 2014 – X Workshop Anual do MPS

# Organização do WAMPS 2014

## **Coordenação Geral**

Nelson Franco (Softex)  
Kival Weber (Softex)  
Cristina Machado (QualityFocus e CELEPAR)  
Ana Regina Rocha (COPPE/UFRJ)

## **Coordenação Científica**

Marcos Kalinowski (UFF)

## **Coordenação Científica da Trilha de Relatos de Experiência**

Monalessa Perini Barcellos (UFES)

## **Coordenação Científica da Sessão de Ferramentas**

Rodrigo Quites Reis (UFPA)

## **Comitê de Avaliação da Trilha Artigos Técnicos**

Adriano Albuquerque, UNIFOR  
Ahilton Barreto, BNDES  
Alexandre Vasconcelos, UFPE  
Ana Regina Rocha, COPPE/UFRJ  
Andrea Barreto, BNDES  
Andreia Malucelli, PUCPR  
Carlos Alberto Pietrobon, UFOP/PUC-Minas  
Clenio Salviano, CTI Renato Archer  
Fernando Brito e Abreu, ISCTE-IUL e CITI/FCT/UNL  
Glauco Carneiro, UNIFACS  
Gleison Santos, UNIRIO  
Kathia Marçal de Oliveira, Université de Valenciennes et du Hainaut-Cambrésis  
Marcello Thiry, UNIVALI  
Marcelo Hideki Yamaguti, PUCRS  
Mariano Montoni, ProMove Soluções  
Reinaldo Cabral, UFAL e Inform Sistemas  
Ricardo Falbo, UFES  
Rodrigo Santos, COPPE/UFRJ  
Rodrigo Spínola, UNIFACS  
Sandro Ronaldo Bezerra Oliveira, UFPA  
Sheila Reinehr, PUCPR  
Tayana Conte, UFAM  
Toacy Oliveira, COPPE/UFRJ

## **Revisores Externos da Trilha Artigos Técnicos**

Maria Pinto-Albuquerque, ISCTE-IUL  
Carlos Serrão, ISCTE-IUL

## **Comitê de Avaliação da Trilha Relatos de Experiência**

Adriano Albuquerque, UNIFOR  
Ahilton Barreto, BNDES  
Ana Liddy Magalhães, UFMG, FUMEC, QualityFocus  
Ana Zabeu, ASR Consultoria  
Analia Irigoyen, ProMove Soluções  
Andrea Barreto, BNDES  
Carla Lima Reis, UFPA  
Cristiane Ramos, Universidade de Brasília (UnB)  
Cristina Machado, CELEPAR, QualityFocus  
Danilo Scalet, CELEPAR  
David Zanetti, ProMove Soluções  
Francisco Vasconcellos, Estratégia TI  
Gleison Santos, UNIRIO  
Kathia Marçal de Oliveira, Université de Valenciennes et du Hainaut-Cambrésis  
Luiz Carlos M. Ribeiro Junior, UnB  
Monalessa Perini Barcellos, UFES  
Rafael Prikladnicki, FACIN/PUCRS  
Reinaldo Cabral, UFAL e Inform Sistemas  
Renato Luiz Della Volpe, ASR Consultoria  
Rosângela Míriam Mendonça, UEMG

## **Comitê de Avaliação da Trilha de Ferramentas**

Adler Diniz de Souza, UNIFEI  
Alexandre Vasconcelos, UFPE  
Ana Liddy Magalhaes, UFMG/FUMEC/QualityFocus  
Ana Paula Bacelo, PUCRS  
Ana Zabeu, ASR Consultoria  
Analia Irigoyen, ProMove Soluções  
Andreia Malucelli, PUCPR  
Clenio Salviano, CTI Renato Archer  
Cristiane Ramos, UnB  
David Zanetti, ProMove Soluções  
Fabio Campos, Senado Federal  
Francisco Vasconcellos, Estratégia TI  
Gleison Santos, UNIRIO  
Luiz Carlos Ribeiro Junior, UnB  
Odisnei Galarraga, Software Process Consultoria  
Reinaldo Cabral, COPPE/UFRJ e UFAL  
Renato Luiz Della Volpe, ASR Consultoria  
Ricardo Falbo, UFES  
Rodrigo Reis, UFPA  
Sandro Ronaldo Bezerra Oliveira, UFPA  
Sarah Kohan, Fund. Carlos Alberto Vanzolini

# Programação do WAMPS 2014

## WAMPS 2014 - X WORKSHOP ANUAL DO MPS - 1 a 3 de dezembro de 2014

**Local:** Hotel Tryp Campinas (by Sol Meliá) - Rua Severo Penteado, 140 – Cambuí, Campinas, SP  
**Coordenação Geral:** Ana Regina Rocha (COPPE/UFRJ), Kival Weber (SOFTEX), Marcos Kalinowski (UFF),  
 Monalessa Perini Barcellos (UFES), Rodrigo Quites Reis (UFPA), Cristina Filipak Machado (CELEPAR) e  
 Nelson Franco (SOFTEX)

**Coordenação Científica:** Marcos Kalinowski (UFF)

**Coordenação Científica da Trilha de Relatos de Experiência:** Monalessa Perini Barcellos (UFES)

**Coordenação Científica da Sessão de Ferramentas:** Rodrigo Quites Reis (UFPA)

X Workshop Anual do MPS (WAMPS 2014)				
2ª feira - 1 de dezembro de 2014				
7:45-8:00	Credenciamento			
8:00-8:30	Curso: "Implementação dos Processos de Engenharia (MPS-SW - nível D) com Métodos Ágeis" Instrutora: Analia Irigoyen	Reunião: Grupo de Trabalho Especial do MPS-RH		
8:30-9:00		Reunião: Coordenadores de II, IA e IOGE		
9:00-9:30				
9:30-10:00				
10:00-10:30	Intervalo			
10:30-11:00	Curso: "Implementação dos Processos de Engenharia (MPS-SW - nível D) com Métodos Ágeis" Instrutora: Analia Irigoyen	Reunião: ETM (Análise dos Planos de Trabalho)		
11:00-11:30				
11:30-12:00				
12:00-12:30				
12:30-14:00	Almoço			
14:00-14:30	Curso: "Implementação dos Processos de Engenharia (MPS-SW - nível D) com Métodos Ágeis" Instrutora: Analia Irigoyen	Curso: "Implementação Multimodelo de Software e Serviços (MPS-SW, MPS-SV, MPT e CERTICS)" Instrutoras: Ana Regina Rocha e Larissa Araújo	Curso: "Design Thinking" Instrutora: Verena Petitinga	Softex
14:30-15:00				RAP - Reunião de Acompanhamento do programa MPS.BR com Diretoria da Softex
15:00-15:30				
15:30-16:00				
16:00-16:30	Intervalo			
16:30-17:00	Curso: "Implementação dos Processos de Engenharia (MPS-SW - nível D) com Métodos Ágeis" Instrutora: Analia Irigoyen	Curso: "Implementação Multimodelo de Software e Serviços (MPS-SW, MPS-SV, MPT e CERTICS)" Instrutoras: Ana Regina Rocha e Larissa Araújo	Curso: "Design Thinking" Instrutora: Verena Petitinga	Softex
17:00-17:30				
17:30-18:00				
18:00-18:30				
18:30-19:00	Painel: "Resultados da Melhoria de Processos de Software no Brasil com base no modelo MPS-SW (Software)"			
19:00-19:30	Painelistas: Marcello Thiry, Guilherme Travassos e Virgínia Duarte Moderador: Kival Weber			
19:30-20:00	Entrega de Placas MPS.BR			
20:00-21:00	Abertura Oficial			
21:00-23:00	Coquetel			

<b>3ª feira - 2 de dezembro de 2014</b>		
8:00-8:30	<p align="center"><b>Sessão Técnica I - CERTICS e Ferramentas:</b></p> <p align="center"><b>8:00 às 8:25</b> - Relato de Experiência Experiência de Desenvolvimento e Utilização do Método de Avaliação CERTICS. Autores: Clenio Salviano, Sônia Maintinguer, Angela Alves (CTI Renato Archer) e Wesley Marinho (Facti).</p> <p align="center"><b>8:25 às 8:45</b> - Ferramenta CERTICSys para Avaliação de Processo da CERTICS e de outros Métodos Baseados na Norma ISO/IEC 15504. Autores: Alan Raldi, Davi Silva (Facti), Clenio Salviano e Angela Alves (CTI Renato Archer).</p> <p align="center"><b>8:45 às 9:10</b> - Relato de Experiência Relato de Experiência da Certificação do Software PRIME Saúde da ECO Sistemas no CERTICS. Autores: Elizabeth Mocny (ECO Sistemas), Larissa Araujo (ECO Sistemas), Mariano Montoni e Analia Irigoyen (ProMove Soluções).</p> <p align="center"><b>9:10 às 9:35</b> - Relato de Experiência Um Relato de Experiência da Implantação e Uso da Rastreabilidade Bidirecional em Projetos de Software em Empresas de Salvador. Autores: Glauco Carneiro, Cristiane Carneiro (Universidade Salvador), João Werther Filho (Desenvolva), Vinicius Santos (Atena), Renato Vilela (Medicware) e Martin Noel (Union).</p> <p align="center"><b>9:35 às 9:50</b> - Relato de Experiência Relato da Experiência de Implementação do Modelo CERTICS em uma Empresa que foi Avaliada de Acordo com o Modelo de Referência MPS-SW Nível G. Autores: Charles Alvarenga (UFMG) e Allan Moura (Fumsoft).</p>	<b>Softex</b>
8:30-9:00		<p align="center"><b>Reunião: CGP</b> - Conselho de Gestão do Programa MPS.BR</p>
9:00-9:30		
9:30-10:00		
10:00-10:30	<b>Intervalo</b>	
10:30-11:00	<p align="center"><b>Sessão Técnica II - Métodos Ágeis, Produtividade e Ferramentas:</b></p> <p align="center"><b>10:30 às 10:55</b> - Relato de Experiência Encontrando o Equilíbrio entre a Metodologia Scrum na Fábrica JAVA e o Modelo MPS-SW Nível F. Autor: Nanachara Silva (Datacooper Software)</p> <p align="center"><b>10:55 às 11:15</b> - Ferramenta RAFTool: Ferramenta de Filtragem de Métodos, Classes e Pacotes com Medições Incomuns de Métricas de Software. Autores: Tarcísio Filó, Mariza Bigonha (UFMG) e Kecia Ferreira (CEFET-MG).</p> <p align="center"><b>11:15 às 11:40</b> - Relato de Experiência SKAM: Um Processo usando Scrum e Kanban para Customização de Software em Dispositivos Móveis. Autores: Jonh Sousa, Alberto Hugo, Andrea Oliveira, Ismayle Santos, Reinaldo Braga, Rossana Andrade e Frederico Silva (UFC).</p> <p align="center"><b>11:40 às 12:00</b> - Ferramenta MTControl: Ferramenta de Apoio à Gestão da Qualidade de Apicativos Móveis Baseada nas Diretrizes da Aliança AQUA. Autores: Juliana Nascimento, Jonathas dos Santos e Arilo Dias Neto (UFAM).</p> <p align="center"><b>12:00 às 12:25</b> - Relato de Experiência Identificação de Fatores que Influenciam a Produtividade em Projetos de Software. Autores: Reinaldo Cabral e Rafael Morais (UFAL).</p>	<p align="center"><b>Reunião: CGP</b> - Conselho de Gestão do Programa MPS.BR</p>
11:00-11:30		
11:30-12:00		
12:00-12:30		
12:30-14:00	<b>Almoço</b>	

<b>3ª feira - 2 de dezembro de 2014</b>	
14:00-14:30	<p align="center"><b>Palestra Keynote Internacional:</b>  <b>"Do we really re-use our knowledge (or not)?"</b>  <b>Palestrante: Luigi Buglione</b></p>
14:30-15:00	
15:00-15:30	
15:30-16:00	<p align="center"><b>Sessão Técnica III - Implementação e Avaliação Multimodelo:</b></p> <p align="center"><b>15:30 às 16:00 - Artigo Técnico</b>  Mapeamento para Implantação Conjunta dos Modelos MR-MPS-SW e CERTICS.  Autores: Larissa Araujo, Ana Regina Rocha (COPPE/UFRJ) e Gleison Santos (UNIRIO).</p> <p align="center"><b>16:00 às 16:25 - Relato de Experiência</b>  Una Experiencia de Implementación y Evaluación Conjunta CMMI-DEV y CMMI-SVC Nivel 5 con MPS-SW Nivel A en Sofrecom Argentina.  Autores: Valeria Chiuki, Andrea Baglietto (Sofrecom), Viviana Rubinstein, Jorge Boria, Andres Rubinstein (Liveware), Silvia Andino (ESCAMPI S.A.) e Ana Regina Rocha (COPPE/UFRJ).</p>
16:00-16:30	
16:30-17:00	<b>Intervalo</b>
17:00-17:30	<p align="center"><b>Palestra: "Governança e Gestão Estratégica de Dados - Novas Fronteiras e Maturidade"</b>  <b>Palestrante: Carlos Barbieri</b></p>
17:30-18:00	
18:00-18:30	<p align="center"><b>Sessão Técnica IV - MPS para Serviços:</b></p> <p align="center"><b>18:00 às 18:25 - Relato de Experiência</b>  Implantação do MR-MPS-SV na Instituição Avaliadora COPPE/UFRJ.  Autores: Ana Regina Rocha, Cristina Cerderal, Natália Chaves Lessa Schots, Taísa Gonçalves (COPPE/UFRJ), Gleison Santos (UNIRIO), Tayana Conte (UFAM) e Reinaldo Cabral (UFAL).</p> <p align="center"><b>18:25 às 18:50 - Relato de Experiência</b>  Avaliação do Processo de Medição em Gerência de Incidentes de Continuidade e Disponibilidade à Luz da Medição do MR-MPS-SV.  Autores: Bianca Trinkenreich e Gleison Santos (UNIRIO).</p>
18:30-19:00	



<b>4ª feira - 3 de dezembro de 2014</b>	
8:00-8:30	<b>Sessão Técnica V - Análises de Influência, Gêneria de Projetos e Gerência de Portfólio:</b> <b>8:00 às 8:30</b> - Artigo Técnico Explorando Efeitos da Adoção do MPS-SW no Desempenho das Organizações. Autores: Matheus Rabello e Victor Stroele (UFJF). <b>8:30 às 9:00</b> - Artigo Técnico Uma Análise da Influência do Programa MPS.BR em Artigos Técnicos Publicados no WAMPS Autores: Diego Cruz, Raphael Freire, Eliezer Dutra e Gleison Santos (UNIRIO). <b>9:00 às 9:30</b> - Artigo Técnico Um Estudo sobre a Gerência de projetos de Desenvolvimento de Software e Órgãos do Governo Estadual com Base no MPS.BR. Autores: Marum Simão Filho (Faculdade 7 de setembro), Rômulo Frota Santos e Adriano Albuquerque (UNIFOR). <b>9:30 às 9:55</b> - Relato de Experiência Relato de Implementação da Gestão de Portfólio como Ferramenta de Execução da Estratégia. Autores: Karina Klein Hartmann e Osmar André Mezetti Pedrozo (SoftDesign).
8:30-9:00	
9:00-9:30	
9:30-10:00	
10:00-10:30	<b>Intervalo</b>
10:30-11:00	<b>Palestra: "Criatividade e Inovação"</b> <b>Palestrante: Allan Costa</b>
11:00-11:30	
11:30-12:00	<b>Sessão Técnica VI - Teste de Software:</b> <b>11:30 às 12:00</b> - Artigo Técnico Mapeamento Sistemático sobre Métricas no Contexto de Métodos Ágeis Aplicadas à Teste de Software. Autores: Thaynã Mota e Arilo Dias Neto (UFAM). <b>12:00 às 12:30</b> - Artigo Técnico Processos de Apoio Gerencial Integrados ao Processo de Teste de Software. Autores: Jeanne Trovão e Arilo Dias Neto (UFAM).
12:00-12:30	
12:30-14:00	<b>Almoço</b>
14:00-14:30	<b>Palestra: "O lado H do sucesso"</b> <b>Palestrante: Dines Schäffer</b>
14:30-15:00	
15:00-15:30	
15:30-16:00	<b>Painel: Experiências de Avaliações MPS.BR fora do Brasil</b> <b>Painelistas: Ana Regina Rocha e Cristina Filipak Machado</b>
16:00-16:30	
16:30-17:00	<b>Intervalo</b>
17:00-17:30	<b>Palestra: "O modelo MPS-RH"</b> <b>Palestrante: Renato Luiz Della Volpe</b>
17:30-18:00	
18:00-18:30	<b>Sessão Técnica VII - Rastreabilidade:</b> <b>18:00 às 18:30</b> - Artigo Técnico Indicações de Abordagens para Rastreabilidade de Requisitos no contexto do MR-MPS-SW por meio de uma Revisão Sistemática da Literatura. Autores: Paulo Malcher e Sandro Oliveira (UFPA). <b>18:30 às 18:50</b> - Ferramenta Spider-QA: Uma Ferramenta de Apoio ao Processo de Garantia da Qualidade. Autores: Iuri Raiol, Luiz Otávio Lima e Sandro Oliveira (UFPA).
18:30-19:00	
19:00-19:15	<b>Encerramento</b>
19:15-20:00	<b>Reunião: Comissão Organizadora WAMPS 2014 e Planejamento Geral WAMPS 2015</b>

# Do we really re-use our knowledge (or not)?

Luigi Buglione

ETS Montréal / Engineering Ingegneria Informatica SpA

Via Riccardo Morandi, 32 – 00148 Rome - ITALY

Luigi.buglione@eng.it; luigi.buglione@computer.org

## ***Abstract***

Looking back to the last 25 years in IT, one of the most used, practical IT ‘inventions’ has been the ‘cut & paste’ mechanism, that’s a reuse of a previous artefact. But looking to ICT organizations in these turbulent years – no matter if they deal with software and/ services – many of them seem to do not be designed for resilience. From a root-cause analysis (RCA), one of the main ‘bones’ for improvement is the lack of reuse of organizations’ experience, in terms of historical data, artefacts, processes, etc.

Thus, re-using the ‘internal knowledge’ (before thinking to software reuse) is – as many Process Improvement models like MPS.BR (both the SW and the SV constellations) affirm – one of the real ‘wheels’ driving an organization to achieve success, measured not only by ROI but also by VOI (Value on Investment), because the more and more relevance in *understanding* the role and value of *intangibles* in our business.

The keynote presentation will move from evidences and well-known industrial stories for discussing and understanding where knowledge has (or not) been the key driver for organizational success, which possible barriers to that and how multi-model approaches as LEGO (Living EnGineering prOcess) could help an organization in achieving such goals in an easier way, betting mostly on people knowledge, as the ‘building perspective’ also looking to the Balanced Scorecard (BSC) schema and logical flow.

## ***Keywords***

Process Improvement; MPS.BR; Knowledge Management; Maturity & Capability Models; LEGO

## Luigi Buglione

Is a Measurement & Process Improvement Specialist at Engineering Ingegneria Informatica SpA (formerly Atos Origin Italy and SchlumbergerSema) in Rome, Italy and Associate Professor at the École de Technologie Supérieure (ETS) – Université du Québec, Canada. Previously, he worked as a Software Process Engineer at the European Software Institute (ESI) in Bilbao, Spain. Dr. Buglione is a regular speaker at international Conferences on Software/Service Measurement, Process Improvement and Quality, and is actively part of several International (ISO WG 10-25-40, IFPUG, COSMIC, ISBSG, MAIN) and National (GUFPI-ISMA, itSMF Italy, AICQ, AutomotiveSPIN Italy) technical associations on such issues. He developed and was part of ESPRIT and of Basque Government projects on metric programs, EFQM models, the Balanced IT Scorecard and QFD for software and is a reviewer of the SWEBOK project (2004 and 2010 editions). Several collaborations with universities for joint research programs and studies. He received a Ph.D in Management Information Systems from LUISS Guido Carli University (Rome, Italy) and a degree cum laude in Economics from the University of Rome “La Sapienza”, Italy. Further information on SEMQ ([www.semq.eu](http://www.semq.eu)). He can be reached at [luigi.buglione@eng.it](mailto:luigi.buglione@eng.it) or [luigi.buglione@computer.org](mailto:luigi.buglione@computer.org).

# Governança e Gestão Estratégica de dados – Novas Fronteiras e Maturidade

Carlos Barbieri

FUMSOFT

A palestra objetiva mostrar os aspectos de valoração crescente dos dados, dentro das empresas, saindo do status de componente de sistemas, para o de ativos organizacionais. Com a crescente produção de dados (Big Data) e o maior rigor das agências reguladoras sobre empresas, aliado à necessidade de tomada de decisão, o tema “dado” cresce de importância e liga um sinal para a necessidade de maior controle sobre esses recursos. Organizações consagradas como a DAMA(Data Management Association) e agora o CMMI Institute já oferecem um conjunto de melhores práticas visando uma abordagem de controle e qualidade dos dados. O DMM-Data Management Maturity Model, modelo do CMMI Institute, para aferição de maturidade no tratamento e na gestão dos dados já chegou ao Brasil e começa a ser aplicado nas empresas, que já despertaram para o valor desse recurso organizacional, dentro dos seus ativos.

## Carlos Barbieri

Engenheiro formado em 1970, MSc pelo INPE-São José dos Campos, Trabalhou por 30 anos na Cemig-Cia Energética de MG, atuando na área de Tecnologia e Dados. Responsável pelo Projeto Bug do Milênio da empresa. Consultor de Eng de Software(MPS.BR) sendo coordenador de implantação do modelo em quase 100 empresas em MG. Autor de 3 livros na área de Dados: Modelagem de Dados(IBPI-Press-1994), BI-Modelagem e Tecnologia(AxcelBooks-2001) e BI2-Modelagem e Qualidade(Elsevier-2011). Gerente da área de Qualidade da Fumsoft-Sociedade Mineira de Software.Um dos dois primeiros certificados CDMP-DMBOK do Brasil e o primeiro certificado CBIP-DMBOK. Foi revisor do modelo DMM-Data Maturity Model do CMMI, convidado pelo CMMI Institute.



# CRIATIVIDADE E INOVAÇÃO

Allan Costa

O mantra das empresas no mundo dos negócios atual é Inovação. Entretanto, embora na teoria muito se fale sobre o tema, obter resultados na prática é uma tarefa árdua e que exige um conjunto de competências e atitudes fundamentalmente diferentes. De forma genérica, o caminho para gerar inovação passa obrigatoriamente por algumas etapas. Em primeiro lugar, o mundo corporativo precisa compreender que inovação é um atributo organizacional que depende de um outro atributo, este individual, para que possa acontecer. Este atributo chama-se criatividade. A única forma de desenvolver organizações inovadoras é contar com líderes e equipes criativos. Em segundo lugar, é preciso alterar o modelo mental que orienta o raciocínio e a ação dos gestores. O modelo mental determina a visão e a percepção que um indivíduo tem de si mesmo e do mundo que o cerca, e dessa forma, condiciona, obrigatoriamente, todas as suas ações e atitudes. Um discurso voltado inovação orientado por um modelo ultrapassado cuja prática não reflita o discurso, gera descrédito e emperra a organização. Para que a mudança e evolução do modelo mental ocorram, é necessário que os indivíduos tragam para a consciência aspectos normalmente despercebidos acerca da forma como agem no cotidiano. Adicionalmente, é preciso ampliar a compreensão sobre alterações radicais em curso na forma das empresas se organizarem e fazerem negócios e encorpar o conjunto de referências a partir de experiências simples e bem sucedidas de empresas que obtiveram sucesso destacado no processo de criar inovação. E por último, uma organização que se pretende inovadora necessita criar um ambiente em que a inovação possa florescer e em que os times e indivíduos tenham espaço e os estímulos adequados para correr os riscos inerentes a qualquer processo inovador. Isso se faz com a incorporação de prática simples mas que geram resultados exponenciais, sempre tendo como premissa desenvolver o protagonismo dos indivíduos, times e líderes das organizações que têm como meta se tornarem mais inovadoras.

## Allan Costa

Allan Costa é AMP pela Harvard Business School. Mestre em Gestão pela FGV; MSc in IT, Management and Organizational Change pela Lancaster University (UK); MBA pelo IBMEC. É palestrante, empreendedor serial, consultor e mentor de startups; co-fundador da rede de investidores-anjo Curitiba Angels e investidor-anjo membro do Harvard Business School Startup Angels. Foi Diretor Superintendente e Diretor Técnico do SEBRAE/PR, Diretor Geral e Secretário de Estado da Secretaria de Planejamento do Paraná. É colunista de empreendedorismo da rádio CBN.



## O LADO “H” DO SUCESSO!

Dines Schäffer

CRAMI CAMPINAS

Qual é o grande patrimônio de uma empresa?

Aposto que a primeira coisa que lhe veio à cabeça foi o capital humano! E minha aposta se baseia no fato de que realmente ele é importantíssimo e porque tem um “H” no título acima.

Mas será essa a resposta?

Será que apenas com o estoque do “conhecimento” existente nos cérebros de sua empresa fica garantido o sucesso de suas atividades?

Quanto vale este conhecimento?

Uma boa formação ... especializações .... títulos de mestrado, doutorado, “pós doctor” ... são muito importantes.

Quanto vale tudo isso?

E isto, por si só, é suficiente?

Há alguns anos, o INTEL, convidou todos os ex-alunos que por lá se graduaram para avaliar as dificuldades que eles encontraram no mercado de trabalho e, assim identificar, o que realmente a escola deixou de “ensinar” e como se poderia melhorar para as turmas futuras.

As respostas convergiram para uma grande carência.

Provavelmente ela (esta carência) continue a afetar o desempenho dos mais privilegiados cérebros das diversas organizações de nossos dias.

E aí todo o valioso cabedal de conhecimentos do capital humano de uma empresa, caro e precioso, não serve para absolutamente nada ou serve apenas para resultados insignificantes e pífios.

A proposta de pensar sobre o assunto e entender o lado prático da aplicação de todo este conhecimento e de trabalhar mais o “como” do que o “o que” é em suma a proposta da palestra cujo título acima, tem o objetivo de “fazer pensar” ...

Agregar resultados de forma prática e simples é o grande objetivo e que está ao alcance de todos aqueles que, sem preconceitos se entregarem na lógica dos que experimentaram e ainda experimentam o doce sabor do SUCESSO!

O resto?

Na hora “H” eu conto...



## **Dines Schäffer**

Administrador de Empresas pela PUC Campinas, foi Diretor Administrativo e Financeiro de empresas nacionais e multinacionais. Autor de curso de Matemática Financeira, é palestrante com formação em PNL. Foi presidente da ADAF - Associação dos Dirigentes Administrativos e Financeiros de Campinas e Região, Conselheiro do CIESP e Diretor Técnico do IBEF-Campinas. Atuou também como Gerente de Vendas. Atualmente é Presidente Voluntário do CRAMI CAMPINAS. Recebeu o Prêmio "Herbet de Souza - o Betinho" outorgado pela Câmara Municipal de Campinas. É membro da Igreja Luterana em Campinas, onde também já foi Presidente.

# O Modelo MPS para a Gestão de Pessoas – MPS-RH

Renato Luiz Della Volpe

ASR Consultoria

O MPS.BR tem como principal objetivo o de definir e aprimorar modelo de melhoria e avaliação de processo de software e de gestão de serviços, visando preferencialmente as micro, pequenas e médias empresas, atender à necessidade de implantar os princípios de engenharia de software e gestão de serviços de forma adequada ao contexto das empresas, estando em consonância com as principais abordagens internacionais para definição, avaliação e melhoria de processos de software e serviços.

Dentro deste contexto e com o enfoque nos componentes de um processo que requer que seja constituído por um conjunto de atividades, métodos, práticas e transformações que as pessoas utilizam para desenvolver e manter software e seus produtos relacionados, o fator pessoas se torna uma das três dimensões críticas do foco das organizações (procedimentos e métodos / infraestrutura / pessoas).

Alinhado a isto, o cenário do atual mercado de tecnologia apresenta uma necessidade de evolução da capacitação e maior comprometimento das pessoas para, não só melhorar e evoluir os resultados das organizações, como também dos próprios profissionais envolvidos.

Neste sentido, o MPS-RH voltado a Gestão de Pessoas possui como objetivo principal oferecer às empresas orientações para a implementação gradativa de práticas de gestão de RH na indústria de TIC, de forma a selecionar, desenvolver e reter talentos humanos necessários ao atendimento dos objetivos organizacionais.

Estes resultados esperados do MPS-RH possuem como meios de sua obtenção:

- prover às pessoas oportunidade de desenvolver seu potencial e carreira;
- gerar motivações das pessoas para melhorar seu desempenho e alinhar este aos objetivos da organização;
- alinhar melhoria e evolução de processos técnicos juntamente com processos de gestão de pessoas e
- estabelecer processos voltados a pessoas que evoluam como os demais processos organizacionais.

## **Renato Luiz Della Volpe**

Sócio Diretor da ASR Consultoria e Assessoria em Qualidade Ltda.

Presidente do SPIN=SP (*Software Process Improvement Network*)

Formado em 1983 em Eng. Mecânica pela FEI e Pós-graduação em Administração pela USP – 2001. Atuação com modelos de gestão como examinador do PNQ em 1997, 1999 e 2001, implementação e avaliação do SW-CMM/CMMI-DEV nos métodos CBA/SCE e SCAMPI pelo Software Engineering Institute. Experiência de 30 anos em engenharia de produção e gestão da qualidade - implantação de SGQ - ISO 9000; Métodos de pesquisa de satisfação de clientes e de avaliação de fornecedores.

Recentemente atuou como coordenador no projeto RELAIS da implementação do modelo MoProSoft em grupo de empresas e é integrante da Equipe Técnica do Modelo – ETM MPS.BR – MPS RH.

# Mapeamento para Implantação Conjunta dos Modelos MR-MPS-SW e CERTICS

Larissa Lopes de Araujo<sup>1,2</sup>, Ana Regina Rocha<sup>1</sup>, Gleison Santos<sup>3</sup>

<sup>1</sup> COPPE/UFRJ – Universidade Federal do Rio de Janeiro – Caixa Postal 68511 – CEP 21.941-972 – Rio de Janeiro, RJ

<sup>2</sup> ECO Sistemas - Rua Dom Gerardo, nº 35 - 9º e 10º andar - Centro CEP 20.090-030 - Rio de Janeiro – RJ

<sup>3</sup> Programa de Pós-Graduação em Informática – Universidade Federal do Estado do Rio de Janeiro (UNIRIO) - Av. Pasteur 458, Urca, CEP 22.290-240 – Rio de Janeiro, RJ darocha@cos.ufrj.br, larissa.araujo@ecosistemas.com.br, gleison.santos@uniriotec.br

**Abstract.** *The Information Technology industry growth in Brazil has stimulated organizations to invest in process improvement to achieve higher quality in their products and also to obtain certifications required in public and international competitions. There is a multitude of models that make it difficult to choose which models to implement and how to implement them together when needed. A great knowledge on the models and their intersections reduces processes redundancy and also effort and costs. This paper presents a mapping of MR-MPS-SW and CERTICS which allows organizations to harmonize their implementation.*

**Resumo.** *O crescimento da indústria de Tecnologia da Informação no Brasil vem impulsionando o investimento das organizações em melhoria de processo, com o propósito de alcançar maior qualidade em seus produtos, além de obter as qualificações requeridas para concorrências públicas. A multiplicidade de modelos existentes torna difícil a escolha de quais implementar e de como implementá-los em conjunto quando necessário. A utilização de multimodelos exige grande conhecimento destes modelos e suas interseções para que a implementação conjunta não gere processos redundantes, bem como esforços e custos desnecessários. Este artigo apresenta um mapeamento entre os modelos MR-MPS-SW e CERTICS visando apoiar as organizações que queiram implementá-los em conjunto.*

## 1. Introdução

Organizações realizam investimentos em melhoria de processos, com o propósito de alcançar maior qualidade em seus produtos, além de obter as qualificações requeridas em concorrências públicas e internacionais. Modelos como CMMI (SEI, 2010), ITIL (itSMF UK, 2011), ISO/IEC 15504 (ISO/IEC 2003) e Six-Sigma (TENNANT, 2001) estão entre as técnicas ou modelos de processos mais utilizados (PARDO *et al.*, 2011). Com mais de 10 anos de existência e com 583 avaliações realizadas até setembro de 2014, o MR-MPS-SW (SOFTEX, 2012a) é hoje um modelo consolidado no Brasil e iniciando sua difusão na América Latina, especialmente na Colômbia, Argentina, Uruguai, México e Peru. O CERTICS (CTI Renato Archer, 2013), modelo de certificação de software publicado em 2013, cujo objetivo é comprovar se um software é resultado de desenvolvimento e inovação tecnológica no Brasil, vem despertando interesse nas organizações, muitas vezes envolvendo uma implementação conjunta com o MR-MPS-SW. Entretanto, nenhum modelo possui uma solução completa para o gerenciamento dos processos no contexto da Tecnologia da Informação e a multiplicidade de modelos

torna difícil a escolha de que modelos implementar e de como implementá-los em conjunto quando isso é necessário. A utilização de ambientes multimodelos exige conhecimento destes modelos e suas interseções para que a implementação conjunta não gere processos redundantes, bem como esforços e custos desnecessários.

Considerando a dificuldade da implantação de múltiplos modelos e o interesse pelos modelos MR-MPS-SW e CERTICS, este trabalho teve como objetivo realizar o mapeamento destes modelos, buscando identificar o grau de cobertura do MR-MPS-SW, considerando seus diversos níveis, com relação aos resultados esperados pelas áreas de competência do CERTICS.

Além desta seção introdutória, este artigo está organizado em outras seis seções. Na seção 2, são apresentadas brevemente as principais questões relacionadas à melhoria de processos multimodelos. A seção 3 descreve os modelos MR-MPS-SW e CERTICS. A seção 4 apresenta a metodologia de pesquisa utilizada para a elaboração do trabalho. O mapeamento realizado e seus resultados são descritos nas seções 5 e 6. E, por fim, na Seção 7 são apresentadas as considerações finais.

## 2. Melhoria de Processos Multimodelos

As necessidades de negócio, a diversidade de modelos e as exigências de mercado conduzem as Organizações para a utilização de multimodelos em suas iniciativas de melhoria de processos. Entretanto, o uso simultâneo de múltiplas abordagens pode trazer problemas, entre os quais a identificação e manipulação de diferentes abordagens com diferentes estruturas, granularidade, terminologia, conteúdo, tamanho e complexidade (KELEMEN, 2013), (PARDO *et al.*, 2012). Estes problemas podem gerar esforço e custo extras, além de aumentar o risco de ineficiências e redundâncias, segundo BALDASSARRE *et al.* (2010). Como possível solução para estes problemas devem-se realizar esforços de harmonização, para identificar interseções e partes comuns, criando uma solução de melhoria multimodelos.

PARDO (2011) analisou o estado da arte de iniciativas de harmonização de múltiplos modelos e concluiu que o estado da arte da harmonização de multimodelos é dividido em dois seguimentos, estudos (i) que harmonizam dois ou mais modelos através de mapeamentos e (ii) que propõe a sua integração com o uso de frameworks e ontologias.

Em relação à integração de modelos, FERREIRA *et al.* (2011) apresentam um modelo conceitual para o gerenciamento das informações para apoiar nas auditorias e avaliações multimodelos. Segundo estes autores um primeiro modelo deve ser implementado. O segundo deve observar as questões para completar a implantação com o primeiro, pois harmonizar significa encontrar similaridades e sinergias que facilitem a implantação conjunta.

No que se refere ao mapeamento de dois ou mais modelos, vários trabalhos vem sendo realizados nesta direção. CATER-STEEL (2006) harmoniza CobiT, ITIL, ISO 9000 e CMMI. BALDASSARRE *et al.* (2010; 2011) harmoniza CMMI com ISO 9001. O CMMI v1.2 é harmonizado com ISO/IEC 15504-2 por diversos autores como PINO (2009) e PELDZIUS e RAGAISIS (2011). MELLO (2011) harmoniza os modelos MPS-SW e CMMI-DEV. A Softex publicou, ainda, outros dois mapeamentos, do MR-MPS-SW em relação à NBR ISO/IEC 29110-4-1:2012 (SOFTEX, 2012b) e do MR-MPS-SW em relação ao modelo MoProSoft (2012c).

### 3. Modelo de Referência MR-MPS-SW e CERTICS

O MPS.BR – Melhoria do Processo de Software Brasileiro é um programa mobilizador iniciado em 2003, coordenado pela SOFTEX (2012a), com o objetivo de melhorar a capacidade das empresas brasileiras e torná-las mais competitivas no mercado internacional. O modelo de referência MPS-SW define 7 níveis de maturidades, que são uma combinação entre o processo e sua capacidade. Assim como 19 processos, agrupados nestes níveis. Os processos no MR-MPS-SW são descritos em termos de propósito e resultados esperados. O propósito descreve o objetivo geral a ser atingido e os resultados esperados do processo estabelecem as evidências esperadas, que devem ser encontradas em produtos de trabalho produzidos pela execução do processo (SOFTEX, 2012a). A capacidade do processo é representada por um conjunto de Atributos de Processo (AP) descritos em termos de Resultados dos Atributos de Processos (RAPs), que evoluem de acordo com o nível de maturidade. Quanto maior a maturidade, um maior nível de capacidade para desempenhar o processo deve ser atingido (SOFTEX, 2012a).

O CERTICS (CTI Renato Archer, 2013) é um modelo de referência para certificação de software desenvolvido pelo Centro de Tecnologia da Informação Renato Acher por solicitação do Ministério da Ciência, Tecnologia e Inovação (CTI Renato Archer, 2013) com o objetivo de verificar se o software avaliado é resultante de desenvolvimento e inovação tecnológica realizado no País. O Modelo de Referência está organizado em quatro camadas, que formam uma estrutura lógica top-down. A primeira camada é formada pelo conceito fundamental de software resultante de desenvolvimento e inovação tecnológica realizados no País. A segunda camada é composta por quatro áreas de competência: (i) Desenvolvimento Tecnológico (DES); (ii) Gestão de Tecnologia (TEC); (iii) Gestão de Negócios (GNE) e (iv) Melhoria Contínua (MEC). A terceira camada é composta por 16 resultados esperados que detalham as áreas de competência e a quarta camada é composta por um conjunto de orientações e indicadores, que detalham os resultados. A Figura 1 apresenta as quatro áreas de competência do Modelo CERTICS e seus resultados esperados.

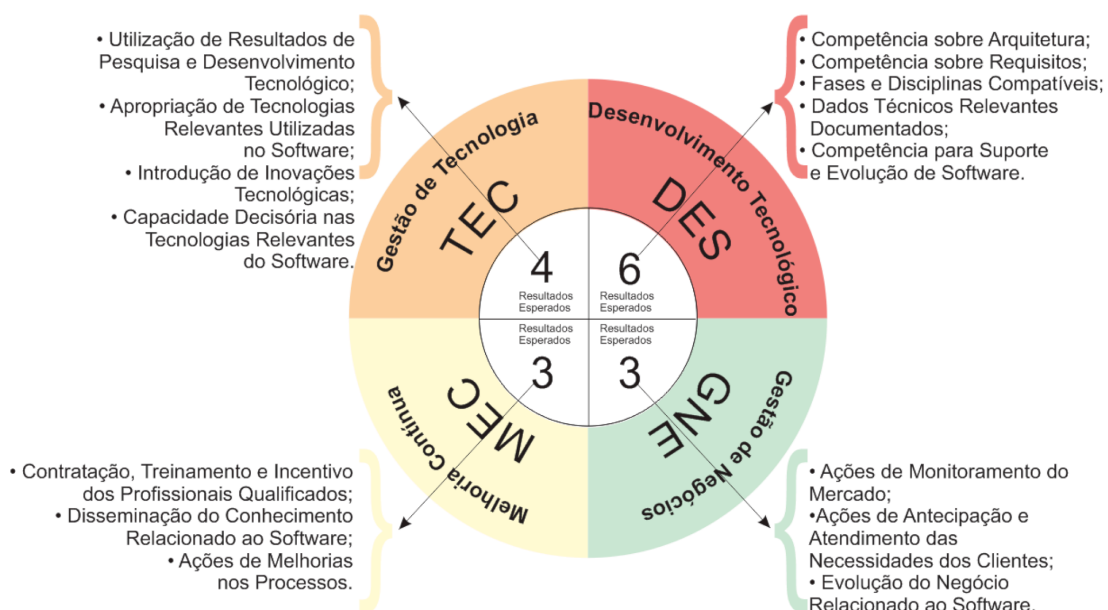


Figura 1: Áreas de Competência do Modelo CERTICS (CTI Renato Archer, 2013)

## 4. Metodologia de Pesquisa

De acordo com BALDASSARRE *et al.* (2010), muitos estudos de mapeamento já foram realizados, porém sem que se definisse qual o processo utilizado para realizá-los, tornando o mesmo não replicável. MELLO (2011) propôs uma metodologia para o mapeamento entre dois modelos, utilizando conceitos replicáveis em outros estudos. Este trabalho utilizou a metodologia definida por MELLO (2011) com pequenas adaptações para melhor adequação às características específicas deste mapeamento. A metodologia utilizada constou das seguintes etapas.

- Revisão da Literatura: teve por objetivo identificar o estado da arte em harmonização multimodelos e realizar um estudo detalhado dos modelos envolvidos neste estudo (MR-MPS-SW e CERTICS).
- Elaboração do Mapeamento dos Modelos, realizada através de quatro atividades: (i) *Análise dos Componentes dos Modelos*, que consiste em uma análise dos modelos a serem mapeados. O objetivo é entender cada modelo com sua estrutura, identificando seus processos e requisitos, para a adequada interpretação e comparação. Com base nos componentes identificados é definida a forma de comparação, assim como os componentes do modelo que serão excluídos do mapeamento, com as respectivas justificativas. (ii) *Definição dos critérios de classificação*, que têm como objetivo permitir que os componentes dos modelos considerados no mapeamento sejam comparados de forma tão clara quanto possível. Para que esta atividade se torne clara e padronizada devem ser definidos critérios de classificação. Além disso ao se realizar o mapeamento, além da aplicação do critério na comparação é incluída uma observação com a justificativa da classificação. (iii) *Definição de formulário padrão* que permita descrever os componentes dos modelos sendo comparados, a indicação da equivalência ou não e as considerações associadas em um único instrumento. (iv) *Mapeamento*, que é a atividade principal e onde os resultados esperados de um modelo serão comparados com os resultados esperados do segundo modelo com o objetivo de se identificar as interseções e as diferenças.
- Revisão por Pares, com objetivo é obter-se uma crítica qualificada sobre o mapeamento, que permita corrigir as inadequações encontradas, a ser conduzida por especialistas nos modelos considerados. Os autores do mapeamento tinham grande experiência no modelo e de implementação do CERTICS (Mocny, 2014).

## 5. Mapeamento MR-MPS-SW e CERTICS

Seguindo a metodologia de pesquisa, após o estudo da literatura, foram realizadas as seguintes atividades: (i) Análise dos componentes do modelo CERTICS e MR-MPS-SW; (ii). Definição dos critérios de classificação; (iii) Definição do formulário padrão; (iv). Mapeamento e (v) Revisão por Pares.

O Modelo CERTICS está organizado em áreas de competência, cada uma delas com seus respectivos resultados esperados. O modelo MR-MPS-SW está organizado em 19 processos cada um com seus resultados esperados e em atributos de processo com seus resultados esperados. No MR-MPS-SW os processos estão organizados em sete níveis de maturidade e o CERTICS só possui um nível de maturidade. O CERTICS não possui atributos de processo. Entretanto, alguns dos resultados

esperados das áreas de competência podem ser mapeados para resultados esperados de atributos de processo no MR-MPS-SW. De acordo com BALDASSARRE *et al.* (2010) para realizar o mapeamento é necessário definir um modelo de origem e um modelo de destino. O CERTICS foi escolhido como o modelo de origem e o MR-MPS-SW como o modelo de destino na comparação, pois foi necessário comparar cada resultado esperado do CERTICS com resultados esperados dos processos do MR-MPS-SW, para cada nível de maturidade. O mapeamento realizado será descrito em detalhes na Seção 6.

Como forma de obter uma comparação clara e padronizada, o mapeamento entre os modelos deve seguir um critério de classificação. Neste caso, foram utilizados os seguintes critérios de classificação: (i) **COB**: Coberto: O MR-MPS-SW cobre todas as exigências do resultado esperado pelo CERTICS. (ii) **COB-**: Parcialmente Coberto. O MR-MPS-SW cobre alguns ou vários aspectos do resultado esperado pelo CERTICS. (iii) **NÃO**: Não Coberto. O MR-MPS-SW não cobre o resultado esperado do CERTICS.

Para manter a padronização foi definido um formulário padrão para a realização do mapeamento com o objetivo de organizar adequadamente o trabalho, além de melhor identificar os modelos e seus respectivos resultados esperados. O modelo apresentado na Tabela 1 foi definido para ser utilizado na comparação entre os resultados esperados por áreas de competências do CERTICS com os resultados esperados de processos MR-MPS-SW por níveis de maturidade. No formulário as descrições dos objetivos dos resultados esperados, por área de competência do CERTICS serão destacadas seguindo as respectivas declarações no modelo. Estas definições são importantes por conter a explicação dos resultados esperados. As orientações do modelo CERTICS também são destacadas, por resultado esperado da área de competência.

**Tabela 1: Modelo de Formulário para Comparação do CERTICS com o MR-MPS-SW**

CERTICS • [ÁREA DE COMPETÊNCIA] [SIGLA] X MR-MPS-SW				
CERTICS		MR-MPS-SW	Cobertura do MPS	Considerações
<p><b>[SIGLA]: [RÓTULO DA ÁREA DE COMPETÊNCIA]</b>  <b>[Definição do Resultado Esperado]</b>  <b>Orientações para que este resultado esperado seja atendido:</b>  <b>(i)</b> [Detalhamento do Resultado esperado, por tópico esperado];  <b>(ii)</b> [..].</p>	<b>G</b>	[Resultado esperado de processo atendido por nível]	<p><b>[Cobertura atendida por nível]</b></p>	<p>[Considerações do Mapeamento].</p>
	<b>F</b>	[Resultado esperado de processo atendido por nível]		
	<b>E</b>	[Resultado esperado de processo atendido por nível]		
	<b>D</b>	[Resultado esperado de processo atendido por nível]		
	<b>C</b>	[Resultado esperado de processo atendido por nível]		
	<b>B</b>	[Resultado esperado de processo atendido por nível]		
	<b>A</b>	[Resultado esperado de processo atendido por nível]		



Como no mapeamento, um resultado esperado pelo CERTICS pode estar relacionado a um conjunto de resultados esperados do MR-MPS-SW e cada nível de maturidade MPS podem acrescentar resultados que complementam o resultado esperado pelo CERTICS, às considerações descrevem esta evolução e fornecem diretrizes para a implantação conjunta dos dois modelos. O mapeamento do CERTICS com o MR-MPS-SW utilizou os critérios de avaliação e os formulários definidos para comparar os resultados esperados das áreas de competência do CERTICS com resultados esperados de processos e atributos de processos do MR-MPS-SW.

Ao final do mapeamento foi realizada uma revisão por pares que teve por objetivos: (i) avaliar se as áreas de competência CERTICS estavam adequadamente comparadas aos processos do MR-MPS-SW: 2012; (ii) avaliar se os resultados esperados das áreas de competência do CERTICS estavam adequadamente comparadas aos resultados esperados dos processos do MR-MPS-SW: 2012 por nível de capacidade; (iii) avaliar se as exclusões dos processos do MR-MPS-SW: 2012 do mapeamento estão adequadas e (v) avaliar se os critérios de comparação aplicados estavam adequados.

A seleção do especialista para executar esta atividade foi realizada com base na notória especialização. Foi convidado a realizar a revisão por pares um dos autores do modelo CERTICS, que também é implementador MR-MPS-SW. Após a realização da revisão, o especialista devolveu o mapeamento revisado com as observações e critérios aplicados. Foram identificados um problema Editorial (E), dois Técnico Alto (TA), cinco Técnico Baixo e três Gerais (um TA e dois TB). Após a análise dos problemas identificados, todas as correções solicitadas do tipo TA, TB, E G foram realizadas.

## 6. COBERTURA DO MR-MPS-SW COM RELAÇÃO AO CERTICS

O mapeamento realizado permitiu identificar o grau de cobertura do MR-MPS-SW com relação aos resultados esperados do CERTICS. Não existe um processo do MR-MPS-SW que seja equivalente a uma área de competência CERTICS mas, sim, vários processos onde os resultados esperados tratam dos resultado esperados do CERTICS. A área de competência GNE (Gestão do Negócio) não tem equivalente no MR-MPS-SW. Da mesma forma, os processos do MR-MPS-SW AQU (Aquisição), GQA (Garantia da Qualidade), MED (Medição), GPP (Gerência de Portfólio), GRU (Gerência de Reuso), VER (Verificação), VAL (Validação), DRU (Desenvolvimento para Reuso), GDE (Gerência de Decisão) e GRI (Gerência de Risco) não possuem equivalentes no CERTICS.

### 6.1 Desenvolvimento Tecnológico (DES)

**DES1: Competência sobre a Arquitetura (COB -)** - Em todos os níveis do MR-MPS-SW, GPR 7 e GPR 14 (do processo Gerência de Projetos) garantem que os profissionais envolvidos no projeto sejam capazes de executar suas atividades com competência profissional. A partir do nível E é implantado o processo GRH (Gerência de Recursos Humanos), onde os resultados GRH 1 e GRH 2 exigem que se identifiquem as necessidades da organização e se recrutem indivíduos com estas habilidades e competências. GRH 3 e GRH 6 tratam dos treinamentos de responsabilidade da organização e GRH 7 garante que se avalie a efetividade dos treinamentos. Porém é a partir do nível D com o processo PCP (Projeto e Construção do Produto) que certamente as questões de DES 1 serão tratadas. Ainda

no nível D, o processo ITP (Integração do Produto), nos resultados ITP 3 e ITP 4, tratam das interfaces internas e externas. A implementação de RAP 7 de ITP e PCP garante a capacitação profissional para os responsáveis da arquitetura. Entretanto algumas exigências de DES 1 não são tratadas no MR-MPS-SW em nenhum processo ou atributo de processo.

Para complementar este resultado do CERTICS, uma empresa com MPS-SW deve: (i) evidenciar que os responsáveis pela arquitetura residam no país, sejam contratados CLT ou seja sócios da empresa; (ii) em caso de aquisição de componentes, ter autonomia para a tomada de decisões sobre esses componentes ; (iii) evidenciar a realização de atualizações nos componentes adquiridos; (iv) caso esteja nos níveis G, F ou E garantir que exista uma definição adequada da arquitetura.

**DES 2: Competência sobre os requisitos (COB-) -** Em todos os níveis do MR-MPS-SW, o resultado esperado GPR 7 garante que os profissionais envolvidos no projeto sejam capazes de executar suas atividades com competência profissional e o GPR 9 garante que os dados relevantes do projeto, o que inclui os requisitos, estejam identificados e armazenados com controle de acesso. GPR14 garante que o planejado nestes dois resultados é implementado. Em todos os níveis MPS-SW deve estar implementado o processo GRE (Gerência de Requisitos) onde os resultado GRE 5 trata da gestão de mudanças, GRE 3 trata da rastreabilidade e GRE 4 da consistência entre os produtos de trabalho e os requisitos. Estes dois últimos resultados garantem a consistência do produto ao serem realizadas mudanças. A partir do nível E é implantado o processo GRH onde os resultados GRH 1 e GRH 2 exigem que se identifiquem as necessidades da organização e se recrutem indivíduos com estas habilidades e competências. GRH 3 e GRH 6 tratam dos treinamentos de responsabilidade da organização, o que certamente deve incluir as questões tratadas em DES 1. GRH 7 garante que se avalie a efetividade dos treinamentos. A partir do nível D é implantado o processo DRE e os resultados DRE 3, DRE 4 e DRE 5 apoiam o cumprimento dos requisitos de DES 2, na definição e documentação dos requisitos. O RAP 7 deste processo garante a competência profissional das pessoas que executam o processo DRE e o RAP 13 garante que os produtos de trabalho estejam armazenados com níveis apropriados de controle.

Para complementar este resultado do CERTICS, uma empresa com MR-MPS-SW deve: (i) evidenciar que os responsáveis pelos requisitos residam no país, sejam contratados CLT ou seja sócios da empresa; (ii) em caso de aquisição de componentes, possuir autonomia para a tomada de decisões sobre os requisitos dos componentes; (iii) evidenciar a realização de atualizações nos requisitos dos componentes adquiridos; (iv) caso seja uma empresa com nível MPS anterior ao D, garantir que exista uma definição adequada dos requisitos.

**DES 3: Fases e disciplinas compatíveis com o software (COB-) -** Este resultado esperado do CERTICS é atendido, em parte, pelo nível G do MR-MPS-SW onde é implementado o processo GPR. DES 3 cobre basicamente dois aspectos: que a empresa seja capaz de mostrar as fases e disciplinas realizadas para o desenvolvimento (independente de que estas tenham sido planejadas) e que elas sejam compatíveis com o software. Estes dois aspectos são cobertos por GPR, pois o cronograma é considerado e deve cobrir todas as fases e disciplinas.

Para complementar este resultado do CERTICS, uma empresa com MPS-SW deve garantir: (i) que a organização tenha autonomia para atualização de componentes adquiridos e (ii) no caso de aquisição contendo tecnologia relevante, demonstrar as fases e disciplinas de uma atualização relevante.

**DES 5: Dados técnicos relevantes documentados (COB- até o Nível E; COB a partir do Nível D)** - Em todos os níveis do MR-MPS-SW, GPR 9 e GPR 14 garantem que os dados relevantes do projeto estejam armazenados e acessíveis. O processo GRE, nos resultados esperados GRE1 e GRE2, tratam da documentação dos requisitos do cliente. GRE 3 trata da documentação da rastreabilidade dos requisitos com os demais produtos de trabalho. A partir do nível F do MR-MPS-SW é implementado o processo GCO (Gerência de Configuração) que garante o armazenamento e disponibilização dos dados do projeto de forma mais rigorosa. A partir do nível D, a implementação dos processos DRE (Desenvolvimento de Requisitos), PCP e ITP garante o total atendimento de documentação dos dados de requisitos e arquitetura.

Para complementar este resultado do CERTICS, uma empresa com MPS-SW níveis G, F ou E deve garantir que os requisitos e a arquitetura estejam definidos e completamente documentados.

**DES 6: Competência para suporte e evolução do sistema (NÃO)** - O MR-MPS-SW não possui processos e resultados de atributos de processos relacionados ao resultado esperado DES 6.

## 6.2 Gestão de Tecnologia (TEC)

**TEC 1: Utilização de resultados de pesquisa e desenvolvimento tecnológico (NÃO)** - O MR-MPS-SW não possui processos e resultados de atributos de processos relacionados ao resultado esperado TEC 1.

**TEC 2: Apropriação das tecnologias relevantes utilizadas no software (COB- até o Nível E; COB a partir do Nível D)** - Em todos os níveis do MR-MPS-SW, GPR 7 garante que os profissionais envolvidos no projeto sejam capazes de executar suas atividades com competência e GPR 9 garante que os dados relevantes do projeto estejam identificados e armazenados com controle de acesso. GPR14 garante que o planejado nestes dois resultados é implementado. A partir do nível E, é implantado o processo GRH onde os resultados GRH 1 e GRH 2 exigem que se identifiquem as necessidades da organização e se recrutem indivíduos com estas habilidades e competências. GRH 3 a GRH 6 tratam dos treinamentos de responsabilidade da organização, o que certamente deve incluir as questões de apropriação das tecnologias relevantes conforme requerido em TEC 2. GRH 7 garante que se avalie a efetividade dos treinamentos. GRH 9, GRH 10 e GRH 11 garantem a implantação de um Sistema de Gestão do Conhecimento, com a disponibilização e compartilhamento dos registros de gestão do conhecimento. A partir do nível D, o RAP 7 garante a competência profissional das pessoas que executam o processo DRE, PCP e ITP, o que pode não estar garantido nos níveis anteriores.

Para complementar este resultado do CERTICS, uma empresa com MPS-SW deve: (i) nos níveis G, F e E garantir que os requisitos e a arquitetura estejam definidos e documentados e (ii) nos níveis G e F garantir, também, os resultados de GRH relacionados a treinamento organizacional e gerência do conhecimento.

**TEC 3: Introdução de inovações tecnológicas (NÃO)** - O MR-MPS-SW não possui processos e resultados de atributos de processos relacionados ao resultado esperado TEC 3.

**TEC 4: Capacidade decisória nas tecnologias relevantes do software (NÃO)** - A capacidade decisória para atualizações no caso de tecnologias relevantes adquiridas não é exigência de nenhum resultado de processo ou atributo de processo MPS.

### 6.3 Melhoria Contínua (MEC)

**MEC 1: Contratação, treinamento e incentivo dos profissionais qualificados (COB-)** - Em todos os níveis do MR-MPS-SW, GPR 7 e GPR 14 garantem que todos os profissionais envolvidos no projeto sejam capazes de executar suas atividades com competência profissional. RAP 7 garante a competência dos profissionais para os processos executados. A partir do nível E é implantado o processo GRH onde os resultados GRH 1 e GRH 2 exigem que se identifiquem as necessidades da organização e se recrutem indivíduos com estas habilidades e competências. GRH 3 a GRH 6 tratam dos treinamentos de responsabilidade da organização. GRH 7 garante que se avalie a efetividade dos treinamentos.

Para complementar este resultado do CERTICS, uma empresa com MR-MPS-SW deve: (i) garantir que os indivíduos contratados para as atividades tecnológicas, de negócio, suporte e evolução do produto possuam as habilidades e competências necessárias ou recebam treinamentos pertinentes; (ii) que existam programas de incentivo para os funcionários.

**MEC 2: Disseminação do conhecimento relacionado ao software (NÃO até o Nível F; COB a partir do Nível E)** - Nos níveis G e F não há resultados do MR-MPS-SW relacionados à gerência do conhecimento, que é implantada a partir do nível E, com o processo Gerência de Recursos Humanos.

Para complementar este resultado do CERTICS, uma empresa com MPS-SW níveis G e F deve implantar algum mecanismo de gerência do conhecimento.

**MEC 3: Ações de melhorias nos processos (COB- até Nível B; COB no Nível A)** - No nível G do MR-MPS-SW, RAP 3 garante que a execução do processo seja planejada o que implica em que pelo menos os processos Gerência de Projetos e Gerência de Requisitos estejam documentados, o que já satisfaz MEC.3 para atividades tecnológicas. A partir do nível E é implementado o processo DFP (Definição do Processo Organizacional) e, para todos os processos dos níveis implementados, é definido o processo padrão com tarefas, atividades, papéis e produtos de trabalho. O processo AMP (Avaliação e Melhoria do Processo Organizacional), também, é implementado neste nível o que implica na existência de dados sobre a execução dos processos, identificação e implantação de melhorias. No nível E é requerida a existência de um profissional ou grupo de profissionais responsável pelos processos. Logo, a partir do nível E, com principalmente DFP e AMP, o MPS cobre mais ainda a atividades tecnológicas, com mais exigências. Porém as atividades de negócios, não estão no escopo deste nível do MR-MPS-SW. No nível A, com a RAP 35 sobre objetivos de negócio, o MR-MPS-SW cobre totalmente MEC.3.

Para complementar este resultado do CERTICS, uma empresa com MPS-SW níveis G ao B deve garantir que as melhorias relativas ao objetivo de negócio sejam implementadas.

## 7. Considerações Finais

Este artigo descreveu um mapeamento para harmonização dos modelos MR-MPS-SW e CERTICS, que poderá nortear a implantação conjunta destes dois modelos. O mapeamento, também, poderá ser útil para organizações que já possuem o MPS-SW e estão se preparando para a certificação CERTICS. As orientações que devem ser seguidas ao longo da implantação do segundo modelo (CERTICS)

estão descritas neste artigo, considerando o nível de capacidade MR-MPS-SW. O resultado fica mais completo de acordo com o nível implantado. Hoje, implantando todos os níveis de maturidade do MPS SW, alcançamos uma média de 26% de resultados esperados do CERTICS. Isto porque o escopo do CERTICS não é apenas de desenvolvimento, mas permeia uma parte de P&D, análise de mercado e manutenção de software.

O mapeamento nos dois sentidos seria possível, porém seria necessário um conjunto de resultados esperados do MPS para atender a um resultado do CERTICS, logo, uma visualização de processos diferentes com vários resultados específicos para representar a cobertura de um único resultado do CERTICS.

Este trabalho, entretanto, tem uma limitação. Embora o mapeamento tenha sido avaliado por meio de revisão por pares de notório especialista, este não foi ainda avaliado em uma situação real. Como trabalho futuro, pretende-se realizar um estudo de caso em uma organização que já possua o MR-MPS-SW e deseje certificar o software produzido no CERTICS para identificar se o mapeamento proposto é útil e de fácil utilização.

## Agradecimentos

Os autores agradecem a Clênio Salviano pela revisão por pares do mapeamento e a Marcelo Thiry, Adriano Albuquerque e Geraldo Xexéo pelas contribuições ao trabalho.

## Referências

- BALDASSARRE, M., CAIVANO, D., PINO, F. J., PIATTINI, M., & VISAGGIO, G. (2010). A Strategy for Painless Harmonization of Quality Standards: A Real Case. PROFES.
- BALDASSARRE, M., CAIVANO, D., PINO, F. J., PIATTINI, M., & VISAGGIO, G. (2011). Harmonization of ISO/IEC 9001:2000 and CMMI-DEV from a theoretical comparison to a real case application. Springer Science+Business Media .
- CATER-STEEL, A., TAN, W.-G., & TOLEMAN, M. (2006). Challenge of adopting multiple process improvement frameworks. ECIS 2006 Proceedings .
- CTI Renato Archer. (2013). Modelo de Referência para Avaliação da CERTICS - Documento de Detalhamento . Campinas: Centro de Tecnologia da Informação Renato Archer.
- FERREIRA, A. L., MACHADO, R. J., & PAULK, M. C. (2011). Supporting Audits and Assessments in Multi-model Environments. Springer-Verlag Berlin Heidelberg , LNCS 6759 (pp. 73-87).
- ISO/IEC. (2003). ISO/IEC 15504-2: Information Technology - Process Assessment - Part 2 - Performing an Assessment (Geneve ed.). Geneve: INTERNATIONAL ORGANIZATION FOR STANDARDIZATION / INTERNATIONAL ELECTROTECHNICAL COMMISSION.
- itSMF UK. (2011). An Introductory Overview of ITIL® 2011. London: The IT Service Management Forum UK.

- KELEMEN, Z. D. (2013). Process Based Unification for Multi-Model Software Process Improvement. Ritter nyomda, Budapest (Hungary): Eindhoven University of Technology Library.
- MELLO, M. S. (2011). Melhoria de Processos de Software Multi-Modelos Baseada nos Modelos MPS e CMMI-DEV. Rio de Janeiro.
- MOCNY, E., ARAUJO, L., MONTONI, M., IRIGOYEN, A. (2014). Relato de Experiência da Certificação do software PRIME Saúde da ECO Sistemas na CERTICS. Rio de Janeiro.
- PARDO, C., PINO, F. J., GARCÍA, F., PIATTINI, M., & BALDASSARRE, M. (2012). An Ontology for the Harmonization of Multiple standards and Models. Computer Standards & Interfaces .
- PARDO, C., PINO, F. J., GARCÍA, F., VELTHIUS, M. P., & BALDASSARRE, M. (2011). Trends in Harmonization of Multiple Reference Models. Springer-Verlag Berlin Heidelberg, CCIS 230, pp. 61-73.
- PELDZIUS, S., & RAGAISIS, S. (2011). Comparison of Maturity Levels in CMMI-DEV and ISO/IEC 15504. Applications of Mathematics and Computer Engineering .
- PINO, F. J., et al. (2009). Harmonizing maturity levels from CMMI-DEV and ISO/IEC 15504. Journal of Software Maintenance and Evolution: Research and Practice (279-296).
- SEI. (2010). CMMI for Development (CMMI-DEV), Version 1.3. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.
- SOFTEX. (2012b). Guia de Implementação Parte 12: Análise da Aderência do MR-MPS-SW:2012 em relação à NBR ISO/IEC 29110-4-1:2012 - Engenharia de Software - Perfis de ciclo de vida para micro-organizações (VSEs) - Parte 4-1: Especificações de perfil: Grupo Perfil Genérico. Campinas: SOFTEX.
- SOFTEX. (2012c). Guia de Implementação Parte 13: Mapeamento e Sistemas de Equivalências entre o MR-MPS-SW:2012 e o MoProSoft:2005. Campinas: SOFTEX.
- SOFTEX. (2012a). Guia Geral MPS de Software. Campinas: Sociedade SOFTEX.
- TENNANT, G. (2001). Six Sigma - SPC e TQM in Manufacturing and Services. Burlington: Gower Publishing.



# Explorando Efeitos da Adoção do MPS-SW no Desempenho das Organizações

Matheus Rabello<sup>1</sup>, Victor Stroele<sup>1</sup>

<sup>1</sup>Departamento de Ciência da Computação – Universidade Federal de Juiz de Fora

Caixa Postal 15.064 – 36.036-330 – Juiz de Fora – MG – Brasil

{matheus,victor.stroele}@ice.ufjf.br

**Abstract.** [Context] The model MPS-SW currently has wide adoption and approaches 600 evaluations conducted in different organizations. [Objective] The purpose of this paper is explore potential effects of the adoption of the MPS-SW in the performance of organizations that adopt this model. [Method] To carry out the work were used as basis the cumulative iMPS survey data. The study consisted in identifying relevant factors and developing a Bayesian network linking these factors with potential causal relationships and feed this network with the iMPS survey data. [Result] The explorative use of the Bayesian Network inferences enabled identified a set of six propositions that may contribute to understanding the potential impact of the adoption of the model. [Conclusions] The Bayesian Network facilitated exploring the information and identifying propositions. Propositions can serve as a basis for formulating hypotheses and conducting further investigations.

**Resumo** [Contexto] O modelo MPS-SW atualmente possui ampla adoção e se aproxima da marca de 600 avaliações realizadas em diferentes organizações. [Objetivo] O objetivo deste artigo é explorar possíveis efeitos da adoção do MPS-SW no desempenho das organizações que adotam este modelo. [Método] Para realizar o trabalho foram utilizados como base os dados acumulativos da pesquisa iMPS. O trabalho consistiu em identificar fatores relevantes e elaborar uma Rede Bayesiana relacionando estes fatores com potenciais relações causais e alimentar esta rede com os dados da pesquisa iMPS. [Resultado] O uso exploratório de inferências na Rede Bayesiana permitiu identificar um conjunto de seis proposições que podem contribuir para compreender o potencial impacto da adoção do modelo. [Conclusões] A Rede Bayesiana facilitou explorar as informações e identificar proposições. As proposições podem servir como base para formular hipóteses e conduzir investigações adicionais.

## 1. Introdução

A melhoria contínua da capacidade de desenvolvimento de software é fundamental para que organizações prosperem em mercados competitivos. Com o passar dos anos, modelos de referência têm surgido para ajudar na implementação da capacidade de processos de engenharia de software. Entretanto, a melhoria baseada nesse tipo de modelo costuma ser de longo prazo e requerer grandes investimentos. Esses obstáculos podem se tornar impeditivos para que organizações melhorem seus processos, especialmente para pequenas e médias empresas (PMEs) que operam sob rígidas restrições financeiras [Santos et al., 2012]. O programa MPS.BR representa uma iniciativa para aumentar a capacidade das empresas Brasileiras em desenvolver software [Softex, 2012].

O principal objetivo do programa MPS.BR é desenvolver e disseminar modelos de referência que atendam a indústria Brasileira de Software e Serviços de TI. Para tanto ele estabeleceu um caminho economicamente viável para que as organizações incluindo as pequenas e médias empresas, alcancem



os benefícios da melhoria de processos e dessa forma, possam usufruir das boas práticas da engenharia de software e da prestação de serviços de TI em um intervalo de tempo razoável [Softex, 2012].

Dentre os principais problemas que inibem as organizações especialmente PMEs, de adotar modelos de processos de software; tais como CMMI, ISO/IEC 12207 e ISO/IEC 15504, estão o alto investimento de implementação, manutenção e avaliação e a dificuldade de convencê-las dos potenciais benefícios dos investimentos em melhoria de processos. Para superar esses problemas são necessárias diferentes abordagens que busquem a mudança do cenário de processos em organizações de software. Neste contexto, a Softex decidiu realizar uma iniciativa nacional, denominada Programa MPS.BR, com o objetivo de disseminar as boas práticas da engenharia de software e tornar as organizações mais competitivas não apenas no mercado local, mas, também, no mercado global [Kalinowski et al., 2010].

O modelo MPS-SW (para software) obteve, com o passar dos anos uma ampla aceitação pelas empresas de software Brasileiras e já conta com mais de 500 avaliações. É imprescindível entender as variáveis de atuação no desenvolvimento técnico e econômico dessas empresas, como prazo, produtividade, custo e qualidade o que promove a compreensão do impacto causado pelo projeto. Nesse contexto foi criado o iMPS, que tem como objetivo acompanhar e evidenciar a evolução das empresas de software que adotaram o modelo MPS-SW [Travassos e Kalinowski, 2014].

Motivado por uma nova abordagem para se estudar o impacto da adoção do MPS.BR foi adotada as Redes Bayesianas também conhecidas como redes causais, modelos gráficos para raciocínio baseado na incerteza, nos quais os nós representam as variáveis (discreta ou contínua) e os arcos a conexão direta entre eles [Pearl, 2000]. O restante deste artigo está organizado da seguinte forma. A Seção 2 contém a fundamentação teórica. A Seção 3 descreve o processo de construção da Rede Bayesiana. A Seção 4 descreve como esta rede foi explorada para identificar proposições. Por fim, a Seção 5 contém as considerações finais do artigo.

## 2. Fundamentação Teórica

Um dos requisitos que o modelo MPS.BR procurou atender foi à incorporação de práticas internacionalmente reconhecidas para a implementação e avaliação de processos de Engenharia de Software [Kalinowski et al., 2010]. O modelo CMMI-Dev foi considerado um complemento técnico para a definição do modelo MPS-SW, e a compatibilidade é importante para as organizações brasileiras que atuam fora do país [Softex, 2012].

Foram definidos sete níveis de maturidade (G, F, E, D, C, B, A) permitindo acesso gradual à melhoria de processos [Softex, 2012]. Cada um dos quais (do nível G - primeiro estágio de maturidade ao nível A - mais maduro) apresenta um conjunto de processos e atributos de processos (AP) que indicam onde a unidade organizacional tem que investir esforço para melhoria. Com o objetivo de compreender o impacto do modelo MPS-SW, o projeto iMPS (informações para acompanhar e evidenciar variação de desempenho nas empresas que adotaram o modelo MPS-SW) foi iniciado em 2007 junto ao Grupo de Engenharia de Software Experimental da COPPE [Kalinowski et al., 2008]. O objetivo desse projeto foi planejar um *survey*, seguindo os princípios da engenharia de software experimental. Esse projeto é fortemente apoiado pela engenharia de software experimental, a qual tem se mostrado importante para evidenciar variações decorrentes da adoção de tecnologias de software.

Para realizar este trabalho serão utilizadas Redes Bayesianas. Esta rede vem se tornando uma metodologia padrão para a construção do conhecimento probabilístico e tem sido aplicada em diferentes áreas de conhecimento [Mendes e Mosley, 2008]. A base teórica das Redes Bayesianas é a obra do matemático Thomas Bayes, chamada Teoria da Probabilidade Bayesiana, publicada em 1763 [Pearl, 2000]. A representação da Rede Bayesiana é feita por meio de um grafo direcionado acíclico no qual os nós representam variáveis de um domínio e os arcos à dependência condicional ou informativa entre as variáveis. Para representar a força da dependência são utilizadas probabilidades associadas a cada grupo de nós pais-filhos na rede [Pearl, 2000].

Muitos trabalhos acadêmicos são desenvolvidos utilizando Redes Bayesianas. Um exemplo é o trabalho *Bayesian Network Models for Web Effort Prediction: A Comparative Study* que teve como objetivo comparar, vários modelos de Rede Bayesiana para a estimativa de esforço *web* através de um conjunto de dados empresariais [Mendes e Mosley, 2008].

### 3. Elaboração da Rede Bayesiana

#### 3.1. Pré-processamento

Os dados utilizados nesse trabalho, conforme já mencionado, foram da pesquisa iMPS [Kalinowski e Travassos, 2014], na qual as organizações respondem a um formulário com várias questões relevantes ao modelo MPS-SW. Em pesquisas anteriores não foram levadas em consideração empresas que apresentavam um comportamento muito discrepante do restante dos dados, ou seja, os *outlier* foram removidos do conjunto de dados [Travassos e Kalinowski, 2012]. Como o trabalho deste artigo foi de propósito exploratório e considera apenas faixas de distribuição de empresas, não visando identificar evidências ou afirmações de validade, mas proposições que podem subsidiar investigações futuras mais elaboradas, o conjunto total de dados foi utilizado.

Para trabalhar com Redes Bayesianas foi necessário escolher um software de apoio. Foi decidido trabalhar com o NETICA já que esse software possui as ferramentas necessárias para apoiar o trabalho. O software tem uma interface intuitiva e permite que os dados sejam inseridos na forma de arquivos de dados, assim, os arquivos da pesquisa iMPS podem ser incluídos diretamente. A ferramenta NETICA possui alguns algoritmos para realizar o aprendizado. Dentre essas opções escolheu-se trabalhar com o algoritmo Expectation Maximization (EM).

O algoritmo EM foi a unificação de uma série de trabalhos, proposto em 1977 por Dempster [Dempster, 1977]. O algoritmo EM, pode ser aplicado quando se deseja estipular algum conjunto de parâmetros que descreve uma distribuição de probabilidades. É utilizado para encontrar os valores de máxima verossimilhança de um<sup>1</sup> modelo estatísticos no qual as equações não podem ser diretamente resolvidas. Isto é, valores de dados faltantes são assumidos, simplificando, assim, a construção do modelo.

---

<sup>1</sup> Informações do software NETICA em: <https://www.norsys.com/netica.html>, acessado em 30/10/2014.

### 3.2. Montagem da Rede Bayesiana

Como base para esse trabalho foram utilizados dados da pesquisa iMPS, da Softex [Travassos e Kalinowski, 2014]. O primeiro passo foi identificar quais dados coletados na pesquisa iriam compor a Rede Bayesiana. Além dos dados selecionados por Travassos e Kalinowski (2014) também foi criado o nó HH/PF(Homem hora por ponto de função), que foi calculado combinando dados da pesquisa iMPS.

Todos os nós da Rede Bayesiana foram divididos em cinco faixas de valores. Os valores que iriam compor cada faixa foram definidos pelos autores e validados por especialistas. Dessa forma obtendo-se uma classificação de faixa do tipo “muito baixa”, “baixa”, “média”, “alta”, “muito alta”.

Após definir os campos (nós) que fariam parte da Rede Bayesiana, o próximo passo foi à montagem da Rede. As ligações entre os campos na Rede estão diretamente relacionadas com as proposições que serão exploradas na próxima seção. A rede elaborada encontra-se na Figura 1. As proposições, neste trabalho levantadas fazendo uso das inferências da rede relatam potenciais efeitos da adoção do modelo de maturidade MPS-SW nas organizações.

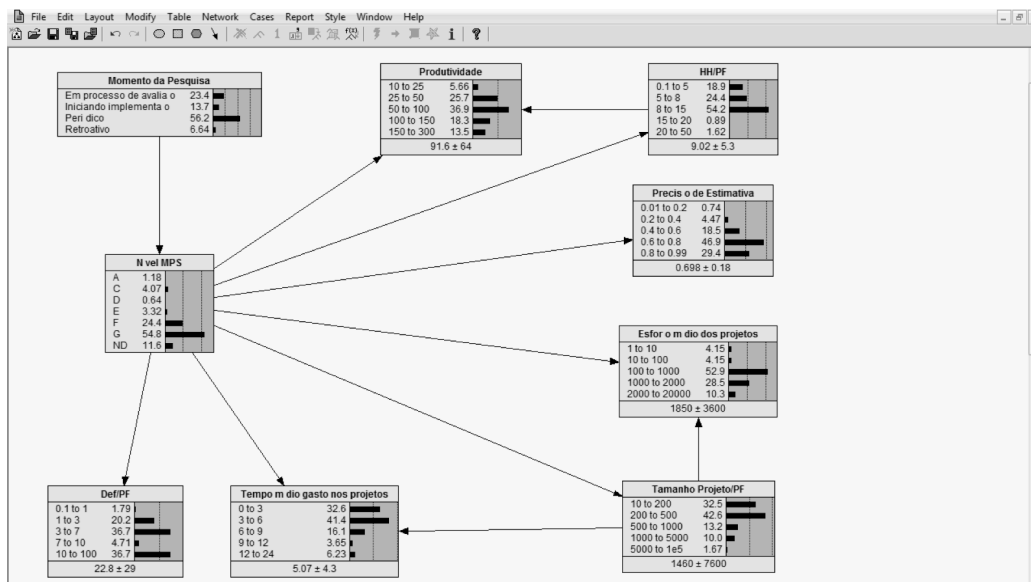


Figura 1: Rede Bayesiana

### 3.3. Análise Crítica

A rede Bayesiana foi elaborada através dos dados da pesquisa iMPS. Essa pesquisa é feita com o auxílio de um questionário composto por questões de livre preenchimento. Assim, o funcionário responsável por inserir dados no sistema da pesquisa pode introduzir as informações como bem entender. Dessa forma, erros por descuido ou pelo não domínio do assunto podem ter sido adicionados aos campos da pesquisa iMPS.

Outro ponto que se deve ter muita atenção: a grande maioria dos dados está concentrada nos níveis de maturidade G e F. É prudente frisar essa questão, uma vez que algumas proposições são levantadas relacionando organizações de níveis G e F com outras de nível A, por exemplo. Apesar dos dados dos níveis E-A não serem maioria, tendem a ser muito representativos, pois são informados por organizações maduras, com grande experiência em desenvolvimento de software.

## 4. Explorando Efeitos da Adoção do MPS-SW

### 4.1. Proposições

Explorar os efeitos da adoção do modelo MPS-SW utilizando uma Rede Bayesiana foi um processo lento e gradativo. Após todos os nós serem escolhidos e a rede devidamente montada, o próximo passo foi explorar as inferências da Redes Bayesianas e identificar algumas proposições. A Rede montada pode fornecer uma grande quantidade de combinações de nós.

O processo de identificação das proposições foi feito com o repetido uso exploratório com base tentativa e erro de se achar questões interessantes. A ideia foi combinar os nós da rede e montar proposições relevantes que relatem potenciais efeitos da adoção do modelo MPS-SW nos processos das organizações.

Antes de tudo é prudente definir o que é uma proposição. Na lógica matemática uma proposição respeita duas regras: a primeira é que a proposição é uma sentença declarativa que pode somente ser julgada como verdadeira (V) ou falsa (F), e a segunda é o princípio da não contradição na qual ela nunca é classificada como verdadeira e falsa simultaneamente. Em síntese, ela deve ser descrita de modo que não possa ser considerada “meio verdadeira e meio falsa”.

Todas as proposições identificadas relacionam-se com o nível MPS-SW o que visa destacar a importância da maturidade de software para as organizações que adotaram o modelo. Assim, o nó “nível” foi combinado de diferentes maneiras com os outros nós da rede. Após o processo de utilização da rede para a descoberta das proposições serão apresentadas algumas análises dos resultados obtidos.

### 4.2. Proposições Identificadas

#### **Primeira proposição: “Organizações de nível A e C gerenciam estimativas melhor que organizações de nível G”**

Essa proposição foi obtida selecionando o nível G no nó nível MPS-SW e depois analisando as faixas de valores no nó precisão de estimativa conforme representado na Figura 2. É possível observar que, no nível G, parte das organizações (28,9%) possui acerto nas estimativas muito altas, ou seja, estão na faixa mais alta de 0,8-0,99 e a maioria (45,6%) possui acerto alto, ficando na faixa de 0,6-0,8. Há entretanto um grande número de empresas (25,8%) distribuído nas faixas de valores inferior a 0,6. No nível A, apesar dos poucos dados, todos estão distribuídos na faixa alta, ou seja de 0,6-0,8. No nível C, a maioria (52,9%) está distribuída na faixa muito alta, ou seja precisão de estimativa entre 0,8-0,99 e outros 29,4% encontram-se na faixa de valores alta, entre 0,6 e 0,8, com apenas 17,68% dos valores nas faixas mais baixas. Esta distribuição permitiu identificar a proposição acima. Tais informações revelam que estar em níveis de maturidade mais altos levam as organizações a estimarem o prazo com mais precisão, já que poucas empresas em altos níveis de maturidade possuem baixa precisão de estimativa.

No nó precisão de estimativa, é importante lembrar que na montagem da Rede Bayesiana, os valores 1 e 0 não foram considerados sendo que o valor 0 significa que a organização não faz qualquer tipo de estimativa e o valor 1 significa que a organização possui 100% de acerto em todos os seus projetos o que é improvável.

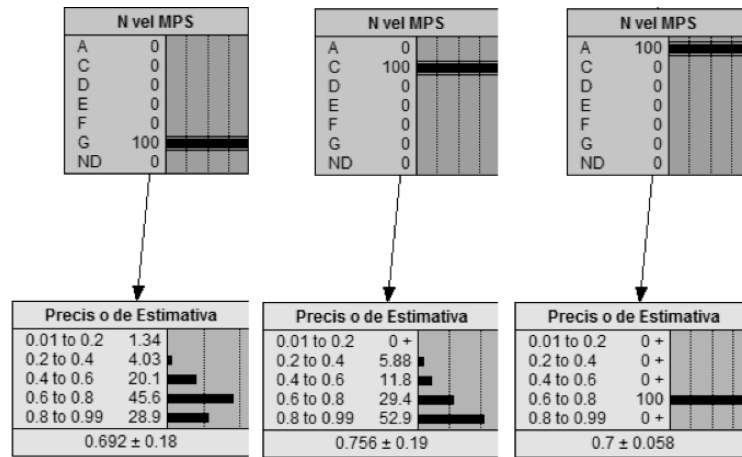


Figura 2: Inferências do Nível de Maturidade na Precisão de Estimativa

**Segunda Proposição: “A produtividade de organizações do Nível F é maior em relação ao nível A. Porém, a qualidade dos projetos (Def./PF) desenvolvidos por organizações nível A é melhor em relação ao nível F”**

Essa proposição indica que a produtividade (considerada de forma isolada e medida em Pontos de Função por mês) é mais alta no nível F, comparado ao nível A, o que não significa que o nível F seja melhor, mas que talvez seja mais objetivo, uma vez que não existem tantos processos a ser executados e, dessa forma, a produção ganha em velocidade. Porém a velocidade tem um preço, já que as organizações nível A possuem menos defeitos por pontos de função. A Figura 3 ilustra quantitativamente a proposição.

Para observar essa proposição na Rede Bayesiana, basta selecionar o nível A e analisar o comportamento dos nós de produtividade e Def./PF (Defeitos por pontos de função), o que também deve ser realizado para o nível F. É possível observar que no nível F, 50,5% dos valores de produtividade estão distribuídos nas faixas de valores baixa e média, e 44% estão nas faixas alta e muito alta. Já no nível A, a produtividade é de 80% nas faixas de valores baixa e média e os Def./PF no nível A estão distribuídos nas faixas de 1-3 e 3-7 Def./Pf. Já no nível F, 56,3% dos valores estão nas faixas mais altas, ou seja, possuem mais de sete Def./PF.

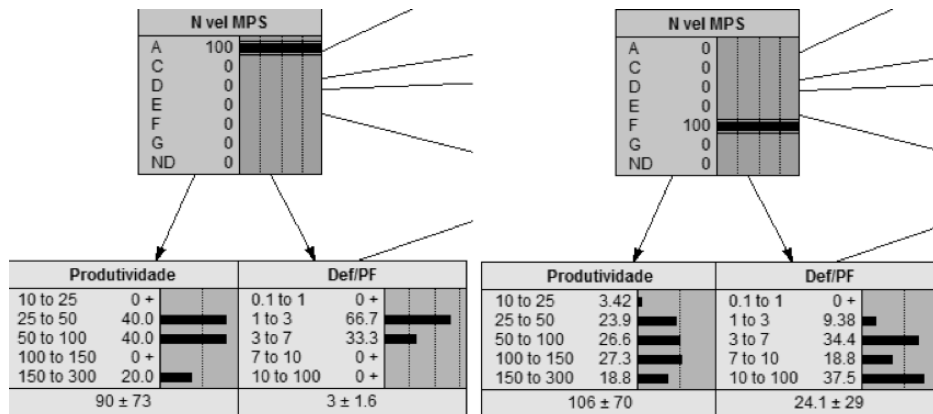


Figura 3: Inferências do Nível na Produtividade e no Número de Defeitos.

**Terceira Proposição: “Organizações sem nível MPS-SW gastam menos tempo para desenvolver projetos muito pequenos do que organizações nível G”**

Essa proposição é importante para comparar o momento anterior à obtenção de um nível MPS-SW com organizações que já têm um nível, no caso, o nível G. Organizações nível G executam dois processos: a Gerência de Projetos e a Gerência de Requisitos. Por outro lado, organizações sem nível muitas vezes desenvolvem projetos sem processos definidos.

Escolhemos essa proposição, pois o número de projetos muito pequenos desenvolvidos por esses grupos são equivalentes. Do total de projetos desenvolvidos 34,5% são muito pequenos. A Figura 4 ilustra quantitativamente a proposição.

Para observar essa proposição na Rede Bayesiana, primeiro deve-se selecionar no nível a opção desejada, nesse caso ND ou G. A seguir, no nó Tamanho projeto/PF, deve-se selecionar a primeira faixa de valores, correspondente aos projetos muito pequenos para então observar as inferências da rede a respeito do tempo médio gasto nos projetos. É possível observar que para as empresas sem nível de maturidade, 86,5% das empresas informaram finalizar seus projetos em até 3 meses, número que reduz para 53,4% para as empresas no nível G. Vale ressaltar entretanto que as medidas das empresas que ainda não iniciaram a implementação do MPS-SW podem ser menos precisas, o que pode afetar diretamente a proposição identificada.

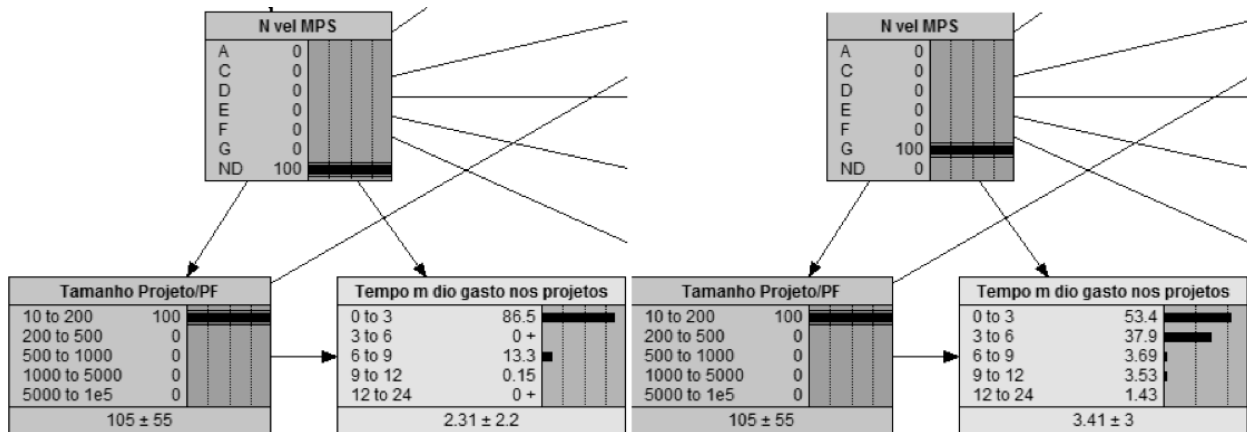


Figura 4: Inferências do Nível MPS no Tamanho e Tempo dos Projetos.

**Quarta proposição: “Organizações nível F produzem mais e gastam menos do que organizações nível C”**

Essa proposição foi obtida observando os nós produtividade (medida em pontos de função por mês) e HH/PF (homem hora por pontos de função). Para observar essa proposição na Rede Bayesiana, primeiro deve-se selecionar no nó nível os campos F ou C. E depois observar o comportamento dos nós produtividade e HH/PF. De acordo com a inferência da rede, a produtividade no nível F se mostra maior e o número de HH/PF necessário no nível F se mostra menor do que no nível C. A Figura 5 ilustra quantitativamente esta proposição.

No nível C, apenas 22,2% das organizações possui produtividade nas duas faixas de valores mais altas. Como mostrado na proposição 2, a produtividade no nível F é superior aos valores apresentados no nível C. Em relação ao nó HH/PF podemos observar que, no nível F, 62,9% utiliza até 8 homens por hora para desenvolver um ponto de função. Já no nível C, todos os projetos utilizam mais de oito homens por hora para desenvolver um ponto de função. Por outro lado, as informações do nível C parecem ser mais confiáveis se compararmos este número com referências técnicas como a base ISBSG, onde a média de horas por ponto de função é acima de 11 [ISBSG, 2014].

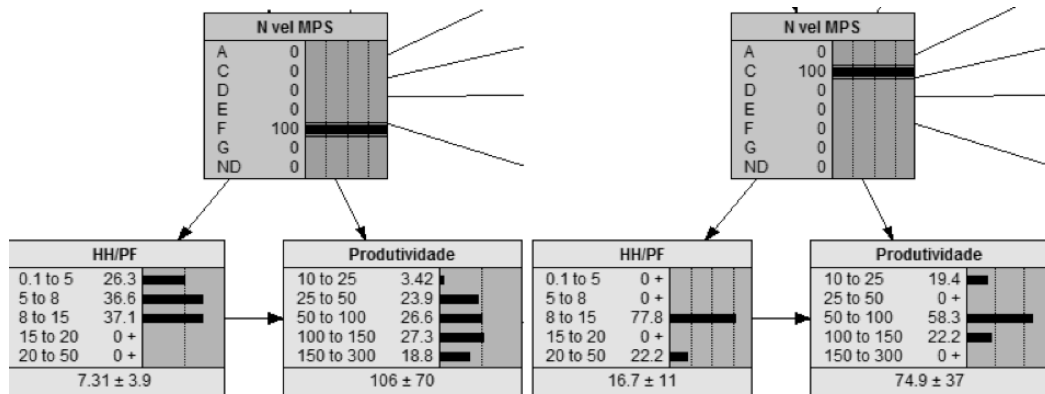


Figura 5: Inferência do Nível no HH/PF e na Produtividade

**Quinta proposição: “Em organizações nível C os projetos gastam mais HH/PF do que no nível F. Porém a qualidade (Def./PF) dos projetos nível C é melhor”**

Essa proposição é um complemento da anterior. A primeira parte refere-se aos HH/PF. No nível C gasta-se mais HH/PF comparado ao nível F, porém, como consequência, no nível C o número de defeitos por pontos de função é menor, o que leva a um ganho de qualidade. A Figura 6 ilustra quantitativamente a proposição.

Enquanto no nível F o número de homem hora por ponto de função se apresenta menor, organizações nível C possuem no máximo sete defeitos por ponto de função. Já 56,2% das organizações nível F possuem mais de sete defeitos por ponto de função.

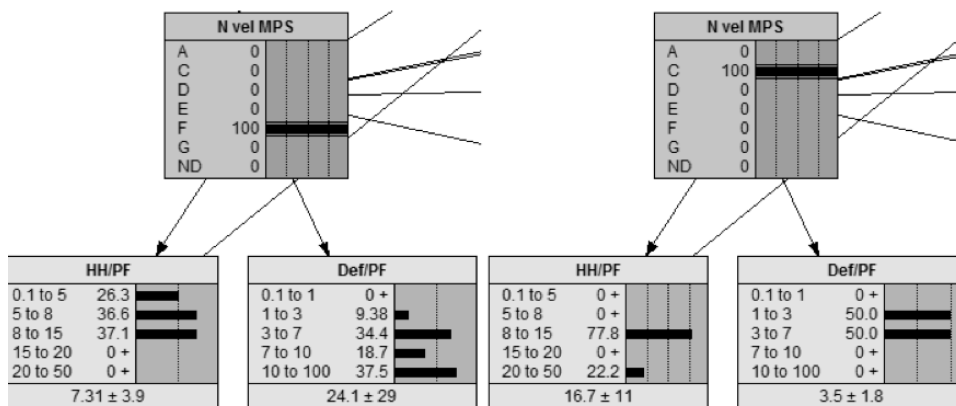


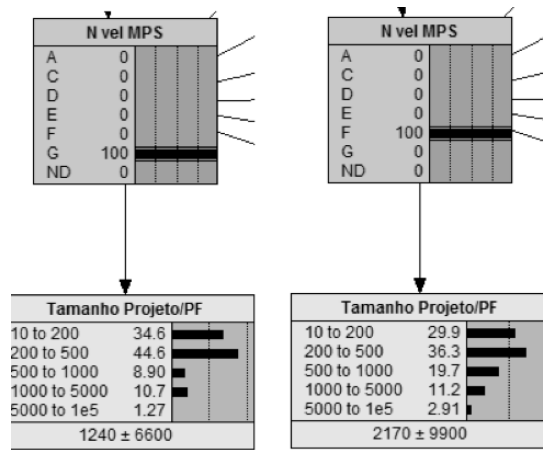
Figura 6: Inferência do Nível no HH/PF e Def/PF

**Sexta proposição: “Os projetos desenvolvidos no nível F, são maiores que os projetos desenvolvidos no nível G”**

Nessa proposição podemos observar que os projetos desenvolvidos por organizações nível F são maiores que os desenvolvidos por organizações nível G. A Figura 7 ilustra quantitativamente a proposição.



Nas três faixas de valores mais elevados, os projetos do nível F correspondem a 33,9% do total e são maiores que os projetos nível G, os quais para esse conjunto de valores possuem 20,9% dos valores. Em outras palavras, há uma tendência das organizações que atingem o nível F de maturidade trabalharem com projetos maiores.



**Figura 7: Inferência do Nível no Tamanho dos Projetos**

De maneira geral, considerando o conjunto de proposições identificadas, o nível F possui bons resultados referentes à produtividade e homem hora por ponto de função. Acreditamos que isso se deva ao fato que os processos nele compreendidos (como, por exemplo, gerência de configuração) garantem a boa execução do projeto. Com base na análise das proposições é possível observar que nos níveis mais elevados aparenta-se ter uma queda nestes valores uma vez que a inclusão de novos processos pode fazer com que o desenvolvimento se torne mais lento. É preciso ressaltar entretanto, que há pouca empresas nos níveis mais altos de maturidade (B e A), onde a gerência quantitativa e a melhoria contínua prometem melhorar estes resultados.

Entretanto, o mesmo nível F apresenta uma aparente queda em relação à qualidade, quando comparado a níveis mais elevados de maturidade, que contam com processos focados na qualidade do produto, como verificação e validação, presentes no nível D e que impactam diretamente na qualidade.

## 5. Considerações Finais

O objetivo desse trabalho foi explorar potenciais resultados da adoção do modelo MPS-SW com base em dados da pesquisa iMPS, respondida anualmente por diferentes organizações, utilizando Redes Bayesianas.

Explorar as inferências da rede bayesiana indicou que adotar o modelo MPS-SW e o nível de maturidade implementado impactam no desempenho do desenvolvimento de software. Foi possível identificar um conjunto de proposições, considerando o relacionamento entre os nós, que tentam ajudar a compreender os fatores relevantes da adoção do modelo de maturidade MPS-SW. Entretanto, é preciso reforçar que por sua natureza, proposições teóricas não representam evidências, mas

oportunidades de investigação futura com técnicas estatísticas mais elaboradas, como as utilizadas por Travassos e Kalinowski (2014). Esperamos portanto que as proposições identificadas sirvam como insumo para análises nas próximas pesquisas iMPS.

É necessário reafirmar que, tendo em vista o caráter exploratório deste artigo, os dados da pesquisa iMPS foram utilizados da maneira que foram fornecidos pelas empresas, sem uma análise detalhada de *outliers*. O impacto de não identificar os *outliers* é reduzido uma vez que a rede trabalha com faixas de distribuição que, diferentemente de médias, não variam significativamente em função da presença de eventuais dados distorcidos. Entretanto há diversos dados claramente incorretos, principalmente concentrados nas empresas de níveis de maturidade mais baixos.

Para a futura evolução desse trabalho, as seguintes sugestões podem ser seguidas: explorar a rede para identificar outras proposições; realizar análises estatísticas mais elaboradas a respeito das proposições identificadas de modo a obter evidências que permitam formular hipóteses mais concretas; adicionar outros nós à rede (por exemplo, o ano de coleta das informações e categoria de serviços), presentes na pesquisa iMPS, para ampliar as possibilidades de inferências.

## Agradecimentos

Os autores gostariam de agradecer à Softex e aos pesquisadores envolvidos na pesquisa iMPS por terem permitido a utilização supervisionada dos dados para a realização desta pesquisa.

## Referências

- Dempster, A.P., Laird, N.M., Rubin, D.B. (1977). "Maximum likelihood from incomplete data via the EM algorithm". *Journal of the Royal Statistical Society: Series B*: pp. 1-38
- Kalinowski, M., Santos, G., Reinehr, S., Montoni, M., Rocha, A. R., Weber, K.C., Travassos, G.H. (2010). "MPS.BR: Promovendo a Adoção de Boas Práticas de Engenharia de Software pela Indústria Brasileira". XIII Congresso Iberoamericano em Software Engineering (CIBSE), Cuenca, Equador.
- Kalinowski, M., Weber, K., Travassos, G.H. (2008). "iMPS: An Experimentation Based Investigation of a Nationwide Software Development Reference Model". 2<sup>nd</sup> International Symposium on Empirical Software Engineering and Measurement (ESEM), Kaiserslautern, Germany.
- Mendes, E., and Mosley, N. (2008). "Bayesian Network Models for Web Effort Prediction: A Comparative Study". *IEEE Transactions on Software Engineering*, 34 (6), pp. 723-737.
- Pearl, J. (2000). "Causality: Models, Reasoning and Inference". Cambridge University Press, ISBN 978-0-521-77362-1.
- Santos, G., Kalinowski, M., Rocha, A.R., Travassos, G.H., Weber, K.C., Antonioni, J.A. (2012). "MPS.BR program and MPS model: main results, benefits and beneficiaries of software process improvement in Brazil", 8<sup>th</sup> International Conference on the Quality in Information and Communications Technology (QUATIC), Lisbon, Portugal.

Softex (2012). “MR MPS-SW Guia Geral MPS de Software”, disponível em <http://www.softex.br/mpsbr/>, último acesso em 20/09/2014

Travassos, G.H., Kalinowski, M. (2014). “iMPS 2013 – Evidências Sobre o Desempenho das Empresas que Adotaram o Modelo MPS-SW”. Softex, ISBN 978-85-99334-75-1.

# Uma Análise da Influência do Programa MPS.BR em Artigos Técnicos Publicados no WAMPS

Diego Cruz<sup>1,3</sup>, Raphael Freire<sup>1</sup>, Eliezer Dutra<sup>1,2</sup>, Gleison Santos<sup>1</sup>

<sup>1</sup> Programa de Pós-Graduação em Informática - Universidade Federal do Estado do Rio de Janeiro (UNIRIO) - Av. Pasteur, 458, Urca, CEP 22290-240 - Rio de Janeiro, RJ

<sup>2</sup> Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (CEFET-RJ) Estrada de Adrianópolis, 1317 - Santa Rita, CEP: 25265-000 - Nova Iguaçu, RJ

<sup>3</sup> E-novasys Tecnologia da Informação - Av. Franklin Roosevelt, 84/903, Centro, CEP:20021-120 - Rio de Janeiro - RJ

{diego.dacruz, raphael.freire, eliezer.goncalves, gleison.santos}@uniriotec.br

**Abstract.** *This paper aims to quantitatively and qualitatively analyze how MPS.BR program and its reference models have been used to support the scientific research on Software Quality in the context of WAMPS. Evaluation was performed based on Grounded Theory concepts and procedures. As a result, we were able to identify last five years processes trends, levels and most referenced documents in technical papers, and then establish a reasoning relevance mapping in scientific research about MPS.BR and its reference models.*

**Resumo.** *Este artigo tem o objetivo de analisar, quantitativamente e qualitativamente, como o programa MPS.BR e seus modelos de referência têm sido utilizados para fundamentar pesquisas de qualidade de software no contexto do WAMPS. Esta análise foi executada com base em conceitos e procedimentos da Grounded Theory. Como resultado, foi possível identificar tendências de processos, níveis e documentos mais referenciados em artigos técnicos nos últimos 5 anos, o que permitiu elaborar um mapeamento de relevância das fundamentações em pesquisas científicas relacionadas ao MPS.BR e seus Modelos de Referência.*

## 1. Introdução

A Engenharia de Software sempre se beneficiou da observação do estado-da-prática para definição de tópicos de pesquisa (BOEHM, 2006) com objetivo de construir software com qualidade e baixo custo (TEIXEIRA e DELAMARO, 2008). Cada vez mais isso é uma necessidade no país (SANTOS *et al.*, 2012). Nesse sentido, a partir das necessidades do estado-da-prática em iniciativas de melhoria de processos de software, pesquisas têm sido desenvolvidas para apoiar a implementação de práticas definidas nos Modelos de Referência do programa MPS.BR (SOFTEX, 2012), o que promove maior interação entre pesquisadores e organizações que desenvolvem software.

O MPS.BR, criado em dezembro de 2003, é um programa mobilizador, de longo prazo, coordenado pela Associação para Promoção da Excelência do Software (SOFTEX), que conta com o apoio do Ministério da Ciência, Tecnologia e Inovação (MCTI), Financiadora de Estudos e Projetos (FINEP), Serviço Brasileiro de Apoio às Micro e Pequenas Empresas (SEBRAE) e Banco Interamericano de Desenvolvimento (BID/FUMIN) (SOFTEX, 2012). Em um trabalho anterior (SANTOS, 2011), indícios mostraram que desde que o MPS.BR foi criado houve um aumento no número de publicações que se fundamentam em algum aspecto dos seus Modelos de Referência, porém, nenhuma análise detalhada havia sido realizada.

Este artigo apresenta resultados da execução de um estudo baseado em mapeamento sistemático e análises qualitativa e quantitativa, sobre como o programa MPS.BR e seus modelos de referência têm sido utilizados para fundamentar pesquisas de qualidade de software. Foram analisados artigos do WAMPS (Workshop Anual do MPS.BR) dos últimos 5 anos para investigar que processos do MR-MPS-SW e do MR-MPS-SV foram utilizados em fundamentações, formando assim um corpo de conhecimento sobre a utilização desses modelos, o que também pode possibilitar a identificação de oportunidades de pesquisa em áreas pouco exploradas.

O artigo está organizado em 4 seções, além desta introdução. A Seção 2 descreve a metodologia utilizada para execução do mapeamento sistemático e conceitos de Grounded Theory aplicados no procedimento de análise qualitativa, enquanto a Seção 3 apresenta detalhes de aplicação da metodologia e seus principais resultados. Por fim, as considerações finais são apresentadas na Seção 4.

## 2. Protocolos de Pesquisa

### 2.1 Mapeamento Sistemático da Literatura

O desenvolvimento de uma abordagem sistemática de revisão visa estabelecer um processo formal para conduzir este tipo de investigação, evitando a introdução de eventuais vieses da revisão de literatura informal (MAFRA e TRAVASSOS 2006). O estudo de mapeamento sistemático fornece uma visão geral de uma área de pesquisa, identificando a quantidade, os tipos de pesquisas realizadas, os resultados disponíveis, além das frequências de publicações ao longo do tempo para identificar tendências (PETERSEN *et. al.*, 2008).

O protocolo de mapeamento sistemático da literatura utilizado para o desenvolvimento desta pesquisa seguiu as diretrizes descritas em (SANTOS, 2011). No contexto do presente trabalho, a aplicação dos protocolos de mapeamento sistemático e de análise qualitativa foi realizada por uma equipe de pesquisadores de Engenharia de Software da UNIRIO, com objetivo de obter informações sobre como o MPS.BR e seus modelos de referência influenciaram as propostas de pesquisa elaboradas no contexto acadêmico, de modo a se obter respostas às seguintes questões de pesquisa:

- Q1: “Como o Programa MPS.BR e os seus modelos de referência têm influenciado na fundamentação das propostas de pesquisa?”
- Q2: “Que processos e níveis do MR-MPS-SW e do MR-MPS-SV são utilizados na fundamentação das publicações selecionadas?”

Como critérios de inclusão no escopo foram definidos: (i) **artigos técnicos** cujos autores incluem membros da **Academia** (ou seja, alguma instituição de ensino ou pesquisa), cuja pesquisa seja **fundamentada** no MPS.BR ou seus Modelos de Referência, (ii) no contexto de publicações do WAMPS (iii) produzidas do período relacionado entre 2009 e 2013. Como critérios de exclusão do escopo foram identificados: (i) artigos que **apenas citam** o MPS.BR ou seus modelos de referência; (ii) artigos identificados como **relatos de experiência**; e (iii) artigos que **não se aplicam** ao contexto do MPS.BR.

Para a estratégia de busca, todos os artigos publicados foram lidos e analisados em relação aos critérios de inclusão e exclusão. Os dados da referência completa e informações relevantes identificadas durante a leitura dos artigos foram coletados, catalogados e armazenados em um Formulário de Condução da Revisão (ver SANTOS, (2011)). Os resultados do mapeamento sistemático bem como conflitos de ideias durante a aplicação do protocolo foram discutidos entre os envolvidos.

## 2.2 Grounded Theory

O método Grounded Theory (GT ou Teoria fundamentada a dados) descreve um conjunto de procedimentos sistemáticos de coleta e análise dos dados para gerar, elaborar e validar teorias substantivas sobre fenômenos essencialmente sociais, ou processos sociais abrangentes (BANDEIRA-DE-MELLO e CUNHA, 2003). Embora a finalidade do método Grounded Theory seja a construção de teorias substantivas, sua utilização não necessariamente precisa ficar restrita apenas aos pesquisadores que tem esse objetivo de pesquisa. Segundo Corbin e Strauss (1998), o pesquisador pode usar apenas alguns de seus procedimentos para satisfazer seus objetivos de pesquisa.

Segundo a linha proposta por Strauss, a Grounded Theory é baseada no procedimento de codificação (*coding*), que é o processo de analisar os dados. Durante a codificação são identificados conceitos (ou códigos) e categorias. Um conceito (ou código) dá nome a um fenômeno de interesse para o pesquisador; abstrai um evento, objeto, ação, ou interação que tem um significado para o pesquisador (CORBIN e STRAUSS, 1998). Categorias são agrupamentos de conceitos unidos em um grau de abstração mais alto. O processo de codificação pode ser dividido em três fases: codificação aberta, axial e seletiva. A codificação aberta envolve a quebra, a análise, a comparação, a conceituação e a categorização dos dados. Segundo Bandeira-de-Melo e Cunha (2006), nas fases iniciais da codificação aberta, o pesquisador explora os dados examinando minuciosamente aquilo que lhe parece relevante devido à leitura intensiva dos textos. Na fase de codificação aberta, os incidentes ou eventos são agrupados em códigos através da comparação incidente–incidente. Após a identificação de categorias conceituais pela codificação aberta, a codificação axial examina as relações entre as categorias que formam as proposições da teoria substantiva (BANDEIRA-DE-MELLO e CUNHA, 2006).

Os artigos no escopo do mapeamento sistemático, ou seja, que respeitavam os critérios de inclusão e exclusão, foram analisados em três rodadas, sendo uma de codificação aberta e duas de codificação axial. Para a realização da codificação foi utilizado o software Atlas.TI. A primeira rodada incluiu a leitura de cada artigo e escrita de um resumo sobre como os autores fundamentaram a pesquisa. Os resumos dos artigos são respostas completas e objetivas às questões de pesquisa Q1 e Q2. Além da elaboração do resumo, durante a primeira rodada também foi aplicado o procedimento de codificação aberta. A aplicação deste procedimento consistiu na associação de trechos do artigo a códigos criados pelo pesquisador. Os códigos criados durante esta etapa representam características do artigo, de forma que um código pode estar associado a trechos de artigos distintos.

## 3. Análise dos Resultados

No período entre 2009 e 2013, inclusive, 92 artigos foram publicados no WAMPS. A aplicação do protocolo foi realizada em 2 etapas. Na Extração de dados, foram coletadas informações padrão para todos os artigos: autores, referência completa e se possui ou não participação da Academia entre os autores. Na *Classificação dos artigos*, os 92 trabalhos foram classificados com relação ao escopo de

pesquisa. 18 artigos cuja análise apontou não participação da Academia foram excluídos do escopo. Dos 74 artigos restantes, todos os *abstracts/resumos* foram lidos e os artigos foram classificados preliminarmente entre Artigo Técnico ou Relato de Experiência. Artigos que não explicitavam esta característica foram lidos por completo para que pudessem ser classificados. Nesta etapa, 30 artigos foram excluídos do escopo por serem relatos de experiência, restando 44 artigos para análise, que foram lidos e avaliados como fundamentação ou citação ao MPS.BR. Caso o artigo faça apenas uma menção ao MPS.BR e seus modelos de referência, é classificado em citação, porém se existir um alinhamento entre o tópico da pesquisa e o MPS.BR, o artigo é classificado em fundamentação. Nesta etapa 6 artigos que não fundamentavam pesquisa no MPS.BR foram excluídos do escopo, restando 38 artigos para análise.

Para estes artigos, foram aplicados os procedimentos da GT descritos anteriormente. O grafo apresentado na Figura 1 representa a execução da primeira rodada em um dos artigos do escopo de pesquisa. O nó com código R105 é um resumo de artigo que busca responder parcialmente às questões de pesquisa Q1 e Q2. Os demais nós representam os códigos associados a trechos do artigo durante o procedimento de codificação aberta. A mensagem presente nesta figura é que o artigo R105 tem pesquisa fundamentada em 6 processos de diferentes níveis do MR-MPS-SW.

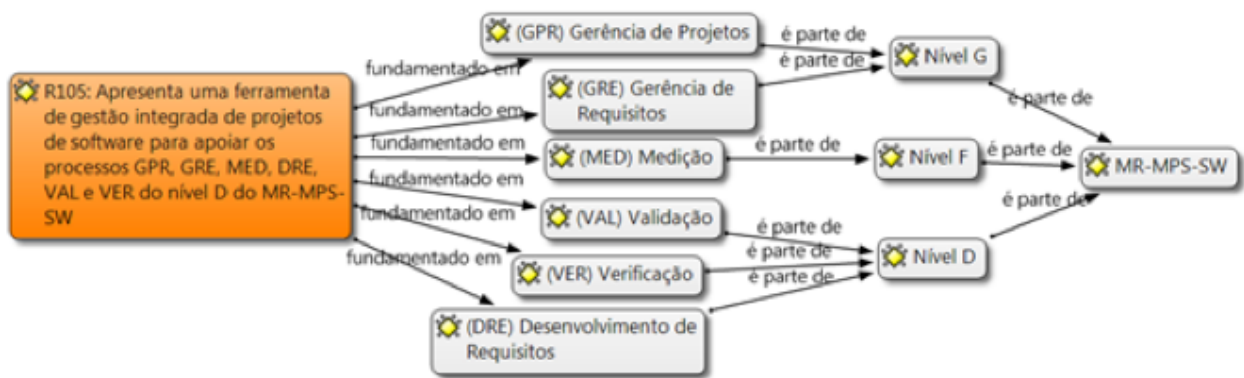


Figura 1: Grafo que representa a codificação aberta

A execução da primeira rodada de análise resultou na criação de 38 resumos e 27 códigos de propriedade. Antes da execução da segunda rodada de análise, foi realizado o processo de auditoria por pares, com objetivo de sanar toda e qualquer questão duvidosa sobre os achados da primeira rodada.

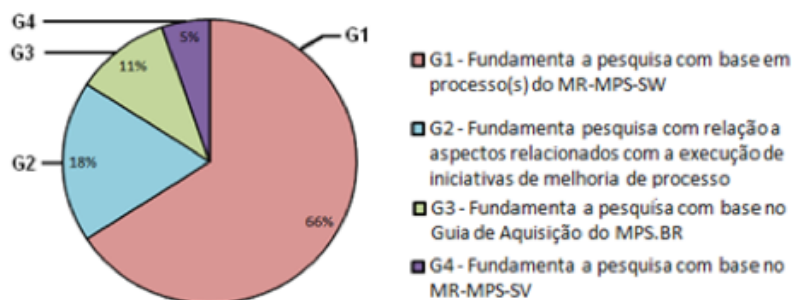
A segunda rodada de análise consistiu na aplicação do procedimento de codificação axial sobre os resumos dos artigos. Neste procedimento, os resumos foram submetidos a um processo de associação por semelhança. O grafo apresentado na Figura 2 exemplifica a abstração dos artigos R101, R105 e R116 à categoria F1. Ao fim desta rodada de análise, os 38 artigos foram abstraídos em 19 categorias.



Figura 2: Grafo que representa a codificação axial inicial realizada

Na terceira rodada de análise, as 19 categorias obtidas na rodada anterior foram associadas por semelhança, obtendo-se assim mais um nível de abstração. Nesta etapa de análise, os 19 códigos foram distribuídos em 4 grandes categorias assim definidas: G1 - Fundamenta a pesquisa com base em processos do MR-MPS-SW; G2 - Fundamenta pesquisa com relação a aspectos relacionados com a execução de iniciativas de melhoria de processo; G3 - Fundamenta a pesquisa com base no Guia de Aquisição do MPS.BR; e G4 - Fundamenta a pesquisa com base no MR-MPS-SV.

A execução da terceira rodada de análise completou o mapeamento dos artigos em categorias. Os códigos e categorias organizados e analisados de forma sistemática proveram informações essenciais para responder às duas questões de pesquisa propostas neste estudo. A partir do quantitativo de artigos classificados por grupo, foi traçado o gráfico sobre como o MPS.BR e seus modelos de referência têm influenciado na fundamentação das propostas de pesquisa (Figura 3). Neste gráfico, é possível observar que grande parte dos artigos técnicos publicados no WAMPS, nos últimos 5 anos, (66%) têm base em processos do MR-MPS-SW, enquanto artigos relacionados ao MR-MPS-SV foram menos frequentes (6%), o que é natural devido a este último ser mais recente. Uma parcela significativa (18%) dos artigos investiga fatores que envolvem a execução de MPS, tais como, fatores críticos de sucesso e benefícios da implementação de MPS, enquanto outros autores (11%) realizam estudos utilizando o Guia de Aquisição do MPS.BR como referência.



**Figura 3: Distribuição das Fundamentaões**

A Tabela 1 apresenta a distribuição das fundamentações por cada grupo e o quantitativo de artigos relacionados a cada fundamentação.

Diante da diversidade de assuntos relacionados às propostas de pesquisa, os achados apresentados nesta tabela apontam que no Grupo G1 um alto índice de fundamentações relacionadas à ferramentas de apoio à implementação de processos do MR-MPS-SW (F1, F2 e F4), que representam 44,7% do total de artigos. Há também autores que definem novos processos, técnicas ou abordagens com base no MR-MPS-SW (F5, F7, F8 e F9), 13,2% do total. O grupo G2 é composto por fundamentações que analisam a voz das empresas avaliadas e implementadores do MPS.BR para buscar entendimento sobre aspectos sociais da implementação de MR-MPS (F10, F11, F12, F13 e F14). O Guia de Aquisição foi utilizado como base para fundamentação em 4 artigos, distribuídos nas fundamentações F15, F16 e F17, estes artigos apontam o Guia de Aquisição como uma referência positiva e exemplar para contratação de serviços.

As informações presentes na Tabela 1 encontram-se em alto nível de abstração, logo, para identificar quais níveis e processos do MR-MPS-SW ou MR-MPS-SV são utilizados na fundamentação das publicações e quais métodos têm sido empregados para se obter informações sobre aspectos execução de iniciativas de melhoria de processo, os 4 grupos foram analisados separadamente.



**Tabela 1: Quantitativo e Distribuição das Fundamentações por Cada Grupo**

<b>Grupo</b>	<b>Fundamentações</b>	<b>Ocorrências</b>	<b>Artigos</b>
G1 - Fundamenta a pesquisa com base em processos do MR-MPS-SW	[F1] Apresenta uma ferramenta para apoiar a implementação de processos definidos no MR-MPS-SW	10	25
	[F2] Apresenta um ambiente para apoiar a implementação de processos definidos no MR-MPS-SW	5	
	[F3] Identifica o conhecimento necessário para implementação de processos do MR-MPS-SW	2	
	[F4] Avalia a aderência de ferramentas a processos do MR-MPS-SW	2	
	[F5] Propõe técnicas relacionadas à implementação de processos do MR-MPS-SW	2	
	[F6] Desenvolve jogos educativos alinhados a conceitos encontrados em resultados esperados de processos	1	
	[F7] Define abordagem para avaliar o potencial de organização com relação à implementação de processos do MR-MPS-SW	1	
	[F8] Propõe um novo processo com base no MR-MPS-SW	1	
	[F9] Define um processo de reutilização com base no Guia Implementação do MR-MPS-SW	1	
G2 - Fundamenta pesquisa com relação a aspectos relacionados com a execução de iniciativas de melhoria de processo	[F10] Os fatores críticos de sucesso na implementação de MPS são identificados	2	7
	[F11] Os benefícios de implementação de MPS são identificados	2	
	[F12] Os valores organizacionais relevantes na implementação de melhoria de processos de software são identificados	1	
	[F13] Os motivos de migração do MPS.BR para o CMMI são identificados	1	
	[F14] Os atributos de processo do Modelo MPS-SW são utilizados para a definição de padrões de processos.	1	
G3 - Fundamenta a pesquisa com base no Guia de Aquisição do MPS.BR	[F15] O Guia de Aquisição do MPS.BR é utilizado como referência para modelos de contratação de outras áreas.	2	4
	[F16] O Guia de Aquisição é base para a definição de um novo processo de aquisição.	1	
	[F17] Analisa fatores que podem impactar atividades descritas no Guia de Aquisição.	1	
G4 - Fundamenta a pesquisa com base no MR- MPS-SV	[F18] Os resultados esperados do nível G do MPS-SV são utilizados na fundamentação de abordagem para orientar implementação deste nível.	1	2
	[F19] Verifica aderência entre os modelos MPS-SW e o MPS-SV	1	
		<b>Total Geral</b>	<b>38</b>

### G1 - Fundamenta a pesquisa com base em processos do MR-MPS-SW

O grupo G1 obteve destaque pela quantidade de publicações relacionadas (25 publicações, em 38), sendo composto por artigos que fundamentam a pesquisa com base em processos do MR-MPS-SW. Grande parte das publicações deste grupo fundamenta a pesquisa em mais de um processo, como pode ser visto na Figura 4. É possível observar que em 2009 e 2010 havia uma tendência de artigos relacionados a processos específicos; em 2011 e 2012, o aumento de publicações sobre ferramentas integradas ambientes de apoio faz com que haja um aumento de publicações fundamentadas em 2 ou mais processos; este período é caracterizado por interesse no estudo da correlação entre os processos; já em 2013 há um maior equilíbrio entre os dois tipos de publicação.

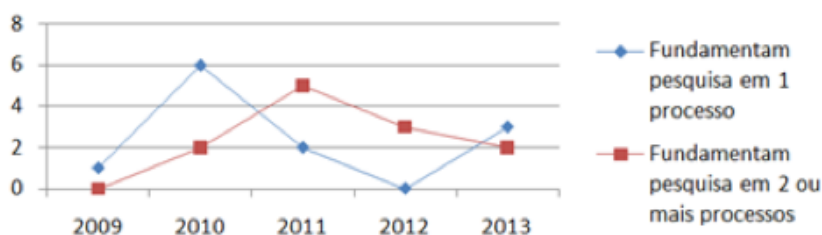


Figura 4: Fundamentações em Processos do MR-MPS-SW

A partir da análise das marcações realizadas durante a etapa de codificação aberta deste estudo, foi possível identificar os processos e níveis do MR-MPS-SW mais utilizados nas fundamentações, de modo a responder à questão de pesquisa Q2. A Figura 5 apresenta um resumo da distribuição das publicações pelos níveis do MR-MPS-SW, onde os níveis G e D ganham destaque. Importante observar que estes dois níveis apresentam processos compatíveis com atividades de um ciclo de vida básico, composto de Levantamento de Requisitos e Validação dos requisitos (GRE), Projeto e implementação (GPR) e Testes e Implantação (VER e VAL), esta correlação entre processos e o modelo de ciclo de vida básico reforça uma tendência de sobreposição de fundamentações relacionadas a estes dois níveis do MR-MPS-SW. A Tabela 2 apresenta as fundamentações por processos e por níveis.

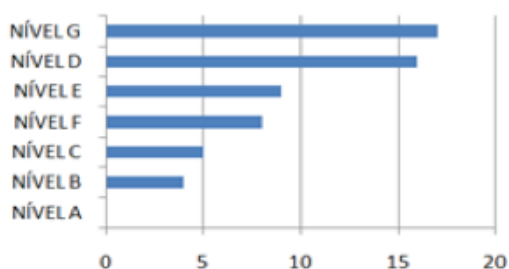


Figura 5: Fundamentações por Nível do MR-MPS-SW

**Tabela 2: Distribuição de Fundamentações por Processos e por Níveis**

<b>Níveis do MPS.BR</b>	<b>Processos</b>	<b>Ocorrências por processo</b>	<b>Ocorrências por nível</b>
Nível A - Em Otimização	Atributo de Processo 5.1 - AP 5.1	-	-
	Atributo de Processo 5.2 - AP 5.2	-	
Nível B - Gerenciado Quantitativamente	Atributo de Processo 4.1 - AP 4.1	2	4
	Atributo de Processo 4.2 - AP 4.2	2	
Nível C - Definido	Gerência de Riscos – GRI	2	5
	Gerência de Decisões – GDE	-	
	Desenvolvimento para Reutilização – DRU	3	
Nível D - Largamente Definido	Verificação – VER	8	16
	Validação – VAL	5	
	Projeto e Construção do Produto – PCP	-	
	Integração do Produto – ITP	-	
	Desenvolvimento de Requisitos – DRE	3	
Nível E - Parcialmente Definido	Gerência de Reutilização – GRU	5	9
	Gerência de Recursos Humanos – GRH	4	
	Definição do Processo Organizacional – DFP	-	
	Av. e Melhoria do Proc. Organizacional – AMP	-	
Nível F - Gerenciado	Medição – MED	4	8
	Gerência de Portfólio de Projetos – GPP	2	
	Garantia da Qualidade – GQA	-	
	Gerência de Configuração – GCO	2	
	Aquisição – AQU	-	
Nível G - Parcialmente Gerenciado	Gerência de Requisitos – GRE	10	17
	Gerência de Projetos – GPR	7	
		<b>Total Geral</b>	<b>59</b>

É possível observar que os processos GRE, GPR (nível G), VER e VAL (nível D) correspondem a mais da metade das fundamentações em processos do MR-MPS-SW, somando 30 artigos (50,8%), os demais 23 processos e atributos de processo de níveis A e B somam 29 artigos (49,2%). Diversos processos não foram utilizados em fundamentações de artigos e houve poucas fundamentações aos níveis mais altos do MPS.BR. Os níveis A e B somam 4 ocorrências, os níveis C, D e E somam 30, enquanto os níveis F e G somam 25. Esta diferença pode estar relacionada ao fato de apenas 1% das avaliações seja nos níveis mais altos (A e B) (SOFTEX, 2014).

## ***G2 - Fundamenta pesquisa com relação a aspectos relacionados com a execução de iniciativas de melhoria de processo***

Este grupo teve representação de 18% do total de publicações. Para o grupo em questão, foi relevante a realização de uma investigação sobre os métodos para realização da pesquisa. Assim, para os artigos que fundamentaram em F10, F12 e F13, a característica em comum foi que ambos aplicaram um *survey* em suas pesquisas para atingir seu objetivo. O objetivo das publicações deste subconjunto

foi desde identificar fatores críticos de sucesso e valores organizacionais relevantes na implementação de MPS, até apontar motivos que levam uma organização a migrar do MPS.BR para o CMMI. Ressalta-se que o público-alvo destes *surveys* foi diversificado em quatro tipos: (i) organizações que implementaram o MR-MPS-SW, (ii) implementadores e avaliadores do MR-MPS-SW, (iii) profissionais e pesquisadores indicados pela SOFTEX, e (iv) principais envolvidos de IIs (Instituições Implementadoras) e IOGEs (Instituições Organizadoras de Grupos de Empresas) credenciadas pela SOFTEX.

Além do *survey*, outro método de pesquisa que foi empregado em dois estudos deste grupo foi a análise qualitativa. Sendo que em um dos artigos buscou-se identificar os benefícios de implementação de MPS (F11), e o outro, já citado anteriormente, almejou identificar fatores críticos de sucesso. Outra fundamentação pertinente (F14) está relacionada a um estudo que propõe o uso de padrões de processo para organizar o conhecimento sobre atributos de processo em iniciativas de MPS.

### **G3 - Fundamenta a pesquisa com base no Guia de Aquisição do MPS.BR**

O Guia de Aquisição MPS.BR descreve, de forma detalhada e com exemplos práticos um processo para aquisição de Software e Serviços Correlatos (S&SC) e está em conformidade com a definição da norma internacional ISO/IEC 12207 (ISO/IEC 2008), sendo consistente com o MR-MPS-SW (SOFTEX, 2013).

No grupo G3, dois estudos (F15) utilizaram o Guia de Aquisição do MPS.BR como referência para modelos de contratação de serviços de TI no âmbito da Administração Pública Federal, o que demonstra influência do Programa MPS.BR também fora do contexto de organizações de software. Um deles utiliza características do Guia de Aquisição para sugerir melhorias na Instrução Normativa nº 4 (IN 4). Enquanto o outro propõe um modelo de processo de contratação de software e serviços de TI com ênfase em entes governamentais. Já em F16, uma publicação apresentou um mapeamento de processos de aquisição e serviços aderente às diretrizes do Guia de Aquisição. Por fim, a fundamentação (F17) está relacionada com um estudo em que é feita análise dos possíveis impactos e influências que os ECOSs (Ecosistemas de Software) provocam no processos de aquisição realizado pelas organizações.

### **G4 - Fundamenta a pesquisa com base no MR-MPS-SV**

Lançado em agosto de 2012, o Modelo de Referência MPS para Serviços (MR-MPS-SV) propõe que é essencial que os provedores de serviços coloquem a eficiência e a eficácia dos seus processos em foco nas empresas, visando à oferta de serviços conforme padrões internacionais de qualidade (SOFTEX, 2012b). Até setembro de 2014, apenas 5 organizações com avaliações aderentes ao modelo, sendo 4 no nível G e uma no nível F (SOFTEX, 2014).

No contexto do WAMPS, O MR-MPS-SV é fundamentado em dois estudos. Rodrigues e Souza (2013) apresentam uma abordagem para orientar a implementação do nível G com base na criação de uma Central de Serviços. Jordão e Kalinowski (2013) realizaram um *survey* com implementadores do MR-MPS-SW o qual identificou que a implantação do MR-MPS-SV em organizações que já adotam o MR-MPS-SW (i) é simples, pela cultura já estabelecida em função da adoção do MR-MPS-SV, (ii) traz benefícios em produtividade e qualidade e que (iii) o modelo deve ter ampla adoção.

### 3.1 Limitações

O protocolo de mapeamento sistemático da literatura aplicado neste estudo é resultado da adaptação do proposto em (SANTOS, 2011), uma vez que o contexto de investigação de ambos os estudos são distintos. Ainda que o escopo de mapeamento sistemático deste trabalho tenha identificado artigos técnicos cujos autores incluem membros da Academia, cuja pesquisa seja fundamentada no MPS.BR ou seus Modelos de Referência, não é possível afirmar que a motivação principal destas pesquisas esteja relacionada MPS.BR ou seus Modelos de Referência. Para isso seria necessária uma etapa de investigação junto aos autores dos trabalhos. O período de buscas se iniciou no ano em que o WAMPS passou a contar com um Coordenador Científico, no entanto, os trabalhos apresentados em edições anteriores, também contêm trabalhos oriundos de pesquisas acadêmicas. Devido a isso, para ter um panorama mais completo, as demais edições do WAMPS também deveriam ser considerados em extensões deste trabalho.

Este trabalho utilizou procedimentos de codificação e comparações constantes fundamentados em GT. Entretanto, não é objetivo desta pesquisa gerar uma teoria substantiva, chegar à saturação teórica, muito menos o estabelecimento de hipóteses. Como o procedimento de codificação permite múltiplas interpretações, foram realizadas auditorias com o objetivo de verificar se os conceitos gerados estavam relacionados a proposições descritas nos artigos e as categorias abstratas representavam a generalização das propriedades. Assim, os resultados obtidos devem ser considerados apenas como indícios de fundamentação e não como resultados conclusivos. Novos estudos, no entanto, devem ser realizados para estender os resultados e confirmar os achados apresentados neste trabalho.

## 4. Considerações Finais

Este artigo apresentou um estudo baseado em mapeamento sistemático e análise qualitativa, executada com base em Grounded Theory, com propósito de investigar que processos do MR-MPS-SW e do MR-MPS-SV foram utilizados em fundamentações de pesquisa, no contexto de publicações no WAMPS. Os resultados demonstraram um alto índice de publicações relacionadas a processos de níveis mais baixos do MR-MPS-SW, em especial, aos níveis G e D. Pôde-se constatar ainda, baixa quantidade de artigos relacionados aos níveis de alta maturidade do MR-MPS-SW e aos processos do MR-MPS-SV, o que pode ser um indício de que nesse contexto haja mais oportunidades futuras de pesquisa. Pôde-se constatar ainda que o Guia de Aquisição tem sido utilizado como referência fora do contexto de organizações de desenvolvimento de software. A contribuição deste trabalho é útil principalmente para a academia, pois após a aplicação do mapeamento sistemático e análise qualitativa, foi possível identificar quais níveis e processos do programa MPS.BR foram mais explorados no contexto dos trabalhos de pesquisa publicados no WAMPS.

Como trabalho futuro espera-se a extensão dos resultados com a análise dos artigos publicados no WAMPS em outras edições e também a inclusão de outros eventos patrocinados pela Sociedade Brasileira de Computação (SBC) nas áreas de Engenharia de Software, Qualidade de Software e Sistemas de Informação, como o Simpósio Brasileiro de Qualidade de Software (SBQS), o Simpósio Brasileiro de Engenharia de Software (SBES) e o Simpósio Brasileiro de Sistemas de Informação (SBSI).

## Agradecimentos

Os autores agradecem à FAPERJ (projeto E-26/110.438/2014) e à CAPES pelo apoio financeiro.

## Referências

- Bandeira-de-Mello, R., Cunha, C. (2003). "Operacionalizando o método da Grounded Theory nas Pesquisas em Estratégia: técnicas e procedimentos de análise com apoio do software ATLAS/TI". Encontro de Estudos em Estratégia. Curitiba
- Boehm, B., "A view of 20th and 21st century software engineering", Proceedings of the 28th International Conference on Software Engineering – ICSE 2006, Shanghai, China, pp.12-29, 2006
- Corbin, J. Strauss, A., 1998, "Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory". 2 ed. London, SAGE Publications
- ISO/IEC, 2008 – The International Organization for Standardization and the International Electrotechnical Commission. ISO/IEC 12207:2008 – Systems and software engineering – Software life cycle processes. Geneve: ISO, 2008.
- Jordão, L., Kalinowski, M. (2013). "Investigando a Aplicabilidade do MPS-SV na Melhoria de Serviços de Desenvolvimento e Manutenção de Software", In: IX Workshop Anual do MPS, Campinas, São Paulo.
- Mafra, S., Travassos, G. H., 2006, Estudos Primários e Secundários apoiando a busca por Evidência em Engenharia de Software, Relatório Técnico RT-ES 687/06, Programa de Engenharia de Sistemas e Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro.
- Petersen, K., Feldt, R., Mujtaba, S., Mattsson, M. -"Systematic Mapping Studies in Software Engineering", in: 12th International Conference on Evaluation and Assessment in Software Engineering, Australia, 2008.
- Rodrigues, R. G., Souza, A. S. (2013). "Implementando o Nível G do MR-MPS-SV com base no conceito de Central de Serviços", In: IX Workshop Anual do MPS, Campinas, São Paulo.
- Santos, G., 2011, "Influência e Impacto do Programa MPS.BR na Pesquisa Relacionada a Qualidade de Software no Brasil", SBQS 2011, Curitiba-PR, pág. 73-87.
- Santos, G.; Rocha, A. R.; Conte, T. ; Barcellos, M. P. ; Prikladnicki, R. . Strategic Alignment between Academy and Industry: A Virtuous Cycle to Promote Innovation in Technology. In: 26th Brazilian Symposium on Software Engineering (SBES), 2012, Natal. 2012 26th Brazilian Symposium on Software Engineering, 2012. p. 196-200.
- SOFTEX (2012). "MR-MPS-SW – Guia Geral MPS de Software". Disponível em: <http://www.softex.br/mpsbr/guias/>.
- SOFTEX (2013). "MPS.BR - Melhoria de Processo do Software Brasileiro - Guia de Aquisição". Disponível em: [www.softex.br/mpsbr](http://www.softex.br/mpsbr)
- SOFTEX (2014). "Avaliações MPS Publicadas". Disponível em <http://www.softex.br/mpsbr/avaliacoes/>, último acesso em 02/10/2014.
- Teixeira, V. S.; Delamaro, M. E. . Geração de Metadados para o Apoio ao Teste Estrutural de Componentes. VII Simpósio Brasileiro de Qualidade de Software, 2008, Florianópolis, Santa Catarina.



# Uma Investigação sobre a Gerência de Projetos de Desenvolvimento de Software em Órgãos do Governo do Ceará com Base no MPS-SW

Marum Simão Filho<sup>1,2</sup>, Rômulo F. Santos<sup>2</sup>, Adriano B. Albuquerque<sup>2</sup>

<sup>1</sup>Faculdade 7 de Setembro – FA7 – Fortaleza, CE – Brasil

<sup>2</sup>Universidade de Fortaleza – UNIFOR – Fortaleza, CE – Brasil

{marumsimao,romulo.frota.santos}@gmail.com, adrianoba@unifor.br

**Abstract.** *The consensus is that effective management is critical to the success of a project. Despite the vast amount of knowledge related to project management, the literature shows that problems similar to those that occurred in previous decades still happen. Unlike technical aspects, which are better treated, managerial problems manifest themselves recursively. This paper conducts a survey on Ceará's Government Agencies to estimate the adherence to the G level Project Management MPS-SW process and suggest that current problems still have high correlation compared with those found in projects in past decades.*

**Resumo.** *É consenso que uma gestão eficaz é fundamental para o sucesso de um projeto. Apesar da grande quantidade de conhecimento relacionado à gerência de projetos, a literatura mostra que ainda ocorrem problemas semelhantes àqueles ocorridos em décadas anteriores. Diferentemente de aspectos técnicos, que são mais bem tratados, os problemas gerenciais se manifestam de forma recorrente. Este trabalho realiza uma pesquisa em órgãos do Governo do Ceará para estimar a aderência ao nível G do processo de Gerência de Projetos do MPS-SW, além de sugerir que problemas atuais ainda mantêm elevada relação com aqueles encontrados em projetos em décadas passadas.*

## 1. Introdução

Há muito se argumenta sobre a importância da gerência de projetos na área de desenvolvimento de software. Processos de gerência de projetos foram estudados, criados e amplamente divulgados. Todo um corpo de conhecimento se estabeleceu em torno dessa relevante área, em especial, o que se conhece por PMBOK (*Project Management Body of Knowledge*) (PROJECT MANAGEMENT INSTITUTE, 2014).

Modelos de referência para o desenvolvimento de software, que abordam a gerência de projetos como um de seus processos, foram criados e amplamente difundidos, tais como CMMI (*Capability Maturity Model – Integration* ou Modelo de Maturidade em Capacidade – Integração) (CMMI INSTITUTE, 2014), em escala global, e MPS-BR (Melhoria de Processo do Software Brasileiro) (SOFTEX, 2012), em um contexto nacional.

As empresas, então, passaram a se organizar para seguir tais modelos de referência, buscando elevar o nível de qualidade dos projetos, e, por conseguinte, melhorar a qualidade dos softwares produzidos. Na esfera governamental, o mesmo cenário de busca pela qualidade no desenvolvimento de software se verifica. Os órgãos de governo estão cada vez mais buscando adaptar seus processos aos modelos



de referência. Porém, devido à grande quantidade de órgãos, à pluralidade de funções cobertas pelos órgãos e à descontinuidade de gestão, muitas vezes, não se tem o domínio preciso da situação da gerência de projetos de desenvolvimento de software frente aos modelos de referência.

O presente trabalho tem como objetivo realizar uma estimativa sobre o grau de aderência das unidades de desenvolvimento de software dos órgãos do Governo do Ceará em relação ao nível G do MPS-SW no que diz respeito à gerência de projetos de desenvolvimento de software. Este trabalho faz parte de um estudo realizado em conjunto pela Coordenadoria de Estratégias de Tecnologia da Informação e Comunicação da Secretaria do Planejamento e Gestão – SEPLAG – do Governo do Estado do Ceará e o Programa de Pesquisa e Pós-graduação em Informática Aplicada da Universidade de Fortaleza – PPGIA/UNIFOR.

O restante deste artigo é organizado como se segue. Na Seção 2, uma visão geral sobre falhas no gerenciamento de projetos e sobre o MPS-SW é fornecida. A pesquisa (*survey*) é descrita na Seção 3 e os resultados são analisados na Seção 4. Por fim, as considerações finais e sugestões de trabalhos futuros são fornecidas na Seção 5.

## 2. Referencial Teórico

### 2.1. Falhas em Projetos de Desenvolvimento de Software

A partir do relato do naufrágio do navio sueco *Vasa*, ocorrido em 1628, foi elaborado por Fairley e Willshire (2003) um estudo comparativo no qual se relatam os vários acontecimentos ocorridos durante a construção do navio, relacionando-os com a construção de software. A análise realizada pontuou, no que pesem as boas condições financeiras e conhecimentos técnicos presentes, uma variedade de equívocos relacionados principalmente ao gerenciamento do projeto.

O navio *Vasa* foi construído entre 1626 e 1628, por solicitação do rei sueco Gustav II Adolf. Durante a construção, foram solicitadas pelo rei inúmeras mudanças na especificação do navio, com a contínua pressão para o término do projeto em tempo exíguo. Foram relatados problemas de comunicação entre as equipes técnicas de construção e falta de documentação técnica. Os responsáveis se eximiram de alertar sobre os riscos iminentes de insucesso do projeto, possivelmente temendo contrariar os desejos do Rei, inclusive ignorando o resultado do teste de estabilidade do navio, que apontava defeitos na construção. Em agosto de 1628, o navio foi lançado, naufragando logo em seguida.

Os problemas ocorridos na construção do *Vasa* correspondem, em grande parte, aos mesmos que comumente ocorrem em projetos de desenvolvimento de software: cronograma apertado, necessidade de mudança no escopo, ausência de especificações técnicas e do plano do projeto, inovações excessivas e secundárias, baixo nível dos requerimentos, ausência de métodos científicos e comportamento antiético dos responsáveis pelo projeto.

Os motivos das falhas em projetos de desenvolvimento de software foram também analisados por Cerpa e Verner (2009), a partir de uma *survey* composta por 70 casos de insucesso, apontando os mais frequentes fatores de falhas. A pesquisa continha 88 questões cobrindo temas tais como análise de requisitos, estimativa e planejamento, processo de desenvolvimento e gerenciamento do projeto.

Foi constatado que as falhas de projeto são originadas por diversos fatores, os quais se relacionam uns com os outros, causando um escalonamento de problemas. Os projetos avaliados foram classificados em desenvolvimento interno (*in-house*) e terceirizado (*outsourced*). Foram identificados 18 fatores de insucesso, sendo os mais frequentes: prazo de entrega geralmente curto impactando no processo de software (citado em 92,9% dos projetos), projeto subestimado (81,4%), riscos não gerenciados ou controlados durante o projeto (75,7%), equipe não recompensada devidamente (74,3%) e prazos contratados sem as informações adequadas dos requisitos (72,9%).

Os fatores encontrados estavam todos relacionados com a gerência de projetos, embora alguns estivessem parcialmente ou totalmente fora de controle do gerente do projeto, tais como as recompensas dos funcionários ou o tempo de disponibilidade dos usuários para o levantamento de requisitos.

Por meio dos estudos realizados constata-se que, apesar da evolução da engenharia de software e da gerência de projetos, os projetos ainda falham em grande parte devido ao gerenciamento inadequado dos projetos, tornando-se fundamental que sejam inseridas e aperfeiçoadas boas práticas referentes ao assunto.

## 2.2. MPS.BR e Gerência de Projetos

O MPS.BR é um modelo de qualidade de processo baseado nas normas ISO/IEC 12207 e ISO/IEC 15504, criado em 2003 pela SOFTEX objetivando melhorar a capacidade de desenvolvimento de software nas empresas brasileiras. É compatível com o CMMI, um modelo de referência internacional que contém práticas necessárias à maturidade em disciplinas específicas relacionadas à engenharia de software (SOFTEX, 2012).

O Modelo de Referência MPS para Software (MR-MPS-SW) define níveis de maturidade que são uma combinação entre processos e sua capacidade. A escala de maturidade se inicia no nível G e progride até o nível A. Para cada um destes sete níveis de maturidade, é atribuído um perfil de processos que indicam onde a organização deve colocar o esforço de melhoria (SOFTEX, 2012). O nível G do MPS-SW, denominado Parcialmente Gerenciado, contemplado neste trabalho, abrange dois processos: a Gerência de Requisitos – GRE e a Gerência de Projetos – GPR.

O propósito do processo Gerência de Projetos é estabelecer e manter planos que definem as atividades, recursos e responsabilidades do projeto, bem como prover informações sobre o andamento do projeto que permitam a realização de correções quando houver desvios significativos no desempenho do projeto (SOFTEX, 2012).

## 3. A Pesquisa

A pesquisa foi conduzida na administração pública do Estado do Ceará, que é constituída por diversos órgãos e entidades, divididos nas duas formas de atuação: administração direta, com as secretarias de governo, e administração indireta, com as autarquias, fundações públicas, empresas públicas e sociedades de economia mista.

Tais órgãos e entidades possuem relativa independência, infraestrutura tecnológica, recursos humanos e financeiros próprios para realizar a contratação, desenvolvimento, implantação e operacionalização de softwares aplicativos, visando dar suporte aos seus negócios.

### 3.1. Planejamento

O objetivo da pesquisa foi realizar uma estimativa sobre o grau de aderência dos órgãos do Governo do Ceará em relação ao nível G do MPS-SW no que diz respeito à gerência de projetos de desenvolvimento de software.

Inicialmente, buscou-se apoio da Coordenadoria de Tecnologia da Informação do Governo do Estado, uma vez que tal entidade concentra as ações estratégicas referentes à Tecnologia da Informação nos diversos órgãos do Governo estadual.

Preparou-se uma palestra de sensibilização de cerca de 2h para a qual foram convidados os representantes e gestores de Tecnologia da Informação (TI) de cerca de 70 órgãos do Governo nos quais existe um departamento ou setor de TI.

Foi elaborado um questionário com perguntas divididas em duas categorias: i) caracterização do respondente e do órgão, e ii) implementação ou não do processo de gerência de projetos do MPS-SW no órgão. Na categoria ii, o que se esperava obter era uma estimativa se, no entendimento de cada respondente, o resultado esperado estava ou não implementado. Além disso, para o caso da não implementação de algum resultado esperado, disponibilizou-se espaço para indicação das principais dificuldades para a não implementação daquele resultado esperado no órgão.

Após a sensibilização, o formulário da pesquisa foi publicado para preenchimento pela Internet por meio da ferramenta *LimeSurvey* (LIMESURVEY, 2014). As 3 perguntas iniciais do questionário visavam caracterizar o órgão e foram elaboradas conforme mostra o Quadro 1 a seguir:

**Quadro 1 – Questões de Caracterização do Perfil do Órgão**

1) Número de Empregados do órgão? <input type="checkbox"/> Até 19 <input type="checkbox"/> De 20 a 99 <input type="checkbox"/> De 100 a 499 <input type="checkbox"/> Acima de 500
2) Tempo de operação do órgão? <input type="checkbox"/> até 3 anos <input type="checkbox"/> de 3 anos a 6 anos <input type="checkbox"/> de 6 anos a 9 anos <input type="checkbox"/> acima de 9 anos
3) O órgão possui certificações (pode selecionar mais de uma)? <input type="checkbox"/> CMMI – Nível: _____ <input type="checkbox"/> MPS-BR – Nível: _____ <input type="checkbox"/> ISO 9001 <input type="checkbox"/> Outra Certificação: _____ <input type="checkbox"/> Não Possui

As 4 perguntas seguintes buscavam obter o perfil do avaliador e foram redigidas conforme segue:

**Quadro 2 – Questões de Caracterização do Perfil do Respondente**

4) Selecione seu cargo/função no órgão: <input type="checkbox"/> Gerente de TI <input type="checkbox"/> Gerente de Projeto <input type="checkbox"/> Analista de Sistemas/Desenvolvedor <input type="checkbox"/> Analista de Infraestrutura
5) Qual seu tempo de experiência na função atual? <input type="checkbox"/> até 3 anos <input type="checkbox"/> de 3 anos a 6 anos <input type="checkbox"/> de 6 anos a 9 anos <input type="checkbox"/> Acima de 9 anos
6) Você possui alguma certificação (pode selecionar mais de um)? <input type="checkbox"/> Não Possui <input type="checkbox"/> PMI <input type="checkbox"/> Microsoft <input type="checkbox"/> IBM <input type="checkbox"/> Sun/Oracle <input type="checkbox"/> Outra Certificação: _____
7) Qual seu nível de escolaridade? <input type="checkbox"/> 2º. Grau ou Nível Médio <input type="checkbox"/> Sup. Incompleto <input type="checkbox"/> Sup. Completo <input type="checkbox"/> Especialização <input type="checkbox"/> Mestrado <input type="checkbox"/> Doutorado

As 19 perguntas seguintes estavam relacionadas aos resultados esperados da área de Gerência de Projetos do MPS-SW voltados ao nível G e continham o conteúdo apresentado no Quadro 3, com respostas do tipo “Implementado/Não implementado” e espaço para informação sobre dificuldades no caso de não implementação.

### Quadro 3 – Questões Relacionadas à Gerência de Projetos no MPS-SW

- Q1) O escopo do trabalho para o projeto é definido? Entende-se por escopo do projeto o trabalho que precisa ser realizado para entregar um produto, serviço ou resultado com as características e funções especificadas.
- Q2) As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados? O tamanho e/ou a complexidade das tarefas e dos artefatos gerados no projeto são estimados utilizando métodos adequados (ex: baseados na EAP ou estrutura equivalente, em técnicas de estimativa ou em dados históricos)?
- Q3) O modelo e as fases do ciclo de vida do projeto são definidos? O modelo do ciclo de vida do projeto é definido, indicando suas fases, as relações de sequência e interdependência entre elas?
- Q4) O esforço e o custo para a execução das tarefas e dos produtos de trabalho são estimados com base em dados históricos ou referências técnicas? São realizadas estimativas de custo e esforço para tarefas e produtos de trabalho com base em dados históricos ou métodos de estimativas e que foram documentadas as suas justificativas?
- Q5) O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos? (i) O orçamento e o cronograma são definidos, revistos e atualizados ao longo do desenvolvimento, conforme necessário? (ii) O cronograma possui marcos e/ou pontos de controle? (iii) O cronograma estabelece as dependências entre tarefas?
- Q6) Os riscos do projeto são identificados e o seu impacto, probabilidade de ocorrência e prioridade de tratamento são determinados e documentados? (i) Existe uma lista dos riscos identificados para o projeto? (ii) É realizada uma análise para determinar a probabilidade, o impacto, o grau de importância (exposição) e a prioridade de cada risco?
- Q7) Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo? (i) A equipe do projeto é selecionada a partir das competências requeridas para realizar as atividades do projeto e considerando o perfil dos candidatos? (ii) É planejado treinamento, quando necessário?
- Q8) Os recursos e o ambiente de trabalho necessários para executar o projeto são planejados? São planejados os recursos e o ambiente de trabalho necessários? (obs: aqui trata-se de outros recursos que não recursos humanos).
- Q9) Os dados relevantes do projeto são identificados e planejados quanto à forma de coleta, armazenamento e distribuição? Um mecanismo é estabelecido para acessá-los, incluindo, se pertinente, questões de privacidade e segurança? Existe um plano para gerência de dados, que relacione todos os documentos gerados no projeto, sua distribuição, mídia para armazenamento, forma de proteção (segurança e sigilo) e recuperação dos dados?
- Q10) Um plano geral para a execução do projeto é estabelecido com a integração de planos específicos? As informações de planejamento do projeto são documentadas, organizadas e relacionadas entre si, de forma a comporem o plano de projeto?
- Q11) A viabilidade de atingir as metas do projeto é explicitamente avaliada considerando restrições e recursos disponíveis? Se necessário, ajustes são realizados? A viabilidade do projeto é avaliada de forma explícita após a elaboração do plano do projeto, e considerando critérios como os objetivos do projeto, os recursos financeiros, técnicos, humanos, bem como das restrições impostas pelo cliente?

Q12) O Plano do Projeto é revisado com todos os interessados e o compromisso com ele é obtido e mantido? (i) Todos os interessados tomam conhecimento, revisam e se comprometem com o planejamento do projeto? (ii) Há recomprometimento conforme necessário?

Q13) O escopo, as tarefas, as estimativas, o orçamento e o cronograma do projeto são monitorados em relação ao planejado? O projeto é monitorado ao longo de seu ciclo de vida, comparando o planejado e o realizado em relação ao escopo, prazo, esforço, custos e cronograma?

Q14) Os recursos materiais e humanos bem como os dados relevantes do projeto são monitorados em relação ao planejado? O projeto é monitorado ao longo de seu ciclo de vida, comparando o planejado e o realizado em relação ao escopo aos recursos materiais, humanos e em relação aos dados relevantes do projeto?

Q15) Os riscos são monitorados em relação ao planejado? O projeto é monitorado ao longo de seu ciclo de vida, comparando o planejado e o realizado em relação aos riscos?

Q16) O envolvimento das partes interessadas no projeto é planejado, monitorado e mantido? O que é planejado em relação ao envolvimento das partes interessadas é monitorado e os compromissos assumidos são cumpridos ou negociados?

Q17) Revisões são realizadas em marcos do projeto e conforme estabelecido no planejamento? Ocorrem revisões nos marcos do projeto e em outros pontos estabelecidos no planejamento, que complementam o acompanhamento do dia-a-dia com uma visão mais ampla e abrangente do projeto?

Q18) Registros de problemas identificados e o resultado da análise de questões pertinentes, incluindo dependências críticas, são estabelecidos e tratados com as partes interessadas? Existem registros de identificação e análise dos problemas ocorridos no projeto e estes problemas são tratados com os interessados?

Q19) Ações para corrigir desvios em relação ao planejado e para prevenir a repetição dos problemas identificados são estabelecidas, implementadas e acompanhadas até a sua conclusão? (i) Na monitoração do projeto são identificadas ações corretivas, tanto para corrigir desvios em relação ao planejado, quanto para prevenir a repetição dos problemas identificados? (ii) Estas ações são acompanhadas e investigadas quanto à efetividade, antes de serem consideradas concluídas? (iii) Os problemas e as ações corretivas são repassados para níveis hierárquicos superiores, quando necessário, para garantir sua conclusão?

### 3.2. Execução

Conforme planejado, a palestra de sensibilização foi realizada com boa parte dos representantes dos 70 órgãos do Governo, além de gerentes de projetos e outros profissionais convidados. Depois, os gestores de TI dos órgãos convidados tiveram um prazo de 15 dias para preenchimento dos questionários. Após preenchimento, os dados foram tabulados pela equipe da Coordenadoria de Tecnologia da Informação do Governo do Estado e enviados aos pesquisadores para realização de análises.

Dos 70 órgãos convidados, um total de 25 gestores preencheu o questionário. Desses, dois órgãos foram excluídos por não realizarem desenvolvimento de software. Assim, o número final de respondentes resumiu-se a 23 órgãos, aproximadamente 33% do total. O grupo dos respondentes é bastante diverso, envolvendo setores como agricultura, água e esgoto, turismo, infraestrutura e educação, dentre outros, o que não permitiu classificações mais elaboradas sobre áreas ou setores. Por fim, as respostas foram analisadas pelos pesquisadores, gráficos foram gerados e conclusões foram desenvolvidas, as quais serão fornecidas na seção seguinte.

## 4. Análise dos Resultados

Foram realizados dois tipos de análises, uma com caráter quantitativo, que considerou as respostas “implementado” e “não implementado”, e outra que explorou as justificativas de não implementação dos resultados esperados por meio de uma abordagem qualitativa.

### 4.1. Análise Quantitativa

O perfil dos órgãos é traçado conforme a Figura 1. Em relação ao número de empregados, a maioria dos órgãos (70%) tem mais de 100 empregados e apenas 4% possuem menos de 20 empregados (Figura 1.a). Quanto ao tempo de operação do órgão, a grande maioria, cerca de 78%, possui mais de 9 anos de operação (Figura 1.b). No que diz respeito às certificações que o órgão possui, a maioria esmagadora (91%) não possui qualquer certificação. 9% dos órgãos são certificadas ISO 9001, mas não se verificou nenhum órgão com certificação na área de software (Figura 1.c).

O perfil dos respondentes é representado conforme ilustra a Figura 2. No que diz respeito ao cargo/ função ocupado pelo respondente no órgão, a maioria é gerente de TI (78%) (Figura 2.a). Em relação ao tempo de experiência na função, cerca de 65% possui mais de 6 anos na função (Figura 2.b). Quanto às certificações do respondente, a grande maioria, 74%, não possui qualquer certificação. Apenas 4% possui certificação PMI em gerência de projetos (Figura 2.c). Em relação ao nível de escolaridade, chamou atenção o número de pós-graduados: 44% são especialistas e 9% são mestres (Figura 2.d).

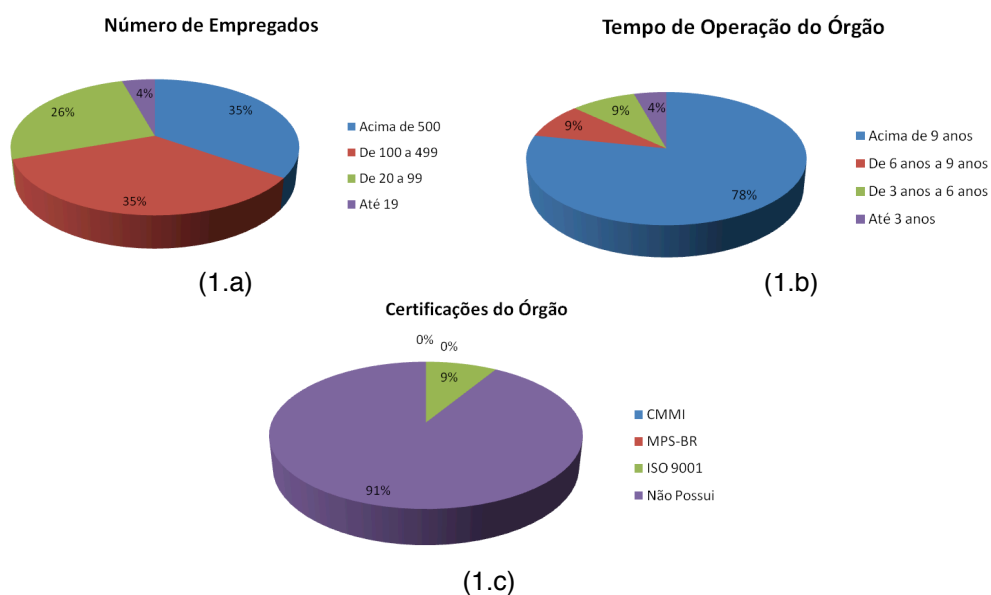
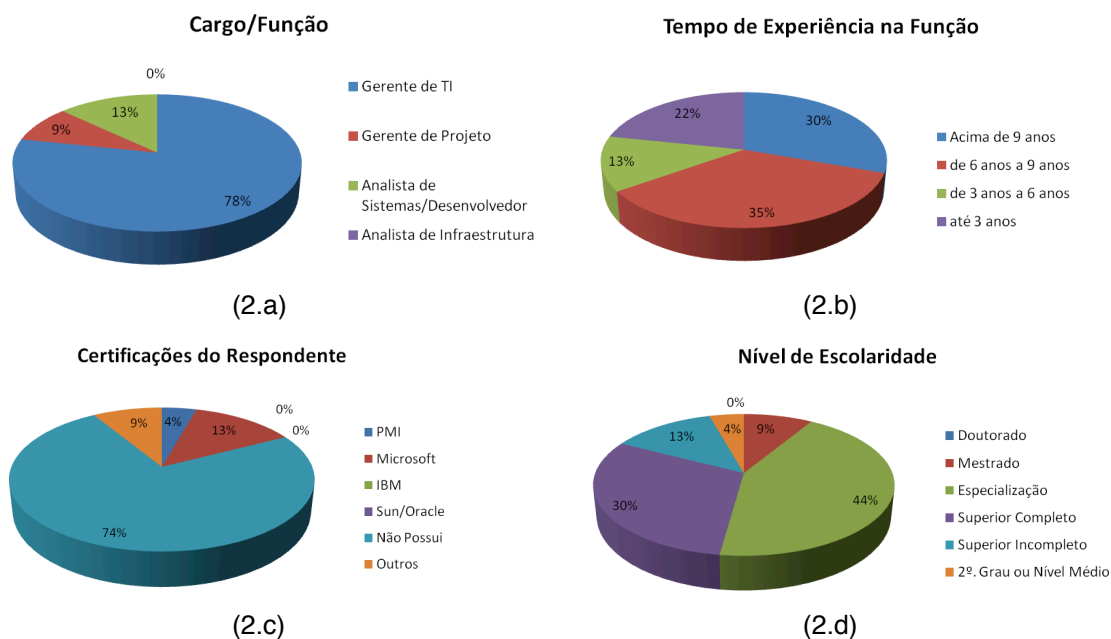


Figura 1. Perfil dos Órgãos Participantes

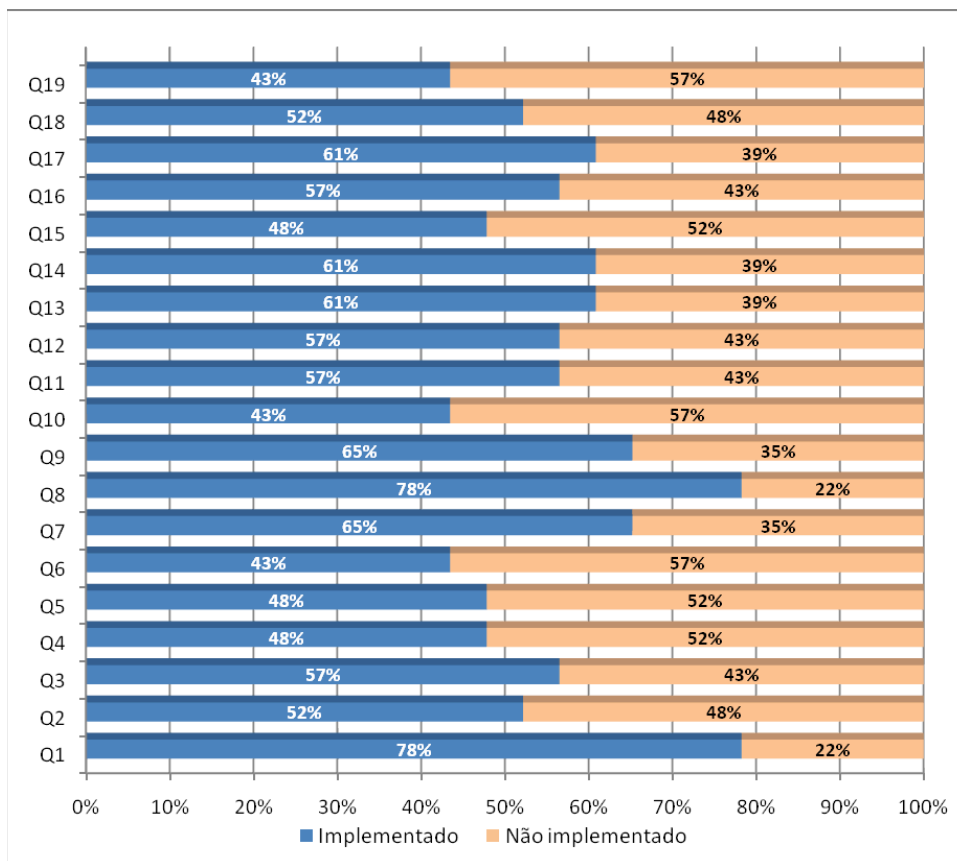


**Figura 2. Perfil dos Respondentes**

As respostas referentes à estimativa de implementação ou não dos resultados esperados do MPS-SW no órgão, que retrata o grau de aderência do processo de gestão de projetos do órgão ao MPS-SW, nível G, estão representadas em forma de gráfico na Figura 3.

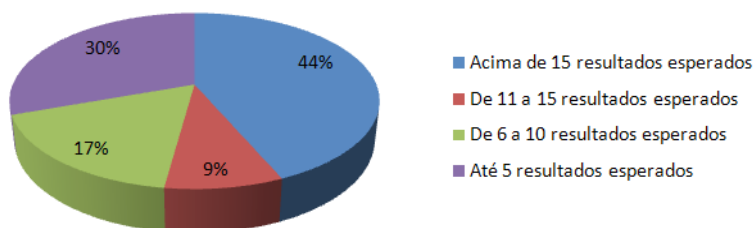
No geral, verificou-se equilíbrio entre os órgãos que estimam implementar e aqueles que estimam não implementar os resultados esperados do processo de gestão de projetos do MPS-SW, nível G. Dois resultados esperados chamaram a atenção em relação ao bom número de respostas positivas, com 78% de implementados cada: aquele que trata da definição do escopo do trabalho (Q1) e aquele referente ao planejamento dos recursos e do ambiente de trabalho (Q8).

Três resultados esperados obtiveram os maiores percentuais de não implementação, com 57% de respostas negativas cada: o resultado esperado relacionado à identificação e qualificação dos riscos (Q6), aquele referente à integração dos planos específicos em um plano geral (Q10), e, finalmente, aquele associado ao monitoramento dos desvios em relação ao planejado (Q19).



**Figura 3. Estimativa de Implementação dos Resultados Esperados do Nível G do Processo de Gestão de Projetos do MPS-SW nos Órgãos do Governo do Ceará**

Os números foram analisados também sob a ótica da quantidade de processos implementados por órgão e estão refletidos na Figura 4. Conforme pode ser visualizado no gráfico, um bom número de órgãos, ou seja, 44%, estima implementar mais de 15 processos, e apenas 9% delas respondeu implementar até 5 processos do nível G.



**Figura 4. Quantidade de Resultados Esperados do Nível G do Processo de Gestão de Projetos do MPS-SW Implementados (Estimativa) nos Órgãos do Governo do Ceará**

Dos órgãos analisados, 4 deles, ou seja, 17,4%, estariam totalmente aderentes ao nível G do modelo MPS-SW, isto é, estimaram implementar os 19 resultados esperados. Outros 4 órgãos, representando os mesmos 17,4%, estimaram não implementar nenhum dos processos analisados, estando, portanto, com nenhuma aderência ao nível G do modelo MPS-SW.



## 4.2. Análise Qualitativa

Para realização da análise qualitativa, primeiramente, foram identificados códigos, ou seja, respostas com semântica comum, porém, redigidos de forma diferente, que se repetiam com frequência nos textos das justificativas. Assim, foram identificados os seguintes códigos: falta de pessoal, ausência de uma unidade (setor/departamento) de gerência de projetos, pouco envolvimento/comprometimento das partes e ausência/deficiência de processo/metodologia. As respostas, então, foram classificadas de acordo com alguns desses códigos. Convém esclarecer que houve outras respostas, porém, não foi possível categorizá-las nos códigos adotados neste trabalho.

A falta de pessoal foi a justificativa mais alegada pelos órgãos para a não implementação dos resultados esperados do nível G do processo de Gerência de Projetos, alcançando a impressionante marca de 131 respostas. Comparando com o trabalho de Cerpa e Verner (2009), 63% dos projetos que falharam apontaram esse fator (“projeto não tinha equipe adequada para cumprir o cronograma”, tradução nossa) como uma das causas do fracasso. O segundo motivo mais citado foi ausência/deficiência de processo/metodologia, aparecendo em 39 justificativas. Em Cerpa e Verner (2009), esse fator (“metodologia de desenvolvimento era inapropriada para o projeto”, tradução nossa) apareceu em 71,4% dos projetos que falharam.

O pouco envolvimento/comprometimento das partes foi citado 7 vezes. Associando com o trabalho de Cerpa e Verner (2009), essa categoria pode ser relacionada a alguns fatores, como “os clientes/usuários não estão envolvidos no estabelecimento das estimativas de cronograma” (tradução nossa), com 69,4%, “clientes/usuários tinham expectativas irrealistas” (tradução nossa), também com 69,4%, e “os clientes/usuários não dedicam tempo suficiente para o levantamento de requisitos” (tradução nossa), com 61,2%.

A ausência de uma unidade de gerência de projetos apareceu em 5 justificativas. Não foi possível fazer uma correlação direta desse aspecto com algum fator específico de Cerpa e Verner (2009).

Em algumas justificativas, percebeu-se certa confusão no entendimento do questionário, quando o respondente relata que “é definido e registrado em documentos internos, contudo não implementado conforme orienta o PMBOK”.

Outras justificativas também alertaram para o elevado volume de demandas e a necessidade de rápido atendimento, não sendo possível se dedicar às atividades de gerência de projetos. Por fim, percebeu-se que, em alguns casos, dependendo da relevância do projeto, este pode ser conduzido de uma forma mais aderente ao MPS-SW ou não.

## 5. Considerações Finais e Trabalhos Futuros

Este trabalho descreveu uma pesquisa visando estimar a aderência de unidades de desenvolvimento de software de órgãos do Governo do Estado do Ceará ao nível G do processo de Gerência de Projetos do MPS-SW. Os questionários foram encaminhados aos gerentes das unidades de TI de 70 órgãos, atingindo um percentual de 33% de respondentes.

De forma geral, verificou-se certo equilíbrio entre os órgãos que estimam implementar e os que estimam não implementar os processos do MPS-SW. O resultado esperado que trata da definição do escopo do trabalho e o resultado esperado referente ao planejamento dos recursos e do ambiente de trabalho são os mais implementados, segundo avaliação dos respondentes. Os resultados esperados relacionados à identificação e qualificação dos riscos, à integração dos planos específicos em um plano geral e ao monitoramento dos desvios em relação ao planejado obtiveram os maiores percentuais de não implementação, nas estimativas dos respondentes.

Um bom número de órgãos (44%) estima implementar mais de 15 resultados esperados. Apenas 9% responderam implementar até 5 resultados esperados. 17,4% dos órgãos estariam totalmente aderentes e outros 17,4% estariam totalmente não aderentes.

Foi possível constatar também que muitas das dificuldades mencionadas pelos respondentes estão associadas a fatores de falhas em projetos de software tipicamente relatados em estudos de referência, como Cerpa e Verner (2009), o que sugere que problemas atuais ainda mantêm elevada relação com aqueles identificados em projetos em décadas passadas.

Como trabalhos futuros, deseja-se ampliar a aplicação da pesquisa, buscando alcançar um número mais abrangente de órgãos respondentes. Também se pretende aplicar o método da *Ground Theory* para mapear, de forma mais apropriada, os conceitos, suas causas e suas relações. Sugere-se ainda a realização de estudos semelhantes em órgãos de outras esferas do Governo ou mesmo de outros Estados visando identificar similaridades e diferenças entre os resultados.

## Referências

- CERPA, Narciso; VERNER, June M. Why did your project fail?. *Communications of the ACM*, v. 52, n. 12, p. 130-134, 2009.
- CMMI INSTITUTE. CMMI - Capability Maturity Model – Integration. Disponível em [www.cmmiinstitute.com](http://www.cmmiinstitute.com). Acessado em 14/07/2014.
- FAIRLEY, Richard E.; WILLSHIRE, Mary Jane. Why the Vasa sank: 10 problems and some antidotes for software projects. *Software, IEEE*, v. 20, n. 2, p. 18-25, 2003.
- LIMESURVEY Partners. Disponível em <http://www.limesurvey.com/>. Acesso em: 25/07/2014.
- PROJECT MANAGEMENT INSTITUTE. Um Guia do Conhecimento em Gerenciamento de Projetos - Guia PMBOK, Saraiva, 5ª Ed. 2014.
- SOFTEX. MPS.BR - Melhoria de Processo do Software Brasileiro – Guia Geral MPS de Software (2012) Disponível em [www.softex.br/mpsbr](http://www.softex.br/mpsbr). Acesso em 2014.



# Mapeamento Sistemático sobre Métricas no Contexto de Métodos Ágeis aplicadas a Teste de Software

Thaynã Gonçalves Mota, Arilo Claudio Dias Neto

Instituto de Computação – Universidade Federal do Amazonas (UFAM)  
Av. Rodrigo Otávio, 3000 – Campus Universitário– 69077-000– Coroado I

Manaus – AM – Brasil

{thayna,arilo}@icomp.ufam.edu.br

**Abstract.** *Software metrics are used to verify project status, progress and point out improvements and difficulties in the development process. Software projects developed according to the values published in the agile manifesto have characteristics and practices aimed at accelerating, for example, the delivery of working software to the customer. To guarantee the quality of the developed product, testing is used. This paper presents the results of a systematic mapping that intended to identify, analyze and evaluate studies on metrics used in software projects developed with agile methods and that can be applied in the testing process. The steps performed to carry out the mapping and the results are presented, the measures identified are interpreted in accordance with the role of each one in the monitoring and project management process.*

**Resumo.** *Métricas de software são utilizadas para verificar o status do projeto, o progresso e apontar melhorias e dificuldades no processo de desenvolvimento. Projetos de software desenvolvidos segundo os valores publicados no manifesto ágil apresentam características e práticas que visam acelerar, por exemplo, a entrega de software funcionando ao cliente. Para garantir a qualidade do produto oferecido, utiliza-se a realização de testes. Este artigo apresenta os resultados de um mapeamento sistemático que visou identificar, analisar e avaliar estudos sobre métricas utilizadas em projetos de software desenvolvidos com métodos ágeis e que podem ser aplicadas no processo de teste. São apresentados os passos conduzidos para a realização do mapeamento e os resultados, analisando as métricas identificadas de acordo com o papel de cada uma no monitoramento e gerenciamento destes projetos.*

## 1. Introdução

Em métodos ágeis (métodos adaptativos), as atividades de teste devem elas ocorrem de forma frequente durante todo o ciclo de desenvolvimento. Desta forma, contextualizando a atividade de teste de software para este cenário, o que é chamado de **Testes Ágeis**, percebe-se que esta, além de possuir grande importância para o projeto, ocorre de forma frequente, procurando detectar defeitos o mais cedo possível em ciclos de desenvolvimento iterativos e curtos (VICENTE, 2010).

Segundo Vicente (2010), uma das práticas que apoia a melhoria contínua de processos de software é a coleta de dados sobre o andamento do projeto por meio de métricas de software. Através de métricas, pode-se medir o progresso e apontar melhorias e dificuldades durante todo o projeto. Desde a década de 80, há uma preocupação de profissionais da área de software em utilizar métricas para auxiliar no gerenciamento e apontar melhorias no processo de desenvolvimento. Hartmann e

Dymond (2006) afirmam que algumas métricas utilizadas em projetos desenvolvidos com métodos preditivos são inadequadas para projetos que seguem princípios ágeis, o que pode ocasionar o desperdício de recursos e ameaçar a integridade de uma cultura ágil emergente. Desta forma, os autores sugerem que as métricas utilizadas em projetos desenvolvidos com métodos ágeis reforcem os princípios ágeis de colaboração com o cliente, entrega de valor, simplicidade e não se preocupem com os números gerados, e sim com as tendências demonstradas.

No contexto do processo de testes de software, as métricas permitem fornecer um *feedback* instantâneo a respeito do software que está sendo desenvolvido e testado [Crispin e Gregory 2009]. Medições devem ser planejadas e coletadas de forma organizada, para que possam ser monitoradas e ajustadas de acordo com os objetivos e necessidades da organização. No cenário de teste de software, as métricas obtidas e interpretadas durante o processo de teste devem ser igualmente planejadas, pois é necessário obter dados confiáveis para que possam prover informações para a melhora contínua do projeto.

Segundo o *TMMi Foundation* (2012), a atividade de teste de software geralmente é responsável por cerca de 30% a 40% dos custos de um projeto, o que motiva a aplicação de estratégias de medição que visam gerenciar estes recursos e garantir a qualidade e melhoria constante do processo de testes. Atualmente, existem vários *frameworks* (normas e modelos de maturidade) de apoio à implantação de melhoria de processos em organizações desenvolvedoras de software que incluem práticas de medição e análise. Alguns são abrangentes a todo o cenário de engenharia de software (ex: normas ISO, CMMI e MPS.BR) e outros são específicos ao cenário de testes (ex: TMMI, TPI e MPT.BR). Todos abordam medição como prática a ser implantada nas organizações, porém não citam as métricas a serem implantadas e seus contextos de aplicação.

Assim, este trabalho propõe um mapeamento sistemático que visa identificar e caracterizar métricas utilizadas em projetos desenvolvidos com métodos ágeis que pudessem ser aplicadas ao contexto de teste de software. O objetivo é oferecer uma fonte de informações quantitativas que pudessem ser utilizadas na indústria para tornar o processo de teste mais gerenciável, eficiente e otimizado, de forma a auxiliar a equipe de teste no acompanhamento das atividades.

Este artigo está organizado da seguinte forma: além desta introdução, é formado por outras três seções. A Seção 2 aborda o referencial teórico e trabalhos relacionados. A Seção 3 apresenta o mapeamento sistemático desenvolvido neste trabalho com o objetivo de identificar e caracterizar métricas relacionadas a testes ágeis, seu planejamento e resultados. Por fim, as considerações finais e trabalhos futuros são discutidos na Seção 4.

## 2. Referencial teórico

### 2.1. Medições e Modelos de Processos de Software

De acordo com o *Capability Maturity Model Integration* – CMMI [SEI2010], “a qualidade de um sistema ou produto é altamente influenciada pela qualidade do processo utilizado”, e uma das formas mais eficientes para avaliação e melhoria da qualidade de processos e produtos de software é por meio da aplicação de medição e análise [Rocha *et al.* 2012].

Como afirmado por Mahnic e Vrana (2007), apesar de todos os esforços para melhorar o processo de software através de modelos de qualidade, ainda existe uma grande quantidade de projetos que fracassam. Pesquisas com mais de 8000 projetos apontam que a maioria das falhas de projeto envolvem problemas de pessoal e de gerenciamento do projeto, ao invés de problemas técnicos [Ceschi *et al.* 2005]. Com o objetivo de tentar amenizar os gargalos de gerenciamento, melhorar e refinar cada vez mais o processo de software, pode-se introduzir o uso de métricas no dia a dia do projeto.

Atualmente, existem vários *frameworks* (normas e modelos de maturidade) de apoio à implantação de melhoria de processos em organizações desenvolvedoras de software que incluem práticas de medição e análise. A norma internacional ISO/IEC 12207 (2008) descreve um processo de medição, com o propósito de coletar, analisar e relatar os dados relativos aos produtos desenvolvidos e aos processos implementados da unidade organizacional, para apoiar a gestão eficaz dos processos e demonstrar objetivamente a qualidade dos produtos. A ISO/IEC 15504 (2003) apresenta uma metodologia para realização de avaliação de processos em organizações. A medição é abordada no nível 3 no atributo de processo “Implantação do Processo” e no nível 4 nos atributos de processos “Medição do Processo” e “Controle do Processo”. O modelo CMMI [SEI 2010] apresenta a área de processo Medição e Análise (MA) no nível de maturidade 2. Essa área tem como propósito fornecer subsídio para desenvolver e manter uma capacidade de medição que é utilizada para apoiar as necessidades de informação gerenciais. Por fim, o Modelo de Referência MR-MPS [SOFTEX 2012] define um processo de Medição no nível de maturidade F (são sete níveis, do A [mais alto] ao G [mais baixo]), com o foco no apoio à tomada de decisão e atendimento aos objetivos organizacionais, e com o propósito de “coletar, armazenar, analisar e relatar os dados relativos aos produtos desenvolvidos e aos processos implementados na organização e em seus projetos, de forma a apoiar os objetivos organizacionais” [SOFTEX 2012].

## 2.2. Trabalhos Relacionados

De acordo com Mahnic e Vrana (2007), experiências têm mostrado que adotar métodos ágeis tem melhorado o gerenciamento do processo de desenvolvimento e as relações com o cliente. Porém, nos últimos anos, pesquisadores e adeptos das metodologias tem reconhecido a necessidade de métricas mais elaboradas para prover uma melhor visão do projeto. Schatz e Abdelshafi (2005) citam a preocupação dos *stakeholders* com a falta de métricas relacionadas ao acompanhamento do status do projeto e Yap (2006) também fomenta a discussão, afirmando que os métodos ágeis precisam prover melhores formas de mensurar a qualidade daquilo que está sendo entregue ao cliente.

Grande parte das métricas citadas neste trabalho tem como principal objetivo oferecer às equipes ágeis uma visão do andamento do projeto, no ponto de vista da atividade de teste, para que com o uso destas informações quantitativas os membros possam efetuar ou melhorar o planejamento da equipe e o seu processo de software.

Foram identificados trabalhos que abordam a utilização de métricas de software. Entretanto, poucos trabalhos propuseram abordagens para a escolha e utilização de métricas para projetos ágeis. Hartmann e Dymond (2006) propuseram uma lista de verificação baseada na abordagem GQM (*Goal/Question/Metric*) e na abordagem *Lean* para criação de métricas para projetos ágeis. Sato (2007) propõe um

conjunto de métricas organizacionais e de acompanhamento para projetos ágeis. Colette (2009) apresenta diversas métricas para projetos ágeis criadas a partir das características do manifesto ágil; e Kulas (2012) buscou descobrir uma maneira de como usar métricas de produtos em desenvolvimento ágil de software de maneira a fornecer os benefícios para todas as partes interessadas afetadas, incluindo as diretrizes gerais e recomendações detalhadas.

Analisando trabalhos que abordam a aplicação de medição em processos de teste, até o momento pouco resultado foi obtido. Vicente (2010) propõe a utilização de métricas para acompanhamento da atividade de teste no contexto de métodos ágeis. Este trabalho buscou eliminar aspectos de testes não produtivos, identificando boas práticas e criando formas de acompanhar e melhorar continuamente a condução da atividade de teste. Entretanto, o foco deste trabalho está em métricas relacionadas ao código, como por exemplo: cobertura de código, fator de teste e porcentagem de assertivas de teste de unidade passando e falhando. Algumas das métricas estudadas foram implementadas na ferramenta *Agile Testing Metrics Management (ATMM)*, desenvolvida para gerenciar as iterações de desenvolvimento do projeto ágil.

Motivado por este cenário, na próxima seção são descritos os resultados de um mapeamento sistemático que visou aprofundar a investigação a fim de identificar e caracterizar métricas de testes ágeis na literatura técnica, de forma a descrever seus objetivos e fatores que influenciam na utilização das mesmas.

### 3. Mapeamento Sistemático

Um Mapeamento Sistemático é um estudo secundário conduzido para fornecer uma visão geral de uma área de investigação, para determinar se existem evidências de pesquisa sobre o tema e fornecer indicação da quantidade [Kitchenham e Charters 2007].

Para a realização do Mapeamento Sistemático foi preciso, primeiramente, ser definido o problema, para depois serem definidos e formalizados as questões de pesquisa, o planejamento da revisão e as buscas e seleções dos trabalhos. O estudo é dividido em três fases principais, de acordo com [Travassos e Biolchini 2007]: Planejamento, Execução e Análise dos Resultados.

#### 3.1. Planejamento e Execução do Mapeamento Sistemático

O mapeamento sistemático realizado neste trabalho tem o propósito de identificar, analisar e avaliar estudos sobre métricas em projetos desenvolvidos com métodos ágeis e que possam ser utilizadas no acompanhamento das atividades de teste de software. Ele foi realizado no período de Dezembro/2013 à Março/2014.

Baseando-se no objetivo da pesquisa, foram formuladas uma questão principal, com duas questões secundárias, as quais são apresentadas a seguir:

- Quais as métricas utilizadas em projetos desenvolvidos com métodos ágeis?
  - o Q1– Quais dessas podem ser aplicadas ao processo de Teste de Software?
  - o Q2 – Quais os objetivos e características das métricas identificadas?

As questões de pesquisa foram estruturadas de acordo com a estratégia PICO (*Population, Intervention, Comparison e Outcomes*) descritas em [Kitchenham e Charters 2007] e foram definidas da seguinte forma:

- **(P) População:** Trabalhos publicados em conferências e periódicos na área de Métodos Ágeis: (“agile” OR “agile method” OR “agile methods” OR “agile methodology” OR “agile methodologies” OR “agile development” OR “agile process” OR “agile processes” OR “agile practice” OR “agile practices” OR “agile project” OR “agile projects” OR “agile lifecycle” OR “SCRUM” OR “extreme programming” OR “lean development” OR “feature driven development” OR “dynamic system development” OR “agile unified process”).
- **(I) Intervenção:** Métricas aplicadas: (“metric” OR “metrics” OR “measure” OR “measures” OR “measured” OR “measurement” OR “measurements” OR “indicator” OR “indicators” OR “evaluation”).
- **(C) Comparação:** Não se aplica em mapeamentos sistemáticos.
- **(O) Resultados:** Projetos de software, produtos de software, processos de software: (“software project” OR “software projects” OR “software process” OR “software processes” OR “software product” OR “software products” OR “software production”).

A pesquisa foi realizada em dois repositórios digitais e retornou as seguintes quantidades de arquivos: *Scopus* (214 artigos) e *IEEEExplore* (62 artigos), além das principais conferências nacionais (SBES, SBQS e WAMPS), sendo removidos 30 artigos duplicados. Após esta etapa, foi realizada a seleção primária, que é a leitura dos títulos e resumos dos documentos, levando em consideração os critérios de inclusão definidos neste estudo. Dentre os 246 artigos identificados, foram pré-selecionados 34 artigos para o segundo filtro (27 da *Scopus* e 7 da *IEEEExplore*). Na seleção secundária, que é a fase onde o pesquisador deve fazer a leitura dos documentos selecionados na íntegra, foram selecionados 16 artigos, citados neste documento.

### 3.2. Estratégia de Categorização das Métricas

Através do Mapeamento Sistemático foram identificadas 80 métricas utilizadas em projetos desenvolvidos com métodos ágeis, dentre as quais apenas 40 métricas podem ser aplicadas ao processo de teste de software. Destas, 16 métricas são equivalentes entre si. Essas métricas foram definidas de acordo com a lista de verificação proposta por Hartmann e Dymond (2006), que é apresentada na Tabela 1.

**Tabela 1. Lista de verificação para o planejamento de uma métrica ágil [Hartmann e Dymond 2006]**

Característica	Descrição
<b>Nome</b>	Deve ser bem escolhido para evitar ambiguidade.
<b>Classificação</b>	De acordo com a Tabela 2
<b>Atividade em que é coletada</b>	Planejamento ou Execução dos Testes, segundo o processo de teste proposto em [Dias-Neto2006].



<b>Objetivo</b>	Motivação, preocupação ou tópico, um objetivo e um ponto de vista.
<b>Pergunta</b>	Associada a uma pergunta específica (GQM).
<b>Base de Medição</b>	Uma clara definição das medidas utilizadas para o cálculo da métrica
<b>Suposições</b>	Devem ser identificadas para um claro entendimento do que os dados estão representando.
<b>Tendência esperada</b>	Uma ideia do comportamento esperado para a métrica.
<b>Quando utilizar</b>	Deve esclarecer os motivos que levaram à criação da métrica e, caso ela já tenha sido utilizada anteriormente, mostrar um pouco do seu histórico.
<b>Quando parar de utilizar</b>	É importante saber até quando uma métrica será útil, antes mesmo de utilizá-la, principalmente para métricas de acompanhamento.
<b>Formas de manipulação</b>	Deve esclarecer como as pessoas tentarão alterar seu comportamento em função da métrica gerar números “mais favoráveis”.
<b>Cuidados e observações</b>	“Recomendações sobre outras métricas similares, limites no uso e perigos associados à má utilização da métrica”.

A seguir, é apresentada a Tabela 2 contendo as informações à cerca das diversas formas de classificação das métricas.

**Tabela 2. Classificações das métricas**

Classificação	Descrição
<b>Objetiva/ Subjetiva</b>	Métrica <b>objetiva</b> depende do objeto de estudo. Métrica subjetiva depende do ponto de vista de quem interpreta [Sato 2007].
<b>Quantitativa/ Qualitativa</b>	Métrica <b>quantitativa</b> pertence a um intervalo de certa magnitude e geralmente é calculada por um número. Métrica qualitativa é representada por símbolos/artefatos, não números [Sato 2007].
<b>Organizacional/ Acompanhamento</b>	Métricas <b>organizacionais</b> medem a quantidade de valor de negócio entregue ao cliente. Métricas de <b>acompanhamento</b> apoiam a equipe no entendimento e melhoria do processo que produz valor de negócio [Hartmann e Dymond 2006].
<b>Direta/ Indireta</b>	<b>Métricas diretas</b> são métricas básicas e <b>métricas indiretas</b> são aquelas derivadas de outras métricas.
<b>PSM (Practical Software Measurement) [Mcgarry 2002]</b>	<ul style="list-style-type: none"> <li>• Cronograma e Progresso: relacionado ao cumprimento de marcos de projeto e à conclusão de unidades de trabalho nos prazos previstos.</li> <li>• Qualidade do produto: relacionada à capacidade do software produzido de atender sem falhas às necessidades do usuário.</li> <li>• Desempenho do Processo: relacionada à capacidade do processo de atender às necessidades apresentadas por cada projeto.</li> </ul>

### 3.3. Análise dos Trabalhos Identificados

As métricas selecionadas dos artigos são métricas de nível de processo, que são para avaliar o processo ágil. Elas servem para serem aplicadas e gerenciadas no projeto e estão associadas com a gestão da equipe, o andamento do trabalho e o planejamento de iteração. Essas métricas contribuem para a

detecção de problemas e acompanhamento da área de melhorias no processo ágil. A Tabela 3 detalha as 24 métricas selecionadas neste estudo. A lista completa de fontes usadas para identificação das métricas está apresentada ao final deste artigo.

O estudo possibilitou a identificação de métricas ágeis que podem ser aplicadas ao processo de Teste de Software. A maioria dos artigos apenas citava a métrica e assim, fez-se necessária a classificação e definição das mesmas como citado anteriormente. A classificação e definição foram realizadas através de análise de trabalhos relacionados e normas, e deste modo foi possível recolher uma boa quantidade de informações sobre a utilização e aplicação das métricas. O intuito desta pesquisa foi de facilitar a atividade de Medição e Análise realizada pela equipe de Teste de Software no gerenciamento de projetos ágeis. Vale ressaltar que a definição das métricas não foi direcionada a uma metodologia ágil específica.

Analisando os resultados da Tabela 3, observa-se que as 24 métricas foram classificadas nas categorias *Objetiva* e *Acompanhamento*. A Figura 1 apresenta a distribuição das métricas nas demais categorias (Quantitativa/Qualitativa; Direta/Indireta; PSM). Pode ser observado um predomínio de métricas *Quantitativas* (23) sobre as *Qualitativas* (1). Houve, ainda, um equilíbrio entre as métricas *Diretas* (14) e *Indiretas* (10). Também pode ser observado um equilíbrio entre as métricas definidas pelo PSM: *Cronograma e Progresso* (6), *Qualidade do Produto* (9) e *Desempenho do Processo* (9).

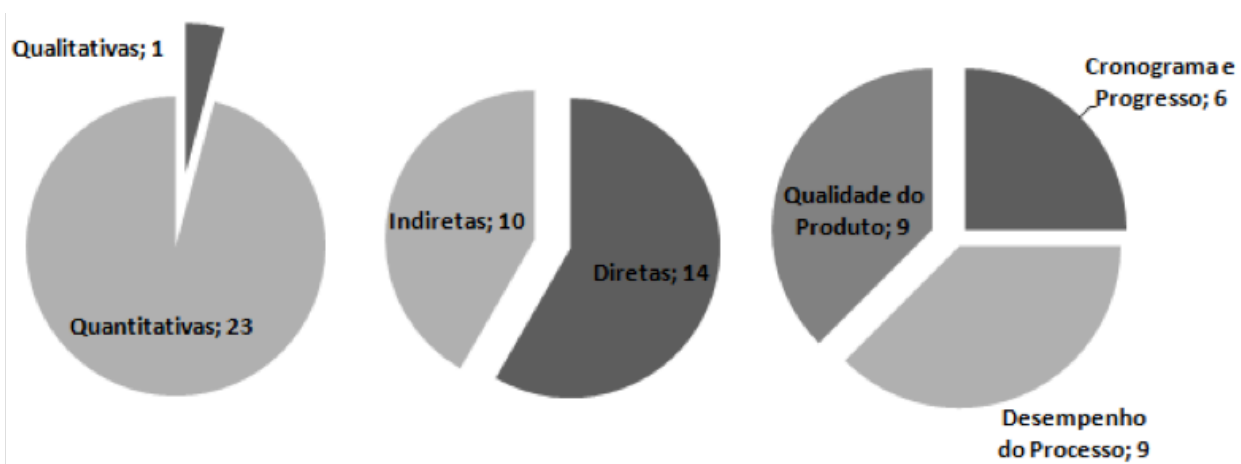


Figura 1. Distribuição das Métricas por Classificação.

Devido às limitações de espaço neste artigo, não é possível apresentar todas as características extraídas a respeito das métricas identificadas. Assim, o protocolo definido para o mapeamento sistemático e o detalhamento completo das métricas utilizadas neste trabalho podem ser obtidos em [www.icomp.ufam.edu.br/experts/metricastestesageis](http://www.icomp.ufam.edu.br/experts/metricastestesageis).

#### 4. Considerações Finais e Trabalhos Futuros

O Mapeamento Sistemático foi realizado com o objetivo de coletar métricas de trabalhos publicados, defini-las e avaliá-las de modo a contribuir com o gerenciamento do processo de teste de software de projetos desenvolvidos com métodos ágeis. Este estudo preliminar revela que existe pouca informação publicada relacionada com métricas de testes ágeis.

Nirpal e Kale (2011) atestam que medir o processo de teste é uma competência requerida para que um gerente de teste de software eficaz possa realizar a concepção e avaliação da sua estratégia de teste. Também afirmam que a gestão eficaz de qualquer processo requer quantificação, medição e modelagem, e que métricas de software oferecem uma abordagem quantitativa para o desenvolvimento e validação de modelos de teste de software.

Os próximos passos da realização do estudo é a avaliação da definição das métricas selecionadas por meio de uma Revisão por Pares com um grupo de revisores composto por pesquisadores e profissionais da área de teste de software e métodos ágeis. Trabalhos futuros incluem também um enriquecimento das métricas para que seja possível aplicá-las de forma eficiente, adotando as diretrizes providas em normas e modelos de melhoria de processo de software e testes. Por fim, ainda está prevista uma possível aplicação prática das métricas no ambiente industrial para que se possa observar a real contribuição com a melhoria do processo de teste de software.

**Tabela 3. Métricas coletadas através do Mapeamento Sistemático**

Métrica	Classificação	Atividade de coleta	Origem
Total de defeitos encontrados nos testes	Objetiva, Quantitativa, Direta, Acompanhamento, Qualidade do Produto	Execução	[1][2][3][4][5]
Tempo de retorno do defeito	Objetiva, Quantitativa, Indireta, Acompanhamento, Desempenho do processo	Execução	[6]
Tempo gasto para correção do defeito	Objetiva, Quantitativa, Direta, Acompanhamento, Desempenho do processo	Execução	[6]
Tempo planejado para correção do defeito	Objetiva, Quantitativa, Direta, Acompanhamento, Desempenho do processo	Execução	[6]
Total de defeitos resolvidos	Objetiva, Quantitativa, Indireta, Acompanhamento, Qualidade do Produto	Execução	[7]
Porcentagem de tempo gasto na correção de defeitos	Objetiva, Quantitativa, Indireta, Acompanhamento, Desempenho do processo	Execução	[1]
Densidade de falhas	Objetiva, Quantitativa, Indireta, Acompanhamento, Qualidade do Produto	Execução	[8][9]
Número de casos de teste automatizados	Objetiva, Quantitativa, Direta, Acompanhamento, Qualidade do Produto	Planejamento	[3]
Número de casos de teste manuais	Objetiva, Quantitativa, Direta, Acompanhamento, Qualidade do Produto	Planejamento	[3]
Número de testes de aceitação por estória	Objetiva, Quantitativa, Direta, Acompanhamento, Qualidade do Produto	Planejamento	[3]
Tempo para corrigir os testes	Objetiva, Quantitativa, Direta, Acompanhamento, Cronograma e Progresso	Execução	[3]
Tempo de conclusão do projeto	Objetiva, Quantitativa, Direta, Acompanhamento, Cronograma e Progresso	Execução	[10][11][12]
Número de casos de teste concluídos no prazo correto	Objetiva, Quantitativa, Direta, Acompanhamento, Cronograma e Progresso	Execução	[7]

Pontos de estória concluídos por iteração	Objetiva, Quantitativa, Direta, Acompanhamento, Cronograma e Progresso	Execução	[13][14] [15][16]
Planejamento do tamanho do backlog	Objetiva, Quantitativa, Indireta, Acompanhamento, Cronograma e Progresso	Execução	[3][13]
Estórias de usuário não verificadas	Objetiva, Quantitativa, Direta, Acompanhamento, Desempenho do processo	Execução	[6][15]
Quantidades de trabalho restante (em horas) para cada tarefa	Objetiva, Quantitativa, Direta, Acompanhamento, Qualidade do produto	Execução	[15]
Quantidade de trabalho gasto (em horas) para cada tarefa	Objetiva, Quantitativa, Direta, Acompanhamento, Desempenho do processo	Execução	[15]
Número de casos de teste executados	Objetiva, Quantitativa, Indireta, Acompanhamento, Desempenho do processo	Execução	[5]
Eficiência na remoção de defeitos	Objetiva, Quantitativa, Indireta, Acompanhamento, Desempenho do processo	Execução	[2][7]
Estimativa de estórias de usuário (tempo)	Objetiva, Quantitativa, Direta, Acompanhamento, Cronograma e progresso	Execução	[16]
Precisão da estimativa de teste	Objetiva, Qualitativa, Indireta, Acompanhamento, Desempenho do processo	Execução	[6]
Percentual de estórias aprovadas	Objetiva, Quantitativa, Indireta, Acompanhamento, Qualidade do Produto	Execução	[7]
Percentual de cobertura de teste	Objetiva, Quantitativa, Indireta, Acompanhamento, Qualidade do Produto	Execução	[3][16]

## Agradecimentos

Os autores agradecem à CAPES e FAPEAM pelo apoio para a condução da pesquisa.

## Referências Bibliográficas

- Ceschi, M., Sillitti, A., Succi, G. and Panfilis, S. (2005) "Project Management in Plan-Based and Agile Companies", IEEE Software, v.22 n.3, p.21-27.
- Colette, D. (2009) "Agilemetricspresentation", In: Apresentação na AgileConference, <http://davenicolette.wikispaces.com/file/view/agile-metrics-v6.ppt>.
- Crispin, L. and Gregory, J. (2009) "Agile testing: A practical guide for testers and agile teams". Addison-Wesley Professional.
- Dias-Neto, A.C. Uma Infra-Estrutura Computacional para Apoiar o Planejamento e Controle de Testes de Software. Dissertação de M.Sc., COPPE/URFJ, Brasil. 2006.
- Hartmann, D. and Dymond, R. (2006) "Appropriate agile measurement: Using metrics and diagnostics to deliver business value", In: Proceedings of AgileConference, pp. 128-134.

- ISO/IEC 15504 (2003) "Software Engineering – Process Assessment", The International Organization for Standardization and the International Electrotechnical Commission.
- ISO/IEC 12207 (2008) "Systems and Software Engineering – Software Life Cycle Process", The International Organization for Standardization and the International Electrotechnical Commission.
- Kitchenham, B.A.; Charters, S. (2007) "Guidelines for performing systematic literature reviews in software engineering". Tech. Rep. EBSE-2007-01, KeeleUniversity.
- Kulas, H. (2012) "Product Metrics in Agile Software Development", Master's Thesis, University of Tampere, Finland.
- Mahnic, V. and Vrana, I. (2007) "Using stakeholder-driven process performance measurement for monitoring the performance of a Scrum-based software development process". *Electrotechnical Review*, 74(5), 241-247.
- McGarry, J. (2002). *Practical Software Measurement: objective information for decision makers*. 1ed. Boston: Addison-Wesley. ISBN 0201715163.
- Nirpal, P.B. and Kale, K.V. (2011) "A Brief Overview Of Software Testing Metrics", *International Journal on Computer Science and Engineering (IJCSE)*, vol. 3, p. 1-8.
- Rocha, A. R., Barcellos, M. P. and Santos, G. (2012) "Medição de Software e Controle Estatístico de Processos", Ministério da Ciência, Tecnologia e Inovação - SEPIN - PBQP Software.
- Sato, D. T. (2007) "Uso eficaz de métricas em métodos ágeis de desenvolvimento de software", *Dissertação de Mestrado, Instituto de Matemática e Estatística – Universidade de São Paulo, SP*, Orientador: Prof. Dr. Alfredo Goldman.
- Schatz, B. and Abdelshafi, I. (2005). *Primavera Gets Agile: A Successful Transition to Agile Development*. *IEEE Software*, pp. 36-42.
- SEI (2010) "Capability Maturity Model Integration (CMMI) for Development", version 1.3, Carnegie Mellon University, Software Engineering Institute, Technical Report CMU/SEI-2010-TR-033.
- SOFTEX (2012) "MPS.BR – Melhoria de Processo do Software Brasileiro – Guia Geral", <http://www.softex.br/mpsbr>.
- TMMiFoundation (2012). *Test Maturity Model integration (TMMi)*. Release1.0.Irlanda, p. 2-11.
- Travassos, G. H. and Biolchini, J. (2007) "Revisões Sistemáticas Aplicadas a Engenharia de Software". Tutorial SBES 2007.
- Vicente, A. A. (2010) "Definição e gerenciamento de métricas de teste no contexto de métodos ágeis", *Dissertação de Mestrado, Instituto de Ciências Matemáticas e de Computação – ICMC/USP, São Carlos*, Orientador: Prof. Dr. Márcio Delamaro.
- Yap, M. (2006). *Value based Extreme Programming*, *Proceedings of AGILE 2006 Conference (AGILE'06)*, pp. 175-184.

## Artigos Utilizados no Mapeamento Sistemático

- [1] Ilieva, S., Ivanov, P. and Stefanova, E. (2004) "Analyses of an agile methodology implementation", Proceedings of the 30<sup>th</sup> EUROMICRO Conference, pp. 326-333.
- [2] Schooenderwoert, N. V. (2006) "Embedded Agile Project by the Numbers with Newbies", Agile 2006 conference report.
- [3] Ayed, H., Habra, N. and Vanderose, B. (2013) "AM-QuICk: A Measurement-Based Framework for Agile Methods Customisation", In Proc. IWSM/Mensura 2013: 71-80.
- [4] Green, P. (2011) "Measuring the Impact of Scrum on Product Development at Adobe Systems", In: 44th Hawaii International Conference on System Sciences, p.1-10, 2011.
- [5] Ruiz, J.C., Osorio, Z.B., Mejia, J., Muñoz, M., Chávez, A.M. and Olivares, B.A. (2011) "Definition of a hybrid measurement process for the models ISO/IEC 15504-ISO/IEC 12207:2008 AND CMMI-DEV1.3, SMEs", Electronics, Robotics and Automotive Mechanics Conference (CERMA), 2011 IEEE, 421-426.
- [6] Heidenberg, J. and Porres, I. (2010) "Metrics Functions for Kanban Guards", In: IEEE International Conference and Workshops on Engineering of Computer-Based Systems. IEEE Computer Society.
- [7] Atiwithayaporn, S. and Rivepiboon, W. (2013) "A development of the effectiveness evaluation model for agile software development using the balanced scorecard", IMECS.
- [8] Ayed, H., Vanderose, B. and Habra, N. (2012) "A metamodel-based approach for customizing and assessing agile methods", In: The International Conference on the Quality of Information and Communications Technology (QUATIC).
- [9] Mahnic, V. and Vrana, I. (2007) "Using stakeholder driven process performance measurement for monitoring the performance of a Scrum based software development process", Electrotechnical Review.
- [10] Kusumasari, T. F., Supriana, I., Surendro, K., Sastramihardja, H. (2011) "Collaboration model of software development". Electrical Engineering and Informatics (ICEEI).
- [11] Brown A., Ambler S. and Royce W. (2013) "Agility in Scale: Economic Governance, Measured Improvement, and Disciplined Delivery", ICSE.
- [12] Desharnais, J.-M., Kocaturk, B. and Abran, A. (2011) "Using the COSMIC method to estimate Agile user stories", Proceedings of the 12th International Conference on Product Focused Software Development and Process Improvement.
- [13] Hodgetts, P. (2004); Refactoring the development process: experiences with the incremental adoption of agile practices. IEEE Computer Society. In: Agile Development Conference.
- [14] Racheva, Z., Daneva, M. and Buglione, L. (2008) "Complementing Measurements and Real Options Concepts to Support Inter-iteration Decision-Making in Agile Projects", Proc. 34th Euromicro Conf. Software Engineering and Advanced Applications, pp. 457-464.

- [15] Mahnic, V. and Zabkar, N. (2012) "Measuring progress of Scrum-based software projects", Electronics and electrical engineering, No. 8(124).
- [16] Ktata, O. and Lévesque, G. (2010) "Designing and implementing a measurement program for Scrum teams: what do agile developers really need and want?", Proceedings of the Third C\* Conference on Computer Science and Software Engineering, p.101-107.

# Processos de Apoio Gerencial Integrados ao Processo de Teste de Software

Jeanne de Castro Trovão, Arilo Claudio Dias Neto

Instituto de Computação – Universidade Federal do Amazonas (UFAM)  
Av. General Rodrigo Octávio, 6200 - Coroado I - 69077-000 – Manaus – AM – Brasil

{jeanne.trova, arilo}@icompu.ufam.edu.br

**Abstract.** *Software testing requires an appropriate management of its process so that the tests achieve the expected performance, within the defined time and budget. To aid this management, processes of managerial support can be integrated into the software testing process. This paper presents the specification of two processes of managerial support to the Software Testing Process, the Test Risks Management Process and the Test Human Resources Management Process, in order to contribute to the management of testing activities. By means of a peer review, specialists from industry and academy evaluated this approach resulting in the evolution of the processes from observations received.*

**Resumo.** *Teste de software requer um gerenciamento adequado de seu processo para que os testes alcancem a performance esperada, dentro do prazo e orçamento definidos. Para auxiliar este gerenciamento, processos de apoio gerencial podem ser integrados ao processo de testes de software. Este artigo apresenta a especificação de dois processos de apoio gerencial ao Processo de Testes de Software, o Processo de Gerência de Riscos de Teste e o Processo de Gerência de Recursos Humanos de Teste, visando contribuir com o gerenciamento das atividades de testes. Por meio de uma revisão por pares, especialistas da indústria e academia avaliaram esta abordagem resultando na evolução dos processos a partir das observações recebidas.*

## 1. Introdução

De acordo com o SEI (2010), “a qualidade de um sistema ou produto é amplamente influenciada pela qualidade do processo utilizado”. Uma forma de contribuir com a qualidade de um processo é a partir de processos de apoio aplicados a um processo principal. Processos de apoio são aqueles cujas atividades presentes nele não fazem parte das atividades mais importantes para a obtenção de um produto principal, porém auxiliam na realização desta tarefa, garantindo o apoio necessário ao funcionamento adequado do processo principal, aumentando sua eficiência e ajudando no controle e gerenciamento do mesmo [ISO/IEC 2008][SEI 2010].

As atividades de teste de software possuem um papel fundamental no desenvolvimento de um software como mecanismo de apoio à garantia da qualidade do produto [Pressman 2010].

No entanto, apesar da notável importância do gerenciamento do processo de testes para que ele obtenha sucesso, observam-se na literatura técnica da área de testes poucas abordagens que apoiam essa atividade e, normalmente, estas abordagens são propostas para um contexto específico (cenário observado a partir dos resultados do mapeamento sistemático realizado neste trabalho).

Neste contexto, visando diminuir a carência de atividades que apoiam o gerenciamento do processo de teste, este trabalho apresenta a especificação de dois processos de apoio gerencial ao processo



de teste de software: Processo de Gerência de Riscos de Teste e o Processo de Gerência de Recursos Humanos de Teste. Ambos os processos foram especificados a partir de uma análise de normas, guias e modelos de qualidade disponíveis na literatura técnica, especializados em teste de software (MPT.BR [SOFTEX Recife 2011], TMMI [TMMi Foundation 2012]) ou abrangente em engenharia de software (MR-MPS-SW [SOFTEX 2012], CMMI-DEV [SEI, 2010], ISO/IEC 12207 (2008) e PMBOK [PMI 2013]), além de artigos científicos identificados durante o trabalho. O objetivo com a definição deste processo é a definição de suas atividades, dependências, papéis e artefatos adotando as recomendações das fontes citadas acima. Além disso, visando a aplicação deste processo ao cenário de teste de software, práticas extraídas da literatura técnica serão definidas na especificação de cada atividade do processo

Este artigo está organizado da seguinte forma: a Seção 2 aborda a aplicação de processos de apoio ao processo de teste, o mapeamento sistemático realizado durante a pesquisa e trabalhos relacionados; a Seção 3 apresenta o Processo de Gerência de Riscos de Teste; a Seção 4 apresenta o Processo de Gerência de Recursos Humanos de Teste; a Seção 5 apresenta a avaliação dos processos; e, por fim, a Seção 6 apresenta as considerações finais e trabalhos futuros.

## **2. Processos de Apoio Aplicados ao Processo de Teste de Software**

### **2.1. Processos de Apoio em Normas e Modelos de Maturidade**

Um processo de teste tem como objetivo estruturar as atividades, os artefatos, os papéis e as responsabilidades do teste de software, permitindo organização e controle de todo o ciclo do teste, agregando qualidade ao software [Eliza e Lagares 2012]. Apesar de toda a complexidade deste processo, existem atividades necessárias que devem ser consideradas para que ele seja executado de forma adequada e com o menor custo possível. Estas necessidades podem ser supridas com a integração de processos que possam dar suporte a elas durante a execução do processo de teste.

Algumas normas como a ISO/IEC 12207 (2008), modelos de melhoria de qualidade do software e do processo de teste como CMMI-DEV [SEI 2010], TMMI [TMMi Foundation 2012], MR-MPS-SW [SOFTEX 2012], MPT.BR [SOFTEX Recife 2011] e até guias como o PMBOK [PMI 2013] não especificam os processos de apoio com os detalhes necessários para sua implantação em organizações de software, porém sugerem práticas e resultados esperados para estes processos que podem ser utilizados no contexto de engenharia (teste) de software para apoiar no gerenciamento das suas atividades.

Assim, este trabalho possui como objetivo a definição de processos de apoio a serem aplicados de forma integrada ao processo de teste de software. No entanto, suprir a carência de todos os processos de apoio para o processo de teste seria inviável, pois exigiria um tempo muito maior que o especificado para a realização deste trabalho. Portanto, para selecionar quais processos seriam priorizados para serem especificados, foi feita uma análise nos modelos e normas citados anteriormente para identificar quais processos de apoio poderiam contribuir de forma mais direta para o sucesso do processo de teste. Com isso, uma lista de processos foi gerada, analisada e categorizada entre processos de apoio técnico (como por exemplo, Gerência de Configuração, Gerência de Reutilização) e gerencial (como por exemplo, Gerência de Riscos, Gerência de Recursos Humanos, Gerência de Portfólio). Neste trabalho, com base nas contribuições identificadas pela análise sobre cada processo de apoio, foram escolhidos dois destes processos gerenciais (Gerência de Riscos e Gerência de Recursos Humanos)

para serem especificados seguindo as diretrizes dos principais modelos e normas de apoio à qualidade de processo de software, que priorizam o aspecto gerencial antes do técnico (os processos gerenciais, em geral, estão em níveis de maturidade inferiores em relação aos técnicos). Outros processos de apoio poderiam ser escolhidos, porém em um primeiro momento optou-se por desenvolver este trabalho com dois processos, e como trabalhos futuros novos processos de apoio serão definidos.

Para a especificação dos processos de apoio, utilizou-se como base o Processo de Testes de Software descrito em [Dias-Neto 2006] para descrever as interações necessárias entre ele e os processos de apoio. Este processo tem sido utilizado em diversas pesquisas do Brasil e suas atividades foram definidas com base em recomendações do IEEE 829 (2008), atividades descritas na literatura de teste de software e em gerência de projetos (CMMI-DEV [SEI 2010] e PMBOK [PMI 2013]), características que levaram à escolha deste processo. Ele é dividido nos subprocessos de Planejamento dos Testes e Execução dos Testes (Figura 1), incluindo as suas macroatividades, papéis e artefatos.

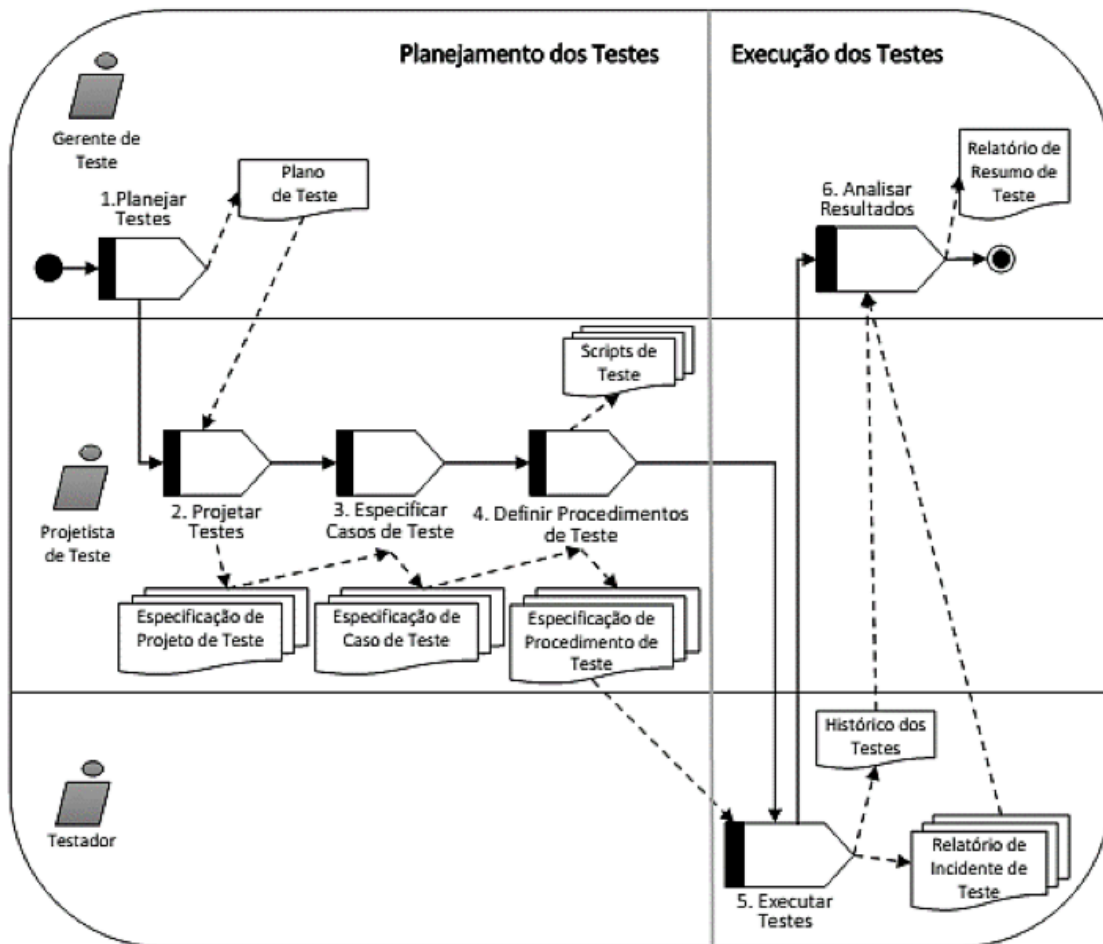


Figura 1. Subprocessos que compõem o processo de testes de [Dias-Neto 2006]

A Figura 2 apresenta o detalhamento das atividades que compõem a macroatividade “1. Planejar Testes”, visualizada na Figura 1. Analisando as atividades da Figura 2, pode ser observado que as atividades “2. Identificar Riscos”, “4. Planejar Recursos Humanos” e “8. Especificar Necessidade de Treinamento” são consideradas atividades externas, pois são realizadas por meio de processos de apoio, a serem descritos nas próximas seções deste artigo. A atividade “2. Identificar Riscos” é realizada pelo subprocesso “Planejamento de Riscos de Teste” que faz parte do processo de “Gerência de Riscos de Teste” (Seção 3). As outras duas atividades externas são realizadas por meio de macroatividades do processo de “Gerência de Recursos Humanos de Teste”, (Seção 4). A atividade “4. Planejar Recursos Humanos” é realizada pela macroatividade “1. Planejar Recursos Humanos de Teste” e a atividade “8. Especificar Necessidade de Treinamento” é realizada pela macroatividade “2. Planejar Obtenção de Conhecimento”. Este processo e os processos de apoio deste trabalho utilizam uma notação gráfica de processos adaptada de [Villela, 2004].

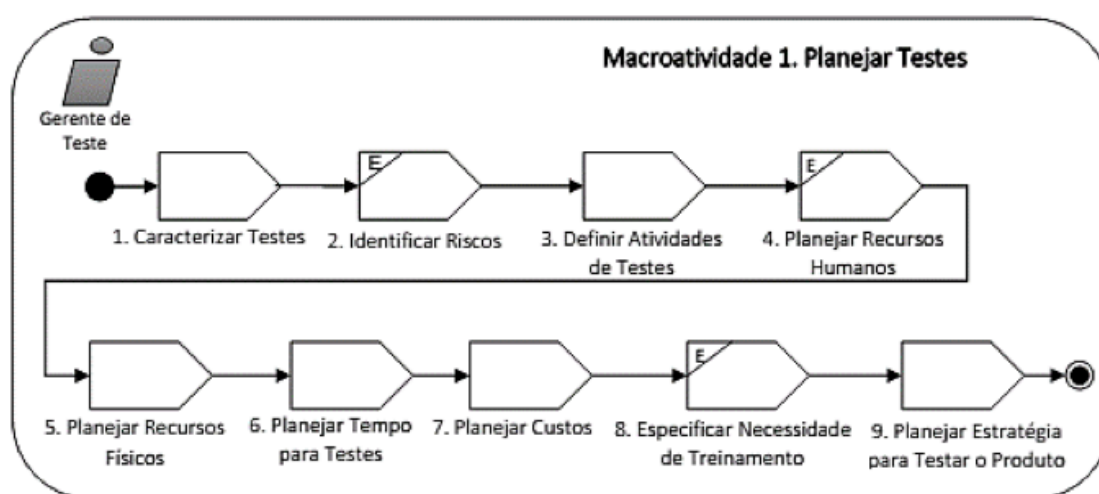


Figura 2. Detalhamento da Macroatividade Planejar Testes (Figura 1)

## 2.2. Mapeamento Sistemático Sobre Processos Gerenciais de Apoio Aplicados ao Processo de Teste de Software

Para obter maior embasamento teórico na literatura técnica sobre o cenário dos processos de apoio priorizados neste trabalho e buscar contribuições para a especificação destes processos, foi realizado um mapeamento sistemático com o objetivo de identificar estudos sobre processos ou atividades de apoio gerencial abordados (gerência de recursos humanos e de riscos) para o processo de testes de software, e sua aplicação neste contexto.

O mapeamento foi realizado nas bibliotecas digitais *Scopus*, *IEEE Explorer*, *ACM Digital Library* e nos anais dos eventos SBES, SQBS e WAMPS, com buscas separadas para cada processo de apoio (*strings* de busca diferentes) de forma a direcionar a filtragem para cada processo. Cada *string* foi formada por 3 conjuntos de palavras-chaves, de acordo com o padrão PICO (*population, intervention, comparison, outcomes*) [Kitchenham e Charters 2007]: População: Processos de teste; Intervenção: Processo de apoio [Gerência de Riscos ou de RH]; Comparação: Não se aplica em mapeamentos sistemáticos; Resultados: Processos, relatos, abordagens, métodos, metodologias. Foram realizados dois filtros de seleção, conforme mostra a Tabela 1.

**Tabela 1. Artigos identificados/selecionados com o mapeamento sistemático**

Processo de Apoio	Retornados	Pré-Selecionados (Título + <i>Abstract</i> )	Selecionados (Artigo Completo)
Gerência de Recursos Humanos	105	6	2
Gerência de Riscos	527	33	8
<b>Total</b>	<b>852</b>	<b>42</b>	<b>10</b>

Neste estudo, foram identificadas algumas características com relação à aplicação dos dois processos de apoio escolhidos ao tentar responder questões, como: qual a aplicação do processo de apoio para o processo de teste; como a/o abordagem/processo foi avaliada(o) (caso tenha sido); e se está associado a algum modelo/norma para melhoria de processo. O resultado obtido a partir da análise dos artigos identificados pode ser conferido na Tabela 2. No entanto, estes trabalhos se encaixam em uma das seguintes situações: são soluções para um contexto específico onde um processo de apoio ou práticas relacionadas a ele são aplicados para resolver o problema existente e, provavelmente, não se adequa a um contexto mais abrangente de um processo de teste; ou o trabalho aborda a importância do processo de apoio para o contexto de teste, reúne práticas a serem realizadas, mas não apresenta um processo especificado que possa ser aplicado ao processo de teste.

**Tabela 2. Resultados do mapeamento sistemático**

Processo de Apoio	Aplicação em Testes	Características avaliadas	Modelo/ Norma
Gerência de RH	<ul style="list-style-type: none"> <li>• Testes terceirizados (2)</li> <li>o Eficácia na realocação de pessoas (1)</li> </ul>	50% (1) avaliou: Aplicabilidade	Nenhum
Gerência de Riscos	<ul style="list-style-type: none"> <li>• Priorização de casos de teste (4)</li> <li>• Estimativa de tempo necessário para testar (1)</li> <li>• Aumento da taxa de detecção de defeitos (1)</li> <li>• Conformidade com o orçamento previsto (3)</li> <li>• Conformidade com o cronograma previsto (6)</li> <li>• Avaliação da confiabilidade após os testes (1)</li> </ul>	80% (8) avaliou: <ul style="list-style-type: none"> <li>• Custo-benefício (2)</li> <li>• Taxa e tempo de detecção de falhas (2)</li> <li>• Severidade das falhas (2)</li> <li>• Eficácia em identificar riscos (2)</li> <li>• Produtividade dos testes (1)</li> </ul>	<ul style="list-style-type: none"> <li>• CMMI (2)</li> <li>• TMMI (1)</li> </ul>

Além dos trabalhos encontrados a partir do mapeamento sistemático, outros dois trabalhos relacionados foram encontrados e abordam o uso de outros processos de apoio ao processo de teste de software, como o trabalho de [Boycan e Chernak 2005], que faz um relato sobre a experiência de uma organização na implementação do processo de gerenciamento de configuração em projetos de teste de software críticos; e [Mäntyniemi *et al.* 2005], que propõe um modelo de processo de teste para o desenvolvimento/uso de ativos de teste reutilizáveis baseando-se em métodos de reuso de software e trazendo-os para o contexto de teste para tentar melhorar a eficiência dos testes.

Por questão de espaço, não é possível detalhar os artigos identificados e os resultados obtidos com mapeamento sistemático, que estão publicados em [www.icomp.ufam.edu.br/experts/processosapoio](http://www.icomp.ufam.edu.br/experts/processosapoio).

### 3. Definição do Processo de Gerência de Riscos de Teste

O objetivo deste processo é prever os riscos que podem afetar as atividades de teste durante seu planejamento ou execução e estabelecer planos para eliminar ou diminuir o impacto destes riscos, de forma que o processo de teste possa seguir seu fluxo com o mínimo de problemas possíveis.

As categorias de riscos podem ser definidas como [Sommerville 2007]:

- **Riscos de Projeto:** afetam o cronograma ou os recursos do projeto;
- **Riscos de Produto:** afetam a qualidade ou o desempenho do software que está sendo desenvolvido;
- **Riscos para os negócios:** afetam a organização que desenvolve ou adquire o software. Os riscos pertencentes a esta última categoria devem ser gerenciados em nível de projeto de software e, portanto, não são tratados neste trabalho, já que o foco é o processo de testes.

O papel existente neste processo é o do Analista de Riscos de Teste, responsável por todas as atividades referentes ao gerenciamento de risco e deve ter conhecimento suficiente sobre o processo, o domínio do software a ser testado e sobre diretrizes da organização que podem estar relacionadas com os riscos. Quando não há um profissional dedicado para este papel, o Gerente de Teste (ou na sua ausência, o Gerente de Projeto) assumiria esta responsabilidade.

Os artefatos gerados por este processo incluem:

- **Plano de Riscos de Teste:** descreve como o gerenciamento de riscos será executado, monitorado e controlado em um determinado projeto de teste;
- **Registro de Riscos:** onde os riscos identificados são documentados e todas as informações referentes a eles são armazenadas e atualizadas no decorrer do processo de riscos [PMI 2013] [IEEE 2008];
- **Relatório de Dados de Riscos:** um relatório final do processo de riscos e serve para comunicar a situação dos riscos no processo de teste, possibilitando a tomada de decisões e o registro de informações.

Os artefatos utilizados que não foram descritos aqui pertencem ao Processo de Teste de Software utilizado [Dias-Neto 2006].

Este processo é dividido em dois subprocessos: de Planejamento de Riscos de Teste e de Monitoramento e Controle de Riscos de Teste. O processo inicia a partir de uma atividade do Processo de Testes de Software (atividade externa 2. *Identificar Riscos*). No entanto, suas atividades são realizadas de forma paralela ao Processo de Testes, permitindo que seus artefatos sejam atualizados sempre que necessário.

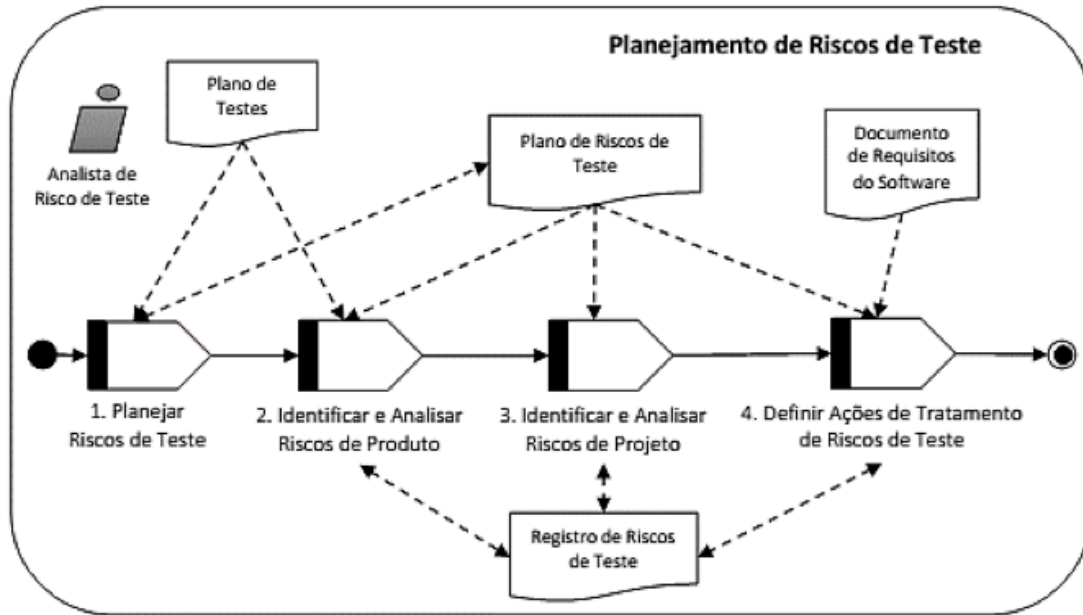


Figura 3. Subprocesso de Planejamento de Riscos de Testes.

O objetivo do primeiro subprocesso é estabelecer o escopo do gerenciamento de riscos a ser realizado durante o processo de teste e todas as informações necessárias para a sua execução, além de prover a documentação dessas informações que servirão de guia para o monitoramento e controle dos riscos. A Figura 3 apresenta este subprocesso.

O segundo subprocesso (Figura 4) tem como objetivo executar as instruções definidas no Plano de Riscos de Teste para monitorar os riscos identificados e realizar as ações especificadas no plano para o controle dos riscos quando aplicável, além de documentar o que for feito no decorrer deste subprocesso.

#### 4. Processo de Gerência de Recursos Humanos de Teste

Todo processo de teste necessita de recursos humanos devidamente alocados aos projetos de teste com habilidades específicas, de acordo com as características de cada projeto e dos tipos de teste a serem aplicados, e que estejam disponíveis para exercer suas funções no tempo que for determinado.

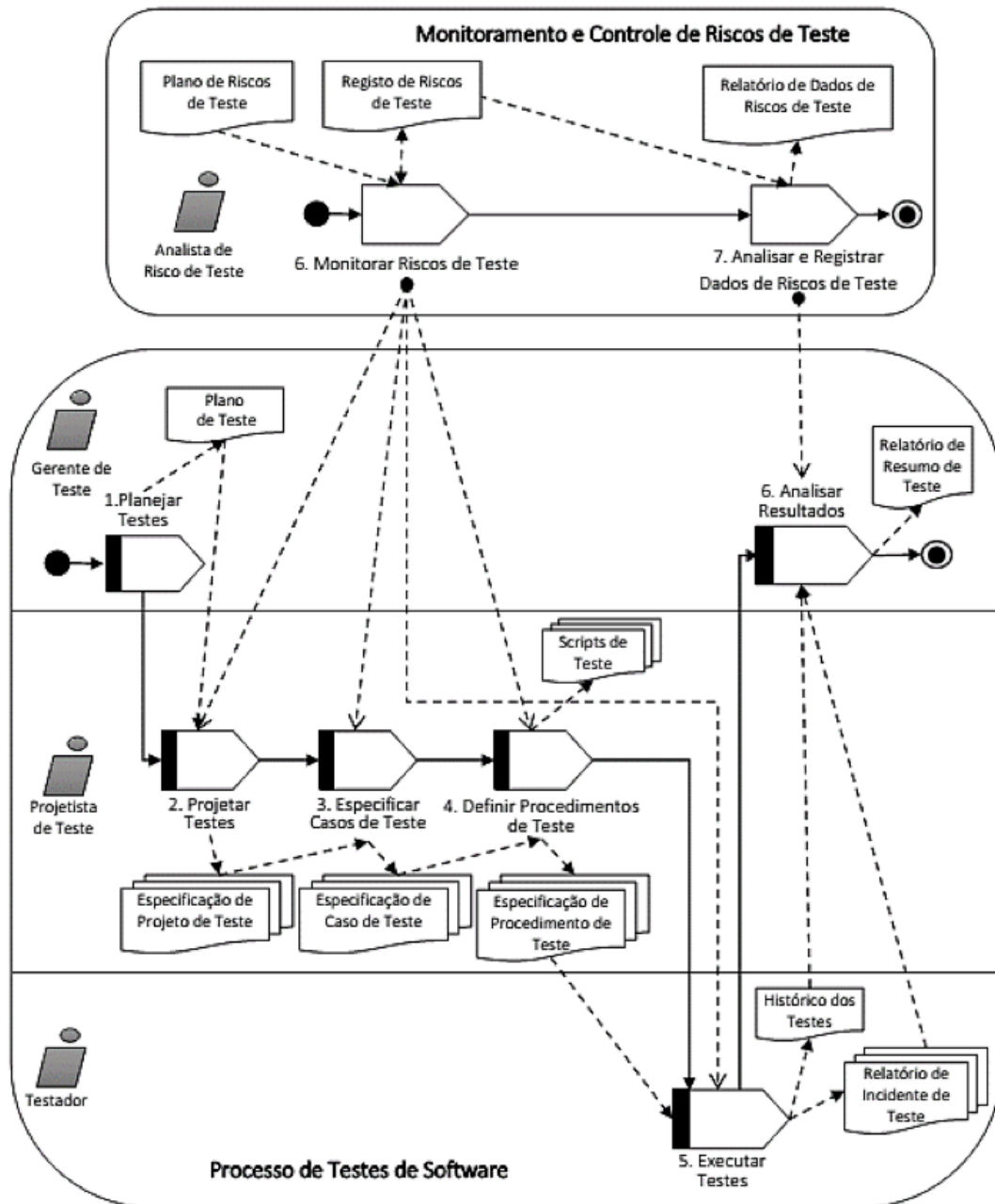


Figura 4. Subprocesso de Monitoramento e Controle de Riscos de Testes

O processo de Gerência de Recursos Humanos de Teste ajuda a identificar quais habilidades são necessárias e a selecionar os recursos disponíveis. Além disso, este processo também ajuda a coordenar atividades referentes a treinamentos e contratações de recursos, caso não haja recursos disponíveis no quadro de funcionários ou se eles não possuírem as habilidades requeridas para o projeto.

O papel responsável pelo processo é o de Gerente de Recursos Humanos, que deve definir as necessidades relacionadas aos recursos humanos requeridos para o projeto de teste e prover treinamento e/ou recrutamento adequado para atender às necessidades do projeto. Na ausência de um profissional dedicado a este papel, o Gerente de Teste pode executar as atividades deste processo por ter um conhecimento mais abrangente com relação ao processo de teste e aos projetos de teste que são realizados.

Os artefatos gerados pelo processo são:

- **Plano de Recursos Humanos de Teste**, onde são especificados os recursos humanos requeridos para a realização dos testes, os papéis, as responsabilidades e os conhecimentos necessários, além dos planos de treinamento e contratação;
- **Repositório de Recursos Humanos de Teste**, com dados a respeito dos recursos humanos de teste, como habilidades, papéis, características comportamentais e profissionais, projetos trabalhados, disponibilizando informações necessárias para uma alocação efetiva destes recursos em um novo projeto.

A Figura 5 apresenta o modelo do Processo de Gerência de Recursos Humanos de Teste. Este processo atua durante a macroatividade 1 do Processo de Teste de Software (Figura 2) e se relaciona de forma similar ao outro processo de apoio. Ao iniciar a subatividade 4. *Planejar Recursos Humanos do Processo de Teste*, a macroatividade 1. *Planejar Recursos Humanos de Teste* é executada e o *Plano de Recursos Humanos de Teste* é gerado. Ao iniciar a subatividade 8. *Especificar Necessidade de Treinamento*, é executada a macroatividade 2. *Planejar Obtenção de Conhecimento*. Ao concluir, o Processo de Teste volta a seguir seu fluxo e as próximas atividades do processo de apoio são realizadas em paralelo ao longo do Processo de Teste.

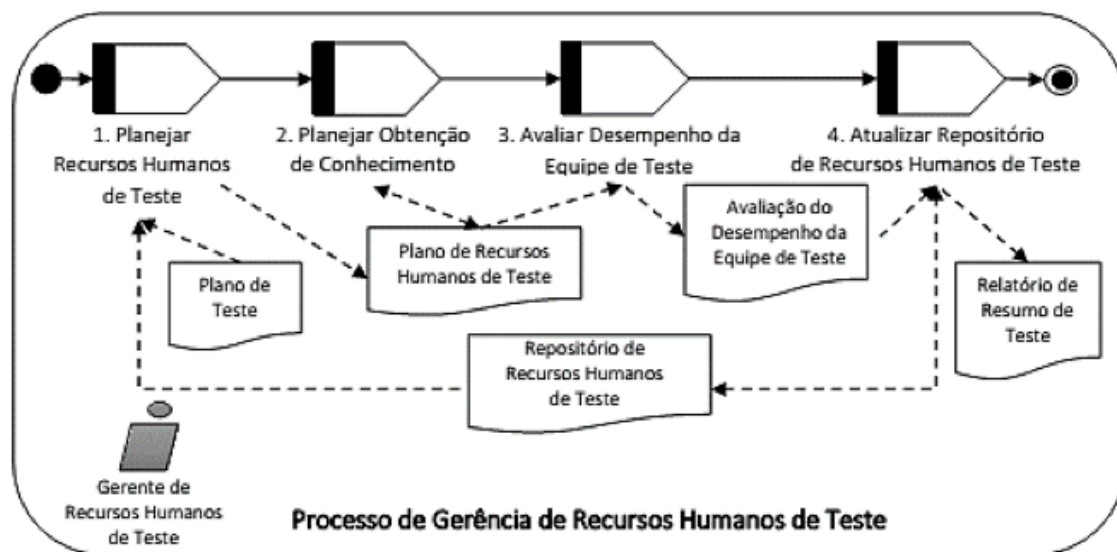


Figura 5. Processo de Gerência de Recursos Humanos de Teste.

Além das interações explícitas com o Processo de Teste ilustradas nos processos de apoio, também foram incluídas ao longo do Processo de Teste várias interações especificando como o gerenciamento de riscos e de recursos humanos influencia na execução de algumas de suas atividades.

Por questão de espaço, não é possível detalhar ambos os processos neste artigo. Assim, o detalhamento completo dos dois processos de apoio definidos neste trabalho podem ser obtidos em [www.icomp.ufam.edu.br/experts/processosapoio](http://www.icomp.ufam.edu.br/experts/processosapoio).



## 5. Avaliação dos Processos de Apoio

A avaliação dos processos de apoio definidos neste trabalho foi feita por um conjunto de especialistas utilizando a técnica de Revisão por Pares, que busca identificar defeitos e sugerir correções ou recomendar mudanças nos componentes revisados [SEI 2010]. Este estudo foi conduzido com o objetivo de avaliar se os processos de apoio especificados neste trabalho estavam adequados para atender às necessidades do processo de testes, sob o ponto de vista de pesquisadores e profissionais com experiência nas áreas de conhecimento dos processos de apoio (riscos e recursos humanos) e de teste.

A revisão foi planejada com base em dois focos de avaliação: o foco na forma (avalia se todos os componentes necessários para execução de um processo foram especificados e estão definidos corretamente) e o foco no conteúdo (avalia se os componentes do processo e o seu conteúdo são adequados para a execução do processo, e se a instanciação dele ao processo de testes foi realizada adequadamente). Para isto, foram selecionados cinco revisores de cinco perfis de conhecimento diferentes, que são:

- Perfil de Definição/Avaliação de Processos de Software (DAPS) (foco na forma);
- Perfil de Gerência de Recursos Humanos (GRH) (foco no conteúdo);
- Perfil de Gerência de Riscos (GRI) (foco no conteúdo);
- Perfil de Processo de Testes – Academia (PTA) (foco no conteúdo), e;
- Perfil de Processo de Testes – Indústria (PTI) (foco no conteúdo).

Para a condução desta revisão, foram desenvolvidos dois *checklists* (um para cada foco de avaliação) e uma planilha de resposta adaptada de [Barcellos 2009]. Como resultado, foram retornadas observações analisadas e categorizadas em: NA, se a observação não foi atendida; PA, se a observação foi parcialmente atendida; e A, se a observação foi atendida. A Tabela 3 apresenta estes resultados.

**Tabela 3. Quantidade de Observações por Revisor e por Resultado da Análise**

Revisor	Decisão após Análise			Total por Revisor
	NA	PA	A	
DAPS	0	0	15	15
GRH	15	2	9	26
GRI	8	1	19	28
PTI	0	0	1	1
PTA	0	0	1	1
<b>Total por Resultado</b>	<b>23</b>	<b>3</b>	<b>45</b>	<b>71</b>

Os resultados da revisão contribuíram bastante para o trabalho possibilitando a correção dos itens a partir das observações, que em sua maioria se relacionavam a termos inadequados e modelagem inconsistente, trazendo mais confiabilidade para os processos que foram especificados. De um conjunto de 71 observações realizadas pelos revisores, 48 (cerca de 67%) foram completamente ou parcialmente atendidas, o que demonstra que os processos evoluíram bastante após as correções, apresentando assim, indícios de sua viabilidade para aplicação em projetos reais da indústria. As observações atendidas visando a melhoria da descrição dos elementos dos processos (tais como a descrição da atividade, critérios de entrada e saída, utilização de termos adequados a cada área), sua representação gráfica e integração ao processo de testes. As observações não atendidas visavam, em geral, ajustes no processo de testes, mas este não foi foco de avaliação, e por isso não foram consideradas.

Por questão de espaço, novamente não é possível detalhar o material produzido na revisão por pares e a lista com as contribuições fornecidas pelos revisores, que podem ser obtidos em [www.icomp.ufam.edu.br/experts/processosapoio](http://www.icomp.ufam.edu.br/experts/processosapoio).

## 6. Considerações Finais e Trabalhos Futuros

Segundo normas e modelos de maturidade relacionados à qualidade de software, os processos de apoio são fatores importantes para a obtenção da qualidade esperada nos processos de uma organização. No entanto, a definição e implantação destes processos pode representar um risco para uma organização, caso não estejam bem fundamentados. Em geral, a literatura técnica e relatos da indústria destacam as aplicações destes processos de apoio ao processo de desenvolvimento de software. No entanto, pouca atenção tem sido dada à sua aplicação de forma integrada ao processo de testes, processo este fundamental para a obtenção da qualidade de um produto de software.

Neste trabalho, foram apresentados processos de apoio gerencial (de riscos e de recursos humanos) a serem instanciados para uma organização de software de forma integrada ao processo de testes, definidos a partir da análise de modelos de maturidade, normas e corpos de conhecimento disponíveis na literatura técnica para o processo de teste ou aplicáveis de forma geral a processos de software. Como principal contribuição, os processos especificados podem ser utilizados por uma organização em seus projetos de teste, identificando atividades necessárias, papéis responsáveis pelas atividades, artefatos produzidos e, principalmente, os pontos de integração entre estes processos de apoio e o processo de teste de software.

Outros processos de apoio poderiam ser especificados e integrados ao processo de teste de software. No contexto desta pesquisa, como um dos trabalhos futuros está prevista a especificação de mais um processo de apoio para o processo de teste: Gerência de Portfólio, visando ajudar a gerenciar os projetos da equipe de teste e a escolher os projetos que atendem ao escopo dos tipos de teste que a equipe realiza, verificando se ela possui recursos para incluir um novo projeto. Outro trabalho futuro é realizar um estudo de caso para verificar a viabilidade dos processos em um cenário real da indústria.

## Referências

- Barcellos, M. P. (2009). Uma Estratégia para Medição de Software e Avaliação de Bases de Medidas para Controle Estatístico de Processos de Software em Organizações de Alta Maturidade. Tese de D.Sc., COPPE/UFRJ, Brasil.
- Boycan, S.; Chernak, Y. (2005). Implementing configuration management for software testing projects. *The Journal of Defense Software Engineering*.
- Dias-Neto, A.C. (2006). Uma Infra-Estrutura Computacional para Apoiar o Planejamento e Controle de Testes de Software. Dissertação de M.Sc., COPPE/URFJ, Brasil.
- Eliza, R.; Lagares, V. (2012). Processo de Teste de Software. *Revista Java Magazine*.101.
- IEEE Computer Society. (2008). IEEE 829: Standard for Software Test Documentation.
- ISO/IEC. (2008). ISO/IEC 12207: System and Software Engineering – Software Life Cycle Processes.
- Kitchenham, B.; Charters, S. (2007). Guidelines for Performing Systematic Literature Reviews in Software Engineering, v.2.3. RT, Evidence-Based Software Engineering.
- Mäntyniemi, A.; Maki-Asiala, P.; Karki, M. (2005). A Process Model for Development and Utilization of Reusable Test Assets. *Software Engineering Research and Practice*.
- PMI - Project Management Institute (2013). A Guide to the Project Management Body of Knowledge (PMBOK® Guide) – 5<sup>th</sup> Ed.
- Pressman, R. S. (2010). *Software Engineering: A Practitioner’s Approach*, 7<sup>th</sup> Ed.
- SEI - Software Engineering Institute (2010). CMMI for Development (CMMI-DEV).
- SOFTEX - Associação para Promoção da Excelência do Software Brasileiro. (2012). MPS.Br - Melhoria do Processo do Software Brasileiro, Guia Geral MPS de Software.
- SOFTEX Recife - Centro de Excelência em Tecnologia de Software do Recife. (2011). MPT.Br - Melhoria do Processo de Teste Brasileiro, Guia de Implementação, N.1 e 2.
- Sommerville, I. (2007). *Engenharia de Software*, 8<sup>a</sup> Ed.
- TMMi Foundation (2012). Test Maturity Model integration (TMMi).
- Villela, K. (2004). Definição e Construção de Ambientes de Desenvolvimento de Software Orientados à Organização. Tese de D.Sc., COPPE/UFRJ, Brasil.

# Indicações de Abordagens para Rastreabilidade de Requisitos no contexto do MR-MPS-SW por meio de uma Revisão Sistemática da Literatura

Paulo Robson Campelo Malcher<sup>1</sup>, Sandro Ronaldo Bezerra Oliveira<sup>1</sup>

<sup>1</sup>Programa de Pós-Graduação em Ciência da Computação (PPGCC) – Universidade Federal do Pará (UFPA) - Rua Augusto Corrêa, 01 – Guamá – Belém - PA – Brasil

prcmalcher@gmail.com, srbo@ufpa.br

**Abstract.** *This paper aims to provide a study of approaches to requirements traceability in the context of software projects, found by conducting a systematic review of the literature, and based on this study carry an indication of which approaches can be better implemented in the context of MR-MPS-SW. This way, it is intended to contribute to the dissemination of the techniques, tools and models found in the literature about requirements traceability and support organizations in implementing this practice in the above model.*

**Resumo.** *Este trabalho objetiva fornecer um estudo sobre abordagens de rastreabilidade de requisitos no contexto de projetos de software, encontradas por meio da realização de uma revisão sistemática da literatura, e com base neste estudo realizar a indicação de quais abordagens podem ser melhor implementadas no contexto do MR-MPS-SW. Com isto, pretende-se contribuir com a disseminação das técnicas, ferramentas e modelos encontrados na literatura sobre rastreabilidade de requisitos e auxiliar organizações na implementação dessa prática no modelo citado.*

## 1. Introdução

Com o objetivo de tornar as organizações mais competitivas nos negócios, as respostas das Tecnologias de Informação têm a necessidade de serem cada vez mais rápidas, porém a qualidade do que é entregue, ou influencia essa entrega, é também um fator muito importante para manter essa competitividade [Ferreira, 2010]. Nesse contexto, a Engenharia de Requisitos (ER), segundo Pressman (2010), surgiu a fim de ajudar os engenheiros de software a compreender melhor o problema por meio de um conjunto de tarefas que levam a um entendimento de qual será o impacto do sistema sobre o negócio, do que os *stakeholders* necessitam e como será a interação dos usuários finais com o sistema. Dentre as atividades que compõem a engenharia de requisitos, pode-se dividi-las em seis áreas genéricas: Levantamento, Análise, Documentação, Verificação, Validação e o Gerenciamento dos Requisitos [Kotonya e Sommerville, 1998].

Este trabalho contempla a área de gerenciamento de requisitos, que abrange toda a compreensão e o controle de mudanças nos requisitos de sistema que é tida como essencial para o sucesso de projetos de software e está presente em padrões e modelos de qualidade, como a Norma ISO/IEC 12207 [ABNT, 2009] e os modelos CMMI [SEI, 2010] e MR-MPS-SW [SOFTEX, 2012]. A Rastreabilidade de Requisitos é uma das práticas que compõem a Gerência de Requisitos (GRE) [SEI, 2010] e é tida como uma tarefa indispensável nesse processo e um fator de qualidade no que diz respeito ao desenvolvimento

de software. Para Gotel e Finkelstein (1997), ela é definida como a capacidade de descrever e seguir o ciclo de vida de um requisito em ambas as direções, da origem ou da implementação, passando entre todas as especificações relatadas.

Outro ponto a ser destacado neste trabalho é a utilização do método de Revisão Sistemática da Literatura (RSL), que é um dos principais métodos da Engenharia de Software Baseada em Evidências (*Evidence-based Software Engineering* – EBSE) e é classificado como estudo secundário, já que depende dos estudos primários utilizados para revelar evidências e construir conhecimento [Dybå et al., 2007; Oates e Capper, 2009; Travassos, 2007 *apud* Costa, 2010]. Costa (2010) ainda afirma que a essência do paradigma baseado em evidência é coletar e analisar sistematicamente todos os dados disponíveis sobre determinado fenômeno para obter uma perspectiva mais completa e mais ampla do que se pode captar por meio de um estudo individual.

Nesse sentido, realizar uma RSL que tenha como foco descobrir quais as abordagens descritas na literatura para auxiliar a rastreabilidade de requisitos, pode ser de grande valia para um maior conhecimento sobre a área. Nesse sentido, a proposta desse trabalho é apresentar abordagens (técnicas, modelos e ferramentas) encontradas na literatura, por meio da revisão sistemática, para o apoio à rastreabilidade de requisitos e realizar uma análise quanto a sua implementação no contexto do MR-MPS-SW.

O restante deste artigo está organizado da seguinte forma: na Seção 2 é apresentado como se deu o planejamento da RSL realizada; a condução da RSL e a extração das abordagens de apoio à rastreabilidade são apresentadas na Seção 3; na Seção 4 é feita uma indicação dessas abordagens a serem implementadas no contexto do MR-MPS-SW; por fim, na Seção 5 são apresentadas as conclusões deste trabalho.

## 2. A Revisão Sistemática da Literatura

A revisão sistemática segue uma metodologia específica e peculiar, apresentada por Kitchenham (2007), onde define que revisões sistemáticas começam pela definição de um protocolo de revisão e resume as etapas de uma revisão sistemática em três fases principais: planejamento, condução e apresentação. Nesta seção serão apresentadas definições referentes à fase de planejamento. O protocolo na íntegra pode ser encontrado em [http://spider.ufpa.br/projetos/spider\\_rastreabilidade/SPIDER\\_ProtocoloRevisao.pdf](http://spider.ufpa.br/projetos/spider_rastreabilidade/SPIDER_ProtocoloRevisao.pdf).

### 2.1. Objetivo da Revisão

A revisão sistemática realizada teve o objetivo de identificar abordagens para apoiar a atividade de Rastreabilidade de Requisitos no contexto de projetos de software, no período de 2003 a 2013, no que tange a processos, modelos, metodologias, técnicas, ferramentas e afins. Desta forma, tem-se a seguinte estrutura para o objetivo, conforme proposto em Santos (2010):

- **Analisar:** relatos de experiência e publicações científicas através de um estudo baseado em revisão sistemática;

- **Com o propósito de:** identificar abordagens para apoiar atividades de Rastreabilidade de Requisitos;
- **Com relação à:** definição e uso de processos, *frameworks*, metodologias, ferramentas e demais instrumentos empregados para a implantação e execução das atividades de Rastreabilidade de Requisitos em organizações de desenvolvimento de software;
- **Do ponto de vista:** de pesquisadores e organizações desenvolvedoras/mantenedoras de software;
- **No contexto:** acadêmico e industrial.

## 2.2. Questões de Pesquisa

Com base no objetivo de investigação desta revisão, foi definida a seguinte questão de pesquisa, a qual guiou esta Revisão Sistemática da Literatura:

**(Q1) Quais as abordagens existentes para apoiar as atividades de Rastreabilidade de Requisitos?**

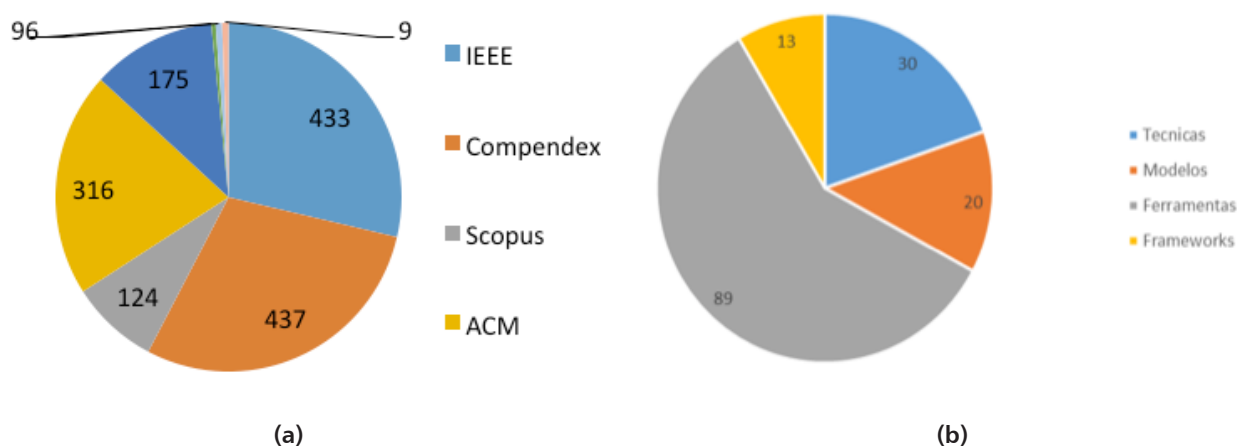
## 2.3. Seleção de Fontes

Com base em critérios de seleção pré-estabelecidos e nas restrições da pesquisa, foram selecionadas as seguintes fontes de pesquisa onde foram realizadas as buscas dos estudos primários, de acordo com a disponibilidade de pesquisa e a relevância para área de Qualidade de Software: IEEEExplore Digital Library; El Compendex; ISI Web of Knowledge; ACM; Scopus; Anais do WAMPS – Workshop Anual do MPS; Anais do WER – Workshop de Engenharia de Requisitos; Anais do SBQS – Simpósio Brasileiro de Qualidade de Software.

Vale resaltar que outras definições realizadas durante a fase do planejamento foram omitidas desse trabalho por não estarem diretamente ligadas aos objetivos do mesmo, tais como: a definição de *strings* de busca; os critérios de inclusão e exclusão dos estudos; os critérios de avaliação de qualidade; e a definição dos processos de seleção, avaliação e extração dos resultados da revisão.

## 3. Extração de Dados

Em acordo com o que foi definido no planejamento da revisão, no que diz respeito à busca realizada nas fontes selecionadas, esta foi executada em cada fonte com uma *string* de busca específica. A partir disso, obteve-se um total de 1509 estudos, no qual: 433 trabalhos foram identificados na IEEE; 437 na El Compedex; 124 na Scopus; 316 na ACM; 175 na ISI Web Knowledge; 6 no SBQS; 9 no WAMPS; e, por fim, 9 foram identificados no WER. Como pode ser visualizado no gráfico da Figura 1(a). E na Figura 1(b) a distribuição das abordagens.



**Figura 1. (a) Distribuição dos Estudos Retornados por Fonte de Pesquisa; (b) Distribuição de Estudos Retornados por Tipo de Abordagens**

A partir do processo de seleção definido no protocolo de revisão, foram selecionados os estudos primários por meio da leitura do título, palavras-chaves, resumo e conclusão de cada estudo retornado, e em seguida foram aplicados os critérios de inclusão e exclusão. Após o processo de seleção, o número de estudos foi bastante reduzido, sendo selecionados 411 estudos. A evolução em números do processo de seleção de estudos primários é apresentada no Quadro 1.

**Quadro 1. Evolução dos Estudos Primários após o Processo de Seleção.**

Seleção de Estudos Primários				
Fontes	Estudos Retornados	Estudos Excluídos		Estudos Incluídos
		Duplicados	Outros	
IEEE	433	0	269	164
El Compedex	437	228	99	110
Scopus	124	95	13	16
ACM	316	61	182	73
ISI Web Knowledge	175	110	41	24
SBQS	6	0	0	6
WAMPS	9	0	0	9
WER	9	0	0	9
Total	1509	0	0	411

Após a seleção dos estudos primários, estes foram lidos em sua totalidade a fim de identificar abordagens que possam auxiliar a rastreabilidade de requisitos. Nesse intuito, foram extraídas as principais abordagens a serem definidas nas subseções a seguir.

### 3.1. Técnicas

Falbo (1998) diferencia método e técnica definindo que um método é um procedimento sistemático para a realização de uma ou mais atividades, definindo passos e heurísticas, já uma técnica é também um procedimento para a realização de uma atividade, contudo, menos rígido e detalhado. Dentre os estudos selecionados foram identificadas cerca de 30 técnicas de apoio à rastreabilidade de requisitos disponíveis na literatura, onde é possível dividi-las em técnicas de visualização de rastreabilidade e técnicas de geração de *links* de rastreabilidade. Dentre as técnicas de geração de *links* de rastreabilidade, pode-se dividi-las em técnicas manuais, semiautomáticas e automáticas. O que diferenciam essas técnicas é a forma com que os *links* de rastreabilidade são gerados, por exemplo: nas técnicas ditas manuais, os *links* são gerados diretamente pelos interessados; já nas semiautomáticas, os *links* de rastreabilidade são geradas de forma automática, porém necessitam da aprovação dos interessados; por fim, as técnicas automáticas são as que não necessitam de fatores externos para a geração dos *links* de rastreabilidade. Por questão de espaço, foram escolhidas as técnicas com mais ocorrência na literatura, segundo a revisão realizada, para serem apresentadas nesse trabalho.

#### 3.1.1. Traceability Matrix

A técnica de matriz de rastreabilidade é a que pode ser definida como de visualização e que pode usar tanto as técnicas de geração de *link* de rastreabilidade da forma manual, semiautomática e automática, porém pode-se verificar que na maioria dos estudos é definida de forma manual, mesmo com o auxílio de algum ferramental em sua implementação.

No resultado da Revisão Sistemática da Literatura realizada foi a técnica tida como manual mais citada nos estudos. Esta técnica utiliza uma matriz para gerar a rastreabilidade bidirecional de forma vertical ou horizontal. Uma limitação desta técnica está no fato de relacionar os artefatos sempre em pares, o que dificulta uma visão mais ampla, quando se analisa o impacto de mudanças, além de ser uma técnica onerosa em sua implementação e manutenção. Hayes (2005) afirma que as Matrizes além de auxiliar nas tarefas referentes à rastreabilidade de requisitos, elas também podem auxiliar na análise de completude do software e em seu teste. Li *et al.* (2013) corroboram com a afirmação de Hayes quando afirma que é uma técnica fundamental para muitas técnicas de segurança, tais como a análise de criticidade, a análise de impacto de mudanças e testes de regressão.

#### 3.1.2. Graphs

Grafos são mais uma forma de visualização de rastreabilidade, que também podem ser definidas na literatura como árvores, que geralmente fazem o uso da geração manual de *links*. Entretanto, existem trabalhos como os de Chen (2010) e de Schwarz (2009) que apresentam a geração automática de grafos. A ideia dessa técnica é utilizar os artefatos a serem relacionados como os vértices do grafo e as arestas como *links*, nesse caso podendo ou não acrescentar a dependência dos mesmos por meio da direção das arestas.



### 3.1.3. *Vector Space Model (VSM)*

Este tipo de técnica faz parte da área denominada Recuperação de Informação (*Information Retrieval* – IR), que é um método que segundo Narayan (2011) possui técnicas que têm se mostrado muito eficazes para auxiliar a geração automática de *links* de rastreabilidade, reduzindo o tempo necessário para gerar o mapeamento de rastreabilidade. Em particular, vários pesquisadores têm aplicado recentemente Técnicas de Recuperação de Informação [Baeza-Yates e Ribeiro-Neto, 1999; Deerwester *et al.* 1990; Harman, 1992 *apud* De Lucia, 2007] para o problema da recuperação de *links* de rastreabilidade entre artefatos de diferentes tipos.

Essa técnica apresenta sua utilidade para a rastreabilidade, dividindo os itens de rastreabilidade em um modelo de espaço vetorial. VSM é uma abordagem padrão que calcula a similaridade do cosseno entre uma consulta e um documento, cada um dos quais é representado como um vetor de termos ponderados [Gibiec, 2010]. Hayes *et al.* (2003) utilizam o modelo de espaço vetorial para recuperar *links* de rastreabilidade entre os requisitos e comparar os resultados obtidos através da aplicação de diferentes variantes do modelo básico: em particular, o uso de palavras-chave do documento é mostrado como um meio para melhorar os resultados do modelo de espaço vetorial.

### 3.1.4. *Latent Semantic Information (LSI)*

LSI é uma técnica de recuperação de informação, proposta pela primeira vez no trabalho de Deerwester *et al.* (1990), que é baseada no modelo de espaço vetorial (VSM), onde assume-se que existe uma estrutura subjacente ou latente no uso da palavra para cada conjunto de documentos. LSI utiliza técnicas estatísticas para estimar essa estrutura latente. A descrição dos termos e documentos com base na estrutura semântica latente subjacente é usada para representar e recuperar informações. Esta representação pode ser usada para comparar e indexar documentos usando uma variedade de medidas de similaridade. Ao aplicar LSI ao código-fonte e sua documentação interna associada (ou seja, comentários), componentes candidatos podem ser comparados com relação a essas medidas de similaridade. Por ser uma técnica de IR baseada em VSM, a técnica de LSI é uma técnica muito citada na literatura no que diz respeito a técnicas automáticas como nos trabalhos de De Lucia (2008), Jiang (2007), e Maletic e Marcus (2001).

### 3.1.5. *Probabilistic Approaches*

Nesse tipo de técnica são utilizados modelos de redes probabilísticas para realizar a recuperação de informação. Esses modelos, segundo Gibiec (2010), têm sido demonstrados por meio de experimentação prévia para executar de forma equivalente a outros modelos de recuperação de informação. Seu uso envolve três atividades de busca principais que são: o pré-processamento do texto; a probabilidade de *links*; e a análise dos resultados. Antoniol (2002) afirma que um modelo probabilístico IR calcula a probabilidade de que dois documentos estão relacionados e usa a probabilidade calculada para classificar *links* gerados. *Jensen-Shannon Model (JSM)* foi proposto por Abadi *et al.* (2008), e é impulsionado por uma abordagem e hipótese técnica de teste probabilístico. JSM representa cada um documento por meio de uma distribuição de probabilidades, isto é, uma matriz de termo-a-documento normalizado.

## 3.2. Modelos

Como resultados da extração de dados da revisão sistemática da literatura, foram identificados cerca de 20 modelos que envolvem a rastreabilidade de requisitos. Alguns deles focados na rastreabilidade de requisitos e outros de uma forma mais ampla envolvendo a gerência de requisitos.

### 3.2.1. *Traceability Information Model (TIM)*

É um modelo básico sobre rastreabilidade, muito referenciado na literatura por ser uma visão conceitual de um projeto de implantação de uma estratégia de rastreabilidade. É definido como um modelo genérico que pode fazer uso de diferentes abordagens em sua implementação, utilizando-se da UML para sua implantação. Basicamente consiste na definição de tipos de entidades, de artefatos e das relações de rastreabilidade entre esses artefatos. Ele também define quais tipos de artefatos que podem ser rastreados, qual o tipo de relação entre esses artefatos e o tipo de relação de rastreabilidade. O Centro de Excelência para Rastreabilidade de Software [CoEST, 2014] afirma que o uso deste modelo é considerado como uma das melhores práticas para a gestão e planejamento de rastreabilidade em todo o ciclo de vida de um projeto, que pode ser enfatizado nos trabalhos de Mäder *et al.* (2009), Cleand-Huang *et al.* (2012) e Mäder *et al.* (2013).

### 3.2.2. *Ramesh's Traceability Model*

Este modelo foi proposto por Ramesh (2001), sendo um modelo de informação de rastreabilidade (TIM), que descreve toda a estratégia da prática de rastreabilidade em uma organização. Este modelo identifica a informação de rastreabilidade, os requisitos, os raciocínios, a alocação de requisitos e de recursos. Como características relevantes deste modelo destacam-se: o relacionamento entre requisitos e suas derivações; entre estes com os documentos de origem; entre requisitos e documentos de testes; entre requisitos e componentes do sistema; entre requisitos e *stakeholders*; dentre outros. O modelo permite representar as solicitações de mudanças, porém não comporta a representação dos impactos gerados e não possibilita representar versionamento e configuração de granularidade.

## 3.3. Ferramentas

Foram encontradas na literatura cerca de 90 ferramentas que possuem alguma relação com rastreabilidade de requisitos. As ferramentas encontradas implementam diferentes técnicas com relação à geração de *links* e da visualização da rastreabilidade em diversos contextos, alcançando a pré e pós rastreabilidade, a rastreabilidade horizontal e vertical, as formas para frente e para trás, e por fim a bidirecionalidade da rastreabilidade.

### 3.3.1. *Requirements TRacing On target (RETRO)*

É uma ferramenta desenvolvida no trabalho de Hayes *et al.* (2006), que utiliza a técnica de geração de *link* de rastreabilidade automática por meio de algumas técnicas de Recuperação de Informação. A mais citada na literatura é a técnica de Modelo de Espaço Vetorial (VSM), o que resulta na geração automática de matrizes de rastreabilidade para a visualização da prática. Hayes realizou um estudo de caso na ferramenta o que comprovou melhor aproveitamento em seu uso quando comparada a forma manual de realizar a rastreabilidade.

### 3.3.2. *DOORS*

Telelogic DOORS [DOORS, 2005] é uma ferramenta comercial para o gerenciamento de requisitos que suporta vários tipos de objetos de requisitos, tipos de atributos e a geração de *links* de rastreabilidade. DOORS utiliza uma matriz de rastreabilidade e árvores com *hiperlinks* para a visualização e a manutenção dos *links* de rastreabilidade. Pode-se definir a ferramenta, no contexto da rastreabilidade, como uma ferramenta que utiliza a técnica de geração de *links* de rastreabilidade de forma semiautomática, pois as relações são identificadas manualmente, mas as ligações entre os elementos que são relacionados são geradas automaticamente pela ferramenta com base nessas relações.

### 3.3.3. *Caliber RM*

CaliberRM (2014) é uma ferramenta para o gerenciamento de requisitos, desenvolvida pela Borland. A ferramenta mantém as informações de rastreabilidade entre os vários requisitos e também entre artefatos em outras ferramentas. Ela também oferece uma série de maneiras de manter uma visão geral de todos os requisitos e combina esses recursos para analisar o impacto das propostas de mudança. Ela também permite que os requisitos de software conectem-se a uma variedade de artefatos de todo o ciclo de vida. A ferramenta utiliza a técnica manual para geração de *link* de rastreabilidade e para a visualização dos *links* da rastreabilidade utiliza matriz de rastreabilidade e grafos.

### 3.3.4. *Poirot*

Poirot [Lin, 2006] é uma ferramenta baseada na *web* de apoio à rastreabilidade. Utiliza um modelo de rede probabilística como técnica de geração automática de *links* de rastreabilidade, ou seja, utiliza uma das técnicas de recuperação de informação. Provê a rastreabilidade entre os requisitos, elementos de *design*, código e outros artefatos armazenados na ferramenta. Utiliza como forma de visualização da rastreabilidade tanto um formato textual quanto um visual, mas especificamente por *hiperlinks*. Em muitos trabalhos, a exemplo dos trabalhos de Cleland-Huang (2007) e Nair (2013), a ferramenta sempre é citada como referência quando se trata da utilização de abordagens probabilísticas para o apoio à rastreabilidade.

### 3.3.5. *Rational RequisitePro*

É uma ferramenta para o gerenciamento de requisitos desenvolvida pela IBM, que fornece a possibilidade de realizar o relacionamento entre requisitos, casos de uso e casos de teste. Além do que, quando ocorre uma mudança em um requisito ela identifica os artefatos de software que são afetados. Um exemplo de aplicação desenvolvida para a utilização dessa matriz pode ser encontrado no RequisitePro [IBM 2010]. No contexto da rastreabilidade de requisitos, ela é reconhecida por diversos autores da área como uma ferramenta de apoio, por exemplo nos trabalhos de Cleland-Huang (2002; 2007) e Hayes (2003; 2005).

## 4. Indicação das Abordagens para Implementar o MR-MPS-SW

Como já apresentado, a prática da rastreabilidade de requisitos no MR-MPS-SW está inserida no processo de GRE do nível de maturidade G. Dentre os resultados esperados desse processo, os que tem ligação direta com a rastreabilidade são os GRE3, GRE4, GRE5. Esses resultados são apresentados nas subseções seguintes, assim como, a indicação das abordagens encontradas na literatura que melhor podem satisfazer sua implementação.

### 4.1. GRE 3 - A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida

Este resultado segundo a Softex (2013), indica a necessidade de se estabelecer um mecanismo que permita rastrear a dependência entre os requisitos e os produtos de trabalho, e que a rastreabilidade bidirecional deve acontecer tanto de forma horizontal quanto vertical. Nesse resultado, dois conceitos principais em rastreabilidade são citados para se alcançar este resultado: o fato de se criar um mecanismo de rastreio; e que a bidirecionalidade deve acontecer tanto de forma horizontal quanto vertical. No contexto das abordagens encontradas e apresentadas neste trabalho, quanto a definição da forma horizontal e vertical da rastreabilidade, pode-se melhor utilizar qualquer técnica de geração de *links* que fazem da recuperação de informação, e quanto à visualização da rastreabilidade, a matriz de rastreabilidade ainda se mostra mais satisfatória para uma visão mais ampla sobre a rastreabilidade. Quanto aos modelos apresentados, estes auxiliam no estabelecimento da estratégia de rastreabilidade, que está fortemente relacionada à satisfação deste resultado. Com relação ao apoio ferramental disponível para o atendimento deste resultado, pode-se destacar a RETRO.

### 4.2. GRE 4 - Revisões em planos e produtos de trabalho do projeto são realizadas visando a identificar e corrigir inconsistências em relação aos requisitos

Este resultado segundo a Softex (2013), define que a consistência entre os requisitos e os produtos de trabalho do projeto deve ser avaliada e os problemas identificados devem ser corrigidos. Nesse sentido, o auxílio que a rastreabilidade pode proporcionar à implementação deste resultado está na definição do relacionamento entre requisitos e outros produtos de trabalho, bem como ela pode auxiliar na identificação da consistência dessas relações. Com base nesse auxílio que a rastreabilidade

pode prover e nas abordagens encontradas e apresentadas nesse trabalho, pode-se verificar que com base nas revisões realizadas na rastreabilidade e em seus *links* pode-se por meio da visualização desses *links* identificar as inconsistências e prover suas correções. Uma técnica que possibilita uma melhor visualização desse tipo é a que envolve a implementação de grafos. Quanto às ferramentas, podem-se citar a DOORS e CaliberRM.

#### **4.2. GRE 5 - Mudanças nos requisitos são gerenciadas ao longo do projeto**

Este resultado segundo a Softex (2013), afirma que requisitos adicionais podem ser incorporados no projeto, requisitos podem ser retirados do projeto e/ou mudanças podem ser feitas nos requisitos já existentes. Desta forma, é necessário tratar essas mudanças e analisar o seu impacto. Com relação a mudanças que podem ocorrer, atualizações devem ser feitas na estrutura de rastreabilidade estabelecida, o que irá diferir entre a forma de geração de *links* de rastreabilidade está na maneira como a mesma acontece: se for de forma manual, deve-se ser feita essa mudança também de forma manual; no caso da utilização de forma semiautomáticas ou automáticas, o impacto geralmente é identificada automaticamente. Um bom exemplo para o auxílio nesse resultado esperado é a ferramenta RequisitePro, que alerta quando uma mudança ocorre e qual o seu impacto.

### **5. Conclusão**

Neste trabalho foi apresentado um resumo, devido ao espaço reduzido deste artigo, da aplicação de uma revisão sistemática da literatura com o objetivo de encontrar abordagens de apoio para a prática de rastreabilidade de requisitos. Adicionalmente, foram realizadas indicações de quais abordagens podem melhor auxiliar na implementação dos resultados esperados do processo de GRE, que possuem alguma relação com a rastreabilidade de requisitos.

Como continuação deste trabalho, pode-se destacar a elaboração de um catálogo dessas abordagens para tentar alcançar um dos desafios propostos no Primeiro Workshop de Grandes Desafios para a Rastreabilidade (GCW'06), encontrado na subárea de conhecimento sobre a rastreabilidade. O princípio deste desafio é de que a rastreabilidade é um fator crítico para o sucesso de projetos de software, porém não há um consenso sobre as melhores técnicas e métodos para sua utilização, tendo como desafio dessa subárea a criação de um banco de conhecimento com melhores práticas, terminologias, abordagens e estudos de caso sobre a rastreabilidade. Outro trabalho em andamento é a escrita de um artigo para um periódico com todos os detalhes da revisão sistemática realizada.

### **6. Agradecimentos**

Este trabalho recebe apoio financeiro da CAPES a partir da concessão de bolsa institucional de mestrado. Este projeto é parte do Projeto SPIDER-UFP.

## Referências

- ABNT - Associação Brasileira De Normas Técnicas, (2009) "NBR ISO/IEC 12207:2009 - Engenharia de Sistemas de Software - Processos de Ciclo de Vida de Software".
- Abadi, A. Nisenson, M. and Simionovici, Y. (2008) "A Traceability Technique for Specifications". Proc. 16th IEEE Int. Conf. Program Comprehension, pp. 103-112, 2008.
- Antoniol, G. *et al.* (2002) "Recovering traceability links between code and documentation". IEEE Transactions on Software Engineering, 28(10):970–983.
- CaliberRM – Gerenciamento de Requisito para Engenharia de Software, Disponível em: <http://www.borland.com/br/products/caliber/index.htm>. Acesso em: 28/01/2007
- Chen, X. (2010) "Extraction and visualization of traceability relationships between documents and source code". Proceedings of the IEEE/ACM international conference on Automated software engineering - ASE '10, 505.
- Cleland-Huang, J., Chang, C. K. & Ge, Y. (2002) "Supporting event based traceability through high-level recognition of change events. In Proceedings of the 26th International Computer Software and Applications Conference on Prolonging Software Life: Development and Redevelopment, Washington, DC, IEEE Computer Society, pp. 595-602.
- Cleland-Huang, J. *et al.* (2007). "Best Practices for Automated Traceability". Computer (Volume: 40 , Issue: 6 ), (June), 27–35.
- Cleland-Huang, J. *et al* (2012). "Breaking the big-bang practice of traceability: Pushing timely trace recommendations to project stakeholders. (RE)". 20th IEEE International Requirements Engineering Conference, 231–240.
- CoEST: Center of excellence for software traceability. Disponível em: <http://www.CoEST.org>. Acesso em: 20/09/2014.
- Costa, C. S. (2010). "Uma abordagem baseada em evidências para o gerenciamento de projetos no desenvolvimento distribuído de software". Dissertação de Mestrado Universidade Federal de Pernambuco, Recife, PE, Brasil.
- Deerwester, S. *et al.* (1990) "Indexing by latent semantic analysis". J. Society for Information Science 41(6) 391–407.
- De Lucia, A. *et al.* (2007) "Recovering traceability links in software artifact management systems using information retrieval methods". ACM Transactions on Software Engineering and Methodology.
- De Lucia, A., Oliveto, R., & Tortora, G. (2008). "ADAMS Re-Trace : Traceability Link Recovery via Latent Semantic Indexing". ICSE'08. ACM/IEEE 30th International Conference on Software Engineering, 839 – 842
- DOORS. (2005) "Dynamic Object Oriented Requirements System". Disponível: em: <http://www.telelogic.com/doors>. Acesso em: 27/09/2014.
- Falbo, R. A. (1998) "Integração de Conhecimento em um Ambiente de Desenvolvimento de Software", Orientadora Ana Regina Cavalcanti da Rocha. Tese de Doutorado, COPPE/UFRJ, Rio de Janeiro-RJ.

- Ferreira, P. (2010). "Rastreabilidade de ponto de função". Dissertação de Mestrado. UNIRIO, Rio de Janeiro, RJ, Brasil.
- Gibiec, M., Czauderna, A., & Cleland-Huang, J. (2010). "Towards minin replacement queries for hard-to-retrieve traces". Proceedings of the IEEE/ACM international conference on Automated software engineering - ASE '10, 245.
- Gotel, O. and Finkelstein, A. (1997) "Extended requirements traceability: Results of an industrial case study". In Proceedings of the 3rd IEEE International Symposium on Requirements Engineering, Washington, DC, p. 169.
- Hayes, J.H., Dekhtyar, A. & Osborne, J. (2003) "Improving requirements tracing via information retrieval". In Proceedings of the 11th IEEE International Conference on Requirements Engineering, Washington, DC, p. 138.
- Hayes, J. H., Dekhtyar, A. & Sundaram, S. K. (2005) "Improving after-the-fact tracing and mapping: Supporting software quality predictions". IEEE Software 22(6) 30-37.
- Hayes, J. H., Dekhtyar, A. & Sundaram, S. K., (2006) "Advancing candidate link generation for requirements tracing: the study of methods". IEEE TSE, 32/1, pp. 4-19.
- IBM (2010). IBM rational requisitepro software. Disponível em: <http://www-01.ibm.com/software/awdtools/reqpro/>. Acesso em 27/09/2014.
- Jiang, H. *et al.* (2007) "Traceability Link Evolution Management with Incremental Latent Semantic Indexing". 31st Annual International Computer Software and Applications Conference, (Compsac), 309–316.
- Kitchenham, B. (2007) "Guidelines for performing Systematic Literature Reviews". In Software Engineering, Technical Report EBSE-2007-01, Departament of Computer Science Keele University, Keele.
- Kotonya, G., Sommerville, I., (1998) "Requirements engineering: processes and techniques. Chichester". England: John Wiley.
- Li, W., & Hayes, J. H. (2013). "Traceability Challenge 2013: Query + Enhancement for Semantic Tracing ( QuEST )". (TEFSE), 2013 International Workshop on Traceability in Emerging Forms of Software Engineering, 95–99.
- Lin J, *et al* (2006) "Poirot: a distributed tool supporting enterprise-wide automated traceability". In: 14th IEEE international requirements engineering conference (RE 2006), pp 356–357
- Mäder, P., Gotel, O. C. Z., & Philippow, I. (2009). "Getting back to basics: Promoting the use of a traceability information model in practice". TEFSE '09. ICSE Workshop on Traceability in Emerging Forms of Software Engineering, 21–25.
- Mäder, P. *et al.* (2013). "Strategic Traceability for Projects". Software, IEEE, 58–66.
- Maletic, J. I. & Marcus, A. (2001) "Supporting program comprehension using semantic and structural information". In Proc. of 23rd International Conference on Software Engineering, pages 103–112, Toronto.

- Nair, S., de la Vara, J. L., & Sen, S. (2013). "A review of traceability research at the requirements engineering conferencere@21". (RE), 2013 21st IEEE International Requirements Engineering Conference, 222–229. doi:10.1109/RE.2013.6636722
- Narayan, N., Bruegge, B., Delater, A., & Paech, B. (2011). "Enhanced traceability in model-based CASE tools using ontologies and information retrieval". Fourth International Workshop on Managing Requirements Knowledge (MARK), 24–28.
- Pressman, R. S. (2011) "Engenharia de Software: Uma Abordagem Profissional". Mcgraw Hill.
- Ramesh, B. and Jarke, M., (2001) "Toward Reference Models for Requirements Traceability". IEEE Transactions on Software Engineering, vol. 27, no. 1, 2001, pp. 58-93.
- Santos, G. (2010) "Revisão Sistemática, Mini-Curso". Simpósio Brasileiro de Qualidade de Software - SBQS 2010, Belém – PA.
- SEI. Capability Maturity Model Integration. (2010) "CMMI for Development". Version 1.3. Carnegie Mellon, USA
- Softex. (2012). "Melhoria do Processo de Software Brasileiro (MPS.BR) - Guia Geral 2012". Disponível em: [http://www.softex.br/wp-content/uploads/2013/07/MPS.BR\\_Guia\\_Geral\\_Software\\_20121.pdf](http://www.softex.br/wp-content/uploads/2013/07/MPS.BR_Guia_Geral_Software_20121.pdf). Acesso em: 24/01/2014.
- Softex. (2013) "Guia de Implementação – Parte 1: Fundamentação para Implementação do Nível G do MR-MPS". Disponível em: [http://www.softex.br/wp-content/uploads/2013/07/MPS.BR\\_Guia\\_de\\_Implementacao\\_Parte\\_1\\_2013.1.pdf](http://www.softex.br/wp-content/uploads/2013/07/MPS.BR_Guia_de_Implementacao_Parte_1_2013.1.pdf). Acesso em: 24/09/2014.
- Schwarz, H., Ebert, J., & Winter, A. (2009). "Graph-based traceability: a comprehensive approach. Software & Systems Modeling". 9(4), 473–492.





# Experiência de Desenvolvimento e Utilização do Método de Avaliação CERTICS

Sonia T. Maintinguer<sup>(2)</sup>, Clenio F. Salviano<sup>(1)</sup>, Wesley Marinho<sup>(2)</sup>, Marcia Martinez<sup>(2)</sup>, Maysa Dias Rezende<sup>(2)</sup>, Adalberto N. Crespo<sup>(2)</sup>, Eduardo Paulo de Souza<sup>(2)</sup>, Davi C. Silva<sup>(2)</sup>, Alan Raldi<sup>(2)</sup>, Giancarlo N. Stefanuto<sup>(2)</sup>, Angela M. Alves<sup>(1)</sup>

<sup>(1)</sup> Centro de Tecnologia da Informação Renato Archer - CTI

Rodovia D. Pedro I, km 143.6, CEP 13069-90, Campinas, SP, Brasil

<sup>(2)</sup> Fundação de Apoio à Capacitação em Tecnologia da Informação – Facti

Rodovia D. Pedro I, km 143.6, CEP 13069-901, Campinas, SP, Brasil

{clenio.salviano, angela.maria}@cti.gov.br; soniamaint, weslei, mmartinezcorso, rezende.maysa, adalberto.crespo, edupaulo2012, davi.certics, alanraldi, gianstefanuto}@gmail.com

**Abstract.** *The CERTICS Assessment Method was developed as a software process assessment method following Standards ISO/IEC 15504-2 and ISO/IEC 15504-7 requirements for software process assessment. This method guides the assessment of a certain software related processes to determine whether the software is a result of technological innovation and development carried out in the Country. The method is part of the CERTICS Methodology. The method presents three main innovations related to other process assessment and improvement initiatives: covering all assessment activities, a software platform supporting all activities the definition of a starting phase for the software companies to explore the CERTICS Methodology. These innovations can be used in other process assessment methods.*

**Resumo.** *O Método de Avaliação da CERTICS foi desenvolvido como um método de avaliação de processo de software conforme os requisitos das Normas ISO/IEC 15504-2 e ISO/IEC 15504-7 para o processo de avaliação de processo. Este método orienta a avaliação os processos relacionados a um determinado software para determinar se o software é resultado de desenvolvimento e inovação tecnológica realizados no País. O método é parte da Metodologia CERTICS. O método apresenta três inovações principais em relação a outras iniciativas de melhoria e avaliação de processo: a abrangência de todas as atividades de uma avaliação, o suporte de uma plataforma de software para todas as atividades e a definição de uma fase inicial para exploração da Metodologia CERTICS pelas empresas de software. Estas inovações podem ser utilizadas em outros métodos de avaliação de processo.*

## 1. Introdução

Avaliações de processos e de maturidade organizacionais são atividades relevantes para a Melhoria de Processo. A série de Normas ISO/IEC 15504 estabelece uma estrutura de medição, requisitos, guias e exemplos de modelos e métodos de avaliação de processos [15504-2 2008] [15504-7 2008] que podem ser utilizados para o desenvolvimento de diferentes modelos e métodos para diferentes objetivos e domínios. Esses requisitos procuram garantir que os resultados da avaliação sejam objetivos, imparciais, consistentes, repetíveis e representativos em relação aos processos avaliados.

Este artigo apresenta um relato de experiência do projeto, desenvolvimento e uso do Método de Avaliação da CERTICS. O Método de Avaliação foi desenvolvido junto com o Modelo de Referência da CERTICS como parte da implantação de um instrumento de política pública do Brasil para software. Este instrumento caracteriza software resultado de desenvolvimento e inovação tecnológicas realizados no País para se beneficiarem de uma margem de preferência em licitações.

O Desenvolvimento deste método foi orientado por dois princípios: (a) identificação e utilização de melhores práticas do estado da arte, e (b) identificação de desafios e características específicas e desenvolvimento de soluções inovadoras para estes desafios e características específicas.

Para o primeiro princípio, foi identificada a Norma ISO/IEC 15504 e utilizada a experiência do CTI Renato Archer em melhoria e avaliação de processo, com destaque para a participação no desenvolvimento da Norma ISO/IEC 15504; experiência como avaliador líder de métodos alinhados com a Norma, incluindo o método SCAMPI [SEI 2011]; e desenvolvimento e utilização de métodos de avaliação de processos, que seguiram os requisitos da Norma, incluindo o método MARES [Anacleto et al 2004] e MA-MPS [Weber et al 2005], ou que utilizaram os requisitos como referência, incluindo o método PRO2PI-WORK [Martinez e Salviano 2012].

Seguindo estes princípios, o método foi desenvolvido em quatro fases, de dezembro de 2011 a junho de 2013. Na primeira fase versões iniciais do método foram utilizadas para experimentação de versões preliminares do modelo de referência em empresas de software. Estas experiências serviram para subsidiar a consolidação, tanto do modelo quanto do método. Na segunda fase foi desenvolvida a versão 1.0 do método. Na terceira fase o método foi submetido à consulta pública. Na quarta fase as contribuições da consulta pública foram analisadas e o resultado da análise indicou uma validação e sugestões de melhoria que foram utilizadas para ajustes na descrição da nova versão. A versão 1.1 foi disponibilizada em junho de 2013 e desde setembro de 2013, esta versão do método tem sido utilizado para as avaliações da CERTICS.

O Método de Avaliação da CERTICS apresenta três grandes inovações com referência aos demais métodos de avaliação de processos existentes no mercado: sua definição é mais abrangente incluindo todas as fases e atividades de uma avaliação, desde a exploração e contratação até a validação e conclusão, sua execução é totalmente apoiada por uma plataforma de software (CERTICSys) que automatiza o processo de avaliação utilizando uma arquitetura orientada a serviços e a existência de uma fase denominada Exploração no método de avaliação que facilita o contato das Organizações com a metodologia incluindo a disponibilização de um tutor automatizado que auxilia a identificação de evidências e provê uma estimativa de sucesso para a avaliação com base nas informações providas.

Estas inovações atenderam a desafios específicos da Metodologia CERTICS. Estes desafios e inovações estão descritos neste artigo. Ainda que o Método de Avaliação da CERTICS tenha sido desenvolvido para avaliações utilizando o Modelo de Referência da CERTICS, ele foi projetado e implementado de modo que simplifica a sua adaptação e uso para avaliações em relação a outros modelos, principalmente aqueles modelos baseados na ISO/IEC 15504. Elementos do método podem também serem utilizados como referência para a evolução de outros métodos já existentes.

Esta Seção 1 introduz o artigo. A Seção 2 apresenta em linhas gerais a Metodologia CERTICS com sua arquitetura. A Seção 3 apresenta os principais desafios e inovações do Método de Avaliação da CERTICS. A Seção 4 introduz o Método de Avaliação da CERTICS e a Seção 5 apresenta considerações finais.

## 2. A CERTICS

A Metodologia de Avaliação da CERTICS para Software surgiu da necessidade de verificar se um software é resultante de desenvolvimento e inovação tecnológica realizados no País. Por meio da aplicação desta metodologia, pretende-se viabilizar as condições para o uso da margem de preferência em compras públicas, contribuindo para o desenvolvimento nacional sustentável. A arquitetura da CERTICS está estruturada em quatro camadas conceituais hierárquicas [CERTICS 2013a].

A primeira camada trata do conceito fundamental: software resultante de desenvolvimento e inovação tecnológica realizados no País. Com base neste conceito, foi realizada uma formulação de conceitos operacionais que orientaram a construção dos elementos do modelo.

A segunda camada é composta por quatro Áreas de Competência que detalham o conceito de software resultante de desenvolvimento e inovação tecnológica realizados no País presente na definição da primeira camada. Essas Áreas de Competência são denominadas: Desenvolvimento Tecnológico (DES), Gestão de Tecnologia (TEC), Gestão de Negócios (GNE), e Melhoria Contínua (MEC).

A terceira camada é composta por Resultados Esperados, que detalham cada uma das Áreas de Competência. Foram definidos dezesseis (16) Resultados Esperados distribuídos nas Áreas de Competência.

A quarta camada é composta por conjuntos de Orientações e Indicadores, que detalham os Resultados Esperados definidos na terceira camada. Cada conjunto de Orientações existente para cada Resultado Esperado orienta a avaliação do Resultado Esperado a partir da análise de evidências do desenvolvimento e inovação tecnológicos do software.

## 3. Principais desafios e inovações do Método de Avaliação da CERTICS

O Modelo de Referência para Avaliação da CERTICS é um modelo novo e possui como principal peculiaridade a análise de aspectos relacionados a avaliação de competências. Por ser um modelo novo, ele requerer um método de avaliação novo devido às peculiaridades do modelo.

Uma das premissas da CERTICS é que as avaliações seriam realizadas por um conjunto de Entidades Avaliadoras distribuídas no território nacional. Desta forma seria necessário uniformizar a execução das atividades de avaliação de um determinado software, para que fosse possível garantir que os resultados das avaliações fossem objetivos e repetíveis. Outro requisito da CERTICS é a difusão do método de avaliação para a mais ampla audiência possível. Isto foi resolvido pela documentação e publicação do Método de Avaliação da CERTICS no Website da CERTICS (<http://www.certics.cti.gov.br>).

Como a CERTICS implementa uma política pública, havia a necessidade de tornar o processo de avaliação o mais transparente possível. Desta forma, o Método de Avaliação da CERTICS foi elaborado cobrindo todas as atividades da avaliação, desde sua contratação até a conclusão. Diferente de outros métodos de avaliação, o Método de Avaliação da CERTICS descreve, por exemplo, em detalhes a contratação da avaliação e a validação da avaliação.

Ainda que a documentação do processo de avaliação em si por meio do Método de Avaliação da CERTICS facilite a padronização das atividades, outro mecanismo foi definido para enfatizar este aspecto: a definição, construção e implantação de uma plataforma de software cujo objetivo seria automatizar as atividades do processo de avaliação, maximizando assim a conformidade da execução da avaliação ao método de avaliação. Esta plataforma de software, denominada CERTICSys, está disponível na forma de um ambiente Web no endereço [www.certicsys.org.br](http://www.certicsys.org.br) e todas as avaliações da CERTICS ocorrem por meio dela. Diferente de ferramentas que apoiam outros modelos de avaliação de processos de software, a plataforma CERTICSys automatiza todo o processo de avaliação, desde o cadastramento dos dados da Organização e do software a ser avaliado até a conclusão do processo, incluindo a liberação do Relatório Final da Avaliação e, quando o resultado da avaliação for positivo, o Laudo da Avaliação.

A plataforma de software CERTICSys também auxilia em outro requisito para o método de avaliação definido para a CERTICS: a otimização do esforço e tempo de avaliação. Como as atividades são automatizadas isto reduz o esforço do Avaliador Líder no decorrer de uma avaliação, como: documentação das evidências observadas, pontuações, pontos fortes, fracos e oportunidades de melhoria, emissão de relatórios e gestão da agenda da avaliação e de todo o planejamento da avaliação.

Outro requisito para o Método de Avaliação da CERTICS era que, uma vez que o processo fosse automatizado pela plataforma CERTICSys, deveria haver a possibilidade de realizar ajustes no método de avaliação e de forma simplificada, aumentando a flexibilidade da solução. Para alcançar este objetivo, optou-se por definir uma arquitetura orientada a serviços para a plataforma, aplicando a orquestração de processos e utilizando uma suíte de gerenciamento de processos de negócio (BPMS - *Business Process Management Suite*) para a definição da solução, sendo todo o processo definido em BPMN (*Business Process Model and Notation*). Esta solução permite que evoluções e melhorias possam ser realizadas no processo de avaliação sem a necessidade de alterações na interface com o usuário e também separa as preocupações entre o processo de avaliação e a interface com o usuário, além de integrar todos os papéis envolvidos em todas as fases do método de avaliação.

A plataforma CERTICSys, além de garantir a uniformidade na execução das avaliações, fazendo com que as atividades de todas as avaliações sejam sempre executadas da mesma forma, também permite que haja visibilidade sobre a execução de todas as atividades das avaliações. Esta visibilidade permite que atividades críticas sejam acompanhadas em tempo real minimizando riscos e maximizando o sucesso das avaliações. Para este fim, foram definidos vários relatórios automatizados que fornecem em tempo real o estado de cada avaliação. As informações desses relatórios são extraídas da Plataforma CERTICSys por meio da ferramenta *Intalio/BPMS Business Activity Monitoring*, podendo ser exibidas em tela ou impressas. Como a Plataforma CERTICSys foi toda desenvolvida e é orquestrada por processos e para o mapeamento desses processos foi utilizada uma ferramenta BPMS a geração desses relatórios foi facilitada. Estes relatórios permitem monitorar o andamento das avaliações, desde o momento em que a Organização Solicitante contrata uma avaliação, passando pelas fases de preparação, visita, validação e conclusão, até o momento em que o processo de avaliação é finalizado. Esse tipo de monitoramento permite aos gestores atuar junto aos responsáveis pela execução das tarefas, facilitando comunicações, esclarecimentos ou orientações. Essa solução também permite à coordenação da CERTICS, tomar decisões estratégicas, pois a mesma funciona como um termômetro da operação.

A CERTICS possui como meta ser uma metodologia inclusiva, ágil e acessível. Desta forma o método de avaliação teve como requisito facilitar o contato das empresas com a metodologia de modo que não fosse uma premissa investir em consultoria para acompanhamento do processo de avaliação, o que culminou com o projeto da fase de Exploração do Método de Avaliação da CERTICS. Durante a exploração, a Organização cadastra uma visão do software e propostas de evidências objetivas que demonstram atendimento ao modelo de acordo com a sua compreensão da metodologia. Isto tem propiciado que o Avaliador Líder, na fase de Preparação, tenha a percepção de quão preciso é esta compreensão. Isto tem facilitado a interação entre as Organizações e os Avaliadores Líderes, agilizando a execução das fases seguintes, pois permite que o Avaliador Líder possa realizar um alinhamento deste entendimento facilmente sem ter que rever toda a metodologia com a Organização e ter mais atenção nas informações relacionadas aos Resultados Esperados que não foram completamente compreendidos pela Organização. Para facilitar o contato da Organização com a metodologia na fase de Exploração, foi projetado e implementado um tutor automatizado. Este tutor, presente na plataforma CERTICSys, tem auxiliado na compreensão da metodologia e identificação de propostas adequadas de evidências pela Organização Solicitante. O tutor realiza uma série de perguntas objetivas que auxiliam a Organização a identificar a que cenário ela se enquadra, sugere possíveis evidências objetivas que devem ser cadastradas e provê uma estimativa de atendimento ao Resultado Esperado com base nas informações fornecidas pela Organização.

Sobre o requisito de otimizar o processo de avaliação, outro ponto relevante adotado pelo Método de Avaliação da CERTICS é a identificação dos profissionais associados a cada evidência realizada pela Organização que tiveram maior envolvimento nas evidências. A Organização deve cadastrar na plataforma CERTICSys esses profissionais, informar o seu relacionamento com o software e associá-los às evidências objetivas com base no seu envolvimento com as evidências. Isto tem facilitado no planejamento das entrevistas, agilizando o planejamento das entrevistas e facilitando a identificação de pessoas chaves associadas a cada Área de Competência.

Sempre visando a melhoria contínua, outro requisito do método de avaliação foi a necessidade de mecanismos para acompanhar o *feedback* sobre as avaliações e sobre a metodologia. Foram projetados e implementados alguns mecanismos, dos quais são levantes citar as Lições Aprendidas e a Avaliação 360°. As Lições Aprendidas são registradas pelos envolvidos nas atividades executadas em cada fase do processo de avaliação na plataforma CERTICSys.

## 4. O Método de Avaliação da CERTICS

O Método de Avaliação da CERTICS, cuja definição foi realizada considerando os pontos citados na Seção 3, é caracterizado pela descrição dos papéis e responsabilidades, artefatos e fases. O método é realizado em seis fases sequenciais: F1-Exploração, F2-Contratação, F3-Preparação, F4-Visita, F5-Validação e F6-Conclusão [CERTICS 2013b].

O objetivo principal da Fase 1 - Exploração é permitir que uma Organização Solicitante explore o relacionamento entre o conjunto modelo e método e o software a ser avaliado, visando o aprendizado da Organização Solicitante sobre a Metodologia de Avaliação da CERTICS para Software. Outro objetivo é possibilitar que a Organização Solicitante forneça informações sobre ela e o software a ser avaliado e informações preliminares sobre as potenciais evidências do desenvolvimento do software.

As informações solicitadas sobre o software incluem a descrição, histórico, tecnologias relevantes e aspectos inovadores. Para cada Resultado Esperado o sistema solicita uma estimativa do grau de atendimento daquele software. A Organização pode registrar esta estimativa ou solicitar ajuda de um tutor disponibilizado na plataforma. O tutor expõe exemplos de cenários e caso a Organização reconhece um determinado cenário para o software, o tutor sugere tipos de evidências. Em ambos os casos, para cada evidência é informado uma descrição e uma justificativa de como ela contribui para evidenciar o atendimento do resultado. Com base nestas informações o sistema pode emitir uma estimativa de sucesso de o software ser avaliado. Esta fase é concluída com uma decisão da Organização sobre a contratação ou não da avaliação.

O objetivo da Fase 2 - Contratação é estabelecer o Contrato de Avaliação para a realização de uma avaliação. Nesta fase a Organização Solicitante escolhe e entra em acordo com uma Entidade Avaliadora para que esta conduza as duas fases seguintes (Fase 3 - Preparação e Fase 4 – Visita). Para tanto a Entidade Avaliadora define uma Equipe de Avaliação liderada por um Avaliador Líder.

O objetivo da Fase 3 - Preparação é preparar a Organização Solicitante e a Equipe de Avaliação para a visita de avaliação. Nesta fase, o Avaliador Líder revisa as informações sobre a Organização e sobre o software, as potenciais evidências identificadas pela Organização Solicitante e verifica a adequação e orienta possíveis ajustes nas informações e evidências fornecidas. Com base nas informações fornecidas e revisadas, a Organização Solicitante recebe uma estimativa do Grau de Prontidão, que indica o quanto a Organização Solicitante está pronta para a realização da Fase 4 - Visita. Para a realização da Fase 4 - Visita é feita a consolidação do cronograma detalhado inserido no Plano de Avaliação e a carga no Sistema, pela Organização, de todas as evidências identificadas e validadas na Fase de Preparação.

O objetivo da Fase 4 - Visita é executar uma visita da Equipe de Avaliação à Organização Solicitante para analisar evidências, pontuar o grau de atendimento dos Resultados Esperados a partir das evidências analisadas, consolidar e apresentar o resultado preliminar da avaliação, conforme acordado no Plano de Avaliação.

O objetivo da Fase 5 - Validação é assegurar que a avaliação foi realizada em conformidade com a Metodologia de Avaliação da CERTICS para Software. Nesta fase um Validador verifica se o processo de avaliação foi conduzido conforme o Método de Avaliação, se as evidências analisadas durante a visita estão coerentes com os requisitos do Modelo de Referência para Avaliação e se a pontuação atribuída aos Resultados Esperados está adequada para as evidências analisadas. Durante esta fase, o Validador pode interagir com o Avaliador Líder para possíveis esclarecimentos e ajustes no relatório preliminar, que passam a ser considerados para a validação.

O objetivo da Fase 6 - Conclusão é concluir o processo de avaliação. Nesta fase, o Relatório Final da Avaliação é enviado ao responsável pela Metodologia para a revisão do resultado da avaliação e a emissão do Laudo de Avaliação. O Relatório e o Laudo são então enviados à Organização Solicitante. As lições aprendidas são coletadas e registradas na plataforma CERTICSys, e o processo de avaliação é terminado com a emissão do Termo de Quitação. Caso o resultado da avaliação seja positivo, a Organização Solicitante pode submeter um pleito à SEPIN/MCTI para a emissão do certificado.

## 5. Considerações Finais

Este artigo teve como objetivo apresentar um relato de experiência sobre a definição e uso do Método de Avaliação da CERTICS. Na Seção 3 foram apresentadas várias questões de projeto, descritas como requisitos do método de avaliação e as soluções adotadas para resolvê-las no Método de Avaliação da CERTICS e na Seção 4 foi apresentado o Método de Avaliação da CERTICS.

Embora haja vários aspectos comuns em avaliações de processos de software foi observado que é importante definir os requisitos específicos de um método de avaliação, para observar se estes requisitos foram atendidos.

Em setembro de 2014, cerca de 180 softwares já iniciaram o processo de avaliação, seguindo o Método de Avaliação da CERTICS e usando a plataforma CERTICSys que apoia todo o processo. Destes 180 softwares, 8 já finalizaram o processo tendo seu certificado da CERTICS publicados no Diário Oficial da União. A plataforma CERTICSys tem permitido acompanhamento em tempo real dos processos do método de avaliação da CERTICS e tem sido amplamente utilizada para garantir a execução e a continuidade dos serviços da Operação da CERTICS. Adicionalmente, as informações obtidas nas lições aprendidas e na Avaliação 360° por meio da plataforma CERTICSys têm sido obtidas continuamente e após a execução da fase de Visita para cada avaliação. Estes instrumentos tem se mostrado um importante mecanismo para obter insumos para as atividades de monitoramento da operação e para a melhoria contínua da Metodologia de Avaliação da CERTICS para Software.

Embora o Método de Avaliação da CERTICS seja específico para a CERTICS ele pode ser adaptado para outros contextos e modelos de qualidade. A plataforma de software CERTICSys que automatiza e apoia a execução do método de avaliação pode ser estendida para apoiar qualquer método de avaliação baseado na norma ISO/IEC 15504. As soluções adotadas na definição do Método de Avaliação da CERTICS podem ser amplamente utilizadas no contexto de elaboração de métodos de avaliação de processos de software.

Este relato de experiência apresentou os principais desafios enfrentados na definição do Método de Avaliação da CERTICS, incluindo a necessidade da maior transparência do método de avaliação por ser parte de uma política pública, o objetivo de tornar o método de avaliação amplamente difundido e com o menor tempo de visita possível e a busca por facilitar a interação das Organizações com um novo modelo. Estes desafios levaram a geração das seguintes inovações em um método de avaliação: abrangência de todas as atividades de uma avaliação, a definição e uso de uma plataforma de software para apoiar a execução de todas as atividades do método de avaliação e a definição de uma fase do processo de avaliação denominada Exploração onde as empresas podem “explorar” a metodologia e, com o auxílio de um tutor automatizado, já identificar as evidências candidatas para atendimento ao modelo. Isto foi realizado com o objetivo de disseminação para reflexões e possíveis utilizações na evolução de outros métodos.



## Agradecimentos

Os autores agradecem o apoio financeiro dos projetos MCTI/01200.001832/2011 - *CTENIC* e FINEP 0113009300 - *Implementação da CERTICS*, o trabalho e energia da equipe da CERTICS e dos demais colaboradores do CTI Renato Archer e Facti, e as contribuições e sugestões dos revisores anônimos do WAMPS 2014.

## Referências

- (15504-2 2008) ABNT NBR ISO/IEC 15504 -Tecnologia da informação - Avaliação de processo - Parte 2: Realização de uma avaliação, ABNT, 2008.
- (15504-7 2008) ABNT NBR ISO/IEC 15504 - Tecnologia da informação — Processos de avaliação — Parte 7: Avaliação da maturidade de uma organização, ABNT, 2008.
- (CERTICS 2013a) Equipe CERTICS, Modelo de Referência para Avaliação da CERTICS - Documento de Detalhamento - Versão 1.1, Relatório Técnico CTI Renato Archer – TRT0084113, 2013.
- (CERTICS 2013b) Equipe CERTICS, Método de Avaliação da CERTICS - Documento de Detalhamento - Versão 1.1, Relatório Técnico CTI Renato Archer – TRT0083113, 2013.
- (Martinez e Salviano 2012) Martinez, M. R. M. e Salviano, C. F., Definição do Método para Estabelecimento de Referências em Ciclos de Melhoria de Processo PRO2PIWORK. In: 9. CONTECSI, 2012, São Paulo. International Conference on Information Systems and Technology Management, 2012. p. 685714.
- (Anacleto et al. 2004) Alessandra Anacleto, Christiane Gresse von Wangenheim, Clênio F. Salviano e Rafael Savi, A Method for Process Assessment in Small Software Companies, in Proceedings of SPICE 2004: The Fourth International SPICE Conference, Lisbon, Portugal, p. 69-76, April 28-29, 2004.
- (SEI 2011) SCAMPI Upgrade Team, Standard CMMI Appraisal Method for Process Improvement (SCAMPI) A, Version 1.3: Method Definition Document, HANDBOOK/ CMU/SEI-2011-HB-001, 2011.
- (Weber et al 2005) Weber, K. C., Araújo, E., Machado, C. F. M., Scalet, D., Salviano, C. F., Rocha, A. R. C. Modelo de Referência e Método de Avaliação para Melhoria de Processo de Software – versão 1.0 (MR-MPS e MA-MPS), In: Anais do IV Simpósio Brasileiro de Qualidade de Software (SBQS 2005). Porto Alegre, Brasil: junho de 2005.

# Relato de Experiência da Certificação do software PRIME Saúde da ECO Sistemas na CERTICS

Elizabeth Mocny<sup>1</sup>, Larissa Lopes de Araujo<sup>1</sup>, Mariano Montoni<sup>2</sup>, Analia Irigoyen<sup>2</sup>

<sup>1</sup>ECO Sistemas – Rua Dom Gerardo, nº 35, 10º andar - Centro, CEP 20.090-030 – Rio de Janeiro – RJ

<sup>2</sup>ProMove Soluções - Rua da Assembléia, nº10, sala 2805 - Centro - CEP 20.011-000 –Rio de Janeiro–RJ

{elizabeth,larissa}@ecosistemas.com.br, {mmontoni,analia}@promovesolucoes.com

**Abstract.** *The Brazilian software market is now the largest in Latin America, with the 7th position in the world ranking. To promote exports of software and the growth of this industry the Ministry of Science, Technology and Innovation Program launches IT Staff (Strategic Program Software and Information Technology Services). The CERTICS, one of his projects, validates the software as a result of technological innovation and development conducted in the country. This experience report presents the strategy and methodology used by the ECO Systems, an IT company with a focus on Public Health and multimodelos environment to certify their software PRIME Saúde, Primary Care, in CERTICS in six months. Today, PRIME has public health evidence on the ability of generating skills, technological autonomy, capacity and innovative business based in P&D.*

**Resumo.** *O mercado brasileiro de software é hoje maior da América Latina com a 7ª posição no ranking mundial. Para fomentar a exportação de software e o crescimento desta indústria o Ministério da Ciência, Tecnologia e Inovação lança o Programa TI Maior (Programa Estratégico de Software e Serviços de Tecnologia da Informação). A CERTICS, um de seus projetos, valida o software como resultado de desenvolvimento e inovação tecnológica realizado no País. Este relato de experiência apresenta a estratégia e metodologia utilizada pela ECO Sistemas, uma Empresa de TI com o foco em Saúde Pública e ambiente multimodelos, para certificar seu software PRIME Saúde de Atenção Básica, na CERTICS em 6 meses. Hoje, o PRIME Saúde possui a comprovação pública na habilidade de geração de competências, autonomia tecnológica, capacidade inovativa e negócios baseados em P&D.*

## 1. Introdução

A globalização seguida pela competição do mercado internacional tem impulsionado as organizações por contínuas inovações e melhorias de processos e produto ( BALDASSARRE, *et al.*, 2010) gerando crescimento econômico sendo a Tecnologia da Informação o principal motor para este crescimento.

Para garantir a participação neste mercado internacional, o País deve investir continuamente em melhoria de processo para se tornar mais eficiente, aumentando assim a sua produtividade, qualidade e inovação.

Em 2012 o Ministério da Ciência, Tecnologia e Inovação (MCTI) lançou o Programa TI Maior (Programa Estratégico de Software e Serviços de Tecnologia da Informação) (MCTI, 2013) para aumentar a competição da indústria nacional no mercado internacional. Com destaque nesse Programa foi apresentada a metodologia de certificação de software e serviços associados, a CERTICS (CTI Renato Archer, 2013). Esta certificação permite identificar softwares desenvolvidos no País, com base em P&D e inovação tecnológica e, com isto, ganhar preferências previstas em lei nas compras públicas. Desta forma desde 19 de setembro de 2013 todo software nacional passou a ter direito a margem de preferência nas compras governamentais desde que submetido a uma avaliação oficial da CERTICS.

A ECO Sistemas é uma empresa de Tecnologia da Informação com foco na saúde pública que atua há 23 anos no mercado brasileiro. Sendo pioneira na criação de softwares das Unidades de Pronto Atendimento (UPAs) espalhadas pelo Estado do Rio de Janeiro e em todo Brasil. Atua, em hospitais públicos e em secretarias municipais de saúde, como Campos dos Goytacazes, Nova Iguaçu, Niterói, Armação dos Búzios, Japeri e Petrópolis e com atuação através de cinco UPAs em Buenos Aires.

Com a estratégia de ser uma empresa focada na Qualidade de seus produtos e processos, a ECO Sistemas mantém desde 2005 um Sistema Gestão da Qualidade, que busca continuamente a excelência em modelos de melhoria de processo, através de um ambiente multimodelos. A organização implementou diversos modelos de referências, normas e certificações com o objetivo de apoiar seus múltiplos segmentos, a saber: (i) a norma ISO 9001:2008; (ii) o modelo de referência para qualidade de software MPS-SW Nível C; (iii) o modelo de referência para qualidade de serviços MPS-SV Nível G; (iv) o modelo de referência para qualidade de testes MPT Nível 1; (v) a certificação SBIS (S-RES) v. 3.3 2009 do produto Prime Saúde e; (vi) a certificação de Software CERTICS, também do produto Prime Saúde. Adicionalmente, a Eco também tem parceria com a Microsoft na competência Silver.

Com o objetivo de oferecer ao mercado uma solução com um alto valor agregado com inovação nacional e, conseqüentemente obter margem de preferências previstas em lei nas compras públicas, em Agosto de 2013, a ECO contratou a consultoria da Empresa ProMove Soluções para estruturar e organizar as evidências existentes do software PRIME Saúde (Sistema de Atenção Básica) com o objetivo de facilitar a Avaliação CERTICS.

O objetivo deste artigo é apresentar a estratégia e metodologia utilizada pela ECO para preparar as evidências e certificar o software, fruto de um projeto de com base em P&D, na metodologia CERTICS, em seis meses.

Este artigo está organizado em quatro seções além dessa introdução. A Seção 2 apresenta a CERTICS. A Seção 3 descreve todo o processo e metodologia utilizada para a condução da Preparação e Avaliação CERTICS pelo Software PRIME Saúde. As lições aprendidas do processo são apresentadas na Seção 4. E por fim, a Seção 5, contém as considerações finais deste artigo.

## 2. A CERTICS

A Lei nº 8.666/93 do Brasil contempla diretrizes singulares para balizar os processos de licitação e contratação de bens e serviços no âmbito da Administração Pública e essa orientação constitui diretriz de política pública adotada por diversos países como: (i) Os Estados Unidos, consolidada no "*Buy American Act*", em vigor desde 1933, estabeleceram preferência a produtos manufaturados no país; (ii) A China conforme disposições da Lei nº 68, de 29 de junho de 2002, estipula orientações para a concessão de preferência a bens e serviços chineses em compras governamentais; (iii) A Colômbia adotou a política nos termos da Lei nº 816, de 2003, com uma margem de preferência entre 10% e 20% para bens ou serviços nacionais; (iv) A Argentina por meio da Lei nº 25.551, de 28 de novembro de 2001, dá preferência aos provedores de bens e serviços nacionais, sempre que os preços forem iguais ou inferiores aos estrangeiros; (v) O México através da *Ley de Adquisiciones, Arrendamientos y Servicios del Sector Público*, de 04 de janeiro de 2000, afirma que os recursos econômicos à disposição do Governo Federal tem que ser administrada de forma eficiente. (Governo Federal, 2010)

No Brasil, a criação da metodologia CERTICS é um caso inédito de distinção de software resultante de desenvolvimento e inovação tecnológica realizados no País pela verificação das competências geradas.

A CERTICS é uma metodologia baseada na Norma ABNT NBR ISO/IEC 15504 para avaliação de software inovador, assim como alguns modelos de certificação desenvolvidos em outros países. As certificações de segurança de software nos segmentos de aviação, transporte ferroviário e automotivo são muito utilizadas, pois estabelecem um conjunto de critérios aceitos pela indústria, baseados em normas de segurança, como por exemplo: IEC 61508: norma para uma ampla classe de sistemas eletrônicos (IEC, 2005); DO-178C: norma para sistemas de aviação (RTCA, 2012); CENELEC: norma para sistemas de transporte ferroviário (Dick, 2008); ISO 26262-8: norma para sistemas do setor automotivo (ISO/IEC, 2009).

### 3. Certificando o Prime Saúde na CERTICS

Com o objetivo de certificar seu software na CERTICS, a equipe da Qualidade da ECO foi buscar o entendimento dos conceitos de inovação através do Curso de Introdução à Metodologia de Avaliação CERTICS para Software em agosto de 2013. A capacitação possibilitou que a equipe da Eco: (i) tivesse o primeiro contato com o funcionamento da certificação; (ii) identificasse o contexto do seu desenvolvimento (iii) tivesse entendimento do modelo de referência, o método de avaliação, o processo de certificação e; (iv) os primeiros contatos com a plataforma *CERTICSys*, que apoia todo o processo da avaliação do software.

O principal desta capacitação foi o entendimento do conceito fundamental da metodologia CERTICS, a identificação de softwares resultantes de desenvolvimento e inovação tecnológica realizados no País. O entendimento deste conceito e das áreas de competências da CERTICS nos guiou na seleção do produto a ser submetido à avaliação: o PRIME Saúde – Sistema de Atenção Básica. A seleção foi feita considerando as seguintes características: 1) foi construído recentemente baseado em P&D e inovação; 2) estava em implantação em diversos clientes municipais e 3) participa de forma frequente em licitações públicas.

A Consultoria ProMove e a ECO definiram um prazo inicial de quatro meses para a execução deste projeto. Nas próximas seções são apresentadas as etapas deste projeto.

#### 3.1. Metodologia adotada pela ECO

A metodologia de elaboração do projeto é um conjunto de etapas que foram ser seguidas e nos conduziram ao sucesso da certificação. O escopo deste projeto era certificar o software PRIME com a metodologia CERTICS, desta forma, foi necessário promover o entendimento do Modelo de Referência para Avaliação da CERTICS em paralelo com a identificação das evidências para o atendimento dos Resultados Esperados contidos nas Áreas de Competência. A Figura1 apresenta as sete etapas que foram definidas para a execução deste projeto, que são detalhadas a seguir:

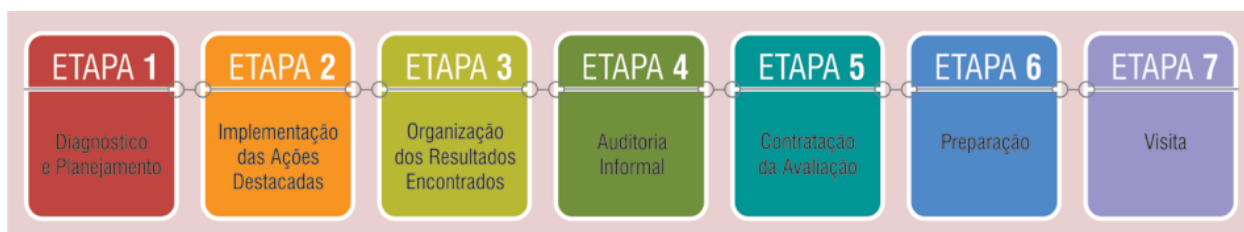


Figura 1: Etapas de Implementação e Avaliação da Metodologia CERTICS

#### a. ETAPA 1: Diagnóstico e Planejamento

Na primeira etapa do projeto foi realizado um estudo em cima do documento do Modelo de Referência para Avaliação da CERTICS v1.1 (CTI Renato Archer, 2013) para realizar o aprofundamento no entendimento das 4 Áreas de Competência e seus 16 Resultados Esperados.

Para a organização deste trabalho foi elaborado um *template* em planilha eletrônica (Figura 2) organizado em abas que refletem as quatro Áreas de Competência da CERTICS. Para cada uma das Áreas de Competência, foram elaboradas perguntas para as equipes de desenvolvimento e de analistas de negócio do software PRIME Saúde. O objetivo principal desta etapa era identificar e localizar as evidências no atendimento de cada Resultado Esperado. As respostas a estas perguntas e a elaboração de um *timeline* orientaram a equipe de qualidade na recuperação e estruturação das evidências.

CERTICS 2013			
Área de Competência Desenvolvimento Tecnológico (DES)			
Implementação/Questões	Evidências	Link	Responsável
Quais são os elementos relevantes da arquitetura?			
Os elementos relevantes foram desenvolvidos com base nos resultados de P&D?			
Existe pesquisa de mercado relacionada ao produto? Foram apontados os diferenciais tecnológicos ou de negócio do produto frente aos concorrentes?			
Existe uma especificação dos requisitos relacionados às tecnologias relevantes e está disponível na empresa?			

Figura 2: Planilha de Identificação de Evidências por Áreas de Competência

Uma das características do processo de inovação é que seu início é caótico. O movimento pela pesquisa começa com uma teoria ainda infundada. Esta pesquisa formaliza os conceitos que irão fundamentar esta nova criação. No meio deste caminho inúmeras ações podem contribuir para a formação do novo produto, como a participação em congressos, a capacitação em alguma nova tecnologia, a contratação de algum *know-how* ou *especialista* dentre outros. Todos estes aspectos, geralmente, são guiados por uma intenção de criação de um novo software, não sendo norteados por um processo.

Por conta dessa intenção, o processo de identificação das evidências necessárias para a certificação foi um processo de reorganização e estruturação de evidências criadas ao longo de três anos de investimentos em P&D, desenvolvimentos, protótipos e provas de conceitos (POC).

### **b. ETAPA 2: Implementação das Ações Destacadas**

Nesta etapa foram realizadas as ações necessárias para identificar e agrupar as evidências dos resultados esperados das quatro Áreas de Competência da CERTICS.

Estas ações foram identificadas a partir da interpretação conjunta (ECO e ProMove) do Modelo de Referência da CERTICS (CTI Renato Archer, 2013), dando início ao lançamento das evidências para cada resultado esperado em uma planilha eletrônica definida com base na metodologia. Este mapeamento foi um item importante para o sucesso da avaliação, pois podíamos ter o percentual de evidências já coletadas na última aba da planilha, obtendo o monitoramento das ações pendentes automaticamente.

Após a recuperação de todas as evidências correspondentes aos resultados esperados de cada área de competência, também foram registradas na planilha de controle: o responsável pela coleta, a data de entrega, e, para cada evidência, o link ao respectivo arquivo no local de armazenamento. Estas informações são importantes para apoiar a apresentação de documentos sigilosos e facilitar o agendamento das entrevistas na fase de Visita

### **c. ETAPA 3: Organização dos Resultados Encontrados**

Na conclusão da Etapa 2, todas as evidências estavam identificadas e submetidas ao controle de configuração, estruturadas abaixo de um projeto "CERTICS PRIME" e organizadas por Área de Competência. As evidências inerentes às evoluções do produto (código e documentos) continuaram no mesmo repositório de dados do Projeto, também submetidos ao controle de configuração.

O maior problema nesta etapa foi decidir a forma de apresentação das diversas evidências de pesquisa de mercado e tecnológicas, respeitando sua ordem cronológica.

Para atender este propósito foi elaborado um *whitepaper* com o objetivo de detalhar em um *timeline* os principais eventos relacionados com o produto PRIME Saúde, tais como: a sua idealização, a autorização para o início, as primeiras demandas, os estudos que foram descartados, as reuniões iniciais, as provas de conceitos, as contratações de especialistas, as participações em simpósios e feiras, as capacitações, as evoluções, a versão final publicada e a implantação no primeiro cliente.

Neste *whitepaper*, foram incluídos também os links para as evidências. Desta forma, um único arquivo integrador foi apresentado aos avaliadores na fase de Visita.

Este arquivo possuía ponteiros para todas as evidências do Projeto de criação do software inovador considerando cada área de processo da CERTICS e seus resultados esperados.

### **d. ETAPA 4: Auditoria Informal**

Nesta etapa, a ProMove elaborou a Auditoria Informal de forma presencial. O responsável pela auditoria utilizou a Planilha de Identificação de Evidências por Áreas de Processo, que foi a fonte utilizada pela Eco para cadastrar as evidências no CerticsSys.

O resultado desta auditoria foi o apontamento de onze oportunidades de melhorias, cujas ações foram definidas pela equipe da ECO e aprovadas pela auditora à distância.

Todas as ações de melhorias aprovadas foram registradas e realizadas.

#### e. **ETAPA 5: Contratação da Avaliação**

Para iniciar o processo de Avaliação foi necessário entrar no Site da CERTICS ([www.certics.org.br](http://www.certics.org.br)) e selecionar a opção "Acesso ao CERTICSys". Neste momento, foi necessário cadastrar um usuário para o cadastro das informações da Organização.

A primeira etapa na ferramenta é incluir no sistema todas as evidências lançadas na planilha por Área de Competência, analisando os requisitos e descrições contidos no próprio sistema. Após realizar a inclusão das evidências para todos os Resultados Esperados e, para todas as Áreas de Competência, o sistema apresenta um percentual de aderência. Este percentual de aderência é baseado no percentual de completude de cada resultado esperado, apontando uma boa probabilidade de sucesso na avaliação, considerando, é claro, a avaliação inicial feita pela FACTI.

O passo a seguir é indicar a Entidade Credenciada, a partir da lista disponível no site da CERTICS, como a responsável pela avaliação. Como na época da solicitação não existiam empresas homologadas para a realização de avaliações, a Unidade de Serviços de Avaliação foi contratada para essa realização.

#### f. **ETAPA 6: Preparação**

Nesta etapa, foi realizada a "Preparação" da Organização e da Equipe de Avaliação para a avaliação, sendo realizada à distância através da plataforma CERTICSys. O Avaliador Líder designado para a avaliação CERTICS, verificou pelas descrições e dúvidas se as evidências candidatas tinham potencial para alcançar o objetivo do resultado esperado.

Caso o artefato indicado, considerando suas descrições, não fosse suficiente para satisfazer o resultado, a organização era conduzida a acrescentar mais evidências ou melhorar descrições das evidências já cadastradas e caso o resultado tivesse evidências em excesso uma nota era recebida pelo sistema para retirar as evidências inadequadas.

Após todos os ajustes das evidências por Área de Competência foi obtida uma estimativa de grau de prontidão indicando que a Visita poderia ser realizada.

#### g. **ETAPA 7: Visita**

A Visita foi realizada em um dia pela equipe de Avaliação, composta por um Avaliador Líder, uma Avaliadora e uma Equipe de Apoio. Esta equipe de Apoio era formada por um dos responsáveis pela metodologia e outro responsável pelo suporte a plataforma CERTICSys. O objetivo da Visita é realizar: a verificação dos artefatos submetidos ao CERTICSys; as entrevistas e; pontuar os Resultados Esperados. Algumas pessoas chaves foram entrevistadas, principalmente para contextualizar todo o processo utilizado para a idealização de desenvolvimento do software.

Durante a pontuação das evidências e resultados esperados não houve a participação de um representante da empresa sendo uma atividade exclusiva da Equipe de Avaliação. Em alguns momentos foi necessária a intervenção de colaboradores que não haviam sido identificados anteriormente, o apoio do Ponto de Contato é sempre fundamental nestes momentos. A Visita foi um sucesso, garantindo a continuidade para as próximas atividades do processo aonde a Validação e Conclusão continuaram a ser realizadas via plataforma CERTICSys, assim como a solicitação por parte da ECO de iniciar o processo de certificação junto à SEPIN do MCTI.

Esse processo finalizou no ato da publicação do Certificado da CERTICS no DOU, publicado no Diário Oficial da União (DOU) em 26 de maio de 2014.

### 3.2. Diferencial Inovador do Prime Saúde

A escolha do software PRIME Saúde dentre outros da Organização para ser objeto de certificação CERTICS foi baseada no aspecto inovador do produto. Em atuação no Mercado Nacional de Saúde Municipal desde 2004 com implantações de sistemas assistenciais foi possível a Empresa adquirir *know-how* que identificasse o *core* tecnológico como diferencial de negócio e observar o que os concorrentes do software, com sistemas de atenção básica, não possuíam.

Dentre os aspectos que posicionam o PRIME Saúde como um software resultado de desenvolvimento e inovação tecnológica desenvolvido no País, estão: (i) O registro clínico do atendimento de médico e enfermeiro orientado a problema com a aplicação da metodologia SOAP (Subjective, Objective, Assessment, Plan – Subjetivo, Objetivo, Análise e Plano); (ii) O tratamento apoiado por linhas de cuidado preconizadas pelo Ministério da Saúde; (iii) A utilização do CIAP 2 (Classificação Internacional da Atenção Primária); (iv) O mapeamento automático do CIAP 2 para o CID 10; (v) O faturamento automático de procedimentos realizados de acordo com o perfil do profissional e o tipo de atendimento que realiza; (vi) A arquitetura leve para apoiar a utilização do sistema pelos municípios com pouca infraestrutura e (vii) A interface gráfica para apoiar o registro do atendimento médico orientado a SOAP.

## 4. Lições Aprendidas

Todo o processo de obtenção do conhecimento para a organização das evidências para a submissão à certificação CERTICS foi chamado de “Implantação CERTICS”. Apesar do modelo não se implantar, só se certificar, é certo falar que existiu uma implantação de fato do modelo. Isto porque toda a Organização que desejar submeter um software a esta certificação deveria conhecer o modelo antes da elaboração do mesmo. O conhecimento do modelo deve ser repassado aos analistas de requisitos e arquitetos de software, pois no início de um projeto inovador devem ser registradas as evidências.

O resgate da memória de idealização e construção do software PRIME através da elaboração de um *timeline* auxiliou a identificação cronológica da ordem dos acontecimentos. Além disso, o software de controle de versões foi fundamental na estruturação e organização dos artefatos, permitindo a tranquilidade da avaliação e a reutilização futura deste diretório para outros softwares que vierem a se certificar na Eco.

A interpretação do Modelo de Referência para Avaliação da CERTICS (CTI Renato Archer, 2013) com sua transformação em perguntas-chave para o levantamento das evidências que poderiam atender os Resultados Esperados foi útil ao projeto, pois facilitou as interações com o Avaliador. A disponibilidade dos entrevistados também é um ponto crucial no momento da Visita.

A utilização da ferramenta CERTICSys foi sem dúvidas um grande diferencial deste modelo de avaliação, pois permite uma facilidade na reutilização de evidências no cadastro dentro da ferramenta.



## 5. Considerações Finais

O investimento do projeto desde a contratação de consultoria equipe de processos e avaliação interna e externa girou em torno de R\$45.000,00 possibilitando benefícios na margem de preferência prevista em lei para as licitações.

Uma fraqueza identificada pelo projeto de avaliação foi a fato do software ser construído com o foco municipal e a lei de preferência do modelo CERTICS ser federal. Porém, após algumas análises, a escolha deste produto foi confirmada, pois as leis federais podem ser utilizadas em todas as esferas municipais, estaduais e federais.

O principal ganho de experiência foi a construção de uma imagem positiva da organização como desenvolvedora de software com desenvolvimento e inovação tecnológica realizados no País e esta certificação vem atestar a qualidade do software e a competência de uma empresa 100% brasileira, com compromisso de oferecer ao mercado solução de alto valor agregado para a melhoria da qualidade de vida da população brasileira, através do Sistema Único de Saúde - SUS.

Os próximos passos da ECO Sistemas é através do *know-how* adquirido nesta primeira certificação, avaliar um novo software da organização na CERTICS.

A equipe da qualidade, ao longo de todo o processo de execução das Fases, identificou algumas sugestões de melhorias na Ferramenta CERTICSys e no método de avaliação enviando como contribuição para a Facti.

## Referências Bibliográficas

- Baldassarre, M., Caivano, D., Pino, F. J., Piattini, M., & Visaggio, G. (2010). "A Strategy for Painless Harmonization of Quality Standards: A Real Case". PROFES 2010: LNCS 6156, pp 395-408.
- CTI Renato Archer. (2013). "Modelo de Referência para Avaliação da CERTICS - Documento de Detalhamento (Versão. 1.1)". Campinas: Centro de Tecnologia da Informação Renato Archer.
- Dick, A. (2008). "Evidence Based Development Applying Safety Engineering Techniques to the Progressive Assurance and Certification of Complex Systems". 3rd IET International Conference on System Safety, pp. 1-6.
- Governo Federal. (2010). "E.M.I. Nº 104/ MP/MF/MEC/MCT - Medida Provisória". Acesso em 2014 de 09 de 09, disponível em Site do Governo Federal: [http://www.planalto.gov.br/ccivil\\_03/\\_Ato2007-2010/2010/Exm/EMI-104-MP-MF-MEC-MCT-MPV-495-10.htm](http://www.planalto.gov.br/ccivil_03/_Ato2007-2010/2010/Exm/EMI-104-MP-MF-MEC-MCT-MPV-495-10.htm)
- IEC. (2005). "Functional Safety of Electrical / Electronic /Programmable Electronic Safetyrelated Systems". IEC 61508.
- ISO/IEC. (2009). "ISO 26262-8 Road vehicles — Functional Safety".
- MCTI. (2013). "Site TI Maior. (MCTI - Ministério da Ciência, Tecnologia e Inovação)". Acesso em 2014 de 09 de 05, disponível em <http://timaior.mcti.gov.br/>
- RTCA. (2012). "DO-178C/ED-12C, Software Considerations in Airborne Systems and Equipment Certification".

# Relato da experiência de implementação do modelo CERTICS em uma empresa que foi avaliada de acordo com o modelo de referência MPS-SW nível G

Allan M. R. Moura<sup>1</sup>, Breno F. Duarte<sup>2</sup>, Charles H. Alvarenga<sup>1</sup>, Paulo Lana<sup>2</sup>

<sup>1</sup>Fumsoft – Sociedade Mineira de Software  
Belo Horizonte – MG – Brasil

<sup>2</sup>Visual Sistemas Eletrônicos LTDA  
Belo Horizonte – MG - Brasil

ribeiromac@gmail.com, breno@visual.com.br, charles.alvarengati@gmail.com, paulolana@visual.com.br

**Abstract.** *This article aims to report how an implementation of improved processes and products based on CERTICS was planned and executed on a software for Service Management in a company that was assessed according to the MPS-SW level G. For this, we sought to describe the scenario in which the company was early in the project, which analyze the adherence of the product to the company as CERTICS model, measure the efforts to implement, and also make a comparison between the results required in the MPS-SW reference model level G and the method CERTICS and present the difficulties and the contributions of the two used models.*

**Resumo.** *Este artigo tem por objetivo relatar como uma implementação de melhoria de processos e produtos baseada na CERTICS foi planejada e executada em um software de Gestão para Atendimento em uma empresa que foi avaliada de acordo com o modelo de referência MPS-SW nível G. Para isso, buscou-se descrever o cenário em que a empresa se encontrava no início do projeto, analisar qual a aderência do produto da empresa quanto ao modelo CERTICS, mensurar os esforços para a implementação, e, ainda, fazer um comparativo entre os resultados exigidos no modelo de referência MPS-SW nível G e a metodologia CERTICS e apresentar as dificuldades e colaborações da utilização dos dois modelos.*

## 1. Introdução

A evolução e as mudanças constantes do setor de software proporcionam que tecnologias inovadoras sejam desenvolvidas; novos clientes e demandas apareçam; concorrentes aumentem e que cada vez mais queiram disputar um pedaço do mercado. Reflexos desta evolução podem ser vistos com o aumento, nos últimos anos, do nível de exigência por qualidade dos serviços prestados pelas organizações de software. Com isto, para sobreviverem, as organizações de software precisam aumentar sua capacidade de implementar melhorias nos seus processos, visando questões críticas, como por exemplo, aumentar a produtividade, reduzir os custos, diminuir os prazos de entrega, aumentar a satisfação dos clientes e melhorar a qualidade dos seus produtos e serviços [Montoni 2010].

Segundo [SOFTEX 2012], alcançar competitividade pela qualidade para as empresas de software, implica tanto na melhoria da qualidade dos produtos de software e serviços correlatos, como dos processos de produção e distribuição de software. Desta forma, assim como para outros setores,

qualidade é fator crítico de sucesso para a indústria de software. Para que se tenha um setor de software competitivo, nacional e internacionalmente, é essencial que os empreendedores do setor coloquem a eficiência e a eficácia dos seus processos em foco nas empresas, visando à oferta de produtos de software e serviços correlatos conforme padrões internacionais de qualidade.

Para apoiar a adoção de modelos de melhoria de processos, a Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) coordena o Programa de Melhoria de Processo do Software Brasileiro (MPS.BR). O modelo MPS baseia-se nos conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade e produtividade de software e serviços correlatos e também para a melhoria da qualidade e produtividade dos serviços prestados [SOFTEX 2012].

Com relação a melhoria da competitividade do produto, uma nova metodologia foi criada para atender a uma demanda da Secretaria de Política de Informática (SEPIN) e do Ministério da Ciência, Tecnologia e Inovação (MCTI): a CERTICS. A Metodologia de Avaliação da CERTICS para Software surgiu da necessidade de verificar se um software é resultante de desenvolvimento e inovação tecnológica realizados no País [CERTICS 2014]. Por meio da aplicação dessa metodologia, pretende-se viabilizar as condições para o uso da margem de preferência em compras públicas, contribuindo para o desenvolvimento nacional sustentável. O modelo de referência para a avaliação certifica que o desenvolvimento do software cria ou amplia competências tecnológicas e correlatas no País, contribuindo para a criação de negócios baseados em conhecimento, para o aumento de autonomia tecnológica e da capacidade inovativa [CTI Renato Archer 2013].

Dessa forma, o objetivo desse trabalho é apresentar o relato de experiência da implantação da metodologia CERTICS na empresa Visual Sistemas que foi avaliada no modelo de referência MPS-SW nível G.

O restante deste artigo está organizado da seguinte maneira: na seção 2, as características da empresa são apresentadas. Na seção 3 é apresentado o projeto de preparação para a avaliação CERTICS e na seção 4 é apresentada um comparativo dos modelos e informações sobre como a metodologia CERTICS foi atendida com o apoio do MPS-SW. Por fim, na seção 5 são apresentadas as dificuldades e fatores de sucesso para o projeto e as considerações finais na seção 6.

## 2. Visual Sistemas

Há 30 anos no mercado, a Visual Sistemas aplica pesquisas para prover soluções eficazes para empresas de todos os portes e segmentos, maximizando sua gestão, comunicação e interatividade, que são ferramentas fundamentais para o sucesso.

Por compreender que cada cliente é único, a Visual Sistemas estuda atentamente suas particularidades e elabora as alternativas mais adequadas para atendê-los. O portfólio de soluções da empresa conta com Placares Esportivos, Painéis Eletrônicos, Relógios Urbanos, Sistemas de Gestão para Atendimento e Sistema Eletrônico de Votação, atendendo a uma enorme demanda de clientes em todo o Brasil. Com isto, a Visual consolidou seu sucesso e tornou-se líder e referência absoluta nos diversos segmentos em que atua, contando com mais de 8.000 clientes atendidos, o que indica a sua competência, respeito e dedicação em cada projeto.

O cenário estabelecido pelo mercado nos dias de hoje exige que empresas e instituições públicas encontrem soluções capazes de acompanhar suas transformações, assegurando seu crescimento e sustentabilidade, além disso, as empresas que já possuem aspectos inovadores em seus softwares precisam se destacar no mercado. Diante disto, este foi um dos principais motivos para a Visual Sistemas tomar a decisão de adotar um conjunto de boas práticas para orientar o desenvolvimento de software com inovação tecnológica com base na metodologia CERTICS. Outro importante fator que contribuiu para a decisão de implementar e obter a certificação foi o aumento da oportunidade de negócios via margem de preferência que a empresa terá nos processos licitatórios e a construção de uma imagem positiva da organização como desenvolvedora de software com desenvolvimento e inovação tecnológica realizados no País.

A Visual Sistemas implementou e avaliou seus processos de desenvolvimento de software com base no modelo de referência MPS-SW nível G e obteve a certificação em 2013 participando de um grupo de empresas formado pela FUMSOFT-Sociedade Mineira de Software. Através desta mesma instituição, a Visual Sistemas tomou conhecimento da CERTICS e contratou o serviço para orientar o desenvolvimento de software com inovação tecnológica com base na metodologia CERTICS.

### **3. O projeto CERTICS**

O projeto de implementação foi executado com a participação de dois consultores da Fumsoft – Sociedade Mineira de Software e da equipe da Visual Sistemas. O projeto teve dois objetivos principais: levantar todo o histórico de desenvolvimento do software evidenciando os resultados esperados das áreas de competência da metodologia CERTICS e a melhoria dos processos tornando-os mais aderentes aos modelos de referência MPS-SW e CERTICS.

O projeto de implementação da metodologia foi dividido em 4 fases: diagnóstico inicial, treinamento, levantamento de evidências e adequação de processos e por fim a fase de avaliação. Abaixo será detalhado cada uma das fases de implementação.

#### **3.1. Fase 1: Diagnóstico inicial**

Esta fase tem como objetivo conhecer o produto, o histórico de versões, os envolvidos com o produto, e identificar os GAPS da empresa frente ao modelo CERTICS. Foi identificado também a situação atual da empresa com respeito a documentação das atividades do software quanto as áreas de conhecimento da CERTICS. Além disso, o diagnóstico visa identificar o quanto os processos estão institucionalizados e padronizados na organização e como isto pode contribuir para a evolução do negócio relacionado ao produto de software.

Um dos entregáveis do diagnóstico era identificar imediatamente quais artefatos poderiam ser utilizados como evidências parciais dos resultados esperados por cada área de competência da CERTICS. Na tabela 1 é apresentada a quantidade de evidências identificadas no diagnóstico inicial e a quantidade total de evidências identificadas no final do projeto e estão separadas por área de conhecimento e por resultado esperado.

**Tabela 1. Evidências para cada resultado da CERTICS identificadas no diagnóstico**

Áreas de conhecimento e resultados esperados	Evidências Diagnóstico	Evidências do Projeto
<b>Desenvolvimento Tecnológico – DES</b>	<b>14</b>	<b>83</b>
DES 1. Competência sobre Arquitetura	2	15
DES 2. Competência sobre Requisitos	2	17
DES 3. Fases e Disciplinas Compatíveis com o Software	1	6
DES 4. Papeis e Pessoas Identificados	3	18
DES 5. Dados Técnicos Relevantes Documentados	1	14
DES 6. Competência para Suporte e Evolução do Software	5	13
<b>Gestão de Tecnologia – TEC</b>	<b>6</b>	<b>53</b>
TEC 1. Utilização de Resultados de Pesquisa e Desenvolvimento Tecnológico	2	17
TEC 2. Apropriação das Tecnologias Relevantes Utilizadas no Software	3	18
TEC 3. Introdução de Inovações Tecnológicas	1	6
TEC 4. Capacidade Decisória nas Tecnologias Relevantes do Software	-	12
<b>Gestão de Negócios – GNE</b>	<b>9</b>	<b>39</b>
GNE 1. Ações de Monitoramento do Mercado	5	17
GNE 2. Ações de antecipação e Atendimento das Necessidades dos Clientes	2	14
GNE 3. Evolução do negócio relacionado ao software	2	8
<b>Melhoria Contínua – MEC</b>	<b>4</b>	<b>23</b>
MEC 1. Contratação, Treinamento e Incentivo dos Profissionais Qualificados	4	8
MEC 2. Disseminação do Conhecimento Relacionado ao Software	-	8
MEC 3. Ações de Melhorias nos Processos	-	7

Conforme apresentado, é possível identificar que as áreas de competência DES e GNE foram as que mais apresentaram evidências no diagnóstico que poderiam ser utilizadas parcialmente como resultado em cada área de conhecimento. As evidências, encontradas no Diagnóstico Inicial, representam 17% do total de evidências encontradas no final do projeto de implementação, que foram utilizadas na avaliação.

### 3.2. Fase 2: Treinamento

Nesta fase, os consultores realizaram um treinamento com a equipe do projeto para apresentação da CERTICS, tendo como objetivo aumentar o conhecimento da equipe da organização na metodologia e facilitar o entendimento para o levantamento de novas evidências e para a melhoria e definição de novos processos.

Durante o treinamento foi instigado a correlação dos processos da organização e os resultados gerados pelos processos e projetos de desenvolvimento das versões do produto com os resultados esperados da metodologia CERTICS, já que no diagnóstico inicial, nem todas as evidências são identificadas em um primeiro momento.

### **3.3. Fase 3: Levantamento de evidências e adequação de processos**

Nesta fase, a equipe do projeto recebeu orientações e foi auxiliada a identificar as evidências para atender os resultados esperados de cada área de conhecimento da metodologia CERTICS e, em vários momentos a equipe se deparou com resultados gerados em projetos de evolução do software executados com o processo certificado com base no modelo de referência MPS-SW nível G que poderiam ser utilizados nos resultados esperados do modelo CERTICS.

Algumas evidências identificadas não apresentavam informações que deixavam claro o objetivo daquela evidência para um resultado do modelo. Para isto, a equipe do projeto foi auxiliada na padronização destes artefatos para facilitar nas próximas avaliações e também para a melhoria dos registros organizacionais.

Durante esta fase, os processos da empresa também foram revistos para promover melhorias internas e adequações para melhor atender ao modelo CERTICS. Dentre estas melhorias internas, destaca-se a melhoria do processo de suporte com relação à classificação dos tipos de chamados atendidos para facilitar sua rastreabilidade.

Ainda nesta fase, a equipe do projeto de implementação da organização foi orientada na exploração da ferramenta CERTICSys, onde foram cadastradas as evidências identificadas para cada resultado esperado das quatro áreas de conhecimento da CERTICS. Além disso, foram incluídas na ferramenta as informações sobre como as evidências eram consideradas válidas para aquele resultado dentro de uma área de conhecimento e inclusão de todas as pessoas envolvidas com a evidência.

### **3.5. Fase 4: Avaliação**

Nesta fase a empresa submeteu seu pedido de certificação junto a FACTI através da ferramenta CERTICSys onde iniciou o processo de contratação da avaliação. Juntamente com esta etapa, a Visual Sistemas selecionou a Instituição Avaliadora e planejou a avaliação com base na metodologia CERTICS e nas evidências já cadastradas no sistema CERTICSys. A previsão para realização da avaliação é no último bimestre de 2014.

## **4. CERTICS e MPS-SW Nível G**

A primeira semelhança entre os modelos é que a norma ISO/IEC 15504 [ISO 2003] foi utilizada como base técnica para definição do método de avaliação no MPS-SW e na CERTICS. Outras semelhanças são apresentadas na Tabela 2, onde é realizada uma comparação entre as evidências utilizadas nos resultados esperados da CERTICS de acordo com as áreas de conhecimento e a quantidade de evidências utilizadas dos resultados esperados do MPS-SW. É importante destacar que a Tabela 2 não representa a quantidade de evidências geradas nos resultados esperados do MPS-SW, mas o número de evidências que tem alguma relação com o resultado esperado da área de conhecimento da CERTICS.

**Tabela 2. Comparação das evidências dos resultados da CERTICS e resultados aproveitados do MPS-SW para o nível G**

CERTICS	MPS-SW Nível G	Qtd
<b>Área de Competência: Desenvolvimento Tecnológico</b>		
DES 1 - Competência sobre Arquitetura	GPR 5. O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos. RAP7 - As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência.	8
DES 2 - Competência sobre Requisitos	GRE2 - Os requisitos são avaliados com base em critérios objetivos e um comprometimento da equipe técnica com estes requisitos é obtido. GPR1 - O escopo do trabalho para o projeto é definido, porém com foco nos requisitos da tecnologia relevante do software. GPR7 - Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo.	15
DES 3 - Fases e Disciplinas Compatíveis com o Software	GPR3 - O modelo e as fases do ciclo de vida do projeto de desenvolvimento de software são definidos e os projetos de evolução do produto seguiram este padrão.	5
DES 4 - Papéis e Pessoas Identificados	GPR7 - Os recursos humanos para o projeto são planejados considerando o perfil e o conhecimento necessários para executá-lo. A certificação MPS-SW Nível G.	7
DES 5 - Dados Técnicos Relevantes Documentados	GPR9 - Os dados relevantes do projeto são identificados e planejados quanto à forma de coleta, armazenamento e distribuição. Um mecanismo é estabelecido para acessá-los, incluindo, se pertinente, questões de privacidade e segurança. GRE 1. O entendimento dos requisitos é obtido junto aos fornecedores de requisitos.	4
DES 6 - Competências para Suporte e Evolução do Software	RAP7 - As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência. GPR 1. O escopo do trabalho para o projeto é definido.	7
<b>Área de Competência: Gestão de Tecnologia</b>		
TEC 1 - Utilização de Resultados de Pesquisa e Desenvolvimento Tecnológico	MPS-SW Nível G	3
TEC 2 - Apropriação das Tecnologias Relevantes Utilizadas no Software	RAP7 - As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência. A certificação MPS-SW Nível G.	7
TEC 3 - Introdução de Inovações Tecnológicas	Não gerou evidências semelhantes.	-

TEC 4 - Capacidade Decisória nas Tecnologias Relevantes do Software	GRE2 - Os requisitos são avaliados com base em critérios objetivos e um comprometimento da equipe técnica com estes requisitos é obtido RAP 2. Existe uma política organizacional estabelecida e mantida para o processo. RAP 6. (Até o nível F). As responsabilidades e a autoridade para executar o processo são definidas, atribuídas e comunicadas. RAP 7 - As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência.	<b>9</b>
<b>Área de Competência: Gestão de Negócios</b>		
GNE 1 - Ações de Monitoramento do Mercado	Não gerou evidências semelhantes.	<b>8</b>
GNE 2 - Ações de Antecipação e Atendimento das Necessidades dos Clientes	Não gerou evidências semelhantes.	<b>5</b>
GNE 3 - Evolução do Negócio Relacionado ao Software	Não gerou evidências semelhantes.	<b>8</b>
<b>Área de Competência: Melhoria Contínua</b>		
MEC 1 - Contratação, Treinamento e Incentivo aos Profissionais Qualificados	RAP7 - As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência. RAP6. (Até o nível F). As responsabilidades e a autoridade para executar o processo são definidas, atribuídas e comunicadas.	<b>2</b>
MEC 2 - Disseminação do Conhecimento Relacionado ao Software	RAP7 - As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência. RAP 2. Existe uma política organizacional estabelecida e mantida para o processo.	<b>4</b>
MEC 3 - Ações de Melhorias nos Processos	A certificação MPS-SW Nível G . RAP7 - As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência.	<b>5</b>

Com base na Tabela 2, das 198 evidências identificadas no final do projeto, 49% destas evidências foram geradas de projetos que utilizaram o modelo de referência MPS-SW Nível G e que foram reaproveitados para os resultados esperados da metodologia CERTICS. Este resultado dependerá da forma como cada empresa definiu os processos com base no modelo MPS-SW e se os projetos de evolução do produto certificado utilizaram o processo.



## 5. Dificuldades e fatores de sucesso

Durante a execução desta iniciativa de melhoria dos processos e da gestão do produto com foco no modelo CERTICS, a equipe enfrentou dificuldades e também passou por situações que foram determinantes para o alcance do seu objetivo. Estas informações encontram-se descritas a seguir, sendo agrupadas em duas percepções: a da equipe da unidade organizacional e a dos consultores de implementação.

Para a equipe da unidade organizacional as dificuldades que merecem destaque são: a disponibilização de recursos humanos para dedicação ao projeto, sendo que estes conflitavam com outras atividades da empresa e o reforço na mudança da cultura para registro das evidências que comprovem a continuidade e evolução do produto da empresa. Os principais fatores de sucesso considerados pela empresa foram: o apoio de consultores de implementação; ter passado pela implementação e avaliação do MPS-SW; já possuir uma cultura de processo na empresa e já possuir institucionalizados muitos dos resultados esperados da metodologia CERTICS.

Para os consultores de implementação, a principal dificuldade que merece destaque está relacionada com a falta de uma equipe com horas alocadas para o projeto, já que dependendo da maturidade dos processos que a empresa se encontra, a CERTICS pode ser uma metodologia de implementação rápida, desde que haja alocação de recursos para as atividades. Com relação aos fatores de sucesso, os consultores destacam como principal a empresa já possuir implementado e certificado os processos de acordo com o modelo de referência MPS-SW e já possuir uma visão de processos institucionalizada na empresa, sendo desenhados de forma prática, definidos de acordo com o operacional e com os projetos da organização. Outro fator de sucesso foi a empresa possuir várias ações e mecanismos voltados para a gestão do negócio da empresa e do software, o que contribuiu significativamente para os resultados esperados da área de conhecimento Gestão de Negócios da CERTICS.

Para este projeto de implementação, devido à indisponibilidade da equipe do projeto, foram necessários sete meses de trabalho da equipe do projeto e 40 horas de consultorias.

## 6. Considerações finais

As organizações de software devem ser capazes de implementar, continuamente, melhorias nos seus processos, como estratégia para aumentar a sua competitividade e até mesmo garantir a sua sobrevivência no mercado [Montoni 2010].

Como a CERTICS documenta um conjunto de boas práticas para orientar o desenvolvimento de software com inovação tecnológica, além de guiar a empresa a relacionar o desenvolvimento tecnológico com a geração de negócios baseados em conhecimento, ela pode ser utilizada como uma referência para entendimento e melhoria das práticas da empresa [CERTICS 2014] juntamente com o MPS-SW, no entanto, diversos problemas e dificuldades são enfrentados pelas organizações na condução de iniciativas de melhorias.

Este artigo buscou relatar a experiência da empresa Visual Sistemas Eletrônicos na implementação da metodologia CERTICS em seu software de Gestão para Atendimento e apontar as dificuldades e também os fatores de sucesso para alcançar o objetivo. Além disso, o trabalho apresentou como contribuições, uma comparação de quais resultados esperados do MPS-SW no nível G podem ser utilizados como resultados esperados em cada área de conhecimento da CERTICS.

Por fim, este artigo ao discutir as informações sobre uma implementação da CERTICS em uma empresa avaliada MPS-SW nível G, buscou colaborar para que novas iniciativas de melhoria de processos e produtos possam ser realizadas, além de apresentar uma contribuição para os gestores de outras organizações, na tomada de decisão para a adoção de melhorias em seus processos e produtos.

## 7. Referências

- CERTICS. (2014) "Benefícios para as empresas desenvolvedoras", [http://www.certics.cti.gov.br/?page\\_id=127](http://www.certics.cti.gov.br/?page_id=127), Setembro.
- CTI Renato Archer. (2013) "Método de Avaliação da CERTICS", Documento de detalhamento, versão 1.1, Campinas-SP.
- ISO. (2003) "Information technology – Software process assessment". vol. ISO/IEC 15504: Parts 1-9, The International Organization for Standardization and the International Electrotechnical Commission.
- Montoni, Mariano Angel. (2010) "Uma Investigação sobre os Fatores Críticos de Sucesso em Iniciativas de Melhoria de Processos de Software", Rio de Janeiro: UFRJ/COPPE.
- Softex. (2012) "MPS-SW-Guia Geral MPS de Software", <http://www.softex.br/mpsbr/guias>, Setembro.



# Encontrando o Equilíbrio entre a Metodologia Scrum na Fábrica JAVA e o modelo MPS.br- SW - nível F

Nanachara V. Silva<sup>1</sup>, Paulo R. S. Antiquera<sup>1</sup>, Edenilson R. Burity<sup>2</sup>

<sup>1</sup>Datacooper Software Ltda  
85.812 - 260 – Cascavel – PR – Brazil

<sup>2</sup>STI – Serviços Técnicos e Inovação – SENAI – Unidade Londrina  
86.026 - 000 – Londrina – PR – Brazil

{nanachara.silva, paulo.antiquera}@datacooper.com.br, edenilson.burity@pr.senai.br

**Abstract.** *In this paper will be described the alternatives found during the implementation of MPS.br model - level F on the ERP development area in conjunction with agile practices. The focus of process team and consulting was the simplicity, the objective was the implementation of the model in a way to get the minimum impact on daily activities of the teams, making possible the use of those methodologies together. This paper presents the means used to reach the balance between MPS.br and SCRUM, as well the strategies adopted to institutionalize the implemented processes.*

**Resumo.** *Neste artigo serão descritas as alternativas encontradas durante a implementação do modelo MPS.br – nível F em uma área onde o desenvolvimento do ERP é feito utilizando práticas ágeis. O foco da equipe de processo e da consultoria durante o projeto de implementação sempre foi a simplicidade, a fim de que houvesse o mínimo de impacto na mudança das atividades diárias das equipes existentes, e fazendo convergir ambas as metodologias. O artigo apresenta os meios utilizados para atingir o ponto de equilíbrio entre o MPS.br e o SCRUM, bem como as estratégias adotadas para a institucionalização dos processos implementados.*

## 1. Introdução

Ao iniciar um novo projeto, geralmente vem à mente do gerente de projeto, os padrões estabelecidos pelo PMI e o PMBok. Esses padrões contemplam um conjunto de orientações, regras e características a serem seguidas, a fim de alcançar a excelência na execução de um projeto, por meio de técnicas já testadas que podem contribuir para a mitigação de falhas [PMI, 2013]. No entanto quando se trata de projetos de desenvolvimento de software há um leque de opções em termos de metodologias, para a condução deste tipo de desafio.

O modelo MPS.Br por exemplo, veio para atender às micro e pequenas empresas que buscam e necessitam de melhores práticas de mercado para o desenvolvimento de software que sejam aplicáveis em sua realidade [SOFTEX, 2012]

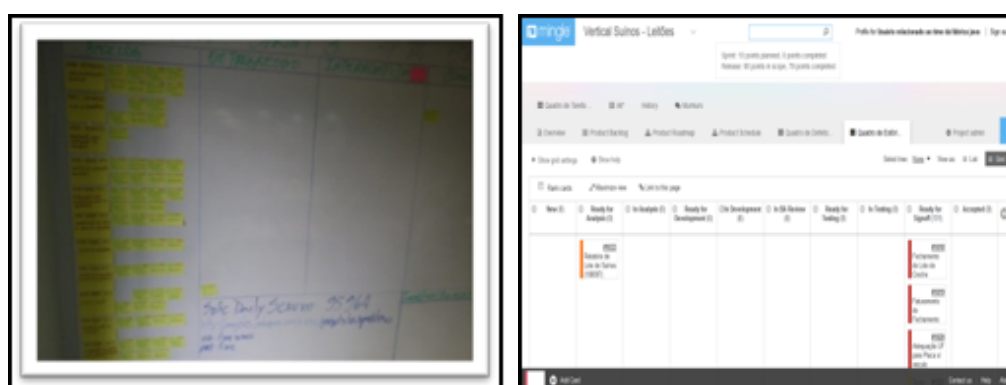
Para [Poppendieck, 2011], “o desenvolvimento é um processo de transformar ideias em produtos”, e esta transformação é normalmente feita utilizando o modelo em cascata ou metodologias ágeis. De acordo com estudos [Boehm, 2006] a partir da década de 90 as metodologias ágeis ganharam força. Os métodos ágeis são por vezes visto como o oposto à metodologias guiadas pelo planejamento ou

disciplina, considerados estes como burocráticos. Em parte esta visão é corroborada por interpretações do Manifesto Ágil [Beck, K et al. 2001] que cita a priorização de pessoas, software em funcionamento e resposta à mudanças, ao invés de processos, documentação e um planejamento. No entanto, priorizar não deveria significar ignorar registros e acompanhamentos, mas sim encontrar equilíbrio entre as coisas que podem colaborar para o encantamento do cliente por meio do fornecimento de produtos e serviços de excelência de forma segura [Poppendieck, 2011].

## 2. A história do MPS.br na Fábrica JAVA

Em 2010 a Datacooper realizou sua primeira avaliação no MPS.Br SW - nível G sendo considerada como escopo da avaliação, a Fábrica JAVA. Esta área é a responsável pelo desenvolvimento do COOPERATEBC, um ERP voltado a empresas do agronegócio. Na época da primeira avaliação, a Fábrica JAVA tinha seu processo de desenvolvimento no modelo cascata, e todo registro documental exigido pelo modelo MPS foi feito utilizando o pacote office, o que era complexo de ser mantido pela equipe, e aos poucos, foi caindo em desuso. Algum tempo depois, o Gestor da Fábrica que já havia sido engenheiro de configuração da área, propôs uma mudança na metodologia de desenvolvimento: de cascata para SCRUM.

A troca de metodologia foi feita incrementalmente com: post-it sendo colados e movidos manualmente pelos colaboradores (Figura 1) da área até a incorporação do trabalho em “sprints” e as reuniões diárias. Após o ganho de maturidade sobre metodologias ágeis, o Gestor optou por fazer uso de uma ferramenta específica para gerenciamento ágil (Figura 1) denominada de MINGLE. Esta ferramenta foi desenvolvida pela ThoughtWorks e possui a capacidade de visualização de tarefas no Mingle como cartões na tela, internamente chamadas de cartas. A ferramenta permitiu realizar o acompanhamento pela aplicação do que os colaboradores haviam praticado com os painéis.



**Figura 1. Início da prática do SCRUM com post it na parede, e posteriormente o uso de uma aplicação web**

Inicialmente para a fábrica, a velocidade da evolução incremental da metodologia, estava aparentemente de forma adequada, no entanto com a proximidade da avaliação do nível F, esse processo de maturidade precisou avançar mais rápido. Da primeira implementação ficou foi a importância e a necessidade de evolução dos controles, além do uso da terminologia empregada em gerenciamento de projetos.

### 3. A implementação

Segundo o PMI, um projeto é um conjunto de atividades temporárias, realizadas em grupo, destinadas a produzir um produto, serviço ou resultado únicos [PMI, 2013]. O ERP no grau de complexidade que desenvolvemos, tem a duração de seu projeto em torno de 2 a 3 anos. Desta forma a melhor alternativa para implementação do modelo, foi estabelecer o conceito de que cada “Sprint” é um projeto.

A seguir apresentaremos as alternativas adotadas para a implementação do modelo de acordo com as maiores dificuldades apresentadas.

### 4. Gerencia de Projetos – GPR

#### 4.1. Plano de Projeto

Uma necessidade básica da Gerência de Projetos é a existência de um Plano de Projeto concentrando as principais informações sobre o projeto em execução. Por mais informações que houvessem atendendo as necessidades da equipe, ainda assim não havia uma concentração destes dados no formato de um plano de projeto.

*Alternativa:*

Foi criada uma página de wiki composta por macros que auto-organiza as cartas de acordo com sua propriedade a medida que são criadas. Dessa forma são definidas em uma única tela a visualização clara do cronograma dos “sprints”, as histórias épicas que compõe o projeto, riscos, pessoas, e principais métricas.

#### 4.2. Cronograma

O Guia do MPS.br faz menção ao uso de uma forma de visualização do cronograma do projeto, e sugere o uso de uma EAP (Estrutura Analítica do Projeto). Em geral as empresas e ferramentas mais comuns no mercado fazem essa visualização por meio do Diagrama de Gantt. Porém, o Mingle não apresenta os dados desta forma.

*Alternativa:*

Em metodologias ágeis é comumente realizado o acompanhamento por meio do gráfico Burndown. A medida que a evolução das cartas ocorre no quadro, o gráfico Burndown é atualizado. Este gráfico também apresenta a data de início e término do projeto, sendo assim, é possível visualizar através do mesmo se o time está adiantado para com as entregas ou atrasado.

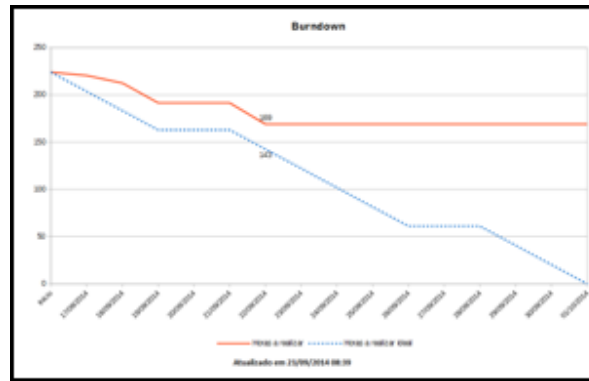


Figura 2. Gráfico Burndown

### 4.3. Ciclo de Vida

No início da implementação, o Scrum ainda estava em construção e não existia um ciclo de vida definido. Havia 5 equipes (times) trabalhando de maneiras diferentes na fábrica com a metodologia.

*Alternativa:*

A implementação do MPS.br foi a oportunidade para nivelar algumas formas de trabalho entre os “times”, inclusive por meio do uso de um mesmo template de projeto definido na ferramenta. O ciclo de vida definido foi estabelecido com o auxílio de algumas pessoas chaves da área, e seu refinamento foi feito até o ponto onde as particularidades de cada “time” ou projeto deixavam de ser comum aos demais. No entanto, nenhuma dessas particularidades foi contraproducente com relação ao template de projeto na ferramenta. O ciclo de vida do projeto (Sprint) estabelecido, foi então composto por: Sprint Plannig, Desenvolvimento e Sprint Retrospective/Review.

### 4.4. Recursos Humanos

O modelo sugere que os recursos humanos sejam planejados de acordo com seu perfil, e a responsabilidade pelas atividades a serem executadas sejam possíveis de rastreamento.

*Alternativa:*

Como a ferramenta possui limitação de usuários, os relacionamentos entre recursos humanos e atividades foram solucionados com a criação de uma carta com as especialidades (papéis) necessárias para o andamento do projeto, e cartas criadas com o nome do colaborador. A carta do colaborador é relacionada à carta da especialidade e também se relaciona à carta da customização ou do defeito, de forma a identificar a responsabilidade do desenvolvimento. Ambas as cartas, especialidade ou pessoas, se auto-organizam na tela do plano do projeto para identificação de quais colaboradores compõem o projeto.

### 4.5. Estimativa de esforço e tamanho

Uma dificuldade que muitas organizações enfrentam é para retirar a subjetividade da forma como os colaboradores mais experientes trabalham e conseguir transcrever de forma objetiva os critérios utilizados para o trabalho.

*Alternativa:*

Em nosso processo de desenvolvimento é realizado dois tipos de estimativas: de tamanho (“story points”) e de esforço (horas). Com relação à estimativa de esforço, a técnica utilizada é o “Planning Poker”, e não haviam grandes divergências de entendimento entre as equipes. No entanto, com relação à estimativa de “Story Points”, a subjetividade possuía uma dimensão bem maior de acordo com o P.O. (“Product Owner”) da equipe. A retirada da subjetividade com relação à estimativa em “Story Points” foi obtida pela criação e utilização de uma régua de “story points” assim como apresentado por [Boria et al, 2013] no livro A história de Tahini-Tahini e sugerido pelo consultor. A régua de “story points” é uma matriz que relaciona os tipos de demanda com a escala de “Story Points”. Cada célula desta matriz, resultante da relação entre os tipos e os níveis da escala, apresenta um código que identifica a relação. Este código é composto por um número identificador do tipo da demanda e o nível na escala de “story points”, separados por ponto (“.”), nesta ordem.

Para que o P.O. faça uso da régua é necessário possuir a estória a estimar no nível de detalhe que proporcione o entendimento. Geralmente estas estórias já devem possuir a narrativa e uma lista inicial de critérios de aceitação. Se possível também devem haver informações extras utilizadas para validação ou detalhamento da demanda com o cliente como, por exemplo, planilhas, documentos e protótipos.

A escala de medição é baseada na série de Fibonacci, onde 13 “story points” é o tamanho máximo de uma estória de uso. Qualquer estória com pontuação superior a 13 é “quebrada” até que as estórias resultantes tenham tamanho igual ou inferior a 13 pontos.

	0	1	2	3	5	8	13
Cadastro Básico		* Sem Seletor	* Com até 5 Seletores	* Com 5 a 10 Seletores			
Cadastro Mestre/Detailhe			* Com 1 detalhe * Validação de Chave * Validação de Vigência	* Com 2 detalhes * Validação de Chave * Validação de Vigência	* De 3 a 4 detalhes * Validação de Chave * Validação de Vigência	* Mais de 4 detalhes * Validação de Chave * Validação de Vigência	
Melhorias de Interface		Padronização de Interface					
Validações		* Até 4 sem SQL * Até 2 com SQL	* Até 10 sem SQL * De 3 a 4 com SQL	* Até 15 sem SQL * De 5 a 6 com SQL			

Figure 3. Exemplo de critérios para uso da régua de Story Points

	0	1	2	3	5	8	13
<b>1. Cadastro Básico</b>	1.0	1.1	1.2	1.3	1.5	1.8	1.13
<b>2. Cadastro Mestre e Detalhe</b>	2.0	2.1	2.2	2.3	2.5	2.8	2.13
<b>3. Melhorias de Interface</b>	3.0	3.1	3.2	3.3	3.5	3.8	3.13
<b>4. Validações</b>	4.0	4.1	4.2	4.3	4.5	4.8	4.13

Figure 4. Exemplo da Régua de Story Points

Dessa forma, por exemplo, uma estória de Cadastro Básico com até 5 seletores (1.2) terá sua estimativa de tamanho em “2 story points”.



## 5. Gerencia de Requisitos – GRE

### 5.1. Rastreabilidade bidirecional

A rastreabilidade deve ser feita entre os requisitos, códigos ou módulos de software. Na primeira avaliação esta rastreabilidade teve sua construção pelo uso de planilhas. Devido à complexidade de alimentação e retroalimentação deste tipo de arquivo, em pouco tempo a matriz caiu em desuso.

*Alternativa:*

Nesta implementação, como todo código fonte alterado e desenvolvido é feito por meio da ferramenta SVN, foi criada uma obrigatoriedade para que os desenvolvedores informem no “commit” o número da solicitação e da carta que exigiu a alteração. A solicitação é um registro utilizado em outra ferramenta interna chamada FNT, onde é realizado o lançamento das horas de trabalho. Com esta informação é possível pelo show log do SVN encontrar os fontes envolvidos em uma determinada carta, o classname e a solicitação contendo o esforço gasto. Como a informação do número da carta ou do número da solicitação, é possível localizar os requisitos, seus critérios de aceitação ou mesmo identificar testes realizados e defeitos encontrados. A rastreabilidade na Fábrica JAVA foi então representada por uma figura:

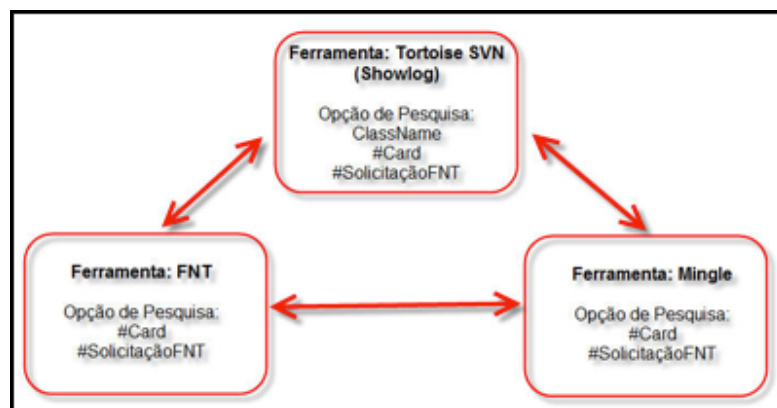


Figura 5. Rastreabilidade bidirecional

### 5.3. Itens de Configuração

A granularidade dos itens de configuração, conforme o próprio modelo coloca, está intimamente relacionada com a utilização e manutenção destes itens. Ao nivelar de forma macro as equipes, foi possível também maximizar o controle dos itens de configuração.

*Alternativa:*

Foi criado um projeto organizacional que concentra todos os planos, guias e demais itens de configuração que são pertinentes ao bom andamento do processo de desenvolvimento da Fábrica JAVA, e todos os projetos são relacionados a ele.

## 6. Medição – MED

### 6.1. Métricas

A gerência de medição solicita que um processo composto por diferentes itens seja estabelecido em um plano e mantido. As equipes possuíam no início da implementação formas diferentes de monitoramento, com métricas nem sempre semelhantes. Foi então proposto que um conjunto de métricas fossem estabelecidas para que as equipes pudesse ser acompanhado por igual pela gerência (Rocha, A. et al.2012).

*Alternativa:*

O Gerente da Fábrica atribuiu aos SM (Scrum Masters), a tarefa de definirem em conjunto quais as métricas relevantes de acompanhamento para as equipes, de forma a nivelar o acompanhamento.

A QA (Responsável pela Qualidade) e o Gerente da Fábrica formataram um modelo de plano de medição com o apoio da consultoria, e posteriormente foi feito um encaixe da métrica de acordo com a gerência (Tabela 1), pois para o nível F, se faz necessário uma métrica para cada processo, ou seja, através do RAP 4.

**Tabela 1. Métricas de acompanhamento estabelecidas**

Gerencia	Métrica	Propósito
Organizacional	Índice de Horas Passíveis de Cobrança	Indica o percentual de horas passíveis de cobrança com relação ao total de esforço dedicado ao cliente
	Desvio de Estimativa	Demonstra o desvio da estimativa de esforço
Projetos	Retrabalho	Avalia a capacidade do time comprometida com retrabalho
	Tempo de Ciclo	Avalia a fluidez da cadeia de desenvolvimento
	Burndown	Demonstra o histórico do progresso diário, o progresso ideal para término da iteração no prazo.
Configuração	%Solicitações de Mudança Processadas	Avaliar as mudanças em Gerencia de Configuração
Requisitos	Velocity	Avalia a capacidade do time em cumprir com os Story Points de um sprint
	Taxa de defeitos	Demonstra a taxa de defeitos encontrados durante a sprint com relação às cartas testadas
Qualidade	%Não Conformidades do Projetos	Avaliar a aderência entre os padrões estabelecidos e os processos executados
Medição	Taxa de Medidas	Esta medida tenta sumarizar a situação geral de todas as medidas coletadas

## 7. Gerencia de Garantia da Qualidade – GQA

### 7.1. Fazer acontecer ou RAP 10

Fazer com que os processos fossem institucionalizados, foi e ainda é, a missão mais complexa. A RAP 10 no modelo coloca que deve haver aderência entre os processos executados e suas descrições. Para a equipe técnica, ainda não são perceptíveis os benefícios que o modelo MPS.br fomenta. Em partes é pelo fato de ser uma equipe que mantém constante evolução de seu processo de desenvolvimento, e seu foco está na qualidade de código e na evolução das pessoas, o que faz com que tenhamos inúmeras ações sendo executadas ao mesmo tempo em função do mesmo propósito.

*Alternativas:*

Uma estratégia adotada por recomendação da consultoria foi um “quizz”, onde ao acertar a resposta de uma pergunta relacionada à metodologia ou as necessidades impostas pelo modelo, o colaborador ganha um prêmio. No nosso caso foi um chocolate. Isto serviu para despertar em cada um o conhecimento da localização e do conteúdo da documentação que passou a existir. As auditorias constantes e a comunicação durante a Daily Scrum sobre as Não – Conformidades detectadas, provocou uma maior atenção dos colaboradores com relação à seguir os processos. A presença do QA durante as cerimônias como um real integrante do time, também trouxe uma maior presença do MPS.br à realidade da Fábrica JAVA.

## 8. Conclusão

Com relação as métricas estabelecidas, a de GCO ainda é uma métrica frágil para se manter futuramente, pelo fato de não haver problemas críticos com a engenharia, em virtude da excelência no desempenho das atividades do engenheiro de configuração atual.

Antes da implementação as cartas, possuíam poucas informações, porém, em virtude da implementação do nível “F” do modelo MPS.Br, hoje elas são mais completas mantendo o histórico do nosso produto e informações que podem vir a ser úteis no futuro.

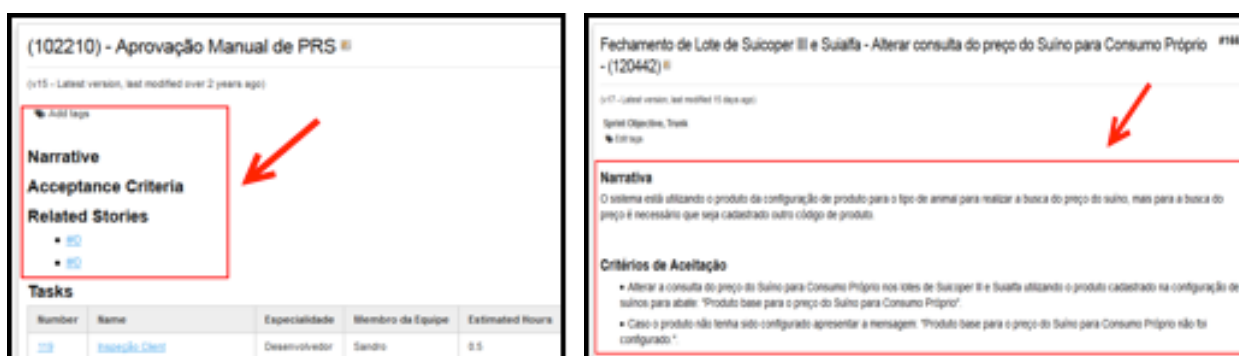


Figura 6. A imagem a esquerda é de uma carta de antes do MPS.Br, e a da direita após a nova implementação do modelo nível “F”

O fato da implementação do modelo não partir de uma necessidade da área, por vezes faz com que a organização apresente resistências. Como o MPS.Br – SW não é facilmente adaptável ao SCRUM, houveram vários impedimentos ou restrições por parte da equipe, devido ao fato de não sentirem que uma exigência do modelo era realmente necessária, e acabava enxergando isso como burocratização do processo.

Uma estratégia para mitigar o impacto das diferenças entre o modelo MPS.Br e a metodologia empregada na fábrica foi iniciar as definições e documentações pelas práticas já existentes na área e a equipe de processo foi composta pela QA, o Gerente da Área e o Consultor do SENAI. Pelo fato de estarmos constantemente buscando a maneira mais simples de atender ao modelo, o apoio da consultoria para a melhor interpretação do modelo com a realidade da empresa e o SCRUM.

## 7. Referências

- Beck, K. et al., Manifesto for Agile Software Development, [www.agilemanifesto.org](http://www.agilemanifesto.org), (2001)
- Boehm, B., A View of 20th and 21st Century Software Engineering, ICSE 2006.
- Boria, et al. “A História de Tahini – Tahini: Melhoria de Processos de Software com Métodos Ágeis e Modelo MPS” . Secretaria de Política de Informática (2013)
- PMI,(2013) “PMBOK – A Guide to the Project Management Body of Knowledge”, 5th Edition.
- Poppendieck, Mary (2011) “ Implementando o Desenvolvimento Lean de Software – Do conceito ao Dinheiro” . Editora: Bookman.
- Rocha, A. et al., “Medição de Software e Controle Estatístico de Processos” . Secretaria de Política de Informática (2012)
- Kninberg, Henrik. “Scrum e XP direto das Trincheiras – Como fazemos Scrum” . C4Media, Publisher of InfoQ.com (2007)
- SOFTEX (2012), “Guia Geral MPS de Software” . Agosto, 2012.



# SKAM: Um Processo usando Scrum e Kanban para Customização de Software em Dispositivos Móveis

Jonh N. B. Sousa<sup>1</sup>, Alberto H. M. T. Monteiro<sup>1</sup>, Andrea F. Oliveira<sup>1</sup>, Frederico V. da Silva, Ismayle S. Santos<sup>1,1</sup>, Reinaldo B. Braga<sup>1</sup>, Rossana M. C. Andrade<sup>1,2</sup>

<sup>1</sup>Grupo de Redes de Computadores, Engenharia de Software e Sistemas (GREat) Universidade Federal do Ceará (UFC)

Fortaleza, Ceará, Brasil

{jonhsousa,albertohugo,andreaoliveira,ismaylesantos,reinaldo,rossana}@great.ufc.br, fredvc@gmail.com

**Abstract.** *There are different proposals for the agile methodologies in the literature that can be used in software project development and management. However, one of the main challenges that remains is to choose the best process model to be used and to determine whether adjustments are necessary in the chosen model or not. In this context, this article presents an experience report on the practical use of Scrum and Kanban in the customization of operating systems for mobile devices. Based on the acquired experience, we propose a process, called SKAM, bringing together practices of Scrum and Kanban. We also present the lessons learned and the results of a survey conducted to measure the degree of adaptability of the team with the SKAM.*

**Resumo.** *Este artigo apresenta um relato da experiência da definição e aprimoramento de um processo com foco no desenvolvimento de customizações de sistemas operacionais para dispositivos móveis. O processo proposto foi denominado de SKAM e utiliza como base práticas do Scrum e o Kanban. São também apresentadas as lições aprendidas durante a definição do SKAM, bem como os resultados de um survey aplicado junto a equipe de desenvolvimento para avaliar o grau de adaptabilidade da equipe com o processo proposto. Os resultados obtidos revelaram indícios de que o SKAM é adequado para as necessidades da equipe, mas que algumas ações devem ser tomadas, como por exemplo, executar mais treinamentos.*

## 1. Introdução

O desenvolvimento de customizações de software embarcado é necessário para adaptar o código base do modelo de celular para atender às especificações de interface com o usuário, funcionalidades e configurações solicitadas pelas operadoras [Cunha et al. 2011]. Além disso, é de interesse das fabricantes e operadoras de telefonia móvel que os dispositivos produzidos e lançados no mercado carreguem consigo diferenciais competitivos que os destaquem dos produzidos pelos concorrentes.

Nesse contexto, além da customização de hardware (e.g., configurações em termos de processador e memória), tem-se explorado customizações de software em caráter tanto publicitário quanto funcional. O caráter publicitário trata da criação de uma identidade visual da operadora no dispositivo

---

<sup>1</sup> Bolsista de Doutorado da CAPES

<sup>2</sup> Bolsista do CNPq de Produtividade DT-2 com o número de processo 314021/2009-4

móvel, fazendo com que os dispositivos possuam características específicas da operadora. O caráter funcional trata das mudanças solicitadas pelas operadoras com relação às funcionalidades do dispositivo.

Logo, como apresentando em [Lima et al. 2010], uma determinada operadora de telefonia pode se interessar em ter sua logomarca configurada como papel de parede padrão e pode exigir que o dispositivo seja comercializado com todos os perfis de conectividade devidamente configurados. Essas customizações da logomarca e dos perfis de conectividade são exemplos de caráter publicitário e funcional.

O GREat<sup>3</sup>, em um dos seus projetos de pesquisa, desenvolvimento e inovação realiza a customização de sistemas operacionais em dispositivos móveis. Ao longo desse projeto em particular, percebeu-se que uma das suas principais características é a dinamicidade das atividades. Essa dinamicidade pode ser percebida em termos de: a) escopo, com requisições variáveis por parte das operadoras; e b) prazo, devido ao tempo de desenvolvimento “elástico”, variando entre poucos dias a meses de atividade, dependendo do interesse da operadora, o qual é influenciado diretamente pelo mercado.

Com o intuito de manter o escopo e o prazo variáveis sob controle, a equipe do projeto optou por utilizar as metodologias ágeis Scrum e Kanban. Também foi adotado um conjunto de ferramentas previamente configuradas para auxiliar na gestão do projeto. Contudo, percebeu-se que certas características das metodologias utilizadas, como as *sprints* do Scrum, eram difíceis de serem adotadas neste projeto.

Na literatura é possível encontrar trabalhos [Nikitina e Kajko-Mattsson, 2012; Brasil et al (2013)] que relatam melhorias ou adaptações de processos já existentes para se adequar as características do desenvolvimento de software em questão. Nikitina e Kajko-Mattsson (2012), por exemplo, descrevem uma experiência sobre a transição do Scrum para o Scrumban. Brasil et al. (2013), por sua vez, propõem uma abordagem para melhoria da qualidade de processos de software a partir do reuso de fragmentos de métodos, utilizando os conceitos definidos pelo *Situational Method Engineering* (SME) associados com o guia para Melhoria de Processo de Software Brasileiro (MPS.BR).

Este artigo descreve na Seção 3 a proposta de um processo, denominado de SKAM, feita com objetivo de atender as necessidades do projeto de customização de sistemas operacionais em dispositivos móveis. Este processo foi definido com base nas experiências da equipe, relatadas na Seção 2, quanto ao uso do Scrum e Kanban, bem como de outras práticas de gestão no processo. Além disso, este artigo apresenta na Seção 4 os resultados de um *survey* aplicado com o objetivo de mensurar a adaptabilidade da equipe ao processo proposto. As conclusões são descritas na Seção 5.

## 2. Experiências Adquiridas

Nesta seção são descritas as experiências da equipe que foram mais relevantes na definição do SKAM.

---

<sup>3</sup> Grupo de Pesquisa em Redes de Computadores, Engenharia de Software e Sistemas, que é vinculado ao Departamento de Computação da Universidade Federal do Ceará e, entre outras áreas de interesse, desenvolve aplicações para dispositivos móveis (i.e., smartphones, celulares, tablets). <http://great.ufc.br>

## 2.1. Uso do Scrum

Segundo Schwaber e Sutherland (2013), o Scrum é considerado um framework estrutural criado em 1990 para gerenciar o desenvolvimento de produtos complexos, sendo utilizado com maior frequência no desenvolvimento de software. Assim como outras metodologias ágeis, a entrega rápida de partes do software é uma de suas particularidades, adequando-se bem a projetos de constante mudança de requisitos. Seu ciclo de vida se dá por iterações chamadas *sprints*.

A primeira aplicação do Scrum no projeto em questão foi feita em 2008. Em 2013, foi incorporado junto ao Scrum uma metodologia de gestão de qualidade para aumentar a produtividade da equipe, conforme apresentado em [Cunha et al. 2011]. Contudo, em meados de 2013 percebeu-se que uma adaptação era necessária no processo, pois algumas dificuldades impediam o uso fiel as prescrições do Scrum. As principais dificuldades foram:

- *Tamanho da Equipe*. A equipe do projeto era de 32 pessoas, e nem sempre era possível dividir a equipe em subprojetos para ficar mais próxima da indicação máxima do Scrum, que seria de nove pessoas;
- *Definição das Sprints*. O cliente possuía datas bem definidas de entregas baseadas em sua equipe de *marketing* externo e políticas internas, dificultando que a organização da *Sprint* fosse feita pela equipe de desenvolvimento;
- *Acumulo de Papéis*. Muitas vezes os líderes dos projetos de customizações acumulavam papéis, exercendo também o que seria o *Scrum Master* ou *Product Owner*. Dessa forma, quando havia uma maior demanda de trabalho, os papéis do Scrum acabavam ficando com menor prioridade, frente às solicitações urgentes do cliente.

## 2.2. Sistematização do uso do Kanban por meio do Jira

Segundo Kniberg (2009), Kanban é um método de desenvolvimento ágil de software, também considerado como uma ferramenta de processo. Para utilizar o Kanban de forma correta, algumas etapas devem ser seguidas, tais como: visualização do fluxo de trabalho e divisão do mesmo em etapas; divisão das tarefas a serem realizadas; definição de uma hierarquia de atividades entre essas tarefas; e limitação da quantidade de tarefas que podem ser executadas simultaneamente em cada uma das etapas (definição de uma *WIP – Work in Progress*).

No projeto em questão, utilizou-se o Jira<sup>4</sup> para o uso do quadro Kanban. Esta ferramenta vem sendo utilizada para diversos fins: banco de dados de atividades e projetos, planejamento de *sprints*, registros de lançamentos de *software*, acompanhamento das atividades com o uso de gráficos, reporte de horas, entre outras funcionalidades. Neste cenário, algumas customizações foram aplicadas ao Jira para atender às necessidades do projeto. Na Figura 1 é apresentado um dos fluxos criados, relacionado ao ciclo de desenvolvimento de uma *issue* de customização, bem como a correspondência deste para um quadro Kanban. Uma *issue*, dentro do Jira, pode ser vista como uma unidade básica de tarefa que possui um identificador único e pertencente a um único projeto. As *issues* possuem um atributo denominado de 'tipo' que pode, por exemplo, representar um bug, um impedimento ou um *story point*.

<sup>4</sup> Ferramenta de Gerenciamento de Projeto. Disponível em <https://www.atlassian.com/software/jira>



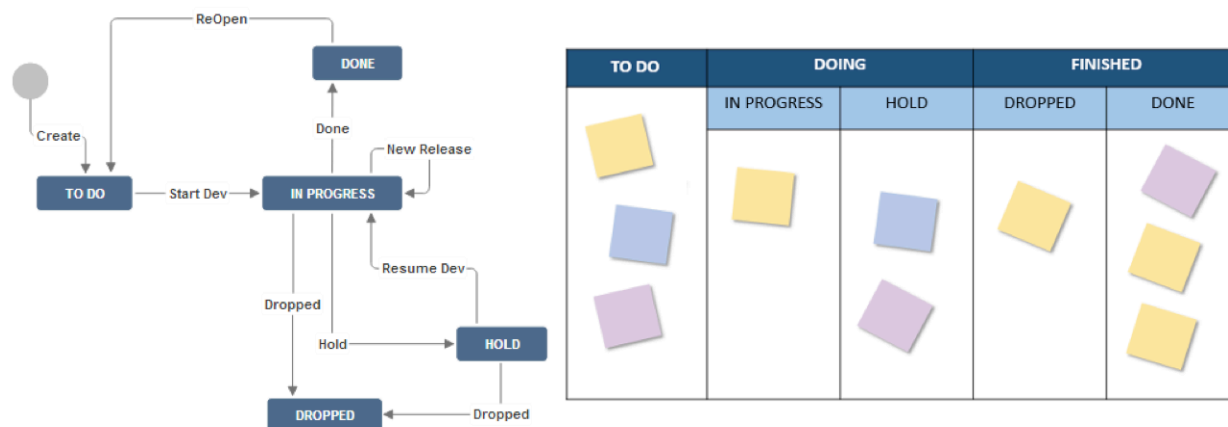


Figura 1. Fluxo de trabalho de uma *issue* do Jira

É importante ressaltar que, com o amadurecimento dos processos internos de desenvolvimento, foi possível notar uma série de atividades que se repetiam a cada novo pacote de customização que era requisitado pelo cliente. A partir desse levantamento, a equipe interna responsável pelo Jira trabalhou para criar um conjunto customizações nos fluxos e nas ferramentas do Jira. Este trabalho resultou em um framework para o desenvolvimento dos projetos de customizações de softwares para celulares de forma que todas as fases do desenvolvimento pudessem ser seguidas pelo Jira. As atividades automatizadas vão desde a criação de *tasks*, padrões de desenvolvimento, teste e lançamento até o envio de e-mails de notificação aos responsáveis pelas atividades editadas. Tudo dentro de um fluxo adequado para cada atividade do desenvolvimento. Com a criação desse framework, foi possível diminuir os riscos de descumprimento de alguma parte do processo ou atividade. Além disso, serviu para facilitar o entendimento do processo de desenvolvimento utilizado.

### 2.3. Uso de um Quadro de Alocação e de Acompanhamento Semanal

Com o intuito de gerir de forma mais eficiente à alocação dos recursos humanos do projeto, foi implementado em 2013 um Quadro de Alocação. Em formato de tabela, este quadro possuía duas colunas: uma para o nome do membro da equipe e outra destinada às atividades nas quais ele estivesse envolvido. Essas atividades eram fixadas no quadro por meio de cartões *post-it*. Esses cartões, por sua vez, subdividiam-se em categorias, definidas por cores (vermelho, amarelo, azul, dentre outras), que representavam cada uma das macroatividades realizadas.

Para complementar o Quadro de Alocação, foi definido também um Quadro de Acompanhamento Semanal de atividades. Esse quadro é composto pelos objetivos dos subprojetos que fazem parte do plano de trabalho da equipe. No início de cada ciclo de trabalho (em torno de uma semana) era realizada uma reunião de acompanhamento com os grupos responsáveis por cada um desses subprojetos, analisando os objetivos a cumprir até o fim do mesmo. Nessa mesma reunião eram revisados os objetivos definidos na reunião do ciclo anterior, verificando quais deles foram de fato alcançados bem como àqueles que tiveram complicações e não puderam ser realizados. No fim de 2013, o Quadro de Alocação foi substituído pela ferramenta Jira na gestão das atividades realizadas nos projetos, já que a mesma possuía um quadro semelhante. Esta mudança facilitou o processo de visualização, controle e principalmente de geração de relatórios de atividades devido à natureza dinâmica das atividades realizadas no projeto.

## 2.4. Uso do Skype para comunicação interna

O Skype<sup>5</sup> é um programa de distribuição gratuita (*freeware*) que permite conversações simultâneas de áudio e vídeo através de computadores. Como forma de agilizar a comunicação entre os membros da equipe, a utilização do Skype como ferramenta de envio de mensagens foi adotada. Possuindo recursos como a formação de grupos e recursos multimídia de comunicação em tempo real, viu-se nele uma ferramenta útil na realização de reuniões de planejamento (definição de cronogramas, envio de avisos e etc), acompanhamento dos projetos (discussões técnicas e envio de dúvidas), controle de atividades (verificação de estado de atividades junto aos responsáveis) e geração de relatórios (servindo de repositório para informações relevantes com relação às atividades realizadas). Essa ferramenta ainda é utilizada no projeto, pois continua respondendo bem às necessidades de comunicação interna (entre a equipe de desenvolvimento e gestão) e também externa (comunicação direta com o cliente).

## 3. SKAM: o processo proposto

Conforme apresentado na Seção 2, em meados de 2013 surgiu a necessidade da busca de uma adaptação do processo atual para melhor atender as necessidades do projeto de customização de software móvel. Com este objetivo e com o apoio da alta gerência, houveram desde junho de 2013 reuniões semanais e dedicação de carga-horária fixa de pesquisadores na análise do processo no projeto.

A partir das pesquisas conduzidas percebeu-se que uma adaptação do Scrum junto com o Kanban poderia atender as necessidades do projeto. Assim, com base na experiência obtida (ver Seção 2) e no estudo do ScrumBan [Nikitina, Kajko-Mattsson, 2012], que une o Scrum e o Kanban, foi definido o SKAM. É importante destacar que o diferencial do SKAM para o ScrumBan está no foco do SKAM no projeto de customização de software para dispositivos móveis. A Figura 2 apresenta o SKAM, modelado na linguagem BPMN<sup>6</sup> (*Business Process Model and Notation*), utilizando a ferramenta Bizagi<sup>7</sup>. Esta versão do SKAM foi consolidada em abril de 2014, após várias reuniões com toda a equipe.

A primeira atividade do processo é o *Planejamento*, que é composta por três reuniões: uma reunião de *kick-off*, uma dos líderes técnicos e uma com a equipe. A reunião de *kick-off* envolve a liderança do projeto e o cliente, definindo escopo, prazos e indicadores do projeto. A reunião de líderes técnicos, tem como resultado a organização dos recursos disponíveis. Por fim, a reunião realizada com a equipe tem o objetivo de definir a alocação de recursos humanos, estipulando o limite de esforço por recurso, definindo a frequência de reuniões de acompanhamento, revisando pontos de melhoria e de sucesso de projetos anteriores e adaptando o projeto de acordo com esses pontos.

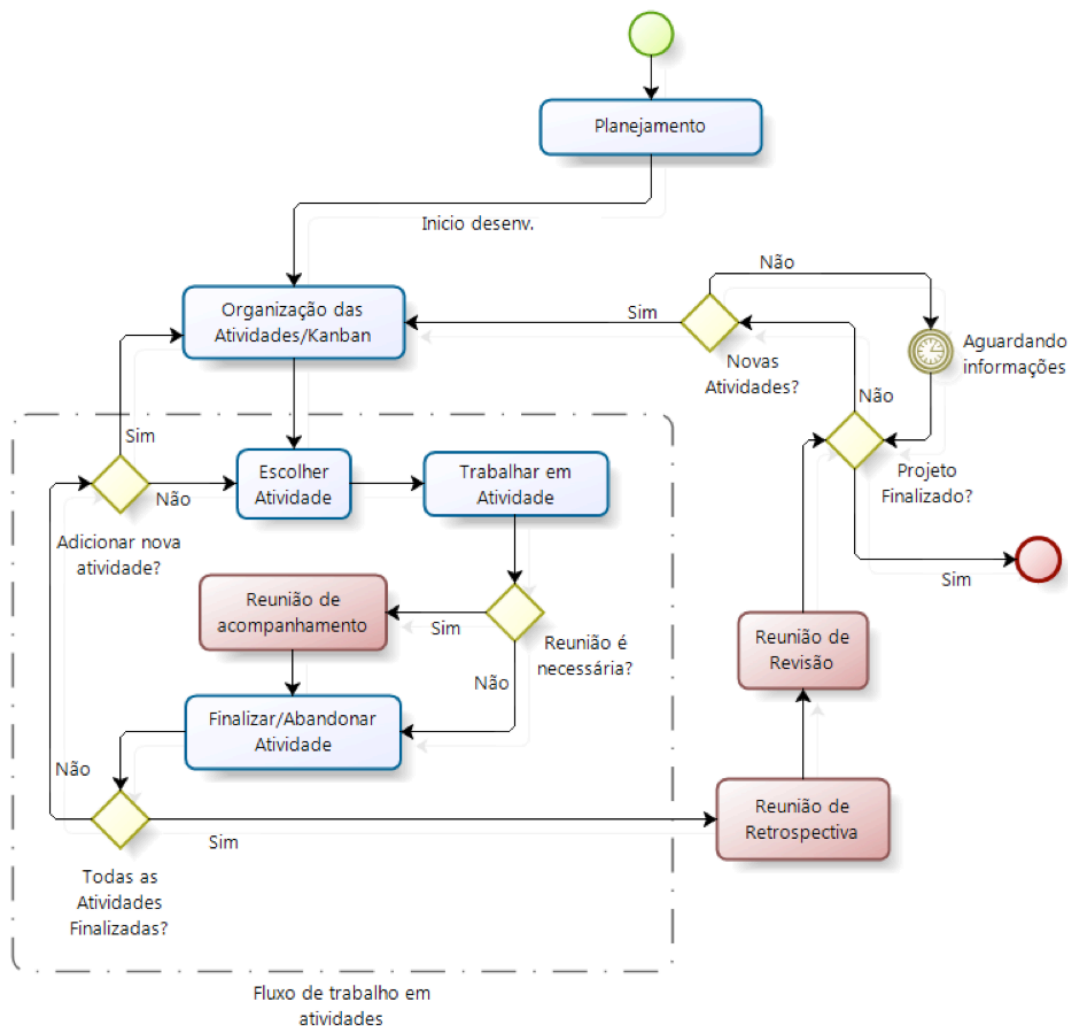
Ao término da atividade de planejamento, inicia-se a etapa de *Organização das Atividades*. No contexto do projeto, uma atividade pode ser representada pelo desenvolvimento de uma nova funcionalidade, adição ou remoção de componentes de *software*, testes e até mesmo lançamentos de versões de *software* desenvolvidas. Nessa etapa, as atividades do projeto são listadas utilizando um quadro Kanban na ferramenta Jira, possibilitando sua visualização e alocação por parte da equipe envolvida.

---

5 [http:// www.skype.com](http://www.skype.com)

6 <http://www.omg.org/spec/BPMN>

7 <http://www.bizagi.com/>



**Figura 2. Processo de desenvolvimento voltado para o projeto de customizações para dispositivos móveis**

Após a organização das atividades, inicia-se o Fluxo de Trabalho em Atividades, composto pela escolha de atividade, o trabalho em atividade e a verificação de necessidade de reunião de acompanhamento. Em seguida, é realizada uma análise objetivando avaliar a condição de completude (finalização ou abandono) das atividades em execução. Após a execução das atividades planejadas, verifica-se a necessidade de adicionar novas atividades ao quadro Kanban, o que faria necessária uma nova etapa de organização. Caso não existam mais atividades, o *Fluxo de Trabalho em Atividade* é encerrado, gerando a necessidade de uma reunião de retrospectiva e outra de revisão.

Um dos diferenciais da forma de trabalho adotada é que o fim do projeto não é algo bem definido, pois, após a reunião de revisão, o projeto pode ser encerrado ou pode entrar em fase de espera, aguardando requisições de melhorias e atualizações no software desenvolvido por parte do cliente. Caso alguma dessas necessidades apareça, novas atividades devem ser cadastradas, fazendo com que uma nova etapa de *Organização de Atividades* deva ser realizada, a fim de atualizar o quadro Kanban e possibilitar que recursos humanos possam ser alocados na resolução dessas atividades.

## 4. Avaliação inicial do SKAM

Com a versão do SKAM apresentada na Seção 3 já foram executados três ciclos de desenvolvimento. Para uma primeira avaliação deste processo, foi conduzido um *survey* por meio de um questionário com a equipe, procurando identificar a visão da mesma para com este processo. O questionário aplicado continha questões sobre, dentre outras coisas, a definição das atividades a serem realizados, o grau de mudanças nessas atividades e o nível de conhecimento dos membros da equipe em relação às metodologias envolvidas no processo (Kanban e Scrum). No total, o questionário possuía 15 questões, nas quais o respondente podia indicar um valor entre 0 e 5, onde “5” representa o máximo na escala e “0” o mínimo. Na Tabela 1 é apresentada a média das repostas fornecidas pela equipe.

**Tabela 1. Média dos resultados obtidos no survey**

Id	Pergunta	Pontuação
1	Nível de conhecimento acerca do processo de desenvolvimento adotado	3,70
2	Nível de dinamicidade do fluxo de atividades no projeto	3,70
3	Frequência na alteração do escopo de atividades	2,91
4	Frequência na alteração do prazo de atividades	2,91
5	Nível de importância da etapa de organização de atividades	4,54
6	Frequência em que ocorre a etapa de organização de atividades	3,33
7	Importância da participação da equipe de desenvolvimento nessa etapa	4,16
8	Nível de importância de se ter autonomia na escolha de atividades	3,87
9	Nível de conhecimento em Kanban	2,50
10	Nível de importância da criação de um quadro Kanban	3,54
11	Nível de conhecimento em Scrum	3,41
12	Nível de importância da existência de sprints bem definidas	4,12
13	Nível de adequação no uso da ferramenta de gerenciamento de atividades	3,45
14	Frequência em que ocorrem reuniões de revisão e retrospectiva no projeto	3,25
15	Nível de adequação do processo de desenvolvimento adotado	3,66

Com este *survey* foi possível observar que a equipe possuía um bom nível de conhecimento do processo utilizado, com uma média de resposta de 3.7 (Id 1). Também foi possível perceber um certo conhecimento em Kanban e Scrum, mas relevou-se a necessidade de mais treinamento nestes pontos. Provavelmente isso é consequência do fato de boa parte da equipe ter sido renovada pouco tempo antes deste *survey*.

Além disso, houveram indícios de que a ferramenta Jira está a contento, mas alguns dos participantes sugeriram melhorias nos fluxos de atividades. Percebeu-se também que um treinamento prático sobre a ferramenta resolveria parte dos problemas de falhas de processo, pois muitos deles estavam acontecendo por desconhecimento da forma adequada de trabalho. A adequação do processo adotado com relação às atividades realizadas também foi questionada. Como pode-se observar, foi obtida uma pontuação de 3.66 (Id 15), o que pode indicar algumas falhas quanto à dinamicidade na administração das atividades. Com base nos resultados do questionário e a observação prática, o SKAM apresentou um desempenho até então satisfatório, porém para o melhor uso do mesmo ainda é preciso a capacitação da equipe com relação às metodologias utilizadas no processo (Scrum e Kanban) e nos processos internos de desenvolvimento (o uso adequado da ferramenta Jira em conjunto as atividades de desenvolvimento).

## 5. Considerações Finais e Trabalhos Futuros

Este artigo apresentou o relato de experiência do GREat em um projeto voltado para customização de sistemas operacionais em dispositivos móveis. O foco do relato foi a proposta de um processo utilizando práticas do Kanban e do Scrum, adaptado às necessidades da equipe. O processo baseou-se na experiência obtida durante o projeto e as lições aprendidas também foram relatadas no artigo. Embora não tenham sido coletados dados quantitativos, foi possível conduzir uma avaliação inicial do processo proposto por meio de um *survey* junto à equipe. Conforme os resultados deste *survey* e da observação na prática, percebeu-se que o processo possui um desempenho satisfatório junto ao modelo de trabalho adotado no projeto.

Visando a utilização mais eficiente do SKAM, como trabalhos futuros, serão realizados treinamentos de atualização com relação às metodologias e processos internos de desenvolvimento que compõem a proposta junto à equipe, bem como a atualização dos fluxos da ferramenta de controle de atividades.

Após a atualização da equipe, será realizado um trabalho utilizando o modelo de melhoria de processos MPS.BR, que é um modelo de melhoria de processos de software, criado em 2003. Vale ressaltar que este modelo volta-se para a realidade de empresas brasileiras, com o objetivo de propor um modelo de processo para alcançar a Melhoria do Processo de Software Brasileiro [Koscianki, 2007]. Além disso, é considerada a dinamicidade do projeto utilizado como base na criação do SKAM. Portanto, o foco desse trabalho conjunto (SKAM e MPS.BR) será GPR (Gerência de Projetos) e GRE (Gerência de Requisitos), visando tornar mais fácil a gerência do projeto em questão, gerenciando os requisitos do produto, identificando inconsistências entre requisitos, planos de atividades e produtos de trabalho do projeto, visando otimizar as atividades previstas no SKAM. Após os treinamentos e o trabalho realizado com SKAM e MPS.BR, pretende-se avaliar novamente o SKAM.

## Referências

- Brasil, M. A.B.; Fontoura, L.M.; Silva, L.A.L. (2013) Uma Proposta para Melhoria da Qualidade de Processos de Software com base em MPS.BR. Simpósio Brasileiro de Qualidade de Software (SBQS).
- Cunha, T. F. V.; Dantas, V. L. L.; Andrade, R. M. C. (2011) SLeSS: uma abordagem de integração do Scrum e Lean Six Sigma no desenvolvimento de customizações de software para celulares. Simpósio Brasileiro de Engenharia de Software (SBES).
- Kniberg, Henrik. (2009) “Kanban em poucas palavras”, em: Kanban e Scrum – Obtendo o melhor de Ambos, p. 25 – 27.
- Kosciank, A.; Soares, M. S. Qualidade de Software: Aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software. 2ª Edição. São Paulo: Novatec Editora, 2007.
- Lima, F. F. P.; Nogueira, T. P.; Nogueira, A. D. B.; Neto, J. S. R.; Filho, J. B. F.; Lima, C. R. F.; Leite, S.J. C.; Viana, W.; Dantas, V. L. L.; Andrade, R. M. C. (2010) Manutenção Adaptativa de Software Embarcado para Telefones Celulares Apoiado por Ferramentas de Automação. Simpósio Brasileiro de Qualidade de Software.
- Nikitina, N.; Kajko-Mattsson, M. (2012) From Scrum to Scrumban: A Case Study of a Process Transition. International Conference on Software and System Process
- Schwaber, K.; Sutherland, J. (2013) “Scrum Guide”. Disponível em: <<http://www.scrum.org/>>. Acessado em 12 Abril de 2014.

# Identificação de Fatores que Influenciam a Produtividade em Projetos de Software

Reinaldo Cabral e Rafael Morais

Inform Sistemas Ltda. – Av. Humberto Mendes, 189, Jaraguá, 57020-580, Maceió-AL

{reinaldo, rafael}@informsistemas.com.br

**Abstract.** *The development of predictive capability of a software company requires the application of practices inherent to high maturity software processes. These practices stimulate the search for factors that influence the behavior of processes. Finding influential factors is not trivial, especially given the diversity of factors and the limited number of project data from historical basis of small and mid-sized enterprises. This paper presents an experience aiming to identify factors that influence the productivity of projects through the application of high maturity practices.*

**Resumo.** *O desenvolvimento da capacidade preditiva de uma empresa de software requer a aplicação de práticas inerente à alta maturidade em processos de software. A aplicação de tais práticas estimula a busca por fatores que influenciam o comportamento dos processos, que não é trivial, especialmente devido à diversidade de fatores e o número limitado de dados de projetos que constituem a base histórica de pequenas e médias empresas. Este trabalho apresenta uma iniciativa com o intuito de identificar fatores que influenciam a produtividade dos projetos por meio da aplicação de práticas da alta maturidade.*

## 1. Introdução

Inúmeros fatores podem influenciar a execução de projetos e contribuir com a incidência de desvios de esforço, custo e prazo. Wagner e Ruhe (2008) realizaram uma revisão sistemática e identificaram 24 fatores técnicos e 27 fatores não técnicos capazes de influenciar a produtividade. Numa perspectiva da alta maturidade em processos, alguns destes fatores podem incidir como causas comuns de variação, isto é, inerente ao comportamento do processo, ou como causas especiais em função da incidência de fato incomum que afeta o comportamento do processo. Cada fator pode ter uma parcela de contribuição nesta variação. Portanto, para entender o comportamento dos processos sob a ótica da produtividade e aperfeiçoá-los é necessário identificar os fatores mais relevantes.

Este trabalho realiza uma investigação em busca dos fatores que mais influenciam a produtividade no contexto de uma fábrica de software a partir da utilização de práticas da alta maturidade. O trabalho está organizado em cinco seções, incluindo esta introdução. A segunda seção apresenta ferramentas que podem ser úteis na detecção dos fatores que podem influenciar o esforço. A seção três descreve o procedimento proposto. Em seguida, a aplicação do procedimento é exemplificada em uma base de projetos de uma fábrica de software. A seção 5 discorre sobre os resultados obtidos e restrições para uso do procedimento. Por fim, são realizadas considerações finais na última seção.

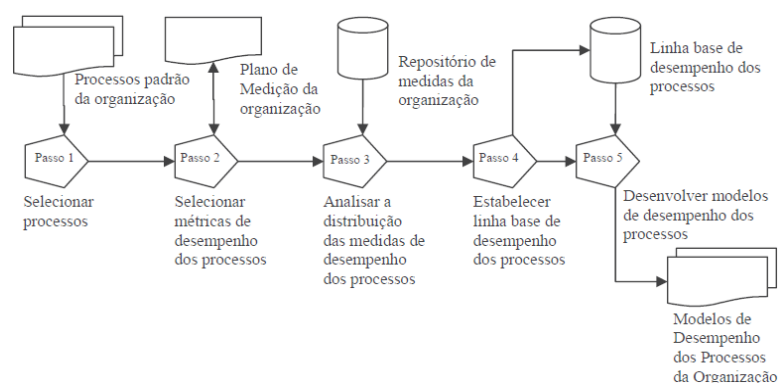
## 2. Ferramentas para detectar e quantificar influências em projetos de software

O uso de procedimentos estatísticos para quantificar a influência entre variáveis é uma das ferramentas mais utilizadas no contexto da engenharia de software. Há uma ampla gama de possibilidades. É preciso compreender cada medida estatística sua forma de aplicação para que não haja equívocos nos resultados e na interpretação. Dentre as medidas mais simples, destacam-se: a variância, a covariância, a correlação e o coeficiente de determinação. Estas medidas, em conjunto com uma análise de variabilidade do processo podem ser bastante úteis na quantificação dos fatores de influência em projetos de software.

Modelos de maturidade como o MR-MPS-SW (SOFTEX, 2012) e CMMI (SEI, 2010), sugerem o uso do controle estatístico de processos para auxiliar na identificação de incidência de fatores que atuam na variação dos processos em prol. Uma vez que os processos estejam estáveis, a previsibilidade aumenta e a identificação dos fatores comuns de variação pode auxiliar no aumento da capacidade e na criação de modelos preditivos acurados. Se o processo está estável, há apenas causas comuns agindo sobre ele. Do contrário, há causas atribuíveis.

Florac e Carleton (1999) propuseram um arcabouço para mensurar o comportamento dos processos e caracterizá-los quanto à estabilidade e capacidade. Os autores sugerem que a organização inicie por uma reflexão sobre os seus objetivos de negócio e do entendimento de quais processos podem dar maior contribuição. Em seguida, deve-se identificar e priorizar as questões a serem tratadas para que os processos atinjam os objetivos de negócio. Medidas devem ser identificadas, definidas, coletadas e analisadas para auxiliar na caracterização dos processos ou produtos. Ao avaliar o comportamento do processo com relação à estabilidade, as causas especiais de variação são identificadas e removidas. Ao atingir a estabilidade o processo é continuamente melhorado em prol do aumento da capacidade.

Montoni et al. (2007) propuseram uma metodologia para criação de modelos de desempenho constituída por um processo para desenvolver modelos de desempenho para gerência quantitativa de projetos de software (Figura 1). Os autores ilustraram a aplicação da metodologia no contexto de uma empresa de software CMMI nível 3, resultando a construção de um modelo para monitorar quantitativamente os projetos de desenvolvimento de software a partir da predição da qualidade do produto sob o ponto de vista do seu conteúdo.



**Figura 1. Processo de desenvolvimento de modelos de desempenho de processos para gerência quantitativa de projetos de software (Montoni et al., 2007)**

Rocha, Souza e Barcellos (2012) provêm detalhes sobre conceitos básicos e a aplicação adequada do controle estatístico de processos de software.

### 3. Contexto da Experiência

A Inform Sistemas Ltda., fundada em 1991, é uma empresa de software Alagoana que vem investindo sistematicamente em melhoria dos seus processos utilizando o MR-MPS como modelo de referência desde 2009, ano em que obteve o MPS nível F. Em 2013 a empresa foi avaliada com êxito no nível C do MR-MPS-SW. A empresa possui uma unidade de fábrica de software que atua no desenvolvimento de aplicações nas plataformas Android, iOS e Web e que também teve seus processos avaliados no nível C. A Figura 2 ilustra a visão macro do Processo Fabril que atualmente está na versão 1.18.

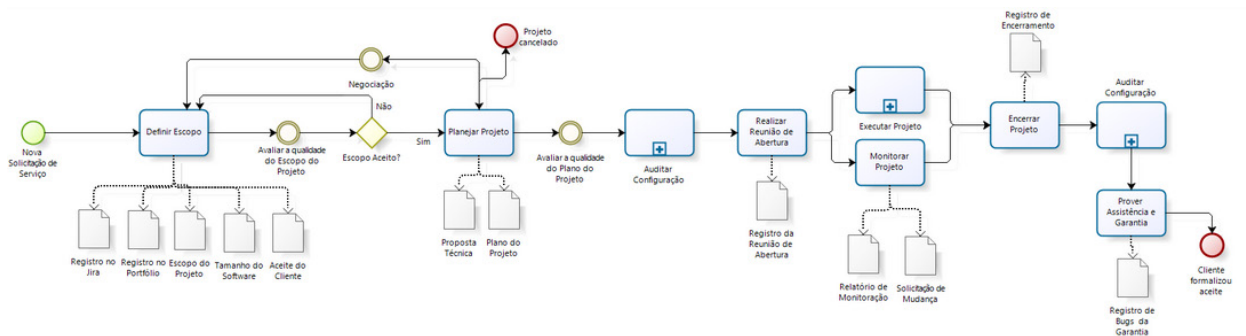


Figura 2. Processo Fabril

Um dos principais indicadores para a Fábrica de Software é a produtividade, mensurada pela razão entre esforço realizado e o tamanho do projeto. O ISBSG<sup>1</sup> também utiliza esta razão como medida de produtividade, porém utiliza o termo taxa de entrega de projetos para referenciá-la.

Nesta experiência foram considerados dezesseis projetos da base histórica da unidade fabril. Os projetos foram caracterizados com relação a um conjunto de fatores que poderiam influenciar a variação da produtividade de acordo com a visão dos líderes de projeto e com os fatores relacionados por Wagner e Ruhe (2008) considerados pertinentes ao contexto organizacional da unidade fabril.

### 3. Identificando fatores que afetam a produtividade

A Inform Sistemas possui como um dos seus objetivos de negócio a necessidade de aumento do faturamento. Portanto, os processos produtivos, neste caso, o processo de desenvolvimento da fábrica de software, possui uma relação direta com este objetivo, uma vez que a melhoria da taxa de entrega poderá viabilizar o aumento da capacidade para atender um número maior de demanda por serviços e, conseqüentemente, possibilitará a expansão da carteira de projetos.

Os procedimentos realizados com o intuito de identificar os fatores que afetam a produtividade foram baseados no arcabouço proposto por Florac e Carleton (1999) e Montoni et al. (2007). Em

<sup>1</sup> International Software Benchmarking Standards Group. <http://www.isbsg.org>



síntese, utiliza-se a avaliação de estabilidade de processos para a identificação de causas especiais de variação (Passo 1) e é feita uma análise sobre um conjunto de fatores candidatos à causas comuns de variação (Passo 2).

### 3.1. Passo 1: Identificando as causas especiais de variação

O primeiro passo em busca dos fatores que afetam a Produtividade foi a realização da análise de estabilidade do processo de desenvolvimento com relação à Produtividade. Inicialmente a intenção era identificar *outliers*, isto é, projetos que pudessem ter sofrido a incidência de causas especiais de variação e que, por esta razão, poderiam dificultar a análise. A partir dos dados obtidos dos dezesseis projetos, foi construído de um gráfico de controle para avaliar a estabilidade da Produtividade. Para avaliar a estabilidade do processo foi utilizado o conjunto de nove testes de estabilidade disponível do software Minitab, com a aplicação dos parâmetros padrão, conforme ilustrado na Figura 3.

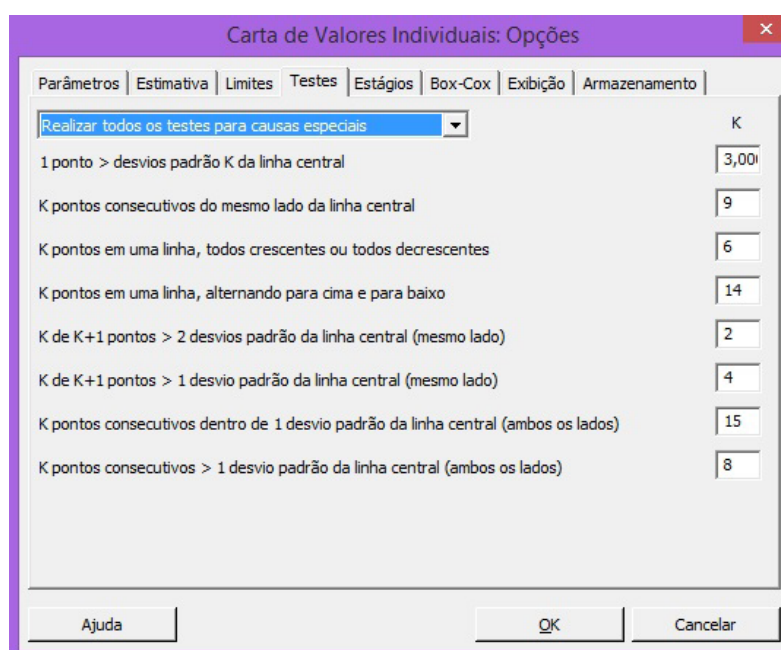


Figura 3. Testes de estabilidade oferecidos na ferramenta Minitab com os respectivos parâmetros.

Em primeira análise, o teste 5 de estabilidade indicou problemas no projeto 13. O teste indicou a existência de dois projetos (11 e 13) há mais de 2 desvios padrão da linha central e do mesmo lado. Os limites inferior e superior computados para o conjunto de projetos foi de 0,44 HH/PF e 8,57 HH/PF, respectivamente (vide Figura 4). Ao analisar o projeto 13 para identificar a causa de tamanha variação da produtividade, foi observado um investimento de esforço adicional para implementação de testes de aceitação automáticos na plataforma Android. Isto implicou no aumento da razão de esforço para cada ponto por função do projeto.

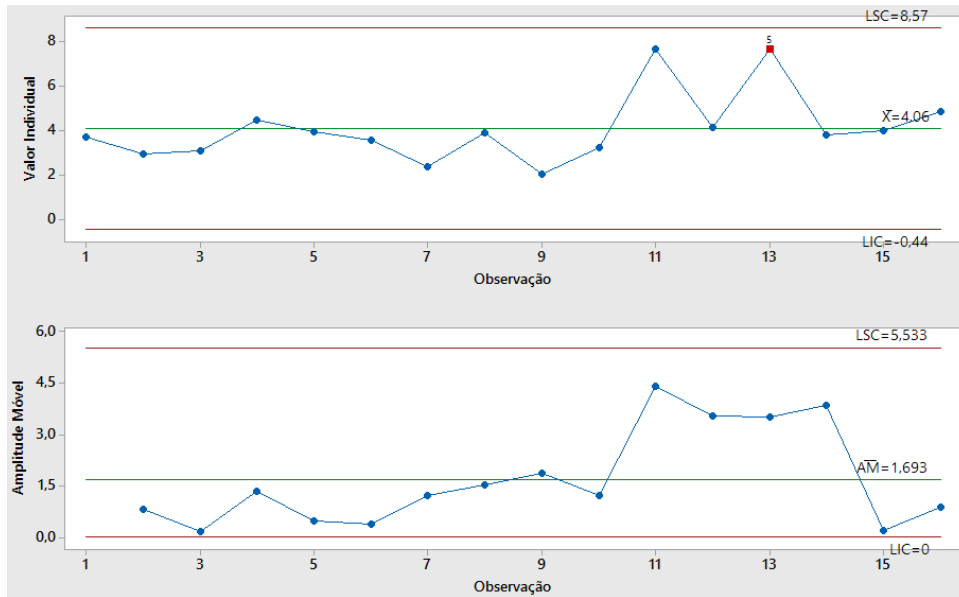


Figura 4. Gráfico de controle para valores individuais e para amplitudes móveis com testes de estabilidade

O projeto 13 foi excluído da amostra de projetos e foi plotado um novo gráfico de controle (Figura 5). O novo gráfico ilustra a violação do teste 1, em função do projeto 11 ter ficado distante de três desvios padrão a partir da linha central. Ao analisar o projeto 11, constatou-se a existência de um requisito não-funcional que foi responsável por aproximadamente 50% do esforço adicional investido no projeto. Vale observar que após a exclusão do projeto 13, os limites inferiores e superiores foram redefinidos.

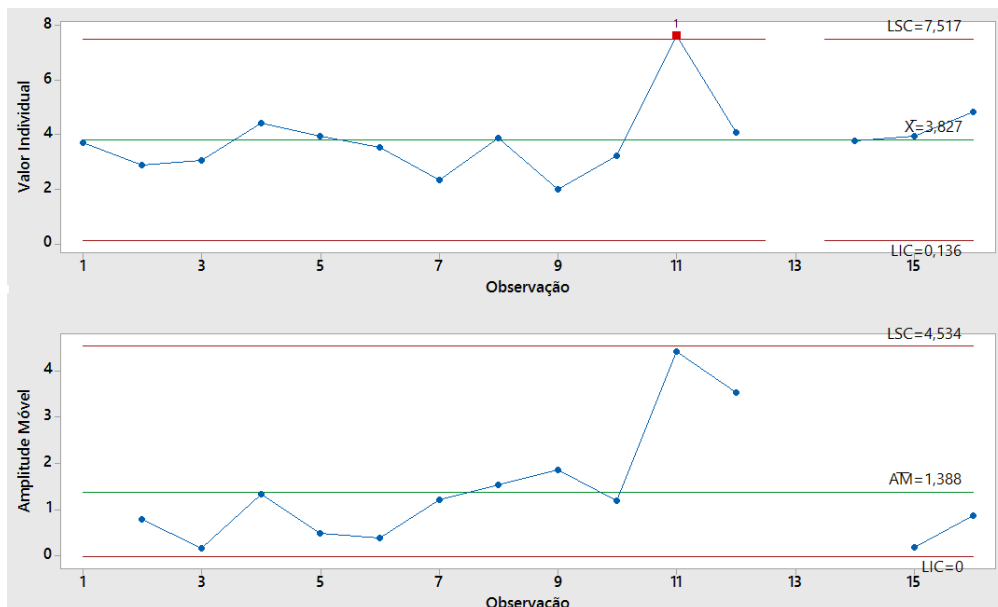


Figura 5. Gráfico de controle para valores individuais e amplitudes móveis com a exclusão do projeto 13.

Outra forma de observar que os projetos 11 e 13 provavelmente não pertencem ao grupo seria por meio de um histograma, conforme ilustrado no grupo 8 da Figura 6.

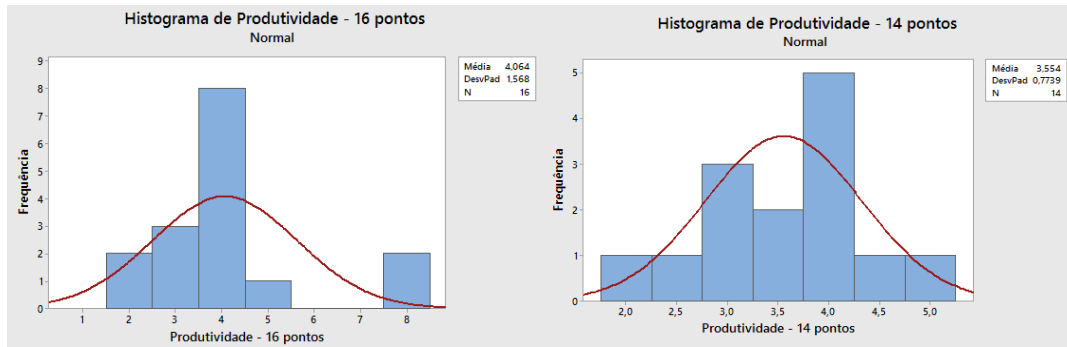


Figura 6. Histograma com os 16 projetos da base histórica (esquerda) e com os dois projetos excluídos (direita)

O projeto 11 também foi excluído da amostra e um novo gráfico de controle foi gerado com apenas 14 projetos e não foram observadas falhas dos testes de estabilidade com relação à produtividade (Figura 7). O novo gráfico indica média de 3,554 HH/PF e limites inferior e superior de 1,118 e 5,989 HH/PF, respectivamente, para a variação (voz do processo). É importante considerar que a aplicação deste procedimento depende da existência de um número significativo de projetos.

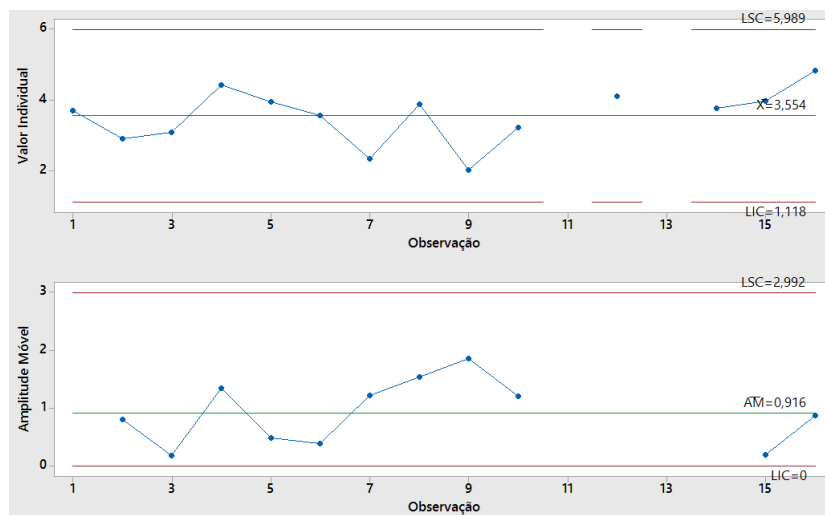


Figura 7. Processo estável após a exclusão dos projetos 11 e 13

### 3.2. Passo 2: Identificando fatores que podem atuar como causas comuns

Com a estabilidade da produtividade para este conjunto de projetos, foi iniciada a busca dos fatores que podem exercer influência na produtividade. Foram enumerados dezessete fatores potenciais baseados nos fatores identificados por Wagner e Ruhe (2008) e em outros fatores identificados pelos líderes de projeto na unidade fabril. Em seguida foi feita uma análise do coeficiente de determinação para verificar qual o nível de influência dos fatores na produtividade dos projetos.

A partir dos valores obtidos na matriz foi elaborado um gráfico de pareto para observar mais claramente quais os fatores mais influenciam na produtividade dos projetos. Dentre estes fatores destacamos a produtividade na arquitetura e o componente de tamanho não funcional que, de acordo com o coeficiente de determinação, explicam aproximadamente 36% e 33% da variação da produtividade.

**Tabela 1. Correlação e coeficiente de determinação entre a produtividade e os Fatores Candidatos**

Fatores Candidatos	Correlação	Coeficiente de Determinação (R2)
1. Plataforma	-0,0246603	0,000608130
2. Tecnologia	0,2017405	0,040699247
3. Tipo Cliente	0,2391585	0,057196793
4. Tipo Serviço	-0,1262498	0,015939016
5. Tamanho Funcional	-0,2442302	0,059648395
6. Testes automáticos	-0,1172448	0,013746342
7. Projeto Novo/Manutenção	0,1727249	0,029833884
8. Prazo Projeto	0,1329754	0,017682444
9. Local Físico (ST/1A)	0,3138628	0,098509888
10. Período de Início	-0,4288604	0,183921239
11. Produtividade na Codificação	0,2887564	0,083380270
12. Produtividade na análise	0,4374972	0,191403761
13. Produtividade na Arquitetura	0,5982483	0,357900986
14. Componente Não-funcional	0,5793906	0,335693477
15. Densidade Bugs Homologação	0,5207029	0,271131555
16. Densidade de Bugs	0,5253219	0,275963071
17. Mudanças	0,0225195	0,000507127

Um dos aspectos que chamaram atenção foi o fato da produtividade da codificação exercer menos influência do que a produtividade na análise e a produtividade na arquitetura, uma vez que a codificação responde a 41% e a atividade de arquitetura corresponde a aproximadamente 14%. Outra questão é que fatores que poderiam indicar a existência de subgrupos de projetos, a exemplo da plataforma, tecnologia, tipo de serviço e projeto novo/manutenção, apresentaram baixa correlação.

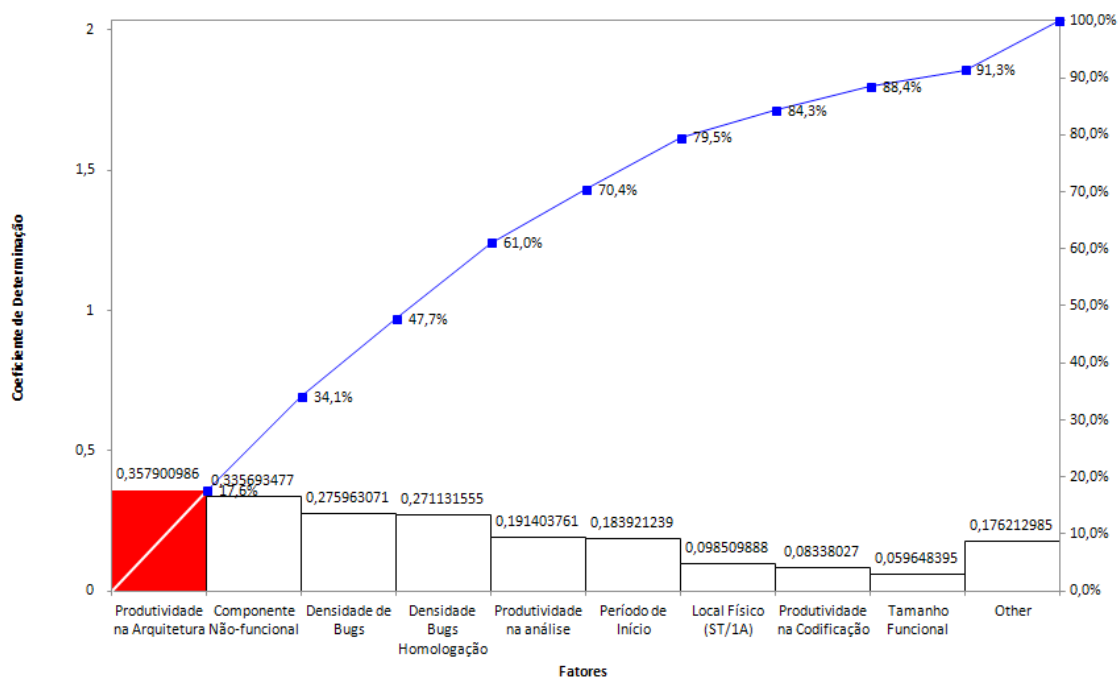


Figura 8. Gráfico de pareto para seleção dos fatores mais influentes

Os passos 1 e 2 serão repetidos para os fatores identificados no primeiro ciclo de análise. Foi observado que a produtividade da arquitetura para os quatorze projetos não estava estável. Dentre as causas identificadas destacamos a curva de aprendizagem do arquiteto e o aumento da complexidade relativa a aspectos não funcionais do projeto. Ao tratar estes fatores espera-se diminuir o grau influência da produtividade da arquitetura na produtividade geral do projeto, o que implicará na definição de novos limites de controle e, possivelmente, no aumento da precisão das estimativas de esforço, que é derivada da produtividade.

#### 4. Considerações finais

Considerando o contexto da análise realizada, foram identificados dois fatores que atuaram como causas especiais de variação dos projetos: a incidência de um requisito não funcional de alta complexidade e a inclusão dos testes automáticos de aceitação para a plataforma Android. A identificação destes dois fatores foi guiada pela análise da estabilidade do processo sob a perspectiva da produtividade. Em um segundo passo foi identificado um conjunto de fatores que podem influenciar na variação comum da produtividade nos projetos. Estes fatores não seriam identificados, caso as causas especiais de variação não tivessem sido identificadas e os projetos afetados não tivessem sido excluídos da análise.

Um aspecto relevante a ser considerado nesta experiência é que outros fatores, que não foram considerados na lista de fatores candidatos, podem atuar como causas comuns de variação na produtividade. Acredita-se que ao aprofundar a compreensão sobre o comportamento dos processos novos fatores de influencia poderão emergir.

Concluindo, destacamos a opinião de Florac e Carleton (1999) quanto ao benefício da aplicação do controle estatístico de processos mesmo quando a organização está em níveis iniciais de maturidade com relação a modelos de qualidade. Este trabalho reitera o ponto de vista dos autores.

## Referências

- Florac, W. A., Carleton, A. D. Measuring the software process: statistical process control for software process improvement, Massachusetts: Addison-Wesley, 1999.
- Rocha, A. R., Souza, G. S., Barcellos, M. P., Medição de software e controle estatístico de processos, Brasília: Ministério da Ciência Tecnologia e Inovação, 2012
- Montoni, M., Kalinowski, M., Lupo, P., Abrantes, J. F., Ferreira, A. I. F., Rocha, A. R. Uma metodologia para desenvolvimento de modelos de desempenho de processos para gerência quantitativa de projetos de software, VI Simpósio Brasileiro de Qualidade de Software (SBQS 2007), pp. 325-339, 2007.
- SEI, Capability Maturity Model Integration (CMMI) for Development, version 1.3, Carnegie Mellon University, Software Engineering Institute - SEI, Technical Report, CMU/SEI-1010-TR-033, 2010.
- SOFTEX, Guia Geral MPS de Software, Associação para Promoção da Excelência do Software Brasileiro – SOFTEX, 2012
- Wagner, S., Ruhe, M., A systematic review of productivity factors in software development, Proceedings of 2nd International Workshop on Software Productivity Analysis and Cost Estimation (SPACE 2008), 2008.



# Una Experiencia de Implementación y Evaluación Conjunta CMMI-DEV y CMMI-SVC Nivel 5 con MPS-SW Nivel A en Sofrecom Argentina

Valeria Chiuki<sup>1</sup>, Viviana Rubinstein<sup>2</sup>, Jorge Boria<sup>2</sup>, Andres Rubinstein<sup>2</sup>, Andrea Baglietto<sup>3</sup>, Silvia Andino<sup>4</sup>, Ana Regina Rocha<sup>5</sup>

<sup>1</sup> Sofrecom Argentina – Reconquista 609 4to. Piso 1003 CABA – Argentina

<sup>2</sup> Liveware Inc – 6609 Valleyside Rd. Austin TX 78731 – USA

<sup>3</sup> Sofrecom Francia – 24 Avenue du Petit Parc – 94300 Vincennes – France

<sup>4</sup> ESCAMPI S.A – Av. Federico Lacroze 3194 – 1426 CABA – Argentina

<sup>5</sup> COPPE/UFRJ – Universidade Federal do Rio de Janeiro  
Caixa Postal 68511 – CEP 21.941-972 – Rio de Janeiro, RJ – Brasil

vchiuki@sofrecom.com.ar,  
{viviana.rubinstein,jorge.boria,andres.rubinstein}@liveware.com,  
andrea.baglietto@sofrecom.com, silvia.andino@escampi.com,  
darocha@centroin.com.br, darocha@cos.ufrj.br

**Abstract.** *Process improvement with multiple models is today, for business reasons, a reality in many companies. In Sofrecom Argentina the Quality Management system, initially based on ISO/IEC 9001, joined over time the CMMI-DEV, CMMI-SVC and MR-MPS-SW models. This paper describes the experience of the implementation and evaluation of these models in Sofrecom Argentina.*

**Resumen.** *La mejora de procesos con múltiples modelos es hoy, por razones de negocio, una realidad en muchas empresas. En Sofrecom Argentina el sistema de Gestión de Calidad, basado inicialmente en ISO/IEC 9001, incorporó a lo largo del tiempo los modelos CMMI-DEV, CMMI-SVC y MR-MPS-SW. Este artículo relata la experiencia de la implementación y evaluación de estos modelos en Sofrecom Argentina.*

## 1. Introducción

Actualmente muchas empresas, por razones de negocio, necesitan llevar su programa de mejoras de procesos con más de un modelo de referencia.

Asimismo los diferentes modelos van siendo incorporados en las empresas a lo largo del tiempo; este es el caso de Sofrecom Argentina, descrita en este artículo. El programa de mejoras de proceso comenzó con la implementación del sistema de Gestión de la Calidad basado en ISO/IEC 9001:1994 [ISO/IEC 1994]. Durante los años siguientes se agregó CMMI-DEV [SEI 2010a], lo que resultó en evaluaciones exitosas de los niveles 3 (2008) y 5 (2011). En el año 2014 fueron agregados dos nuevos modelos, CMMI-SVC [SEI 2010b] y MR-MPS-SW [SOFTEX 2012].



Este artículo presenta la experiencia actual donde los cuatro modelos conviven de manera armónica en la empresa. En la sección 2 se presenta a Sofrecom Argentina y su historia de mejora de procesos con múltiples modelos. En la sección 3 se describe la implementación conjunta de los modelos CMMI-DEV, CMMI-SVC y MR-MPS-SW. En la sección 4 se presentan las lecciones aprendidas y mejores prácticas que resultaron de la experiencia de implementación y evaluación conjunta CMMI-DEV, CMMI-SVC y MR-MPS-SW. Finalmente en la sección 5 son presentadas las conclusiones del trabajo.

## 2. Sofrecom Argentina

Sofrecom Argentina, empresa del Grupo Orange (uno de los 10 operadores de telecomunicaciones más importantes del mundo), tiene más de 22 años en el mercado local y regional, más de 150 clientes y 700 especialistas. Tiene una posición única en el mercado mundial con una gama de servicios para satisfacer las necesidades de los clientes:

- Soluciones IT: Proveer un sistema de información abierto y optimizado capaz de apoyar los cambios en sus actividades
- Business Consulting: Diseñar e implementar decisiones estratégicas para el desarrollo de sus negocios y optimizar su rendimiento
- Redes y Servicios: Tener infraestructuras fiables, abiertas y suficientemente configuradas disponibles para crear valor y controlar los costos

Sofrecom Argentina, apuesta a lograr la Calidad en todos los desarrollos y procesos en los cuales participa. Prueba de ello es el proceso de mejora continua que lleva adelante, a partir de la primera certificación obtenida en todos sus procesos.

Certificó por primera vez en el año 1999 la Norma ISO/IEC 9001:1994, en el año 2002 la norma ISO/IEC 9001:2000 [ISO/IEC 2000] y en el año 2008 la norma ISO/IEC 9001:2008 [ISO/IEC 2008] y CMMI-DEV Nivel 3 de madurez.

Durante el año 2010 Sofrecom Argentina se había planteado como objetivos estratégicos la mejora de la calidad de sus entregables y la reducción de tiempos de desarrollo para sus productos.

De esta manera se comenzó el análisis estadístico de los datos de estimación de las diferentes tareas de un proyecto. El siguiente gráfico (Figura 1) muestra el análisis de los valores estimados (Evaltam) y los valores reales de todas las estimaciones realizadas desde el año 2008 al 2011. Aquí se puede ver claramente que los valores de estimación presentes a ese momento estaban deteriorados y que era necesario realizar una línea base ya que los tiempos reales eran superiores a los estimados, generando así una pérdida a la Empresa.

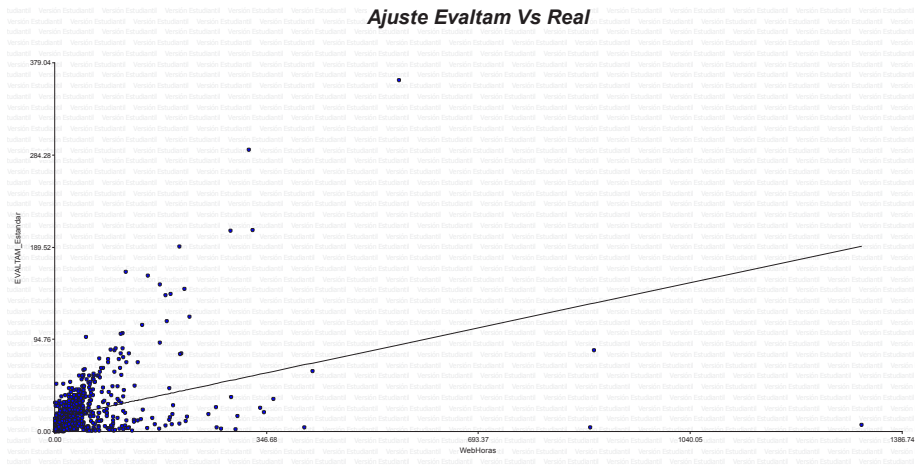


Figura 1 Ajuste Evaltam Vs Real

Los datos se segmentaron por complejidad de los requerimientos y por tipo de tarea y se quitaron outliers que distorsionaban la información. El proceso de selección de outliers fue realizado analizando la información existente junto con los Gerentes de Proyecto y la información documentada de los diferentes proyectos. El gráfico de cajas (Figura 2) muestra el resultado del análisis de cada tipo de tarea para cada tipo de complejidad de las horas reales insumidas.

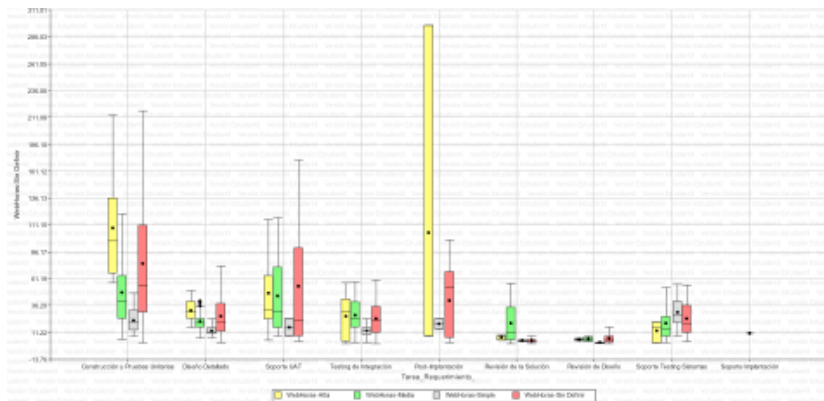


Figura 2 Tipo de Tarea por Tipo de Complejidad

A continuación, la idea fue expresar los coeficientes para la línea base por tarea y complejidad a partir de un intervalo de confianza para la media de cada uno (un intervalo de confianza es un conjunto de valores posibles en donde el valor estimado es el más probable y en la medida que nos alejamos de él la probabilidad de ocurrencia disminuye). Para esto necesitamos utilizar una distribución de probabilidad como referencia, y si se observan los gráficos de cajas no es exactamente la forma de las distribuciones. Entonces llevamos las distribuciones a una log normal (Figura 3).

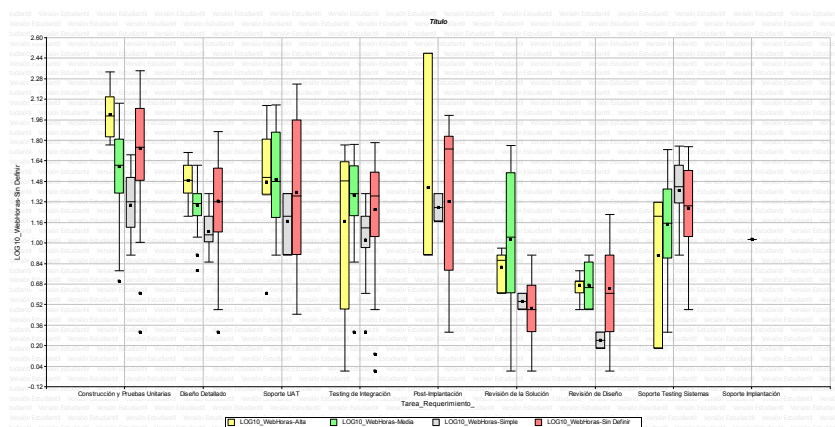


Figura 3 Tipo de Tarea por Tipo de Complejidad (log normal)

Con esto mejoramos significativamente los coeficientes de variación, y estuvimos en condiciones de utilizar estos datos reales para establecer la línea base para futuras estimaciones. Esta información fue trasladada a nuestra herramienta de estimaciones que utilizan los Gerentes de Proyecto.

Haciendo la validación de la nueva herramienta de estimación se observa que para mediciones “REALES”, en contraste con el nuevo evaltam se encuentra una diferencia del 1% mientras que contra valores estándar la herramienta anteriores un 35%. Teniendo en cuenta los sueldos promedio de cada perfil que desarrolla las tareas y los precios de venta, mostramos a continuación el ROI de esta mejora en el cambio de herramienta de estimación.

Tabla 1 Costo compañía promedio por hora

ROL	Junior	No Aplica	Semi Senior	Senior	Trainee	Promedio
DESARROLLADOR	\$ 43		\$ 62	\$ 76	\$ 30	\$ 53
LÍDER DE PROYECTO		\$ 100		\$ 77		\$ 89
TESTER	\$ 44		\$ 56	\$ 54		\$ 51
	\$ 43	\$ 100	\$ 59	\$ 69	\$ 30	\$ 64

Tabla 2 Precio de venta promedio por hora

ROL	Junior	No Aplica	Semi Senior	Senior	Trainee	Grand Total
DESARROLLADOR	\$ 57		\$ 83	\$ 101	\$ 40	\$ 70
LÍDER DE PROYECTO		\$ 133		\$ 103		\$ 118
TESTER	\$ 58		\$ 74	\$ 72		\$ 68
	\$ 58	\$ 133	\$ 78	\$ 92	\$ 40	\$ 85

Luego tomando un requerimiento de ejemplo de 1897 hs. reales de ejecución, y aplicando los parámetros anteriores se llega a la conclusión que, de haberse utilizado la línea base estándar original, el costo por subestimación habría sido de \$65,567 (considerando las siguientes etapas y de acuerdo a una distribución por rol/tarea):

**Tabla 3 Costo Requerimiento (ejemplo)**

	Análisis	Diseño Detallado	Construcción y Pruebas Unitarias	TOTAL
DESARROLLADOR	\$ 2,437.29	\$ 1,200.01	\$ 11,617.55	\$ 15,254.84
LIDER DE PROYECTO	\$ 8,216.63	\$ 2,421.93	\$ 13,395.43	\$ 24,033.98
TESTER		\$ 1,324.45	\$ 24,954.66	\$ 26,279.11
TOTAL	\$ 10,653.92	\$ 4,946.39	\$ 49,967.63	\$ 65,567.94

Con estos modelos , Sofrecom Argentina pudo recorrer la primera experiencia de gestión de alta madurez y a finales del año 2011 fue evaluada satisfactoriamente con el nivel de madurez 5 de CMMI-DEV, convirtiéndose así en una de las tres empresas argentinas con ese galardón y la certificación ISO/IEC 9001.

En 2014 incorporó dos nuevos modelos CMMI-SVC y MR-MPS-SW al Sistema de Gestión de Calidad.

### 3. Implementación de los Modelos CMMI-DEV, CMMI-SVC y MR-MPS-SW

Sofrecom Argentina posee actualmente con un Sistema de Gestión de Calidad basado en su proceso de negocio donde conviven 4 modelos de calidad (ISO/IEC 9001, CMMI-DEV, CMMI-SVC y MR-MPS-SW) de manera transparente para todos los colaboradores.

Cuenta con metodologías para la gestión de proyectos y servicios y la ingeniería de software, que conforman un mapa de la calidad del producto, actuando como herramienta para el trabajo diario del responsable operativo del proyecto asignado por la compañía. El área de Aseguramiento de Calidad es la responsable de garantizar que las exigencias de cada modelo se vean incluidas naturalmente dentro del proceso.

Habiendo logrado e institucionalizado un Sistema de Gestión de Calidad que responde tanto al modelo ISO/IEC 9001 como al modelo CMMI-DEV, la incorporación de la gestión cuantitativa de los proyectos permitió mejorar las estimaciones de tiempos ya que es muy necesario no sobre estimar para no perder competitividad o subestimar para no tener impacto económico en el resultado de la compañía.

Sofrecom utiliza modelos estadísticos y de control de procesos como parte de su modelo de negocio. Gracias a esto se puede diferenciar sus productos y servicios y hacer sus propuestas basadas en datos reales y apoyados en la realidad. El Plan de Negocios establece objetivos y metas de crecimiento. Estos se traducen en metas de productividad y calidad en un Plan de Medidas.

Al día de hoy, como lo hemos explicado anteriormente, los dos modelos que se utilizan en la estimación son: EVALTAM y PFRUCP. Siguen la misma inspiración y son muy similares en su naturaleza, pero responden a diferentes procesos utilizados en la prestación de los servicios. Ambos tienen como punto de partida los objetivos cuantitativos de la organización para la calidad y el rendimiento del proceso, que son trazables a los objetivos de negocio. Cada uno es una lista de los pasos que se

pueden ejecutar en la prestación de un servicio, mostrando subprocesos seleccionados incluidos en el análisis de rendimiento de los procesos de la organización; las tablas mantienen la trazabilidad de los objetivos de negocio.

Con base en los datos históricos, que se actualizan frecuentemente, se estima un límite estadístico superior e inferior, así como un valor medio. Este valor se revisa, siguiendo los grandes pasos mostrados en el punto anterior, para correcciones que podrían ser introducidas basadas en factores conocidos que causarán desviaciones, tales como un requerimiento que es de naturaleza similar a uno que fue ejecutado con éxito por el mismo equipo o uno muy similar, lo que sugeriría una media y una desviación estándar más pequeñas, o un requerimiento de un cliente históricamente difícil, con peticiones de cambio constantes en el medio del proyecto, lo que sugiere un mayor riesgo y algunos intervalos, o incluso valores medios más grandes. Usando estos ajustes es posible proporcionar al cliente una propuesta que cumpla con sus requisitos y siga los procesos establecidos de Sofrecom. Esto permite un control constante de la variación de las variables durante la ejecución del proyecto y el ajuste de las líneas de base, según sea necesario. Los gerentes utilizan los modelos durante todo el proyecto para controlar y ajustar, así como identificar los posibles cambios que pueden resultar en mejoras.

Otro valor agregado de haber incorporado áreas de proceso de alta madurez en el ciclo del negocio, estuvo dado por la posibilidad de predecir, en etapas tempranas de la construcción, la cantidad de errores a descubrir en las fases posteriores a la entrega del producto basado en los tiempos estimados en cada etapa del ciclo de desarrollo de software. Esto es de una importancia clave ya que impacta en la continuidad de la sustentabilidad de la compañía. Un producto entregado con baja calidad genera insatisfacción del cliente y como consecuencia una alta probabilidad de cancelación y/o de no obtención de nuevas oportunidades de negocio con dicho cliente.

A fines del año 2012 comenzó un nuevo proyecto de mejora de procesos para incorporar prácticas de los modelos MR-MPS-SW y CMMI-SVC.

Sofrecom Argentina inició el proyecto analizando las similitudes y diferencias de los modelos de desarrollos (CMMI-DEV y MR-MPS-SW) para detectar las necesidades de mejora en los puntos que son comunes y no comunes donde el MR-MPS-SW posee exigencias más específicas no contempladas explícitamente en el modelo CMMI-DEV.

Las diferencias más importantes estuvieron dadas por los procesos de Gerencia de Portafolio de Proyectos, Gestión de Reutilización y Gerencia de Recursos Humanos que son inexistentes en el modelo de CMMI-DEV. Dado que Sofrecom Argentina es una filial de Sofrecom Francia varias de las exigencias del modelo estaban tenidas en cuenta en nuestro proceso de reporte a nuestra Casa Matriz pero mejoramos la selección de los proyectos por medio de la utilización de una herramienta de evaluación de proyectos en la etapa de preventa donde se tienen en cuenta los diferentes criterios definidos.

Con respecto a la Gestión de Reutilización se introdujeron mejoras en la definición de la metodología para clasificar, agregar, revisar y dar de baja componentes de software reutilizables para la realización de los productos finales a entregar a nuestros clientes. Por último, el proceso que menos modificaciones tuvo por la incorporación del MPS-SW fue el de Recursos Humanos ya que ya contaba con el gerenciamiento de las competencias de nuestros recursos, la evaluación de desempeño en función a dichas competencias ya que las personas y sus conocimientos es el principal activo de Sofrecom Argentina.

El paso siguiente fue definir lo que era considerado servicio en Sofrecom Argentina. Por sus características, gran parte de lo que hace son servicios de desarrollo de software.

Se incluyeron, así, las prácticas específicas de CMMI-SVC a partir de la formalización de las actividades que se llevan a cabo de manera cotidiana dado que Sofrecom Argentina gestiona servicios desde hace más de dos décadas. Se interpretó que el modelo de CMMI-SVC brinda un muy buen marco para la mejora en la gestión de los contratos de servicio de desarrollo de software. Y el MR-MPS-SW agrega buenas prácticas para aplicar a cada pedido de mantenimiento evolutivo del servicio que no existía en la implementación del CMMI-DEV como reutilización, gestión de portafolio y gestión de conocimientos.

Los principales desafíos del proyecto de mejora de procesos y luego dejar implementado 4 modelos de calidad fueron:

- Enriquecer los procesos y activos del Sistema de Gestión de Calidad para que cubran los cuatro modelos minimizando el impacto en la operatoria de las prestaciones y evitando introducir grandes cantidades de documentación sin un objetivo específico.
- La adaptación cultural del equipo de Calidad, para tener la capacidad de entrar por uno u otro modelo y realizar el mapping con el resto para asegurar la cobertura de cada uno de ellos con un nuevo proceso, procedimiento, formulario, herramienta, ,etc. Es decir, institucionalizar en el equipo Calidad una cultura "Calidad Multidimensional"

#### **4. Lecciones Aprendidas y Mejores Prácticas**

Una implementación y evaluación exitosa trae consigo un gran número de lecciones aprendidas y mejores prácticas. En esa sección reunimos un conjunto de ellas, recogidas del equipo responsable por la implementación y del equipo de evaluación.

La experiencia de la implementación nos permite apuntar:

- La utilización de un modelo como el MR-MPS-SW nos ayudó a completar un marco de procesos guiado en la reutilización de nuestros activos.
- El trabajo realizado por un grupo inter disciplinario permite tener diferentes visiones y objetivos de implementación de cada buena práctica.
- Tener las herramientas informáticas adecuadas para implementar los procesos ayuda en gran medida a la institucionalización efectiva de los mismos.
- El uso de los modelos ISO/IEC 9001 y CMMI-SVC principalmente para la gestión de los contratos de servicios y CMMI-DEV y MR-MPS-SW para los pedidos de los servicios es la manera natural en Sofrecom Argentina.
- La incorporación de prácticas formales de reutilización de activos y gestión del conocimiento, nuestro capital más importante, enriqueció las competencias de la empresa.
- La gestión de nuestro portafolio de proyectos se vio favorecida por la inclusión de nuevas buenas prácticas basadas en criterios objetivos.

- Las capacitaciones en procesos, realizadas con cada mejora del Sistema de Gestión de Calidad, apuntan a la institucionalización de las prácticas y a que todo el personal mantenga una forma de trabajo homogénea haciendo de Sofrecom Argentina una empresa con ADN propio.
- De esta evaluación podemos destacar los siguientes puntos:
- Un factor crítico para el éxito de la evaluación fue contar con un equipo con mucha experiencia en evaluaciones CMMI, MPS e incluso en evaluaciones conjuntas CMMI-MPS.
- Un aspecto, también, importante fue disponer de un tiempo un poco más largo que el usual en evaluaciones MPS que nos ha permitido trabajar con calma y discutir las semejanzas y diferencias entre los tres modelos.
- El hecho de tener presentaciones detalladas de cómo los profesionales ejecutan su trabajo, permitió al equipo una comprensión muy completa de la implementación de los modelos.
- Contar con una planilla de apoyo a evaluaciones conjuntas (CMMI y MPS) con doble entrada ha dado agilidad al trabajo del equipo de evaluación.
- El conocimiento internacional y multicultural ha permitido enriquecer no sólo el entendimiento de los modelos sino también las diferentes maneras de su implantación dentro el contexto de la organización.
- La evaluación conjunta de varios modelos nos impuso la realización de un análisis 360 grados de las evidencias.

## 5. Conclusiones

La implementación del sistema de Gestión de la Calidad con múltiples modelos se fue configurando en Sofrecom Argentina como un camino natural y necesario por razones de negocio y para alcanzar más eficazmente los objetivos organizacionales. Esto permitirá a Sofrecom posicionarse mejor en la región y poder abrir nuevos mercados. Estamos convencidos que el camino de la mejora continua los fortalece como compañía asegurando la satisfacción de los clientes y contribuyendo a la mayor certeza en el cumplimiento de los objetivos de negocio.

## Agradecimientos

Los autores agradecen a Adrian Hettema (CEO Sofrecom Argentina), a todos los que contribuyeron en este camino de mejora y a Softex por facilitar esta evaluación conjunta en un ámbito ameno y de gran valor agregado.

## Referencias

- ISO/IEC (1994), ISO/IEC 9001:1994 – *Quality systems - Model for quality assurance*, The International Organization for Standardization and The International Electrotechnical Commission.
- ISO/IEC (2000), ISO/IEC 9001:2000 – *Quality management systems — Requirements*, The International Organization for Standardization and The International Electrotechnical Commission.
- ISO/IEC (2008), ISO/IEC 9001:2008 – *Quality management systems — Requirements*, The International Organization for Standardization and The International Electrotechnical Commission.
- SEI (2010a), *Capability Maturity Model Integration (CMMI) for Development, version 1.3*, Carnegie Mellon University, Software Engineering Institute, Technical Report CMU/SEI-2010-TR-033.
- SEI (2010b), *Capability Maturity Model Integration (CMMI) for Services, version 1.3*, Carnegie Mellon University, Software Engineering Institute, Technical Report CMU/SEI-2010-TR-034.
- SOFTEX (2012), *ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO – SOFTEX., MPS.BR – Guia Geral MPS de Software:2012*, agosto 2012. Disponible en: <http://www.softex.br>.





# Implantação do MR-MPS-SV na Instituição Avaliadora COPPE/UFRJ

Ana Regina Rocha<sup>1,2</sup>, Gleison Santos<sup>2,3</sup>, Cristina Cerdeiral<sup>1,2</sup>, Marcelo Schots<sup>1,2</sup>, Natalia Schots<sup>1,2</sup>, Taísa Guidini Gonçalves<sup>1</sup>, Elaine Nunes<sup>2</sup>, Tayana Conte<sup>2,4</sup>, Reinaldo Cabral Silva Filho<sup>2,5</sup>, Mylene Cabral<sup>2</sup>

<sup>1</sup> Programa de Engenharia de Sistemas e Computação – COPPE/UFRJ

<sup>2</sup> Instituição Avaliadora COPPE/UFRJ

Caixa Postal 68.511 – CEP 21.945-970 – Rio de Janeiro – RJ – Brasil

<sup>3</sup> Programa de Pós-Graduação em Informática – UNIRIO

Av. Pasteur 458, Urca – CEP 22290-240 – Rio de Janeiro – RJ – Brasil

<sup>4</sup> Instituto de Computação – Universidade Federal do Amazonas (UFAM)

Manaus – AM – Brasil

<sup>5</sup> Núcleo de Tecnologia da Informação – Universidade Federal de Alagoas (UFAL)

Maceió – AL – Brasil

{darocha, cerdeiral, taisa}@cos.ufrj.br,

gleison.santos@uniriotec.br, tayana@icomp.ufam.edu.br, reinaldo@nti.ufal.br

**Abstract.** *The MPS for Services Reference Model (MR-MPS-SV), created in 2012, is the most recent component of the MPS.BR Program. As part of the MR-MPS-SV deployment strategy, Softex coordinated the conduction of pilot assessments. In April 2013, COPPE/UFRJ Assessment Institution was assessed MR-MPS-SV maturity level G. This paper presents the main implementation details of this IT service process improvement initiative and the main lessons learned.*

**Resumo.** *O mais recente componente do Programa MPS.BR é o Modelo de Referência MPS para Serviços (MR-MPS-SV), criado em 2012. Como parte da estratégia de implantação do MR-MPS-SV, a Softex coordenou a execução de avaliações piloto. Em abril de 2013, a Instituição Avaliadora COPPE/UFRJ foi avaliada no Nível G do MR-MPS-SV. Este artigo apresenta os principais detalhes da implementação dessa iniciativa de melhoria de processos de serviços de TI e as principais lições aprendidas.*

## 1. Introdução

A COPPE/UFRJ é formada por um conjunto de programas de pós-graduação, entre eles o Programa de Engenharia de Sistemas e Computação (PESC). Cada programa é formado por linhas de pesquisa e, estas, podem ser compostas por áreas. A linha de Engenharia de Software é responsável por atividades de ensino, pesquisa e consultoria. Dentre suas áreas de atuação, encontra-se a Área de Qualidade de Software, através da qual desde 1987 é realizada uma intensa atividade de interação universidade-empresa. Desta forma desde o início do Programa MPS a área de Qualidade de Software abrigou uma Instituição Implementadora (II) e também uma Instituição Avaliadora (IA) credenciada pela Softex.

Como parte da estratégia de formação de pessoal, sempre procurou-se envolver alunos e ex-alunos de mestrado e doutorado em atividades relacionadas à implantação de processos em organizações de software (parte das atividades desempenhadas pela II COPPE/UFRJ) e relacionadas à avaliação de

organizações no MR-MPS-SW [SOFTEX, 2012c] (atividades desempenhadas pela IA COPPE/UFRJ). Mais recentemente, com a criação do Modelo de Referência MPS para Serviços (MR-MPS-SV) [SOFTEX, 2012b], a preparação para uma avaliação oficial nesse modelo foi vista como uma oportunidade de aprendizado da equipe alinhada à necessidade de formalização de alguns procedimentos adotados pela IA COPPE/UFRJ.

Este artigo apresenta a experiência de implantação do MR-MPS-SV da IA COPPE/UFRJ que culminou com a avaliação no nível G desse modelo em abril de 2013 e possibilitou, entre outros, parte do aprendizado necessário para a qualificação de seus implementadores e avaliadores. Na seção 2 é apresentado o MR-MPS-SV e desafios associados à implementação do Nível G. Na Seção 3 são apresentadas mais informações sobre a Área de Qualidade de Software da COPPE/UFRJ, incluindo outra experiência de auto-implantação de processos. A Seção 4 apresenta a iniciativa de melhoria de processos que levou à avaliação no nível G do MR-MPS-SV e as lições aprendidas nesse contexto. Por fim, a Seção 5 apresenta as considerações finais deste trabalho.

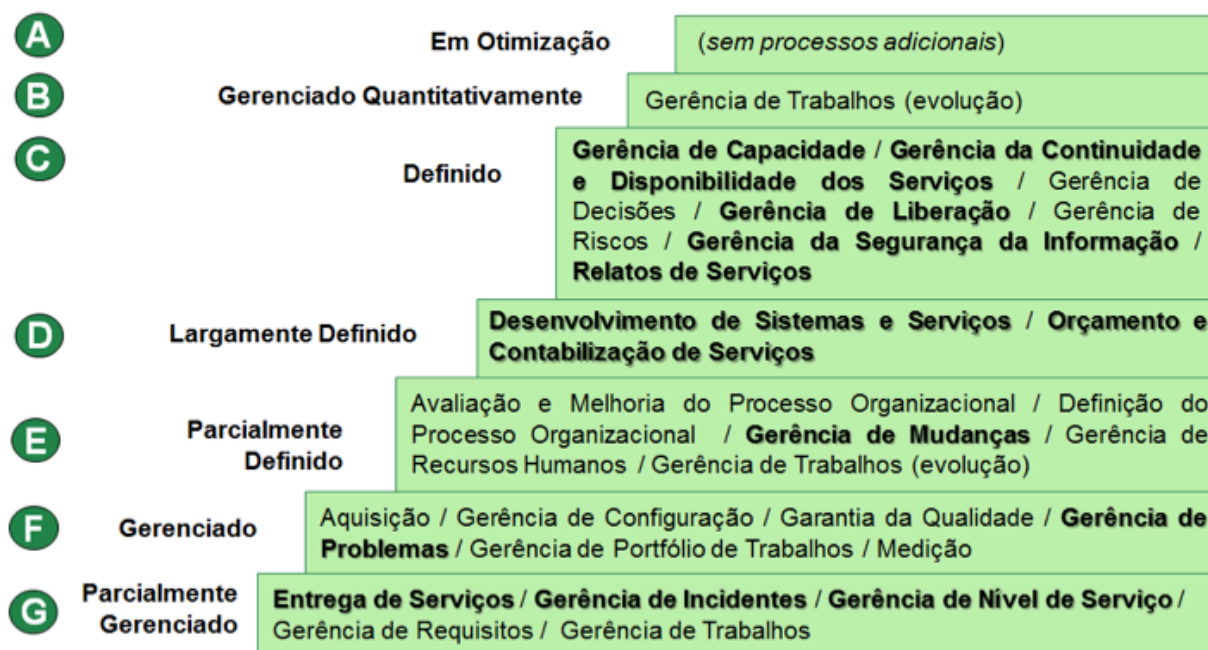
## 2. Nível G do MR-MPS-SV

Modelo de Referência MPS para Serviços (MR-MPS-SV) [SOFTEX, 2012b] foi criado em 2012. Seu objetivo é apoiar organizações prestadoras de serviços de TI a melhorarem seus processos relacionados à prestação de serviços a clientes. De forma similar ao Modelo de Referência MPS para Software (MR-MPS-SW) [SOFTEX, 2012c], o MR-MPS-SV é baseado nos conceitos de maturidade e capacidade de processo para a avaliação e melhoria da qualidade e produtividade de serviços prestados e, assim, possibilitar que atinjam seus objetivos de negócio. O MR-MPS-SV também possui 7 níveis de maturidade e compartilha com o MR-MPS-SW alguns processos, como pode ser visto em negrito na Figura 1.

Uma das diferenças dos processos comuns aos dois modelos é que para o MR-MPS-SV, usa-se o conceito *trabalho* em vez de *projeto* (como pode ser visto, por exemplo, no processo Gerência de Trabalhos em contraposição ao processo Gerência de Projetos do MR-MPS-SW). Para o MR-MPS-SV, *serviço* é “um produto que é intangível e não armazenável, é o resultado entregue por um provedor de serviços”. Ainda para o MR-MPS-SV, um *trabalho* é “um conjunto gerenciado de pessoas e outros recursos alocados que entregam um ou mais serviços para um cliente ou usuário final”.

Dessa forma, no nível G do MR-MPS-SV, os processos Gerência de Trabalhos (GTR) e Gerência de Requisitos (GRE) são similares aos seus equivalentes do MR-MPS-SW. Além disso, os processos adicionais para o nível G são [SOFTEX, 2012b]:

- Entrega de Serviços (ETS) – cujo propósito é entregar os serviços em conformidade com os acordos de serviços;
- Gerência de Incidentes (GIN) – cujo propósito é restaurar os serviços acordados e cumprir as solicitações de serviços dentro de um Acordo de Nível de Serviço (ANS);
- Gerência de Nível de Serviço (GNS) – cujo propósito é garantir que os objetivos dos acordos de nível de serviço para cada cliente sejam atendidos.



**Figura 1. Estrutura do MR-MPS-SV (Fonte: baseado em material do Curso Oficial C1-SV)**

Segundo o Guia de Implementação do MR-MPS-SV [SOFTEX, 2012a], dois pontos são desafiadores na implantação do nível G: (1) mudança de cultura organizacional, orientando a definição e melhoria dos processos de gerenciamento de serviços; e definição do conceito acerca do que é “serviço” para a organização. Além disso, pode-se destacar que, assim como a implementação de um modelo para melhoria de processos de software, a implementação de um modelo para melhoria de processos de serviços de TI também é influenciada por fatores críticos de sucesso, a exemplo da estratégia de implementação do projeto; do apoio, comprometimento e envolvimento; e da disponibilidade de recursos e processos adequados [Diirr e Santos, 2013].

### 3. Histórico de Investimento em Melhoria de Processos da Área de Qualidade de Software da COPPE/UFRJ

A COPPE/UFRJ é credenciada pela SOFTEX como Instituição Avaliadora (IA) e Instituição Implementadora (II) MPS desde 2006. Como II a equipe atuou na implementação de melhoria de processos de software em micro, pequenas, médias e grandes empresas, localizadas no Rio de Janeiro e em outros estados do país, obtendo expressivos resultados nestas implementações. Atualmente, as atividades de implementação estão em processo *spin off* para a Implementum, uma empresa nativa COPPE, criada para abrigar a II. Como Instituição Avaliadora, a COPPE foi responsável por várias avaliações do MPS. BR, contribuindo para a formação de avaliadores líderes em seu grupo e também para a formação de avaliadores de outras IAs, por meio da participação de avaliadores adjuntos em avaliações MPS lideradas pela COPPE [Santos et al., 2009a; Santos et al., 2009b].

No final de 2007, os membros da II e da IA COPPE decidiram pela implementação do nível E do MR-MPS-SW na Área de Qualidade do Laboratório de Engenharia de Software (LENS) da COPPE/UFRJ [Santos et al., 2009a]. A implementação de melhoria de processos no LENS visava à construção de um ambiente de alta maturidade a ser utilizado como laboratório para a aplicação de estratégias de

implementação de melhoria de processos de software, e para que estas estratégias fossem utilizadas em organizações cliente, aumentando a probabilidade de sucesso de iniciativas de melhoria. À época, esse desafio trouxe aos implementadores uma experiência única, pois os consultores passaram ao papel de colaboradores e foram avaliados. Além disso, muitas lições aprendidas foram obtidas a partir desta experiência, por exemplo [Santos *et al.*, 2009a]:

- Deve-se procurar gerenciar a diversidade de ideias de forma a diminuir as divergências entre a equipe e concentrar esforços em soluções não conflitantes;
- Os benefícios da iniciativa de melhoria de processos de software devem ser reconhecidos para aumentar o comprometimento com as ações realizadas;
- O trabalho em equipe e em grupos grandes é fundamental para o sucesso da iniciativa de melhoria de processos;
- A organização deve ser preparada para ouvir as críticas dos avaliadores de forma construtiva e evitar criar conflitos devido a divergências na interpretação dos resultados dos processos.

#### 4. Implantação do MR-MPS-SV na IA COPPE/UFRJ

Com o surgimento do modelo MPS-Serviços, percebemos os benefícios que o modelo poderia trazer para a Instituição Avaliadora em relação a uma formalização ainda maior de seus processos. Além disso, um dos fatores determinantes foi o fato de a equipe estar convencida que aprender fazendo é a melhor forma de aprender, como já havia comprovado na implementação do MR-MPS-SW [Santos *et al.*, 2009a; Santos *et al.*, 2009b].

Dentre as características da COPPE/UFRJ, em relação às atividades de Instituição Avaliadora, as relevantes para essa implementação foram: a equipe envolvida ser grande e estar geograficamente dispersa (entre Rio de Janeiro, Maceió e Manaus); a realização de muitas avaliações MPS; e a necessidade de formar a equipe no MR-MPS-SV.

Um dos itens iniciais da implementação do MR-MPS-SV foi definir o que seria considerado um serviço e o que seria considerado um trabalho na IA COPPE/UFRJ. Foi definido que o serviço prestado seria a realização de avaliações MPS com dois tipos de trabalhos (avaliações MPS de qualquer nível e avaliações conjuntas MPS/SCAMPI). Cada contrato concreto daria lugar a trabalhos específicos. No escopo da avaliação foram incluídos dois trabalhos, um trabalho composto por duas avaliações MPS e um trabalho referente a uma avaliação conjunta MPS/SCAMPI.

A implementação e preparação para avaliação da IA COPPE/UFRJ durou 4 meses, entre novembro de 2013 e março de 2014. As avaliações inicial e final foram executadas em abril de 2014. As principais etapas executadas foram:

- Reunião da equipe para discutir o modelo MR-MPS-SV;
- Definição do processo e de templates de documentos;
- Reunião para discussão do processo e documentação definidos;
- Uso inicial dos processos e documentos em dois trabalhos (conforme descrito anteriormente);
- Reunião para avaliação do processo e melhoria;
- Avaliação no Nível G do MR-MPS-SV.

#### 4.1. Principais Documentos e Definições

Além das definições de serviço e trabalho, alguns outros elementos precisaram ser definidos para viabilizar a avaliação no MR-MPS-SV, por exemplo, a definição do processo para a execução dos trabalhos, a definição do que seria considerado um incidente e do que seria considerado um acordo de nível de serviço (ANS). Além disso, foi necessário definir como gerenciar os trabalhos e os requisitos associados à prestação do serviço de avaliação MPS. A seguir são discutidas questões relevantes a cada um dos processos do Nível G do MR-MPS-SV e da implementação dos atributos de processo.

- *Processo Gerência de Trabalhos:* Para implementação da gerência de trabalhos foram definidos dois documentos: (i) Plano de Trabalho para Avaliações MPS e (ii) Plano de Trabalho para Avaliações Conjuntas MPS/CMMI. Estes Planos de Trabalho são instanciados para cada trabalho específico. A monitoração do trabalho foi definida para ocorrer ao final de cada fase do ciclo de vida definido para os trabalhos: (i) Marco 1 (Aprovação da Proposta); Marco 2 (Conclusão da Avaliação Inicial, evidenciado pela finalização nos ajustes solicitados pela auditoria); (iii) Marco 3 (Conclusão da Avaliação Final, evidenciado pelo envio do Relatório Final de Avaliação à empresa); (iv) Marco 4 (Publicação do resultado da avaliação no site SOFTEX).
- *Processo Gerência de Requisitos:* Para implementação da Gerência de Requisitos, também, foram definidos dois documentos, um para cada tipo de trabalho e que são instanciados para cada trabalho contratado considerando requisitos específicos. Um aspecto interessante e de grande valor para a qualidade dos trabalhos foi o procedimento estabelecido para avaliação da consistência. Esta avaliação, realizada pelo avaliador adjunto, é realizada em três momentos: (i) 1ª. Avaliação: Consistência entre Requisitos, Comunicação à SOFTEX, Acordo de Confidencialidade, Planilha de Seleção de Projetos e Plano de Avaliação para a Avaliação Inicial; (ii) 2ª. Avaliação: Consistência entre Requisitos, Resultado da Avaliação Inicial, Acordo de Confidencialidade e Plano de Avaliação para a Avaliação Final; (iii) 3ª. Avaliação: Consistência entre Requisitos, Plano de Avaliação, Resultado da Avaliação Final e Relatório Final de Avaliação.
- *Processo Entrega de Serviços:* A estratégia para entrega de serviços está descrita no Guia de Avaliação MPS, no processo da IA COPPE, nos Planos de Trabalho para avaliações MPS e avaliações conjuntas MPS/CMMI e nos Documentos de Requisitos para avaliações MPS e avaliações conjuntas. A colocação do serviço em operação é feita, após a formalização do aceite a uma proposta ou de um contrato para avaliação, através da comunicação à SOFTEX da avaliação.
- *Processo Gerência de Incidentes:* para Gerência de incidentes foi definido um documento com os procedimentos para Gerência de Incidentes da IA, onde estão descritos os procedimentos para incidentes de ocorrência mais comum ou que já ocorreram na IA. Muitos destes procedimentos estão presentes no Guia de Avaliação MPS. Outros incidentes podem ocorrer e, neste caso, não devem ser resolvidos pelo Avaliador Líder. O procedimento estabelecido para este caso foi o Avaliador Líder entrar em contato com o Coordenador da IA, buscando a solução e caso o Coordenador da IA não possa dar uma solução entrar em contato com o Coordenador Executivo do MPS, Gerente da Qualidade SOFTEX ou com o Coordenador da ETM por telefone. E, por fim, registrar a solução acordada por telefone através de um e-mail. Também foi definido um documento da IA para registro dos incidentes, que de fato ocorreram, solução adotada e status. Boa parte dos incidentes previstos anteriormente eram situações previstas no Guia de Avaliação do MPS (por exemplo, a exclusão de representante da empresa

na equipe de avaliação por mudança de data da avaliação final), no entanto, outros itens foram acrescentados ao longo do tempo (por exemplo, a mudança de Lead Appraiser CMMI por alteração de data da avaliação final).

- *Processo Gerência de Nível de Serviço*: O acordo do nível de serviço entre a IA e os clientes está definida nas propostas apresentadas que contém atividades, cronograma, custos, desembolso, recursos a serem fornecidos pelo cliente e responsabilidades. O acordo é gerenciado através do Plano de Trabalho e Relatório de Monitoração. De forma geral, o prazo para tratamento dos incidentes era decorrente das regras contidas no Guia de Avaliação MPS (por exemplo, a troca de um membro da equipe de avaliação deve ser executada dentro do cronograma de avaliação e prazos da SOFTEX para reenvio de comunicação de avaliação).

## 4.2. Benefícios Percebidos e Lições Aprendidas

Dentre os benefícios percebidos, pode-se destacar:

- A formalização dos processos executados pela IA.
- Maior organização do trabalho da IA em alguns aspectos.
- Maior visibilidade e melhor acompanhamento da execução dos serviços da IA tanto por parte do Coordenador da IA quanto da equipe.
- Disseminação do conhecimento incorporado nos ativos de processo da IA que foi capturado a partir das lições aprendidas da equipe.
- Satisfação da equipe.
- Formação da equipe com experiência de implementar e ser avaliada no MPS-Serviços.
- Um atestado de qualidade importante sobre os serviços executados como Instituição Avaliadora fornecido pela SOFTEX.

Dentre as lições aprendidas, pode-se destacar:

- A implementação do processo Gerência de Trabalhos e implantação do Plano de Trabalho fez tornar mais visível aos participantes dos trabalhos (ou seja, dos avaliadores) a distribuição de recursos e responsabilidades;
- A avaliação de consistência, executada como uma revisão por pares, mesmo não sendo requerida para o nível G, deu mais segurança à equipe na execução do processo e possibilitou a identificação de pequenos erros nos documentos produzidos antes de serem enviados aos clientes. Mesmo avaliadores experientes não estão imunes a esses tipos de problemas.
- Ocorreram 2 incidentes durante o período de preparação para a avaliação, sendo 1 em cada trabalho (ambos citados na subseção anterior). Assim, é possível dizer que mesmo com os incidentes sendo previstos, parcialmente, pelo Guia de Avaliação do MPS, é necessário instituir e refinar procedimentos de tratamento de ações adequados e pertinentes a cada IA. Para exemplificar, a Tabela 1 apresenta os 2 incidentes ocorridos e o tratamento indicado.

- A estratégia de implementação adotada se mostrou adequada, com o envolvimento do máximo de pessoas da IA (todos avaliadores líderes ou adjuntos com grande experiência na condução de avaliações) que puderam contribuir para a definição de processos e mecanismos de controle adequados à necessidade da IA.
- O apoio, comprometimento e envolvimento de todos foram fundamentais para o sucesso dessa iniciativa. Mesmo à distância foi possível contar com todos para a definição e revisão dos processos e, obviamente, para a condução dos trabalhos avaliados.
- A existência de um processo padrão formal auxiliou a todos a adotarem os mesmos procedimentos e, assim, estarem cientes de suas responsabilidades e, também, poderem contribuir com melhorias.
- Mesmo com recursos limitados foi possível executar todas as atividades de implementação e avaliação em um tempo bastante limitado. A otimização dos procedimentos para definição e execução dos processos foi fundamental para que isso fosse possível. A formação dos recursos humanos também contribuiu neste aspecto.
- Uma dificuldade relacionada à implementação do MR-MPS-SV é compreender adequadamente o serviço que a organização presta, pois em comparação com o desenvolvimento de software, os serviços prestados por uma organização podem ser muito diversificados.

**Tabela 1 – Exemplo de Procedimento para Tratamento de Incidentes**

<b>Incidente</b>	<b>Procedimento</b>	<b>Responsabilidade</b>
Impossibilidade de participação do <i>lead appraiser</i> em avaliação conjunta	- Como não há problemas do ponto de vista do SCAMPI pode ser feita a alteração do lead appraiser - Rever o Acordo de Confidencialidade na avaliação final - Comunicar a SOFTEX.	<i>Análise:</i> Avaliador Líder <i>Resolução:</i> Avaliador Líder
Incidentes com equipe de avaliação previstos no Guia de Avaliação	- Seguir procedimentos estabelecidos no Guia de Avaliação - Em caso de custos extras de transporte ou hospedagem, estes devem ser apresentados e aprovados pelo cliente. - Caso não seja conveniente apresentar os custos ou a empresa não concorde, os custos devem ser assumidos pela IA.	<i>Análise:</i> Avaliador Líder <i>Resolução:</i> Avaliador Líder

## 5. Considerações Finais

Este artigo apresentou a experiência de implementação do Nível G do MR-MPS-SV na Instituição Avaliadora COPPE/UFRJ que culminou com o sucesso de uma avaliação oficial nesse modelo em abril de 2013. Foram apresentados os principais detalhes da implementação dessa iniciativa de melhoria de processos e as principais lições aprendidas.

Pode-se destacar que a implementação do MPS-Serviços na Instituição Avaliadora (IA) COPPE foi uma experiência enriquecedora para toda a equipe. Tanto a implementação quanto a avaliação foram



importantes para maior conhecimento da equipe sobre o modelo MPS-Serviços, tornando a COPPE mais habilitada para implementá-lo em outras empresas. Também foi bastante útil para a Instituição Avaliadora, visto que possibilitou a introdução de controles e atividades que contribuíram para o incremento da qualidade na prestação de serviços. A satisfação e reconhecimento da importância do modelo de referência MR-MPS-SV foi grande. Por isso, dentre os trabalhos futuros pode-se destacar a evolução para os próximos níveis na Instituição Avaliadora e a implementação do MR-MPS-SV para os serviços relacionados à implementação de processos de software e de serviços.

Por fim, mas não menos importante, devido aos benefícios obtidos pela implementação do MR-MPS-SV, os participantes da equipe avaliada (e também autores deste artigo) recomendam fortemente a implementação do modelo para outras organizações por acreditarem que o modelo tem grande potencial para auxiliar outras empresas em busca da qualidade na implementação de seus serviços.

## Agradecimentos

Os autores gostariam de agradecer a Sheila Reinehr e Renato Machado pelas observações e melhorias sugeridas durante a realização avaliação MPS. Agradecem também à SOFTEX pelo apoio à realização da avaliação.

## Referências

- Diirr, T., Santos, G. (2013) "Fatores Críticos de Sucesso para Melhoria de Processos de Serviços de TI: Um Mapeamento Sistemático". In: XII Simpósio Brasileiro de Qualidade de Software - SBQS 2013, Salvador, BA, Julho de 2013.
- Santos, G., Katsurayama, A.E., Zanetti, D., *et al.* (2009a) "Lições Aprendidas em uma Iniciativa de Melhoria de Processos de Software sob Diferentes Perspectivas: Membros da Organização, Implementadores e Avaliadores ". In: VIII Simpósio Brasileiro de Qualidade de Software - SBQS 2009, Ouro Preto - MG, Junho 2009.
- Santos, G., Montoni, M., Silva Filho, R.C., *et al.* (2009b) "Indicadores da Implementação do Nível E do MR-MPS em uma Instituição de Pesquisa". In: VIII Simpósio Brasileiro de Qualidade de Software - SBQS 2009, Ouro Preto - MG, Junho 2009.
- SOFTEX (2012a) MPS.BR - Melhoria de Processo do Software Brasileiro, Guia de Implementação de Serviços - Parte 1: Nível G, Associação para Promoção da Excelência do Software Brasileiro – SOFTEX.
- SOFTEX (2012b) MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral de Serviços, Associação para Promoção da Excelência do Software Brasileiro – SOFTEX.
- SOFTEX (2012c) MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral de Software, Associação para Promoção da Excelência do Software Brasileiro – SOFTEX.

# Avaliação do Processo de Medição em Gerência de Incidentes e Gerência de Continuidade e Disponibilidade à Luz do MR-MPS-SV

Bianca Trinkenreich, Gleison Santos

Programa de Pós-Graduação em Informática, Departamento de Informática Aplicada, Universidade Federal do Estado do Rio de Janeiro (UNIRIO), Av. Pasteur, RJ, Brasil

{bianca.trinkenreich, gleison.santos}@uniriotec.br

**Abstract.** *An effective measurement process helps organizations to understand their capabilities to develop workable plans to execute and deliver agreed services to customers. MR-MPS-SV measurement process supports the company on understanding process behavior and, with that, also supports decision making, to control and improve them. This article presents an experience report about measuring Incident Management and Service Continuity and Availability Management processes in a large global company, to evaluate adherence to MR-MPS-SV Measurement process.*

**Resumo.** *Um processo de medição eficaz ajuda a compreensão da habilidade de uma organização em desenvolver planos viáveis para executar e entregar os serviços acordados com seus clientes. No MR-MPS-SV, o processo Medição apoia a compreensão do comportamento dos processos e, com isso, a tomada de decisão para controlá-los e melhorá-los. Este artigo apresenta um relato de experiência sobre a medição nos processos de Gerência de Incidentes e Gerência de Continuidade e Disponibilidade de uma empresa global de grande porte, visando avaliar a aderência do processo adotado à luz do processo Medição do modelo de maturidade MS-MPS-SV.*

## 1. Introdução

Orientação sobre como desenvolver e melhorar as práticas de maturidade de serviços é um fator chave para o desempenho do fornecedor de serviços e a satisfação do cliente. Os modelos CMMI-SVC [CMMI Institute, 2010] e MR-MPS-SV [Softex, 2012a] foram criados para atender a esta necessidade. Em ambos os modelos, a medição é fundamental à tomada de decisões ao apoio para melhoria de processos da organização.

Quando realizada apropriadamente, a medição é fundamental em atividades de melhoria de processos. O fato de ser capaz de fazer previsões em relação aos serviços que entrega é um fator de sucesso para a organização. Um processo de medição eficaz ajuda a empresa a compreender suas capacidades, para que possa desenvolver planos viáveis para executar e entregar os serviços acordados com seus clientes. Medições também ajudam a detectar tendências e antecipar problemas, proporcionando assim um melhor controle de custos, redução de riscos, melhoria da qualidade e garantia de alcance dos objetivos de negócio [Florac e Carleton, 1997].

Este artigo apresenta um relato de experiência sobre a avaliação da medição realizada sobre os processos de Gestão de Incidentes e Gerência de Continuidade e Disponibilidade dos Serviços de

uma empresa global de grande porte. A avaliação do processo atual foi guiada à luz dos resultados esperados do processo Medição do MS-MPS-SV e a avaliação dos indicadores foi guiada à luz de revisões na literatura.

O artigo está composto da seguinte forma: a Seção 2 apresenta a revisão da literatura sobre o Modelo de Maturidade MR-MPS-SV (e mais especificamente os processos considerados neste trabalho - Medição, Gerência de Incidentes e Gerência de Continuidade e Disponibilidade dos Serviços), a Seção 3 apresenta a descrição da experiência e, por fim, a Seção 4 apresenta as considerações finais.

## 2. Medição e o Modelo de Referência MR-MPS-SV

O Modelo de Referência MPS para Serviços (MR-MPS-SV) [Softex, 2012a] é um modelo de maturidade voltado principalmente para micro, pequenas e médias empresas que fornecem serviços de TI e permite avaliar a execução dos processos com diferentes graus de efetividade. A definição dos processos se baseia nos seguintes modelos e normas: ISO/IEC 15504-2 [ISO/IEC, 2003], MR-MPS-SW [Softex, 2012b], ISO/IEC 20000 [ISO/IEC, 2011], CMMI-SVC [CMMI Product Team, 2010].

O processo Medição (MED) faz parte do nível F do MR-MPS-SV e inclui coleta, armazenamento, análise e relato sobre dados relativos aos serviços fornecidos e os processos realizados na organização. A medição fornece informações quantitativas para apoiar a tomada de decisão. É fundamental selecionar medidas adequadas para apoiar esta tomada de decisão, baseadas nos processos mais relevantes para os objetivos da organização, aqueles que podem contribuir para seu sucesso ou fracasso. Como não há direcionamento claro sobre que processos seriam relevantes, uma das dificuldades das organizações é selecionar processos e medidas adequados para monitorar e melhorar o gerenciamento da qualidade dos serviços fornecidos, afetando positivamente a percepção de qualidade dos clientes [Softex, 2012a].

Todos os processos do MR-MPS-SV devem ser medidos (devido ao RAP 4), mas não há uma indicação direta sobre que aspectos relevantes devem ser selecionados como alvo. O guia de implementação sugere 8 exemplos, todos de processos específicos do MR-MPS-SV: Gerência de Configuração (GCO), Gerência de Incidentes (GIN), Gerência de Problemas (GPL), Gerência de Nível de Serviço (GNS), Gerência de Mudanças (GMU), Gerência de Liberação (GLI), Gerência de Continuidade e Disponibilidade de Serviço (GCD) e Gerência de Capacidade (GCA) [Softex, 2012c]. Neste artigo será analisada a medição sobre os processos GIN e GCD, conforme explicado na Seção 3.

Um incidente é “uma interrupção não planejada de um serviço, uma redução na qualidade de um serviço ou um evento que ainda não impactou o serviço para o cliente” [ISO/IEC, 2011].

O propósito do GIN, nível G do MR-MPS-SV, é restaurar os serviços aos níveis acordados e cumprir as solicitações de acordo com os Acordos de Nível de Serviço (ANS) entre o fornecedor e seu cliente [Softex, 2012a]. Gerenciar inclui registrar, acompanhar, escalonar (se necessário), e encerrar o incidente. Quanto mais rapidamente um incidente é tratado, maior a disponibilidade do serviço e a satisfação do usuário. O prazo para atendimento dos incidentes deve ser alinhado com os objetivos do negócio e registrado no ANS. Desta forma, GIN se relaciona com o processo Gerência de Nível de Serviço (GNS), também parte do nível G do MR-MPS-SV, cujo objetivo é garantir o cumprimento do ANS [Softex, 2012a].

Um processo especial para tratar incidentes mais graves deve ser criado, definindo o que é um incidente grave, quem tem a autoridade para declará-lo desta forma, quem deve coordenar sua resolução, como ser resolvido, quais comunicações devem ser feitas durante resolução e qual relacionamento com o processo Gerência de Continuidade e Disponibilidade dos Serviços (GCD), quando sua invocação é necessária [ISO/IEC, 2012]. O processo GCD, do nível C do MR-MPS-SV, tem como propósito assegurar que acordos de níveis de serviços sejam cumpridos em circunstâncias previsíveis, protegendo os interesses dos clientes e demais partes interessadas. Mesmo com altos níveis de disponibilidade, sempre existe a possibilidade de ocorrerem falhas. A gerência de disponibilidade garante que haja uma resposta profissional para situações indesejáveis.

### 3. Descrição da Experiência

A experiência relatada nesse artigo aconteceu na área de Tecnologia de Informação de uma empresa global de grande porte, que executa seus processos seguindo as práticas da biblioteca ITIL [TSO, 2011b], mas não é avaliada em nenhum modelo de maturidade de software ou serviços. O propósito da experiência foi utilizar os resultados esperados do processo Medição do MR-MPS-SV para guiar a avaliação da qualidade do processo atual, considerando indicadores relacionados a subprocessos GIN e GCD, apoiando a organização com a proposta de ações de melhoria do processo e dos indicadores.

A motivação para esta experiência ocorreu devido ao grande esforço despendido pelos funcionários da área de Serviços de Tecnologia da Informação em realizar a medição para atender os indicadores criados a partir dos objetivos estratégicos da organização. Os indicadores são transformados em metas mensuráveis que os funcionários precisam alcançar anualmente, e o monitoramento das ações em execução para o alcance nestas metas é realizado durante todo o período. Os indicadores são criados relacionando diferentes áreas, com objetivo de motivar o trabalho intra e entre equipes em prol da melhoria da qualidade do serviço como um todo. Desta forma, os membros das equipes não se preocupam apenas com os processos que as suas áreas atendem, mas também apoiam a execução de outras áreas. Além de metas de projetos e redução de custos, também existem metas relacionadas ao atendimento de incidentes e ao tempo de disponibilidade de sistemas considerados críticos de alto impacto ao negócio. O escopo deste trabalho se limitará à avaliação do processo de medição para metas relacionadas à GIN (Percentual de incidentes resolvidos dentro do prazo acordado no ANS) e GCD (Tempo total de indisponibilidade ao usuário em casos de crise – *TTR Time to Repair*). Os dois indicadores foram selecionados por serem aqueles que a empresa têm maior dificuldade de atender. Um analista da equipe de qualidade foi entrevistado para explicar como as medidas são coletadas e como os indicadores são calculados. Analistas das equipes de Disponibilidade e Incidentes foram entrevistados para explicar o motivo da seleção dos indicadores em questão e a associação deles com os objetivos do negócio.

Para entender os indicadores é importante conhecer o conceito de “Crise” e “Aplicações Críticas de Alto Impacto”. Este indicador está relacionado a GIN e GCD. Um incidente candidato a ser escalado para uma situação de “Crise” está relacionado a uma aplicação crítica de alto impacto (KPE - *Key Production Environment*). As aplicações classificadas desta forma impactam o negócio da organização caso seu nível de serviço seja alterado. Elas são divulgadas para o Service Desk saber como classificar o incidente caso ele chegue por um usuário ou por um evento de monitoramento. Um incidente sobre uma localidade inteira indisponível também é candidato a “Crise”. Assim, os processos GIN e GCD

estão relacionados. A assertividade na definição dos incidentes candidatos a crises permitem que o processo de escalação de incidentes seja mais preciso, aumentando a produtividade e apoiando na redução do tempo dos indicadores.

### 3.1. Descrição do Processo de Medição Atual em Termos dos Resultados Esperados de Medição (MED) e das Etapas Propostas pelo Método GQM

Os indicadores e suas respectivas medidas foram identificados a partir dos objetivos estratégicos da organização, documentados na Tabela 1, seguindo a etapa de definição do GQM [SOLINGEN e BERGHOUT, 1999]. Desta forma, verificou-se que a Tabela 1 evidencia os resultados esperados *MED1 - Objetivos de medição são estabelecidos e mantidos a partir dos objetivos de negócio da organização e das necessidades de informação de processos técnicos e gerenciais* e *MED2 - Um conjunto adequado de medidas, orientado pelos objetivos de medição, é identificado e definido, priorizado, documentado, revisado e, quando pertinente, atualizado*

**Tabela 1. Objetivos, questões e medidas**

Objetivos Organizacionais	Objetivo da Medida	Processo Associado	Medida Base	Medida Derivada	Indicador
Reduzir o tempo de resolução de incidentes	Fornecer uma visão sobre incidentes atendidos no prazo	GIN	MB1: Qtde incidentes atendidos no prazo MB2: Qtde total de incidentes atendidos	MB1/MB2	Incidentes atendidos no prazo (por criticidade)
Reduzir tempo de indisponibilidade de aplicações críticas	Fornecer uma visão sobre a efetividade de resolução de crises	GCD	MB3: Tempo para escalar incidente a crise MB4: Tempo total da crise	MB3+MB4	Diminuir tempo total em crise

Mensalmente o analista de qualidade executa coleta e armazena as medidas. Para o indicador "Diminuir tempo total em crise", ele recebe um relatório da equipe de crises e somente copia os valores para a planilha de medição que contém as fórmulas para análise do indicador. Para o indicador "Incidentes atendidos no prazo", ele executa uma busca no banco de dados do sistema de gerenciamento de incidentes para extrair a quantidade de incidentes atendidos no prazo e o total de incidentes atendidos. Os valores são exportados para a planilha de medição com fórmulas para análise do indicador.

O procedimento sobre como coletar os dados é seguido e o armazenamento das medidas é feito, mas não está documentado. Portanto, o processo atual não atende ao *MED3 - Os procedimentos para a coleta e o armazenamento de medidas são especificados*. O plano de medição e análise para cada indicador contém a fonte da medida, a unidade, seu percentual da meta total (que é formada por um conjunto de indicadores) sua fórmula de cálculo, seu objetivo, descrição e escopo, seus valores de referência e respectivas pontuações, conforme a Tabela 2, e assim, verificou-se que o processo atual atende ao resultado esperado *MED4 - Os procedimentos para a análise das medidas são especificados*.

Tabela 2. Planos de Medição para os indicadores abordados

<b>Indicador:</b> Incidentes atendidos no prazo	<b>Período de Apuração:</b> Anual	<b>Responsável pela medição:</b> Equipe de Qualidade	<b>Percentual da meta total:</b> 10,00%
<b>Evidências:</b> Relatórios gerados pela equipe de qualidade		<b>Unidade de medida:</b> %	<b>Início:</b> 01/01/2014
<b>Cálculo:</b> Número de incidentes fechados dentro do SLA / Número incidentes fechados			<b>Fim:</b> 31/12/2014
<b>Objetivo, descrição e escopo e fonte de apuração:</b> Este indicador visa atingir a eficiência para a maior parte de incidentes Severidade 1 e 2 para aplicações críticas de alto impacto, fonte: base de dados de incidentes.			
<b>Níveis de atingimento:</b>	<b>Mínimo:</b> 85,36%	<b>Target:</b> 89,94%	<b>Máximo:</b> 92,53%
<b>Pontuação:</b>	50	100	125
<b>Indicador:</b> Diminuir tempo em crise	<b>Período de Apuração:</b> Anual	<b>Responsável pela medição:</b> Equipe de Qualidade	<b>Percentual da meta total:</b> 10,00%
<b>Evidências:</b> Relatórios gerados pela equipe de crises		<b>Unidade de medida:</b> Horas	<b>Início:</b> 01/01/2014
<b>Cálculo:</b> Tempo para escalar incidente a crise + Tempo total da crise			<b>Fim:</b> 31/12/2014
<b>Objetivo, descrição e escopo e fonte de apuração:</b> Reduzir o impacto ao negócio causado por indisponibilidade de aplicações críticas de alto impacto.			
<b>Níveis de atingimento:</b>	<b>Mínimo:</b> 1900 horas	<b>Target:</b> 1500 horas	<b>Máximo:</b> 1100
<b>Pontuação:</b>	50	100	125

Tabela 3. Coleta de dados para os indicadores abordados

	Tempo entre incidente e crise	Tempo em crise	Tempo total de indisponibilidade	Total de incidentes atendidos no prazo	Total de incidentes atendidos	Percentual de incidentes atendido no prazo
<b>Jan/2014</b>	26:06:00	90:51:00	116:57:00	416	1038	40,08%
<b>Fev/2014</b>	24:13:00	126:59:00	151:12:00	422	745	56,64%
<b>Mar/2014</b>	14:50:00	117:28:00	132:18:00	378	654	57,80%
<b>Abr/2014</b>	11:50:00	259:22:00	271:12:00	479	661	72,47%
<b>Mai/2014</b>	1:23:00	6:48:00	8:11:00	411	741	55,47%
<b>Jun/2014</b>	30:12:00	69:11:00	99:23:00	375	870	43,10%
<b>Jul/2014</b>	27:16:00	85:25:00	112:41:00	451	759	59,42%
<b>Ago/2014</b>	48:09:00	217:55:00	266:04:00	436	698	62,46%
<b>Total até Agosto/2014</b>			1157:58:00			55,93%

O processo MED não requer que sejam feitas previsões através de uma análise quantitativa e estatística como os níveis mais altos de maturidade (A e B), mas para avaliar se os processos são capazes de atingir tais metas mensais, os processos precisam ser analisados estatisticamente. Apesar de a Figura 1 mostrar que os processos são estáveis, os intervalos entre os limites de controle de ambos indicadores são muito grandes, e não permitem atender efetivamente aos objetivos da organização ou tomar conclusões efetivas de previsibilidade sobre o comportamento futuro dos processos. A organização deveria investigar oportunidades de melhoria para diminuir esta variação, de forma a facilitar o planejamento da equipe e o esforço necessário para atender incidentes dentro do SLA e diminuir o tempo de indisponibilidade de aplicações críticas de alto impacto.

Existe uma reunião mensal de líderes, mas a análise da medição de cada indicador é discutida apenas trimestralmente, quando são avaliados os valores acumulados no ano, para avaliar a situação atual em relação ao alcance das metas. Nesta reunião são verificados os sistemas que mais influenciaram o indicador que estiver abaixo da meta e são criados planos de ações corretivas para melhoria. Entretanto, não há um local onde todos podem acessar diretamente e acompanhar os resultados das medições até o momento. Desta forma, resultado esperado MED 7 - *Os dados e os resultados das análises são comunicados aos interessados e são utilizados para apoiar decisões* foi considerado parcialmente atendido.

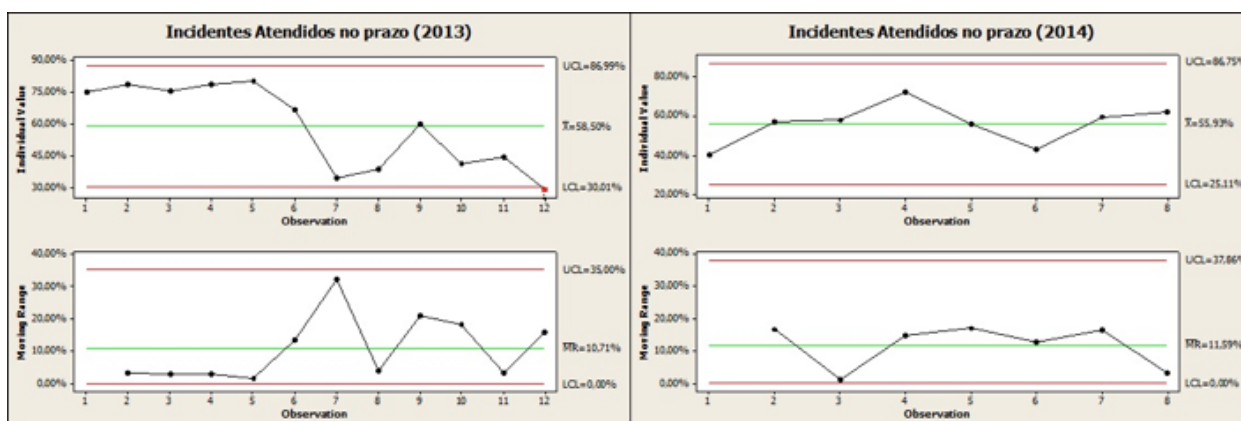


Figura 1 - Gráficos de Controle para “Incidentes atendidos no prazo”

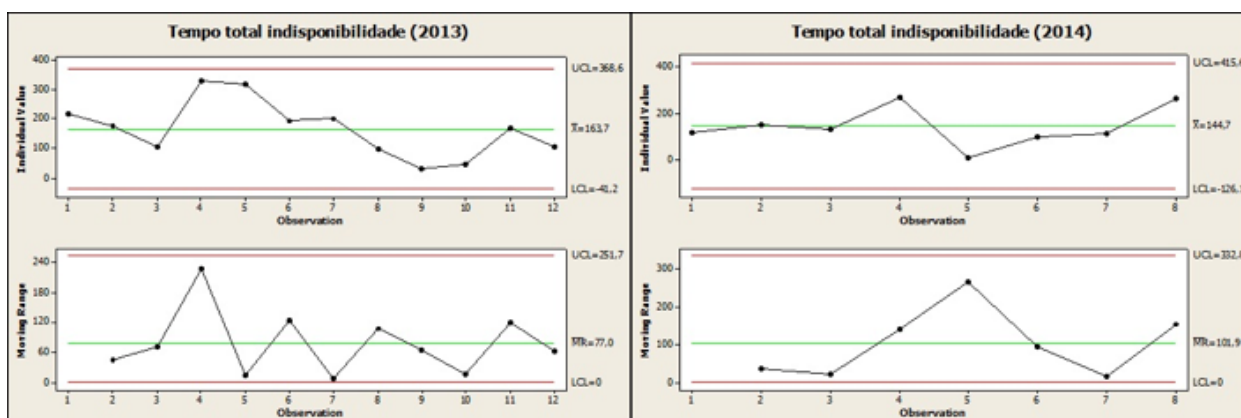


Figura 2 - Gráficos de Controle para “Diminuir tempo em crise”

### 3.2. Análise dos indicadores e propostas de melhoria

Foi observado que o indicador “Diminuir tempo total em crise” atende diretamente o processo GCD, mas é também associada a outros processos do MR-MPS-SV, como: (i) GIN, pois um incidente é escalado à crise quando está impactando uma aplicação crítica de alto impacto e ferindo o ANS acordado entre a área de TI e o cliente; (ii) GNS, pois o objetivo do indicador é reduzir ao máximo o tempo total de impacto ao usuário, visando atender o ANS em relação ao tempo de disponibilidade dos sistemas; (iii) GPL, pois um registro de problema é sempre aberto após uma crise para investigação da causa raiz da falha. Observamos também que o indicador «Incidentes atendidos no prazo” atende diretamente o processo GIN, mas é também associada ao processo GNS, pois o objetivo do indicador é atender os incidentes dentro do tempo acordado no ANS.

A qualidade dos indicadores pode ser melhorada. O indicador “Diminuir tempo total em crise” seria mais efetivo medindo diretamente o tempo de indisponibilidade do serviço, em crise ou não. O objetivo deste indicador é incentivar os envolvidos a fazerem o melhor possível para que a crise seja fechada o quanto antes, mas medir a indisponibilidade diretamente através de ferramenta de monitoramento teria a coleta de dados automatizada, menos passível a erro humano, mais barata e confiável. Outras medidas de avaliação da disponibilidade poderiam ser: Tempo médio para restaurar um serviço após falha e Tempo médio entre falhas de serviço [CATER-STEEL *et al.*, 2014].

O indicador “Incidentes atendidos no prazo” é observado na literatura sobre medição de qualidade de serviços de TI, mas outras medidas para avaliação do gerenciamento de incidentes poderiam ser usadas: Incidentes relacionados a desempenho, capacidade, continuidade, integridade, disponibilidade e confidencialidade de informação; Incidentes atendidos por dia [CATER-STEEL *et al.*, 2014]; Tempo de resolução dos incidentes fechados dentro/fora do SLA; Quantidade de incidentes repetidos, causados por execução de mudanças e que geraram abertura de problemas; Taxa de aumento de problemas em relação a incidentes [GAO *et al.*, 2011]; Quantidade de incidentes por estado, abertos e resolvidos no mês [VALVERDE, 2014], entre outros.

### 3.3. Modificações Propostas ao Processo de Medição e Lições Aprendidas

A utilização da abordagem de avaliação do processo atual de medição à luz do modelo de maturidade MR-MPS-SV, mesmo sem pretensão de avaliação da empresa, poderá apoiar a melhoria do processo, que precisa documentar os diferentes processos de coleta e armazenamento de dados e comunicar mais objetivamente o resultado das medições. Esta verificação do processo será replicada a outros indicadores relacionados aos demais processos (Gerência de Mudanças, Capacidade, Trabalho etc.). As ações indicadas de melhoria são: criar uma área no Microsoft Sharepoint, que é a ferramenta de colaboração oficial da empresa, documentar formalmente o processo de coleta e armazenamento dos dados (MED3), armazenar a documentação e os dados de forma centralizada, organizada e independente de acesso de um único funcionário (MED6), e prover acesso a medição de todos os indicadores, mostrando a situação até o momento (MED7).

As melhorias propostas ao processo de medição poderão apoiar a melhoria da qualidade dos serviços, uma vez que os gestores terão visibilidade constante do resultado que os indicadores estão apresentando, sendo capazes de agir em melhorias de seus processos de serviços e logo observar se as ações surtiram efeito nos indicadores.



A Tabela 4 apresenta algumas lições aprendidas com esta experiência.

**Tabela 4 – Lições Aprendidas**

#	Lição Aprendida
1	
2	A utilização de indicadores relacionados a mais de uma área de processo faz com que as equipes diferentes trabalhem juntas e apoiem-se mutuamente para atingir o resultado esperado.
3	A empresa consegue fomentar o trabalho entre membros das equipes ao condicionar o resultado individual ao resultado do grupo. Por exemplo, o analista de suporte da produção responsável pelo sistema X pode ter os incidentes de seu sistema atendidos no prazo, mas ele precisa que o analista da sua equipe, responsável pelo sistema Y também tenha os incidentes de seu sistema atendidos no prazo, para alcançar a meta da equipe que ambos fazem parte.
4	A empresa consegue fomentar o trabalho entre diferentes equipes, ao utilizar, mesmo com pesos diferentes, o indicador de uma equipe para a outra. Por exemplo, o gerente de projetos pode ter 20% de sua meta total relacionado a entrega do projeto no prazo. Para contar com o apoio da produção, a meta total da equipe de produção também conta com o indicador de entrega de projetos no prazo, mesmo que o percentual seja menor, de 5%.
5	“ equipes”(entre os membros da mesma equipe)

## 4. Conclusões

Este artigo apresentou uma avaliação sobre a medição nos processos de Gerência de Incidentes e Gerência de Continuidade e Disponibilidade de Serviços de uma empresa global de grande porte, sob à luz do processo Medição do modelo de maturidade MS-MPS-SV. Verificou-se que as medidas são corretamente derivadas dos objetivos estratégicos da organização, mas que o processo de medição e os próprios indicadores podem ser ampliados para atenderem melhor os objetivos da empresa.

Os próximos passos desta experiência serão avaliar os próprios processos de serviços GCD e GIN, para verificar como eles podem ser melhorados e assim a melhoria refletir no resultado de seus indicadores. Também será verificado como a medição e o Controle Estatístico de Processos podem ajudar a empresa a melhorar os processos. Além disso, avaliar os demais indicadores da empresa, estudar a correlação entre os processos de serviços e avaliar a estabilidade e capacidade de processos associados aos indicadores para encontrar aqueles que podem ser sugeridos para aplicação do Controle Estatístico de Processos (CEP) [FLORAC e CARLETON, 1997], apoiando a organização na previsão do comportamento futuro dos processos e com isso avaliar se estará apta a atender suas metas definidas.

### Agradecimentos

Os autores agradecem à FAPERJ pelo auxílio financeiro (projeto E-26/110.438/2014).

## Referências

- Forrester, E., Buteau, B., Shrum, S. (2010) -CMMI For Services, Guidelines for Superior Service. CMMI-SVC Version 1.3 - 2nd Edition. SEI. Addison-Wesley Professional.
- Demarco, T. (1982). Controlling software projects, Prentice-Hall, 1982.
- Florac e Carleton (1997) "Practical Software Measurement: Measuring for Process Management and Improvement" Software Engineering Institute
- International Organization for Standardization/ International Eletrotechnical Commision (2003) "ISO/IEC 15504-2: Information Technology - Performing an Assessment".
- International Organization for Standardization/ International Eletrotechnical Commision (2011) "ISO/IEC 20000 Information Technology – Service Management".
- Softex (2012a) Associação para Promoção da Excelência do Software Brasileiro "MPS.BR – Guia Geral MPS de Serviços" Disponível em [www.softex.br](http://www.softex.br)
- Softex (2012b) - Associação para Promoção da Excelência do Software Brasileiro "MPS.BR Guia Geral MPS de Software" Disponível em [www.softex.br](http://www.softex.br).
- Softex (2012c) Associação para Promoção da Excelência do Software Brasileiro – "Guia de Implementação – Parte 2: Fundamentação para Implementação do Nível F do MR-MPS-SV:2012" Disponível em [www.softex.br](http://www.softex.br). (p.50)
- Solingen, R., Berghout, E. (1999) "The Goal/Question/Metric Method: A Practical Guide for Quality Improvement of Software Development". McGraw-Hill.
- TSO, 2011a (2011) The Stationay Office ITIL - Service Operation.
- Cater-Steel et al (2014) "The Evaluation of the IT Service Quality Measurement Framework in Industry", Global Journal of Flexible Systems Management
- Gao et al. (2011) "Case Study on IT Service Management Process Evaluation Framework Based on ITIL" Business Management Electronic Information Conference
- Valverde et al. (2014) "DSS Based IT Service Support Process Reengineering Using ITIL: A Case Study" Journal Intelligent Decision Technologies



# Relato de Implementação da Gestão de Portfólio como Ferramenta de Execução da Estratégia

Karina K. Hartmann<sup>1</sup>, Osmar A. M. Pedrozo<sup>1</sup>

<sup>1</sup> SoftDesign Consultoria e Sistemas,  
Rua Siqueira Campos, 1184 – Sala 101 – 90010-001 – Porto Alegre – RS – Brazil

{karina.hartmann, osmar.pedrozo}@softdesign-rs.com.br

**Abstract.** *This paper is an experience report of the implementation of Portfolio Management at SoftDesign in compliance with MPS-SW level F. The implementation includes the AHP method, used to derive portfolio selection criteria from Balanced Scorecard. Our results include the lessons learned and next steps.*

**Resumo.** *Neste artigo é apresentado o relato da experiência da SoftDesign na implantação de Gestão do portfólio de projetos, aderente ao MPS-SW nível F. Esta implantação inclui o uso de método AHP para derivação de critérios a partir do Balanced Scorecard. Nossos resultados incluem lições aprendidas desta implantação.*

## 1. Introdução

Pesquisas apresentadas em Kaplan e Norton (2009) indicam que: a maior parte das empresas fica muito longe de alcançar seus objetivos; não dispõe de sistemas e processos de execução das estratégias; e seus funcionários declaram não conhecer a estratégia da organização. Muito esforço é posto em realizar um bom Planejamento Estratégico, no entanto “as estratégias dão errado mais frequentemente porque não são bem executadas” [Bossidy e Charan 2005].

Surge daí a necessidade de identificar ferramentas capazes de apoiar a execução da estratégia. Neste sentido, buscando a essência de uma execução de sucesso, temos em Kaplan e Norton (2009) que a execução da estratégia requer a gestão coordenada de várias iniciativas que devem ser tratadas como um portfólio integrado. Menard (2012) explica que os projetos selecionados e financiados hoje irão determinar o desempenho dos negócios da organização amanhã.

A Gestão do Portfólio de Projetos é uma ferramenta de apoio à execução da estratégia que na etapa de seleção visa “criar uma combinação de projetos que melhor apoie os objetivos da organização” [Levine 2005], e durante o acompanhamento de sua execução apoia o monitoramento do desempenho dos projetos, para que estes “continuem aderentes e satisfazendo os objetivos pelos quais foram iniciados” [SOFTEX 2013].

A próxima questão que se apresenta é como comparar os projetos possíveis durante a fase de seleção, de forma a selecionar o conjunto mais apropriado de projetos dadas as restrições de recursos da organização.

Neste sentido é necessário adotar ou desenvolver algum modelo de pontuação que possa ser usado para avaliar os projetos e torná-los comparáveis entre si [PMI, 2008]. Um modelo deste tipo é dado por um conjunto de critérios chave, sendo que cada critério recebe um peso. Avaliar todos os projetos a partir do mesmo conjunto de critérios faz com que cada projeto receba uma nota em uma escala comum.

A criação destes critérios de avaliação requer alinhamento com os objetivos da organização. Apresentamos neste artigo um caso de alinhamento feito diretamente a partir da estratégia da organização declarada no mapa estratégico. Já para a atribuição de pesos aos critérios, este relato apresenta a adoção da técnica AHP - *Analytic Hierarchy Process*.

O artigo está organizado da seguinte forma. A seção 2 é dedicada ao contexto do relato, apresentando a empresa e as particularidades do ambiente que afetam a gestão do portfólio. A seção 3 apresenta o relato em si, subdividida em uma visão geral do processo implantado na SoftDesign, um aprofundamento na fase de seleção e por fim a apresentação do uso do método AHP. Na seção 4 são apresentadas lições aprendidas e a seção 5 conclui o relato.

## **2. Contexto do relato**

### **2.1. Empresa**

Este relato está baseado na experiência da empresa SoftDesign Consultoria e Sistemas, sediada em Porto Alegre – RS. A empresa tem aproximadamente 60 colaboradores e está no mercado desde 1997, prestando serviços de desenvolvimento de software, outsourcing e consultoria em Tecnologia da Informação.

A SoftDesign realizou a avaliação de nível G do MPS-SW em 2009, e durante os anos de 2012 e 2013 realizou a implementação dos níveis F e E simultaneamente, período no qual desenvolveu-se o processo objeto deste relato, realizando avaliação formal em novembro de 2013.

### **2.2. Portfólio de projetos da SoftDesign**

Pela natureza da empresa, o portfólio de projetos da SoftDesign inclui tanto projetos internos, que visam desenvolver capacidades novas ou reduzir custos da própria organização, quanto projetos de consultoria ou desenvolvimento de software para clientes.

Este fator torna mais desafiadora a comparação e seleção de projetos, já que os projetos são de naturezas bem distintas e não podem ser todos reduzidos à uma medida financeira única como, por exemplo, ROI ou TIR.

Antes da implantação tratada neste artigo a gestão do portfólio era realizada de maneira informal. Já havia a identificação do responsável pela decisão e de um conselho que, dentro da estrutura de governança, apoiava a decisão. No entanto, não havia registro formal das decisões e os critérios eram subjetivos.

### 3. Relato da implantação

#### 3.1. Visão geral do processo

O processo de Gestão de portfólio foi desenhado a partir da combinação do conjunto de boas práticas propostas pelo PMI em seu *Standard for Portfolio Management* e dos resultados esperados e orientações do MPS-SW nível F.

Entre os objetivos desta implementação, a organização definiu:

- Fazer da gestão de portfólio uma ferramenta da implantação da estratégia, garantindo que o conjunto de projetos em execução esteja alinhado à estratégia organizacional.
- Garantir que os projetos sejam monitorados em conjunto, para que decisões sejam tomadas de acordo com sua prioridade atualizada e com sua situação.

A Figura 1 mostra uma visão resumida do processo, dividido em três grandes fases: Planejamento, Seleção, Monitoramento. No Planejamento é feita a preparação para executar o processo, na Seleção os projetos são avaliados e autorizados e no Monitoramento os projetos já autorizados são monitorados e ações são tomadas para que o portfólio se mantenha aderente aos seus objetivos.

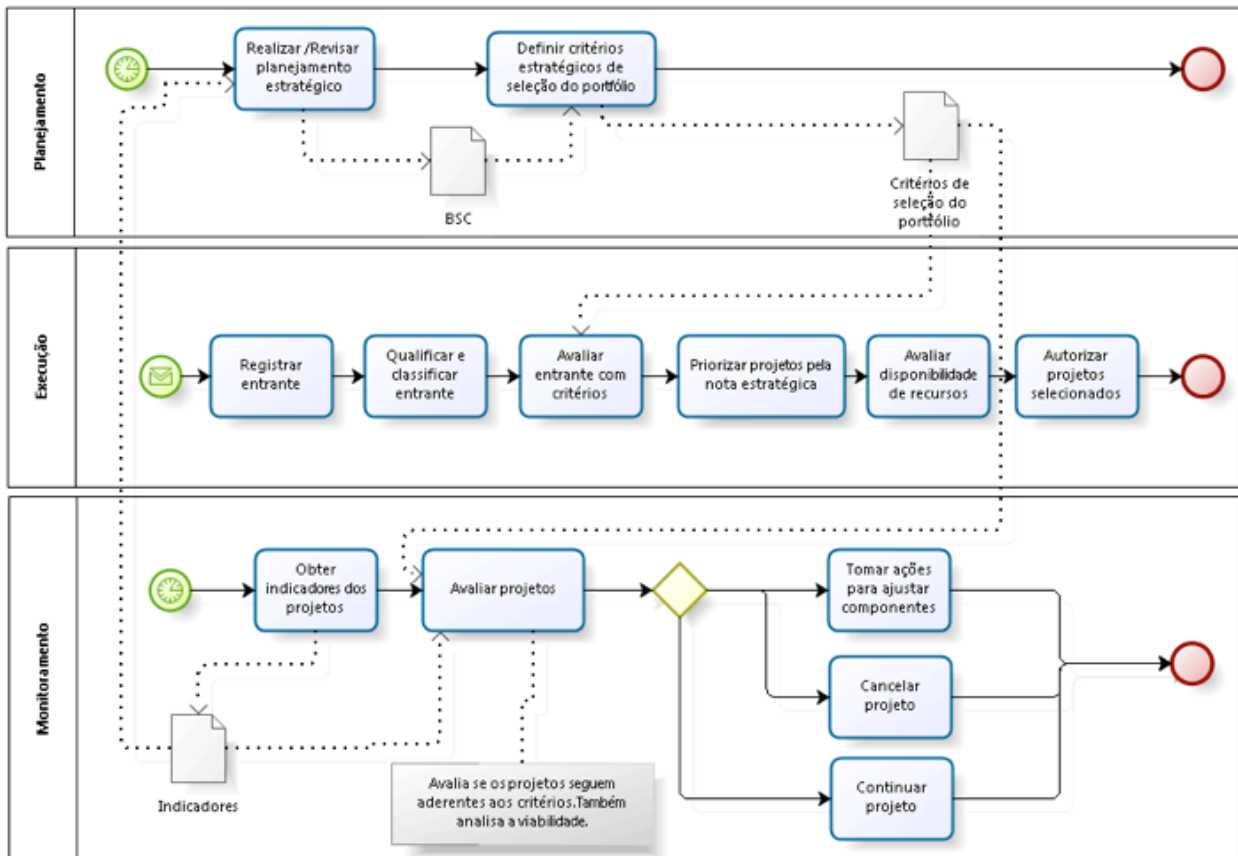


Figura 1 - Visão resumida do processo implantado. Fonte: os autores.

O ciclo inicia sempre que o planejamento estratégico da organização é realizado ou revisado. A estratégia é traduzida em um mapa estratégico, que é um dos componentes do BSC – *Balanced Scorecard* – proposto por Kaplan e Norton. O BSC é um sistema de gerenciamento estratégico através do qual a empresa traduz sua missão e estratégia em objetivos e medidas tangíveis em quatro diferentes perspectivas: Financeira, Clientes, Processos internos e Aprendizagem. [Kaplan e Norton 2001].

O BSC é utilizado como entrada para definir os critérios que serão utilizados na fase de Seleção. Nesta fase as oportunidades de projetos e outros trabalhos, chamados de entrantes do portfólio, são identificados, qualificados, avaliados, priorizados e sua necessidade de recursos é avaliada contra a disponibilidade. Por fim são confeccionados termos de abertura para os projetos autorizados.

Estando selecionados os projetos, a etapa de execução requer acompanhamento. A organização deve basear o relato e acompanhamento em “um conjunto de métricas pré-definidas e pré-aprovadas que monitorem o atingimento de objetivos estratégicos, a contribuição financeira, satisfação dos stakeholders, perfil de risco e utilização de recursos” [PMI 2008].

Para o acompanhamento, a organização define um conjunto de indicadores a serem usados por todos os projetos. Estes indicadores são utilizados para monitorar os projetos, realizar avaliações de viabilidade em marcos e tomar ações para ajustar seu desempenho. Além disso, estes indicadores são agregados para fornecer uma visão da situação do portfólio como um todo e por fim retroalimentam o processo de gestão da estratégia.

### **3.2. Seleção de projetos e derivação de critérios**

Um dos principais conceitos motivadores para a gestão do portfólio é que há mais projetos possíveis do que recursos disponíveis para realizá-los. Seja por recursos humanos, financeiros ou outros, a capacidade da organização de realizar projetos e demais trabalhos é limitada. Assim, é necessário selecionar quais projetos serão realizados de forma a obter os melhores resultados para a organização.

No processo implantado na SoftDesign esta avaliação é realizada para cada projeto entrante, ou seja, cada novo projeto de investimento interno ou oportunidade externa. Estes entrantes, segundo o processo, devem ser registrados na ferramenta apropriada juntamente com as informações de qualificação que permitam a avaliação.

Para esta etapa de avaliação o PMI (2008) sugere três técnicas:

- Modelo de pontuação abrangendo critérios chave ponderados: permite comparar os itens sendo avaliados em um escala única.
- Representações gráficas: Representações como gráficos de bolha, gráficos de linha e histogramas podem ajudar a comparar visualmente os itens.
- Opinião de especialista: usada para analisar questões como o relacionamento entre os componentes sendo avaliados, como redundância ou itens mutuamente excludentes.

No processo implantado na SoftDesign foi usada uma combinação destes métodos: um modelo de pontuação permite avaliar e comparar todas as oportunidades; a visão gráfica permite compreender a distribuição e balanceamento do portfólio; por fim, quando a opinião especialista do responsável pela avaliação vai contra a pontuação obtida matematicamente, este deve registrar sua opinião para que seja apreciada por um conselho que apoia a decisão.

O modelo de pontuação é uma das formas possíveis de atender o resultado esperado GPP1 do MPS. BR [SOFTEX 2013] que requer critérios objetivos para a avaliação e seleção. A necessidade de critérios objetivos é explicada por Saaty (1994), segundo o autor, existem duas formas de decisão, intuitiva e analítica, e a decisão intuitiva tem diversas fraquezas, como a dificuldade de ser aceita por outros envolvidos por não ter justificativas claras, e a impossibilidade de revisões futuras para aprendizado.

Conforme PMI (2008), os projetos concorrendo por recursos em uma organização devem ser avaliados de forma a permitir a comparação, facilitando o processo de seleção. No caso do portfólio, é necessário comparar projetos e outros trabalhos, para os quais não há uma escala padronizada que permita a comparação, e cada organização precisa construir uma escala que faça sentido em seu contexto.

Um modelo de pontuação “consiste de uma série de critérios de avaliação com peso expresso em percentual e com uma pontuação associada” [PMI 2008]. Além de incluir critérios objetivos, um modelo de pontuação tem a vantagem de colocar todos os itens em uma escala única, que permita comparação.

Para o desenvolvimento do modelo de pontuação foi usado o método AHP – *Analytic Hierarchy Process* – proposto pelo professor Dr. Thomas L. Saaty em 1977. Este método permite a criação de uma escala para grandezas relativas através de comparações pareadas. Segundo Chen (2006), quando diferentes critérios compõem uma seleção, sejam eles objetivos ou subjetivos, o método AHP possibilita a atribuição de pesos para cada um, com base na avaliação da importância de um critério sobre o outro mediante uma escala comum. Para Finocchio (2010), quando usada para implementação de um portfólio, a técnica AHP cria uma estrutura hierárquica com os múltiplos objetivos do planejamento estratégico.

### 3.3. Aplicação do método AHP

A primeira etapa do método AHP é a decomposição de um problema em um objetivo, critérios e alternativas. Dado que um dos objetivos da implantação de Gestão de portfólio para a SoftDesign é que o portfólio seja uma ferramenta de execução da estratégia, o objetivo, neste caso, é “Selecionar o melhor conjunto de projetos de forma a realizar a estratégia da organização”.

A segunda etapa do método é o julgamento comparativo entre os critérios. Cada critério deve ser comparado com os demais, de modo a ser possível definir uma hierarquia entre eles. A comparação entre os critérios é dada em uma matriz, onde a intensidade do relacionamento entre os critérios é estabelecida utilizando a escala proposta pelo método AHP, conforme Tabela 1. Ao final é calculado o autovetor da matriz que determina os pesos de cada critério.

No caso da SoftDesign, para que o portfólio apresentasse alinhamento com a estratégia, os critérios adotados foram os objetivos estratégicos da organização, obtidos do mapa estratégico.



**Tabela 1. Escala de comparação de critérios. Adaptada de Saaty (1994).**

Intensidade da importância	Definição	Observação
1	Importância igual	Os dois elementos contribuem de forma igual para o objetivo da seleção.
3	Importância moderada	Experiência e julgamento ligeiramente favorecem um elemento sobre o outro.
5	Importância forte	Experiência e julgamento favorecem fortemente um elemento sobre o outro.
7	Importância muito forte	Um elemento é muito fortemente favorecido em detrimento de outro.
9	Importância extrema	A evidência favorecendo um elemento sobre o outro é da mais alta possibilidade de afirmação.
2,4,6,8	Intermediários	Quando um valor intermediário for necessário.
Inversa	$a_p = 1 / a_{ij}$	Se um elemento i receber uma graduação quando comparado com o elemento j, então a graduação de j será o inverso da graduação de i.

Os objetivos estratégicos foram comparados em pares e assim foi obtida a hierarquia e o peso de cada objetivo para a organização, conforme Figura 2. Importante observar que, para a construção do exemplo a seguir, os dados da empresa foram modificados, de forma a não expor a estratégia da organização.

	Aument.. 1	Conqui... 2	Aument.. 3	Desenv.. 4	Reduz... 5	Incentiv.. 6		
Aumentar lucratividade 1	1	5	3	3	5	3	33,8%	
Conquistar novos clientes 2	1/5	1	1/3	7	1/3	5		15,1%
Aumentar a satisfação dos clientes 3	1/3	3	1	5	5	7		25,8%
Desenvolver produtos inovadores 4	1/3	1/7	1/5	1	5	1		10,4%
Reduzir defeitos dos produtos 5	1/5	3	1/5	1/5	1	5		10,8%
Incentivar a qualificação contínua 6	1/3	1/5	1/7	1	1/5	1		4,1%

**Figura 2 - Matriz de comparação. Fonte: os autores, adaptado de Saaty (1994).**

Utilizando os objetivos hierarquizados em uma escala, foi construída uma planilha de avaliação, onde os projetos são inseridos e avaliados contra cada um dos critérios, como ilustrado pela Figura 3.

Um fator de sucesso para um modelo de pontuação, segundo PMI (2008), é prover orientação para atribuição de cada nível na escala. Esta orientação foi implementada na planilha através da criação uma lista de opções para cada critério, já vinculadas à sua pontuação, garantindo que a avaliação seja objetiva.

Entrantes	Dados de identificação		Critério 1		Critério 2		Critério 3		Critério 4		Critério 5		Critério 6		Nota ponderada de priorização
			Aumentar lucratividade		Conquistar novos clientes		Aumentar a satisfação dos clientes		Desenvolver produtos inovadores		Reduzir defeitos dos produtos		Incentivar a qualificação contínua		
	Origem	Tipo	Em qual alternativa esse projeto se encaixa?	Peso: 33,8	O cliente deste projeto é novo?	Peso: 15,1	De que forma o projeto contribui?	Peso: 25,8	Em qual categoria esse projeto se encaixa?	Peso: 10,4	Contribui para a redução de defeitos e aumento da qualidade?	Peso: 10,8	Apoia a qualificação das pessoas envolvidas?	Peso: 4,1	
Projeto x	Externo	Fábrica de software	Lucratividade acima de X%	33,8	Cliente totalmente novo	15,1	Não se aplica	0	Projeto de desenvolvimento de produto inovador para cliente	5,2	Não se aplica	0	Indiretamente	2,05	0,56
Projeto y	Externo	Fábrica de software	Lucratividade entre Y% e X%	16,9	Cliente novo nesta categoria	7,55	Não se aplica	0	Não se aplica	0	Não se aplica	0	Indiretamente	2,05	0,27
Projeto z	Externo	Fábrica de software	Lucratividade abaixo de Y%	0	Cliente antigo	0	Não se aplica	0	Não se aplica	0	Não se aplica	0	Indiretamente	2,05	0,02
Projeto a	Externo	Consultoria	Lucratividade entre Y% e X%	16,9	Cliente novo nesta categoria	7,55	Não se aplica	0	Não se aplica	0	Indiretamente, contribui para melhoria dos serviços/processos	5,4	Indiretamente	2,05	0,32
Projeto b	Interno	Melhoria de processos	Lucratividade abaixo de Y%	0	Cliente interno	0	Iniciativa de melhoria de processos e serviços	25,8	Projeto de desenvolvimento de produto inovador interno	10,4	Projeto de melhoria de processos	10,8	Diretamente	4,1	0,51

Figura 3 - Planilha de avaliação por critérios ponderados. Fonte: os autores.

O resultado desta avaliação é uma nota de priorização que estabelece a contribuição do projeto para a estratégia da organização. Esta nota é então avaliada juntamente com o balanceamento do portfólio e com a disponibilidade de recursos.

A disponibilidade de recursos pode ser decisiva para a autorização de projetos. Isto significa que um projeto de menor nota pode ser iniciado em detrimento de outro de maior nota, caso o primeiro exija recursos que estão disponíveis e o segundo exija recursos indisponíveis.

Outro fator importante é que os projetos são constantemente reavaliados, já que novos entrantes são comparados com os projetos em execução. Um projeto entrante com uma nota mais alta terá prioridade para obter recursos, levando a análises sobre a possibilidade de reduzir a equipe, parar ou descontinuar projetos em andamento.

## 4. Resultados e lições aprendidas

Dentre os aspectos positivos apontados pelos usuários do processo temos:

- A adoção de um sistema de informação para registro de entrantes, juntamente com a formalização do processo de seleção, tornou mais organizada e rápida a avaliação, aumentando a eficiência da organização para dar retorno para oportunidades.
- O uso de critérios objetivos tornou a seleção de projetos mais rápida, objetiva e profissional. O uso dos critérios também facilitou a justificativa para projetos não aceitos.
- O alinhamento dos critérios com a estratégia fez com que a implementação da estratégia aconteça de forma mais natural, já que os projetos selecionados colaboram para a estratégia. Também possibilitou que as equipes de projetos tenham maior clareza sobre como seu trabalho colabora para a organização, funcionando como um instrumento de comunicação da estratégia.

- O processo se propôs a integrar todos os projetos e trabalhos da organização em um único portfólio de projetos. Este aspecto foi positivo, porque todos os projetos e trabalhos competem por recursos entre si, então todos devem ser comparados em conjunto para uma correta priorização.

Dentre os aspectos negativos apontados pelos usuários do processo temos:

- A avaliação dos entrantes é feita em uma planilha. Se os critérios estivessem implementados no mesmo sistema onde os entrantes são cadastrados, o processo de seleção seria mais rápido e ainda mais transparente.

## 5. Conclusão

O objetivo deste artigo foi relatar um caso de implantação de Gestão do Portfólio de Projeto em uma organização. O artigo relata como esta implantação tratou a seleção de projetos de forma a fazer da Gestão do Portfólio uma ferramenta de execução da estratégia, integrando projetos, portfólio e estratégia pelo estabelecimento de critérios de seleção. Dentre os principais resultados da implantação é possível citar o aumento da maturidade da organização na execução de sua estratégia empresarial.

Espera-se que este relato venha a contribuir como modelo para outras organizações que buscam implementar a Gestão de Portfólio, principalmente por contemplar técnicas e lições aprendidas da implementação da Gestão de Portfólio.

Como trabalhos futuros, é possível explorar para novos relatos os demais aspectos do processo de gestão do portfólio de projetos como o balanceamento do portfólio, retroalimentação da estratégia organizacional a partir do monitoramento do portfólio e análise de riscos do portfólio, bem como os resultados do uso destes conceitos nas organizações.

## Referências

- Bossidy, L. and Charan, R. "Execução: a disciplina para atingir resultados". Rio de Janeiro: Elsevier, 2005.
- Chen, C. "Applying the Analytical Hierarchy Process (AHP) Approach to Convention Site Selection". EUA: SAGE Publication, 2006.
- Finocchio, J. J. "Alinhando o Portfólio de Projetos ao Mapa Estratégico". Mundo PM, número 32, abril/maio, 2010.
- Kaplan, S. Norton, P. "A Execução Premium". Rio de Janeiro: Elsevier, 2009.
- Levine, H. "Project Portfolio Management – A Practical Guide to Selecting Projects, Managing Portfolios, and Maximizing Benefits". San Francisco: John Wiley & Sons, 2005. Kindle Version.

Menard, M. "A fish in your ear – The New Discipline of Project Portfolio Management". CreateSpace, 2012. Kindle Version.

PMI. "The Standard for Portfolio Management – second edition", PMI Publishing Division, 2008.

Saaty, T. "Fundamentals of Decision Making and Priority Theory with the Analytic Hierarchy Process". Pittsburgh: RWS Publications, 1994.

SOFTEX. "MPS.BR – Guia de Implementação – Parte 2: Fundamentação para Implementação do Nível F do MR-MPS-SW:2012". SOFTEX, 2013.



# Um Relato de Experiência da Implantação de Soluções de Rastreabilidade Bidirecional em Empresas de Salvador

Glauco de F. Carneiro<sup>1</sup>, João Werther Filho<sup>2</sup>, Cristiane Marise Perez da S. Carneiro<sup>1</sup>, Vinicius Santos<sup>3</sup>, Renato Pinto Vilela<sup>4</sup> e Martin Noel<sup>5</sup>

<sup>1</sup>Programa de Pós Graduação em Sistemas e Computação – Universidade Salvador (UNIFACS)

<sup>2</sup>Desenvolva Tecnologia em Sistemas Ltda. – Salvador – BA – Brasil

<sup>3</sup>Atena Tecnologia – Salvador – BA – Brasil

<sup>4</sup>Medicware Sistemas de Informática Ltda. – Salvador – BA – Brasil

<sup>5</sup>Union Informática – Salvador – BA – Brasil

glauco.carneiro@unifacs.br, werther@desenvolva.info, cristiane.mpcarneiro@gmail.com, vinicius@atenaonline.com.br, renato.vilela@medicware.com.br, martin@unioninformatica.com.br

**Resumo.** *A rastreabilidade bidirecional de requisitos é uma importante técnica em Engenharia de Software. Além de possibilitar a avaliação de impacto de mudanças em projetos de software, também apoia a identificação de possíveis inconsistências entre os produtos de trabalhos de um projeto de desenvolvimento de software e os seus requisitos. Este artigo relata exemplos reais da adoção da rastreabilidade bidirecional de requisitos em empresas de desenvolvimento de software de Salvador/BA em seus respectivos projetos de software no contexto da implementação do MPS.BR Software. O artigo também apresenta as etapas seguidas por cada empresa para a solução de rastreabilidade, incluindo a decisão por qual apoio ferramental adotar.*

## 1. Introdução

A rastreabilidade de requisitos tem o objetivo de identificar e relacionar os requisitos de um determinado projeto de software com outros artefatos dele derivados ou que os implementem nas diversas fases do ciclo de vida do software. A rastreabilidade é fundamental para a análise de impacto de mudanças de requisitos. A rastreabilidade deve ser bidirecional, ou seja, desde os requisitos até as entidades que o implementam (por exemplo, código fonte correspondente), podendo passar antes disto pelos documentos que descrevem estes requisitos (por exemplo, casos de uso ou estórias de usuário) e também no caminho inverso. O relacionamento entre os requisitos e os artefatos que o implementam compõem um grafo acíclico direcionado complexo, também conhecido como matriz de rastreabilidade [Bourque e Fairley, 2014]. Manter uma matriz atualizada é uma atividade que deve ocorrer durante todo o ciclo de vida de um projeto de software. O resultado de informações de rastreabilidade não atualizadas ao longo da ocorrência de mudanças de requisitos é a não confiabilidade das consultas realizadas na matriz para a análise de impacto. A rastreabilidade bidirecional também é fundamental para a implementação do processo Gerência de Requisitos em modelos como o MPS.BR Software e o CMMI-DEV.

Diversas ferramentas apoiam a rastreabilidade bidirecional de requisitos. Desde ferramentas de software livre tais como o *RedMine*<sup>1</sup> e *OSRMT*<sup>2</sup> até ferramentas cujas licenças devem ser adquiridas, tais como a *Enterprise Architect*<sup>3</sup> e *Rational Doors*<sup>4</sup>. Além destas opções, pode-se também desenvolver e customizar uma solução própria para apoiar a rastreabilidade de requisitos. Outra opção é a customização de uma planilha eletrônica. O que se verifica na prática é que esta última opção é mais susceptível a falhas do que as demais. Uma das limitações está relacionada ao seu uso e manutenção que geralmente não é trivial para projetos de software de tamanho médio e grande. A identificação de qual destas opções é a mais adequada, efetiva e com melhor relação custo-benefício é um desafio enfrentado na prática por diversas empresas, especialmente aquelas que planejam a implementação do processo de Gerência de Requisitos baseado em um modelo de referência de processo de software.

Este artigo apresenta um relato de experiência prática do planejamento e implantação de soluções de rastreabilidade bidirecional em cinco empresas de Salvador/BA. Estas empresas estão implementando os níveis G (quatro empresas) e F (1 empresa) do MPS.BR Software. O artigo descreve as seguintes fases de implantação das soluções: a compreensão dos conceitos de rastreabilidade, a identificação dos itens para compor a rastreabilidade, a tomada de decisão pelo apoio ferramental a ser utilizado e o plano de ação para a sua implantação.

O restante deste artigo está organizado da seguinte forma. A Seção 2 apresenta conceitos importantes de rastreabilidade de requisitos. A Seção 3 apresenta o perfil das empresas que fazem parte deste relato de experiência. A Seção 4 aborda a solução adotada por cada empresa para a rastreabilidade de requisitos considerando as fases citadas. Por último, a Seção 5 apresenta as lições aprendidas com essa experiência prática e considerações finais.

## 2. Conceitos Relacionados à Rastreabilidade de Requisitos

A literatura apresenta diversas definições da rastreabilidade de requisitos. Uma definição bastante conhecida estabelece que a rastreabilidade de requisitos refere-se à habilidade em se descrever e seguir o ciclo de vida de um requisito tanto na direção ao código fonte como na direção inversa [Gotel e Finkelstein, 1994]. Ou seja, os relacionamentos existentes entre os requisitos e diversos artefatos utilizados ao longo do ciclo de vida do projeto de software, incluindo o código fonte, podem ser relacionados entre si e apoiar atividades como a análise de impacto e a identificação de inconsistências entre os requisitos, os planos de projeto e os produtos de trabalho do projeto [Softex, 2012]. Vale ressaltar que a interdependência entre os requisitos exerce influência em várias atividades e decisões no ciclo de vida do projeto. Não considerar todas as dependências quando da avaliação do impacto de uma mudança pode resultar em situações e resultados não esperados para o projeto e para a aplicação [Aurum e Wohlin, 2005].

Outra questão importante é que os requisitos de uma solução automatizada podem mudar a qualquer momento e por diversos motivos. Para lidar com esta realidade, é necessária uma abordagem para integrar estes novos requisitos e/ou resultados da sua modificação com os requisitos existentes e também com os demais produtos de trabalho e artefatos relacionados ao projeto de software. Isto possibilitará à equipe do projeto avaliar possíveis impactos desta mudança. Assim, quando mudanças são propostas em requisitos ou em outros artefatos a ele relacionados direta ou indiretamente,

---

1 <http://www.redmine.org/>

2 <http://sourceforge.net/projects/osrmt/>

3 <http://www.sparxsystems.com.au/>

4 <http://www-03.ibm.com/software/products/pt/ratidoor>

estas mudanças precisam ser analisadas quanto ao seu impacto para que sejam identificadas quais partes do produto de software e artefatos relacionados também precisarão ser modificados. A este procedimento chamamos de análise de impacto de mudanças. Sabe-se que o tamanho e a complexidade do produto de software podem tornar esta atividade custosa e complexa, fato que demandará tempo considerável para sua execução. Daí a necessidade de uma abordagem apropriada para a rastreabilidade de requisitos. Isto viabiliza a identificação de possíveis inconsistências ao longo do ciclo de vida do software, podendo de certa forma evitar impactos prejudiciais no projeto, e o inconveniente do retrabalho quando estas inconsistências forem detectadas tardiamente.

### 3. Perfil das Empresas deste Relato de Experiência

Este artigo apresenta a experiência de quatro empresas implementando o nível G do MPS.BR Software (Medicware, Infocraft, Topos e Union) e uma empresa implementando o nível F (Atena). A Tabela 1 apresenta os perfis de cada empresa.

**Tabela 1: Perfil das Empresas deste Relato de Experiência**

Empresa/Perfil de Atuação
<p><b>Medicware:</b> atua no mercado nacional oferecendo a seus clientes soluções integradas de software para a gestão de saúde. Na sua equipe, além de profissionais de tecnologia, também participam profissionais com expertise em gestão de saúde, especialmente em sistemas de informação hospitalar, gestão de consultórios e clínicas, e gestão de laboratórios. A empresa desenvolve seus produtos e soluções usando a plataforma Powerbuilder<sup>5</sup>.</p>
<p><b>Infocraft Comércio, Serviços e Informática Ltda:</b> atua no desenvolvimento de software, consultoria, comercialização de soluções em TIC, digitalização de documentos e outsourcing, tanto no segmento público como privado. Atualmente possui uma carteira de 250 clientes e possui como seus principais produtos para a área privada o Grafite, software de gestão escolar, o Salescraft, software de gestão de vendas e o Tactium, software de gestão de Call Center. Na área publica destaca-se o GRF, software de gestão fiscal utilizado por diversas prefeituras no estado da Bahia.</p>
<p><b>Union Informática:</b> atua no mercado nacional e também no Peru e Angola. A empresa atua em consultoria, desenvolvimento de software e implantação de projetos de infraestrutura desde 1998. No desenvolvimento de software, oferece aos seus clientes produtos como o Sapiens – ERP Educacional e Phoenix – Planejamento e Controle de Produção. Nas atividades de Infraestrutura os focos da empresa são alta disponibilidade e virtualização.</p>
<p><b>Atena Tecnologia:</b> atua no mercado nacional nas áreas de Desenvolvimento e Licença de Uso de Soluções para o segmento de Previdência Complementar, Consultoria em TI e Suporte e Serviços de Infraestrutura de TI. O seu principal produto, a Solução Corporativa Atenaprev.Net adota o conceito de Linha de Produtos de Software e Reuso, na plataforma WEB, baseado em inovação de Produto e de Processo. A Atena foi fundada em 1997 e possui certificação nas normas ISO 9001:2008 e 20000-1:2011.</p>
<p><b>Topos:</b> tem foco em projetos e implementação de sistemas geográficos de informação e consultoria em geotecnologias e aplicações de Geoprocessamento.</p>

<sup>5</sup> [www.sybase.com.br/products/modelingdevelopment/powerbuilder](http://www.sybase.com.br/products/modelingdevelopment/powerbuilder)



### 3. Solução Adotada pelas Empresas para a Rastreabilidade de Requisitos

As etapas utilizadas nas empresas para a análise e implantação de uma solução de rastreabilidade de requisitos são descritas na Tabela 2. Os parágrafos seguintes à Tabela 2 descrevem como estas etapas foram executadas no contexto da solução adotada em cada empresa.

**Tabela 2: Etapas Adotadas para a Solução de Rastreabilidade de Requisitos**

Etapa/Descrição da Etapa
<p><b>(1) Compreensão do Conceito de Rastreabilidade de Requisitos:</b> Para a implementação do MPS.BR Software nas empresas é necessário alinhamento do conceito e objetivo da rastreabilidade de requisitos, quais as vantagens decorrentes do seu uso no produto de software em desenvolvimento, quais as principais dificuldades e desafios e qual seria a relação custo-benefício estimada para esta adoção. Esta etapa é importante para a identificação de quais itens no mínimo deveriam ser considerados na rastreabilidade.</p>
<p><b>(2) Identificação dos itens para compor a rastreabilidade de requisitos:</b> Considerando as práticas adotadas na empresa, deve-se decidir pela seleção dos itens a serem incluídos na rastreabilidade. Vale observar que diversos artefatos e itens podem ser inseridos na rastreabilidade. Entretanto, deve-se considerar aqueles artefatos que de fato são utilizados na empresa. Por exemplo, nem todas as empresas utilizam Diagrama de Classes e Diagrama de Sequência, fato que implica no seu não uso na matriz de rastreabilidade destas empresas. Outra questão importante neste momento é identificar a viabilidade das rastreabilidades horizontal e vertical com os itens selecionados. A rastreabilidade horizontal relaciona itens do mesmo tipo da matriz, por exemplo, Casos de Uso versus Casos de Uso. A rastreabilidade vertical relaciona itens de tipos diferentes como, por exemplo, Casos de Uso versus Requisitos Funcionais. A identificação dos itens da matriz é fundamental para a próxima etapa relacionada à solução do apoio ferramental.</p>
<p><b>(3) Decisão do Apoio Ferramental para a Rastreabilidade de Requisitos:</b> Uma vez discutidos os conceitos relacionados à rastreabilidade e identificados os itens e artefatos candidatos à inclusão na matriz, deve-se decidir pelo apoio ferramental a ser adotado na empresa baseado em critérios conforme descrito a seguir. A empresa deve considerar as diversas <i>opções factíveis para o apoio ferramental</i>, passando desde o uso de uma <i>planilha eletrônica</i> para inclusão e gestão da rastreabilidade, ou pela <i>aquisição de licenças de uma ferramenta comercial</i>, instalação e configuração de uma solução de <i>ferramenta livre</i> ou até mesmo pelo <i>desenvolvimento de uma solução própria</i>. Neste momento, sugere-se considerar as <i>vantagens e desvantagens</i> de cada opção, a sua <i>compatibilidade</i> com as práticas já adotadas e com a <i>cultura da empresa</i> e, principalmente, a <i>relação custo-benefício</i>.</p>
<p><b>(4) Plano de Ação para a Solução Adotada:</b> A partir da decisão do apoio ferramental a ser adotado, parte-se para a elaboração do plano de ação de forma que esteja compatível com o cronograma de implementação do MPS.BR Software na empresa. Este plano de ação deve conter as atividades a serem realizadas de acordo com a solução selecionada, juntamente com as ações, os respectivos prazos previstos, os responsáveis pelas ações e recursos necessários (materiais e humanos) para a atividade e ação descritas.</p>

### 3.1. Solução Adotada na Atena

A equipe da Atena identificou e reconheceu os conceitos de rastreabilidade pelo fato de já utilizá-los, ainda que parcialmente, na evolução dos seus produtos de software para a área previdenciária. Na identificação dos itens para compor a rastreabilidade de requisitos a empresa identificou a necessidade de ampliação dos itens a serem incluídos na matriz. Isto ficou claro quando se constatou que a rastreabilidade vertical era baseada tão somente no relacionamento entre as histórias de usuário (EU), sentenças que descrevem como as funcionalidades deverão ser implementadas, e as classes .Net (CL), código fonte no Team Foundation Server<sup>6</sup>, e vice-versa. Este cenário indicou a oportunidade de inclusão dos requisitos funcionais (RF) na rastreabilidade para que a matriz fosse percorrida desde os requisitos funcionais até o código fonte e vice-versa. A equipe da empresa reconheceu que isto daria maior visibilidade de como estes requisitos estariam sendo detalhados e em quais partes do código estariam sendo implementados. Considerada esta ampliação, o cenário resultante para a rastreabilidade horizontal foi RF x RF, EU x EU, CL x CL e para a rastreabilidade vertical: RF x EU, EU x CL, RF x CL. Uma vez definidos estes tipos de rastreabilidade, a Atena analisou como poderia utilizar o ferramental da Microsoft já instalado na empresa para apoiar a solução em discussão. Considerando que na rastreabilidade estão envolvidos artefatos, produtos de trabalho e requisitos da aplicação em desenvolvimento, a equipe da empresa identificou o Team Foundation Server (TFS) e o Source Control como dois alicerces importantes para a solução customizada. O TFS é uma plataforma de desenvolvimento colaborativo que provê gerenciamento de código fonte, gerenciamentos de requisitos, gerenciamento de projetos, automatizações de builds entre outros. O Source Control<sup>7</sup> é um sistema de controle de versões utilizado em conjunto com o TFS. Através do TFS foi possível fazer o registro de requisitos funcionais por intermédio das entidades chamadas de *work items*. Um *work item* é considerado o elemento básico do TFS. Através dele podem ser registradas atividades de um projeto, um requisito funcional, uma história de usuário, um caso de teste, um bug, entre outros. Os *work items* são armazenados em um banco de dados do TFS. Estes *work items* também podem ser customizados de acordo com a necessidade do projeto e da empresa. No caso da Atena, a partir dos requisitos funcionais registrados no TFS, tem-se o detalhamento dos mesmos nas histórias e, em seguida são designadas pelos programadores as classes que implementam cada uma destas histórias. Para implementar o relacionamento entre os elementos da matriz de rastreabilidade, a Atena precisou entender como são feitos os vínculos entre os *work items* no banco de dados do TFS. Após este entendimento, partiu-se então para a identificação de uma ferramenta de consulta à matriz e disponibilização dos resultados de relacionamentos entre os seus itens. A ferramenta selecionada foi o Power Pivot<sup>8</sup> que é um suplemento do Microsoft Excel utilizado para executar análises avançadas de dados, agregando funcionalidades de *Business Intelligence* à ferramenta, oferecendo cálculos avançados e habilidades para importar dados de múltiplas fontes. Estas funcionalidades viabilizaram o uso do Power Pivot de forma integrada ao TFS, possibilitando a consulta à matriz de rastreabilidade através do relacionamento entre os *work items* no banco de dados do TFS. Na solução encontrada, as consultas ficaram armazenadas no Power Pivot que, por sua vez, acessava diretamente o banco de dados do TFS. Com esta solução definida, configurada e disponibilizada para uso, a cada *check-in* realizado no TFS, os envolvidos no projeto (por exemplo, analista de requisitos e desenvolvedores) estabeleceram o relacionamento do *work item* que está sendo atualizado com os

6 [www.visualstudio.com/en-us/products/tfs-overview-vs.aspx](http://www.visualstudio.com/en-us/products/tfs-overview-vs.aspx)

7 [http://msdn.microsoft.com/en-us/library/zxd4dfad\(v=vs.90\).aspx](http://msdn.microsoft.com/en-us/library/zxd4dfad(v=vs.90).aspx)

8 <http://office.microsoft.com/pt-br/excel-help/suplemento-powerpivot-HA101811050.aspx>

demais *work itens* já incluídos na base (requisitos, histórias, tarefas, bugs). A partir deste momento, o Power Pivot foi habilitado para disponibilizar as consultas atualizadas dos relacionamentos da matriz de rastreabilidade.

### 3.2. Solução Adotada na Infocraft

A equipe da Infocraft concordou desde o início que o uso da rastreabilidade de requisitos traria benefícios para a gerência de requisitos dos seus projetos de desenvolvimento de software. Para esta finalidade, os seguintes itens foram selecionados para compor a matriz de rastreabilidade na empresa: (i) requisitos de cliente (RC) que foram cadastrados a partir dos dados obtidos da proposta comercial aprovada pelo cliente; (ii) requisitos funcionais (RF) e não funcionais (RN) identificados na análise de requisitos e aprovados pelo cliente; (iii) casos de uso (UC), criados logo após o comprometimento dos requisitos para detalhamento das funcionalidades do sistema. O próximo elemento definido na matriz foi o código fonte. Como os arquivos com o código fonte PHP (linguagem de programação utilizada pela empresa) que atendem uma determinada funcionalidade são armazenados em pastas específicas, decidiu-se incluir na matriz a referência para estas pastas. Quanto ao apoio ferramental, a empresa decidiu pela aquisição das licenças da ferramenta Enterprise Architect (EA). Na realidade, a empresa já tinha como meta usar esta ferramenta para apoiar as atividades de modelagem e de projeto da solução arquitetural das aplicações desenvolvidas pela empresa. E agora com a possibilidade de uso das suas funcionalidades de rastreabilidade de requisitos, verificou-se que a relação custo-benefício era interessante e vantajosa. Este cenário influenciou bastante na decisão tomada pela empresa. Para incluir o código fonte na rastreabilidade, convencionou-se na empresa que após a codificação e aprovação nos testes, o código fosse importado para o EA através de opção existente na ferramenta. Com isto foi criado mais um nível na matriz de rastreabilidade, relacionando casos de uso e requisitos não funcionais com as pastas que contém os arquivos de código fonte. Os seguintes tipos de rastreabilidades foram adotadas na solução na Infocraft: Rastreabilidade horizontal (RF x RF) e (UC x UC); rastreabilidade vertical (RC x RF), (RC x RN), (UC x PCF).

### 3.3. Solução Adotada na Medicware

A Medicware já possuía um portal que era utilizado para a gestão dos seus projetos de desenvolvimento de software. A equipe da empresa decidiu por unanimidade que o portal poderia ser customizado para viabilizar a solução. Considerando que a própria equipe que faz parte da implementação do MPS. BR também é responsável pela gestão do portal, ela especificou as funcionalidades necessárias para permitir a rastreabilidade vertical e horizontal. Isto viabilizou a avaliação de impacto e a identificação de inconsistência entre os requisitos e os produtos de trabalho e artefatos gerados a partir dele. Os seguintes itens foram incluídos na matriz de rastreabilidade na Medicware: Requisitos Funcionais (RF), Casos de Uso (UC) e Objetos (OBJ) do Powerbuilder. As seguintes consultas ficaram disponíveis no portal: rastreabilidade horizontal (RF x RF, UC x UC, OBJ x OBJ) e rastreabilidade vertical (RF x UC, UC x OBJ, RF x OBJ). Para a implementação da solução, utilizou-se como referência a forma como o RedMine é utilizado para a rastreabilidade de requisitos. A ideia foi permitir a consulta para cada um destes itens registrados na área do portal de cada projeto de desenvolvimento de software na Medicware. Esta consulta permitiu identificar quais outros itens estavam relacionados ao item

selecionado na consulta. O portal também disponibilizou uma janela para cadastro de cada um destes itens em cada projeto e um recurso para relacioná-lo a outros itens já existentes. Pelo fato desta solução estar integrada ao portal de gestão de projetos, o seu uso teve boa aceitação pelos envolvidos no projeto, em especial pelos interessados na consulta da matriz de rastreabilidade.

### **3.4. Solução Adotada na Union**

A equipe da Union já estava motivada para uso da rastreabilidade de requisitos mesmo antes da implementação do nível G do MPS.BR Software. Inclusive, uma definição preliminar dos itens para compor a matriz já havia sido iniciada pela empresa antes da implementação do MPS.BR Software. A empresa também já tinha intenção de desenvolver uma solução própria para a rastreabilidade de requisitos. A principal justificativa para isto foi a possibilidade de ter a consulta da matriz de rastreabilidade integrada a outros resultados esperados para o processo de Gerência de Requisitos e também com atividades da Gerência de Projetos. A empresa tinha como meta ter um portal único para os dois processos do nível G do MPS.BR Software. Este portal, chamado de Sistema de Documentação de Projetos (SDP), apoiava tanto a criação como o armazenamento de artefatos e produtos de trabalho de projeto tais como lista de requisitos, requisitos funcionais e não funcionais, casos de uso, casos de testes, entidades, entre outros. Cada artefato podia ser incluído e editado. Quando o mesmo era finalizado e aprovado, seu estado mudava para publicado e o sistema gerava uma nova versão do artefato. Os requisitos, por exemplo, eram cadastrados individualmente na ferramenta em formato RTF e depois era gerado um documento com todos os requisitos em um único arquivo de saída e em formato PDF. A mesma estratégia foi aplicada aos casos de uso. Neste caso, cada detalhamento de caso de uso é um artefato do sistema em desenvolvimento e a ferramenta depois emite um relatório consolidado de todos os casos de uso detalhados existentes para um dado projeto. Na realidade, a ferramenta concebida na empresa pode emitir relatórios de agrupamento de quaisquer itens por projeto. Outros artefatos utilizados no desenvolvimento do projeto, tais como os casos de testes, por exemplo, também podiam ser cadastrados no sistema. Arquivos de código fonte também podiam ser importados para a ferramenta. Na medida em que os artefatos eram cadastrados, foi possível definir o relacionamento/rastreabilidade de qualquer artefato cadastrado para outro, dentro de um mesmo projeto. O relacionamento era livre, qualquer artefato podia relacionar-se com qualquer outro, independente do tipo. E a partir de um artefato qualquer era possível ver todos os outros artefatos relacionados, assim como visualizá-los por inteiro. Apesar da possibilidade de diversidade de relacionamentos, a Union optou por relacionar verticalmente apenas requisitos com casos de uso, casos de uso com casos de testes, casos de uso com entidades e casos de uso com código fonte. A relação de rastreabilidade horizontal pode ocorrer em todos eles.

### **3.5. Solução Adotada na Topos**

A Topos optou por utilizar uma ferramenta que já era utilizada pela empresa, que era o Enterprise Architect (EA). Por já conhecerem a ferramenta, bastaram alguns ajustes no processo para que o processo de rastreabilidade fosse implementado. No primeiro nível da matriz de rastreabilidade existiam os requisitos de escopo (EP) extraídos da proposta comercial ou técnica aprovada. Eles foram cadastrados na ferramenta e possíveis rastreabilidades horizontais (EPxEP) foram também registradas.

A partir daí, no processo de elicitação, foram especificados e cadastrados na ferramenta os requisitos funcionais (RF) e os não funcionais (NF). Após a aprovação dos requisitos pelo cliente, a matriz de rastreabilidade ganhou seu segundo nível, que foi o registro da rastreabilidade entre os requisitos de escopo com os requisitos não funcionais (EPxNF) e também com os requisitos funcionais (EPxRF). Eventuais relacionamentos horizontais (RFxRF e NFxNF) também foram registrados. Em seguida, casos de uso (UC) foram especificados e cadastrados na ferramenta. Após sua aprovação, a rastreabilidade alcançou mais um nível, com o registro da rastreabilidade entre casos de uso e requisitos funcionais (UCxRF). Havia também a possibilidade de existir registros de rastreabilidade entre casos de uso (UCxUC). Com a construção e aprovação do código fonte (CF) para o software, ele podia ser importado por completo para a ferramenta, fato que viabilizou a rastreabilidade entre código fonte e casos de uso (UCxCF) e também com requisitos não funcionais (NFxCF), finalizando assim os elementos significativos para rastreabilidade. A Tabela 3 apresenta o resumo da solução de rastreabilidade em cada uma das empresas relatadas neste artigo.

#### 4. Lições Aprendidas e Considerações Finais

Este artigo apresentou um relato de experiência da adoção da solução de rastreabilidade em empresas de Salvador/BA no contexto da implementação do MPS.BR Software. Embora o foco deste relato não seja na avaliação de impacto de mudanças, as empresas já estão utilizando suas respectivas soluções de rastreabilidade para esta finalidade, pois todas já iniciaram seus respectivos projetos piloto. Conforme já apresentado e discutido, cada empresa adotou a solução a princípio mais apropriada para sua realidade considerando critérios como ferramentas já instaladas e disponíveis na empresa, e integração com estas ferramentas, relação custo-benefício, possibilidade de utilizar a ferramenta para atividades de modelagem e projeto, customização de ferramentas já em uso e viabilidade do uso da ferramenta pela equipe. Através deste relato pretende-se disponibilizar para a comunidade exemplos de diferentes soluções de rastreabilidade de requisitos aderente às necessidades das empresas. Para isto podem ser adotadas soluções com software livre, software com licenças adquiridas ou até mesmo desenvolvimento de ferramenta própria. Em relação às dificuldades, a compreensão inicial dos conceitos de rastreabilidade por toda a equipe é uma questão que deve ser considerada. Para isto sugere-se apresentar exemplos do uso da rastreabilidade considerando projetos de desenvolvimento reais da empresa e solicitando aos seus participantes que forneçam exemplos de solicitações de mudanças e como seria a análise de impacto destas mudanças com e sem uma solução de rastreabilidade bidirecional. Em relação às lições aprendidas, constatou-se que as equipes das empresas reconheceram a importância da inclusão, atualização e relacionamento entre os requisitos, artefatos e produtos de trabalho deles derivados ao longo do ciclo de vida do projeto de software. Este reconhecimento ficou evidente para as equipes quando estas puderam avaliar as atividades antes e depois da solução de rastreabilidade. Com a solução de rastreabilidade ficou evidente que a obtenção da lista de relacionamentos de requisitos com outros requisitos e ou com outros artefatos poderia ser feita sem a necessidade de consulta a um expert da aplicação. Também em relação às lições aprendidas, os patrocinadores puderam constatar a boa relação custo-benefício como um forte argumento para a adoção de uma solução de rastreabilidade apoiada por ferramenta. Isto ficou claro quando foi possível reduzir consideravelmente a possibilidade de alterações com impacto inesperado no projeto.

**Tabela 3: Resumo das Soluções de Rastreabilidade por Empresa**

<b>Empresa</b>	<b>Itens Incluídos na Rastreabilidade</b>	<b>Tipos de Rastreabilidade Bidirecional</b>	<b>Apoio Ferramental</b>
Atena	Requisitos Funcionais (RF), Estórias de Usuário (EU), Classes (CL)	Horizontal (RF x RF, EU x EU, CL x CL) Vertical (RF x EU, RF x CL, EU, CL)	Customização e Integração do Team Foundation Server (TFS), Source Control e Power Pivot
Infocraft	Requisitos de Clientes (RC), Requisitos Funcionais (RF), Requisitos Não Funcionais (RN), Casos de Uso (UC), Pastas de Código Fonte (PCF)	Horizontal (RF x RF, UC x UC) Vertical (RC x RF, RC x RNF, UC x PCF)	Enterprise Architect (EA)
Medicware	Requisitos Funcionais (RF), Casos de Uso (UC), Objetos (OBJ)	Horizontal (RF x RF, UC x UC, OBJ x OBJ) Vertical (RF x UC, UC x OBJ, RF x OBJ)	Desenvolvimento de Solução própria no Portal de Gestão e Acompanhamento de Projetos da Medicware
Union	Lista de requisitos (LR), Requisitos (RQ), Casos de Uso (UC), Casos de Teste (CT), Entidades (E), Código Fonte (CF).	Horizontal (RQ x RQ, UC x UC, CT x CT, E x E, CF x CF) Vertical (RQ x UC, UC x TC, UC x E, UC x CF)	Desenvolvimento de Solução própria no Portal de Projetos da Union
Topos	Requisitos de Escopo (EP), Requisitos Funcionais (RF), Requisitos Não Funcionais (NF), Casos de Uso (UC) e Código Fonte (CF)	Horizontal (EPxEP, RF x RF, NFxNF, UC x UC) Vertical (EP x RF, EP x NF, RF x UC, UC x CF, NF x CF)	Enterprise Architect (EA)

## Referências

- Bourque, P.; Fairley, R. (2014) "Guide to the Software Engineering Body of Knowledge – SWEBOK", IEEE Press, Piscataway, NJ, USA, 2004.Version 3.0, 2014.
- Gotel, O.; Finkelstein, A. (1994) "An analysis of the requirements traceability problem", Proc. 1<sup>st</sup>. International Conference on Requirements Engineering, 1994.
- Softex (2012) "Associação para Promoção da Excelência do Software Brasileiro - SOFTEX. MPS.BR - Guia Geral MPS de Software", agosto 2012.
- Aurum, A., Wohlin. C. Eds. (2005), "Engineering and Managing Software Requirements", Springer Verlag.



# CERTICSys para avaliações de processo da CERTICS e de outros métodos baseados na Norma ISO/IEC 15504

Alan Raldi<sup>1</sup>, Davi Silva<sup>1</sup>, Clenio F. Salviano<sup>2</sup> e Angela M. Alves<sup>2</sup>

<sup>1</sup>Fundação de Apoio a Capacitação em Tecnologia da Informação  
Rod. Dom Pedro I - km 143,6 - CEP 13069-901- Campinas-SP

<sup>2</sup>Centro de Tecnologia da Informação Renato Archer  
Rod. Dom Pedro I - km 143,6 - CEP 13069-901- Campinas-SP

{alan.raldi,davi.carvalho}@facti.com.br, {clenio.salviano,angela.alves}@cti.gov.br

**Abstract.** *CERTICS is a methodology that identifies if a software is the result of technological development and innovation carried out in Brazil. CERTICSys is a web-based software available as a SaaS (Software as a Service) that supports all phases and activities of the CERTICS assessment method. CERTICSys is based on SOA (Service-oriented architecture) and BPMS (Business Process Management Suite) in order to orchestrate the phases, activities and actors involved in CERTICS assessment. The platform and its architecture were developed with the possibility of integration with other systems. As the Methodology CERTICS follows the requirements of ISO/IEC 15504 for process assessment, the CERTICSys was developed in order to accommodate changes and to implement other methods based on ISO/IEC 15504.*

**Resumo.** *CERTICS é uma metodologia que identifica se um determinado software é resultante de desenvolvimento e inovação tecnológicos realizados no Brasil. A plataforma de software CERTICSys é uma ferramenta acessível como um serviço que apoia todas as fases e atividades do método de avaliação de processo de software da CERTICS. O CERTICSys tem uma arquitetura orientada a serviços (SOA - Service-oriented Architecture) e uma suíte de gerenciamento de processos de negócios (BPMS - Business Process Management Suite). O BPMS é o responsável pela orquestração das fases, atividades e atores que participam da avaliação. A plataforma foi desenvolvida com a possibilidade de integração com outras plataformas. Como a Metodologia CERTICS segue os requisitos da Norma ISO/IEC 15504 para avaliação de processo, o CERTICSys foi desenvolvido de forma a facilitar sua customização para outros métodos e modelos também baseados na ISO/IEC 15504.*

## 1. Introdução

A ferramenta CERTICSys é uma solução que foi desenvolvida para suportar todas as fases de avaliação de um software realizada de acordo com a Metodologia CERTICS para Software [1]. Este artigo apresenta esta ferramenta com ênfase na sua implementação e na possibilidade de apoiar outros métodos de avaliação, desde que estes sejam também baseados na Norma ISO/IEC 15504 [2]. Outro artigo apresenta a ferramenta com ênfase na arquitetura e nas tecnologias utilizadas [3].

A Metodologia de avaliação da CERTICS para Software tem dois componentes principais: o Modelo de Referência e o Método de Avaliação ([www.certics.cti.gov.br](http://www.certics.cti.gov.br)).



Este artigo está organizado em seis seções. Esta primeira seção introduz o artigo. A Seção 2 apresenta a Metodologia CERTICS. A Seção 3 apresenta a arquitetura e infraestrutura da plataforma de software da qual a ferramenta CERTICSys é a parte central. A versão atual da ferramenta CERTICSys é apresentada na Seção 4, que inclui telas da ferramenta. A Seção 5 apresenta as possibilidades da plataforma incorporar novos métodos de avaliação baseados na Norma ISO/IEC 15504. A Seção 6 resume as conclusões deste artigo.

## 2. Metodologia CERTICS

A Metodologia de Avaliação da CERTICS para Software foi desenvolvida pelo Centro de Tecnologia da Informação Renato Archer (CTI Renato Archer), com a participação de uma equipe de pesquisadores e especialistas da Fundação de Apoio à Capacitação em Tecnologia da Informação (Facti) dedicada a este objetivo. Durante seu desenvolvimento, a Metodologia CERTICS recebeu contribuições de dezenas de outros especialistas do setor de Tecnologia da Informação e Comunicação (TIC) e ficou disponível para consulta pública entre os meses de agosto a dezembro de 2012.

A Metodologia CERTICS para Software orienta como identificar e avaliar um software resultante de desenvolvimento e inovação tecnológicos realizados no País, ou seja, um software cujo desenvolvimento cria ou amplia competências tecnológicas e correlatas no País, contribuindo para a criação de negócios baseados em conhecimento, para o aumento de autonomia tecnológica e para o aumento da capacidade inovativa.

A arquitetura do Modelo de Referência (Figura 1) ilustra como o modelo foi desenvolvido a partir do conceito fundamental (software resultante de desenvolvimento e inovação tecnológicos realizados no País) detalhado em áreas de competência, resultados esperados, orientações e indicadores. O Método de Avaliação é realizado com a análise de evidências para o julgamento dos indicadores, resultados esperados e áreas de competência para determinar o resultado da avaliação.



Figura-1: Arquitetura do Modelo de Referência

As fases do método de avaliação são: Exploração, Contratação, Preparação, Visita, Validação e Conclusão. A partir da fase de Preparação, a empresa deve informar os dados solicitados pelo formulários do módulo de monitoramento, e deve completá-lo antes do final da fase de validação.

### 3. Arquitetura da plataforma CERTICSys

O CERTICSys, disponível em [www.certicsys.org.br](http://www.certicsys.org.br), entrou em produção em 19/09/2013. A aplicação e os demais componentes da arquitetura foram concebidos, projetados e desenvolvidos para suportar todas as fases da avaliação previstas no método de avaliação da Metodologia CERTICS.

A Figura 2 ilustra a arquitetura da ferramenta CERTICSys e sua plataforma.

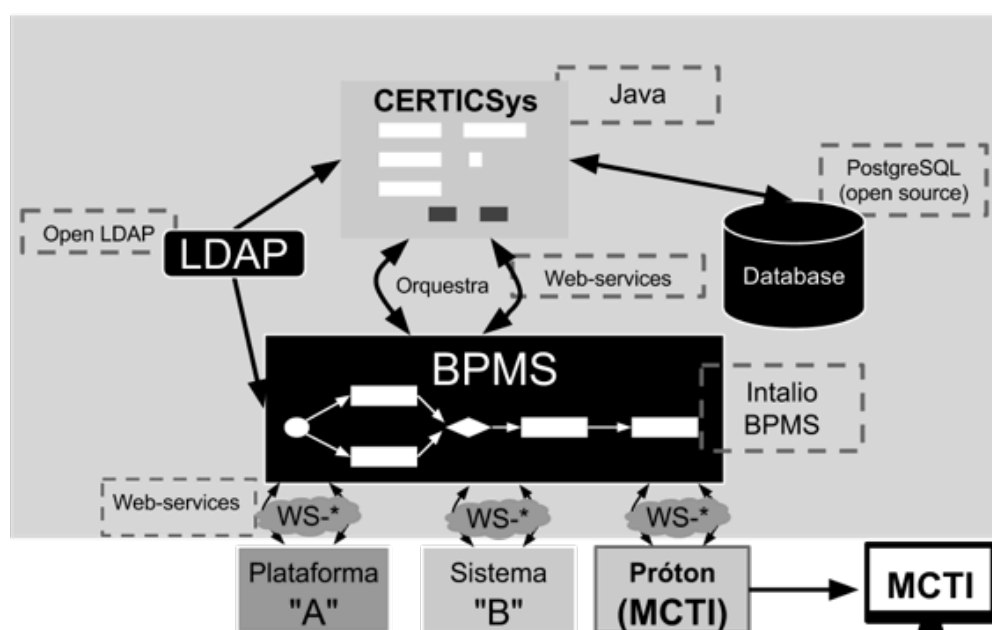


Figura-2: Arquitetura de alto nível do CERTICSys com seus principais componentes

Alguns princípios e escolhas arquiteturais utilizadas no desenvolvimento do sistema e na estruturação da plataforma que o suporta são:

- **SOA (Service-oriented Architecture):** SOA é um estilo arquitetural que suporta a abordagem de implementação de sistemas orientados a serviço, ou seja, composição de agentes de software, com propósito específico e interfaces bem definidas, são orquestrados de tal forma a executar uma ação (um serviço).
- **BPMS (Business Process Management Suite):** BPMS é uma suíte de software que automatiza a gestão de processos de negócio: modelagem, execução, controle e monitoração. No contexto da Metodologia CERTICS, o processo de avaliação foi mapeado, desenhado e automatizado na suíte BPMS. O motor (*engine*) de processos do BPMS orquestra todas as fases do processo de avaliação, bem como as atividades dos atores envolvidos.

- **LDAP (*Lightweight Directory Access Protocol*):** LDAP é um serviço usado para a autenticação, autorização e acesso tanto para a plataforma CERTICSys como para a suíte BPMS.
- **Plataforma JAVA:** Esta plataforma é utilizada no desenvolvimento da interface gráfica do usuário. Vários *frameworks* e bibliotecas baseadas nesta plataforma foram utilizados no desenvolvimento do CERTICSys. Os *frameworks* Holy e Hibernate são dois exemplos.

## 4. Versão atual do CERTICSys

A seguir estão relacionadas algumas telas e funcionalidades da versão atual da plataforma CERTICSys:

- Acesso e segurança:** A plataforma foi desenvolvida para ser executada via web, com segurança de acesso (usuário e senha, com possibilidade de certificado digital).
- Área de trabalho do usuário:** É um ambiente multi perfil que atende a todos os usuários e atores do processo de avaliação CERTICS, incluindo, por exemplo, Ponto de contato, Avaliador, Avaliador Líder, Validador e Comitê. O usuário tem acesso ao cadastro ou consulta das informações relevantes para a avaliação, além de poder acompanhar e executar tarefas. Também é possível colaborar com lições aprendidas a cada fase do processo de avaliação (Figura 3).



Figura 3: Tela inicial do CERTICSys com o menu

- Cadastro:** O usuário pode cadastrar informações sobre os seguintes itens: Organização Solicitante, Software, Profissionais, Evidências e Lições Aprendidas.
- Acompanhamentos:** Nesta funcionalidade o usuário acompanha as tarefas que foram atribuídas (Ações), verifica o status do software em avaliação, resultado preliminar, pendências, documentos e relatórios (Figura 4).



Figura-4: Tela CERTICSys para acompanhamento por software

- e. **Tutor:** A plataforma possui um tutor que apresenta cenários para facilitar o entendimento da CERTICS e a identificação e descrição das evidências.
- f. **Estimativa de Sucesso:** Com base nas informações que a Organização Solicitante fornece, o sistema gera uma Estimativa de Sucesso como uma referência para a continuação da avaliação do software (Figura 5).

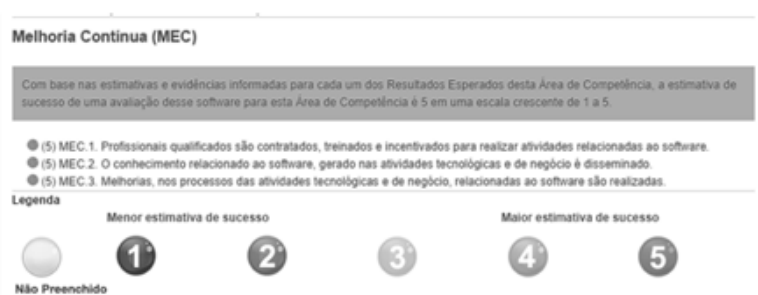


Figura-5: Tela CERTICSys com Relatório de Estimativa de sucesso

- g. **Motor de Processo (engine):** Todas as fases do método de avaliação são orquestradas via processo, da Exploração até a Conclusão.
- h. **Protocolo do MCTI:** A plataforma possui a funcionalidade de se comunicar com serviços externos como, por exemplo, o serviço de protocolo do MCTI para emissão do certificado CERTICS.
- i. **Alto nível de automação:** A plataforma foi projetada para ser 100% automatizada sem qualquer processo manual (Figura 6).
- j. **Assinatura eletrônica:** A plataforma está preparada para assinatura eletrônica de documentos, como, por exemplo, contratos e evidências.
- k. **Monitoramento:** A plataforma tem funcionalidade para coleta de informações que permite o monitoramento dos resultados da CERTICS.

## 5. Possibilidades de customização do CERTICSys

O CERTICSys foi construído com a possibilidade de customizações do método de avaliação implementado para atender o objetivo da CERTICS. Isto é possível porque utiliza uma suite de processos (BPMS), onde estão localizados os processos.

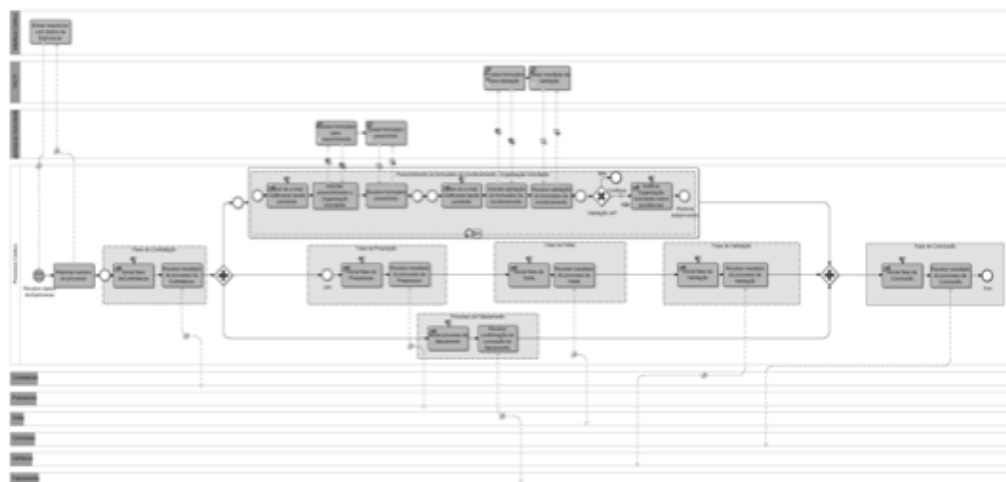


Figura-6: Macro processo de avaliação CERTICS em notação BPMN

O Modelo de Referência para Avaliação da CERTICS está descrito por Áreas de Competência que correspondem aos processos da ISO/IEC 15504. Cada Área de Competência define um conjunto de Resultados Esperados. Ao avaliar um determinado software, a organização identifica e vincula um conjunto de evidências para justificar o atendimento de cada Resultado Esperado (Figura 7). Esta estrutura pode ser reutilizada com ajustes para outros modelos e métodos também baseados na ISO/IEC 15504.

Figura-7: Tela CERTICSys - Pontuação de um Resultado Esperado

## 5. Conclusão

O CERTICSys está implementado em uma arquitetura orientada a serviços, com componentes com baixo acoplamento e com toda lógica e fluxo do processo de avaliação em uma suíte externa (BPMS). Estas escolhas arquiteturais, aliada a uma interface de usuário totalmente desacoplada do *workflow* do processo de avaliação, permitem a customização ou a adição de novos métodos de avaliação, mantendo a integridade de todo o ecossistema já desenvolvido.

## Agradecimentos

Os autores agradecem o apoio financeiro dos projetos MCTI/01200.001832/2011 - *CTENIC* e FINEP 0113009300 - *Implementação da CERTICS*, o trabalho e energia da equipe da CERTICS e dos demais colaboradores do CTI Renato Archer e Facti, e as contribuições e sugestões dos revisores anônimos do WAMPS 2014.

## Referências

Equipe CERTICS, Método de Avaliação da CERTICS - Documento de Detalhamento - Versão 1.1, Relatório Técnico CTI Renato Archer TRT0083113, 2013.

ABNT NBR ISO/IEC 15504 - Tecnologia da informação - Avaliação de processo - Parte 2: Realização de uma avaliação, ABNT, 2008.

Davi C. Silva, Alan Raldi, Thiago Messias, Angela M. Alves, Clenio F. Salviano, A Process driven software platform to full support Process Assessment Method, The Euromicro Conference on Software Engineering and Advanced Applications, 2014, pp. 135-136, DOI 10.1109/SEAA.2014.45



# Spider-QA: Uma Ferramenta de Apoio ao Processo de Garantia da Qualidade

Iuri Igonez Silva Raiol<sup>2</sup>, Luiz Otávio Danin de Lima<sup>2</sup>, Sandro Ronaldo Bezerra Oliveira<sup>1,2</sup>

<sup>1</sup>Programa de Pós-Graduação em Ciência da Computação (PPGCC) – Universidade Federal do Pará (UFPA) - Rua Augusto Corrêa, 01 – Guamá – Belém - PA - Brasil

<sup>2</sup>Faculdade da Computação – Faculdade de Computação, Instituto de Ciência Exatas e Naturais, Universidade Federal do Pará (UFPA)

{iuriraiol,otaviodanin}@gmail.com, srbo@ufpa.br

**Abstract.** *This paper presents a tool to support the Quality Assurance process, the Spider-QA, which aims to help organizations that wish to implement the Quality Management based on the MR-MPS-SW and CMMI-DEV models. The tool runs on a Web platform, which facilitates the deployment and availability of access to it. The Spider-QA facilitates this management with the use of Checklists of work products and processes that are present in a software project to be audited, with the premise supporting the planning, execution, monitoring and controlling the activities inherent to Quality Assurance process. After the development, the tool was applied in two companies, and was applied a questionnaire in order to evaluate its adherence to the MR-MPS-SW model.*

**Resumo.** *Este artigo apresenta uma ferramenta de apoio ao processo de Garantia da Qualidade, a Spider-QA, que tem como objetivo auxiliar as organizações que desejam implementar a Gerência da Qualidade com base nos modelos MR-MPS-SW e CMMI-DEV. A ferramenta funciona em uma plataforma web, o que facilita a implementação e disponibilidade de acesso. A Spider-QA facilita esta gestão com o uso de Checklists de produtos de trabalho e processos que estão presentes em um projeto de software que será auditado, tendo como premissa o apoio ao planejamento, execução, monitoramento e controle das atividades inerentes ao processo de Garantia da Qualidade. Após o seu desenvolvimento, a ferramenta foi aplicada em duas empresas, e foi utilizado um questionário para avaliar a aderência ao modelo MR-MPS-SW.*

## 1. Introdução

Na ciência da computação o principal objeto de estudo é o software, que com o passar dos anos torna-se mais necessário dentro das organizações que buscam aperfeiçoar seus processos internos com o objetivo de satisfazer seus clientes. Nesse contexto, a produção de softwares mais complexos propiciou a necessidade de se criar uma ciência que hoje é um dos pilares da tecnologia da informação, a Engenharia de Software.

Conforme o IEEE (1990), a Engenharia de Software é “(1) a aplicação de uma abordagem sistemática, disciplinada e quantificável para o desenvolvimento, operação e manutenção de um software. (2) O estudo das abordagens definidas em (1)” (tradução livre). A partir deste conceito observa-se que a engenharia de software é uma ciência exata que necessita de monitoramento e controle durante o processo de desenvolvimento de software para garantir a qualidade do produto e do processo, e como Pressman (2011) define: “A pedra fundamental que sustenta a engenharia de software é o foco na qualidade”. Nesse contexto, a garantia de qualidade em um processo de desenvolvimento de



software é fundamental para que se obtenha êxito em um projeto. A partir disso surgem modelos que definem padrões que auxiliam a criação do processo que uma equipe de desenvolvimento de software, um destes modelos faz parte do programa MPS-BR (SOFTEX, 2012), mantido pela SOFTEX – Associação para a Promoção do Software Brasileiro. “A iniciativa foi responsável pelo desenvolvimento do Modelo de Referência para Melhoria do Processo de Software Brasileiro (MR-MPS-SW), que levou em consideração normas e modelos internacionalmente reconhecidos, boas práticas da engenharia de software e as necessidades de negócio da indústria de software nacional.”

É a partir deste modelo que se observa a garantia da qualidade do processo de software como um dos pilares para a construção de um desenvolvimento de software com qualidade, efetividade e eficiência. Em meio a tudo isso é apresentado o projeto SPIDER - *Software Process Improvement – DEvelopment and Research* (Oliveira et al., 2011) institucionalizado na UFPA em 2009, que consiste na criação de um conjunto de ferramentas livres que dão apoio à implementação dos processos constantes no MR-MPS-SW. No contexto das ferramentas de apoio aos processos, neste trabalho é apresentada a Spider-QA, uma ferramenta de apoio à gerência de qualidade que busca atender os resultados esperados do processo de Garantia da Qualidade (GQA) constante no Nível F de Maturidade do MR-MPS-SW (SOFTEX, 2013). Assim, este artigo tem o objetivo geral de apresentar a ferramenta Spider-QA e descrever todo o apoio provido pela ferramenta ao atendimento das práticas constantes no processo de Garantia da Qualidade.

Além desta seção introdutória, este trabalho está estruturado em seis outras seções, a saber: a Seção 2 apresenta a ferramenta, bem como os seus principais objetivos e público alvo; na Seção 3 são apresentadas as principais funcionalidades da ferramenta Spider-QA; Seção 4 apresenta as tecnologias implementadas na aplicação, a infraestrutura necessária para a instalação e o tipo de licença da ferramenta; na Seção 5 descreve-se o apoio da ferramenta ao modelo MR-MPS-SW relativo ao processo de GQA; a Seção 6 apresenta um relato de uma avaliação da ferramenta; por fim, a Seção 7 descreve as conclusões, os principais resultados obtidos e os trabalhos futuros.

## 2. A Ferramenta Spider-QA

Visando garantir o desenvolvimento de projetos de software com qualidade, pautados nos modelos e normas de qualidade (CMMI, MPS.BR, ISO/IEC 12207, PMBOK, SWEBOK), pensou-se em um processo de Garantia da Qualidade genérico, que pudesse atender às diferentes realidades de desenvolvimento de software de empresas, ou seja, um processo de desenvolvimento que não estivesse voltado para a análise de uma organização específica, mas sim conseguisse envolver as demais normas e modelos pelos quais os projetos de desenvolvimento de software orientam-se.

Nesse intuito, orientado por esse referencial, foi elaborado o *framework* do processo de Garantia da Qualidade (Teles, 2011), que visa minimizar a adoção das práticas constantes nos diferentes modelos citados, valendo-se como base os resultados esperados do MR-MPS-SW, onde a partir dele outras normas e padrões foram comparados. Para facilitar a utilização deste *framework*, foi desenvolvida a ferramenta Spider-QA, auxiliando a implementação dos ativos de processo (atividade, artefatos, procedimentos, etc.) de Garantia da Qualidade definidos. É dentro deste contexto que o processo de desenvolvimento de um software relaciona-se com a qualidade do produto, sendo assim a ferramenta Spider-QA auxilia na melhoria do processo de produção do software de uma organização, apoiando a gerência da qualidade por meio de subsídios necessários para as revisões dos produtos e processos.

Esta ferramenta pretende objetivamente auxiliar a execução da Garantia da Qualidade e gerar os indicadores necessários para que este processo esteja sendo corretamente executado, no que diz respeito aos resultados esperados definidos no MR-MPS-SW. Sendo assim, esta ferramenta pode ser utilizada por uma organização que pretenda alcançar uma avaliação neste modelo de maturidade.

A ferramenta é gratuita, com licença GPL - *General Public License*, tornando-se uma importante vantagem no quesito custos para a empresa que pretende adotar um modelo de qualidade de software. Aliás, a organização poderá utilizá-la em conjunto com o *framework* de processo, possibilitando uma solução mais completa e adequada às organizações. Contudo, é possível uma empresa utilizar somente o *framework* do processo ou a ferramenta. Vale ressaltar que a ferramenta pode ser utilizada em diferentes organizações, independente do seu porte, podendo, ainda, ser utilizada durante a realização de consultoria que envolva a implantação das práticas do processo de Garantia da Qualidade em empresas. A ferramenta está disponível em [www.spider.ufpa.br](http://www.spider.ufpa.br).

### 3. Principais Funcionalidades

Nesta seção serão apresentadas as principais funcionalidades da Spider-QA, que conta com módulos referentes à Gerência de *Checklists*, Gerência do Projeto de Software, Plano de Qualidade, Realização de Auditorias e Plano de Ação.

#### 3.1. Gerência de *Checklists* de Produtos e Processos de Software

A Spider-QA possui uma estrutura baseada em *Checklists*, e a partir destes são criados itens que descrevem cada etapa ou parte de um **Produto de Trabalho** ou um **Processo/Disciplina**, previamente cadastrados. Estes itens compõem o *Checklist*.

A Gerência do *Checklist* é realizada por meio das funcionalidades apresentadas na Figura 1, a saber: Adicionar *Checklist*, Detalhes, Itens, Editar e Excluir. O botão "Detalhes" apresenta as informações cadastradas do *Checklist* e itens pertencentes ao *Checklist*. O botão "Itens" representa as operações de adicionar, editar e excluir os itens que compõem o *Checklist* criado.

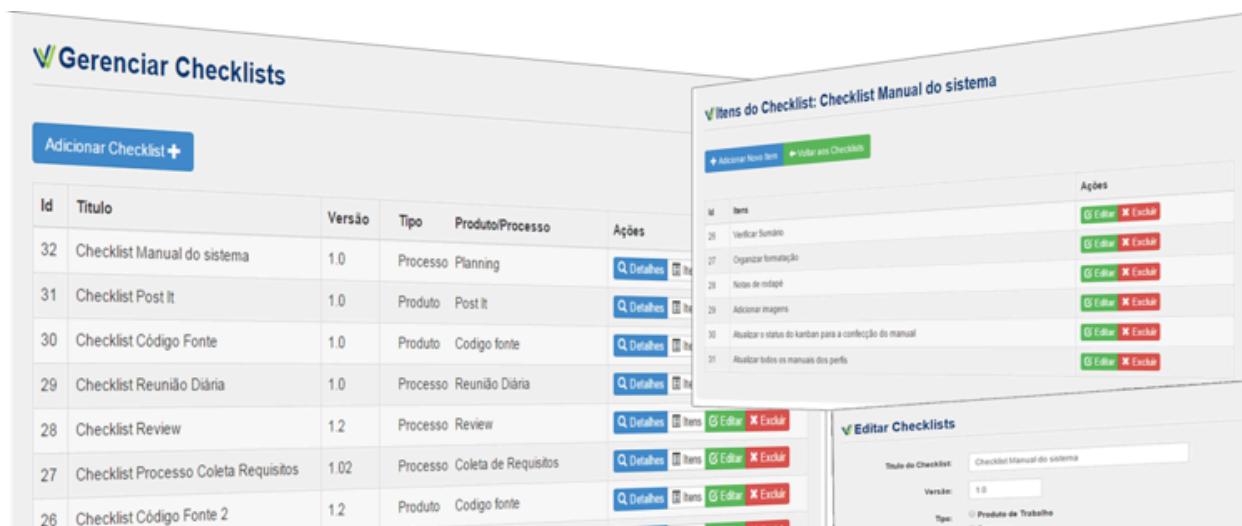


Figura 1. Gerência do *Checklist*

O *Checklist* é vinculado a um Produto de Trabalho ou Processo/Disciplina previamente cadastrado no sistema, a fim de representar um instrumento de avaliação objetiva deste produto ou processo.

### 3.2. Gerência do Projeto de Software

A Gerência do Projeto de Software é realizada por meio das ações apresentadas nas Figura 2: Adicionar Projeto, Participantes, Fases, Disciplina/Produto, Editar e Excluir. Neste módulo, o projeto de software criado receberá a execução da Garantia da Qualidade, vinculando os elementos que o compõe. Na ação "Participantes" são vinculados o perfil e nome do participante, por exemplo, João participa do projeto como Analista de Requisitos. Na ação "Fases" são vinculadas as fases e o período de duração no projeto, por exemplo, a fase de Concepção do RUP tem um período inicial e final. Na ação "Disciplina/Produto" são vinculados os Processos/Disciplinas e os Produtos de Trabalho do ciclo de vida do projeto, por exemplo, uma disciplina é a de Requisitos e um produto de trabalho seria o documento de Especificação de Requisitos. Além dessas ações é possível editar e excluir as informações do projeto registradas.

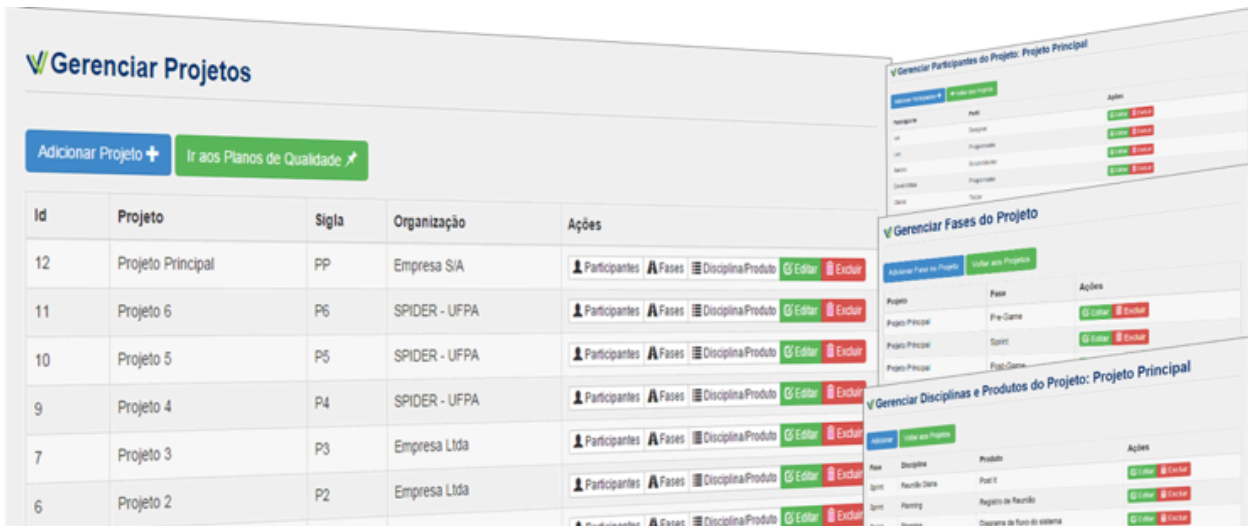


Figura 2. Gerência dos Projetos de Software

### 3.3. Plano de Qualidade

A gerência do Plano de Qualidade, vista na Figura 3, é realizada por meio da seleção de um projeto, previamente cadastrado, no qual é possível vincular a este projeto um Plano de Qualidade para orientar as ações de execução do processo de Garantia da Qualidade. O Plano de Qualidade é composto por descrição, tarefas, objetivos, detalhamento das auditorias de produto e do processo, métodos, contexto do projeto, resultados e um controle de versão. O Plano de Qualidade possibilita a inclusão de uma ou mais auditoria(s), que será descrita a seguir, na Subseção 3.4.

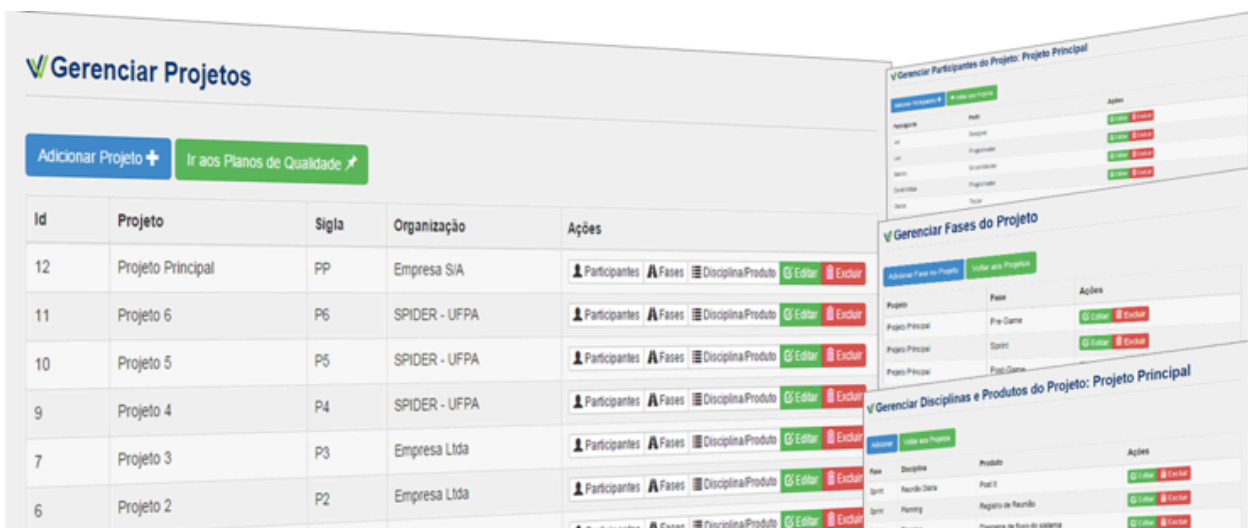


Figura 3. Gerência do Plano de Qualidade

### 3.4. Auditorias

A Gerência da Auditoria é realizada por meio da ação “Adicionar Auditoria”, vista na Figura 4. Esta ação trata da descrição de como, quando e onde irão ocorrer as auditorias registradas no Plano de Qualidade, e quais itens que compõem o *Checklist* serão usados. Estes itens dizem respeito aos Produtos de Trabalho, a exemplo o Código Fonte, e aos Processos, a exemplo o Teste de Software, vinculados anteriormente no projeto, pertencentes a esta auditoria. Neste caso, cada um destes será vinculado ao *Checklist* que servirá como base para a execução da auditoria registrada. É possível também comunicar a realização da auditoria planejada aos participantes do projeto por e-mail pela Spider-QA, possibilitando a comunicação entre todos os participantes do projeto auditado. A tela de Gerência da Auditoria pode ser observada na Figura 4.

The screenshot displays the 'Gerenciar Auditoria' (Manage Audit) interface. It features a main table listing audits and two modal windows for editing and communication.

Data Prevista	Sigla	Tipo	Produto/Processo	Ações
26/08/2014	ADT02	Processo	Reunião Diária	[Comunicar] [Editar] [Excluir]
26/08/2014	ADT03	Processo	Planning	[Comunicar] [Editar] [Excluir]
29/08/2014	ADT04	Produto	Codigo fonte	[Comunicar] [Editar] [Excluir]
11/09/2014	ADT066	Processo	Reunião Diária	[Comunicar] [Editar] [Excluir]
16/10/2014	ADT07	Processo	Planning	[Comunicar] [Editar] [Excluir]

The 'Editar Auditoria' modal form includes the following fields:

- Sigla: ADT066
- Data Prevista: 15/09/2014
- Local: UFR
- Horário: 15:30
- Tipo:  Produto de Trabalho  Processo
- Designação: Reunião Diária
- Checklist: Checklist Reunião Diária

The 'Comunicar Auditoria' modal form shows a table for selecting participants:

Nome do Participante	Nome	E-mail	Selecionar
...	...	...	<input type="checkbox"/>
...	...	...	<input type="checkbox"/>
...	...	...	<input type="checkbox"/>
...	...	...	<input type="checkbox"/>
...	...	...	<input type="checkbox"/>
...	...	...	<input type="checkbox"/>
...	...	...	<input type="checkbox"/>
...	...	...	<input type="checkbox"/>

Figura 4. Gerência da Auditoria

Após a Gerência da Auditoria, pode-se realizar a execução da auditoria nos itens do *Checklist* vinculado, como visto na Figura 5. Esta execução ocorre com a classificação dos itens auditados conforme os seguintes critérios definidos: “Contempla”, “Contempla em Partes”, “Não Contempla” e “Não se Aplica”; sendo que cada item pode conter observações detalhadas em cada um dos critérios, itens que compõem o *Checklist*. A auditoria pode ser colocada em três estados diferentes: “Novo”, “Andamento” e “Finalizada”. Após finalizada, a auditoria não pode ser executada novamente, restando somente a realização do Plano de Ação, que será discutido adiante.

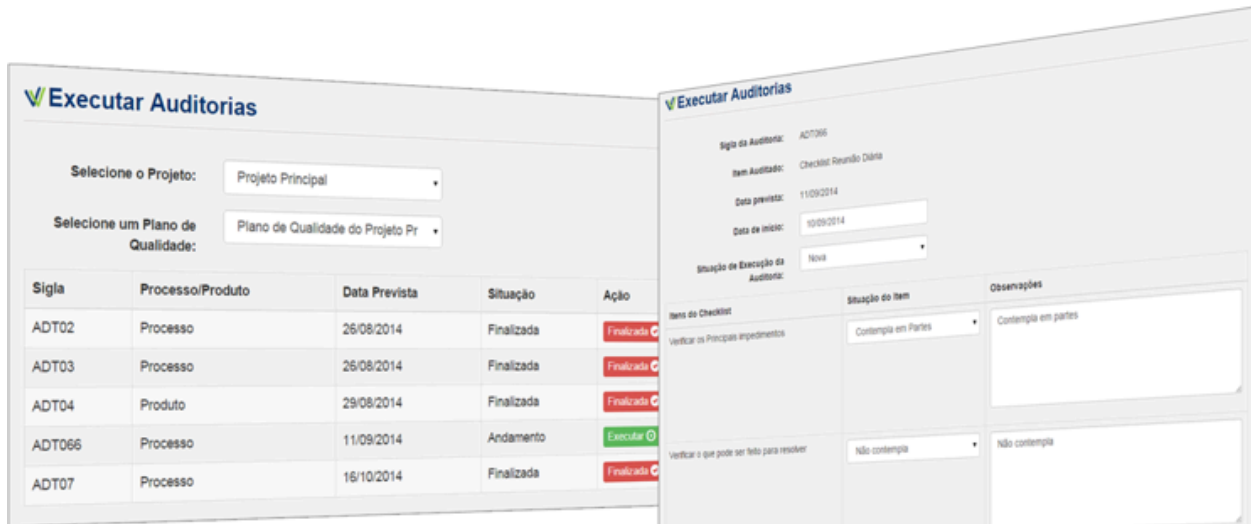


Figura 5. Execução da Auditoria

### 3.5. Plano de Ação

O Plano de Ação está diretamente relacionado a um Projeto de Software e um Plano de Qualidade, previamente definidos, podendo somente executar auditorias finalizadas. A gerência pode ser realizada por meio das operações: “Adicionar Plano de Ação”, “Itens”, “Executar”, “Ações” e “Comunicar”, conforme observado na Figura 6.

O plano pode ser adicionado após a vinculação deste a uma auditoria finalizada, definindo uma data de início e término da realização do plano e o controle das versões. Após este registro, pode-se “Executar” o Plano de Ação, sendo necessária a vinculação de um ou mais itens auditados a partir do registro de uma ação de resolução para as não conformidades coletadas na auditoria, definindo um responsável, prazo e o item da ação definido na ferramenta de *bugTracking*. A integração com a ferramenta de *bugTracking* independe da ferramenta utilizada, pois previamente é cadastrado o *link* de acesso à ferramenta no projeto o qual este plano de ação está vinculado. A necessidade do registro deste item dá-se para o acompanhamento da resolução das não conformidades a partir de uma máquina de estados.

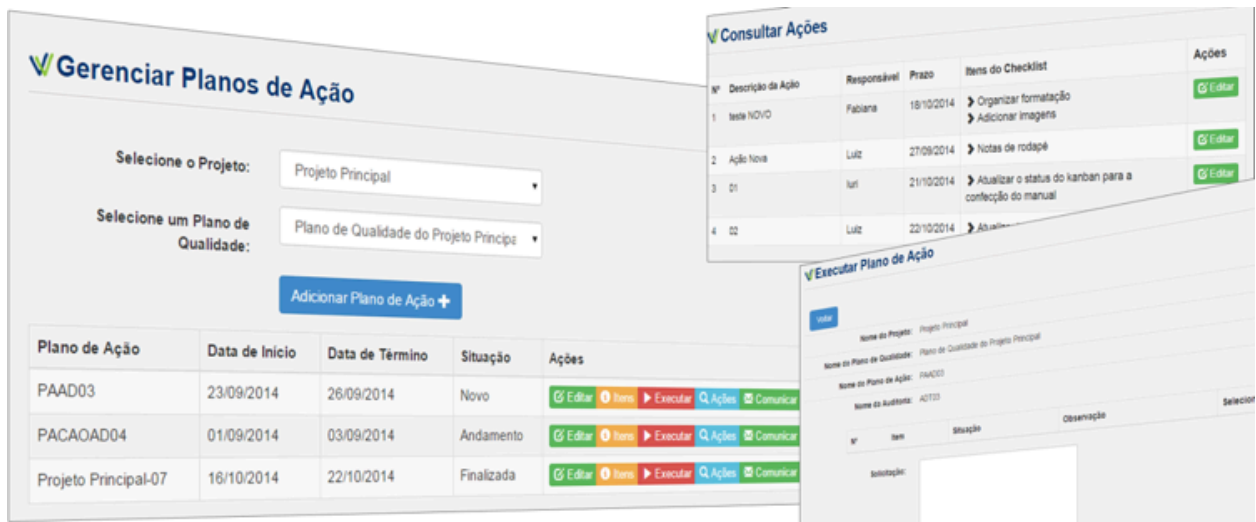


Figura 6. Gerência do Plano de Ação

## 4. Infraestrutura para Instalação e Uso

Para que houvesse maior facilidade de utilização da ferramenta, optou-se em realizar a implementação da mesma em uma plataforma *Web*, sendo seu desenvolvimento pautado no uso de tecnologias de software livre sob licenças *open source*, como a linguagem de programação PHP 5.4, facilitada pela utilização do *framework* de desenvolvimento CakePHP versão 2.4.6, o SGBD MySQL 5.6.12 e o servidor de aplicação Apache 2.4.4. Para a correta operação da ferramenta faz-se necessária a instalação do PHP com versão 5.4 ou superior, o SGBD MySQL 5.6.12 e o servidor de aplicação Apache 2.4. Também, é necessária a habilitação do módulo *mod\_rewrite* do Apache, bem como a extensão do PHP *openssl* para envio de e-mail de forma segura.

## 5. Descrição do Apoio ao Modelo MR-MPS-SW e CMMI-DEV

A ferramenta Spider-QA apresenta os seguintes apoios aos resultados esperados do processo de Garantia da Qualidade constante no MR-MPS-SW e na área de processo *Process and Product Quality Assurance* (PPQA) constante no CMMI-DEV, a saber:

- **GQA1 e PPQA SP1.2:** atendidos pelos módulos de Gerência de *Checklists*, Gerência do Projeto de Software e Plano de Qualidade a partir da execução das operações Definição do Plano de Qualidade, Cronograma de Auditorias, Aprovação do Plano de Qualidade, Preparação da Auditoria, Seleção do Material da Auditoria e Execução da Auditoria;
- **GQA2 e PPQA SP1.1:** atendido pelos módulos de Gerência de *Checklists*, Gerência do Projeto de Software e Plano de Qualidade a partir da execução das operações Definição do Plano de Qualidade, Cronograma de Auditorias; Aprovação do Plano de Qualidade, Preparação da Auditoria, Seleção do Material da Auditoria, Seleção de Participantes para a Auditoria e Execução da Auditoria;

- **GQA3 e PPQA SP2.1 e SP2.2:** atendido pelo módulo de Realização de Auditorias a partir da execução das operações Execução da Auditoria e Notificação das Não Conformidades;
- **GQA4 e PPQA SP2.1:** atendido pelo módulo de Plano de Ação a partir da execução das operações Estabelecimento das Datas e Responsáveis para a Correção, Correção das Não Conformidades, Verificação das Correções, Escalonamento das Não Conformidades para a Gerência de Alto Nível, Finalização das Não Conformidades e Notificação dos Resultados.

Importante destacar que o apoio ao acompanhamento da resolução das não conformidades é realizada por qualquer ferramenta de *bugtracking* (Redmine, Bugzila, Mantis, Trac) integrada à ferramenta Spider-QA, ficando sob a responsabilidade desta última a associação dos itens registrados nas ferramentas de *bugtracking* às ações cadastradas no Plano de Ação da auditoria.

## 6. Avaliação da Ferramenta

Para avaliar a ferramenta, foram escolhidas 2 empresas de Belém avaliadas no Nível F de maturidade do MR-MPS-SW, onde foram feitas demonstrações da ferramenta e questionários de avaliação foram distribuídos. Assim, pretendeu-se analisar se a ferramenta estava aderente ao MR-MPS-SW, verificar a usabilidade da ferramenta e se ela poderia de fato atender as necessidades da organização e ser utilizada no seu dia a dia.

Para essa análise, foram utilizados 20 questionários sendo 2 com pessoas que desempenham o papel de Analistas da Qualidade e os demais com pessoas que conhecem a gerência, mas não costumam desempenhar esse papel. Todos os entrevistados têm de 1 a 2 anos de conhecimento em modelos de qualidade e têm algum curso oficial do MPS.BR. O questionário avaliou o perfil do entrevistado e se este já havia utilizado alguma ferramenta com função similar. Depois avaliou-se o uso da ferramenta, como pode ser visualizado em forma de gráfico na Figura 6, e se a ferramenta estava em conformidade com o MR-MPS-SW.

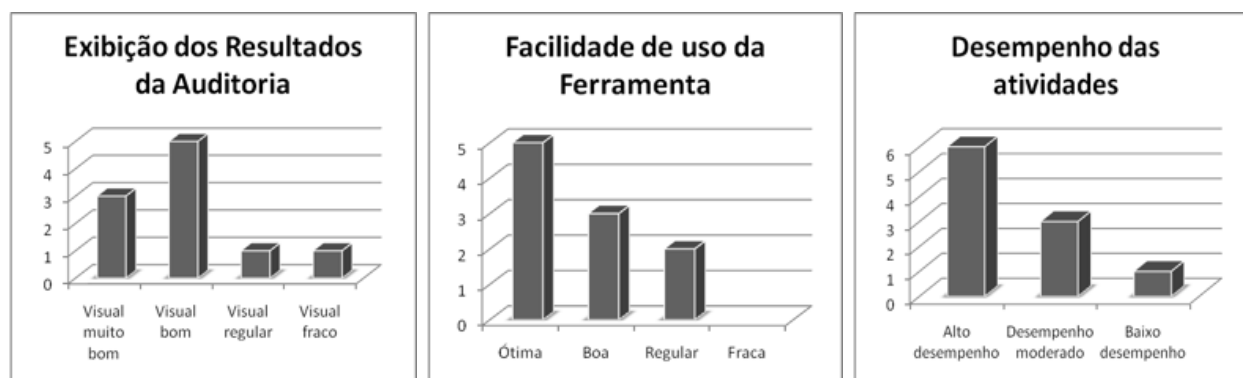


Figura 6. Gráficos dos Resultados da Aplicação do Questionário

Quanto à ferramenta aderir ao modelo MR-MPS-SW, foi afirmado que a ferramenta está aderente juntamente com outras ferramentas. Quando perguntado qual a funcionalidade que estaria faltando para que a ferramenta fosse totalmente aderente ao MPS.BR, foi mencionado que seria o registro de não-conformidades, visto que este registro é feito de maneira integrada a uma ferramenta de *bugtracking* e não na própria Spider-QA, apesar da ferramenta manter todo o controle de acompanhamento destas não conformidades registradas.



A avaliação da ferramenta, no geral, foi muito boa. Os entrevistados deram sua opinião em um campo de observações, onde foram encontradas idéias que deverão ser implementadas, como por exemplo: customizar relatórios para atender ao padrão organizacional; e integrar a ferramenta a um repositório de arquivos (CVS, Subversion), a fim de possibilitar a visualização dos artefatos durante a auditoria. Melhorias foram sugeridas na interface gráfica e se acredita que a ferramenta pode ser utilizada em uma revisão real, e que ela facilitaria a implementação da GQA.

## 7. Considerações Finais

Em síntese a ferramenta Spider-QA objetiva auxiliar o processo de Garantia da Qualidade em projetos de desenvolvimento de software, principalmente em organizações de pequeno e médio porte com base no MR-MPS-SW, estando também em conformidade com outros modelos ou normas, como o CMMI, ISO/IEC 12207, o PMBOK e o SWEBOK. A ferramenta atua como facilitadora dando apoio à execução do *framework* de processo de Garantia da Qualidade proposta em (Teles, 2011), elaborado para apoiar a gerência aderente aos resultados esperados do MR-MPS-SW, gerando assim indicadores de que a gerência está sendo executada de forma correta. A utilização desta ferramenta além de não gerar custos financeiros às organizações, por ser um software livre, pode proporcionar uma gestão de projetos de software com base em modelos de qualidade eficazes e eficientes a um programa de melhoria de processo de software organizacional.

Como trabalhos futuros espera-se realizar algumas integrações funcionais a outros processos constantes no MR-MPS-SW, a saber: Gerência de Configuração, a fim de manter e disponibilizar as várias versões dos produtos de trabalho e processos desenvolvidos em uma organização; Gerência de Projetos, possibilitando consultas às informações relacionadas ao projeto como cronograma e recursos humanos alocados diretamente a partir da ferramenta Spider-QA; Gestão de Decisão, facilitando a tomada de decisão de quais processos e produtos de trabalhos serão selecionados para realizar auditorias por amostragem.

## Referências

- IEEE (1990) "IEEE Standard Glossary of Software Engineering Terminology". IEEE Standards Board.
- Oliveira, S. R. B. *et al.* (2011) "SPIDER – Uma Proposta de Solução Sistêmica de um SUITE de Ferramentas de Software Livre de Apoio à Implementação do Modelo MPS.BR". Revista do Programa Brasileiro da Qualidade e Produtividade em Software, SEPIN-MCT. 2ª Edição. Brasília-DF.
- Pressman, S. R. (2011) "Engenharia de Software, uma Abordagem Profissional". AMGH editora Ltda. 7ª edição.
- SOFTEX (2012). "Guia Geral do MR-MPS-SW". Disponível em: [www.softex.br/mpsbr](http://www.softex.br/mpsbr). Acesso em 02/10/2014.
- SOFTEX (2013) "Guia de Implementação Parte 2 - Nível F do MR-MPS-SW". Disponível em: [www.softex.br/mpsbr](http://www.softex.br/mpsbr). Acesso em 02/10/2014.
- Teles, M. P. (2011) "SPIDER-QA: Um Ferramental de Apoio ao Processo de Garantia da Qualidade no Contexto de Modelos e Norma para Processo de Software". Dissertação de Mestrado do Programa de Pós-Graduação em Ciências da Computação. Universidade Federal do Pará.

# **MTControl: Ferramenta de Apoio à Gestão de Testes de Aplicativos Móveis Baseada nas Diretrizes do AQuA**

Juliana P. do Nascimento, Jonathas S. dos Santos, Arilo C. Dias-Neto

Instituto de Computação – Universidade Federal do Amazonas (UFAM)

Caixa Postal nº 6.200 – 69.077– 000 – Manaus – AM – Brasil

{jpn, jss, arilo}@icomp.ufam.edu.br

**Abstract.** *Currently, ensuring the quality of mobile applications has become essential due to the high complexity and importance that they have assumed. In this context, AQuA alliance proposes quality criteria to evaluate mobile applications in different platforms. Thus, developers should to apply them using process management concepts in order to evaluate the quality of mobile applications. This work aims to present the MTControl tool, which aims to suggest and manage testing criteria for multiplatform mobile applications according to their characteristics. The original recommendations proposed by the tool are published as guidelines by the AQuA alliance.*

**Resumo.** *Atualmente, garantir a qualidade de aplicativos móveis se tornou essencial por conta da alta complexidade e importância que eles têm assumido em nosso dia-a-dia. Neste contexto, a aliança AQuA propõe critérios de qualidade para avaliação de aplicativos móveis para diferentes plataformas. Assim, cabe aos desenvolvedores aplicá-los utilizando os conceitos de gestão de processos de software para avaliação da qualidade de aplicativos móveis. Este trabalho tem como objetivo apresentar as funcionalidades da ferramenta MTControl, que tem como objetivo recomendar e gerenciar diretrizes de qualidade para aplicativos móveis de diferentes plataformas das suas características. As recomendações originais disponibilizadas pela ferramenta são prescritas utilizando as diretrizes publicadas pela aliança AQuA.*

## **1. Introdução**

A construção de aplicações para dispositivos móveis, ou simplesmente *apps*, de qualidade e que atendam a todas as necessidades dos usuários já deixou de ser um diferencial no mercado e tornou-se uma exigência (Harisson et al., 2013). Uma das formas de melhorar a qualidade das *apps* é por meio da aplicação de técnicas de teste de software.

Segundo Dantas et al. (2009), a complexidade para se testar uma *app* exige uma adaptação no processo de teste tradicional, decorrente do fato de que não é possível executar testes da mesma maneira que se realiza em outros tipos de aplicações por conta das características específicas dessa plataforma, principalmente em relação à simulação exata do ambiente onde tal software irá funcionar. Isso acontece devido às limitações de recursos existentes nestes dispositivos e a necessidade de interação das *apps* com os demais serviços providos pelo dispositivo, como por exemplo atender chamadas, oscilação de carga da bateria, comunicação por redes *wi-fi*, *bluetooth*, acesso à câmera fotográfica, dentre outros, o que não pode ser simulado totalmente pelos emuladores providos para cada plataforma (Janicki et al., 2012).

Em 2012, surgiu a antiga *Unified Testing Initiative* alterou seu nome para *App Quality Alliance* (AQuA) com o objetivo de reforçar a importância da qualidade de software, direcionando seus esforços para a plataforma móvel. AQuA é uma associação fundada por membros de várias empresas de TI (ex: AT&T, LG, Motorola, Nokia, Oracle, Orange, Samsung e Sony Mobile) com o propósito de apoiar desenvolvedores a melhorar a qualidade de *apps*. As principais iniciativas providas pela AQuA são: diretrizes com melhores práticas para desenvolvedores, que visam a inserção de práticas de qualidade durante o desenvolvimento como forma de reduzir o esforço com testes, e definição e publicação de critérios de teste para diferentes plataformas, a fim de ajudar desenvolvedores e testadores a prepararem suas *apps* para publicação.

Apesar dos critérios de teste providos pela AQuA, a gestão dos testes no desenvolvimento de uma *app* ainda se caracteriza como um elemento importante para o sucesso dos testes (Santos et al., 2012). AQuA oferece uma ferramenta *online* que apoia testadores gerenciarem o progresso dos testes em suas *apps* em relação aos critérios de teste definidos para a sua plataforma. No entanto, este serviço possui alguns limitadores, tais como: funcionalidades limitadas para filtro ou personalização dos testes para cada categoria de *app*, ausência de suporte a testes de compatibilidade de uma mesma *app* em diferentes plataformas, ausência de suporte a reuso de testes de projetos/rodadas de testes anteriores. Assim, o foco deste artigo é a descrição de uma nova ferramenta, chamada *MTControl*, que visa prover um apoio mais abrangente à gestão dos testes em *apps*, reutilizando os critérios de teste providos no repositório AQuA e estendendo as funcionalidades providas na ferramenta online fornecida por esta associação. Neste caso, a ferramenta proposta visa complementar a gestão das atividades do processo de testes (que em geral inclui a seleção de recursos humanos, definição de cronograma ou custo dos testes), atendida em ferramentas genéricas de gestão de testes como TestLink (Testlink, 2014), provendo um apoio à gestão das decisões técnicas relacionadas a testes de *apps*, incluindo filtro e seleção dos critérios de teste a serem aplicados de acordo com a plataforma da *app* sob teste e controle dos testes de *apps* em diferentes versões e plataformas, embasada pelas propriedades e diretrizes apresentadas pela AQuA.

O restante do artigo está organizado da seguinte forma. Na seção 2 é descrito o processo de gestão de testes utilizado atualmente pela Aqua. Na seção 3, são relatados os objetivos e funcionalidades embasados na *MTControl*, bem como seu funcionamento. Na seção 4, as considerações finais são descritas.

## **2. Gestão de Testes para Aplicativos Móveis: *App Quality Alliance* – AQuA**

Em (Santos et al., 2012), os autores identificaram desafios, diversidades e ferramentas aplicáveis no contexto de teste para *apps*. A gestão destes testes é caracterizada nestes trabalhos como um desafio, visto que a complexidade para se testar uma *app* exige uma adaptação no processo de teste tradicional, decorrente das limitações já conhecidas dos dispositivos móveis. Analisando as ferramentas, neste estudo observou-se muita ênfase em ferramentas de apoio à geração e/ou execução de casos de teste, com foco no aspecto técnico, e pouco apoio às tomadas de decisões que compõem o aspecto gerencial.

Neste cenário, a *App Quality Alliance* (AQuA) surgiu com uma importante iniciativa de apoio à gestão de testes em *apps*. AQuA é uma associação global com foco em apoiar a indústria a continuamente **melhorar e promover qualidade de apps**, aonde diversas empresas trabalham juntas em projetos de interesse mútuos minimizando, assim, o trabalho a ser feito por cada um. AQuA também atua como um órgão de referência e avaliação, creditando a qualidade de desenvolvedores e prestadores de serviços de teste que estão no mercado, e também suas *apps*. Esta ferramenta de critérios de teste online (*Online Testing Criteria Tool*) é disponibilizada para apoiar testadores a aplicarem os critérios de teste AQuA e avaliarem os resultados para *apps* desenvolvidas, atualmente, para as plataformas Android e iOS.

Informando alguns o tipo de teste a ser aplicado na *apps* (dentre 4 opções disponibilizadas pela ferramenta), a ferramenta apresenta os testes relevantes (Figura 1– Esquerda). Então, à medida em que os testes são executados e registrados na ferramenta, ela vai pontuando a *app* (Figura 1– Direita). Ao concluir os testes, os resultados são apresentados (aprovado ou rejeitado) e um relatório é criado.



Figura 1. (Esquerda) Caracterizando o aplicativo a ser testado e (Direita) Acompanhamento da execução dos testes.

Uma limitação observada é que a ferramenta recomenda para qualquer *app* todos os casos de teste que compõem o critério definido para uma plataforma, mesmo quando suas características não se aplicam a esses cenários (ex: a *app* não requer conexão com Internet, mas ainda assim os testes de conectividade são recomendados). Outra limitação observada é que a ferramenta aceita apenas uma rodada de testes por *app*. Assim, caso uma *app* já testada sofra uma evolução, a ferramenta apenas oferece a opção de realizar um novo teste para uma nova *app*, sem vincular testes anteriores ao

mesmo *app*. De forma similar, a ferramenta não oferece suporte para testes em *apps* multiplataforma, visto que ela considera a versão de cada plataforma como uma nova *app*.

Assim, com o objetivo de prover uma evolução da ferramenta online proposta pela AQuA em relação às limitações destacadas acima, este artigo apresenta uma nova ferramenta de apoio à gestão de testes para *apps*, chamada *MTControl* (*Mobile Testing CONTROL tOOL*), descrita na próxima seção.

### 3. A Ferramenta *MTControl*

#### 3.1. Objetivo e Funcionalidades

A *MTControl* visa prover um apoio mais abrangente para a gestão dos testes realizados em aplicativos móveis em relação às funcionalidades atuais providas na ferramenta online de critério de teste da AQuA. Seus objetivos principais são a recomendação de testes mais adequados às características, funcionalidades e plataformas de uma *app*, utilizando como base os critérios de testes de cada plataforma publicada pela AQuA, além de prover apoio para o controle e a reutilização de testes entre diferentes versões de uma mesma *app* (em uma mesma plataforma ou multiplataforma).

Com base nesses objetivos, as principais funcionalidades providas pela *MTControl* são:

**Gerenciar Usuários:** CRUD<sup>1</sup> de usuários que possuem acesso à ferramenta com diferentes perfis (administradores [acesso irrestrito à ferramenta] ou testadores [acesso restrito à gestão de suas *apps*]);

- **Gerenciar Plataformas Móveis:** CRUD de plataformas móveis utilizadas pelos usuários da ferramenta;

**Gerenciar Critérios de Teste:** CRUD de critérios de teste (por plataforma) de *apps* publicados (ou não) pela AQuA. Critérios de testes independentes de plataforma podem ser adicionados para uso na *MTControl*. Os *critérios* de teste são compostos por um conjunto de casos de teste adequados a cada cenário/característica de uma *app*;

**Cadastrar Apps:** criação de uma nova *App* a ser ter seus testes gerenciados pela *MTControl*. Ao cadastrar uma *app*, suas características devem ser detalhadas a fim de auxiliar na recomendação dos testes mais adequados;

**Criar Rodada de Teste:** criação de uma nova instância de teste para um *app* já gerenciado pela ferramenta. Isso pode ser feito a cada nova versão ou plataforma do aplicativo. Neste momento, a ferramenta analisa as características e plataformas da *app* para selecionar quais casos de teste dos critérios cadastrados na ferramenta que se adequam à *app*. Caso não seja a primeira rodada de teste de uma *app*, a ferramenta oferece a possibilidade de o usuário selecionar os mesmos testes realizados em versões anteriores da *app*;

**Controlar Testes:** uma vez instanciada uma nova rodada de teste, *MTControl* oferece uma funcionalidade de registro e controle da execução dos testes no aplicativo, de forma a visualizar o status atual dos testes (em percentual) em geral ou por característica da *app*. Ao final, um relatório é provido com o resultado dos testes realizados na *app*;

<sup>1</sup> Acrônimo em inglês referente às operações de *Create-Retrieval-Update-Delete*.

- **Visualizar histórico:** o usuário pode visualizar a qualquer momento os testes em andamento ou já realizados em uma *app* por meio de relatórios e gráficos de acompanhamento. A ferramenta possibilita, ainda, a comparação de resultados entre diferentes rodadas de testes;

### 3.2. Funcionamento da Ferramenta

*MTControl* é desenvolvida como um sistema Web e no idioma inglês, de modo a facilitar o uso por usuários de instituições brasileiras ou estrangeiras. Funcionará como uma ferramenta gerencial de testes, então ela não visa prover apoio neste momento à execução dos testes. A seguir, é descrito o processo principal de uso da ferramenta, composto pelos seguintes passos:

1. **Cadastro de uma *app*.** Como entrada de dados, a ferramenta cadastra o aplicativo a ser testado, e todas as características da plataforma que ele foi implementado (**Figura 2** – esquerda). Essa escolha de características será implementada por meio de questionários técnicos (**Figura 2** – direita).
2. **Seleção e acompanhamento dos testes.** Quando o cadastro das características de uma *app* é finalizado e uma nova rodada de teste é instanciada, a ferramenta mostra ao usuário apenas os testes que devem ser executados para as funcionalidades informadas, diferentemente do que ocorre na ferramenta provida pela AQuA (primeiro box da **Figura 3**). À medida que escolhe uma categoria a ser testada, a ferramenta exibe os casos de testes que compõem o critério de teste para a plataforma da *app* (na **Figura 3**, a categoria *Connectivity* foi selecionada e no momento está sendo registrado o caso de teste *HTTP: Send/Receive Data*). Assim, conforme os testes são informados como Aprovados (*Pass*) ou Reprovados (*Fail*), a porcentagem de conclusão dos testes da *app* aumenta na análise geral (box esquerdo da **Figura 3**) e por categoria (label à esquerda do nome da categoria que está sendo testada). A ferramenta ainda controla os casos de testes, indicando quais passaram, falharam ou ainda não foram executados (segundo box da **Figura 3**), exibindo um gráfico em barras analisando os testes por categoria.

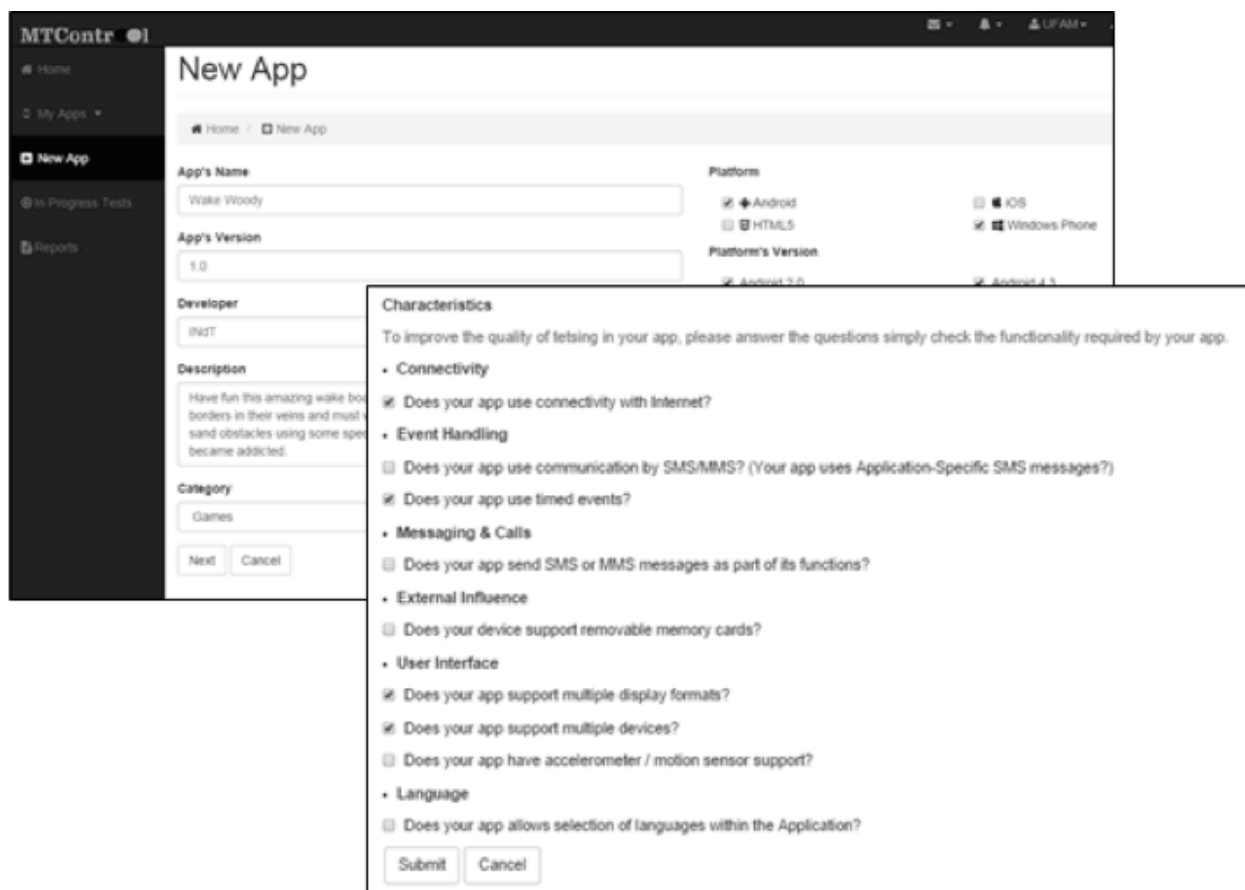


Figura 2. (Esquerda) Tela de Cadastro e (Direita) de Caracterização de App.

3. **Visualização dos testes de uma app.** Outro diferencial provido pela MTControl é o apoio a testes em *apps* multiplataforma ou o controle do versionamento dos testes em uma *app*, sem a necessidade de criar outro cadastro para a *app*. Isso pode ser observado na opção *Manage Previous Test Rounds* na **Figura 3**.

A ferramenta *MTControl* ainda está em desenvolvimento e avaliação em estudos de caso na indústria de software. Os interessados em utilizá-la para analisar os detalhes de suas funcionalidades podem entrar em contato com os autores deste trabalho.

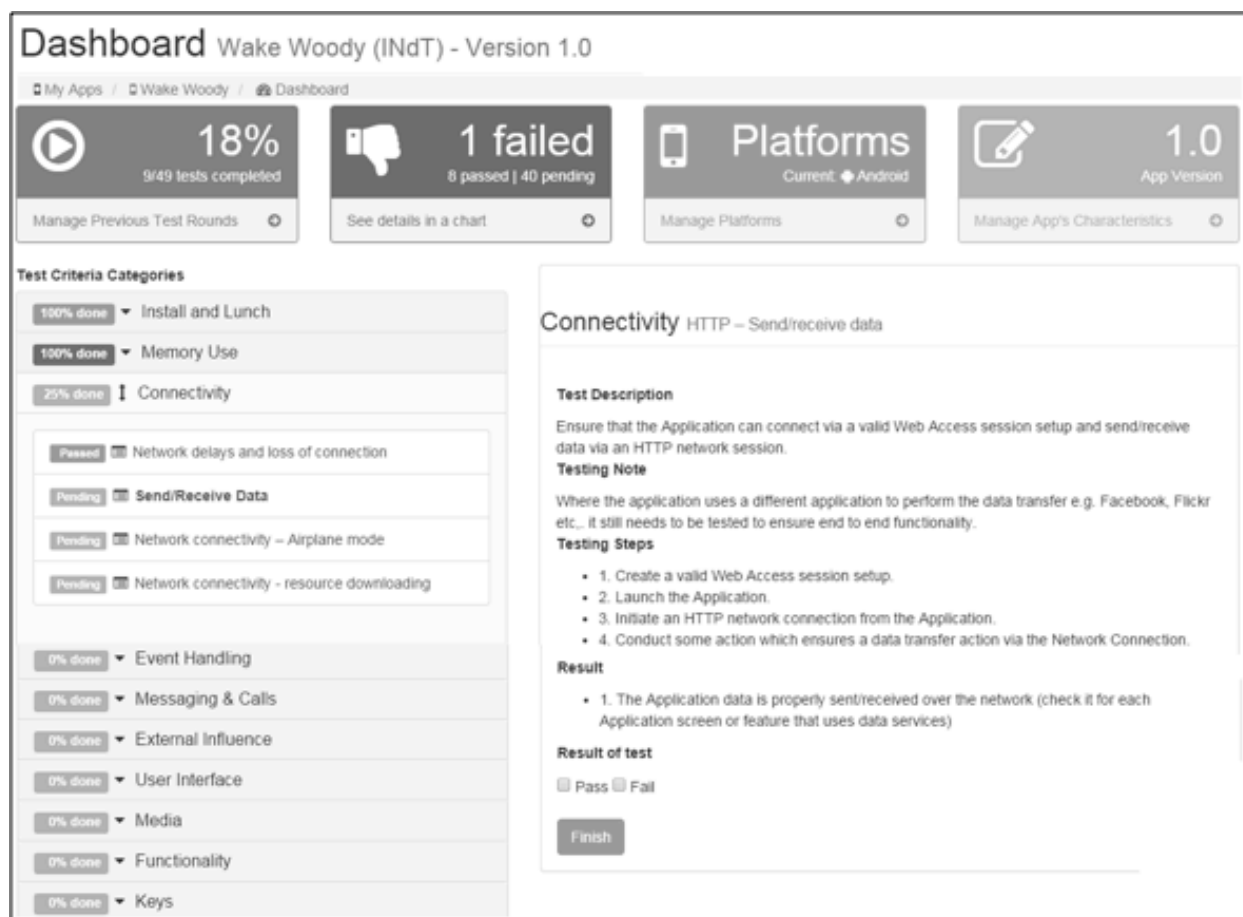


Figura 3. Tela de Controle dos Testes.

## 4. Conclusões e Trabalhos Futuros

A ferramenta *MTControl* apresenta diversas funcionalidades importantes para a gestão de testes em *apps* utilizando como base os critérios de teste para cada plataforma provido pela AQuA. Ela consiste em uma evolução da ferramenta online provida pela AQuA. Espera-se que com esse novo apoio ferramental, o gerenciamento dos testes seja realizado de forma eficiente por empresas e desenvolvedores de aplicativos móveis.

Como próximo passo deste trabalho, está prevista a realização de estudos de caso para a aplicação da ferramenta em uma empresa que desenvolve *apps* com a qual o grupo de pesquisa desenvolvedor da ferramenta possui projeto de cooperação, e como consequência, o relato desta experiência por meio de novos artigos técnico-científicos.



## Referências

- Dantas, V. L.L.; Marinho, F. G.; Costa, A. L.; Rossana M. C. Andrade (2009). Testing Requirements for Mobile Applications. In Proceedings of the 24th International Symposium on Computer and Information Sciences. Guzelyurt: IEEE, p. 555-560.
- Harrison, R.; Flood, D.; Duce, D.; Usability of mobile applications: literature review and rationale for a new usability model. *Journal of Interaction Science*, 2013, pp. 1-16.
- Global mobile statistics 2011, <http://mobithinking.com/mobilemarketing-tools/latest-mobile-stats> acessado em 02 de Outubro de 2014.
- The AQuA Deliverables, <http://www.appqualityalliance.org/the-deliverables> acessado em 30 de outubro de 2014.
- Janicki, M.; Katara, M.; Paakkonen, T. Obstacles and opportunities in deploying model-based GUI testing of mobile software: a survey, *Software Testing, Verification and Reliability*, 22, 5, 2012: 313-341.
- Santos, I. S.; Dantas, V. L. L.; Santos, R. M.; Andrade, R. M. C. . Testes de Aplicações Móveis: Uma Análise das Pesquisas Científicas via Revisão Sistemática. In: XI Simpósio Brasileiro de Qualidade de Software, 2012, Fortaleza, CE.
- TestLink Open Source Test Management; <http://testlink.org/>; Acessado em 02/11/2014.

# RAFTool - Ferramenta de Filtragem de Métodos, Classes e Pacotes com Medições Incomuns de Métricas de Software

Tarcísio G. S. Filó<sup>1</sup>, Mariza A. S. Bigonha<sup>1</sup>, Kecia Aline Marques Ferreira<sup>2</sup>

<sup>1</sup>Departamento de Ciência da Computação - Universidade Federal de Minas Gerais (UFMG) - Belo Horizonte - MG - Brasil

<sup>2</sup>Departamento de Computação - Centro Federal de Educação Tecnológica de Minas Gerais (CEFET-MG) - Belo Horizonte - MG - Brasil

{tfilo, mariza}@dcc.ufmg.br, kecia@decom.cefet.mg.br

**Resumo.** Este artigo apresenta a ferramenta RAFTool (*Risk Artifacts Filter Tool*), cujo propósito é realizar a filtragem de métodos, classes e pacotes que possuem medições anômalas de métricas de softwares orientados por objetos. Essa ferramenta apoia a realização de análise de medições coletadas sobre as estruturas internas do código-fonte em softwares orientados por objeto, podendo ser utilizada em Processos de Medição - MED do nível F (Gerenciado) do MR-MPS-SW. Uma das consequências da sua efetiva aplicação para as organizações é transformar dados de métricas de código-fonte em informações úteis para apoiar decisões relacionadas ao controle e avaliação da qualidade interna de software dentro da organização. RAFTool foi desenvolvida em Java e é uma ferramenta open source.

**Palavras-chave:** métricas, valores referência, qualide de software, medição de software, análise, apoio à decisão.

## 1. Introdução

Este artigo descreve a ferramenta RAFTool (*Risk Artifacts Filter Tool*), cujo propósito é realizar a filtragem de métodos, classes e pacotes que possuam medições anômalas de métricas de softwares orientados por objetos, no contexto do processo de medição de software. Medições anômalas são aquelas que se afastam significativamente do que é comum, podendo indicar problemas de qualidade no artefato medido [Sommerville 2010]. Um processo de medição que pode fazer parte de um processo de controle de qualidade de software é exibido na Figura 1. Um estágio chave nesse processo é a identificação de medições anômalas, que é feita por meio da comparação das medidas obtidas com um histórico de medições de projetos anteriores, com o objetivo de identificar valores que estejam fora da normalidade. Os valores referência propostos por Filó *et al.* (2014) são capazes de indicar o que são esses valores “fora dos limites definidos como desejáveis” para as métricas. Isso facilita a aplicação desse processo de controle, afinal, não é prática comum das organizações a manutenção de um banco de dados de medições de projetos anteriores, como idealiza Sommerville (2010).

Além da definição de valores referência ser de suma importância para a efetiva aplicação das métricas na indústria de software, o fornecimento de ferramentas apropriadas de medição e análise quantitativa de software é fundamental, pois de acordo com Sommerville (2010), o processo de

medição deve ser simples e fácil. Para a medição de softwares orientados por objetos, existem ferramentas que têm sido utilizadas em pesquisas relacionadas a métricas de softwares, tais como *Eclipse Metrics Plugin*<sup>1</sup> e *Google CodePro AnalytiX*<sup>2</sup>. Essas ferramentas realizam a medição de métricas de software, porém, não fornecem meios para realizar uma análise quantitativa dos resultados da medição. Nesse contexto, foi desenvolvida a ferramenta *RAFTool*, que suporta, baseada em uma entrada de medições de software em formato XML, a identificação de métodos, classes e pacotes que apresentam medições anômalas de métricas de softwares orientados por objetos. *RAFTool* é um complemento do trabalho de Filó *et al.* (2014), que propuseram e aplicaram um método para identificar valores referência para um conjunto de métricas de softwares orientados por objetos. O restante desse artigo está organizado da seguinte forma: Seção 2 apresenta os valores referência para métricas de software utilizados por *RAFTool*. Seção 3 descreve o apoio de *RAFTool* às atividades do modelo MR-MPS-SW. Seção 4 apresenta os requisitos e funcionalidades da ferramenta. Seção 5 mostra a arquitetura utilizada no desenvolvimento de *RAFTool*. Seção 6 traz a conclusão, analisando as aplicações da ferramenta e extensões futuras.

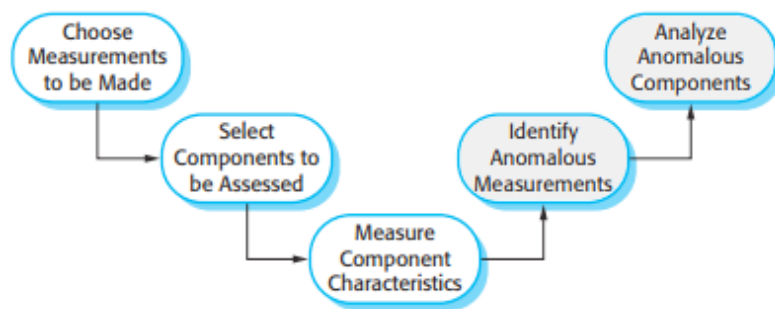


Figure 1. Processo de Medição. [Sommerville 2010].

## 2. Valores Referência

Filó *et al.* (2014) propõem valores referência para um conjunto de métricas de softwares orientados por objetos por meio de três faixas de valores: *Bom/Frequente*, *Regular/Ocasional* e *Ruim/Raro*. A faixa *Bom/Frequente* corresponde a valores com alta frequência, caracterizando os valores mais comuns da métrica, na prática. A faixa *Ruim/Raro* corresponde a valores com baixa frequência, e a faixa *Regular/Ocasional* é intermediária, correspondendo a valores que não são nem muito frequentes nem raros. Por exemplo, para a métrica *Número de Métodos* (NOM) têm-se os seguintes valores: *Bom/Frequente* ( $NOM \leq 6$ ), *Regular/Ocasional* ( $6 < NOM \leq 14$ ) e *Ruim/Raro* ( $NOM > 14$ ). Os valores referência não expressam necessariamente as melhores práticas da Engenharia de Software, e sim um padrão seguido pela maioria dos softwares. Com a utilização dos valores referência desse trabalho de pesquisa, *RAFTool* irá informar ao usuário quais são os artefatos em risco baseado em aspectos quantitativos do código-fonte por meio da combinação dos valores referência de uma ou mais métricas.

1 <http://metrics2.sourceforge.net/>

2 <https://developers.google.com/java-dev-tools/codepro/doc/>

### 3. O Processo de Medição no MPS.BR

O propósito do Processo de Medição - MED do Nível F (Gerenciado) do MR-MPS-SW é coletar, armazenar, analisar e relatar os dados relativos aos produtos desenvolvidos e aos processos implementados na organização e em seus projetos, de forma a apoiar os objetivos organizacionais [Softex 2012]. *RAFTool* é uma ferramenta que apoia a análise dos dados coletados dentro do Processo de Medição, permitindo gerenciar aspectos essenciais da qualidade interna do código em softwares orientados por objetos, identificando métodos, classes e pacotes que mostrem alguma ameaça à qualidade interna do software. Dessa forma, *RAFTool* proporciona uma real conexão entre a coleta de dados e a sua utilização para apoiar decisões dentro da organização, permitindo que o MED implementado tenha impactos positivos nas organizações.

### 4. Requisitos

Nesta seção são especificados os requisitos funcionais de *RAFTool*.

#### 4.1. Instanciação do Sistema Analisado

*RAFTool* permite a instanciação de um sistema a ser analisado a partir da digitação de um nome. Posteriormente, ela permite a importação de um ou mais arquivos XML que contenham medições de métricas de software. É necessário que a ferramenta importe mais de um arquivo XML para um único sistema pois ele pode ser constituído de vários projetos. O formato do arquivo XML de entrada é o mesmo usado para a saída do *Eclipse Metrics Plugin*. Isso ocorre porque o *Eclipse Metrics Plugin* foi a ferramenta utilizada para extrair o *dataset* de métricas usado no processo de derivação dos valores referência, e também por ela ser uma ferramenta integrada com a IDE mais usada no mundo para desenvolvimento em Java, o Eclipse. Dessa forma, são evitadas as diferenças de medições de métricas de software que ocorrem por implementações distintas entre as ferramentas de extração e que geram ruído quando da utilização dos valores referência e a sua interpretação dos valores extraídos. O ideal é a utilização conjunta do *Eclipse Metrics Plugin* e da *RAFTool*, apesar de não ser um fato obrigatório. *RAFTool* permite, após a seleção do nome e a importação das medições, a opção de concluir aquela instanciação do sistema analisado. Nesse momento, a ferramenta processa os arquivos XML, inserindo as medições em um banco de dados *standalone*, que persistirá os dados para trabalhos de filtragem até o término daquela execução do programa. Esses dados não ficam persistidos para uma próxima execução.

#### 4.2. Processo de Filtragem de Métodos, Classes ou Pacotes

O usuário deve selecionar o tipo de artefato filtrado: método, classe ou pacote. Posteriormente, ele deve digitar a expressão booleana de filtragem de métodos, classes ou pacotes. Essa expressão booleana deve ser uma combinação de ANDs e ORs dentro das faixas dos valores referência das métricas que serão utilizadas na filtragem que está sendo criada. Para isso, o seguinte formato é

definido: COMMON|CASUAL|UNCOMMON[METRIC\_ID], onde COMMON corresponde à faixa *Bom/Regular*, CASUAL à faixa *Regular/Ocasional* e UNCOMMON à faixa *Ruim/Raro*. Quando o usuário escolhe a faixa CASUAL, o sistema retorna os métodos, classes e pacotes que estão na faixa *Regular/Ocasional* ou *Ruim/Raro*. METRIC\_ID é o identificador da métrica que deve estar naquela faixa, devendo seguir o padrão estabelecido pelo Eclipse Metrics Plugin como ID. Logo, se o usuário digitar CASUAL[MLOC], por exemplo, o processo de filtragem retorna todos os métodos do sistema instanciado cuja classificação pelo valor referência sugerido para a métrica *Número de Linhas de Código por Método* (MLOC) for *Regular/Ocasional*. Com a utilização de ANDs e ORs, o usuário pode compor sua filtragem com vários valores referência. Por exemplo, se o usuário digitar CASUAL[MLOC] AND UNCOMMON[NBD], o processo de filtragem retorna todos os métodos do sistema classificados como *Regular/Ocasional* para a métrica *Número de Linhas de Código por Método* (MLOC) e, desses métodos, todos aqueles classificados como *Ruim/Raro* pela métrica *Profundidade de Blocos Aninhados* (NBD). Parênteses são aceitos nessa expressão como indicador da precedência das operações. RAFTool acessa o banco de dados criado na fase de instanciação e seleciona os artefatos conforme os parâmetros estabelecidos.

### 4.3. Visualização/Exportação dos Resultados

Os artefatos filtrados são exibidos em uma tabela. Para o caso dos métodos, são exibidos o nome do método, o nome da classe e o nome do pacote. Para as classes, são o seu nome, o nome do arquivo da classe e o nome do pacote. A exibição do nome do arquivo é útil pois um arquivo pode conter mais de uma classe. Para os pacotes, são exibidos somente os nomes. Os artefatos filtrados podem ser exportados para uma planilha em formato CSV, arquivo escolhido pelo usuário na hora da exportação.

## 5. Arquitetura

### 5.1. Model-View-Controller (MVC)

Para o desenvolvimento da ferramenta RAFTool foi utilizado o padrão arquitetural Model-View-Controller (MVC) [Sommerville 2010], que separa a apresentação e a interação da lógica de negócio e dados do sistema. O Modelo encapsula, além dos objetos de domínio, o acesso ao banco de dados utilizado pela aplicação. Outra responsabilidade do modelo dentro da RAFTool é relativo à lógica de negócio da aplicação, que são objetos cujas responsabilidades consistem em realizar a lógica relativa ao processo de filtragem. Esse processo de filtragem envolve a interpretação das expressões booleanas utilizadas pelos usuários, bem como a sua aplicação nos artefatos selecionados. Utilizou-se o *framework* Javaluator<sup>3</sup>, que auxilia na avaliação de expressões regulares em Java. Na parte de persistência, foi utilizado o banco de dados HSQLDB. HSQLDB é um servidor de banco de dados embarcado, que foi desenvolvido na plataforma Java e é de código aberto.

A Visão foi desenvolvida com a utilização do *Swing*. *Swing* implementa um conjunto de componentes para construir interfaces gráficas para interação com usuários (GUIs). Os componentes *Swing* são

3 <http://javaluator.sourceforge.net/en/home/>

implementados diretamente em Java. Utilizou-se a IDE Netbeans, que possui recursos de construção de interfaces *Swing* com a utilização de *Drag and Drop*, facilitando a construção de interfaces gráficas.

O Controle tem a responsabilidade de mapear as ações dos usuários em ações da camada de Modelo. Além disso, mediante a resposta da Camada de Modelo, ele é responsável por selecionar a visão apropriada ao usuário. Se a interface for trocada, por exemplo, o Controle cuida para que a camada de Modelo se mantenha inalterada.

## 5.2. Estrutura de Pacotes da RAFTool

Nesta seção são documentados os pacotes do sistema, suas responsabilidades e como interagem entre si. A Figura 2 mostra o diagrama de pacotes de *RAFTool*, que exhibe os módulos do sistema e as dependências entre eles, ilustrando a arquitetura proposta para o sistema. A seguir são descritas as responsabilidades de cada um desses pacotes:

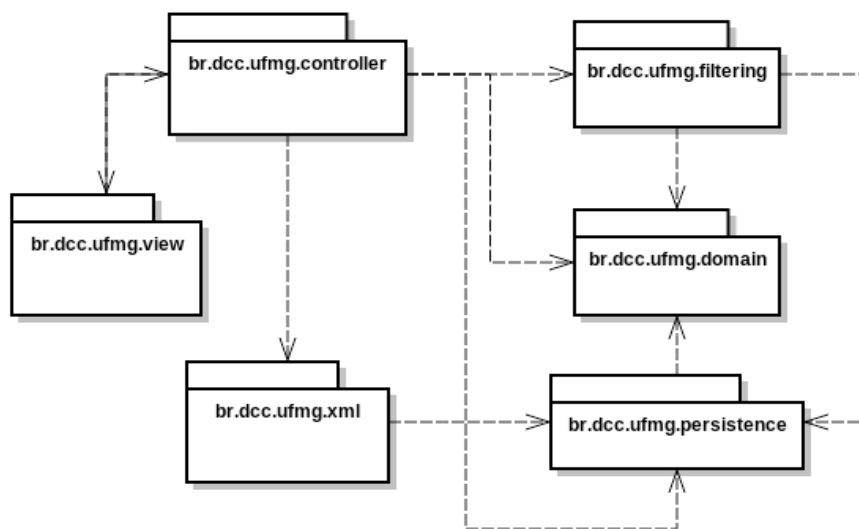


Figure 2. Diagrama de Pacotes da *RAFTool*.

1. *br.dcc.ufmg.view*: agrupamento das classes responsáveis pela camada de visão do sistema. Possui dependência com o módulo *controller*.
2. *br.dcc.ufmg.controller*: agrupamento das classes responsáveis pela camada de controle do sistema. Possui dependência com o a *view*, bem como com os restante dos módulos do sistema, que constituem a camada de modelo.
3. *br.dcc.ufmg.xml*: agrupamento das classes que trabalham com a leitura dos documentos XML, conversão para objetos Java e importação dos dados lidos desses documentos para o banco de dados. Para tal, possui dependência em relação ao módulo *persistence*, responsável pelas operações de banco de dados.
4. *br.dcc.ufmg.filtering*: agrupamento de classes responsáveis pelas operações de filtragem a serem realizadas nos artefatos do software analisado. Esse módulo do sistema possui dependência

com o módulo *persistence*, que é responsável por acessar os dados do sistema analisado. Possui dependência com o módulo *domain*, que modela o cenário real de um software orientado por objetos.

5. *br.dcc.ufmg.domain*: agrupamento de classes responsáveis por representar o vocabulário e descrever os conceitos chaves de um software orientado por objetos, bem como as relações entre as entidades estabelecidas. Sobre os objetos do domínio são executadas as operações de filtragem de dados.
6. *br.dcc.ufmg.persistence*: agrupamento de classes responsáveis por tratar operações de inserção, leitura e atualização dos dados utilizados no sistema. É um módulo independente, que modela o domínio do problema.

As principais classes do sistema foram documentadas com a utilização do *Javadoc* e disponibilizados em formato HTML por meio do seguinte *link*:

```
http://www.dcc.ufmg.br/~tfilo/raftool/doc
```

### 5.3. Infraestrutura para Instalação e Utilização da *RAFTool*

Para instalar e utilizar a *RAFTool*, é necessário um computador com a máquina virtual Java instalada, com versão igual ou superior a 1.6. O executável está disponível em:

```
http://www.dcc.ufmg.br/~tfilo/raftool/raftool.jar
```

## 6. Conclusão

A ferramenta *RAFTool* realiza a filtragem de métodos, classes e pacotes que possuem medições anômalas de métricas de softwares orientados por objetos segundo os valores referência sugeridos por Filó *et al.* (2014). Como exemplo da utilização da *RAFTool*, disponibilizamos um vídeo-tutorial *online* que mostra o uso da ferramenta:

```
http://www.dcc.ufmg.br/~tfilo/raftool/demo_raftool.htm
```

*RAFTool* é uma ferramenta que suporta a utilização desses padrões de forma efetiva na filtragem de métodos, classes e pacotes com medições anômalas. A ferramenta apoia de forma efetiva os Processos de Medição - MED do nível F (Gerenciado) do MR-MPS-SW, proporcionando a efetiva transformação de uma série de medidas associadas às estruturas internas do código-fonte em informações úteis sobre sua qualidade, gerando informações úteis para a tomada de decisão. Isso permite à organização direcionar esforços de forma pontual dentro do código-fonte para realizar, por exemplo, refatorações e maior carga de testes, haja visto o maior risco. Dessa forma, espera-se que, em conjunto com o *Eclipse Metrics Plugin* e os valores referência sugeridos por Filó *et al.*, *RAFTool* colabore com os esforços de pesquisa que visam a aplicar de forma efetiva as métricas

de software no gerenciamento da qualidade interna de software por meios quantitativos. Como trabalhos futuros, vislumbramos a integração da *RAFTool* ao *plugin Eclipse Metrics*, monitorando o código de forma dinâmica durante o desenvolvimento, manutenção e evolução de software, disparando alarmes que possibilitem aos desenvolvedores a utilização de métricas de forma integrada à ferramenta de desenvolvimento. Nessa ferramenta, o desenvolvedor não precisaria se preocupar com números, pois os valores referência propostos funcionariam como sinais de trânsito: a faixa *Bom/Frequente* seria o sinal "verde", a faixa *Regular/Ocasional* o sinal "amarelo", sugerindo atenção, e a faixa *Ruim/Raro* o sinal "vermelho", indicando que deve-se parar e analisar o que está sendo codificado. O código-fonte da *RAFTool* está disponível no *GoogleCode*, que hospeda projetos em um ambiente colaborativo de desenvolvimento, sobre os termos da GPLv3. Utilizou-se o *Subversion* como ferramenta de controle de versão. O código-fonte pode ser baixado diretamente no Eclipse e está disponível por meio do seguinte *link*:

<https://code.google.com/p/raftool/source/checkout>

## References

Filó, T., Bigonha, M., and Ferreira, K. (2014). Um método de extração de valores referência para métricas de softwares orientados por objetos. In CBSOFT-WTDSOFT 2014.

Softex, X. (2012). MPS.BR - Melhoria de Processo do Software Brasileiro. Technical report, Association for Promoting the Brazilian Software Excellence - SOFTEX.

Sommerville, I. (2010). Software Engineering. Addison-Wesley, Harlow, England, 9 edition.







O WAMPS 2014 (X Workshop Anual do MPS) tem por objetivo reunir a Comunidade de Prática do MPS, com foco nos implementadores, avaliadores, consultores de aquisição e instrutores MPS, membros da ETM/MPS - Equipe Técnica do Modelo e do FCC/MPS - Fórum de Credenciamento e Controle, IOGE/MPS – Instituições Organizadoras de Grupos de Empresas e Agentes Regionais Softex, clientes MPS e pessoal da Indústria, Governo, Academia do país e exterior, interessados tanto no aprimoramento de processos de software e serviços de TI usando o modelo MPS quanto nos resultados alcançados no programa MPS.BR – Melhoria de Processo do Software Brasileiro.

Este ano se comemora a décima edição do WAMPS e a superação da marca de 600 avaliações MPS publicadas, a programação desta nona edição do WAMPS conta com um palestrante internacional e palestrantes nacionais especialmente convidados, além de oferecer três cursos e dois painéis de grande interesse nesta área, acolher importantes reuniões como a reunião semestral do CGP/MPS.BR – Conselho de Gestão do Programa e realizar a entrega de placas para empresas recém-avaliadas MPS.

Além disso, com o apoio da SBC - Sociedade Brasileira de Computação, o WAMPS promoveu uma chamada de trabalhos envolvendo engenharia de software, que estejam relacionados ou sejam aplicáveis ao contexto de iniciativas de melhoria de processos de software e serviços. Os vinte e dois trabalhos aceitos serão apresentados em sete sessões técnicas. Nesta publicação encontram-se todos os artigos selecionados pelo Comitê de Programa do WAMPS 2014 para apresentação nas sessões técnicas.

apoio:



Ministério da  
Ciência, Tecnologia  
e Inovação



**WWW.SOFTEx.BR/MPSBR**