

WAMPS 2010

VI Workshop Anual do MPS

Campinas-SP, 26 a 29 de outubro de 2010

FICHA CATALOGRÁFICA ELABORADA PELO

Sistemas de Bibliotecas da UNICAMP /

Diretoria de Tratamento da Informação

W892a

Workshop Anual do MPS, 6., Campinas, SP, 2010.
Anais do VI WAMPS 2010, realizado em Campinas, de
26 a 29 de Outubro de 2010; organizadores: Ana Regina
Cavalcanti da Rocha, Carla Lima Reis, Tayana Conte. —
Campinas, SP : Associação para Promoção da Excelência
do Software Brasileiro - SOFTEX, 2010.
264p.

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

I. Rocha, Ana Regina Cavalcanti da. II. Reis, Carla Lima.
III. Conte, Tayana. IV. Título.

CDD – 001.6425
- 001.642

ISBN 978-85-99334-19-5

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 Campinas, SP, Brasil.

A SOFTEX é responsável pela gestão do Programa Prioritário em Informática do Governo Federal para Promoção da Excelência do Software Brasileiro, o Programa SOFTEX.

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 1.600 empresas com atividades em software e serviços de TI.

Presidente da SOFTEX

Waldemar Alberto Borges Rodrigues Neto

Vice-Presidente Executivo da SOFTEX

Arnaldo Bacha de Almeida

Diretoria Executiva da SOFTEX

Descartes de Souza Teixeira – Assessor de Planejamento e Gestão

Djalma Petit – Diretor de Mercado

Ephrain Guilherme Neitzke – Controladoria

John Lemos Forman – Diretor de Capacitação e Inovação

José Antonio Antonioni – Diretor de Qualidade e Competitividade

Dentre as atividades da SOFTEX no âmbito da Diretoria de Qualidade e Competitividade, 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

Nelson Henrique de Oliveira Franco – Gerente de Operações

André Luis Chamelet Sotovia

Cleide Gonçalves da Silva

Elidiane Teixeira Barroso

Sumário

Apresentação	10
Organização do WAMPS 2010	11
Programação do WAMPS 2010	12
1 - Palestrantes convidados	
1.1 – “O que é relevante, certificado ou competência?”. Arndt von Staa (PUC-Rio)	16
1.2 - “Change does not just “Happen” – Building sustainable changes in organizations”. Jorge Boria (Liveware)	18
2 – Painel MPS.BR	
2.1 – “Programa MPS.BR - Avanços, conquistas e resultados alcançados”. José Antonio Antonioni (SOFTEX)	20
2.2 – “Resultados Iniciais iMPS 2010: Variação de Desempenho nas Empresas que Adotaram o Modelo MPS”. Guilherme Horta Travassos (COPPE/UFRJ)	24
3 - Palestra convidada	
3.1 – “Confiabilidade de Software”. Odair Silva (OSPRA)	34
4 – Artigos técnicos selecionados	
4.1 – “PCSSCEG - Processo de contratação de software e serviços correlatos para entes governamentais”, Claudio Cruz, Edmeia Leonor Pereira Andrade, Rejane Costa (TCU; EMBRAPA e UNB)	36
4.2 - “Proposta e Aplicação de um Método de Seleção de Fornecedores Baseado no Guia de Aquisição do MPS.BR”, Ana Marcia Duarte , Jefferson Mello, Denio Duarte (Unochapeco e Universidade Federal da Fronteira Sul - UFFS)	46
4.3 - “Aplicação da Memória Organizacional no contexto da implantação do processo de melhoria de software”, Edson Yhe, Jandira Palma e Rafael Parente (Universidade Estadual de Londrina)	56
4.4 - “Construindo um Modelo Especialista de Estimativa de Esforço de Aplicações Web usando Redes Baesianas”, Emilia Mendes (The University of Auckland)	66

4.5 - "Implementando o nível F do MR-MPS com práticas da metodologia ágil Scrum", Edmar Catunda, Camila Nascimento, Cristina Cerdeiral, Gleison Santos, Ana Regina Rocha (Rightway Consultoria & Sistemas; COPPE/UFRJ; Unirio; COPPE/UFRJ)	78
4.6 - "Implantação de Modelos de Maturidade com Metodologias Ágeis: Um Relato de Experiências", Rafael Prikladnicki, Ana Liddy Magalhães (PUCRS; Universidade FUMEC / QualityFocus)	88
4.7 - "Padrões de Processo para Atributos de Processo", Juliano Oliveira, Paulo Henrique Rocha Faleiro, Adriana Silveira de Souza (Universidade Federal de Goiás)	100
4.8 - "Uso do Ambiente WebAPSEE na Implementação do Nível G do MPS.BR na Equilibrium Web", Ernani Sales, Carla Lima Reis, Rodrigo Reis, Luciana Nascimento, Sebastião Junior (UFPA; University of Para; Equilibrium Web)	108
4.9 - "Aplicação de Jogos Educativos para Aprendizagem em Melhoria de Processo e Engenharia de Software", Alessandra Zoucas, Marcello Thiry, Rafael Gonçalves, Antonio Carlos Silva, Clenio Salviano (UNIVALI; CTI - Centro de Tecnologia da Informação Renato Archer)	118
4.10 - "MPS.BR Nível A: Experiência da Stefanini", Washington Souza, Marcelo Ramasco, Andrea Mattos, Elaine Pinheiro, Rodrigo Ricci (Stefanini)	128
4.11 - "Resultados de um Estudo Qualitativo sobre a implementação do modelo MPS em empresas do programa AmazonSoft", Davi Viana, Jacilane Rabelo, Carlos Mar, Elisângela Aguiar, Paulino Viana, Dalton Vilela (Universidade Federal do Amazonas; FabriQ; InfoSigma; FUCAPI)	138
 5 – Artigos selecionados sobre ferramentas	
5.1 - "Análise de Ferramentas para Apoio a Gerência de Projetos e Gerência de Requisitos de Software", Fabiana Mendes, Patrícia Fernandes, Juliano Olkiveira, Caroline Mota, Maria Dalva Martins e Rosângela Nunes (UFG)	148
5.2 - "Um Estudo Experimental sobre Abordagens de Apoio à Rastreabilidade de Requisitos", Anna Marques, Jacilane Rabelo, Sérgio Vieira e Tayana Conte (UFAM)	158
5.3 - "CPqD Developer Suite", Gustavo Chaves, André Villas-Boas (Fundação CPqD, Telecom & IT Solutions)	166
5.4 - "Brechó 2.0: Uma Ferramenta para Apoiar a Gerência de Reutilização", Rodrigo Santos, Anderson Marinho, Marlon Silva, Claudia Werner, Leonardo Murta (COPPE/UFRJ; UFF)	174
5.5 - "Ferramenta Colaborativa para Apoio à Gestão do Conhecimento em Implementações MPS", Luiz Leandro Fortaleza, Gleison Santos, Tayana Conte (UFAM; UNIRIO; UFAM)	184
5.6 - "Project Builder: uma Ferramenta de Apoio a Implementação do Processo Gerência de Projetos do MPS.BR", Bernardo Grassano, Eduardo Carvalho, Analia Ferreira e Mariano Montoni (Project Builder; Informal Informática; ProMove Soluções)	192

5.7 - "Implantação dos Processos Gerência de Projeto e Medição com Auxílio de Ferramenta baseada em Planilhas", Carlos Simões, Cláudia Lasmar, Gleison Santos (Synapsis Brasil; Ampla Energia e Serviço S.A.; UNIRIO)	202
5.8 - "Uma Implementação do Processo de Gerência de Projetos Usando Ferramentas de Software Livre", Ewelton Yoshidome, Maurício Souza, Wallace Lira e Sandro Ronaldo Bezerra Oliveira (UFPA)	212
5.9 - "O Uso do dotProject no Processo Prodepa de Desenvolvimento de Software", Renata Monteiro, Leila Daher, Claudio Martins (PRODEPA)	220
5.10 - "WebAPSEE Pro: Um Ambiente de Apoio a Gerência de Processos de Software", Ernani Sales, Murilo Sales, Anderson Costa, Carla Lima Reis, Rodrigo Reis (UFPA)	228
5.11 - "Process Publisher: Um Gerador de Guias Eletrônicos de Processos de Software", Adailton Lima, Rodrigo Reis, Carla Lima Reis, Breno França, Marcelo Pereira (UFPA)	236
5.12 - "WKM: Uma Ferramenta para Auxiliar a Gerência de Conhecimento Integrada a um ADS Centrado em Processos", Jadielly Oliveira, Liken Lima, Silvia Nunes das Dores, Ernani Sales, Gabriela Andrade, Carla Lima Reis (UFPA)	244
5.13 - "Integração da Gerência de Configuração com a Gerência de Projetos e de Requisitos em um Ambiente Colaborativo", Jonnathan Carvalho, Mario Amaral, Mara Barcelos, Simone Vasconcelos, Aline Vasconcelos (Instituto Federal Fluminense)	254

Apresentação

O WAMPS 2010 - VI Workshop Anual do MPS, realizado pela SOFTEX, tem por objetivo reunir os representantes da Indústria, Governo e Academia e, em especial, as entidades, órgãos e agências oficiais de fomento tanto do Brasil quanto de países latino-americanos envolvidos e interessados na utilização e evolução do Modelo MPS e do Programa MPS.BR – Melhoria de Processo do Software Brasileiro, uma realidade viável para as organizações que procuram aprimorar seus processos de software.

Desde 2009 o Workshop Anual do MPS integra os eventos anteriormente executados e representados pelos Workshop de II – Instituições Implementadoras MPS, Workshop de IA – Instituições Avaliadoras MPS, Workshop de IOGE – Instituições Organizadoras de Grupos de Empresas MPS e Workshop de Empresas MPS. A integração ocorrida a partir da edição de 2009 teve êxito em seu objetivo de aproveitar as experiências, aumentar a sinergia entre os grupos e tirar proveito da maturidade adquirida ao longo de quase 7 anos de trabalho intenso. 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.

A novidade da edição do WAMPS 2010 é a Sessão de Ferramentas. O objetivo da Sessão é proporcionar um fórum para intercâmbio de experiências e de soluções automatizadas para apoiar a implementação e avaliação do MR-MPS em suas mais diversas necessidades e manifestações. O público-alvo inclui tanto empresas que desenvolveram e/ou “customizaram” ferramentas para apoio aos processos do MPS, bem como membros da comunidade acadêmica, que poderão demonstrar os resultados de seus projetos de pesquisa que resultaram em ferramentas de apoio ou em estudos sobre ferramentas para implementação e/ou avaliação MPS. Esta seção é uma oportunidade de apresentar e divulgar esses produtos para os participantes do WAMPS. Em sua 1ª edição, a Sessão de Ferramentas consistirá de apresentações e demonstrações práticas das ferramentas selecionadas.

Durante a semana do workshop, a programação do WAMPS 2010 inclui cursos oficiais e especiais; palestras e painéis; palestras convidadas; e, a apresentação de artigos técnicos e ferramentas. Também estão programadas reuniões do CGP – Conselho de Gestão do Programa MPS.BR e de Coordenadores de II, IA e IOGE.

Os artigos técnicos e ferramentas a serem apresentados foram submetidos em atenção a uma Chamada de Trabalhos e foram selecionados por um Comitê de Programa composto por pesquisadores e profissionais especialistas no Modelo MPS.

Nestes Anais do WAMPS 2010 encontram-se tanto o resumo das apresentações dos palestrantes internacionais, do painel MPS.BR e de uma palestra convidada, quanto a íntegra dos artigos técnicos e artigos selecionados para publicação da Sessão de Ferramentas.

Campinas, outubro de 2010

Carla Alessandra Lima Reis (UFPA)

Coordenadora científica do WAMPS 2010 – VI Workshop Anual do MPS

Tayana Conte (UFAM)

Coordenadora da Sessão de Ferramentas do WAMPS 2010 - VI Workshop Anual do MPS

José Antonio Antonioni (SOFTEX) / Kival Weber (SOFTEX/MPS.BR)

Nelson Franco (SOFTEX) / Ana Regina Rocha (COPPE/UFRJ)

Coordenação Geral do WAMPS 2010 – VI Workshop Anual do MPS

Organização do WAMPS 2010

Coordenação Geral

José Antonio Antonioni (SOFTEX)
Kival Weber (SOFTEX/MPS.BR)
Ana Regina Rocha (COPPE/URFJ)
Nelson Franco (SOFTEX)

Coordenação Científica

Carla Alessandra Lima Reis (UFPA)

Coordenação Científica da Sessão de Ferramentas

Carla Alessandra Lima Reis (UFPA)

Comitê de Avaliação da Trilha Artigos Técnicos

Alexandre Vasconcelos (UFPE)
Ana Liddy Magalhães (FUMEC e QUALITY FOCUS)
Ana Paula Terra Bacelo (PUCRS)
Ana Regina Cavalcanti da Rocha (COPPE/URFJ)
Carla Alessandra Lima Reis (UFPA e QR)
Carlos Pietrobon (UFOP e PUC MINAS)
Christiane Gresse von Wangenheim (UFSC)
Cristina Angela Filipak Machado (QUALITYFOCUS)
Danilo Scalet (CELEPAR)
Edmeia Leonor Pereira de Andrade (EMBRAPA e UCB)
Fábio Bianchi (UCB)
Gleison dos Santos Souza (UNIRIO)
Juliano Lopes de Oliveira (UFG e ESTRATÉGIA TI)
Marcello Thiry Comicholi da Costa (UNIVALI e INCREMENTAL)
Marcelo Yamaguti (PUCRS)
Mariano Montoni (ProMove)
Rafael Prikladnicki (PUCRS)
Ricardo Falbo (UFES)
Rodrigo Quites Reis (UFPA e QR)
Rosângela Mirian Lemos Oliveira Mendonça (FUMSOFT)

Revisores Externos da Trilha Artigos Técnicos

Breno França (UFRJ)
Ellen Polliana Ramos Souza (UFRPE)
Ernani Sales (UFPA)
Hugo Vieira L. de Souza (UFPE)
Jadielly Oliveira (UFPA)
Luciana Nascimento (UFPA)
Murilo Sales (UFPA)
Paulino Viana (FUCAPI)

Comitê de Avaliação da Trilha de Ferramentas

Adriano Bessa (UNIFOR)
Alexandre Vasconcelos (UFPE)
Ana Cecília Peixoto Zabeu (ASR)
Ana Liddy Magalhães (FUMEC e QUALITY FOCUS)
Ana Regina Cavalcanti da Rocha (COPPE/URFJ)
Carla Alessandra Lima Reis (UFPA e QR)
Clenio Salviano (CTI Renato Archer)
Fábio Bianchi (UCB)
Gleison dos Santos Souza (UNIRIO)
Juliano Lopes de Oliveira (UFG e ESTRATÉGIA TI)
Marcello Thiry Comicholi da Costa (UNIVALI e INCREMENTAL)
Rafael Prikladnicki (PUCRS)
Ricardo Falbo (UFES)
Rodrigo Quites Reis (UFPA e QR)
Rosângela Mirian Lemos Oliveira Mendonça (FUMSOFT)

Revisores Externos da Trilha de Ferramentas

Anderson Costa (UFPA)
Aristides Vicente de Paula Neto (UFPE)
Eder Barbosa (UFPA)
Edgar Lopes Banhesse (CTI Renato Archer)
Eduardo Moraes (UFPE)
Ernani Sales (UFPA)
Jadielly Oliveira (UFPA)
Luciana Nascimento (UFPA)
Nielso Oliveira Junior (Serviço Federal de Processamento de Dados)

Programação do WAMPS 2010

WAMPS 2010 - VI WORKSHOP ANUAL DO MPS – 26 a 29 de outubro de 2010

Local: Hotel Tryp Campinas (by Sol Meliá) - Rua Severo Penteado, 140 – Cambuí, Campinas, SP

Coordenação Geral: José Antonio Antonioni, Kival Weber, Ana Regina Rocha e Nelson Franco

Coordenação Científica: Carla Lima Reis (UFPA)

Coordenação Científica da Sessão de Ferramentas: Tayana Conte (UFAM)

3ª feira - 26 de outubro de 2010		
08:00-09:30		Prova para Instrutores de C1-MPSBR
09:30-10:00	Artigo aceito I: "PCSSCEG - Processo de contratação de software e serviços correlatos para entes governamentais", Claudio Cruz, Edmeia Leonor Pereira Andrade, Rejane Costa (TCU; EMBRAPA e UNB)	
10:00-10:30	Intervalo	
10:30-11:00	Artigo aceito II: "Proposta e Aplicação de um Método de Seleção de Fornecedores Baseado no Guia de Aquisição do MPS.BR", Ana Marcia Duarte, Jefferson Mello, Denio Duarte (Unochapeco e Universidade Federal da Fronteira Sul - UFFS)	
11:00-11:30	Artigo aceito III: "Aplicação da Memória Organizacional no contexto da implantação do processo de melhoria de software", Edson Yhe, Jandira Palma e Rafael Parente (Universidade Estadual de Londrina)	
11:30-12:00	Artigo aceito IV: "Construindo um Modelo Especialista de Estimativa de Esforço de Aplicações Web usando Redes Baesianas", Emilia Mendes (The University of Auckland)	
12:00-14:00	Almoço	
14:00-15:30	Palestrante Nacional: Arndt von Staa (PUC-Rio) "O que é relevante, certificado ou competência?"	
15:30-16:00	Intervalo	
16:00-16:30	Artigo aceito V: "Implementando o nível F do MR-MPS com práticas da metodologia ágil Scrum", Edmar Catunda, Camila Nascimento, Cristina Cerdeiral, Gleison Santos, Ana Regina Rocha (Rightway Consultoria & Sistemas; COPPE/UFRJ; Unirio; COPPE/UFRJ)	
16:30-17:00	Artigo aceito VI: "Implantação de Modelos de Maturidade com Metodologias Ágeis: Um Relato de Experiências", Rafael Prikladnicki, Ana Liddy Magalhães (PUCRS; Universidade FUMEC / QualityFocus)	
17:00-18:30	Painel: "MPS.BR", painelistas: José Antonio Antonioni (Programa MPS.BR - Avanços, conquistas e resultados alcançados) e Guilherme Horta Travassos (Resultados Iniciais iMPS 2010: Variação de Desempenho nas Empresas que Adotaram o Modelo MPS), moderador: Kival Weber.	
18:30-20:00	Abertura do WAMPS 2010 e Entrega de Placas MPS.BR	
20:00-23:00	Coquetel do WAMPS 2010	

4ª feira - 27 de outubro de 2010			
08:00-09:00		Prova para Instrutores de C1-MPSBR	Reu CGP - Conselho de Gestão do Programa MPS.BR (9:00 - 12:00)
09:00-09:30			
09:30-10:00	Artigo aceito VII: " Padrões de Processo para Atributos de Processo ", Juliano Oliveira, Paulo Henrique Rocha Faleiro, Adriana Silveira de Souza (Universidade Federal de Goiás)		
10:00-10:30	Intervalo		
10:30-11:30	Palestra Convidada: "Odair Silva (OSPRA) " "Confiabilidade de Software"		
11:30-12:00	Artigo aceito VIII: " Uso do Ambiente WebAPSEE na Implementação do Nível G do MPS.BR na Equilibrium Web ", Ernani Sales, Carla Lima Reis, Rodrigo Reis, Luciana Nascimento, Sebastião Junior (UFPA; University of Para; Equilibrium Web)		
12:00-14:00	Almoço		
14:00-15:30	Palestrante Internacional: "Jorge Boria (Liveware)" "Change does not just "Happen" "Building sustainable changes in organizations"		
15:30-16:00	Intervalo		
16:00-16:30	Artigo aceito IX: " Aplicação de Jogos Educativos para Aprendizagem em Melhoria de Processo e Engenharia de Software ", Alessandra Zoucas, Marcello Thiry, Rafael Gonçalves, Antonio Carlos Silva, Clenio Salviano (UNIVALI; CTI - Centro de Tecnologia da Informação Renato Archer)		
16:30-17:00	Artigo aceito X: " MPS.BR Nível A: Experiência da Stefanini ", Washington Souza, Marcelo Ramasco, Andrea Mattos, Elaine Pinheiro, Rodrigo Ricci (Stefanini)		
17:00-17:30	Artigo aceito XI: " Resultados de um Estudo Qualitativo sobre a implementação do modelo MPS em empresas do programa AmazonSoft ", Davi Viana, Jacilane Rabelo, Carlos Mar, Elisângela Aguiar, Paulino Viana, Dalton Vilela (Universidade Federal do Amazonas; FabriQ; InfoSigma; FUCAPI)		
17:30-19:00	Reunião Coordenadores de II e de IA		

5ª feira - 28 de outubro de 2010			
08:00–10:00	Curso especial de Implantação de Modelos de Maturidade com Metodologias Ágeis (Ana Liddy Magalhães - FUMEC/Qualityfocus)	Exame de Controle Estatístico de Processos (CEP) para Avaliadores	Exposição de Ferramentas
10:00-10:30	Intervalo		
10:30-11:00	Artigo aceito XII: "Análise de Ferramentas para Apoio a Gerência de Projetos e Gerência de Requisitos de Software", Fabiana Mendes, Patrícia Fernandes, Juliano Olkiveira, Caroline Mota, Maria Dalva Martins e Rosângela Nunes (UFG)		
11:00-11:30	Artigo aceito XIII: "Um Estudo Experimental sobre Abordagens de Apoio à Rastreabilidade de Requisitos", Anna Marques, Jacilane Rabelo, Sérgio Vieira e Tayana Conte (UFAM)		
11:30-12:00	Artigo aceito XIV: "CPqD Developer Suite", Gustavo Chaves, André Villas-Boas (Fundação CPqD, Telecom & IT Solutions)		
12:00-14:00	Almoço		
14:00-15:30	Painel: "Implementação MPS com Métodos Ágeis", painelistas: Edney Mossambani (Accion), Gustavo Lima e João Geroldo (SIMUS e SOCIN), Paulo Alvim (Powerlogic), moderadora: Ana Liddy Magalhães (Qualityfocus).		
15:30-16:00	Intervalo		
16:00-17:00	Palestra Convidada: "Pesquisa da Qualidade no Setor de Software Brasileiro - 2009" Diva da Silva Marinho (MCT/SEPIN) e Sheila dos Santos Reinehr (PUCPR/Qualityfocus)		
17:00-17:30	Artigo aceito XV: "Brechó 2.0: Uma Ferramenta para Apoiar a Gerência de Reutilização", Rodrigo Santos, Anderson Marinho, Marlon Silva, Claudia Werner, Leonardo Murta (COPPE/ UFRJ; UFF)	Curso Avaliação de Bases de Medidas para Controle Estatístico (Monalessa Perini Barcellos - UFES)	
17:30-18:00	Artigo aceito XVI: "Ferramenta Colaborativa para Apoio à Gestão do Conhecimento em Implementações MPS", Luiz Leandro Fortaleza, Gleison Santos, Tayana Conte (UFAM; UNIRIO; UFAM)		
18:00-19:30	Reunião Coordenadores de IOGE		
19:30-20:00			

6ª feira - 29 de outubro de 2010		
08:00-10:00	Continuação do Curso especial de Implantação de Modelos de Maturidade com Metodologias Ágeis (Ana Liddy Magalhães - FUMEC/ Qualityfocus)	
10:00-10:30	Intervalo	
10:30-11:00	Artigo aceito XVII: "Project Builder: uma Ferramenta de Apoio a Implementação do Processo Gerência de Projetos do MPS.BR" , Bernardo Grassano, Eduardo Carvalho, Analia Ferreira e Mariano Montoni (Project Builder; Informal Informática; ProMove Soluções)	
11:00-11:30	Artigo aceito XVIII: "Implantação dos Processos Gerência de Projeto e Medição com Auxílio de Ferramenta baseada em Planilhas" , Carlos Simões, Cláudia Lasmar, Gleison Santos (Synapsis Brasil; Ampla Energia e Serviço S.A.; UNIRIO)	
11:30-12:00	Artigo aceito XIX: "Uma Implementação do Processo de Gerência de Projetos Usando Ferramentas de Software Livre" , Ewelton Yoshidome, Maurício Souza, Wallace Lira e Sandro Ronaldo Bezerra Oliveira (UFPA)	
12:00-14:00	Almoço	
14:00-14:30	Artigo aceito XX: "O Uso do dotProject no Processo Prodepa de Desenvolvimento de Software" , Renata Monteiro, Leila Daher, Claudio Martins (PRODEPA)	Exposição de Ferramentas
14:30-15:00	Artigo aceito XXI: "WebAPSEE Pro: Um Ambiente de Apoio a Gerência de Processos de Software" , Ernani Sales, Murilo Sales, Anderson Costa, Carla Lima Reis, Rodrigo Reis (UFPA)	
15:00-15:30	Artigo aceito XXII: "Process Publisher: Um Gerador de Guias Eletrônicos de Processos de Software" , Adailton Lima, Rodrigo Reis, Carla Lima Reis, Breno França, Marcelo Pereira (UFPA)	
15:30-16:00	Intervalo	
16:00-16:30	Artigo aceito XXIII: "WKM: Uma Ferramenta para Auxiliar a Gerência de Conhecimento Integrada a um ADS Centrado em Processos" , Jadielly Oliveira, Liken Lima, Silvia Nunes das Dores, Ernani Sales, Gabriela Andrade, Carla Lima Reis (UFPA)	
16:30-17:00	Artigo aceito XXIV: "Integração da Gerência de Configuração com a Gerência de Projetos e de Requisitos em um Ambiente Colaborativo" , Jonnathan Carvalho, Mario Amaral, Mara Barcelos, Simone Vasconcelos, Aline Vasconcelos (Instituto Federal Fluminense)	
17:00-17:30	Encerramento do WAMPS 2010	

O que é relevante, certificado ou competência ?

A.v. Staa (arndt@inf.puc-rio.br)

Departamento de Informática
Pontifícia Universidade Católica
22451-070 Rio de Janeiro,
Brasil
Outubro 2010

Resumo

Recomenda-se insistentemente obter certificados para inúmeras coisas. Organizações buscam certificados indicadores da maturidade de sua capacitação, tais como MPS.br, CMMI, e outros. Profissionais procuram obter certificados de competência no uso do software A, do gerenciador de bases de dados B e por aí vai. Fica a pergunta, isso realmente leva a software de melhor qualidade? O usuário observa esta melhoria?

Diferentes autores dizem que a competência entre excelentes profissionais e profissionais pouco hábeis varia de 1 para 20 a 1 para 35. Ou seja, um excelente profissional tem uma produtividade, criatividade, habilidade de resolver problemas, etc. de 20 a 35 vezes maior do que um profissional medíocre. Independentemente de como se mediu esses números, de quão confiáveis são, de o que eles efetivamente medem, a observação qualitativa prática é que de fato excelentes profissionais são muito (ordem de grandeza) melhores do que os medíocres. Mas quantos desses excelentes profissionais existem? Será que possuem um talento inato, ou pode-se despertar de alguma forma este tipo de talento?

Outro tipo de pergunta: Certificados contribuem para despertar talentos individuais e empresariais? Certamente, organizar a forma de trabalhar das equipes trará benefícios, independentemente da forma dessa organização. Por outro lado, equipes talentosas talvez consigam auto-organizar-se, não necessitando de alguém que indique como devem trabalhar. Claro que organizar segundo recomendações de pessoas com experiência na área trará mais benefícios do que as inevitáveis sequências de tentativa e erro que seriam realizadas pelo inexperiente. De forma similar, ensinar a mecânica do uso de um software melhorará o desempenho do indivíduo que não conhece esta mecânica. Mas queremos essas pequenas melhorias, ou queremos melhorias significativas? Queremos melhorias que envolvam situações imprevistas, ou queremos trabalhar somente em coisas perfeitamente previsíveis?

A literatura de Informática, mesmo a científica, é muitas vezes viciada, uma vez que são muito raras as narrativas a respeito das coisas que deram errado ou não funcionaram como esperado. Ao ler-se artigos e livros tem-se muitas vezes a impressão de que são propaganda velada. Precisa-se, então, verificar se as promessas são reais, ou foram acasos observados pelos redatores. Como fazer isso? Como desconfiar sem que seja por ideologia ou princípio?

Neste seminário tenho por objetivo discutir propostas, em particular as que envolvem processos de software, à luz de medições e observações qualitativas. O objetivo não é apresentar mais uma solução

miraculosa, mas sim induzir uma discussão a respeito de como procurar soluções satisfatórias para cada caso e, também, a respeito de como proceder para despertar talentos e, dessa forma, gerar um número maior de excelentes profissionais. É minha convicção que é a excelência do profissional o fator preponderante e não tanto os certificados que ele e/ou a empresa em que trabalha venham a obter.

Arndt von Staa

É Professor Associado do Departamento de Informática da PUC-Rio. É PhD em Ciência da Computação (1974, Engenharia de Software) pela Universidade de Waterloo, Ontário, Canadá. Trabalha em computação desde 1962. Desenvolveu ferramentas de apoio ao desenvolvimento de software, arcabouços de apoio ao teste automatizado, sistemas de suporte e sistemas aplicativos técnicos e administrativos. Suas atividades de pesquisa e desenvolvimento mais recentes concentram-se em desenvolvimento e manutenção de software fidedigno. A pesquisa segue duas vertentes. Uma, a geração automatizada de massas de teste e a aplicação dessas massas a ferramentas de teste automatizadas. A segunda, a organização do trabalho através de métodos e processos de desenvolvimento de software ágeis apoiados por ambientes de desenvolvimento de software assistidos por computador. Idealizou, projetou e conduziu o desenvolvimento de Talisman, um meta-ambiente mono-usuário de engenharia de software assistido por computador. Esse meta-ambiente é composto por diversas ferramentas integradas e adaptáveis, fornecendo suporte mecanizado a uma grande parte das etapas do desenvolvimento, ao controle de qualidade, à manutenção e à gerência do desenvolvimento de sistemas de software. Mais especificamente, apóia a especificação, o projeto, a geração de programas, o planejamento e a documentação de sistemas de software, onde os componentes são desenvolvidos independentemente por um ou mais desenvolvedores. Orientou mais de 70 dissertações de mestrado e 8 teses de doutorado, publicou mais de 100 artigos em periódicos, e congressos com avaliação formal, 5 livros, e vários capítulos de livros. Coordenou diversos projetos de pesquisa e desenvolvimento em parceria com a indústria. Foi professor visitante nas universidades *University of Waterloo*, *University of California Irvine* e USP, tendo lecionado disciplinas em todas estas universidades. Foi agraciado com a inscrição na Ordem do Mérito Científico e Tecnológico na categoria de Comendador, área Tecnológica.

Change does not just “Happen” Building sustainable change in Organizations.

Jorge Luis Boria, M Eng

A proposal for theme for a keynote presentation at WAMPS 2010.

While the Capability Maturity Model Integration (CMMI®) has initiated a flurry of change, many organizations have struggled with its implementation because they have failed to recognize that resistance to change, and not immediate adoption of change, is the norm. This is not a fault of the model other efforts to perform “business reengineering” have failed as have most efforts to introduce CASE tools because they have neglected the human factor in change. What hinders adoption is that during the transition the old and the new coexist and people have a hard time “letting go” of the past. This conflict leads to loss of productivity. If the transition period is too long or too expensive, the change is abandoned and the improvement fails. The secret lies in how to bound and manage the transition. We will describe a set of tools that will help the change agent (or the similar but more powerful role of change facilitator) to better conduct a transition. Our presentation is based on three known models linked to transitions: The categories of adopters, as presented by Everett Rogers and made famous by the book “Crossing the Chasm” will be the first one. We will use the Building Commitment steps of Conner and Patterson, to describe how to handle each phase in the change, and we will present a model by Kim Cameron called the Competing Values Framework to diagnose and change organizational culture. In the end, we tie all these tools into a framework that allows the change facilitator to work within the culture and with the culture, introducing change gradually and supporting it.

About the author

Mr. Boria holds a Masters of Engineering in Computer Science from Cornell University. He is a Certified High Maturity SCAMPI Lead Appraiser for SCAMPI A and an Authorized Trainer of the Introduction to the CMMI (DEV and SVC constellations) class by the Software Engineering Institute (SEI) of Carnegie-Mellon University. He works for the SEI as a Visiting Scientist, in the role of Qualified Observer of Candidate Lead Appraisers in the Quality Group and Candidate Instructors in the Instruction Group. Mr. Boria has performed a range of roles from programmer to technical program manager in diverse domains including embedded software, operating systems, compiler construction, MIS, and general IT applications. Mr. Boria is a popular presenter with over 200 activities at conferences, seminars, and as invited speaker. He has authored two published books on Software Development. He is fluent in English, Spanish, and Portuguese.

Programa MPS.BR

Avanços, conquistas e resultados alcançados

José Antonio Antonioni
 Diretor de Qualidade e Competitividade
 SOFTEX
 Campinas – São Paulo, Brasil
 jaa@nac.softex.br

Desde seu início, em dezembro de 2003, o Programa MPS.BR tem tido uma trajetória de sucessos e realizações com resultados muito acima das expectativas iniciais. Dados objetivos demonstram que esta iniciativa da SOFTEX responde, efetivamente, a uma necessidade crítica das empresas de software e serviços relacionados no que tange a melhoria da qualidade de seus processos de software e correspondente incremento de sua competitividade no mercado global. Trata-se de um programa de interesse da sociedade brasileira, de abrangência nacional que conta com a participação de representantes da academia, do governo e do setor privado, envolvendo um grande número de profissionais qualificados e estudantes que disseminam o Programa MPS.BR e o Modelo MPS em todas as regiões do Brasil.

O Programa foi planejado em 03 (três) períodos: Implantação (2004-2007); Consolidação (2008-2011) e Internacionalização (2012-2015), estando, portanto, em fase de conclusão de seu segundo período. Neste trabalho apresentamos alguns dados recentes, atualizados em outubro de 2010, visando evidenciar o trabalho realizado e os resultados obtidos nos últimos 12 meses, desde outubro de 2009, quando foi realizado o WAMPS 2009:

Entre os resultados obtidos destacam-se:

- 58 (cinquenta e oito) empresas tiveram a avaliação MPS segundo o Método de Avaliação MA-MPS publicadas desde o último WAMPS, em outubro de 2009. Com estas avaliações atingiu-se o total de 233 (duzentas e trinta e três) empresas avaliadas MPS desde a primeira avaliação realizada, em setembro 2005. A figura 1 mostra o número de avaliações já realizadas por nível.

Totais por Níveis								
Ano	A	B	C	D	E	F	G	Totais por Ano
2005	0	0	0	0	1	3	1	5
2006	2	0	0	1	1	1	7	12
2007	1	0	0	0	1	12	41	55
Total 2005 a 2007	3	0	0	1	3	16	49	72
2008	1	0	0	0	1	9	40	51
2009	2	0	2	0	2	33	41	80
2010	0	0	5	0	0	7	18	30
Total 2008 a 2010	3	0	7	0	3	49	99	161
TOTAIS	6	0	7	1	6	65	148	233

Figura 1. Empresas avaliadas MPS: 2005-2010

- 31 (trinta e uma) avaliações encontram-se em andamento e temos a expectativa de chegarmos a 300 empresas avaliadas MPS, ainda em 2010.
- Alguns estados tiveram este ano as primeiras empresas avaliadas: Amazonas, Alagoas e Rio Grande do Norte.
- Neste momento o Programa MPS.BR conta com 18 (dezoito) Instituições Implementadoras, 12 (doze) Instituições Avaliadoras, 1 (uma) Instituição de Consultoria de Aquisição e 13 (treze) Instituições Organizadoras de Grupos de Empresas. São 361 implementadores credenciados (sendo 124 alocados em Instituições Implementadoras), 99 avaliadores (sendo 63 alocados em Instituições Avaliadoras) e 6 Consultores de Aquisição.
- Tem-se implementadores e avaliadores em todas as regiões do país, embora ainda sejam necessários esforços na formação de implementadores e avaliadores em alguns estados onde os recursos humanos MPS são, ainda, escassos ou inexistentes.
- Desde o início do programa, a SOFTEX investiu R\$ 12,2 milhões captados junto as diversas fontes de fomento nacionais e internacionais. Em 2010, o investimento no Programa atingiu até a presente data R\$ 2,5 milhões em recursos do Ministério de Ciência e Tecnologia através da FINEP para apoio a grupos de empresas o que permitirá à SOFTEX apoiar 110 (cento e dez empresas) sendo 70 empresas para os níveis G e F (base da pirâmide) e 40 empresas para os níveis E, D e C (meio da pirâmide).
- Em 2009 realizamos a primeira avaliação conjunta MPS/CMMI, cujo relato encontra-se publicado no site da SOFTEX/MPS e no site do SEI/CMMI. Neste momento estão em andamento uma avaliação conjunta MPS C - CMMI 3 e uma avaliação MPS E - CMMI 2.
- Foi elaborado o código de ética do MPS, criada a comissão de ética e eleitos os seus primeiros representantes.
- Foi ampliado o *Board* de *senior advisors* da ETM (Equipe Técnica do Modelo MPS), do qual fazem parte agora: Prof. Arndt von Staa da PUC-Rio, Prof. Guilherme Travassos da COPPE/UFRJ, Jorge Boria da Liveware, Prof. Jorge Audy da PUC-RS e Profa. Shari Pfleeger do Dartmouth College.

Algumas ações importantes foram iniciadas neste último ano:

- Após intensos trabalhos de preparação de ementas, estamos prontos para iniciar, em convênio com Agentes SOFTEX e Universidades em diversos locais do país, um curso de pós-graduação *latu sensu* em Engenharia e Qualidade de Software com foco nos processos MPS. As primeiras edições desse curso já estão sendo trabalhadas, em diferentes cidades brasileiras, para serem oferecidas no primeiro semestre de 2011.
- Temos aprovado e estamos iniciando o novo Projeto RELAIS com o BID, desta vez em conjunto com México, Colômbia e Peru. Um dos objetivos do Projeto RELAIS é levar o Modelo

MPS a estes países latino americanos, o que consideramos o primeiro passo para a internacionalização do MPS. Como consequência incluímos uma nova área na Equipe Técnica do MPS, a área de Apoio à Internacionalização. Também reestruturamos a equipe de apoio operacional criando a posição da Gerência de Operações Internacionais.

- Estamos, também, finalizando a implementação da Comunidade de Prática do MPS, que deverá estar disponível para seus usuários já no início do próximo ano.

Finalmente é importante assinalar os fatores críticos para o sucesso do Programa MPS:

- O apoio efetivo do governo federal através do MCT - Ministério das Ciência e Tecnologia e da FINEP - Financiadora de Estudos e Projetos.
- O apoio do BID (Banco Interamericano de Desenvolvimento), em um primeiro projeto que permitiu a implantação do MPS em 77 empresas (onde 71 foram avaliadas = 92% de sucesso total) e agora através do Projeto RELAIS, primeiro passo rumo à internacionalização.
- A interação universidade-empresa-governo, liderada pela SOFTEX.

Sobre o autor

José Antonio Antonioni é engenheiro eletricitista formado pela Escola de Engenharia da Universidade Federal do Rio Grande do Sul – UFRGS com especialização em Software Marketing na Nova Southeastern University, Fort Lauderdale, Florida, USA.

Atua na área de Tecnologia da Informação e da Comunicação (TIC), desde 1981, tendo trabalhado por 12 anos como engenheiro de testes e qualidade de software na EDISA-HEWLETT-PACKARD (HP), onde participou de projetos de pesquisa e desenvolvimento atuando na HP de Boise, Idaho, USA e na HP de Guadalajara, Jalisco, México.

Em 1994, publicou o livro “Qualidade em Software: Manual de Aplicação da ISO-9000”, editado pela MAKRON Books, São Paulo.

Atualmente é diretor de Qualidade e Competitividade da SOFTEX - Associação para Promoção da Excelência do Software Brasileiro, onde dirige programas e projetos voltados ao incremento da competitividade da indústria brasileira de software, em especial, os relacionados à implementação das melhores práticas como o Programa MPS.BR.

Resultados Iniciais iMPS 2010: Variação de Desempenho nas Empresas que Adotaram o Modelo MPS

Guilherme Horta Travassos e Marcos Kalinowski (gght, mkali)@cos.ufrj.br)

Programa de Engenharia de Sistemas e Computação
COPPE/UFRJ
Caixa Postal 68511 – CEP 21945-970 – Rio de Janeiro, Brasil

Resumo

Este artigo apresenta uma síntese dos resultados iniciais da rodada de 2010 do projeto iMPS (informações para acompanhar e evidenciar variação de desempenho nas empresas que adotaram o Modelo MPS). Neste terceiro ano consecutivo de coleta de dados, os resultados são apresentados sob três perspectivas: caracterização, análise de variação 2009/2010 e análise de variação 2008/2009/2010. De forma geral, a satisfação das empresas com o modelo é notória, com mais de 92% das empresas se dizendo parcialmente ou totalmente satisfeitas. Em relação ao retorno de investimento da adoção do modelo, mais de 72% das empresas informaram ter recuperado mais do que o investimento feito na implementação e avaliação do modelo. Além disso, assim como em 2009, principalmente para aquelas empresas que evoluíram ou internalizaram o MPS em seus processos, foi possível observar tendências a redução de custos, aumento da qualidade, redução de prazos e aumento de produtividade.

1. Introdução

Em dezembro de 2003 foi criado o programa MPS.BR, representando uma iniciativa para melhorar a capacidade de desenvolvimento de software nas empresas brasileiras. Seu objetivo é desenvolver e disseminar um modelo de melhoria de processos (o modelo de referência MPS) visando estabelecer um caminho economicamente viável para que organizações, incluindo as pequenas e médias empresas, alcancem os benefícios da melhoria de processos e da utilização de boas práticas da engenharia de software em um intervalo de tempo razoável.

O modelo foi desenvolvido levando em consideração normas internacionais, modelos internacionalmente reconhecidos, boas práticas da engenharia de software e as necessidades de negócio da indústria de software brasileira. Em relação a empresas avaliadas, até setembro de 2010 contava-se com 233 avaliações MPS publicadas. Os resultados destas avaliações estão disponíveis na seção Avaliações em www.softex.br/mpsbr.

Tendo em vista a adoção do modelo MPS pelas empresas brasileiras, revela-se o interesse por compreender qualitativamente os resultados de desempenho obtidos por estas empresas, tais como prazo, produtividade, custo e qualidade. Com este objetivo, o projeto iMPS (informações para acompanhar e evidenciar variação de desempenho nas empresas que adotaram o modelo MPS) foi iniciado em 2007 junto ao Grupo de Engenharia de Software Experimental (<http://ese.cos.ufrj.br>) da COPPE/UFRJ.

O objetivo global do iMPS é planejar um *survey*, seguindo os princípios da Engenharia de Software Experimental, e periodicamente executá-lo para acompanhar e evidenciar resultados de desempenho nas empresas de software que adotaram o modelo MPS. Maiores detalhes sobre o plano da pesquisa, os momentos de captura das informações e o tratamento dado às ameaças à validade podem ser encontrados em [Kalinowski et al., 2008]. As rodadas de 2008 (*baseline*) e 2009 do iMPS forneceram evidências objetivas iniciais [Travassos e Kalinowski, 2008] [Travassos e Kalinowski, 2009], a serem complementadas anualmente por outras rodadas iMPS que permitirão análises comparativas.

Este artigo apresenta uma síntese dos resultados iniciais da rodada de 2010. Neste ano, o preenchimento dos questionários se deu em formato eletrônico (através de uma aplicação Web), o que ajudou a reduzir significativamente problemas nos dados informados (que antes eram fornecidos em formato de texto livre). Os resultados serão apresentados sob três perspectivas: caracterização 2010, análise de variação 2009/2010 e análise de variação 2008/2009/2010. O objetivo da caracterização é delinear o desempenho das empresas que adotaram o MPS em 2010. O da análise da variação 2009/2010, por sua vez, é observar a variação do desempenho, entre 2009 e 2010, das empresas que adotaram o MPS. Por fim, o objetivo da análise de variação 2008/2009/2010 é observar a variação do desempenho, entre 2008 e 2010, das empresas que adotaram o MPS e possuem avaliações vigentes desde 2008. É importante ressaltar que, para análises de variação a empresa é comparada somente com ela mesma e que dados de desempenho individuais das empresas não são divulgados.

O restante deste artigo está organizado da seguinte maneira. Na seção 2, os resultados preliminares da caracterização 2010 são apresentados. Na seção 3, os resultados preliminares da análise de variação 2008/2009 são apresentados. Na seção 4, os resultados preliminares da análise de variação 2008/2009/2010 são apresentados. As considerações finais do artigo são apresentadas na seção 5.

2. Resultados iMPS: Caracterização 2010

A caracterização visa delinear o desempenho das empresas que adotaram o MPS em 2010. Assim, apenas dados de um ano da pesquisa foram utilizados (entre 01/08/2009 e 31/07/2010) nesta análise. No total, questionários de 156 empresas diferentes (23 iniciando a implementação MPS, 11 em processo de avaliação, 79 avaliadas MPS nível G, 36 avaliadas MPS nível F e 7 avaliadas MPS níveis E-A) foram utilizados.

Tendo em vista a concentração da maioria das empresas ainda nos níveis iniciais de maturidade, optou-se por dividir o conjunto de dados nas seguintes 4 categorias: Empresas Iniciando a Implementação, Empresas em Processo de Avaliação, Empresas Avaliadas em Nível de Maturidade G, Empresas Avaliadas em Nível de Maturidade F e Empresas Avaliadas em Níveis de Maturidade E-A. Como a análise de caracterização agrega dados de diferentes empresas, é natural que as medidas apresentem desvios padrão muito altos. Assim, acreditamos que a mediana, representando o valor central para a medida, possa fornecer informação mais adequada para a caracterização das empresas. Adicionalmente, durante a preparação dos dados, medidas com valores a mais de três desvios padrão da média (*outliers*) foram descartadas até que o conjunto final de dados não contivesse mais medidas nesta situação. Desta forma foi possível aproveitar o máximo possível de respostas e ao mesmo tempo não influenciar os resultados com dados eventualmente distorcidos. Neste processo foi possível identificar que a maioria dos *outliers* se encontrava nas empresas iniciando a implementação ou no

nível G, onde o desvio padrão das medidas também se mostrava maior. Isto pode estar relacionado com o fato de o processo de medição ser do nível F do MPS, o que nos leva a acreditar que os resultados das medidas das empresas sejam mais confiáveis a partir deste nível de maturidade. Seguem alguns resultados preliminares desta dimensão de análise.

Satisfação com o Modelo MPS. Em relação à satisfação das empresas com o modelo MPS, 64,74% (101 empresas) relataram estar totalmente satisfeitas com o modelo e 28,21% relataram estar parcialmente satisfeitas. Apenas 0,64% (1 empresa, do nível G) relatou estar insatisfeita e 6,41% (10 empresas) informaram ainda não conhecer o seu nível de satisfação. Este resultado demonstra um quadro de ampla satisfação.

Tamanho dos Projetos. Em relação ao tamanho dos projetos, das 156 empresas consideradas, 40 (25,64%) mencionaram medir o tamanho de seus projetos em pontos de função. Esta foi a medida de tamanho mais utilizada, seguida por pontos de caso de uso, informada por 19 empresas (12,18%).

Figura 1. Mediana do tamanho dos projetos. A Figura 1 apresenta as medianas do tamanho médio dos projetos das empresas que utilizam pontos de função para cada agrupamento utilizado no estudo.

Enquanto a mediana do tamanho para empresas iniciando a implementação é de 45 pontos de função (apenas duas empresas desse grupo informaram o tamanho de seus projetos em pontos de função), a mediana para as empresas nos níveis E-A é de 215. Existe uma correlação positiva entre o aumento da mediana e o aumento do nível de +0,80. Um comportamento parecido pode ser observado na caracterização de 2009 (Travassos e Kalinowski, 2009)

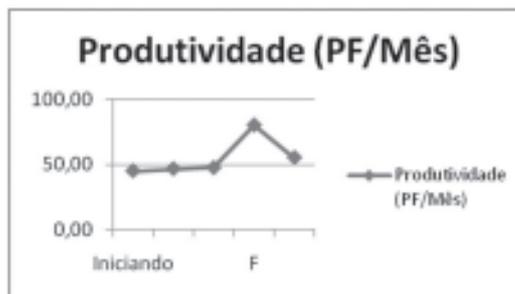


Figura 1. Mediana do tamanho dos projetos

Precisão de Estimativa de Prazo. Como muitas empresas informaram que o tempo médio gasto nos projetos era igual ao prazo dos projetos (ou seja, precisão de estimativa 1), acreditamos que esta variável seja melhor observada olhando a variação dentro de cada conjunto de empresas.

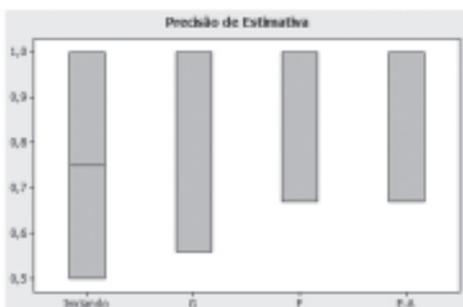


Figura 2. bloxpot da Precisão de Estimativa

A Figura 2 ilustra esta variação, através de um *boxplot*, que destaca os valores máximo, mínimo e a mediana. Enquanto a mediana apresenta variação pequena (tem valor 1 para as empresas dos níveis G, F e E-A), as empresas de níveis de maturidade F e E-A apresentaram uma menor variação e uma precisão de estimativa mínima maior (ambas variando entre 0,67 e 1). Ou seja, assim como na caracterização de 2009, as empresas de maior maturidade informaram maior precisão nas estimativas.

Produtividade. Este ano, novamente, a produtividade se apresentou maior para as empresas que adotaram o MPS. A maior mediana foi das empresas do nível F. Entretanto, é importante ressaltar que a produtividade está sendo observada de forma isolada e que a produtividade se mostra naturalmente diferente de acordo com o tipo de projeto e as expectativas em relação à qualidade¹ e ao custo². Adicionalmente, o cálculo da produtividade leva em consideração outras medidas base que, conforme discutido anteriormente, podem ser mais confiáveis para empresas a partir do nível F, que devem possuir um processo de medição institucionalizado.

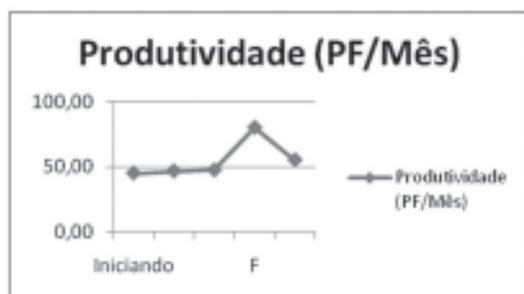


Figura 3. Mediana da Produtividade (em PF/Mês)

A Figura 3 apresenta as medianas da produtividade dos projetos das empresas que utilizam pontos de função para cada agrupamento utilizado no estudo. Enquanto a mediana da produtividade para empresas iniciando a implementação é de 45 pontos de função por mês, a mediana para as empresas nos níveis E-A é de 55,33 e para o nível F chegou a 80,36 pontos de função por mês. Existe uma correlação positiva entre o aumento da mediana e o aumento do nível de +0,58.

Tendo apresentado este resumo de alguns dos resultados da caracterização das empresas em 2010, a seção seguinte apresentará a variação de desempenho entre 2009 e 2010 das empresas que adotaram o MPS.

3. Resultados iMPS: Análise da Variação 2009/2010

Para análise dos dados enviados pelas empresas que responderam ao questionário periódico de 2010 e já haviam também fornecido informações para o questionário periódico em 2009 foi utilizado o mesmo conjunto de critérios, com análise e eliminação de pontos extremos (*outliers*). Os indicadores que foram considerados nesta análise preliminar estão definidos no plano de estudo do iMPS [Travassos e Kalinowski, 2008]: A. *Variação do Faturamento*, B. *Número de Clientes no País*, C. *Número de Funcionários*, D. *Custo Médio dos Projetos*, E. *Prazo Médio dos Projetos*, F. *Tamanho Médio dos Projetos*, G. *Produtividade*, H. *Qualidade*. Ainda de acordo com o plano, além destes indicadores, o Retorno do Investimento para implementar e avaliar o modelo, desde a avaliação, é apresentado. Este ano foi considerado um conjunto com 65 empresas que possuem um questionário periódico para o ano de 2009 e outro para o ano de 2010. Em complemento, um novo conjunto com 11 empresas pôde ser identificado e analisado a parte, contendo as empresas que renovaram/mudaram de nível nesse período e responderam a ambos os questionários periódicos.

¹ A qualidade é capturada nos questionários em função do número de defeitos por unidade de tamanho. Como muitas empresas tratam defeitos de uma forma distinta estas respostas são consideradas somente na dimensão de análise de variação, ou seja, comparando a empresa com ela mesma no decorrer do tempo.

² O custo é capturado nos questionários em função de um percentual do faturamento, servindo como base de comparação da empresa com ela mesma para a análise de variação.

O cálculo dos indicadores seguiu rigorosamente as fórmulas definidas no plano do estudo do iMPS. Em adição, a interpretação dos resultados associados aos indicadores teve como base as premissas de comportamento apregoadas na Engenharia de Software para projetos de software, que se diferenciam naturalmente dos processos produtivos tradicionais. Por exemplo, o conceito de produtividade no contexto iMPS se refere a *'tamanho médio de projeto dos últimos 12 meses / tempo médio gasto nos projetos dos últimos 12 meses'*, portanto relacionando exclusivamente características de projeto de software e apresentando uma representação simplificada se comparada ao conceito usual de produtividade utilizada em processos produtivos.

A seguir serão apresentados os resultados iniciais obtidos a partir dos dados. Conforme definido no iMPS, os dados são sempre coletados de forma a não permitir comparação competitiva entre as empresas. Por estar tratando de variação de desempenho o valor individual do indicador de cada empresa faz sentido apenas para a própria empresa, perdendo significado se ocorrer tentativa de comparação. Para observar o comportamento foi gerada então uma distribuição contendo 3 faixas de valores para categorizar o desempenho das empresas em cada indicador. Estas faixas representam o percentual relativo de empresas (baseado no número de respostas válidas) que tiveram aumento, redução ou não sofreram alteração.

A avaliação do significado do impacto do aumento ou redução de um indicador depende do indicador e, em algumas situações, pode ser relacionada com outro indicador. Por exemplo, espera-se que o custo médio dos projetos reduza ao mesmo tempo em que a produtividade aumente. Portanto, neste caso, tanto a redução quanto o aumento representam impactos positivos para as empresas em análise. Por isso, acreditamos que apresentar as tendências de comportamento das empresas pode ajudar a ter uma melhor compreensão geral dos benefícios do MPS ao mesmo tempo em que pode indicar os pontos onde exista necessidade de investir esforços para aprimorar o rendimento geral do modelo. O nível de confiança [Gardner e Altman, 1989] para as respostas referentes a cada indicador foi calculado considerando-se a população como sendo o número total de questionários válidos para cada grupo e a amostra sendo o número de respostas válidas para cada questão. A finalidade é tentar sugerir o quanto o comportamento descrito pelo indicador poderia estar representando o comportamento das empresas considerando o grupo específico sob análise.

Como se pode observar na Figura 4, os resultados gerais indicam tendências interessantes em relação às empresas que adotaram o modelo (e enviaram os questionários). Por exemplo, é possível perceber que as empresas entre os anos de 2009 e 2010 reportaram tendência no aumento do Faturamento e Número de Clientes no País e pequena tendência de aumento no Número de Funcionários e na Qualidade (capacidade de encontrar defeitos). O cálculo do indicador é realizado através de comparação do número de defeitos identificados em cada ano pela empresa por unidade de tamanho do software desenvolvido. Entretanto, a questão do comportamento do indicador Qualidade precisa ser analisada de forma mais detalhada.

O Tamanho dos Projetos das empresas se manteve aproximadamente igual, com o número de empresas que tiveram redução do tamanho aproximadamente igual ao de empresas que tiveram aumento. É possível observar ainda que a maioria das empresas informou ter tido redução do Custo Médio e do Prazo Médio de seus projetos. Isto contrasta com a Produtividade, em que um número sutilmente maior de empresas informou ter apresentado redução. Entretanto, análises adicionais precisam ser realizadas para identificar se o impacto é positivo ou negativo, pois, aparentemente ocorreu alteração na capacidade de identificação de defeitos e aumento no número de funcionários.

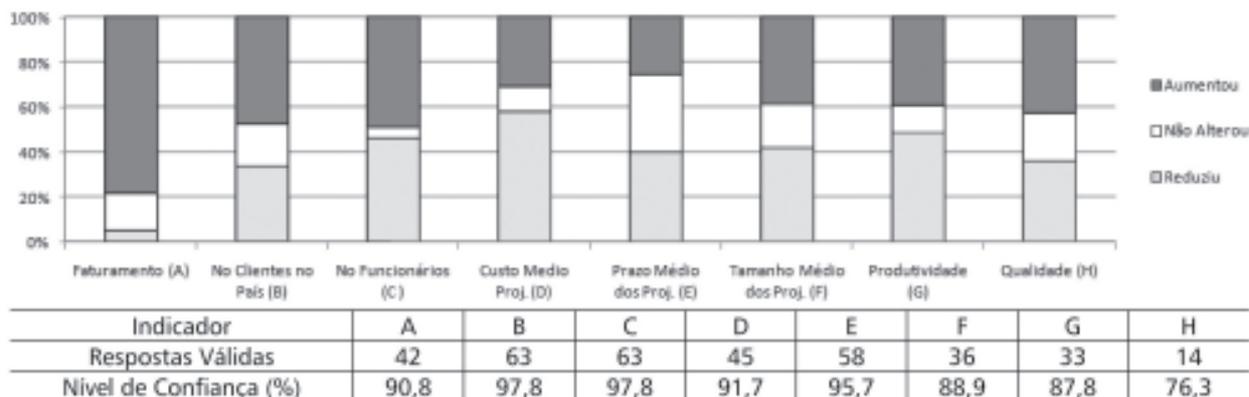


Figura 4. Variação de Desempenho das 65 Empresas que Adotaram o MPS e Participaram da Pesquisa Periódica iMPS em 2009 e 2010

Em relação ao retorno de investimento obtido por estas empresas, 18 empresas haviam fornecido informações relativas a custos de implementação e avaliação (informações coletadas no contexto da pesquisa iMPS durante o processo de avaliação) e a variação do faturamento (informação coletada no questionário periódico), representando um nível de confiança de 80,0%. Entre estas, 72,2% já aumentaram seu faturamento o suficiente para recuperar completamente o investimento realizado na implementação e avaliação do MPS (ROI > 100%). Outros 11,11% já recuperaram parcialmente o investimento realizado.

Assim como em 2009, no ano de 2010 foi realizada uma análise adicional, considerando os resultados da variação de desempenho das empresas que mudaram ou revalidaram seus níveis de maturidade junto ao MPS. A característica principal destas empresas, independentemente do nível a que estejam avaliadas, se refere à adoção do MPS e continuidade do desenvolvimento seguindo as diretrizes oferecidas por ele. Conforme pode ser visto na Figura 5, entre estas empresas a grande maioria obteve um aumento no faturamento (de fato nenhuma informou ter reduzido o faturamento).

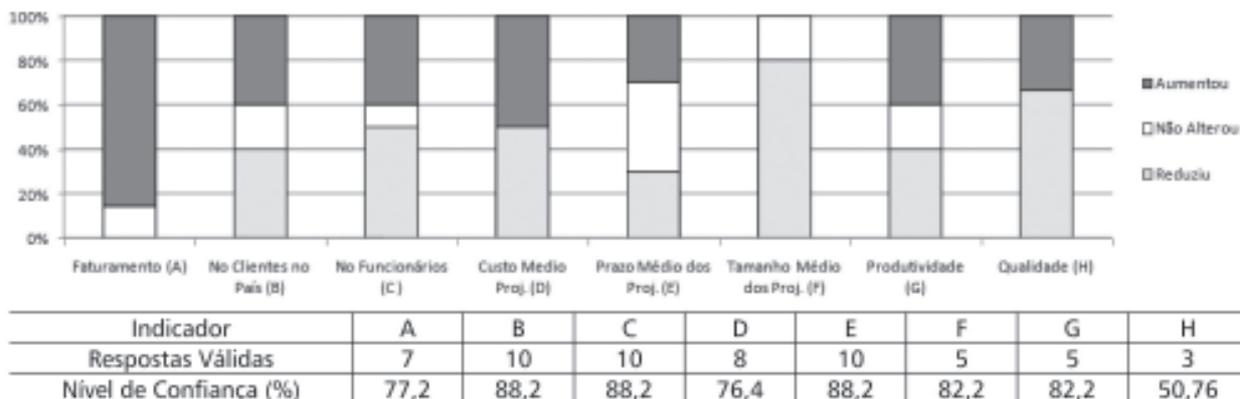


Figura 5. Variação de Desempenho de 9 Empresas com MPS que Revalidaram/Mudaram de Nível

É possível observar ainda que, de acordo com os dados fornecidos pelas empresas, os indicadores apresentam comportamento coerente com algumas hipóteses associadas à utilização de processos de desenvolvimento de software combinado com boas práticas da engenharia de software. Por exemplo, é possível observar a redução de custos e prazos. Entretanto, é possível observar também que estas empresas passaram a lidar com projetos menores, o que pode refletir uma maneira diferente de organizar o desenvolvimento de produtos e soluções em projetos, visando sistematização e maior controle. É forte a indicação que a redução no tamanho dos projetos possa contribuir para justificar a redução de custo e prazo, o que é reforçado pelo fato de a Produtividade aparentemente não ter sofrido alterações.

O indicador relacionado à Qualidade não apresentou informação suficiente que permitisse uma observação mais aprimorada da tendência e comportamento das empresas. Apenas 3 empresas apresentaram resultados passíveis de interpretação porém com nível de confiança muito baixo, aumentando consideravelmente os riscos em generalizar o comportamento observado para toda a população. Entretanto, a título de ilustração, os resultados estão apresentados na Figura 5. Uma investigação adicional precisa ser realizada no sentido de se tentar identificar possíveis fatores de confusão que possam estar influenciando a baixa disponibilidade de informação para este indicador.

A sessão seguinte apresenta a análise de variação para empresas que mantiveram avaliações vigentes do modelo MPS nos últimos três anos (2008/2009/2010).

4. Resultados iMPS: Análise da Variação 2008/2009/2010

Os 75 questionários válidos (com 1 resposta válida para cada ano de avaliação) incluem 25 empresas em 2 níveis MPS: G (11 empresas) e F (14 empresas). Empresas em outros níveis não apresentaram respostas adequadas para os indicadores iMPS em um dos anos do período avaliado e por isto não estão sendo incluídas neste conjunto. Considerando os questionários identificados como válidos, nem todas as empresas responderam a todas as questões adequadamente em um dos anos considerados. Desta forma, não é possível utilizar a mesma perspectiva para observar os resultados destas 25 empresas para os diferentes indicadores. Por isso, como feito anteriormente, para cada indicador iMPS é apresentado o Nível de Confiança calculado (Tabela 1), que intenciona fornecer um nível de segurança que permita aprimorar a percepção sobre os riscos envolvidos na interpretação dos resultados e em sua generalização para comparação com situações semelhantes no dia a dia de trabalho.

Indicador	Respostas Válidas	Nível de Confiança
Varição Faturamento	9	73,33
Número de Clientes no País	23	94,10
Número de Funcionários	23	94,10
Custo Médio Projeto	16	85,00
Prazo de Projeto	23	94,10
Tamanho Médio dos Projetos	14	82,27
Produtividade	13	80,78
Qualidade	9	73,33
Faturamento com Exportação	18	87,53

Tabela 1. Nível de Confiança para os indicadores iMPS

O cálculo dos indicadores utilizou o conceito de correlação. Os dados de uma dada empresa foram tratados entre si. Não há comparação, conforme previsto no plano iMPS, dos dados de uma empresa com outra. Desta forma, para observar a evolução dos indicadores, realizou-se a correlação entre a data de preenchimento do questionário (como fator Tempo) e cada uma das características da empresa (por exemplo, número de funcionários, número de clientes, dentre outras). Portanto, o mapeamento realizado está relacionado a representar o aumento (correlação positiva), estabilização (correlação nula) ou redução (correlação negativa) relatada pelas empresas ao longo do período de avaliação. A partir do cálculo da correlação para cada característica das empresas, identificou-se o percentual de empresas que apresentaram tendência de aumento, estabilização ou redução, que foi então usado para gerar o gráfico da Figura 6, conforme os níveis de confiança e número de respostas válidas apresentados na Tabela 1.

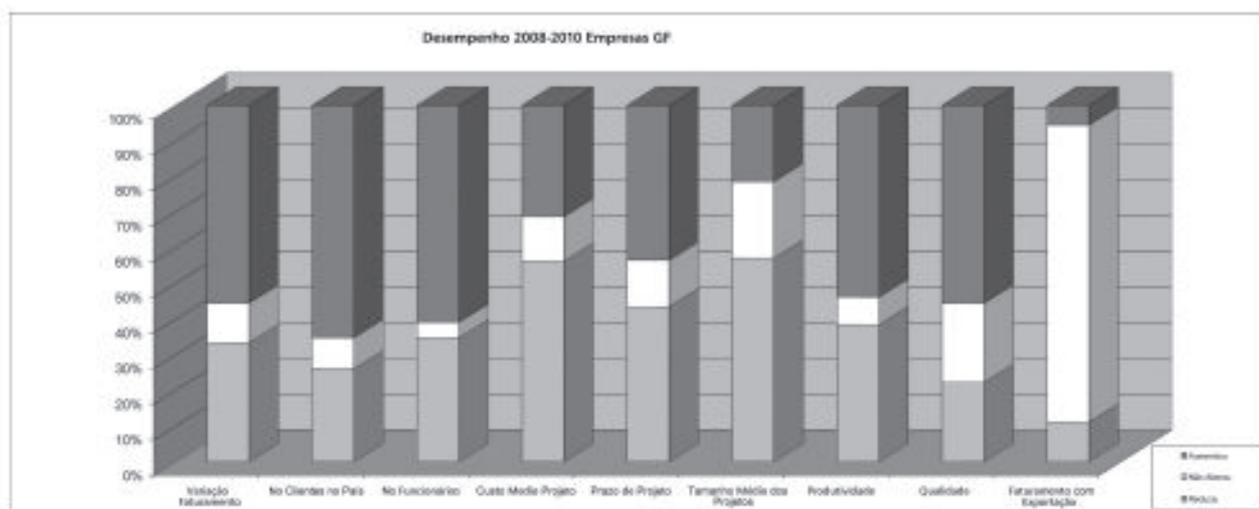


Figura 6. Comportamento de 25 Organizações níveis G e F MPS de 2008 a 2010

Para Variação do Faturamento, conforme pode ser observado na Figura 6, a tendência das empresas ao longo deste período foi de aumento, ou seja, aparentemente as empresas reportaram ter conseguido incrementar a entrada de recursos. É interessante observar que este aumento pode ter tido contribuição do aumento do Número de Clientes no País, que pode também ter influenciado positivamente o aumento no Número de Funcionários. Considerando que o período de avaliação envolveu uma série de eventos financeiros mundiais que colocaram diversas empresas clientes (e as próprias empresas) em situação de risco, identificamos este comportamento geral como positivo durante este período. Entretanto, a capacidade de generalização é reduzida para outras empresas em diferentes níveis e perfis que utilizam o MPS. Em complemento, é necessário realizar comparação com o desempenho de empresas com perfil semelhante e ainda não usuárias do MPS, para identificar a influência do MPS neste comportamento observado.

Em relação aos projetos de software usualmente desenvolvidos por estas empresas, é possível perceber que o Custo Médio do Projeto apresenta tendência de redução, com tendência de aumento da Produtividade e aparente redução do Tamanho Médio. Nota-se também que os projetos apresentaram uma aparente melhora da Qualidade, com as empresas tendendo a uma maior capacidade de identificar defeitos nos projetos. Este comportamento é aderente aos princípios da Engenharia de Software e já observados na última rodada do iMPS em 2009. Entretanto, ao observar o comportamento dos Prazos de Projeto nota-se que não existe indicação explícita de tendência. Algumas organizações tendem a

aumentar os prazos, outras a reduzir e algumas poucas mantêm inalterado os prazos ao longo do período de avaliação. Seria razoável esperar que, tendo em vista a melhoria nos indicadores de custo, prazo e produtividade, os prazos de projeto também apresentassem tendência de redução. Uma possível explicação para este comportamento pode estar associado a, por exemplo, o aumento do número de funcionários que, nos casos que sejam inseridos nos projetos de software, podem estar afetando os indicadores durante o período de aprendizagem; ou num possível aumento de tamanho dos projetos de software que ainda não foi possível investigar. Independentemente da causa consideramos este cenário bastante positivo, pois existe melhoria relatada pelas organizações em benefício da qualidade dos projetos de software que vem sendo desenvolvidos para o mercado brasileiro, conforme pode se perceber pelo Faturamento com Exportação que se mantêm inalterado de acordo com a informação fornecida pelas organizações.

5. Considerações Finais

Neste artigo foram apresentados os resultados iniciais referentes a caracterização e avaliação da variação do desempenho das empresas em função da adoção do modelo MPS observados a partir dos dados coletados em 2010 no contexto do projeto iMPS.

Para permitir mostrar o comportamento das empresas, a avaliação foi realizada a partir de três perspectivas: caracterização em 2010, análise de variação das empresas de 2009 a 2010 e análise de variação das empresas de 2008 a 2010. Utilizando estas perspectivas de observação nota-se indicações de tendência favorável nos diferentes indicadores tratados no iMPS.

De forma geral, a satisfação das empresas com o modelo é alta, com mais de 92% das empresas se dizendo parcialmente ou totalmente satisfeitas. Em relação ao retorno de investimento da adoção do modelo, mais de 72% das empresas informaram ter recuperado mais do que o investimento feito na implementação e avaliação do modelo. Além disso, assim como em 2009, principalmente para aquelas empresas que evoluíram ou internalizaram o MPS em seus processos, foi possível observar tendências a redução de custos, aumento da qualidade, redução de prazos e aumento de produtividade, comportamento esperado quando se aplica os princípios da Engenharia de Software nos projetos.

Agradecimentos

Este trabalho não teria sido possível sem a participação das empresas e dos profissionais Kival Chaves Weber (Coordenador Executivo do Programa MPS.BR), Nelson Henrique Franco de Oliveira e André Luis Chamelet Sotovia (Gerência de Operações do MPS.BR), aos quais agradecemos imensamente pela contribuição.

Bibliografia

Gardner, M.J; Altman, D. G. (1989), "Statistics with Confidence: confidence intervals and statistical guidelines". London: BMJ Publishing Group.

Kalinowski, M., Weber, K. and Travassos, G.H. (2008). iMPS: An Experimentation Based Investigation of a Nationwide Software Development Reference Model. ACM/IEEE 2nd International Symposium on Empirical Software Engineering and Measurement. October, 9-10. Kaiserslautern. Germany.

Travassos, G. H. e Kalinowski, M. (2009). iMPS 2009: Caracterização e Variação de Desempenho de Organizações que Adotaram o Modelo MPS. – Campinas, SP: SOFTEX, 2009 (ISBN 978-85-99334-11-9) (disponível em <http://www.softex.br/mpsbr/>)

Travassos, G. H. e Kalinowski, M. (2008). iMPS: Resultados de desempenho de empresas que adotaram o modelo MPS. – Campinas, SP: SOFTEX, 2008 (ISBN 978-85-99334-11-9) (disponível em <http://www.softex.br/mpsbr/>)

Sobre os autores

Guilherme Horta Travassos é doutor em Engenharia de Sistemas e Computação pela COPPE/UFRJ e realizou estágio de pós-doutorado em Engenharia de Software Experimental na University of Maryland-College Park. Professor de Engenharia de Software do Programa de Engenharia de Sistemas e Computação da COPPE/UFRJ. Pesquisador 1D CNPq. Líder do Grupo de Engenharia de Software Experimental. Atualmente é Diretor de Planejamento e Administração da COPPE/UFRJ, membro da ISERN e da Comissão de Educação da SBC – Sociedade Brasileira de Computação. Atua em projetos de P&D com a indústria através da Fundação COPPETEC. Informações adicionais podem ser obtidas em <http://www.cos.ufrj.br/~ght>.

Marcos Kalinowski é mestre e doutorando em Engenharia de Software pela COPPE/UFRJ. Bacharel em Ciência da Computação pela UFRJ. Membro do Grupo de Engenharia de Software Experimental da COPPE/UFRJ. Instrutor, Implementador, Avaliador e membro da equipe técnica do MPS.BR, sendo afiliado à Instituição Implementadora e Avaliadora COPPE/UFRJ e tendo participado da avaliação de diversas empresas em diferentes estados do país. Coordenador do curso de Engenharia da Computação da UVA RJ. Professor da pós-graduação e-IS Expert do NCE/UFRJ. Diretor da Kali Software desde 2004, tendo participado de treinamentos e consultorias em engenharia de software para empresas de diferentes portes, dentro e fora do país.

Confiabilidade de Software: Visão Geral

Odair Jacinto da Silva (osilva@in3software.com)
IN3 Software – Campinas – SP – Brasil

Resumo

O tamanho e a complexidade das aplicações de software têm crescido muito ao longo dos anos. A alta demanda por software de apoio às atividades econômicas e científicas tem causado a criação de aplicações cada vez mais complexas. A velocidade em que esta complexidade cresce é, por vezes, maior do que a velocidade do desenvolvimento da tecnologia para construí-los. Dados indicam que o tamanho das aplicações tem crescido exponencialmente nos últimos quarenta anos. Estima-se que o tamanho das aplicações cresça a uma taxa de dez vezes a cada cinco anos [Lyu07].

Software permeia o dia a dia do mundo atual. Existem aplicações de software em elevadores, nos aparelhos de TV, nos automóveis, em aparelhos celulares, sem contar as aplicações críticas já conhecidas como software de controle de usinas nucleares, software de apoio à navegação em aviões, software embutido em mísseis e radares etc. Estima-se que cerca de 9% do custo de fabricação de um carro médio é devido ao software, e esta estimativa deve aumentar para 15% nos próximos anos. O ônibus espacial da NASA voa com aproximadamente 500.000 linhas de código de software a bordo e mais cerca de 3,5 milhões de linhas de código nos sistemas de controle em terra [Lyu96].

Organizações comerciais de todos os portes e segmentos de mercado e nível de abrangência, bem como instituições científicas e até mesmo países dependem de software confiável para que possam atender às fortes demandas por produtos e serviços de alta qualidade.

Neste sentido, faz-se necessário desenvolver e aplicar métodos que permitam a avaliação da qualidade de software, uma vez que falhas podem levar a grandes perdas nos negócios ou até mesmo colocar em risco vidas humanas. A engenharia de software busca a qualidade aplicando métodos e medidas técnicas sólidas, conduzindo revisões técnicas formais e efetuando testes de software bem planejados.

A confiabilidade é uma das componentes da qualidade de software que tem sido extensivamente considerada na análise da qualidade do software, pois se um software não é confiável, pouco importa se outras características da qualidade são aceitáveis. Por outro lado, medir a confiabilidade de um software tem-se mostrado uma tarefa desafiadora [Jino07].

Um dos primeiros esforços para a modelagem da confiabilidade de software foi realizado por Hudson em 1967 utilizando um processo de Markov. Os primeiros modelos de confiabilidade foram desenvolvidos, independentemente, por Jelinski-Moranda e Shooman. O modelo de Jelinski-Moranda foi criado para o projeto Apollo, na McDonnell Douglas Astronautics Company. Ambos foram publicados em 1971.

Desde então, diversos modelos para avaliação do crescimento da confiabilidade de software têm sido propostos. Em geral os modelos propostos modelam a informação de falhas ocorridas durante um

período de teste do software para tentar prever seu comportamento de falhas futuro. Esta abordagem utiliza o número de falhas observadas por período de tempo ou o tempo entre falhas, neste caso, pode ser utilizado o tempo cronológico ou o tempo de execução, por isso são denominados de modelos baseados no domínio do tempo.

A maioria dos modelos existentes sobre crescimento da confiabilidade de software é baseada no domínio de tempo que utiliza o tempo decorrido entre falhas do software ou o número de falhas ocorridas durante um período de tempo determinado. Entretanto, um crescente número de pesquisadores tem argumentado que o tempo de execução não deve ser o único fator que afeta o comportamento de falhas de um software. A qualidade de predição dos modelos de confiabilidade de software pode ser melhorada se utilizarem outros fatores importantes que a afetam. A cobertura de teste tem sido utilizada como um indicador de completude e eficácia do teste de software e pode ser utilizada para melhorar a capacidade de predição dos modelos de confiabilidade de software tradicionais [An10].

Neste sentido novos modelos para estimar a confiabilidade de software, utilizando a informação da cobertura obtida pelos elementos requeridos pelos critérios de teste, têm sido propostos e, em geral, produzem melhores estimadores de confiabilidade do que aqueles baseados apenas na informação do tempo de teste. Os modelos “Modelo Tipo Binomial Baseado em Cobertura” e “Modelo de Categoria de Falhas Infinitas Baseado em Cobertura”, propostos por Crespo [Crespo97], são exemplos de modelos de confiabilidade de software que utilizam a informação de cobertura. Estudos têm mostrado que a capacidade preditiva destes modelos não é afetada pelas variações do perfil operacional do software, uma importante característica de modelos de confiabilidade de software.

Referências

[An10] An, J., Zhu, J., “Software Reliability Modeling with Integrated Test Coverage”, Fourth IEEE International Conference on Secure Software Integration and Reliability Improvement, Singapore, 2010.

[Crespo97] Crespo, A. N., “Modelos de Confiabilidade de Software Baseados em Cobertura de Critérios Estruturais de Teste.” Tese de Doutorado, DCA/FEE/UNICAMP - Campinas, SP, Dezembro 1997.

[Jino07] Jino, M., Maldonado, J. C., Delamaro, M. E., Introdução ao Teste de Software. Elsevier, Rio de Janeiro, 2007.

[Lyu96] Lyu, M., “Handobook of Software Reliability Engineering”, McGraw-Hill, 1996. [Lyu07] Lyu, M., “Software Reliability Engineering: A Roadmap”, Proceedings ICSE’2007, pp. 153-170, 2007.

Sobre o autor

Odair Jacinto da Silva é Diretor de Tecnologia da IN3 Software, Professor Universitário, Mestrando em Engenharia de Software pelo FEEC/Unicamp e Bacharel em Estatística pelo IMECC/Unicamp. Possui mais de 20 anos de experiência no desenvolvimento e implantação de projetos de software para empresas como: Petrobras, Merial Saúde Animal, Bosch, Motorola, Banco Alfa, CPFL etc. Possui artigos sobre melhoria de processo de software e teste de software publicados no Brasil e exterior.

PCSSCEG - Processo de contratação de software e serviços correlatos para entes governamentais.

Cláudio Silva da Cruz (cscruz@cscruz.org)
Tribunal de Contas da União (TCU)

Edméia Leonor Pereira de Andrade² (edmeia.andrade@embrapa.br)
Empresa Brasileira de Pesquisa Agropecuária (Embrapa)

Rejane Maria da Costa Figueiredo³ (RejaneCosta@unb.br)
Faculdade FGA, Campus Gama – Universidade de Brasília (UnB)

Abstract. *The Brazilian Federal Public Administration is a large IT services contractor and must adopt formal processes for IT outsourcing. MPS.BR-Acquisition Guide outsourcing process model do not comprehend legal requirements for the public sector. This research aimed to propose an outsourcing process model derived from MPS.BR-Acquisition Guide and aligned to legal requirements. The developed model has four phases: IT planning, contracting planning, provider selection and contractual management. The phases were divided onto activities and tasks and two examples of tasks are presented in this article.*

Resumo. *A Administração Pública Federal brasileira é grande contratadora de serviços de tecnologia da informação (TI) e deve adotar processos formais para esse tipo de contratação. O modelo de processo de contratação do MPS.BR-Guia de Aquisição:2009 não contempla as exigências do atual marco legal. O objetivo desse projeto foi propor um modelo de processo de contratações de serviços de TI derivado do Guia de Aquisição e alinhado ao marco legal. O modelo desenvolvido possui quatro fases: planejamento de TI, planejamento da contratação, seleção do fornecedor e gestão contratual, subdivididas em atividades e tarefas. O modelo alia aspectos técnicos e jurídicos em um único modelo de processo e tem potencial para reduzir as divergências entre as áreas envolvidas nas contratações (TI, jurídica, administrativa, orçamentária e controle).*

1. Introdução

O processo de contratação de software é considerado fundamental no modelo de ciclo de vida de software definido pela ISO/IEC (ABNT, 2009). No entanto, trata-se de disciplina ainda bastante desconhecida por profissionais de tecnologia da informação (TI) do setor público, provavelmente em decorrência da elevada demanda por conhecimento jurídico e administrativo que envolve (CRUZ, 2008, p. 201).

Para Cruz (2008, p. 201-202), a capacitação dos gestores de contratos de TI e a existência de um processo formal de contratação são precondições para que as contratações públicas se deem com boa aderência legal, entre outras.

A Administração Pública Federal brasileira gasta pelo menos R\$ 12,5 bilhões por ano em tecnologia da informação (BRASIL, 2010d), sendo parte significativa desse orçamento consumido com serviços relacionados a software, segundo detalhamento do orçamento para 2010 (BRASIL, 2010a). No controle do gasto desse orçamento, o Tribunal de Contas da União detectou elevada frequência de irregularidades e impropriedades, pelo que recomendou a adoção de processos de contratação mais maduros (BRASIL, 2006b). De fato, ainda há carência de um processo governamental de aquisição de software e serviços correlatos, pois mesmo o modelo de processo do MPS.BR-Guia de Aquisição (SOFTEX, 2009) ainda não contempla as

exigências do atual marco legal, como a Instrução Normativa SLTI/MP nº 04/2008 (BRASIL, 2008b).

O objetivo do presente projeto foi propor uma versão do MPS.BR-Guia de Aquisição para entes da Administração Pública Federal, compatibilizando o processo de aquisição de software e serviços correlatos com a legislação pública brasileira para contratações de serviços de tecnologia da informação (TI) e a respectiva jurisprudência. Este projeto concorreu e foi laureado com o Prêmio Dorgival Brandão Júnior da Qualidade e Produtividade em Software, ciclo 2009 (BRASIL, 2010b) e publicado na revista PBQP (CRUZ; ANDRADE; FIGUEIREDO, 2010).

Na Seção 2 apresenta-se a metodologia utilizada. Na Seção 3, os resultados obtidos, com a estrutura do modelo proposto. Na Seção 4, uma análise da aplicabilidade do modelo. Na Seção 5, um estudo de caso como validação do modelo. Finalizando, as Conclusões e perspectivas futuras.

2. Metodologia de execução

Para alcance do objetivo geral, foram definidos os seguintes objetivos específicos: (a) identificar o marco legal relevante para as aquisições de software e serviços correlatos; (b) identificar as necessidades de adequação do MPS.BR-Guia de Aquisição ao marco legal, segundo o COBIT ME3 (ITGI, 2007); e (c) ajustar o processo com base na versão do MPS.BR-Guia de Aquisição (SOFTEX, 2009) e no marco legal e definir artefatos para sua aplicação prática.

A identificação do marco legal tomou por base o levantamento de Cruz (2008, p. 90), que incluiu a análise de 64 documentos legais e 57 documentos jurisprudenciais. A esses ainda foram acrescentadas normas recentes, como a Instrução Normativa SLTI/MP nº 02/2008 (BRASIL, 2008a) e a nº 04/2008 (BRASIL, 2008b), e jurisprudência recente, como os acórdãos 1.603 e 2.471/2008-TCU-Plenário.

A identificação das necessidades de adequação do Guia de Aquisição e o ajuste do processo foram realizados com base em comparações entre os elementos extraídos do marco legal, especialmente o Quadro Referencial Normativo - QRN (CRUZ, 2008, p. 92) e a Instrução Normativa SLTI/MP nº 04/2008 (BRASIL, 2008b), e o Guia de Aquisição do MPS.BR (SOFTEX, 2009), à semelhança das pesquisas concluídas pelos pesquisadores (ANDRADE; SCALET, 2007; CRUZ, 2008; GILLIS; ANDRADE, 2009; SCALET; ANDRADE, 2009; SOUSA et al., 2005), e também com base no conhecimento advindo de atividades de ensino (CRUZ, 2009a, 2009b, 2009c, 2009d) e em experiências práticas dos autores deste trabalho.

Para a definição do modelo de processo foi empregada a estrutura de processo definida pela norma ABNT NBR ISO/IEC 12207:2009 (ABNT, 2009), que é padrão internacional para processos do ciclo de vida de software, e pelo Guia de Aquisição do MPS.BR (SOFTEX, 2009). Para cada fase foram descritos o propósito, as atividades e as respectivas tarefas.

3. Resultados obtidos

Mais de cem documentos do marco legal brasileiro foram identificados e analisados. Ao comparar o Guia de Aquisição do MPS.BR com o marco legal levantado, as principais lacunas identificadas foram: (a) ausência

de clara vinculação entre a aquisição e as diretrizes estabelecidas no planejamento estratégico de TI; (b) ausência de identificação clara de papéis relevantes e seus atores, tais como gestor do contrato, requisitante

ATIVIDADE	TAREFA
FASE 1. PLANEJAMENTO DE TI	
1.1 Identificar as necessidades de negócio 1.2 Definir diretrizes de TI alinhadas ao negócio 1.3 Definir macro-ações de TI 1.4 Identificar as principais contratações de TI 1.5 Reservar os recursos (RH, orçamento, prioridades etc.)	
FASE 2. PLANEJAMENTO DA CONTRATAÇÃO	
2.1 Analisar a viabilidade da contratação	2.1.1 Analisar e justificar a necessidade 2.1.2 Fundamentar a motivação 2.1.3 Levantar e especificar requisitos 2.1.4 Analisar soluções de TI 2.1.5 Analisar preliminarmente riscos, impactos e a viabilidade econômica 2.1.6 Selecionar a solução de TI e justificar
2.2 Elaborar o plano de sustentação	2.2.1 Elaborar proced. de segurança da informação 2.2.2 Elaborar proced. de gestão de recursos 2.2.3 Elaborar proced. de transfer. de conhecimento 2.2.4 Elaborar proced. de transição contratual 2.2.5 Elaborar proced. de continuidade de negócio
2.3 Elaborar a estratégia de contratação	2.3.1 Indicar o tipo de contratação 2.3.2 Definir termos contratuais 2.3.3 Definir a estratégia de independência 2.3.4 Indicar o gestor do contrato 2.3.5 Definir as responsabilidades das partes 2.3.6 Definir orçam. detalhado e fonte de recursos 2.3.7 Definir os critérios técnicos de seleção
2.4 Analisar e tratar riscos	2.4.1 Identificar riscos da licitação 2.4.2 Identificar riscos dos serviços 2.4.3 Analisar probabilidade e impacto dos riscos 2.4.4 Planejar ações de mitigação de riscos 2.4.5 Planejar ações em contingências
2.5 Analisar e demonstrar a economicidade da contratação 2.6 Elaborar e encaminhar a minuta do termo de referência (ou do projeto básico)	
FASE 3. SELEÇÃO DO FORNECEDOR	
3.1 Formalizar e aprovar o termo de referência (ou o projeto básico)	
3.2 Selecionar fornecedor (por contratação direta)	3.2.1 Justificar a contratação direta 3.2.2 Negociar o contrato 3.2.3 Obter o exame e a aprovação jurídicos 3.2.4 Obter a aprovação da contratação
3.3 Selecionar fornecedor (por licitação)	3.3.1 Elaborar o edital 3.3.2 Obter o exame e a aprovação jurídicos 3.3.3 Realizar consulta ou audiência pública 3.3.4 Realizar ajustes no edital 3.3.5 Publicar o edital 3.3.6 Responder a questionamentos/recursos 3.3.7 Receber e julgar as propostas 3.3.8 Adjudicar o objeto 3.3.9 Homologar a licitação
3.4 Ajustar a minuta do contrato 3.5 Assinar o contrato	

da contratação, autoridade competente, fiscal do contrato etc.; (c) ausência de previsão da fase licitatória; (d) menor rigor com a formalização e a motivação do que o exigido pela legislação para o setor público; (e) ênfase na negociação com fornecedores, o que não é possível no setor público; e (f) ausência de rigor formal na fase de gestão do contrato (p.ex. uso da ordem de serviço). Essas e outras lacunas foram tratadas no modelo proposto.

O Quadro 1 apresenta a estrutura do processo proposto, que consiste de 4 fases, subdivididas em atividades e tarefas, que podem ser adotadas para contratação de qualquer tipo de serviço de TI no setor público.

Quadro 1. Estrutura geral do modelo de processo de contratação proposto

FASE 4. GESTÃO DO CONTRATO	
4.1 Iniciar o contrato	4.1.1 Elaborar o plano de inserção 4.1.2 Repassar conhecimento 4.1.3 Alocar infraestrutura 4.1.4 Designar as pessoas para papéis 4.1.5 Realizar a reunião de iniciação
4.2 Encaminhar demandas	4.2.1 Receber demandas de negócio 4.2.2 Elaborar a ordem de serviço 4.2.3 Assinar a ordem de serviço
4.3 Realizar o monitoramento técnico	4.3.1 Acompanhar a execução da ordem de serviço 4.3.2 Gerenciar riscos 4.3.3 Determinar ação corretiva 4.3.4 Providenciar alterações da ordem de serviço
4.4 Executar a atestação técnica	4.4.1 Receber os serviços concluídos 4.4.2 Avaliar os serviços 4.4.3 Rejeitar serviços inadequados 4.4.4 Aceitar serviços adequados
4.5 Realizar o monitoramento administrativo	4.5.1 Acompanhar os serviços em execução 4.5.2 Avaliar cumprimento do contrato 4.5.3 Aplicar sanções 4.5.4 Corrigir desatendimento contratual 4.5.5 Calcular valores dos serviços e impostos 4.5.6 Liquidar a despesa 4.5.7 Autorizar emissão de fatura/nota fiscal 4.5.8 Verificar regularid.fiscal/comercial/trabalh./previd. 4.5.9 Efetuar pagamento dos serviços
4.6 Tratar as demandas por alterações contratuais	4.6.1 Tratar alteração de quantitativos 4.6.2 Tratar alteração de prazos de execução 4.6.3 Tratar alteração de cláusulas contratuais 4.6.4 Tratar alteração técnica 4.6.5 Tratar readequação de preços 4.6.5.1 Tratar reequilíbrio econômico-financeiro 4.6.5.2 Tratar reajuste de preço 4.6.5.3 Tratar repactuação 4.6.6 Tratar prorrogação do contrato 4.6.7 Tratar suspensão do contrato 4.6.8 Tratar rescisão do contrato
4.7 Realizar o encerramento contratual e a transição	4.7.1 Realizar proced. de transferência de conhecimento 4.7.2 Realizar proced. de transferência de recursos 4.7.3 Realizar proced. de segurança da informação 4.7.4 Verificar a quitação de encargos trabalh./previd.

Quadro 1. Estrutura geral do modelo de processo de contratação proposto

3.1. Exemplo de descrição de tarefa

Sucintamente são apresentados dois exemplos de descrição de tarefa. Elegeram-se as tarefas “analisar e justificar a necessidade” e “levantar e especificar os requisitos” da atividade “analisar a viabilidade da contratação” da fase de “planejamento da contratação” para facilitar ao leitor a comparação com a versão original do Guia de Aquisição.

Para cada tarefa foi definido um identificador (Id) que representa as três letras iniciais da fase do processo “planejamento da contratação” (Pla), seguida da letra t, indicando uma tarefa, e a sua numeração, conforme apresentado a seguir.

Quadro 2. Exemplos de detalhamento de tarefa

Id.	Tarefa
Pla-t1	<p>Analisar e justificar a necessidade</p> <p>Descrição: Estabelecer as necessidades a serem atendidas por meio da contratação, desenvolvimento ou melhoria de um produto ou serviço de software. Durante essa tarefa são analisadas o efetivo escopo das necessidades e resultados que a organização pretende atingir com o projeto de contratação. Esta tarefa é fundamental, pois indica a primeira tomada de decisão quanto ao prosseguimento do projeto e que resultados são esperados pela organização após a efetivação da contratação.</p> <p>Produtos requeridos: Pti-p1 – Identificação das necessidades de negócio Pti-p2 – Ações de TI priorizadas</p> <p>Responsável: Área de Negócio Participantes: Área de TI</p> <p>Produtos gerados: Pla-p1 – Documento da análise e justificativa da necessidade da contratação</p> <p>Legislação relacionada: Decreto nº 2.271/1997, art. 2º (Brasil, 1997); Decreto nº 3.555/2000, art. 8º, III, “b” (Brasil, 2000); Instrução Normativa SLTI/MP nº 04/2008, art. 10, I e II (Brasil, 2008b).</p>
Pla-t2	<p>Levantar e especificar os requisitos</p> <p>Descrição: Identificar e especificar os requisitos do cliente para a contratação do produto ou serviço de software. Se necessário, as organizações poderão solicitar informações de fornecedores ou realizar pesquisas para identificar as melhores práticas de outras organizações, que adquiriram produtos e serviços semelhantes, visando identificar os requisitos a partir de soluções disponíveis no mercado. Durante esta tarefa devem ser especificados os seguintes requisitos a serem considerados no projeto de contratação:</p> <ul style="list-style-type: none"> • dos interessados (stakeholders): as necessidades devem ser transformadas em requisitos mais específicos que contemplem os diversos tipos de interessados, tais como, usuários, planejadores, gestores, desenvolvedores e beneficiários do produto de software; • do sistema: requisitos envolvendo processos, hardware, software, integrações, ambiente e pessoas que irão compor a solução que atenderá as necessidades estabelecidas; • do software: requisitos do(s) produto(s) de software que irá(ão) compor o(s) sistema(s) a ser(em) implementado(s). Devem ser especificados os requisitos funcionais; • de qualidade: alguns requisitos de qualidade são: usabilidade - estilo ou princípios de diálogo que são aplicáveis; tipo de documentação a ser entregue (on-line, manuais de usuário); portabilidade - regras de portabilidade que deverão ser adotadas (tanto para a parte de servidores quanto para acesso via estações de trabalho); interoperabilidade - integração das aplicações novas com os bancos de dados e aplicações legadas; manutenibilidade: tipos e características dos artefatos gerados, de modo a permitir a manutenção por parte do contratado e facilitar eventuais repasses de conhecimento; • de projeto: ciclo de vida a ser adotado, técnicas, metodologias, forma de gestão e de documentação do projeto; • de manutenção: requisitos de manutenção do software após a sua entrega; • de treinamento: características esperadas e quantidade de treinamento relacionado ao produto de software a ser entregue; e

<p>• de implantação: descrição dos procedimentos necessários para a implantação do software no ambiente de operação, como, por exemplo, a carga do banco de dados, a implementação numa configuração distribuída, entre outros.</p> <p>Além destes requisitos, podem ser considerados outros requisitos e restrições que afetam diretamente o projeto de aquisição como, por exemplo, restrições legais, financeiras, de prazo do projeto e de número de usuários do sistema em operação. O adquirente poderá definir e analisar os requisitos com sua própria equipe ou contratar um fornecedor para executar estas atividades. Neste caso, o adquirente deverá manter a responsabilidade pela aprovação do resultado da especificação dos requisitos.</p> <p>NOTA: Esses requisitos variam de acordo com o tipo da contratação. No caso de contratação de serviços de desenvolvimento e manutenção de escopo aberto, sabe-se que será adquirida uma certa quantidade de Pontos de Função, baseado na lista de necessidades identificadas no Plano Diretor de Tecnologia da Informação (PDTI), na capacidade de gerenciamento e na disponibilidade de recursos financeiros da organização, mas a especificação dos requisitos de cada software será realizada posteriormente em cada ordem de serviço a ser aberta durante a execução do contrato, ou pela equipe interna.</p> <p>Produtos requeridos: Pla-p1 – Documento da análise e justificativa da necessidade da contratação Pla-p2 – Documento da análise de mercado e soluções de TI NOTA: Este documento é, em geral, elaborado como parte do processo de análise de viabilidade de iniciar o projeto de aquisição, avaliando-se o problema da organização e as alternativas possíveis de solução que o mercado oferece.</p> <p>Responsável: Área de TI Participantes: Área de Negócio</p> <p>Produtos gerados: Pla-p1 – Documento da análise e justificativa da necessidade da contratação revisado. NOTA: Esta tarefa pode provocar a revisão do escopo das necessidades a serem atendidas em função das características dos requisitos a serem contemplados para atender as necessidades, que podem causar impactos em custo e prazo. Pla-p3 – Documento de especificação de requisitos</p> <p>Legislação relacionada: Lei nº 8.666/1993, arts. 6º, IX, e 12; Decreto nº 3.555/2000, art. 8º, I, II e III, "a"; Instrução Normativa SLTI/MP nº 04/2008, art. 10, III.</p>
--

Quadro 2. Exemplos de detalhamento de tarefa

4. Aplicabilidade dos Resultados

A Instrução Normativa SLTI/MP nº 04/2008 (BRASIL, 2008b) e os acórdãos 1.603 (BRASIL, 2008c) e 2.471/2008-TCU-Plenário (BRASIL, 2008d) estabelecem a obrigatoriedade de os entes públicos adotarem processos formais de contratação de serviços de TI, em particular em processos de software. O modelo de processo apresentado pode auxiliar os governantes e os gestores públicos de TI no atendimento dessa demanda legal, visto que o processo em si já é aderente aos requisitos da legislação e pode favorecer: (a) à ampliação do valor da TI aos processos de negócio dos entes públicos; (b) à ampliação do estímulo à adoção da melhoria da qualidade do software brasileiro, em linha com o PBQP-SW/MCT (BRASIL, 2010c); e (c) à ampliação da participação de pequenas e médias empresas na comercialização de software e serviços correlatos para Governo, em linha com a Lei Complementar 123/2006 (BRASIL, 2006a) e também com o PBQP-SW/MCT, uma vez que foram incluídos mecanismos de divisão de objeto, de exclusão de exigência desnecessárias e de aumento do uso do pregão eletrônico.

O modelo proposto foi concebido com base na legislação da Administração Pública Federal. Porém, como a legislação de licitações e contratos de estados e municípios também está vinculada à Lei nº 8.666/1993 (BRASIL, 1993), considera-se que o modelo é perfeitamente aplicável a todas as esferas,

sem distinção, com pouca necessidade de adaptação. Também se observa que este modelo pode ser aplicado não somente a serviços de software, mas também a qualquer serviço de TI.

A inovação do presente modelo reside na integração de aspectos técnicos da Engenharia de Software com os aspectos legais em um só modelo de processo. Desse modo, abre-se à oportunidade de interação mais produtiva entre os gestores públicos de TI e as áreas jurídica, administrativa e de controle (interno ou externo), com base em contextos mais compartilhados e uma linguagem comum. Com isso, tornam-se possíveis: (a) a redução do retrabalho, pois os documentos de contratação podem ser produzidos já aderentes à legislação da primeira vez; (b) a redução do tempo de contratação, mesmo utilizando-se dos rigorosos procedimentos licitatórios; e (c) o aumento da eficiência, seja pela redução de prazos de contratação ou pela redução dos valores pagos pela Administração em suas contratações, em virtude do aumento da concorrência.

5. Estudo de Caso - EMBRAPA

Atualmente, este processo encontra-se em validação junto à Embrapa, por meio de capacitação de até 270 empregados lotados na Sede e em 45 Unidades Descentralizadas de Pesquisa, por meio de videoconferência. Os participantes são originários das áreas de TI, administrativa, de licitações, consultoria jurídica e controle interno.

Tal capacitação, estimada em 72 horas, envolve as quatro fases do processo, com exercícios práticos na fase de planejamento da contratação. Espera-se que os participantes possam avaliar as atividades e tarefas previstas em cada fase, para fins de validação em uma organização pública, como a Embrapa.

6. Conclusão e Perspectivas Futuras

No ciclo 2009, foi possível construir um modelo de processo de contratação de software e serviços correlatos (aplicável também a outros serviços de TI) com base no marco legal do setor público e no Guia de Aquisição do MPS.BR. O modelo alia aspectos técnicos e jurídicos em um único modelo de processo. Por essa razão, o modelo tem potencial para reduzir as divergências entre as áreas envolvidas nas contratações (TI, jurídica, administrativa, orçamentária e controle), acelerando o processo sem perda de segurança jurídica e reduzindo preços em razão do aumento da competição.

Espera-se que o modelo seja discutido e aperfeiçoado no âmbito da Administração Pública e contribua: (a) com a Softex e com a SLTI/MP na produção e aperfeiçoamento de modelos e normas sobre o assunto; (b) com os entes públicos na capacitação de gestores de contratos de TI e na facilitação do atendimento da Instrução Normativa SLTI/MP nº 04/2008 (BRASIL, 2008b), permitindo aumento da eficácia, eficiência, efetividade, economicidade, legalidade e legitimidade dos projetos de TI e aumentando o valor agregado pela TI ao negócio institucional, em termos de aumento de benefícios produzidos, de redução de riscos legais e contratuais e de aumento da integração entre as áreas de TI, de licitação, jurídica e os gestores de negócio; e (c) com o MCT na política de melhoria da qualidade do software brasileiro (PBQP-SW/MCT) e na ampliação da participação de pequenas e médias empresas nas compras públicas (Lei Complementar nº 123/2006) (BRASIL, 2006a).

Em continuação da pesquisa, será possível: definir e estabilizar modelos de documento que implementem o presente modelo de processo; realizar testes de campo, com medição da melhoria alcançada; validar o modelo de processo e de documentos em diferentes domínios da administração pública.

Referências bibliográficas

ANDRADE, Edméia Leonor Pereira de; SCALET, Danilo. **RELATÓRIO TÉCNICO: Personalização do Processo de Aquisição do MPS.BR para o Ministério da Agricultura, Pecuária e Abastecimento - MAPA**. Campinas: Softex, 2007. Disponível em: <http://www.softex.br/portal/softexweb/uploadDocuments/_mpsbr/RELATORIO_TECNICO_Consultor_edmeia_220107.pdf>. Acesso em: 30 set. 2010.

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT. **ABNT NBR ISO/IEC 12207:2009 - Engenharia de sistemas e software - Processos de ciclo de vida de software**. Disponível em: <<http://www.abntcatalogo.com.br/norma.aspx?ID=38643>>. Acesso em: 30 set. 2010.

BRASIL. **Lei nº 8.666, de 21 de junho de 1993**. Regulamenta o art. 37, inciso XXI, da Constituição Federal, institui normas para licitações e contratos da Administração Pública e dá outras providências. 1993. Disponível em: <http://www.planalto.gov.br/ccivil_03/Leis/L8666cons.htm>. Acesso em: 30 set. 2010.

_____. **Lei Complementar 123, de 14 de dezembro de 2006**. Institui o Estatuto Nacional da Microempresa e da Empresa de Pequeno Porte; altera dispositivos das Leis nos 8.212 e 8.213, ambas de 24 de julho de 1991, da Consolidação das Leis do Trabalho – CLT, aprovada pelo Decreto-Lei no 5.452, de 1º de maio de 1943, da Lei no 10.189, de 14 de fevereiro de 2001, da Lei Complementar no 63, de 11 de janeiro de 1990; e revoga as Leis nos 9.317, de 5 de dezembro de 1996, e 9.841, de 5 de outubro de 2006a. Disponível em: <http://www.planalto.gov.br/ccivil_03/Leis/LCP/Lcp123.htm>. Acesso em: 30 set. 2010.

_____. Tribunal de Contas da União. **Acórdão 786/2006-TCU-Plenário**. 2006b. Disponível em: <[http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=\(acordao+adj+786/2006+adj+plenario\)\[idtd\]\[b001\]](http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=(acordao+adj+786/2006+adj+plenario)[idtd][b001])>. Acesso em: 30 set. 2010.

_____. Ministério do Planejamento, Orçamento e Gestão. Secretaria de Logística e Tecnologia da Informação. **Instrução Normativa SLTI nº 2, de 30 de abril de 2008**. Dispõe sobre regras e diretrizes para a contratação de serviços, continuados ou não. 2008a. Disponível em: <<http://www.governoeletronico.gov.br/biblioteca/arquivos/instrucao-normativa-no-02>>. Acesso em: 30 set. 2010.

_____. _____. **Instrução Normativa SLTI nº 4, de 19 de maio de 2008**. Dispõe sobre o processo de contratação de serviços de Tecnologia da Informação pela Administração Pública Federal direta, autárquica e fundacional. 2008b. Disponível em: <<http://www.governoeletronico.gov.br/biblioteca/arquivos/instrucao-normativa-no-04-2>>. Acesso em: 30 set. 2010.

_____. Tribunal de Contas da União. **Acórdão 1.603/2008-TCU-Plenário**. 2008c. Disponível em: <[http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=\(acordao+adj+1603/](http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=(acordao+adj+1603/)

2008+adj+plenario)[idtd][b001]>. Acesso em: 30 set. 2010.

_____. _____. **Acórdão 2.471/2008-TCU-Plenário**. 2008d. Disponível em: <[http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=\(acordao+adj+2471/2008+adj+plenario\)\[idtd\]\[b001\]](http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=(acordao+adj+2471/2008+adj+plenario)[idtd][b001])>. Acesso em: 30 set. 2010.

_____. Dados de proposta orçamentária para 2010 com detalhamento para despesas de TI. **Sistema SIDOR**. Brasília: SOF, 2010a.

_____. Ministério da Ciência e Tecnologia. Programa Brasileiro da Qualidade e Produtividade em Software. **Ciclo 2009 - Prêmio Dorgival Brandão Júnior da Qualidade e Produtividade em Software**. Brasília: MCT, 2010b. Disponível em: <<http://www.mct.gov.br/index.php/content/view/319747.html>>. Acesso em: 30 set. 2010.

_____. Ministério da Ciência e Tecnologia. **Programa Brasileiro da Qualidade e Produtividade em Software. Estratégias e ações**. Brasília: MCT, 2010c. Disponível em: <<http://www.mct.gov.br/index.php/content/view/4794.html>>. Acesso em: 30 set. 2010.

_____. Tribunal de Contas da União. **Acórdão 2.308/2010-TCU-Plenário**. Relatório de levantamento. Avaliação da governança de tecnologia da informação na administração pública federal. Constatação de precariedades e oportunidades de melhoria. Determinações, recomendações e comunicações. 2010d. Disponível em: <[http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=\(acordao+adj+2308/2010+adj+plenario\)\[idtd\]\[b001\]](http://contas.tcu.gov.br/portaltextual/MostraDocumento?lnk=(acordao+adj+2308/2010+adj+plenario)[idtd][b001])>. Acesso em: 30 set. 2010.

CRUZ, Cláudio Silva da Cruz; ANDRADE, Edméia Leonor Pereira de Andrade; FIGUEIREDO, Rejane Maria da Costa. Processo de contratação de software e serviços correlatos para entes governamentais. **Revista Programa Brasileiro da Qualidade e Produtividade em Software**. 2010. Projetos Ciclos 2008 e 2009, p. 103-110. ISSN 2178-0277. Disponível em: <http://www.mct.gov.br/upd_blob/0212/212192.pdf>. Acesso em: 30 set. 2010.

CRUZ, Cláudio Silva da. **Governança de TI e conformidade legal no setor público: um quadro referencial normativo para a contratação de serviços de TI**. 2008. 252f. Dissertação (Mestrado em Gestão do Conhecimento e da Tecnologia da Informação). Universidade Católica de Brasília, Brasília, 2008. Disponível em: <http://www.bdt.d.ucb.br/tede/tde_arquivos/3/TDE-2008-11-25T123713Z-687/Publico/Texto Completo Cruz - 2008.pdf>. Acesso em: 30 set. 2010.

_____. **Contratação de serviços de TI**. Curso integrante do MBA em Governança de TI. Brasília: Fundação Universa/Universidade Católica de Brasília, 2009a.

_____. **Elaboração do Plano Diretor de Tecnologia da Informação - PDTI**. Curso integrante do programa de desenvolvimento de gestores de TI da Secretaria de Logística e Tecnologia da Informação/MP. Brasília: ENAP, 2009b.

_____. **Gestão de contratos de serviços de TI - GCTI**. Curso integrante do programa de desenvolvimento de gestores de TI da Secretaria de Logística e Tecnologia da Informação/MP. Brasília: ENAP, 2009c.

_____. **Planejamento de contratação de serviços de TI - PCTI**. Curso integrante do programa de desenvolvimento de gestores de TI da Secretaria de Logística e Tecnologia da Informação/MP. Brasília: ENAP, 2009d.

GILLIS, Eric Robert; ANDRADE, Edméia Leonor Pereira de. Comparação entre a Instrução Normativa SLTI/MP nº 04 e o Guia de Aquisição do MPS.BR. IN: **Workshop Anual do MPS**, 5. WAMPS 2009. Campinas-SP, 19 a 22 de outubro de 2009, p. 92-101. Disponível em: <<http://www.softex.br/portal/softexweb/uploadDocuments/Softex%20WAMPS%202009%20Web.pdf>>. Acesso em: 30 set. 2010. INFORMATION TECHNOLOGY GOVERNANCE INSTITUTE - ITGI. **COBIT - Control Objectives for Information and related Technology**. 4.1. ed. Rolling Meadows: ITGI, 2007. Disponível em: <<http://www.isaca.org/Knowledge-Center/cobit/Pages/Downloads.aspx>>. Acesso em: 30 set. 2010.

SCALET, Danilo; ANDRADE, Edméia Leonor Pereira de. O Guia de Aquisição do MPS.BR e o Pregão Eletrônico. IN: **Workshop Anual do MPS**, 5. WAMPS 2009. Campinas-SP, 19 a 22 de outubro de 2009, p. 82-91. Disponível em: <<http://www.softex.br/portal/softexweb/uploadDocuments/Softex%20WAMPS%202009%20Web.pdf>>. Acesso em: 30 set. 2010.

SOFTEX, Sociedade para Promoção da Excelência do Software Brasileiro. **MPS.BR - Melhoria de Processo do Software Brasileiro – Guia de Aquisição**. São Paulo: SOFTEX, Junho 2009. Disponível em: <http://www.softex.br/mpsbr/_guias/guias/MPS.BR_Guia_de_Aquisicao_2009.pdf>. Acesso em: 30 set. 2010.

SOUSA, Marcelo Paiva de; SÂNDI, Vanessa Teixeira de Oliveira; OLIVEIRA, Káthia Marçal de; FIGUEIREDO, Rejane Maria da Costa. Processo de aquisição de produtos e serviços de software para uma instituição bancária. In: **SIMPROS - Simpósio Internacional de Melhoria de Processos de Software**, 7., 2005. São Paulo, SP – Brasil, 21-23 nov. 2005. Disponível em: <http://www.simpros.com.br/upload/A08_2_artigo14644.pdf>. Acesso em: 30 set. 2010.

Proposta e Aplicação de um Método de Seleção de Fornecedores Baseado no Guia de Aquisição do MPS.BR

Ana Marcia Debiasi Duarte¹, Jefferson Oestreich de Mello², Denio Duarte³

¹Universidade Comunitária da Região de Chapecó (Unochapecó) - ACEA

²H2J Sistemas e Consultoria

³Universidade Federal da Fronteira Sul (UFFS) - Campus Chapecó

anamarcia@unochapeco.edu.br, jefferson@h2jsolucoes.com.br, duarte@uffs.edu.br

Resumo. *Este trabalho apresenta a proposta de um método para avaliação da capacidade de fornecedores baseado no Guia de Aquisição do MPS.BR e o resultado da aplicação dessa proposta. O método é composto por critérios objetivos para auxiliar no processo de avaliação de fornecedores em uma indústria moveleira no Oeste de Santa Catarina, proporcionando assim maior clareza e objetividade nas atividades realizadas.*

1. Introdução

O processo de aquisição de um *software* não é uma atividade trivial, principalmente para sistemas de gestão empresarial pois envolve um grande investimento de dinheiro e tempo. Geralmente, as empresas encontram dificuldades na aquisição de software [Conry-Murray 2009]. Este fato não ocorre necessariamente por falta de ofertas e alternativas dadas pelas desenvolvedoras de software, mas em grande parte pela complexidade e dificuldade de escolha e comparação entre os produtos disponíveis [Guerra and Alves 2004].

Existem vários problemas apontados como sendo fatores que levam à aquisições desastrosas nas organizações. Uma das causas fundamentais de discordância entre os adquirentes e fornecedores tem origem na ambigüidade do contrato [Jones 2001]. Em pesquisa realizada por Conry-Murray [Conry-Murray 2009], dois itens relativos a satisfação do cliente aparecem em destaque: o desejo dos clientes de que os fornecedores entendam melhor as suas necessidades e as promessas de que o produto contém mais requisitos do que de fato contempla.

O processo de aquisição tem sucesso quando o adquirente e o fornecedor ficam satisfeitos, e para isso, o fornecedor deve entender de forma clara os requisitos do adquirente e o *software* fornecido deve atender a estes requisitos, respeitando prazo e custo concordados. A aquisição é, assim, um processo complexo e que se não for devidamente gerenciado, gera prejuízos para o adquirente e para o fornecedor.

Neste contexto, este trabalho descreve a experiência da definição e da aplicação de um método para a aquisição de *software*, contemplando os dois primeiros subprocessos do guia de aquisição do

MPS.BR [SOFTEX-a 2009]: (i) de preparação da aquisição, para determinar a estratégia de aquisição e a classificação e pontuação dos requisitos, e (ii) de seleção de fornecedor, para avaliar e selecionar os fornecedores pelo grau de atendimento dos requisitos. Esses dois subprocessos são base para a proposição de um método para avaliação da capacidade de fornecedores baseado em um padrão de critérios pré-determinados.

O restante deste trabalho está organizado da seguinte forma: a próxima seção apresenta o cenário da aplicação do método, descrevendo a empresa adquirente e suas necessidades. A Seção 3 descreve o método proposto e a Seção 4 apresenta os resultados da aplicação do método. Finalmente, a Seção 5 conclui este trabalho.

2. Cenário

O método proposto neste trabalho foi aplicado em uma indústria de móveis situada na região oeste de Santa Catarina. A indústria conta com aproximadamente 500 funcionários e um faturamento anual aproximado de R\$ 120 milhões. A produção atinge todo o mercado brasileiro e é exportado para mais 10 países. Fundada em 1992, a indústria possui como apoio para sua operação, um *software* que vem sendo desenvolvido e mantido desde o início da operação da indústria. Com o passar dos anos o *software* passou a apresentar limitações, tais como, dificuldade de se ajustar ao crescimento da indústria e abertura de novos mercados, demora nos ajustes de ordem legal, mas principalmente limitações impostas pela arquitetura original e que foi sendo modificada e adaptada ao longo dos anos.

Assim, através de uma consultoria inicial, a indústria criou uma comissão de aquisição para identificar qual seria a melhor estratégia para adquirir um produto que fosse mais aderente as necessidades atuais. Essa comissão, composta por dois consultores e diretores superintendente, comercial, de produção, administrativo, financeiro, tributário e de suprimentos, iniciou um processo para identificar um método de aquisição que resultasse na compra de um *software* que atendesse às necessidades da indústria. Cada membro da comissão possuía papéis definidos, sendo que a proposta do método de aquisição foi feita pelos consultores e aprovada pela comissão. Todas as decisões, ao longo do processo de aquisição, foram tomadas no comitê.

A primeira decisão tomada foi medir o quanto o modelo de fornecimento de *software* e serviços existente atendia de fato as necessidades da indústria. Após alguns estudos (que não são descritos neste trabalho) concluiu-se que o fornecimento existente não atendia o que a empresa buscava. Desta forma, foi tomada a decisão de iniciar um processo de aquisição de um novo produto, onde foi aplicado o método proposto, aprovado pelo comitê e relatado neste trabalho.

3. Método Utilizado

Neste trabalho optou-se pelo uso do Guia de Aquisição do MPS.BR como instrumento para orientar o processo de aquisição. Além de possuir um guia específico, é de fácil entendimento e fornece modelos de artefatos. O método proposto contempla as atividades de “Preparação da aquisição” e “Seleção do fornecedor”. Já as atividades de “Monitoração do Contrato” e “Aceitação pelo cliente”

serão tratadas futuramente no método. Este método foi proposto, inicialmente, para atender aquisições baseadas em *software* MOTS ou COTS [IEEE 1998].

3.1 Preparação da Aquisição

Na atividade de preparação da aquisição do guia de aquisição são identificadas quatro tarefas a serem realizadas. A seguir é descrita a equivalência entre as atividades do guia e como o método proposto trata cada uma delas.

Estabelecer a Necessidade: nesta tarefa o método propõe uma revisão das necessidades já levantadas pelo comitê e a formalização dos objetivos que devem ser atingidos com o processo de aquisição. O produto gerado nesta tarefa é o primeiro item do plano de aquisição “Objetivos da Aquisição”.

Definir Requisitos: nesta tarefa, o método propõe identificar e descrever os requisitos funcionais (RF) e não funcionais (RNF). Cada requisito deve ter os seguintes atributos: (i) Tipo - identifica se o requisito é funcional ou não-funcional, (ii) Código - identificação única do requisito (formato REQ-*nnn*), (iii) Título - frase curta que identifica o requisito, (iv) Descrição - descreve o requisito de acordo com a visão do usuário, (v) Setor, e (vi) Departamento - descreve onde o requisito foi identificado. Além dos atributos acima, os RF recebem um tratamento com base em critérios objetivos quantitativos que permitem a comparação entre os fornecedores que participam do processo de aquisição. Para a definição dos critérios foram seguidos alguns aspectos sugeridos em [Sommerville 2007, Pressman 2002], que auxiliam durante o processo de aquisição. Após discussão, e levando em conta necessidades estratégicas apontadas pelo comitê, optou-se por definir risco e prioridade como critérios para avaliar os RF.

O risco permite avaliar o requisito sob a ótica de negócio. Os valores foram definidos e um peso foi associado a cada valor: alto (10), médio (5) ou baixo (1). Como exemplo, pode-se citar o RF “bloqueio automático de clientes inadimplentes”. O risco foi considerado alto pelo responsável da área financeira, pois a venda para clientes inadimplentes apresenta um alto risco de perdas financeiras para a indústria. Neste caso, o peso desse requisito recebe o valor 10.

A prioridade é considerada como a urgência de ter o requisito funcionando no *software* adquirido. Os possíveis valores para a classificação da prioridade e os pesos foram definidos como: indispensável (10), desejável (5) ou indefinido (1). Este critério também permite a análise para determinar uma estratégia de aquisição parcial do produto, contemplando RF de maior prioridade.

Quanto aos RNF, sabe-se que a especificação quantitativa não é trivial [Sommerville 2007] e que o custo de verificar objetivamente critérios para avaliá-los é muito alto. Neste sentido, optou-se por trabalhar os RNF com avaliação objetiva, no entanto não serão quantificados em relação ao conjunto total dos requisitos. Assim, cada RNF possui critério específico para avaliação. Foram definidas categorias para organizar e orientar a sua identificação: interessados, sistema, projeto, manutenção, treinamento, implantação, financeiro, legal e tecnológico.

Além dos atributos básicos dos RF, os RNF possuem os seguintes novos atributos: categoria, critério avaliado e valores possíveis. Por exemplo, um determinado RNF pode ser representado por *?código*:

REQ-020; *tipo*: não-funcional; *título*: avaliação modelo qualidade; *descrição*: fornecedor possui uma avaliação de modelo de maturidade de processo; *setor*: comitê de aquisição; *departamento*: direção; *categoria*: Interessados; *critério avaliado*: Possuir avaliação MPS.BR ou CMMI (qualquer nível); *valores possíveis*: Sim e Não?

A tarefa de definição de requisitos prevê o levantamento de todos os requisitos através de entrevistas com os usuários e análise do *software* em uso. Já os RNF são identificados através de entrevistas, análises do ambiente, ou proposições que terão sua pertinência discutida no comitê.

A revisão dos requisitos, que é apresentada como uma tarefa independente no guia de aquisição do MPS.BR, neste modelo é tratada como uma etapa na tarefa de definição dos requisitos. O produto gerado é um documento anexo ao plano de aquisição que contém todos os RF e RNF, respeitando os seus formatos definidos.

Definir uma estratégia de seleção dos fornecedores: esta tarefa foi definida no método proposto, exatamente como apresentada no guia de aquisição do MPS.BR. O método propõe uma análise detalhada de todos os requisitos e leva em conta principalmente as necessidades da aquisição para definir a estratégia. A definição dos critérios de seleção do fornecedor também faz parte desta tarefa, pois existe uma dependência entre a estratégia e os critérios para seleção do fornecedor. O produto gerado nesta tarefa é o plano de aquisição com todos os itens completos.

Definir o pedido de proposta: diferente do guia de aquisição, neste método, foi definido que o pedido de proposta deveria ser elaborado a partir de uma tarefa específica. O pedido de proposta é um artefato importante para o processo de aquisição e expressa o conjunto de necessidades que devem ser atendidas pelos fornecedores. O produto gerado nesta tarefa é o pedido de proposta elaborado e pronto para ser enviado aos possíveis fornecedores.

3.2 Seleção do Fornecedor

A atividade de seleção do fornecedor é composta por três tarefas: avaliar a capacidade dos fornecedores, selecionar o fornecedor e preparar e negociar um contrato. No método proposto, esta atividade sofreu algumas mudanças na organização das tarefas, no entanto, respeitou as preconizações do guia de aquisição do MPS.BR.

Enviar o pedido de proposta: esta é a primeira tarefa derivada de “Avaliar a capacidade do fornecedor”. É tratada como uma tarefa individual pois deve identificar os possíveis fornecedores, definir um prazo para o envio das respostas, estabelecer um controle para gerenciar a comunicação com os fornecedores, esclarecer as dúvidas dos fornecedores e receber e protocolar as propostas recebidas. O produto gerado nesta tarefa é o conjunto de propostas enviadas pelos fornecedores.

Avaliar a capacidade dos fornecedores: a avaliação da capacidade dos fornecedores deve acontecer em etapas. Ao final de cada etapa, fornecedores participantes do processo podem ser eliminados por não atender os critérios no plano de aquisição.

A primeira etapa avalia a capacidade dos fornecedores com relação aos RF. Esta etapa é eliminatória e

a análise determina quais os fornecedores devem seguir no processo de aquisição. Assim, é possível minimizar o esforço avaliando os RNF somente dos fornecedores que atendem os RF.

A segunda etapa avalia o conjunto de RNF. São relacionados os fornecedores selecionados e analisados os resultados atingidos em cada requisito. Neste momento podem ser eliminados fornecedores do processo de seleção por não atenderem satisfatoriamente os RNF que são julgados fundamentais pelo comitê de aquisição.

A terceira etapa avalia os produtos propostos. Devem ser utilizadas duas formas de avaliação. Na primeira, os fornecedores selecionados na etapa anterior devem apresentar o seu produto em funcionamento. Esta etapa serve para confirmar as afirmações realizadas pelo fornecedor. Devem fazer parte desta avaliação profissionais designados pelo comitê. É importante colher as impressões destes profissionais, pois serão estes os usuários do produto adquirido. Esta etapa também é importante para esclarecimentos de afirmações ou omissões verificadas nas propostas. Outra forma é a visita a clientes destes fornecedores para verificar o funcionamento do produto em um ambiente real e para colher as opiniões de clientes (preferencialmente) do mesmo ramo de negócio. Esta visita permite identificar a satisfação relacionada ao produto e aos serviços correlatos fornecidos. O produto desta tarefa é um relatório que contém a descrição das três etapas da tarefa de avaliação com o resultado da capacidade dos fornecedores.

Este método sugere que a avaliação e a negociação financeira sejam feitas posteriormente à análise dos resultados. Após esta avaliação, é possível determinar o fornecedor selecionado. A tarefa de preparar e negociar um contrato não foi abordada pelo método proposto.

4. Aplicação e Resultados Obtidos do Método

Após a proposta do método ser elaborada e aprovada, iniciou-se a sua aplicação. Assim, inicialmente foi estabelecida a necessidade. No total foram definidos seis grandes objetivos durante o estabelecimento da necessidade relacionados às características desejáveis para a aquisição:

- Robustez ou que a aplicação seja capaz de armazenar e processar todo o fluxo de informação que a empresa possui de forma satisfatória;
- Escalabilidade ou que tenha habilidade de manipular uma porção crescente de trabalho de forma uniforme e estar preparada para o crescimento do mesmo;
- Completude ou que satisfaça todos os requisitos funcionais e de qualidade;
- Segurança onde os dados estejam seguros de usuários mal intencionados ou de imperícia dos mesmos;
- Flexibilidade para adaptar-se à evolução do negócio e;
- E que possua confiabilidade e integridade nos dados armazenados.

Em seguida os requisitos foram definidos através da análise de 14 setores da indústria. A identificação e a descrição dos requisitos aconteceram através de entrevistas com os responsáveis pelas principais tarefas do setor. Em seguida, foi analisado o *software* existente para uma possível complementação do conjunto de requisitos identificados.

A medida que os RF eram identificados e descritos, os critérios também eram preenchidos. Os RNF, além de identificados e descritos também eram analisados para definir os critérios de avaliação. Todos os critérios foram revistos e verificados pelo comitê de aquisição.

A definição dos requisitos gerou um total de 152 RF. As Tabelas 1 e 2 apresentam os resumos dos resultados dos RF relacionados aos riscos e à classificação conforme a prioridade, respectivamente. As duas tabelas mostram que a maioria dos requisitos identificados representa um alto risco para o negócio da indústria e são imprescindíveis. Esta evidência permite reforçar que um bom processo de avaliação do fornecedor é fundamental pois pode pôr em risco o funcionamento da indústria.

Nível Risco	Quantidade de Requisitos	%
Alto	78	51,3
Médio	47	30,9
Baixo	27	17,8

Tabela 1. Quantidade de requisitos por nível de risco

Nível Prioridade	Quantidade de Requisitos	%
Desejável	48	31,6
Imprescindível	104	68,4

Tabela 2. Quantidade de requisitos por nível de prioridade

Foram identificados 20 RNF e os mesmos são apresentados na tabela 3 agrupados por categoria. Após a conclusão da identificação dos RF e RNF, o comitê se reuniu para definir a estratégia de aquisição. Neste momento, duas alternativas foram definidas como possíveis: (i) manter uma equipe interna para o desenvolvimento de um software que atendesse todos os requisitos pois a empresa tinha receio de contratar um fornecedor externo, depois vários anos mantendo um fornecedor quase exclusivo para atender suas necessidades ou (ii) contratar os serviços de um fornecedor que possua uma solução pronta e customizável para as necessidades da indústria. Depois de várias discussões e do entendimento do comitê dos riscos de cada alternativa, foi definido que manter uma equipe interna, além dos altos custos, da escassez de mão de obra e do longo prazo para obter um software funcional, representava um risco muito grande para a indústria. Neste momento, pesou muito o fato da indústria estar localizada no interior do estado de Santa Catarina e de que a contratação de mão de obra capacitada para manter uma equipe interna de qualidade era praticamente inviável. Desta forma ficou definido que a aquisição seria feita através do contrato com fornecedor de produto e serviços, que aceitasse a customização do produto. Os critérios de aceitação do fornecedor foram definidos como o conjunto de RF e RNF já definidos.

Categoria	RNF
Projeto	Modelagem de processo preliminar, Diretrizes recomendadas pelo PMI para a gerência do projeto de implantação, gerente de projetos com experiência
Implantação	Consultores tempo integral na implantação, Tempo de implantação
Interessados	Avaliação MPS.BR ou CMMI, 5 projetos do porte do adquirente no portfólio
Treinamento	Treinamento de 30% de todos usuários do produto
Manutenção	Métrica formal para definição de tamanho de manutenção, Critério de criticidade para manutenção corretiva
Sistema	Evolução do produto, Fornecimento de módulo de Recursos Humanos
Legal	Atendimento das exigências legais
Financeiro	Manutenções corretivas e evolutivas na proposta, Inclusão de licenças BD na proposta, Número de usuários habilitados, Total de horas contempladas na proposta
Tecnológico	Fornecer ferramenta para customização, Fornecer gerador de relatórios, Possuir ferramenta de workflow

Tabela 3. RNF identificados por categoria

A partir do plano de aquisição aprovado, foi discutido o pedido de proposta. Nessa proposta foi incluído um item que descrevia o adquirente e um item para definir o modelo no qual as propostas deveriam ser apresentadas. No item financeiro, foi decidido não revelar o orçamento da indústria para a aquisição e nem a fonte de recursos.

O pedido de proposta foi enviado para nove fornecedores identificados, doravante, como A, B, C, D, E, F, G, H e I. Estes fornecedores foram escolhidos por serem fornecedores de soluções compatíveis com o pedido de proposta. A Tabela 4 apresenta o resumo dos resultados das tarefas “Enviar pedido de proposta” e “Selecionar Fornecedor”, sendo que esta última, para avaliação, foi dividida em três etapas até chegar na escolha do fornecedor final. Das propostas enviadas, apenas 6 fornecedores foram selecionados para apresentação de seus produtos (linhas 2 e 3 da Tabela 4). Após as apresentações, apenas 2 fornecedores foram para a avaliação do produto. Finalmente, apenas um dos fornecedores foi selecionado (Fornecedor A).

Tarefa/Etapa	Selecionados
Envio pedido proposta	A, B, C, D, E, F, G, H e I
Avaliação RF	A, B, C, D, E e F
Avaliação RNF	A, B, C, D, E e F
Avaliação produto	A e B
Fornecedor selecionado	A

Tabela 4. Resumo dos resultados da seleção do fornecedor

A Tabela 5 apresenta o resumo dos resultados da avaliação dos RF e um exemplo de avaliação de RNF dos fornecedores selecionados. Perceba que todos os fornecedores atenderam mais de 90% dos RF propostos. A coluna RF possui 4 subcolunas que indicam o percentual de não atendimento, de atendimento parcial, de atendimento e a soma dos percentuais de atendimento e atendimento parcial, respectivamente. A última coluna objetiva apresentar de forma positiva o atendimento dos RF pelos fornecedores. Esse resultado era esperado pois o pedido de proposta enviado previamente aos

fornecedores permitiu a seleção daqueles que mais aderiam às necessidades do adquirente. Já os RNF serviram para identificar quais fornecedores atendiam as necessidades mais amplas do adquirente: experiência do fornecedor na área de negócio do adquirente, tempo de implantação, entre outras. A coluna RNF apresenta um dos requisitos apontados: se o fornecedor foi avaliado nos modelos CMMI ou MPS.BR, sendo que, em caso de *Sim*, entre parênteses é mostrado qual modelo e qual nível avaliado.

Fornecedor	RF				RNF
	Não	Parcial	Sim	Parcial+Sim	Avaliação CMMI ou MPS.BR
A	5,04%	10,08%	84,31%	94,39%	Sim (CMMI Nível 3)
B	0,00%	0,00%	100,00%	100,00%	Não
C	4,20%	10,64%	85,15%	95,79%	Não
D	4,76%	9,24%	85,99%	95,23%	Sim (CMMI Nível 3)
E	7,00%	34,73%	58,26%	92,99%	Não
F	8,96%	14,85%	76,19%	91,04%	Não

Tabela 5. Resumo da Avaliação RF e RNF

Na avaliação do produto foram feitas apresentações dos seis fornecedores que participavam do processo. As apresentações foram formatadas no conteúdo, na sequência de apresentação e no tempo que deveria ser utilizado. O objetivo da apresentação foi ratificar as informações apresentadas nas propostas enviadas pelos fornecedores e verificar o produto oferecido de forma prática na apresentação. Para isso, participaram da apresentação um grupo de 17 profissionais representativos das áreas da indústria adquirente, selecionados pelo comitê. Como resultado da apresentação, foi gerado um relatório com o seguinte conteúdo: os pontos fortes observados, os pontos fracos e as percepções de cada profissional presente na apresentação sobre o produto e os serviços oferecidos.

Após as apresentações dos seis fornecedores, o comitê analisou os resultados dos relatórios e definiu que seguiria com apenas dois, mesmo antes de realizar a visita as instalações dos fornecedores e aos seus clientes como previsto no processo. Durante as apresentações, os fornecedores fizeram afirmações e apresentaram esclarecimentos que foram muito valiosos para a avaliação. Muitas das afirmações que constavam da proposta dos fornecedores e que determinavam o percentual de atendimento dos RF e RNF, foram julgadas insuficientes quando demonstradas. Dessa forma a apresentação foi suficiente para que o comitê, juntamente com os profissionais selecionados tomassem a decisão sobre os fornecedores que seguiriam no processo, deixando para a etapa seguinte a visita as instalações do fornecedor e de seus clientes. A postura dos representantes do fornecedor, seu conhecimento do negócio do adquirente, e outros RNF selecionados foram julgados nas apresentações. Neste momento foram designados os fornecedores A e B para seguir no processo e foi determinado que um cliente de cada fornecedor seria visitado. A partir das visitas, concluiu-se que os dois fornecedores selecionados atendiam o pedido de proposta de forma satisfatória e a avaliação de valores seria decisiva. Desta forma, o comitê determinou que a análise financeira seria feita por alguns membros do comitê, incluindo o Diretor Superintendente, proprietário da indústria. Depois de análise e negociações de valores e condições de contrato, foi tomada a decisão pelo fornecedor A.

5. Conclusão

Através da aplicação do método proposto, foi possível selecionar o fornecedor de forma clara e transparente para todos os envolvidos no processo. Ficou evidente para a indústria adquirente, que o fato de possuir um produto que não a atendia, não era o suficiente para adquirir um novo produto e dessa forma, o estudo inicial que gerou o plano de aquisição, baseado no MPS.BR, foi fundamental para conhecer as necessidades da aquisição.

O processo de aquisição, colocado em prática, serviu como base importante para suportar todas as atividades necessárias para uma boa aquisição. Os fornecedores participantes do processo fizeram comentários positivos durante o processo de aquisição, principalmente ao receber um pedido de proposta formatado e com o conjunto de requisitos, funcionais e não funcionais, que deveriam atender.

Houve um fornecedor que criticou o processo, pois no seu entendimento, privilegiava as grandes empresas. Este fornecedor entendeu que possuir processos documentados de implantação, avaliação da qualidade, métrica para identificar tamanho e esforço para manutenções, entre outros, eram privilégios de empresas de grande porte e que desta forma, os fornecedores menores estariam fora do processo de seleção de fornecedores. Raciocínio que não é válido pois os processos de engenharia de software podem ser aplicados em empresas de qualquer porte, mesmo a avaliação em modelos de qualidade (principalmente o MPS.BR) é acessível à pequenas empresas através de auxílios da Softex.

Após a aplicação do método, ficou claro que a avaliação dos RF serviu para definir os fornecedores que tinham condições de entrar no processo de aquisição. A avaliação dos RNF permitiu avaliar os fornecedores de forma mais ampla, determinando a capacidade de atendimento, a experiência em outros projetos de áreas semelhantes, a estrutura para realização de projetos de implantação, entre outros. Este procedimento permitiu que os fornecedores que tivessem maior aderência aos RF, principalmente os indispensáveis, continuassem no processo. Aqueles fornecedores que atendiam parcialmente os RF, e que seguiram no processo, poderiam propor estratégias para atender estes requisitos. Como os seis fornecedores que responderam o pedido de proposta possuíam um percentual alto de aderência aos RF, os RNF foram decisivos para a seleção do fornecedor. Os preços propostos pelos fornecedores só foram discutidos na última etapa da seleção, quando já havia certeza de que as opções de fornecedores não comprometeria o resultado da aquisição. Neste sentido, foi observada a importância de permanecerem dois fornecedores na etapa de negociação, pois a concorrência seria favorável ao adquirente. Mesmo assim, não foi o fornecedor com melhor preço o selecionado.

Deve ser também relatada a satisfação do adquirente com o processo de aquisição. No início do processo havia muitas dúvidas sobre como aconteceria a escolha do fornecedor. Após a definição do método de seleção do fornecedor, havia a clareza necessária para realizar o projeto de forma objetiva e com medições dos resultados. Neste processo, o Guia de Aquisição proposto pelo MPS.BR foi fundamental e provou a sua eficiência na adequação para uma aquisição específica e complexa realizada na indústria.

Com o resultado positivo da aplicação do método, que abrange parte do processo de aquisição proposto pelo MPS.BR, pretende-se complementar o método proposto com as outras atividades de aquisição: monitoração do contrato e aceitação pelo cliente. A experiência adquirida com o trabalho realizado permite afirmar a importância da gestão do processo completo de aquisição e os benefícios colhidos.

Como o processo proposto foi elaborado para atender empresas privadas, é importante destacar a necessidade de flexibilização para viabilizar a sua reutilização. Para novas aplicações do método é necessário verificar se as atividades propostas atendem às necessidades do adquirente ou se ajustes no método devem ser feitos antes de iniciar a sua aplicação.

Referências

[Conry-Murray 2009] Conry-Murray, I. (2009). A measure of satisfaction - what customers hate and love about their vendors – and what they can do to build better relationships. *InformationWeek*, 1(1218):19–24.

[Guerra and Alves 2004] Guerra, A. C. and Alves, A. M. (2004). *Aquisição de produtos e serviços de software*. Elsevier, 1st edition.

[IEEE 1998] IEEE (1998). *IEEE STD 1062 – IEEE Recommended Practice for Software Acquisition*. IEEE.

[Jones 2001] Jones, C. (2001). Conflict and litigation between software clients and developers. Technical report, Software Productivity Research, Inc.

[Pressman 2002] Pressman, R. (2002). *Engenharia de Software*. São Paulo – McGraw-Hill, 5th edition.

[SOFTEX-a 2009] SOFTEX-a (2009). Guia de aquisição. In *MPS.BR - Melhoria de Processo do Software Brasileiro*, page 87. SOFTEX.

[Sommerville 2007] Sommerville, I. (2007). *Engenharia de Software*. São Paulo – Pearson - Addison Wesley, 8th edition.

Aplicação da Memória Organizacional no contexto da implantação do processo de melhoria de *software*

Edson Wu Quan Yhe (edson.wu@gmail.com)
Jandira Guenka Palma (jgpalma@uel.br)
Departamento de Computação – Universidade Estadual de Londrina (UEL)
CEP 86051-990 – Bloco J – Sala 305A – Londrina – Paraná

Rafael Soares Parente (sparente.rafael@gmail.com)
Guenka Desenvolvimento de Software
Av. Adhemar Pereira de Barros, 730
CEP 86050-190 – Londrina – Paraná

Abstract. *This work analyses the influence which is provided by the Organizational Memory on a deployment of the software process improvement, where the MR-MPS (Reference Model for Software Process Improvement) is adopted. To that end, it was performed a case study in a small company where is being deployed the specifications of the levels G and F of the MPS.BR (Brazilian Software Process Improvement), in order to improve the implementation and the institutionalization of the processes of the model employed.*

Resumo. *Este trabalho analisa a influência proporcionada pela Memória Organizacional na implantação dos processos de melhoria contínua de software, sendo o MR-MPS (Modelo de Referência para Melhoria do Processo de Software) o modelo de software empregado. Para tanto, foi realizado um estudo de caso em uma pequena empresa, na qual está sendo implantada as especificações dos níveis G e F do MPS.BR (Melhoria de Processos do Software Brasileiro), com a finalidade de aprimorar a implantação e a institucionalização dos processos do modelo empregado.*

1. Introdução

Com o crescimento da competitividade no mercado atual, tornou-se necessário agregar valor aos processos e produtos, embutindo como fator diferencial a qualidade. Tendo em vista tal situação do mercado e tendo como foco principal o cliente, a empresa Guenka Desenvolvimento de *Software*, teve a iniciativa de implantação do MR-MPS (Modelo de Referência para Melhoria do Processo de *Software*), visando maior satisfação dos clientes e melhores resultados em seus projetos.

Em 2005 a empresa iniciou o processo de implantação do MR-MPS objetivando o nível G. Um ano após o início da implantação a empresa é avaliada positivamente pela Softex no nível G do MPS.BR, porém como resultado no pós-avaliação houve um alto grau de deterioração no uso dos processos propostos, causada pela má formulação de um ambiente de melhoria, consistente e favorável ao aprendizado.

Desse modo, o texto aborda o uso da memória organizacional como metodologia fundamental para uma efetiva implantação da melhoria na empresa, tendo como base experiências vivenciadas pela equipe de implantação da melhoria e relatos de outras implantações [1]. A seção 2 descreve,

com uma breve explanação, o início do processo de melhoria na empresa e o seu estado corrente. A seção 3 descreve o processo de implantação do nível F e as evidências da mudança de paradigma realizada na empresa e, por fim, na seção 4 são descritas as considerações finais.

2. Implantação da Melhoria Contínua de *Software*

Com apoio e incentivo da alta gerência e dos sócios da empresa, o processo de melhoria é instituído com o intuito de proporcionar, de forma generalizada, a redução de custos e tempo de desenvolvimento e entrega dos projetos e, principalmente aumentar de forma substancial a satisfação dos clientes e colaboradores internos à empresa.

Nessa vertente de melhoria, a empresa iniciou a implantação do modelo MR-MPS em 2005 e atualmente encontra-se em fase de finalização da implantação do nível F de maturidade, já apresentando diversas melhorias alcançadas em comparação à implantação anterior.

Os indicativos de tal maturidade serão abordados nas próximas seções, descrevendo os pontos negativos e positivos de cada uma das estratégias de implantação utilizadas pela empresa para os dois níveis de maturidade, o nível inicial G e o atual F.

2.1. Dificuldades encontradas no nível G e soluções propostas

O processo de implantação da melhoria e a gestão do conhecimento e das informações geradas, no nível G, foram realizados de forma ainda imatura, sem um controle específico. Devido a este fato, ativos intangíveis, importantes para qualquer tipo de processo de melhoria organizacional, não eram armazenados e disponibilizados corretamente. Algumas das dificuldades encontradas na implantação deste nível foram:

- **Ausência de controle no armazenamento de informações:** os ativos gerados pela empresa, tais como documentos, planejamento de projetos, levantamentos de requisitos e processos executados careciam de políticas de gerenciamento que favorecessem a manutenção das mesmas;
- **Ausência de uma gestão do conhecimento eficaz:** muitas das informações importantes e necessárias para o acesso imediato encontravam-se descentralizadas e redundantes, encontradas apenas em computadores pessoais, outras em repositórios, sem nenhum nível de controle de armazenamento e distribuição explícito, sendo tal controle realizado apenas verbalmente;
- **Ausência de ferramentas ou metodologias de apoio:** não havia na empresa a adoção de ferramentas de apoio que auxiliassem na implantação da melhoria, além da insuficiente maturidade em tal processo, por parte da equipe de implantação para que os processos, produtos de trabalho e documentos gerados fossem documentados e disponibilizados mediante uma metodologia que apoiasse o plano de melhoria e o processo em si.

Tais fatos, ligados à inerente dificuldade de implantação de uma cultura organizacional diferente, criaram na empresa, por parte de colaboradores internos, certa dificuldade no uso dos processos padronizados e na utilização dos documentos existentes. As atividades relacionadas à melhoria eram vistas muitas vezes como burocráticas, pois além da utilização dos processos, não havia a disseminação necessária do conhecimento para estes em meios de armazenamento organizados e gerenciados. Um dos efeitos de tal carência, veio à tona com a expansão da empresa e, conseqüente contratação de novos colaboradores, criando-se assim, novas funções e cargos na empresa, fazendo com que os funcionários, inicialmente comprometidos com o processo de melhoria, assumissem novas responsabilidades de forma que a delegação de atividades e responsabilidades do processo de melhoria, para outros membros da empresa, fosse inicialmente omitida.

Assim, como pontos desmotivadores ao uso dos processos criados na implantação do nível G, se destacam:

1. Dificuldade de acesso e visualização dos fluxos de atividades e processos;
2. Dificuldade de obtenção de informações e falta de credibilidade no meio de retenção das informações;
3. Processos burocráticos e de difícil compreensão. Os objetivos específicos de cada processo não se encontravam disponível à consulta, o que pode comprometer a visão de benefícios que as atividades pudessem trazer.

Contudo, com o intuito de avançar no processo de melhoria e alcançar objetivos ainda pendentes da implantação anterior, a empresa dá continuidade ao plano de melhoria, encontrando-se atualmente na implantação do nível F do MR-MPS, onde teve início a reestruturação dos processos e procedimentos anteriormente utilizados no nível de maturidade predecessor.

3. Implantação do Nível F

A oportunidade de mudança, e correção dos problemas evidenciados na implantação anterior, veio com a implantação do nível F, onde para o avanço coordenado no processo de melhoria, objetivando-se resultados não alcançados na implantação anterior, a reestruturação dos processos anteriores tem fundamental importância.

A tabela 1 demonstra uma relação de causa e efeito entre os problemas evidenciados na implantação anterior e uma breve descrição das ações corretivas realizadas na implantação atual, sendo estas detalhadas através das seguintes atividades:

- Criação de funções e cargos na empresa dedicados ao processo de melhoria;
- Estabelecimento e desenvolvimento de uma metodologia de apoio: a memória organizacional;
- Desenvolvimento e distribuição dos processos;

- Institucionalização e treinamento nos processos criados;
- Acompanhamento das atividades.

	Causa	Efeito	Ação Corretiva
1	Ausência de controle na manipulação das informações	Documentos e/ou artefatos utilizados em atividades gerais eram dificilmente encontrados em função da descentralização dessas informações.	Centralização do meio de retenção da informação, organizando estes em grupos de conhecimento similares.
2	Não formalização de um política de gestão do conhecimento	Os conhecimentos gerados e/ou existentes eram mal reutilizados em função da falta de políticas e/ou práticas que fomentassem o aprendizado organizacional	O conhecimento embutido nos processos se apresentam distribuídos coerentemente mediante instruções de trabalho, fases e atividades e guias complementares. A reutilização dos conhecimento gerados pode ser realizada diretamente nos artefatos gerados, estes podendo ainda apresentar problemas no projeto ou bem sucedidos.
3	Ausência de ferramentas ou metodologias de apoio	Deterioração dos processos criados, dificultando suas institucionalizações.	Uso de conceitos da Memória Organizacional, auxiliando-se numa ferramenta chamada Confluence® para tal procedimento.
4	Dificuldade de acesso e visualização dos fluxos de atividades e processos	Perda do controle sobre os processos, gerando conhecimentos pouco reutilizáveis e comprometedores à veracidade quanto ao fluxo normal dos processos. Outro impacto é o julgamento dos processos como burocráticos, sem objetivos explícitos.	Criação do fluxo de atividades visuais e detalhadas, contendo processos disponíveis à todos (quando necessário) e que apresentam especificações de atividades predecessoras e sucessoras e uma instrução de trabalho. Há também reuniões que têm o objetivo de levantar melhorias quanto aos próprios processos, desde a documentação das mesmas até a implantação ou remoção de alguns desses processos.
5	Ausência de cargos e funções específicos para a manutenção dos processos de melhoria	Os funcionários, inicialmente comprometidos com suas próprias atividades, passaram a assumir novas responsabilidades.	Estabelecimento de funções e cargos na empresa dedicados ao processo de melhoria, sendo também realizados treinamentos para facilitar a institucionalização dos tais processos.

Tabela 1. Causa Efeitos dos Problemas encontrados e Ações Corretivas

Tais atividades serão apresentadas em maiores detalhes nas próximas seções.

3.1. Criação de funções e cargos na empresa dedicados ao processo de melhoria

Como passos primordiais para a reestruturação do plano de melhoria foram criados cargos e funções na empresa, os quais se apresentam exclusivamente dedicados ao processo de melhoria, objetivando-

se assim uma dedicação especial ao processo de implantação, capacitação dos colaboradores da empresa e acompanhamento de todas as etapas da melhoria.

Com a criação de tais cargos, alguns aspectos passaram a ser discutidos e questionados, efetivando assim a criação de um ambiente pró-ativo, onde muitos detalhes da implantação anterior passaram a serem vistos por um grupo o qual não teve participação em tal processo, facilitando a compreensão da necessidade de reestruturação do processo de melhoria.

3.2. A Memória Organizacional como metodologia de apoio

A memória organizacional (MO), a qual pode ser definida como um conjunto de quaisquer artefatos e documentos registrados na organização, onde o processo de criação e desenvolvimento desse conjunto deve estar disponível para possíveis reutilizações. A centralização do ponto de acesso desse conhecimento deve facilitar a organização, recuperação e armazenamento de seus processos, das experiências e tecnologias [2,3,4,5].

A função de uma MO, além de fornecer um ambiente provedor de informações, é aumentar a competitividade da organização por aperfeiçoar sua forma de gerir conhecimentos, proporcionando um maior compartilhamento e reuso do conhecimento corporativo e individual e das lições aprendidas na execução das tarefas da organização, além de responder às principais questões que se faça sobre a organização, seu ambiente, seus processos, seus produtos, tomadas de decisão, e ainda disponibilizar conhecimentos, de quem o tenha, para executar determinada tarefa no tempo hábil e com as informações necessárias para o trabalho [2,3,6].

Como supracitado a memória organizacional representa o controle efetivo dos ativos tangíveis e intangíveis criados na empresa, e devido a este fato, a atual implantação do nível F tem sua base solidificada na memória organizacional. Porém o uso ou apenas assimilação da memória organizacional por si só, não tem efetiva contribuição nos processos se não respeitadas os seguintes fatores: (i) os meios de retenção (pessoas, sistemas, documentos, etc.) e a informação contida nestes; (ii) os processos de aquisição e disponibilização da informação e, (iii) os efeitos da memória e dos processos da informação na organização [2,3,6].

Com objetivo de encontrar soluções para os problemas observados na implantação anterior e com objetivo maior de sustentar a melhoria com alto grau de aceitação, foram estipulados alguns mecanismos o quais são apoiados por uma ferramenta comercial, denominada *Confluence®*, possibilitando o acesso controlado para todos os membros da empresa com *login* e senha na *intranet* empresarial, facilitando assim a disseminação e assimilação do conhecimento e informações geradas. A figura 1 mostra a estrutura criada de acordo com o conceito de memória organizacional, onde tal estrutura apresenta-se como o escopo essencial do conhecimento agregado para os processos de desenvolvimento da empresa, onde cada um dos *links* apresentados aponta para algum tipo de instrução, algumas mais elaboradas (Instruções de Trabalhos), outras com um grau menor de detalhes, tendo objetivo demonstrarem o fluxo das atividades (Fases e Atividades).

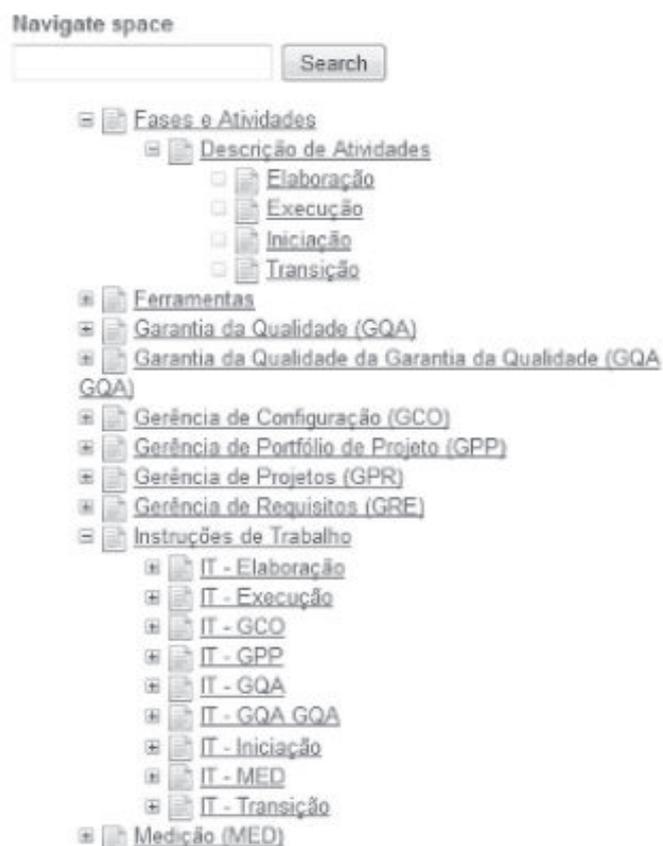


Figura 1. Estrutura organizacional dos processos implantados

3.3. Desenvolvimento e disseminação dos processos

Iniciado o uso da memória organizacional, os processos desenvolvidos passam a ser objeto de avaliação e visualização de todos, em função de suas inserções e disponibilizações realizadas internamente à empresa. Para exemplificar tais processos, a figura 2 apresenta uma das fases definidas no PSG (Processo de Software Guenka). Todas as atividades, para cada uma das 4 fases (Iniciação, Elaboração, Execução e Transição) apresentam: (i) escopo, (ii) documentos necessários para suas efetivações e (iii) suas instruções de trabalho a fim de garantir a compreensão e, conseqüentemente, a eficaz implantação de tais processos na empresa.

A figura 2 representa a fase de elaboração, a qual apresenta três atividades sendo elas: Detalhar Requisitos, Estabelecer/Atualizar plano de projeto e Reunião de Kick-off/Iteração. Além disso, há subprocessos, indicando que estes, fazem parte de um processo específico, neste caso, o processo de Gerência de Configuração (GCO), tendo como seus subprocessos: Solicitar *Baseline*, Estabelecimento de *Baseline*, Auditoria de Configuração e Controle de Mudanças. Havendo ainda nesta fase subprocessos de Garantia da Qualidade (GQA), sendo este a Auditoria de Qualidade.

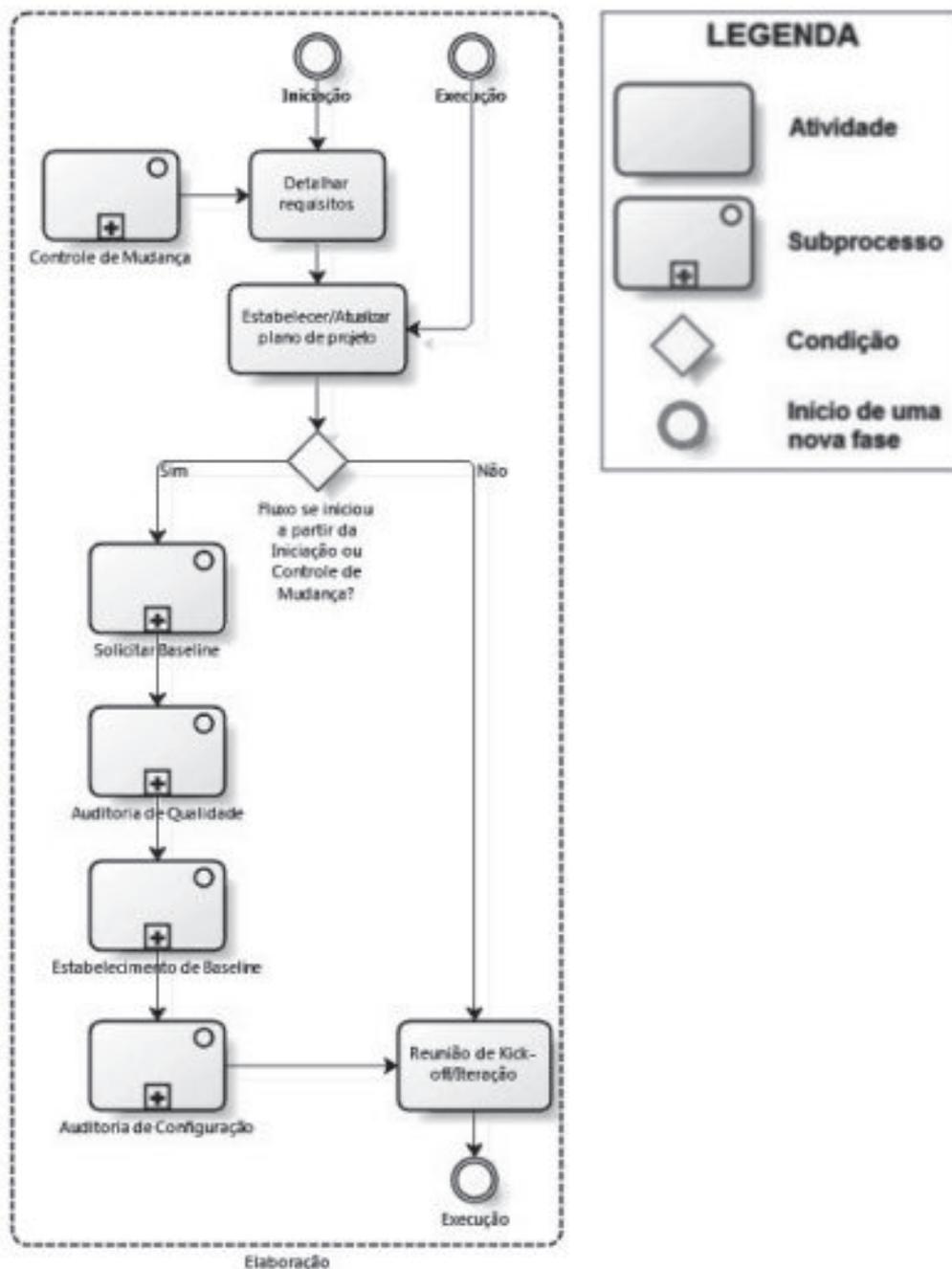


Figura 2. Fluxo de Atividades da Fase de Elaboração

A figura 3 apresenta as atividades vinculadas à Gerência de Configuração. Verifica-se que cada um dos seus subprocessos encontra-se definidos com um escopo de atividades próprio. O aspecto a ser considerado é o desmembramento dos subprocessos citados acima na forma de atividades, o Controle de Mudança, por exemplo, se divide em 4 atividades: Criar Solicitação de Mudança, Avaliar Solicitação de Mudança, Liberar *Baseline* e Comunicar rejeição, cada uma possuindo instruções de trabalho específicas. O desmembramento das atividades em subprocessos e atividades visam melhorar a compreensão dos processos.

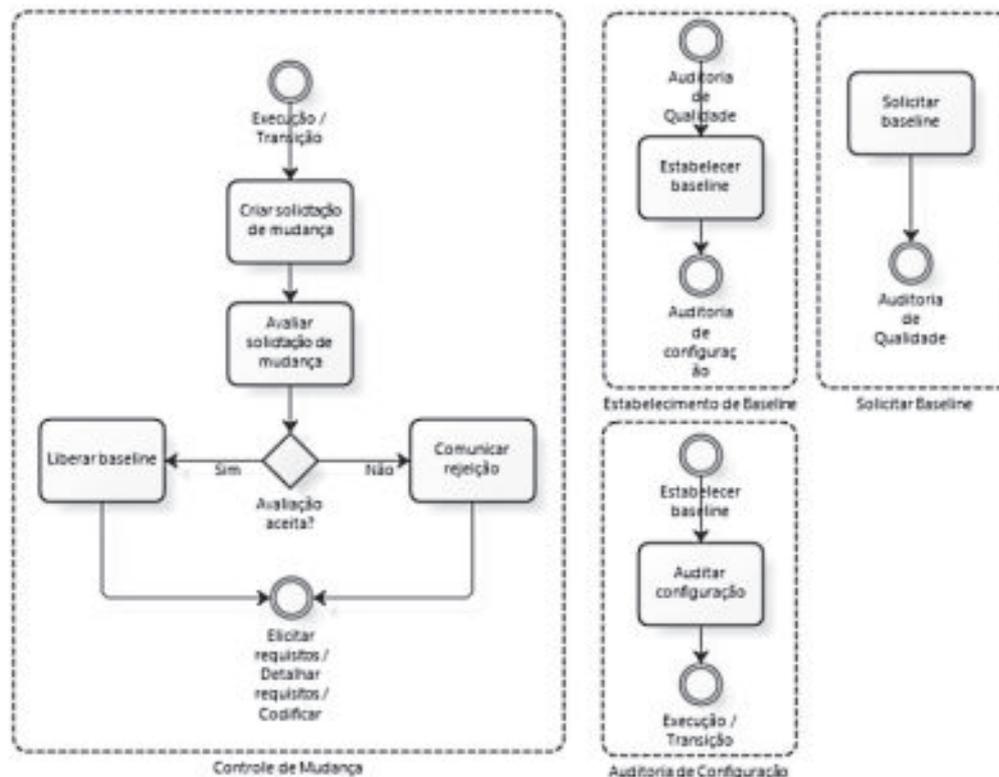


Figura 3. Atividades do Processo de Gerência de Configuração

3.4. Institucionalização e treinamento nos processos criados

Como todo processo de implantação, existe a necessidade de treinamento, capacitação e institucionalização dos processos desenvolvidos para com os colaboradores da empresa, sendo esta uma das tarefas de maior importância, pois sem essa atividade, os processos podem vir a não serem utilizados, devido à falta de conhecimento nos mesmos.

Com a organização atual dos processos e o devido armazenamento e disponibilização dos mesmos, o tempo de treinamento e o grau de dificuldade de uso dos processos e utilização dos documentos foram minimizados, pois a formulação atual apresenta-se intuitiva e inerentemente descritiva, de forma que, por mais que tenha havido um esforço maior no desenvolvimento dos processos, houve proporcionalmente um decréscimo no tempo gasto com a institucionalização e treinamento dos colaboradores nos processos padrão.

3.5. Acompanhamento das atividades

Acompanhar o andamento das atividades e processos utilizados tem importância fundamental, pois permite que sejam avaliadas falhas e desvios no processo em tempo real, com o uso da memória organizacional efetivamente implantada, o controle de status dos produtos de trabalho, produzidos por cada uma das equipes de projeto da empresa, pode ser acompanhado, garantindo assim maior aderência aos processos e padrões implantados.

Tal acompanhamento é realizado através de reuniões de status de projetos, atividade a qual pertence ao nível F do MR-MPS [7], porém, além deste acompanhamento, podemos acompanhar cada uma das atividades realizadas nos projetos em tempo real, pois a ferramenta implantada para auxílio à memória organizacional provê um mecanismo de atualizações recentes. Deste modo, torna-se mais eficaz o controle do fluxo de atividades, através dos acompanhamentos dos produtos de trabalho e artefatos criados, em determinado período de tempo.

4. Considerações finais

Este trabalho apresentou a iniciativa da empresa Guenka para resolver conflitos inerentes da implantação do processo de melhoria, onde o modelo utilizado foi o MR-MPS, na implantação dos níveis G e F, ressaltando a importância do bom uso da memória organizacional e de seus efeitos na eficácia da implantação.

Como benefícios da efetiva implantação da memória organizacional é observado o uso contínuo e aprimorado dos processos instaurados, de forma que os membros da empresa se sintam mais seguros em utilizá-los por passarem a visualizar a estrutura criada como uma gestora de conhecimentos, auxiliando empresa na tomada de decisões e na execução das diversas atividades do desenvolvimento de *software* e a adoção desta metodologia estando inerentemente ligada à cultura organizacional.

Com os resultados obtidos no decorrer deste trabalho, a empresa será avaliada posteriormente para a qualificação no nível F do MPS.BR, tendo a certeza de que os processos após a certificação continuarão ativos na empresa, e muito mais, sabendo que o conhecimento não se encontra mais de forma desorganizada e individualista. Como trabalhos futuros a empresa tem um projeto de desenvolvimento de um portal do conhecimento, aberto para o público externo a empresa, onde clientes e colaboradores da empresa poderão interagir de forma inteligente e eficaz no processo de desenvolvimento de *software*, agregando assim, muito mais valor a seus produtos e clientes.

Agradecimentos

Os autores agradecem a Heber Alexsandro Nascimento, gerente de processos e a Brahim José Malaque Neto, gerente de área e portfólio, colaboradores da empresa Guenka, pelo apoio e oportunidade concedidos, viabilizando assim o presente trabalho.

Referências

- [1] A. Barreto, M. Montoni, G. Santos, e A. R. Rocha, “Gerência de conhecimento como apoio para a implantação de processos de software”, in ProQualiti - Qualidade na Produção de Software, vol. 2, Novembro 2006, pp. 45–50
- [2] O. de Gusmão Freitas Junior, “Um modelo de sistema de gestão do conhecimento para grupos de pesquisa e desenvolvimento,” Tese de doutorado. Programa de Pós-Graduação em Engenharia de Produção, Universidade Federal de Santa Catarina, Florianópolis, 2003.
- [3] M. Ramage e F. Reiff, “Links between organizational memory and cooperative awareness”. 1996. Disponível em: http://www.comp.lancs.ac.uk/computing/research/cseg/projects/evaluation/OM_CA.html
- [4] T. H. Davenport e L. Prusak, Conhecimento Empresarial. Rio de Janeiro: Campus, 1998.
- [5] J. T. Filho, Gerenciando conhecimento: como a empresa pode usar a memória organizacional e a inteligência competitiva no desenvolvimento de negócios. Rio de Janeiro: SENAC, 2000.
- [6] H. E. S. ao, “Memória organizacional”. Disponível em: <http://www.batebyte.pr.gov.br/modules/conteudo/conteudo.php?conteudo=1716>
- [7] SOFTEX, MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral. SOFTEX, 2009. Disponível em: http://www.softex.br/mpsBr/_guias

Construindo um Modelo Especialista de Estimativa de Esforço de Aplicações Web usando Redes Baesianas

Emilia Mendes (emilia@cs.auckland.ac.nz)

Computer Science Department – The University of Auckland
Private Bag 92019, Auckland, New Zealand

Abstract. *The objective of this paper is to describe a case study where Bayesian Networks (BNs) were used to construct an expert-based Web effort model. We built a single-company BN model solely elicited from expert knowledge, where the domain expert was an experienced Web project manager from a small Web company in Auckland, New Zealand. This model was validated using data from eight past finished Web projects. The BN model has to date been successfully used to estimate effort for four Web projects, providing effort estimates superior to those based solely on expert opinion. Our results suggest that, at least for the Web Company that participated in this case study, the use of a model that allows the representation of uncertainty, inherent in effort estimation, can outperform expert-based estimates. Another nine companies have also benefited from using Bayesian Networks, with very promising results.*

Resumo. *O objetivo deste trabalho é descrever um estudo de caso onde Redes Bayesianas (RBs) foram utilizados para construir um modelo especialista de estimativa de esforço de aplicações Web. Construímos um modelo RB específico a uma empresa Web de pequeno porte localizada em Auckland. Esse modelo foi exclusivamente criado a partir do conhecimento do especialista de domínio dessa empresa, que era um experiente gestor de projetos Web. Este modelo foi validado utilizando dados de oito projetos Web que já haviam sido terminados. Uma vez validado, o modelo RB passou a ser utilizado com sucesso pela empresa para estimar o esforço de novos projetos Web, fornecendo estimativas de esforço superiores às que se basearam apenas na experiência do especialista. Os resultados sugerem que, pelo menos para a Empresa Web que participou neste estudo de caso, a utilização de um modelo que permite a representação da incerteza, inerente à estimativa de esforço, pode superar as estimativas de especialistas. Outras nove empresas também já se beneficiaram do uso das RBs, com resultados muito promissores.*

1. Introdução

Um dos alicerces do gerenciamento de projetos Web é a estimativa de esforço, o processo pelo qual o esforço de desenvolvimento de um projeto é estimado e utilizado como base para prever os custos e alocar recursos de projeto de forma eficaz, permitindo assim que estes sejam entregues no prazo e dentro do orçamento. A estimativa de esforço é um domínio muito complexo, onde a relação entre fatores não é determinista, e de natureza inerentemente incerta. No contexto da estimativa de esforço de desenvolvimento de projetos Web, diversos estudos investigaram o uso de técnicas de previsão do esforço. No entanto, até agora, apenas Mendes (2007^a; 2007^b; 2007^c), e Mendes e Mosley (2008) investigaram a inclusão explícita, e uso, da incerteza, inerente à estimativa de esforço, em modelos de estimativa de esforço para Web. Nesses estudos, modelos híbridos de Redes Baesianas (estrutura criada pelo especialista, e probabilidades aprendidas automaticamente usando dados) foram construídos, e apresentaram de uma forma geral resultados promissores, quando comparados a outras técnicas de estimativa de esforço. Uma Rede Baesiana (RB) é um modelo que permite a representação de um domínio complexo de conhecimento de uma forma que permite a manipulação desse

conhecimento levando em consideração o grau de incerteza inerente a esse domínio. Tal incerteza deve idealmente ser considerada em todas as tomadas de decisão relativas a esse domínio do conhecimento (Jensen, 1996).

Toda RB possui duas partes, detalhadas a seguir.

A primeira é a parte qualitativa da Rede, descrita como um grafo direto acíclico (dígrafo) (veja Figura 1). Os nós desse dígrafo representam as variáveis (fatores) que são relevantes ao domínio que está sendo representado, variáveis estas que podem ser de diversos tipos, como por exemplo, variáveis visíveis ou ocultas e variáveis categóricas ou numéricas. Os arcos do dígrafo representam o relacionamento causal entre as variáveis da Rede, e representam de forma probabilística a incerteza associada a esses relacionamentos. A segunda parte de uma BN, a sua parte quantitativa, associa a cada nó da Rede uma Tabela de probabilidades condicionais (TPC), que na verdade representa a distribuição de probabilidades relativa àquele nó. A TPC de um nó Pai (nó que não é apontado por nenhum arco) descreve a probabilidade relativa a cada estado (valor) desse nó. Um exemplo disso é mostrado na Figura 1 na TPC do nó "Tamanho (Páginas Web Novas)". A TPC de um nó filho f (nó que é apontado por pelo menos um arco) descreve a probabilidade relativa a cada estado de f condicional a todas as combinações de estados dos nós que são os Pais de f . Um exemplo é mostrado na Figura 1 através das TPCs dos nós "Esforço Total" e "Tamanho (número total de páginas Web)". Uma vez que a BN seja especificada, pode então ser utilizada para diversos tipos de raciocínio, tal como análises de raciocínio preditivo ou diagnóstico. Para que isso ocorra, basta que dados correspondendo a evidência que se possui (como exemplos, evidência sobre o número estimado de páginas Web a serem desenvolvidas, o número estimado de desenvolvedores a trabalharem no projeto Web) sejam automaticamente re-calculadas utilizando o teorema de Bayes.

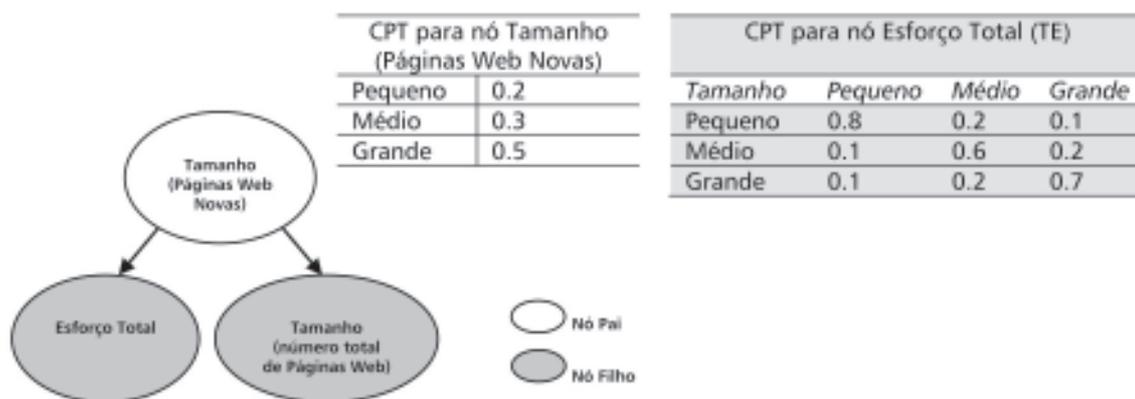


Figura 1. Uma RB com três nós e duas CPTs

Uma vez que uma RB tenha sido construída, evidência (por exemplo, valores) pode ser inserida em qualquer nó da rede, e as probabilidades para os nós restantes é calculada automaticamente usando a regra de Bayes. Portanto RBs podem ser usadas para diferentes tipos de raciocínio, como preditivo e "what-if", analisando e investigando o impacto de como as mudanças em alguns nós afetam os outros nós. No contexto da estimativa de esforço de desenvolvimento de aplicações Web, há problemas com a construção de modelos que sejam gerados automaticamente a partir de dados, ou modelos híbridos, pelos seguintes motivos:

- O volume de dados usados para construir um modelo RB deve idealmente ser grande o suficiente para capturar todas (ou quase todas) as combinações de estados relevantes entre variáveis, de forma que as probabilidades possam ser aprendidas a partir dos dados, ao invés de elicitadas manualmente. Em tais circunstâncias, é muito improvável que os dados utilizados sejam fornecidos por uma única empresa, e estudos anteriores mostraram que o uso de dados voluntariados por diversas empresas Web não apresenta uma precisão de estimativa de esforço tão boa quanto o uso de dados de uma única empresa Web. Até o momento, o maior conjunto de dados de projetos Web está disponível na base de dados Tuketuku (195 projectos) (Mendes et al. 2005b). Este conjunto tem sido usado para construir modelos RB orientados a dados e híbridos, porém os resultados, embora algumas vezes promissores, não têm sido excelentes. Acreditamos que um dos motivos é devido ao tamanho reduzido do conjunto de dados disponíveis.
- Mesmo quando existe um grande volume de dados, há a questão relativa ao conjunto de variáveis que caracteriza os dados. É improvável que estas representem todos os fatores relevantes a um único domínio (por exemplo, estimativa de esforço de aplicações Web) para todas (ou grande parte) das empresas Web que são candidatas a usar o modelo orientado a dados ou híbridos criados usando aquele conjunto de dados. Este foi o caso do conjunto de dados Tuketuku, embora suas variáveis tenham sido selecionadas através de duas avaliações. No entanto, pode-se argumentar que se o modelo que está sendo criado é híbrido, novas variáveis (fatores) podem ser adicionadas ou removidas desse modelo, após a sua criação. O problema é que cada nova variável acrescentada ao modelo representa um conjunto de probabilidades que devem ser elicitadas a partir do zero, o que pode ser uma tarefa extremamente demorada e desestimulante para a empresa.
- Diferentes estruturas e algoritmos de aprendizagem de probabilidade podem levar a diferentes precisões da estimativa, portanto, pode-se precisar usar diferentes modelos RBs de forma a comparar a sua exatidão, o que também pode ser uma tarefa muito demorada.
- Ao utilizar um modelo híbrido, a estrutura da RB deveria idealmente ter sido elicitada conjuntamente por mais de um especialista de domínio, e de preferência pertencendo a empresas distintas; caso contrário, o modelo construído não poderá ser genérico o suficiente para atender a uma ampla gama de empresas. Há situações, porém, onde não é possível ter vários especialistas de diferentes empresas trabalhando cooperativamente em uma única estrutura RB. Tal situação ocorre quando todas as empresas envolvidas compartilham potencialmente o mesmo mercado de clientes. Este foi o caso das empresas que participaram no contexto desta pesquisa.
- Idealmente, as probabilidades usadas pelos modelos orientados a dados ou híbridos devem ser revistas por pelo menos um especialista de domínio, após terem sido aprendidas automaticamente pela ferramenta RB. No entanto, dependendo da complexidade do modelo RB, isto pode representar ter que inspecionar milhares de probabilidades, o que pode não ser viável. Uma forma de minimizar este problema é adicionar fatores (nós) adicionais ao modelo RB, a fim de reduzir o número de relações causais atingindo os nós filho (divórcio) (Jensen 1996); no entanto, todas as probabilidades referentes a esses novos fatores adicionais ainda precisariam ser elicitadas manualmente pelos especialistas do domínio.

- A escolha da discretização de valores, os algoritmos de aprendizagem da estrutura da rede e da estimativa de parâmetros, e o número de categorias utilizadas na discretização todos afetam a precisão dos resultados e não há diretrizes claras sobre qual seria a melhor escolha para se empregar. Pode simplesmente ser dependente do conjunto de dados sendo utilizado, a quantidade de dados disponíveis, e tentativa e erro para encontrar a melhor solução.

Portanto, dadas as limitações acima referidas, como parte de um projeto financiado pelo governo Neozelandês, vários modelos RB de estimativa de esforço de desenvolvimento de aplicações Web foram criados separadamente para as empresas participantes do projeto de pesquisa, a partir da elicitación do conhecimento dos especialistas de domínio de cada empresa. Todas as empresas participantes foram empresas locais na região de Auckland, Nova Zelândia. O desenvolvimento e a implantação bem-sucedida de um desses modelos RB é o tema e a contribuição deste trabalho. Nossa contribuição vai além da área de engenharia Web, uma vez que o processo aqui apresentado também pode ser usado para construir modelos RB para empresas não-Web. Note que não estamos sugerindo que os modelos RB orientados a dados e híbridos não devam ser utilizados. Pelo contrário, eles têm sido empregados com sucesso em vários domínios (Pollino et al. 2007), porém o contexto de domínio específico deste trabalho - o de estimativa de esforço de desenvolvimento de aplicações Web, proporciona outros desafios (descritos acima), que levam ao desenvolvimento de apenas modelos RB que sejam criados a partir do conhecimento de especialistas do domínio. Gostaríamos também de salientar que em nossa opinião o desenvolvimento de aplicações Web e o desenvolvimento de software convencional diferem em várias áreas, tais como: Características da Aplicação, principais tecnologias utilizadas, Abordagem de qualidade fornecida, Motivadores do Processo de Desenvolvimento, disponibilidade do aplicativo, os clientes participantes (Stakeholders), frequência de atualização (ciclos de manutenção), as pessoas envolvidas no processo de desenvolvimento, Arquitetura e Rede, disciplinas envolvidas, questões legais, sociais e de ética e Estruturação de Informação e Design. Uma discussão detalhada sobre esta questão é apresentada em (Mendes et al. 2005a).

O restante do artigo está organizado da seguinte forma: a Seção 2 descreve todo o processo utilizado para construir e validar RBs. A Seção 3 detalha esse processo, enfocando na criação e validação de um modelo RB de estimativa de esforço de aplicações Web real. Finalmente, as conclusões e comentários sobre os trabalhos futuros são apresentados na Seção 4.

2. Processo Geralmente Utilizado para Construir Redes Baesianas

O modelo RB apresentado neste trabalho foi construído e validado utilizando uma adaptação do processo de Engenharia de Conhecimento de Redes Baesianas (KEBN) proposto em (Woodberry et al. 2004) (ver Figura 2). Na Figura 2 setas representam os fluxos através dos diferentes processos, representados por retângulos. Tais processos são executados tanto por pessoas - o Engenheiro do Conhecimento (KE) e os especialistas do domínio (DEs) (retângulos brancos), ou por algoritmos automáticos (retângulos cinza escuro). No contexto deste trabalho a autora foi a KE, e um gerente de projeto da Web de uma empresa de Web bem estabelecido em Auckland foi o DE. Os três principais passos dentro do processo KEBN, aqui adaptados, são o desenvolvimento estrutural, estimativa de parâmetro e validação do modelo. O processo KEBN itera sobre essas etapas até que o modelo RB tenha sido completamente construído e validado. Cada uma das três etapas parte do processo KEBN é detalhada abaixo:

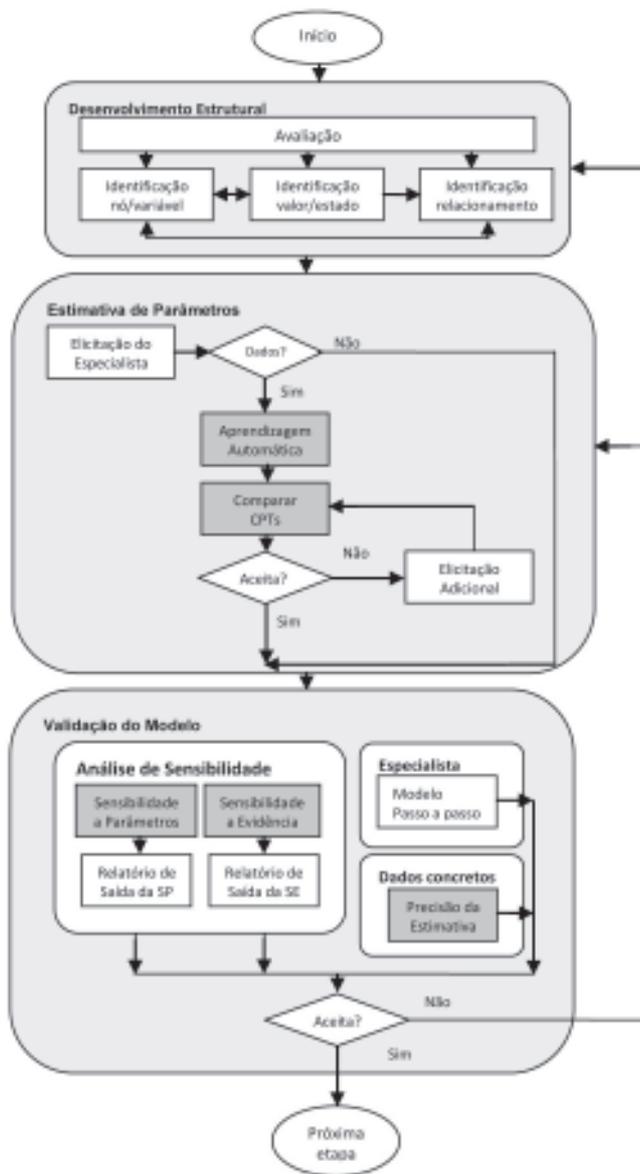


Figura 2. KEBN

Desenvolvimento Estrutural: Esta etapa representa o componente qualitativo da RB, o que resulta em uma estrutura gráfica composta de, no nosso caso, elementos (nós, as variáveis) e as relações causais identificadas como fundamentais para a estimativa de esforço de projetos Web. Além de identificar as variáveis e seus tipos (por exemplo, variável de consulta, a variável de evidência) e relações causais, esta etapa inclui também a identificação dos estados (valores) que cada variável deve ter, e se serão discretos ou contínuos. Na prática, as ferramentas atualmente disponíveis para criação de modelos RB exigem que as variáveis contínuas sejam discretizadas, convertendo-os em variáveis multinomiais (Jensen, 1996). Essa restrição também se aplicou ao modelo apresentado neste artigo. A estrutura do modelo

RB é refinada através de um processo iterativo. Este processo de construção da estrutura foi validado em estudos anteriores (e.g. Woodberry et al. 2004) e utiliza os princípios de resolução de problemas utilizados na modelagem de dados e desenvolvimento de software. Como será detalhado posteriormente, a literatura existente na área de estimativa de esforço de desenvolvimento de aplicações Web e conhecimento do especialista do domínio foram utilizados para elicitar a estrutura do modelo RB descrito neste artigo.

Durante todo este passo, o engenheiro de conhecimento também avaliou essa estrutura em dois estágios. O primeiro estágio foi usado para verificar se as variáveis selecionadas e seus valores tinham um significado claro, se todas as variáveis relevantes haviam sido incluídas, se as variáveis haviam sido nomeadas convenientemente, se todos os estados eram adequados (exaustiva e exclusiva), e a checagem da possibilidade de estados diversos serem combinados. O segundo estágio reviu a estrutura do modelo RB (estrutura causal) para garantir que todas as dependências identificadas estavam adequadas às suposições de causalidade pertinentes. Uma vez que a estrutura do modelo RB seja identificada pelos especialistas como praticamente pronta, é ainda necessário checar se essa estrutura precisa ser otimizada. A otimização visa reduzir o número de probabilidades que necessitam ser elicitadas. Se a otimização é necessária, técnicas que mudam a estrutura causal (por exemplo, o divórcio (Jensen 1996) e o uso de distribuições de probabilidades paramétricos (por exemplo, portões Noisy-OR (Jensen 1996)) são empregados.

Estimativa de Parâmetros: Esta etapa representa o componente quantitativo de um modelo RB, onde as probabilidades correspondentes à quantificação das relações entre as variáveis, são obtidas. Essas probabilidades podem ser elicitadas manualmente de especialistas do domínio, obtidas automaticamente utilizando um conjunto de dados, obtidas de literatura existente na área, ou utilizando a combinação destas. Quando as probabilidades são elicitadas a partir do zero, ou mesmo se elas só precisam ser revisitadas, este passo pode ser muito demorado. A fim de minimizar o número de probabilidades a serem elicitadas algumas técnicas têm sido propostas na literatura (e.g. Tang et al. 2007), no entanto, tanto quanto sabemos, não há nenhuma evidência empírica até o momento comparando a eficácia de diferentes técnicas para previsão em modelos RB complexos, em relação às probabilidades elicitadas a partir do zero. Este é um dos tópicos do nosso trabalho futuro.

Validação do Modelo RB: Esta etapa valida o modelo RB que resultou das duas etapas anteriores, e determina se é necessário re-visitar qualquer um desses passos anteriores. Dois diferentes métodos de validação são geralmente usados - Passo a passo e Precisão da Estimativa. O modelo Passo a Passo representa a utilização de cenários reais que estão preparados e utilizados por especialistas do domínio para verificar se as previsões fornecidas por um modelo RB correspondem às previsões que teriam sido feitas por esses especialistas do domínio, com base nas suas próprias experiências. Aqui o sucesso é medido pela frequência com que o valor previsto pelo modelo RB para a variável a ser estimada (por exemplo, qualidade, esforço) que apresenta a maior probabilidade corresponde ao esforço que seria sugerido pelos próprios peritos. A precisão da estimativa utiliza os dados de projetos anteriores, já terminados (por exemplo, os dados dos projetos passados), ao invés de cenários, para obter previsões. Dados correspondendo a projetos que já foram concluídos (evidências) são inseridos no modelo RB, e o sucesso é medido através da frequência com que o valor previsto pelo modelo RB para uma variável alvo (por exemplo, qualidade, esforço) contendo a maior probabilidade de ocorrência corresponde ao esforço real do projeto cujos dados foram utilizados como evidência.

3. Processo Utilizado para a Construção do Modelo RB

Esta seção revê o processo KEBN, descrito na seção anterior (ver figura 2), detalhando agora as tarefas realizadas em cada uma das três principais etapas desse processo, mas contextualizando-as com relação do modelo RB específico que é foco deste artigo. Antes de iniciar a elicitação do modelo de estimativa de esforço de desenvolvimento de aplicações Web foco deste artigo, o especialista de domínio (DE) que participou da elicitação foi introduzido ao conceito de modelos de Redes Bayesianas, ao processo de desenvolvimento desses modelos, e a exemplos de cenários "what-if" utilizado outros modelos RB. Acreditamos que essa introdução facilitou todo o processo de elicitação que ocorreu posteriormente, e também ajudou a fornecer uma compreensão concreta sobre o que esperar. Nós também enfatizamos que o engenheiro do conhecimento era apenas um facilitador do processo de desenvolvimento do modelo RB, e que o compromisso da empresa Web era fundamental para que o modelo a ser criado fosse adequado às necessidades da empresa. O processo inteiro de desenvolvimento do modelo RB durou 18 horas para ser concluído. Como duas pessoas participaram do processo (o engenheiro do conhecimento e o especialista do domínio) o esforço total de desenvolvimento do modelo correspondeu a 36 horas. A duração desse processo foi de 6 semanas, com reuniões semanais de 3 horas de duração. O especialista do domínio que participou no desenvolvimento do modelo RB é o gerente de projeto (e proprietário) de uma empresa Web, localizada em Auckland (Nova Zelândia), que é bem estabelecida na área de desenvolvimento de aplicações Web. Essa empresa possui um gerente de projetos, dois empregados que desenvolvem aplicações Web, e vários sub-contratados. Na época em que o modelo foi desenvolvido, o gerente de projetos já havia trabalhado no desenvolvimento de aplicações Web por mais de 10 anos, e a sua empresa desenvolvia uma ampla gama de aplicações Web, de aplicações estáticas e multimídia, até aplicações do tipo e-commerce.

Essa empresa também utilizava uma ampla variedade de tecnologias Web, permitindo o desenvolvimento de aplicações Web 2.0. Anteriormente ao uso do modelo RB desenvolvido, as estimativas de esforço fornecidas para os clientes variavam num intervalo de 10% a 40% do esforço atual desses projetos. Detalhamento do desenvolvimento estrutural e Estimação de Parâmetros: A fim de identificar os fatores fundamentais que o especialista do domínio levava em conta na preparação de uma estimativa de esforço de desenvolvimento de um novo projeto, utilizamos como ponto de partida o conjunto de variáveis do conjunto de dados (ver Tabela 2). Primeiro, estas variáveis foram transcritas para um quadro branco, cada uma representada como uma figura oval. Em seguida, explicou-se o que cada uma significava no contexto do projeto Tukutuku. O uso de um conjunto inicial de variáveis foi utilizado como ponto de partida para facilitar o processo de elicitação, e para evitar que o especialista se bloqueasse cognitivamente ao se deparar com uma "tela em branco". No âmbito do projeto Tukutuku, uma nova função/característica de esforço alto exige, pelo menos, 15 horas para ser desenvolvido por um programador experiente; uma função/característica de esforço alto a ser reusada e adaptada exige pelo menos, quatro horas para ser adaptado por um desenvolvedor experiente.

Estes valores são baseados nos dados coletados. Uma vez que as variáveis Tukutuku haviam sido delineadas e explicadas, o próximo passo do processo de elicitação consistiu em remover todas as variáveis Tukutuku que não eram consideradas relevantes para o especialista do domínio, seguida pela adição ao quadro branco de quaisquer outras variáveis (fatores) identificadas pelo especialista como relevantes para estimar esforço de desenvolvimento de aplicações Web. Durante essa etapa, eram também documentadas as descrições de todas as variáveis identificadas.

	Variável	Descrição
Dados Projeto	<i>TypeProj</i>	Tipo de projeto (novo ou modificação).
	<i>nLang</i>	Número de linguagens de desenvolvimento utilizadas
	<i>DocProc</i>	Se o processo de desenvolvimento foi bem definido e documentado.
	<i>ProImpr</i>	Se a equipe participava de num programa de melhoria de processos.
	<i>Metrics</i>	Se a equipe de projeto é parte de um programa de métricas.
	<i>DevTeam</i>	Tamanho da equipe de projeto.
	<i>TeamExp</i>	Média de anos de experiência da equipe com as linguagens utilizadas.
Aplicação Web	<i>TotWP</i>	Número total de páginas Web (novas e reusadas).
	<i>NewWP</i>	Número total de novas páginas Web.
	<i>TotImg</i>	Número total de imagens (novas e reusadas).
	<i>NewImg</i>	Número total de novas imagens criadas.
	<i>Num_Fots</i>	Número de funções/características reusadas sem adaptação.
	<i>HFotsA</i>	Número de funções/características de esforço alto reusadas e adaptadas.
	<i>Hnew</i>	Número de novas funções/características de esforço alto.
	<i>TotHigh</i>	Número total de funções/características de esforço alto.
	<i>Num_FotsA</i>	Número de funções/características de esforço baixo reusadas e adaptadas.
	<i>New</i>	Número de novas funções/ características de esforço baixo.
	<i>TotNHigh</i>	Número total de funções/características de esforço baixo.

Tabela 1. Variáveis Tukutuku

Em seguida, foram identificados os estados a serem utilizados para medir cada fator (por exemplo, baixo, médio e alto; sim e não). Todos os estados tiveram que ser discretos em função da ferramenta utilizada para criar e compilar o modelo RB. Sempre que um fator representava uma medida de esforço (por exemplo, o esforço total), também era documentado o intervalo de esforço correspondente a cada estado, para evitar qualquer ambiguidade futura. Por exemplo, o estado “muito baixo” associado à variável esforço total poderia corresponder a um intervalo de 0+ a 8 horas-pessoa.

Depois que todos os estados serem identificados e documentados, a próxima etapa consistiu em obter as relações de causa e efeito entre as variáveis previamente identificadas. Como ponto de partida para esta tarefa foi utilizado um exemplo médico simples (Jensen 1996), mostrado na Figura 3.

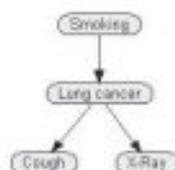


Figura 3. Exemplo médico, adaptado de Jensen (1996)

Este exemplo apresenta claramente um dos pontos importantes a considerar ao identificar relações de causa e efeito – a cronologia de eventos. Se o fumo é uma causa de câncer de pulmão, é importante que a causa preceda o efeito. Isto pode parecer óbvio em relação ao exemplo utilizado, no entanto, é

nossa opinião que a utilização deste exemplo simples ajudou significativamente o especialista de domínio a compreender a noção de causa e efeito, e como este conceito deveria ser aplicado durante a construção do modelo RB para a estimativa de esforço de desenvolvimento de aplicações Web. Uma vez que as relações de causa e efeito foram identificados a primeira versão da estrutura causal do modelo RB foi a seguinte (ver Figura 4). Note que a Figura 4 não é um modelo RB que só inclui variáveis da Tabela 2. A Figura 4 mostra que muitas relações de causa & efeito se destinavam às variáveis 'Total Effort (Esforço Total)' e 'Overall Complexity (Complexidade Geral)', o que significou a necessidade de que o número de relações atingindo cada uma dessas duas variáveis fosse reduzida, de forma a simplificar essa estrutura e reduzir o número de probabilidades a serem elicitadas posteriormente. Como consequência, novas variáveis foram sugeridas pelo engenheiro do conhecimento (nomes terminados em "_N", veja Figura 5), e checados pelo especialista do domínio. Esse especialista também sugeriu algumas mudanças em outras relações, o que transformou a estrutura causal original do modelo RB na estrutura apresentada na Figura 5.

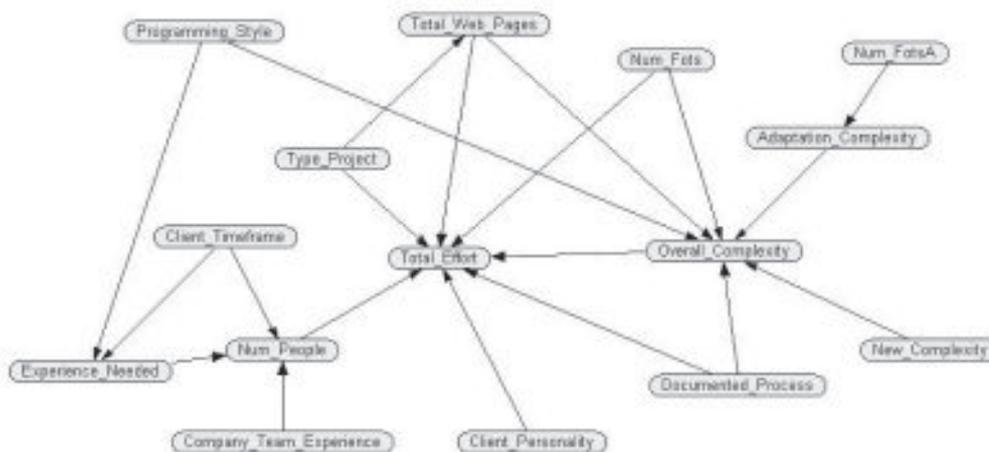


Figura 4. Primeira Estrutura Causal do Modelo RB

Neste momento o especialista do domínio estava satisfeito com a estrutura causal da RB e a fase relativa à eliciação das probabilidades foi iniciada. Todas as probabilidades foram criados a partir do zero, uma tarefa que consumiu de 8 a 10 horas. Durante esse processo, o especialista de domínio resolveu re-visitara estrutura causal da RB, após rever o seu processo de estimativa de esforço. Portanto, uma nova iteração da etapa de desenvolvimento estrutural ocorreu. A estrutura causal do modelo RB final é mostrada na Figura 6. Aqui nós apresentamos a estrutura causal usando barras de crença ao invés de fatores rotulados, para que os leitores possam ver as probabilidades de foram elicitadas. Note que esta estrutura causal corresponde à estrutura final e atual que está sendo utilizada pela empresa Web.

Validação Detalhada do Modelo: Tanto o uso de cenários quanto o da precisão de estimativa foram utilizados para validar o modelo RB de esforço. Os cenários foram utilizados primeiramente, seguidos da avaliação da precisão da estimativa. O especialista do domínio utilizou quatro cenários diferentes para verificar se a maior probabilidade associada a uma estimativa de esforço para a variável Total_effort corresponderia a estimativa de esforço sugerida pelo próprio especialista para esta mesma variável. Todos os cenários foram executados com sucesso, mas também foi necessário o uso de dados de projetos anteriores finalizados, para os quais o esforço total era conhecido, a fim de verificar a calibração do modelo.

Um conjunto de validação, contendo dados sobre oito projetos já terminados foi utilizado. O especialista selecionou um conjunto de projetos com diferentes tamanhos e níveis de complexidade, para que todos os projetos utilizados na validação do modelo fossem representativos dos tipos de projetos desenvolvidos pela empresa Web: quatro eram projetos pequenos, três eram de tamanho médio e um era largo.

Para cada projeto utilizado para validação, os seus dados foram inseridos como evidência no modelo RB (de forma similar ao que é apresentado na Figura 6.b), e os valores referentes ao intervalo de esforço correspondente à maior probabilidade fornecida para "Total effort" são comparados ao valor do esforço real do projeto. Em outras palavras, a validação envolve checar se o valor do esforço atual do projeto se insere no intervalo de esforço parte da variável "Total effort" que apresenta a maior probabilidade. Por exemplo, na figura 6.b, isto corresponderia a "Total effort" = Medium. A empresa também definiu o intervalo de valores de esforço associados a cada uma das categorias utilizadas para medir "Total effort". No caso da empresa descrita aqui, o esforço médio correspondeu de 25 a 50 horas-pessoa.

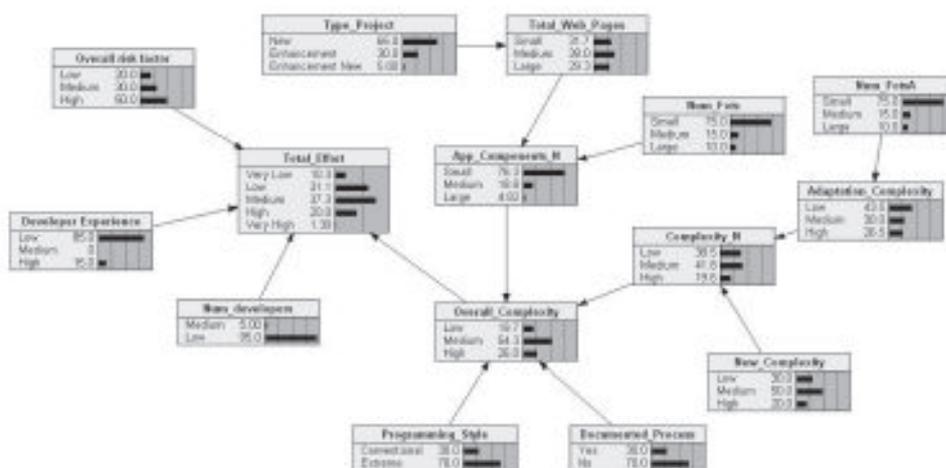


Figura 5. Versão final do modelo RB

Sempre que o esforço real não caísse dentro da faixa de esforço associado com a categoria com maior probabilidade sugerida pelo modelo RB, havia um descompasso, o que significava que algumas probabilidades precisavam ser ajustadas. Para saber quais as variáveis que deveriam ser ajustadas primeiramente, utilizou-se um relatório de análise de sensibilidade, que previa o efeito de cada variável pai sobre uma determinada variável de consulta. Dentro do nosso contexto, a variável de consulta era "Total effort". Sempre que as probabilidades eram re-ajustadas, os dados dos projetos sendo utilizados para validação do modelo eram re-inseridos como evidência no modelo RB para garantir que a calibração já realizada não tinha sido afetada. Isto foi feito para garantir que cada calibragem seria sempre uma melhoria sobre a anterior. Depois que todos os oito projetos foram usados para calibrar o modelo RB a etapa de validação foi concluída. A Figura 6 mostra dois cenários de uso do modelo RB.



Figura 6. Cenários de Diagnóstico e Predição utilizando o modelo RB

O primeiro (Figura 6 (a)) mostra as probabilidades de todos os fatores (variáveis) prováveis na RB quando a variável "Total_effort" = very high (nó cinza na Figura 6). Por outro lado, o outro cenário mostra as probabilidades prováveis para "Total_effort" quando alguma evidência é inserida em algumas variáveis (nós cinza). Este modelo RB tem estado em produção por pelo menos dois anos e tem sido utilizado com sucesso para estimar o esforço dos novos projetos desenvolvidos pela empresa. Isso significa que a empresa obtém estimativas de esforço providas pelo modelo, ao invés de depender de estimativas subjetivas fornecidas pelo especialista do domínio, como ocorria anteriormente. Acreditamos que o êxito do desenvolvimento deste modelo RB foi fortemente influenciado pelo comprometimento da empresa, e também pelo bom conhecimento e da experiência do especialista de domínio com relação a estimativas de esforço de desenvolvimento de aplicações Web.

4. Conclusões

Este artigo apresentou um estudo de caso que descreveu a construção de um modelo de redes baesianas para estimar o esforço de desenvolvimento de aplicações Web, utilizando como base o conhecimento de um especialista do domínio que trabalha numa empresa Web bem estabelecida. Este modelo foi desenvolvido utilizando a engenharia de conhecimento para processo de redes baesianas (ver figura 2). Sua estrutura causal passou por duas iterações, representando a evolução do trabalho em função do amadurecimento do especialista com relação a um melhor entendimento do processo utilizado para estimar o esforço de projetos, e dos fatores relevantes. Cada sessão com o especialista de domínio não durou mais de três horas. O modelo RB final foi calibrado com dados sobre oito projetos anteriores. Esses projetos representaram projetos típicos desenvolvidos pela empresa, e julgados como adequados e suficientes para a calibração do modelo. Desde a adoção do modelo, este tem sido usado com sucesso para fornecer estimativas de esforço para os projetos Web desenvolvidos pela empresa. Aqui as estimativas de esforço tem sido obtidas utilizando apenas esse modelo. Todo o processo utilizado para construir e validar o modelo RB durou em torno de 18 horas-pessoa, onde a maior quantidade de tempo foi gasta elicitando as probabilidades. Esta é uma questão para aqueles que constroem modelos desse tipo, e é atualmente o foco de nosso trabalho futuro. O processo de elicitação permite que os especialistas pensem profundamente sobre o seu processo de estimativa de esforço e os fatores considerados durante esse processo, o que em si já é vantajoso para uma empresa. Essa vantagem tem sido citada não só pelo especialista de domínio cujo modelo foi aqui apresentado, mas também por outras empresas com as quais trabalhamos na elicitação de outros modelos.

Referências

- Jensen, F. V. (1996) *An introduction to Bayesian networks*. UCL Press, London.
- Mendes, E., (2007a). E. Mendes, Predicting Web Development Effort Using a Bayesian Network, Proceedings of EASE'07, pp. 83-93.
- Mendes, E. (2007b), The Use of a Bayesian Network for Web Effort Estimation, Proceedings of International Conference on Web Engineering, pp. 90-104, LNCS 4607.
- Mendes, E. (2007c), A Comparison of Techniques for Web Effort Estimation, Proceedings of the ACM/IEEE International Symposium on Empirical Software Engineering, pp. 334-343.
- Mendes, E., and N. Mosley (2008), Bayesian Network Models for Web Effort Prediction: a Comparative Study, Transactions on Software Engineering, Vol. 34, Issue: 6, Nov/Dec 2008, pp. 723-737.
- Mendes, E., Mosley, N. and Counsell, S. (2005a) The Need for Web Engineering: An Introduction, *Web Engineering*, Springer-Verlag, Mendes, E. and Mosley, N. (Eds.) ISBN: 3-540-28196-7, pp. 1-28.
- Mendes, E., N. Mosley, and S. Counsell (2005b). *Investigating Web Size Metrics for Early Web Cost Estimation*, Journal of Systems and Software, 77(2), 157-172.
- Pollino, C., A. White, and B.T. Hart (2007). Development and application of a Bayesian decision support tool to assist in the management of an endangered species. *Ecological Modelling* 201, 37-59.
- Tang, Z., and B. McCabe (2007). Developing Complete Conditional Probability Tables from Fractional Data for Bayesian Belief Networks, *Journal of Computing in Civil Engineering*, 21(4), pp. 265-276.
- Woodberry, O., A. Nicholson, K. Korb, and C. Pollino (2004). Parameterising Bayesian Networks, *Proc. Australian Conference on Artificial Intelligence*, pp. 1101-1107.

Implementando o Nível F do MR-MPS com Práticas da Metodologia Ágil Scrum

Edmar Catunda¹, Camila Nascimento¹, Cristina Cerdeiral², Gleison Santos^{3,2}, Ana Regina Rocha²

¹Rightway Consultoria & Sistemas – Rua do Rosário 103, 12º andar – Centro CEP 20041-004 – Rio de Janeiro, Brasil

²COPPE/UFRJ – Universidade Federal do Rio de Janeiro Caixa Postal 68511 – CEP 21945-970 – Rio de Janeiro, Brasil

³Programa de Pós-Graduação em Informática – Universidade Federal do Estado do Rio de Janeiro (UNIRIO) - Caixa Postal 56011 – CEP 22290-970 – Rio de Janeiro, Brasil
{edmar.catunda, camila.nascimento}@rightway.com.br,
{cerdeiral, darocha}@cos.ufrj.br, gleison.santos@uniriotec.br

Abstract. *Rightway Consultoria & Sistemas is specialized in IT solutions and projects development. This paper describes the methodology used in its software process improvement initiative to achieve the MR-MPS level F with the support of Scrum's agile practices, that were already used by the company. This paper also presents the standard development process phases adapted to Scrum's agile practices and the case tools used to support the execution of this process.*

Resumo. *A Rightway Consultoria & Sistemas é uma empresa especializada em desenvolvimento de soluções e projetos de Tecnologia da Informação. Este artigo descreve a metodologia utilizada na iniciativa de melhoria de processos de software para alcançar o nível F do MR-MPS em conjunto com práticas ágeis da metodologia Scrum, já utilizadas pela empresa. O artigo apresenta ainda as fases do processo padrão de desenvolvimento adaptadas às práticas da metodologia Scrum e as ferramentas utilizadas para apoiar a execução deste processo.*

1. Introdução

A Rightway Consultoria & Sistemas é uma empresa com sede no Rio de Janeiro, especializada em desenvolvimento de soluções e projetos de Tecnologia da Informação que atua em quatro linhas de serviços: sistemas sob medida, infraestrutura, business intelligence e alocação de pessoal. A empresa entrou no grupo coordenado pela Riosoft no modelo cooperado para a implementação do nível F do MR-MPS [Softex 2009] com o objetivo de assegurar aos clientes a prestação de serviços com alto nível de qualidade [Sistemas 2010]. A empresa já utilizava algumas práticas ágeis da metodologia Scrum [ScrumAlliance 2010] na execução de seus projetos, como as reuniões diárias para monitoração do projeto (*Daily Scrum*), o *dashboard* com a situação das atividades, o ciclo e vida incremental e curto, com *sprints* semanais, as homologações por *sprint* sempre que possível e o envolvimento do cliente sempre que possível em reuniões iniciais e finais do *sprint*.

A empresa desejava implementar processos para organizar melhor o desenvolvimento de software, permitindo um maior controle e alcançando uma melhor qualidade nos seus produtos; porém, sem perder a agilidade nos seus projetos. Este artigo explica a metodologia utilizada para implantar as melhorias de processos necessárias para alcançar o nível F do modelo (seção 2), as fases do processo padrão de desenvolvimento com as práticas da metodologia Scrum [ScrumAlliance 2010] adotadas (seção 3) e as ferramentas utilizadas para apoiar a execução do processo (seção 4). As considerações finais são apresentadas na seção 5. Os autores esperam que estas informações possam auxiliar outras

empresas na combinação de práticas ágeis e modelos de maturidade e capacidade em seus processos de desenvolvimento e na seleção de ferramentas de apoio para a execução de seus processos.

2. Metodologia de Implementação

A Rightway Consultoria & Sistemas, juntamente com a consultoria da COPPE/UFRJ, iniciou a implementação das melhorias de processo necessárias para alcançar o nível F do MR-MPS [Softex 2009], seguindo a estratégia SPI-KM [Santos *et al.* 2007a; Santos *et al.* 2007b], na qual a gerência de conhecimento é utilizada para auxiliar a implementação de melhorias de processo de software. As fases da estratégia utilizada podem ser observadas na Figura 1 e são descritas a seguir.



Figura 1. Fases da Estratégia de Implementação

i. Análise das práticas, processos e procedimentos da empresa: As práticas, os processos e procedimentos já utilizados pela empresa foram analisados, com o objetivo de identificar e entender as necessidades da empresa, pertinzendo a elaboração de um processo padrão que atendesse a estas necessidades. Neste momento, a empresa colocou a necessidade de manter a agilidade de seus processos adequada ao tamanho dos projetos e ao ciclo de vida utilizado.

ii. Primeira Versão do Processo Padrão: Foi elaborada a primeira versão do processo padrão de desenvolvimento buscando manter as práticas ágeis já utilizadas e atender às necessidades identificadas na fase anterior. O objetivo desta primeira versão é permitir que a empresa utilize o processo rapidamente em um projeto piloto e aprenda com esta experiência, evoluindo o processo e adequando-o à realidade da empresa.

iii. Projeto Piloto: O processo padrão foi utilizado em um primeiro projeto com o apoio das ferramentas do ambiente TABA [Ferreira *et al.* 2006; Montoni *et al.* 2006; Montoni *et al.* 2007]. Através desta execução do processo, a empresa identificou várias oportunidades de melhoria no processo padrão.

iv. Capacitação Scrum Master: Em paralelo à execução do projeto piloto, a gerência da Fábrica e todos os coordenadores de projetos da empresa fizeram o curso e se tornaram Certified Scrum Masters, aumentando o conhecimento da empresa nesta metodologia ágil.

v. Segunda Versão do Processo Padrão: As oportunidades de melhoria identificadas na execução do projeto piloto e a capacitação Certified Scrum Masters geraram novas sugestões de alteração no processo padrão. Foi elaborada, então, uma segunda versão do processo padrão, mais voltada para as práticas do Scrum e mais adaptada à realidade da empresa.

vi. *Demais Projetos*: O processo padrão revisado foi utilizado nos demais projetos da empresa, com o apoio de ferramentas que tornassem mais ágil a execução do processo e automatizassem algumas atividades. Atualmente, a empresa possui os processos organizacionais já definidos e em execução, e está executando o terceiro projeto seguindo a segunda versão do processo padrão, estando em estágio avançado no projeto de implementação das melhorias de processo necessárias para alcançar o nível F do MR-MPS [Softex 2009]. A avaliação da empresa está agendada para novembro.

3. Processo e Práticas do Scrum

A segunda versão do processo padrão de desenvolvimento da Rightway Consultoria & Sistemas possui cinco fases, que podem ser observadas na figura 2. As três primeiras fases são obrigatórias, pois ocorrem em todos os projetos da empresa; as duas últimas fases são opcionais, pois nem sempre se encontram no escopo dos projetos.



Figura 2. Processo Padrão de Desenvolvimento

3.1. Fase 0 – Pré-Projeto

A Fase 0 contempla as atividades que são executadas antes de o projeto iniciar, para a elaboração da Proposta Técnica. Nesta fase são realizadas as atividades do processo Gerência de Portfólio de Projetos relacionadas com a identificação de uma nova oportunidade, o entendimento inicial dos requisitos, as estimativas iniciais para o projeto utilizando um método próprio da empresa baseado na complexidade das funcionalidades e nas tecnologias utilizadas, a análise inicial do projeto e a alocação inicial de recursos humanos para o projeto (caso seja de interesse da empresa realizá-lo). Nesta fase não são utilizadas práticas da metodologia Scrum.

3.2. Fase 1 – Especificação e Planejamento do Projeto

A Fase 1 contempla o planejamento e a especificação técnica do projeto e pode ser observada na Figura 3. Inicialmente é realizado um planejamento para a Fase 1, seguido do levantamento e análise dos requisitos, englobando a identificação dos requisitos de cliente (já parcialmente identificados na Fase 0), dos requisitos funcionais e não funcionais e dos casos de uso. Com base no levantamento e análise dos requisitos, as estimativas são refinadas para o projeto e é realizado um planejamento macro para todo o projeto, que consiste da decisão do número de *sprints* necessários para o desenvolvimento do projeto e de uma alocação prévia dos casos de uso nos *sprints* (*Release Planning*).

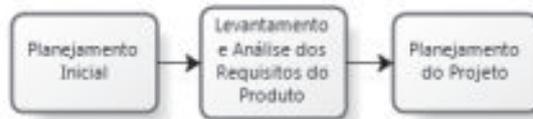


Figura 3. Fase 1 do Processo Padrão de Desenvolvimento

As práticas da metodologia Scrum utilizadas nesta fase são o *Release Planning* (na qual a estimativa do número de *sprints* necessários e uma alocação prévia das estórias nos *sprints* gerando uma estimativa inicial de fim do projeto são realizadas) e a divisão do projeto em *sprints* de duração fixa, usualmente duas semanas. As adaptações realizadas pela empresa nas práticas da metodologia Scrum são a utilização de casos de uso no lugar das estórias, mantendo a compatibilidade com o método de estimativas criado pela empresa. A velocidade da equipe não é utilizada como forma de estimar quais casos de uso serão desenvolvidos em cada *sprint*, e sim um método de estimativa próprio de complexidade por caso de uso e tecnologias utilizadas no projeto e a base histórica contendo as datas reais de início e fim e esforço para casos de uso de tamanhos e tecnologias similares.

3.3. Fase 2 – Análise, Projeto, Construção, Homologação e Implantação (*Sprint*)

A Fase 2 representa um *sprint* e tem a duração fixa de uma ou duas semanas, sendo esta última a mais usual. Nesta fase são realizadas as atividades relacionadas com a análise, projeto, construção, homologação e implantação de alguns casos de uso do projeto, como pode ser observado na Figura 4. Pode ser definido mais de um *sprint* no projeto, se necessário, o que torna esta fase iterativa.



Figura 4. Fase 2 do Processo Padrão de Desenvolvimento

O *sprint* é iniciado com o seu planejamento, no qual são utilizadas as práticas da metodologia Scrum *Sprint Planning 1* (na qual as estórias a serem desenvolvidas no *sprint* atual são selecionadas) e *Sprint Planning 2* (na qual as estórias selecionadas para o *sprint* são detalhadas e analisadas), sempre que possível com a participação do cliente, definindo, detalhando e analisando os casos de uso a serem desenvolvidos no *sprint*. Cada caso de uso é, então, trabalhado por toda a equipe do projeto, desde a sua análise até sua homologação interna, na ordem de maior valor para o cliente. Apenas após a equipe concluir o primeiro caso de uso do *sprint*, o segundo caso de uso começa a ser trabalhado. Desta forma, a fase 2 é uma iteração no ciclo de vida do projeto e dentro dela ocorrem iterações de desenvolvimento para cada caso de uso.

Durante todo o *sprint* são utilizadas as práticas da metodologia Scrum *dashboard*, quadro contendo a situação dos casos de uso do *sprint*, *burndown chart*, gráfico contendo o andamento do desenvolvimento dos casos de uso e as reuniões *Daily Sprint* para monitoramento do projeto com a

participação do coordenador do projeto e da equipe do projeto. São realizadas reuniões semanais do coordenador do projeto com a gerência da Fábrica nas quais o andamento do projeto naquela semana é apresentado, juntamente com os impedimentos ainda não solucionados e são definidas ações para tratá-los. É gerada uma ata desta reunião com as informações de monitoração do projeto. O *sprint* se encerra com as reuniões *Sprint Review*, na qual as funcionalidades desenvolvidas são apresentadas para o cliente, homologadas e implantadas e *Sprint Retrospective*, onde são recolhidas lições aprendidas. Em alguns projetos, o cliente não apresenta interesse em receber versões intermediárias do produto e solicita receber apenas uma versão no final do desenvolvimento. Neste caso, as atividades de homologação e implantação são removidas dos *sprints* intermediários e são mantidas apenas no último *sprint* do projeto. As reuniões *Sprint Review* e *Sprint Retrospective*, no caso de o cliente não poder participar, ocorrem apenas com a participação da equipe do projeto, do coordenador do projeto e da gerência da Fábrica.

3.4. Fase 3 – Treinamento

Esta fase contempla o planejamento e execução dos treinamentos necessários para o cliente sobre uso e operação do produto. Esta fase não é obrigatória por não se encontrar no escopo de todos os projetos da empresa. Nesta fase não são utilizadas práticas da metodologia Scrum.

3.5. Fase 4 – Produção Assistida

Esta fase contempla o planejamento e execução do período de produção assistida, no qual a empresa tira dúvidas sobre a utilização do produto e corrige problemas encontrados pelo cliente. Esta fase não é obrigatória por não se encontrar no escopo de todos os projetos da empresa. Caso sejam identificados problemas a serem corrigidos, novos *sprints* são planejados para o projeto, utilizando as mesmas atividades descritas na Fase 2.

3.6. Atividades de Apoio

As atividades com práticas ágeis são as relacionadas com o processo Gerência de Projetos e Gerência de Requisitos. Não foram selecionadas medidas baseadas nas práticas do Scrum, permanecendo medidas como esforço e datas reais de início e fim da cada atividade. As avaliações de Garantia da Qualidade se encontram distribuídas nas fases do processo padrão e são realizadas ao término da elaboração dos documentos a serem avaliados. As não conformidades identificadas são explicadas ao responsável pela sua correção ao término da avaliação e possuem um prazo de apenas 2 dias, devido ao tamanho dos *sprints*. As baselines são geradas ao final das fases e são auditadas ainda no final da fase, antes da entrega para o cliente.

4. Apoio Ferramental

Para auxiliar a execução da primeira versão do processo padrão no projeto piloto foram utilizadas as ferramentas do ambiente TABA [Ferreira *et al.* 2006; Montoni *et al.* 2006; Montoni *et al.* 2007], que possuem conhecimento embutido sobre Engenharia de Software. A empresa, no entanto, identificou a necessidade de ferramentas mais voltadas para a execução de projetos com práticas ágeis.

A empresa analisou algumas ferramentas do mercado e selecionou as ferramentas abaixo:

- *Microsoft Project [Microsoft 2010]* – Ferramenta utilizada para definir e monitorar o cronograma do projeto. Um template do cronograma foi elaborado contendo todas as atividades do processo padrão de desenvolvimento, com a sequência e a dependência entre elas e os papéis responsáveis pela execução de cada atividade, além das atividades consideradas marcos do projeto. Desta forma, o coordenador do projeto só precisa refletir as alterações (adições e remoções de atividades) realizadas na adaptação do processo padrão para o projeto e estimar as datas de início, fim e esforço para cada atividade.
- *Microsoft Word e Excel [Microsoft 2010]* – Ferramentas utilizadas para elaborar vários templates utilizados na execução do processo padrão de desenvolvimento. A abordagem adotada em todos os documentos foi a de não repetir informações. Sendo assim, todas as informações e conhecimento que se repetiam nos projetos foram definidos em documentos organizacionais que são citados nos documentos específicos dos projetos. Desta forma, a elaboração dos documentos ficou mais fácil e rápida, pois é necessário preencher apenas as seções que podem ser adaptadas ou são específicas dos projetos. Além disso, os templates foram elaborados com o máximo de informações já preenchidas com os dados mais comuns, auxiliando e agilizando o preenchimento.
- *Subversion (SVN) [Tigris.org 2010a] e TortoiseSVN [Tigris.org 2010b]* – A ferramenta SVN foi utilizada para o versionamento e controle de configuração dos documentos e código fonte. Os mecanismos de *branch* e *tags* foram utilizados para implementar o processo Gerência de Configuração, gerando *baselines* e controlando as alterações nos itens de configuração. A ferramenta TortoiseSVN foi utilizada como interface de acesso aos repositórios do SVN. Ambas as ferramentas são *open source* e gratuitas.
- *Enterprise Architect [Sparxsystems 2010]* – Ferramenta utilizada para a análise e projeto dos casos de uso e para a elaboração e manutenção da rastreabilidade entre os componentes do projeto.
- *Pentaho [Pentaho 2010]* – Ferramenta de *Business Intelligence* utilizada para definir uma base de medição para a empresa. Atualmente as medidas estão sendo armazenadas em uma planilha Excel, mas serão migradas para esta base, facilitando e automatizando ainda mais a execução do processo Medição e Análise. Esta ferramenta é *open source* e gratuita.
- *Redmine [Redmine 2010]* – Ferramenta utilizada para o planejamento e controle das atividades do projeto. Permite uma maior visibilidade do andamento do projeto e a automatização de algumas atividades, além de possuir integração com o SVN. Esta ferramenta será detalhada a seguir.

4.1. Redmine

A Rightway Consultoria & Sistemas utiliza o Redmine para apoiar a gerência dos projetos de forma abrangente. Antes desta ferramenta, a empresa chegou a utilizar a ferramenta Trac [Edgewall 2010]

para a mesma funcionalidade, porém ela exige a definição de um novo ambiente para cada projeto. O Redmine possui um número grande de *plugins* disponíveis, o que possibilita agregar funcionalidades para um maior controle do projeto.

Após a definição do cronograma do projeto no Project, um *plugin* do Redmine é utilizado para gerar uma tarefa no Redmine correspondente para cada atividade do cronograma, já importando as datas de início e fim e o esforço estimado para cada atividade. A equipe do projeto reporta as datas de início e fim reais através das mudanças realizadas do estado das tarefas no Redmine. E-mails são enviados para os interessados em cada mudança de estado das tarefas do projeto, informando o andamento do projeto.

O Redmine possui funcionalidades específicas para projetos com práticas de metodologias ágeis, como o Scrum. Ele permite visualizar o *dashboard* e o *burndown chart* do projeto, com o andamento das tarefas do projeto na sua visão "Painel de Controle", como pode ser observado na Figura 5.

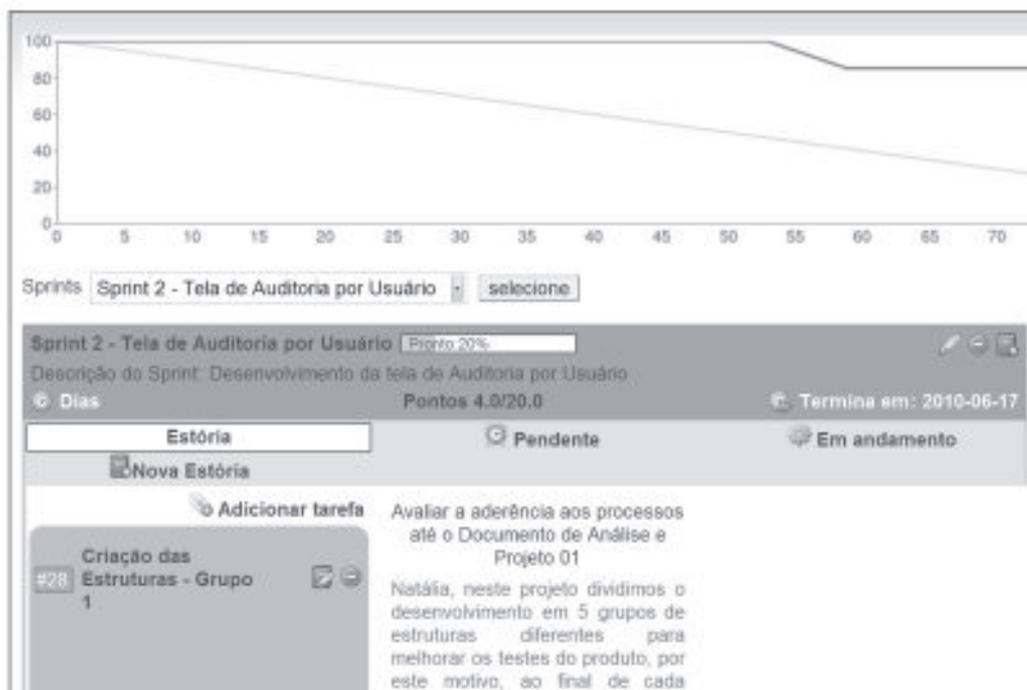


Figura 5. Visão do Painel de Controle do Redmine

A empresa definiu os tipos específicos "Avaliação GQPP", "Avaliação Técnica", "Alteração de Requisito", "Análise de Impacto" e "Solicitação de Modificação", como pode ser observado na Figura 6, com estados e mudanças de estado específicos, para apoiar: (i) as avaliações de conformidade e aderência realizadas pelo grupo de garantia da qualidade do processo e do produto (GQPP), relatando as não conformidades encontradas e criando e monitorando os planos de ação, com o tipo padrão do Redmine "Bug", (ii) as avaliações técnicas realizadas, relatando as não conformidades encontradas e criando e monitorando os planos de ação, com o tipo padrão do Redmine "Bug", (iii) as alterações de requisitos, relatando todas as informações necessárias no tipo "Alteração Requisito" e associando a

cada alteração de requisito um tipo “Análise de Impacto” relatando os resultados da análise de impacto da alteração solicitada, e (iv) as alterações nos itens de configuração, através do tipo “Solicitação de Modificação”, contemplando o pedido de alteração, a análise e avaliação do pedido, a aprovação do pedido, a implementação do pedido, a avaliação da alteração realizada e sua aprovação.

Planejamento

Sprint 1 - Tela de Auditoria por Estrutura

aproximadamente 2 meses atrasado (31/05/2010)

Construção da tela de Auditoria por Estrutura

98%

265 fechadas (100%) 1 aberta (0%)

Tarefas relacionadas

- Tarefa #263: Realizar Reunião de Planejamento do Sprint 1
- Tarefa #264: Criação da Ata de Reunião de Levantamento de Requisitos
- Tarefa #265: Elaborar diagrama de Classes
- Tarefa #266: Elaborar diagrama de Caso de Uso
- Tarefa #267: Elaborar IGF - Documento de Interface Gráfica
- Tarefa #268: Planejar testes do produto
- Tarefa #269: Planejar testes do produto - 1º grupo de 5 telas
- Tarefa #270: Planejar testes do produto - 2º grupo de 5 telas
- Tarefa #271: Planejar testes do produto - 3º grupo de 5 telas
- Tarefa #272: Planejar testes do produto - 4º grupo de 5 telas
- Tarefa #273: Planejar testes do produto - 5º grupo de 2 telas
- Tarefa #274: Elaborar Documento de Requisitos
- Tarefa #275: Elaborar Listagem de Documentos Criados/Alterados

Planejamento

- Bug
- Suporte
- Tarefa
- Avaliação GQPP
- Avaliação Técnica
- Alteração de Requisito
- Análise de Impacto

Exibir versões completas

Aplicar

Versões

Sprint 1 - Tela de Auditoria por Estrutura

Sprint 2 - Tela de Auditoria por Usuário

Sprint 3 - Manutenção

Figura 6. Visão de Planejamento do Redmine com os Tipos Específicos

O Redmine possui uma integração com o Subversion (SVN) que exige que cada *commit* no repositório do projeto seja associado a alguma tarefa definida no Redmine. Também é possível alterar o estado de uma tarefa (por exemplo, passar para o estado de concluída) através de comentários específicos no *commit*. Os relatórios de *baseline* são gerados a partir da visão do repositório SVN do Redmine, que pode ser observada na Figura 7.

root

Estadísticas | Revisão: []

Nome	Tamanho - Revisão	Idade	Autor	Comentário
branches	381	aproximadamente 1 mês	camila.nascimento	
tags	495	aproximadamente 1 mês	rodrigo.figueiredo	Criação da BaseLine da Fase 02 Sprint 02
trunk	656	1 dia	gustavo.valle	atualização das estruturas log_execcao, log_por...

Últimas revisões

#	Data	Autor	Comentário
656	06/06/2010 18:46	gustavo.valle	atualização das estruturas log_execcao, log_por... e log_tab_referencia_item_valor - 3 telas
655	06/06/2010 18:38	thiago.marques	alteração das telas de perfil var
654	06/06/2010 18:38	rodrigo.henriques	Ajuste nos campos do tipo usuário da estrutura log_negociacao
653	06/06/2010 18:14	camila.nascimento	close #1486
652	06/06/2010 17:31	rodrigo.henriques	Ajuste dos campos do tipo usuário na estrutura log_custo_operacional
651	06/06/2010 17:12	gustavo.valle	atualização das janelas de estrutura e usuario
650	06/06/2010 17:07	rodrigo.henriques	Alteração dos campos de usuário da estrutura log_credenciado_execcao
649	06/06/2010 17:04	gustavo.valle	close #1383, close #1391, close #1400, close #1416, close #1430, close #1438
648	06/06/2010 16:05	rodrigo.henriques	Alteração dos campos do tipo usuário na estrutura credenciado conta
647	06/06/2010 15:34	rodrigo.figueiredo	Atualização de cronograma

Ver diferenças

Ver todas as revisões

Exportar para [] Atoms

Figura 7. Visão do Repositório SVN através do Redmine

A utilização do Redmine para gerenciar todas as atividades do projeto permite a coleta automatizada de todas as medidas definidas pela empresa (com a exceção do formulário de satisfação do cliente ao final do projeto), o que auxilia a execução das atividades do processo Medição e Análise. Mensalmente, são utilizados alguns *scripts* de banco de dados na base do Redmine que exportam os dados dos projetos para uma planilha Excel, onde gráficos são gerados para os indicadores definidos no plano de medição organizacional.

Uma base OLAP está sendo construída para armazenar as medições, e um processo de ETL está sendo desenvolvido para trazer os dados do Redmine para a base de medição, permitindo que investigações possam ser realizadas nas medidas coletadas para identificar causas de problemas e possíveis soluções. Por último, toda a comunicação é realizada pelo Redmine. O coordenador do projeto e a equipe do projeto recebem os e-mails de mudança de estado das tarefas, evidenciando o andamento do projeto e provendo maior visibilidade e controle do projeto para o coordenador. O coordenador do projeto divulga as atas de reunião através dos fóruns do Redmine e solicita à equipe o comprometimento com o plano do projeto e com os requisitos através de comentários no fórum correspondente, como pode ser observado na Figura 8. Sempre que possível, a comunicação com o cliente também ocorre no Redmine, através de mudanças de estado das tarefas e comentários nos fóruns.

Título	Autor	Criado em	Respostas	Última mensagem
Registro de Comprometimento – Documento de Análise e Projeto	rodrigo.figueiredo	06/08/2010 12:36	1	Adicionado por rodrigo.figueiredo 3 dias atrás RE: Registro de Comprometimento – Documento de Análise e ...
Registro de Comprometimento – Plano do Projeto r74 e Especificação Funcional r46	rodrigo.figueiredo	26/07/2010 17:17	6	Adicionado por gao 3 dias atrás RE: Registro de Comprometimento – Plano do Projeto r74 e ...
Registro de Comprometimento – Plano do Projeto r92	rodrigo.figueiredo	29/07/2010 17:39	7	Adicionado por gao 3 dias atrás RE: Registro de Comprometimento – Plano do Projeto r92

Figura 8. Comunicação e Comprometimentos pelos Fóruns do Redmine

5. Considerações Finais

Este trabalho apresentou a metodologia utilizada para implantar as melhorias de processos necessárias para alcançar o nível F do modelo MR-MPS na Rightway Consultoria & Sistemas. As fases do processo padrão de desenvolvimento da empresa com as práticas da metodologia Scrum [ScrumAlliance 2010] adotadas pela empresa foram explicadas. As ferramentas utilizadas para apoiar a execução do processo também foram detalhadas. Os autores esperam que estas informações possam auxiliar outras empresas na combinação de práticas ágeis e modelos de maturidade e capacidade em seus processos de desenvolvimento e na seleção de ferramentas de apoio para a execução de seus processos.

Atualmente, a empresa possui os processos organizacionais já definidos e em execução, e está executando o terceiro projeto seguindo a segunda versão do processo padrão, estando em estágio avançado no projeto de implementação das melhorias de processo necessárias para alcançar o nível F do MR-MPS [Softex 2009]. A avaliação da empresa está prevista para novembro de 2010.

Referências

Edgewall (2010) "Trac". In: <http://trac.edgewall.org/>, accessed in 30/08/2010.

Ferreira, A.I.F., Santos, G., Cerqueira, R., et al. (2006) "Taba workstation: Supporting software process improvement initiatives based on software standards and maturity models". In: European Conference on Software Process Improvement, v. 4257 NCS, pp. 207-218, Joensuu, Finland, October 11-13.

Microsoft (2010) "Microsoft Office". In: <http://office.microsoft.com/>, accessed in 30/08/2010.

Montoni, M., Santos, G., Rocha, A.R., et al. (2006) "Taba workstation: Supporting software process deployment based on CMMI and MR-MPS.BR". In: 7th International Conference, PROFES 2006, v. 4034 NCS, pp. 249-262, Amsterdam, Netherlands, June 12-14.

Montoni, M., Santos, G., ROCHA, A.R., et al. (2007) "MPS Model and TABA Workstation: Implementing Software Process Improvement Initiatives in Small Settings". In: Fifth Workshop on Software Quality held in conjunction with the 29th Int. Conference on Software Engineering (ICSE), Minneapolis, USA, May.

Pentaho (2010). In: <http://www.pentaho.com/>, accessed in 30/08/2010.

Redmine (2010). In: <http://www.redmine.org/>, accessed in 30/08/2010.

Santos, G., Montoni, M., Figueiredo, S., et al. (2007a) "SPI-KM Lessons Learned from Applying a Software Process Improvement Strategy Supported by Knowledge Management". In: 8th International PROFES (Product Focused Software Development and Process Improvement), LNCS 4589, pp. 81-95, Riga, Latvia, July.

Santos, G., Montoni, M., Vasconcellos, J., et al. (2007b) "Implementing Software Process Improvement Initiatives in Small and Medium-Size Enterprises in Brazil". In: Quality of Information and Communications Technology, 2007. QUATIC 2007. 6th International Conference on the, pp. 187-198, Lisboa, Portugal, September 12-14.

ScrumAlliance (2010). In: <http://www.scrumalliance.org/>, accessed in 30/08/2010.

Rightway Consultoria & Sistemas (2010). In: <http://www.rightway.com.br>, accessed in 30/08/2010.

Softex (2009) "MPS.BR - Melhoria de Processo do Software Brasileiro, Guia Geral", pp. 57, accessed in 30/08/2010, disponível em <http://www.softex.br>.

Sparxsystems (2010) "Enterprise Architect". In: <http://www.sparxsystems.com.au/>, accessed in 30/08/2010.

Tigris.org (2010a) "Subversion (SVN)". In: <http://subversion.tigris.org/>, accessed in 30/08/2010.

Tigris.org (2010b) "TortoiseSVN". In: <http://tortoisesvn.tigris.org/>, accessed in 30/08/2010.

Implantação de Modelos de Maturidade com Metodologias Ágeis: Um Relato de Experiências

Rafael Prikladnicki¹, Ana Liddy C. C. Magalhães²

¹Faculdade de Informática (FACIN), PUCRS

Avenida Ipiranga, Prédio 32 - 90.619-900 - Porto Alegre - RS - Brasil

²Quality Focus - Rua Luiz Antonio Biazzetto, 174 - 82.200-050 - Curitiba - PR - Brasil

rafaelp@pucrs.br, ana.liddy@qualityfocus.com.br

Abstract. *The agile adoption in software development has been discussed for a long time. The purpose of this paper is to present a set of lived experiences with the implementation of maturity models using the agile approach. This work discusses the main concepts, values, principles and practices related to agile methods, as well presents the main challenges to make it possible the implementation of processes and practices that at the same time respect the agile culture and satisfy the requirements of maturity models such as the MR- MPS. The shared experience is based on several cases within organizations.*

Resumo. *A adoção do paradigma ágil no desenvolvimento de software tem sido um assunto amplamente debatido. Este artigo visa apresentar um conjunto de experiências vivenciadas na implantação de modelos de maturidade usando a abordagem ágil. Inicialmente são discutidos os principais conceitos, valores, princípios e práticas relacionados a abordagens ágeis para desenvolvimento e gerenciamento de software. Logo após são apresentados os cuidados necessários para realizar uma implementação que ao mesmo tempo respeite a adoção de uma cultura ágil e satisfaça as exigências de modelos de maturidade, como o MR-MPS. A experiência aqui compartilhada está baseada em diversos casos de implementação vivenciados.*

1. Introdução

A indústria de software tem evoluído para se tornar uma das mais importantes indústrias da era moderna (Cohn, 2009). O software está presente desde atividades mais simples, como processar informações básicas de compra e venda, até atividades mais complexas, como controlar carros em movimento ou aviões que voam por horas sem escala.

Por ser um ambiente altamente competitivo, as empresas geralmente se esforçam para serem as primeiras a lançar novidades, produtos e serviços diferenciados. Esta velocidade muitas vezes não é acompanhada de qualidade, fazendo com que funcionalidades, requisitos de desempenho e confiabilidade sejam sacrificados ao longo do caminho (Larman, 2003). O que separa as empresas de sucesso das empresas que fracassam? O que diferencia o melhor do segundo colocado? Muitas empresas criam novos produtos sem ter consumidores predefinidos ou às vezes possuem apenas uma vaga idéia do que precisa ser feito. Sendo software um processo criativo e de inovação, na maioria das vezes a diferença entre ser o primeiro ou o segundo está na habilidade de criar e entregar soluções de forma rápida e que agreguem valor ao negócio do cliente.

A velha máxima de entregar software funcionando com a maior qualidade possível, no menor tempo e no menor custo, continua presente. Ao mesmo tempo, a organização que “entrar por último” no mercado terá dificuldades em adquirir uma fatia desse mercado. Por isso, o *time-to-market* tornou-se um fator dominante para o sucesso industrial, criando uma grande pressão nos fabricantes e

desenvolvedores em gastar menos esforço em qualidade e desempenho, ou encontrar maneiras mais eficientes de produzir software de alta qualidade para esses produtos e serviços (Cohn, 2009).

É neste contexto que, a partir dos anos 90, surgiram as chamadas metodologias ágeis para desenvolvimento de software (Manifesto Ágil, 2010). Além de uma nova visão sobre como desenvolver software, ser ágil está associado a uma mudança cultural, uma nova forma de pensar, e não apenas mais uma abordagem, modelo, método ou processo de desenvolvimento. Ser ágil é assumir um conjunto de valores, princípios e práticas que flexibilizam o desenvolvimento de software de forma a responder às constantes mudanças do mercado, enfocando o que é importante (Larman, 2003).

A adoção do paradigma ágil no desenvolvimento de software tem sido um assunto amplamente debatido. Muitos destes debates concentram-se no uso de metodologias ágeis com modelos de maturidade para desenvolvimento de software. Neste contexto, este artigo visa apresentar experiências vivenciadas na implantação de modelos de maturidade usando a abordagem ágil. Inicialmente são discutidos os principais conceitos, valores, princípios e práticas relacionados a abordagens ágeis para desenvolvimento e gerenciamento de software. Logo após são apresentados os cuidados necessários para realizar uma implementação que ao mesmo tempo respeite a adoção de uma cultura ágil e satisfaça as exigências de modelos de maturidade, como o MR-MPS. A experiência compartilhada neste artigo está baseada em diversos casos de implementação vivenciados em organizações desenvolvedoras de software.

2. Metodologias Ágeis para Desenvolvimento de Software

O termo “metodologias ágeis” tornou-se popular em 2001 quando dezessete¹ especialistas em desenvolvimento de software representando diversas metodologias já existentes, tais como o *Extreme Programming* (XP), Scrum, DSDM (*Dynamic Systems Development Methodology*), Crystal e outros, estabeleceram princípios comuns compartilhados por todas elas. O resultado foi o estabelecimento do Manifesto Ágil (www.agilemanifesto.org) e da Aliança Ágil. Os conceitos chave do Manifesto Ágil são:

Indivíduos e interações são mais importantes do que processos e ferramentas;
Software funcionando é mais importante do que documentação abrangente;
Colaboração com o cliente é mais importante do que negociação de contratos;
Respostas rápidas a mudanças são mais importantes do que planos pré-estabelecidos.

O Manifesto Ágil não rejeita os processos e ferramentas, a documentação abrangente, a negociação de contratos ou o plano pré-estabelecido, mas indica que eles têm importância secundária quando comparados com os indivíduos e interações, com o software funcionando, com a colaboração com o cliente e as respostas rápidas a mudanças. A questão não é a mudança em si, mesmo porque ela ocorre de forma frequente nos projetos. O problema é como receber, avaliar e responder às mudanças.

¹ Os dezessete autores signatários do manifesto ágil são: Kent Beck, Mike Beedle, Arie van Bennekum, Alistair Cockburn, Ward Cunningham, Martin Fowler, James Grenning, Jim Highsmith, Andrew Hunt, Ron Jeffries, Jon Kern, Brian Marick, Robert C. Martin, Steve Mellor, Ken Schwaber, Jeff Sutherland, Dave Thomas.

Em geral, as metodologias ágeis não trazem conceitos novos para o desenvolvimento de software. O que as diferencia de outras metodologias são o enfoque e os valores. A ideia das metodologias ágeis é priorizar o enfoque nas pessoas e não em processos. Além disso, uma característica das metodologias ágeis é que elas são adaptáveis ao invés de serem prescritivas. Com isso, elas se adaptam a novos fatores decorrentes do desenvolvimento do projeto, ao invés de procurar analisar previamente tudo o que pode acontecer no futuro.

A tabela 2 ilustra um comparativo proposto entre o desenvolvimento tradicional e o desenvolvimento utilizando metodologias ágeis. Apesar de variarem em termos de práticas e ênfases, elas compartilham algumas características, como desenvolvimento iterativo e incremental, comunicação intensa e redução de produtos intermediários, em especial documentação extensa. Desta forma, existe mais possibilidade de atender aos requisitos solicitados pelo cliente, que na maioria das vezes não são claros no início dos projetos. Neste contexto, as metodologias ágeis têm interpretado o sucesso dos projetos de forma diferente da tradicional, onde sucesso significava entregar projetos dentro do prazo, dentro do orçamento e de acordo com as especificações definidas.

	Tradicional	Metodologias Ágeis
Pressupostos fundamentais	Sistemas previsíveis, que podem ser desenvolvidos a partir de um planejamento extensivo e meticuloso	Software adaptativo e de alta qualidade pode ser desenvolvido por equipes pequenas usando os princípios da melhoria contínua e rápida resposta a mudanças
Controle	Orientado a processos	Orientado a pessoas
Estilo de gerenciamento	Comando e controle	Liderança e colaboração
Gestão do conhecimento	Explícito	Tácito
Atribuição de papéis	Individual – favorece a especialização	Times auto-organizáveis – favorece a troca de papéis
Comunicação	Formal	Informal
Ciclo do projeto	Guiado por tarefas ou atividades	Guiado por funcionalidades do produto
Modelo de desenvolvimento	Modelo de ciclo de vida (Cascata, Espiral ou variação)	Modelo iterativo e incremental de entregas
Forma/estrutura organizacional desejada	Mecânica (burocrática com muita formalização)	Orgânica (flexível e com incentivos à participação e cooperação social)

Tabela 1. Comparação entre metodologias ágeis e metodologias tradicionais

O conhecido relatório do caos (*Chaos Report*) define projetos bem sucedidos como aqueles finalizados dentro do prazo, dentro do orçamento e que contemplam todas as funcionalidades originalmente especificadas; projetos desafiadores são os que finalizaram com um orçamento maior que o previsto, ou fora do prazo, ou sem contemplar todas as funcionalidades originalmente especificadas; e projetos que não obtiveram sucesso são os que foram cancelados em algum ponto do ciclo de vida de desenvolvimento. Mas nos últimos anos estas definições têm sido questionadas, pois um projeto pode ser bem sucedido e nunca trazer valor ao cliente ou pode ser considerado desafiador e trazer milhões de reais em lucro. Em 2006, um artigo publicado na revista CIO Magazine (<http://www.cio.com/archive/090106/applied.html>) alertava para o fato de que projetos que contemplam todos os critérios tradicionais de sucesso – tempo, orçamento e escopo – ainda podem ser falhos ao final por falharem em atender a intenção dos usuários ou porque simplesmente falham em agregar valor ao negócio.

Ou ainda, projetos considerados como falhos de acordo com as métricas tradicionais podem ser considerados bem sucedidos, pois, mesmo com atrasos de tempo, mais custo e menos funcionalidades, o produto é bem recebido pelo seu público-alvo.

Por estes motivos, as metodologias ágeis têm desempenhado um papel fundamental para o desenvolvimento de software moderno, estabelecendo uma nova forma de pensar projetos, onde o valor que o projeto agrega e as interações entre as pessoas são considerados tão ou mais relevantes que o cumprimento de prazos, custo ou atendimento ao escopo inicialmente definido. Em pouco mais de uma década, as metodologias ágeis têm modificado a forma que empresas conduzem seus projetos, negociam seus contratos, gerenciam seus times, capacitam seus profissionais. É visível a mudança de postura de gerentes, de membros de times, de executivos de empresas. Profissionais têm se tornado mais completos. Produtos têm sido desenvolvidos com mais qualidade. Clientes têm ficado mais satisfeitos. Mas sempre é bom lembrar que metodologias ágeis podem não funcionar em todos os casos.

3. Implantação de Modelos de Maturidade com Metodologias Ágeis

Em função da escassez de relatos de experiências no tema, surgiu a idéia de desenvolver um estudo informal junto a implementadores e avaliadores credenciados pelo Programa MPS.BR, bem como colaboradores de empresas que estão conduzindo programas de melhoria em empresas brasileiras que utilizam princípios e práticas da abordagem ágil para desenvolvimento ou gerenciamento de projetos de software. Em busca de uma visão mais ampla deste cenário nas empresas, um e-mail foi distribuído para a comunidade envolvida com a melhoria de processos solicitando informações acerca da abordagem ágil empregada, ferramentas utilizadas, além de lições aprendidas em implementações e avaliações ágeis. Informações referentes a 20 fontes diferentes, de várias localidades, foram compiladas, classificadas e tratadas por assunto em uma planilha. A relação das pessoas que colaboraram com este estudo é apresentada na seção de agradecimentos. Os resultados apresentados não foram quantificados, pois não houve em nenhum momento o interesse em se estabelecer uma visão quantitativa do que está ocorrendo no mercado, mas uma visão qualitativa, que pudesse apresentar idéias, sugestões e lições para a comunidade envolvida com o Programa MPS.BR e para a comunidade ágil. Os dados coletados foram divididos em tópicos, apresentados a seguir.

Características das organizações

Em primeiro lugar, é interessante apresentar as características gerais das organizações que estão realizando implementação MPS de forma ágil. Em relação ao objetivo da implementação, as organizações estão buscando a melhoria de seus processos e produtos, acreditando ser a abordagem ágil adequada para a sua realidade, porém nem todas possuem como meta a obtenção de um nível de maturidade. Algumas organizações já foram concebidas seguindo a cultura ágil – ou seja, trabalham com XP e/ou Scrum desde seu início. Outras não possuíam processos e perceberam na abordagem ágil uma boa opção para sua estruturação. Algumas seguiam a abordagem tradicional e resolveram experimentar a abordagem ágil em alguns de seus projetos – estas englobam não só empresas particulares, mas também órgãos governamentais (Secretaria Estadual de Saúde, Secretaria da Fazenda, Tribunal de Justiça).

Formas de implementação

As formas de implementação adotadas variam bastante. Algumas aplicam XP de forma adaptada, pela dificuldade em seguir todas as práticas ou por características específicas da empresa, como por exemplo, empresa que desenvolve jogos e aplica parcialmente o XGD (*eXtreme Game Development*).

Outras optaram por aplicar o Scrum de forma completa ou adaptada, sendo a adaptação decorrente de possuir equipes grandes e/ou distribuídas. As principais adaptações são decorrentes de: adequação do tamanho dos *sprints*; forma de apresentação de *backlogs*; variações na forma de realizar estimativas baseadas em *planning poker*; frequência e registros derivados das reuniões (em especial das reuniões rápidas de acompanhamento). Também foi citado o uso simultâneo das abordagens XP e Scrum de forma adaptada, visando empregar a filosofia ágil para desenvolvimento e gerência. Uma empresa ainda citou aplicar uma metodologia própria baseada em XP e Scrum, com influência de *Lean Programming*, na qual um “evangelizador” interno influenciou a equipe de processos. Outra desenvolveu sua abordagem baseada no OpenUP, fazendo uma transição a partir do RUP.

A quebra de paradigma

A quebra de paradigma foi apontada como um aspecto que merece atenção. Devido à dificuldade de se quebrar posicionamentos radicais, torna-se necessário avaliar as condições da empresa para assimilar as práticas ágeis e/ou os modelos de maturidade antes de iniciar o programa de melhoria. Um fator determinante para o sucesso é a motivação (“motivo para ação”) dos envolvidos. Qualquer que seja a nova abordagem a ser utilizada, a falta de motivação para seu uso dificulta todo o trabalho de melhoria, sendo mais propício implantar práticas ágeis em equipes motivadas e abertas, que estejam comprometidas com os valores da empresa. De toda forma, envolver a equipe nas discussões facilita a institucionalização. Também é necessário manter o foco dos trabalhos e evitar sobrecarga (tanto pela equipe quanto pelo implementador), tendo em mente os princípios da abordagem ágil durante toda a implementação e enfocando principalmente valores e princípios, uma vez que as práticas podem ser adaptadas.

Respeito à cultura local

A cultura local deve ser respeitada e seguir à risca uma metodologia, independente de qual seja, não é em geral uma boa abordagem. Cada empresa tem suas características e o bom senso deve prevalecer. Em empresa acostumada com as práticas ágeis, a adaptação para atender o modelo de maturidade é mais sentida, uma vez que qualquer ajuste causa impacto perceptível e questionável, sendo necessário estudar cada ação, pesquisar formas de reverter em ganho para a equipe qualquer “burocracia” implantada. Já em uma empresa acostumada com a abordagem tradicional, a adoção da abordagem ágil pode trazer uma mudança cultural muito grande para a organização, e eventuais dificuldades na execução do processo podem gerar resistência. Assim, o ideal é buscar a migração da forma mais suave possível, adotando passos pequenos e introduzindo as práticas aos poucos. Também é interessante criar consciência da necessidade de uma prática usando métricas e avaliar a experiência obtida na adoção da prática, sempre buscando a compatibilidade com os processos e artefatos já em uso.

Contratação e envolvimento de consultoria

Alguns colaboradores das empresas alertaram sobre a necessidade de se tomar alguns cuidados na contratação e envolvimento de consultoria. É importante que esta não modifique os valores da organização e que procure adequar o modelo à realidade da organização e não de forma contrária. Neste sentido, orientam buscar no mercado empresas de consultoria que tenham experiência com as duas abordagens e que saibam lidar com as dificuldades culturais na implementação híbrida.

Consultores e avaliadores não estão ainda acostumados a conciliar modelos de maturidade com métodos ágeis, o que pode não só dificultar a comunicação, o entendimento e acabar gerando resistência, mas também consumir maior número de horas. De toda forma, é necessário alinhar claramente os objetivos definidos no início do projeto, buscando o comprometimento de todos os envolvidos – patrocinador do projeto de melhoria, equipe interna e consultoria.

Uso de ferramentas

Para uma implementação eficiente, torna-se necessário utilizar uma ferramenta de controle apropriada. Embora o Scrum pregue o uso do quadro com *post-its*, na prática isto dificulta manter o histórico. Nesta direção, o uso de uma ferramenta que ofereça uma abstração deste quadro possibilita maior agilidade e visibilidade para a equipe, sem quebrar a filosofia Scrum. Percebe-se também que o apoio de ferramenta ajuda na implementação, e empresas “ágeis” tendem a usar ferramentas de software livre.

Necessidade e nível de detalhamento da documentação

Os modelos de maturidade reforçam a importância de se documentar e não a necessidade do documento em si. Dados e informações relevantes podem ser disponibilizadas em ferramentas, enquanto filmes de sessão de levantamento de requisitos e fotografias de quadro demonstrando acompanhamento são válidos como evidências objetivas para uma avaliação. Ao buscar o equilíbrio entre a agilidade e a maturidade, em geral fica difícil estabelecer um nível de detalhamento e documentação do produto e de seu processo de construção. Em geral, modelos de maturidade exigem documentação técnica detalhada da solução, enquanto a abordagem ágil dispensa documentação detalhada, ao possibilitar troca de informação e detalhamento técnico da solução de forma constante e iterativa, já próximo ao desenvolvimento. O próprio código pode ser a documentação se os padrões de codificação e boas práticas levarem a um código auto-explicativo. Na busca de um equilíbrio, deve-se preservar o que é considerado essencial da documentação, em especial a documentação técnica, uma vez que uma documentação pesada dificulta a gestão e a capacidade de reação quando ocorrem mudanças de requisitos.

3.1. Considerações sobre os Níveis de Maturidade

Implementação do nível G

Empresas que empregam o *scrum* (ou uma variação deste) e pretendem implementar o processo

Gerência de Projetos (GPR) tendem a ter um quadro com *post-its* mais detalhado. Como os modelos de maturidade requerem o registro de dados históricos, as empresas acabam gerando registros em alguma ferramenta de apoio (em geral gratuita) ou mesmo em planilhas eletrônicas. A adoção do conceito de que estória pronta é a estória implementada, testada e aprovada pelo *product owner* (representante do cliente no *scrum*) contribuiu para a detecção antecipada de problemas, aliviando custos relacionados a defeitos ao longo do projeto. O conhecimento de impedimentos, obtido durante as reuniões diárias, despertou maior colaboração e comunicação entre os membros do time, reforçando a necessidade de participação de todos em busca do objetivo definido para o *sprint*. Já a atribuição de penalidade para quem falta ou atrasa para reuniões da equipe mostrou-se válida como forma de reforçar o comprometimento de todos. De uma maneira geral, o ritmo da equipe se mantém sustentável, sendo possível produzir sem utilizar horas extras e com espaço para inovação e diversão.

Quanto à realização de estimativas segundo a técnica do *planning poker*, notou-se que o time *scrum* necessita de algum tempo para compreender melhor o problema em questão visando estimar de forma mais adequada. Assim, a velocidade da equipe e a quantidade de estórias finalizadas nos primeiros *sprints* podem não ocorrer no ritmo esperado e isso pode não significar baixa produtividade do time. Foi também relatado um caso no qual a técnica do *planning poker* sofreu adaptação para contemplar níveis de conhecimento diferenciados (ex: júnior, pleno e sênior), o que auxiliou no ajuste das estimativas em função do perfil alocado para cada tarefa.

Ao considerar o *scrum* como uma forma de gerenciar projetos, é importante observar que não existe uma correspondência direta entre o papel do *scrum master* e do gerente de projetos – o que nem todo implementador ou avaliador tem conseguido entender. O papel do gerente de projetos é diluído dentro da equipe, uma vez que passa a ser responsabilidade desta avaliar escopo, estimativa e prazo, deixando para o *scrum master* a responsabilidade pela remoção de impedimentos e pelo tratamento de aspectos administrativos do projeto. Entre as adaptações realizadas, destacam-se duas propostas, que ainda estão sendo monitoradas: (1) o gerente de projetos é responsável pelo planejamento macro, pela gestão dos recursos e pelo acompanhamento em marcos, podendo responder por vários projetos, enquanto o *scrum master* cuida do acompanhamento diário e dos impedimentos, atuando em um único time de cada vez; (2) o gerente de projetos fica responsável pela evolução e melhoria do produto, atuando de forma semelhante a um *product owner*, enquanto o *scrum master* gerencia as estórias por ele encaminhadas (que compreendem requisitos funcionais e não funcionais).

Com relação aos papéis na equipe de desenvolvimento, em empresas que possuem funções técnicas específicas, é mais difícil implantar times *scrum* de fato, onde todos fazem “de tudo”. Neste caso, uma sugestão (ainda sendo monitorada) é realizar uma abordagem de transição, com o time dividido dentro de um *sprint*, no qual os analistas atuam como *scrum master* e geram os requisitos do *sprint* seguinte enquanto os desenvolvedores e *testers* constroem requisitos detalhados no *sprint* anterior. De toda forma, é importante que cada integrante do time saiba exatamente o seu papel e seus limites de atuação na metodologia seguida pela empresa. Para o *scrum*, o *product owner* representa os interesses do cliente e neste caso, mesmo se a pessoa que executa este papel possuir perfil técnico, não deverá interferir nas atividades técnicas da equipe, pois isso pode atrapalhar e gerar desconforto. Da mesma forma, *scrum master* é o facilitador do processo e responsável pelos impedimentos, não devendo se comportar como um gerente de projetos, pois a equipe deve ser autogerenciável.

Para equipes grandes e/ou distribuídas, o uso de *scrum* de *scrum* pode ser aplicado, dividindo o time

em grupos menores. Como sugestão, pode ser criado um time de apoio, responsável por centralizar definições de arquitetura, padrões de desenvolvimento, repositórios e ambientes de programação. Ao final de uma *sprint* ocorreriam *retrospective meetings* internas (para os pequenos times) e uma externa (envolvendo o *scrum master*, o *product owner* e o time de apoio). Reuniões de *scrum* de *scrums* poderiam ser semanais, preferencialmente presenciais, ou alternativamente utilizando *skype* e câmera de vídeo. O foco destas reuniões estaria na resolução de impedimentos globais e no fornecimento de visibilidade do andamento geral do projeto, com discussões técnicas ou resolução de dependências sendo tratadas separadamente. A comunicação informal entre *scrum masters* poderia ser virtual.

Para a implementação do processo Gerência de Requisitos, foi observado que a rastreabilidade dos requisitos até o código se torna meio “nebulosa” com o conceito da estória, pois uma estória pode possuir uma relação “de muitos para muitos” com os requisitos e casos de uso, além de não ser documentada em detalhes. Além disso, apesar do *scrum* propor escopo aberto, modelos de maturidade requerem definir um escopo que será estimado, planejado e acompanhado pelo projeto, sendo necessário adaptar o *scrum*. Desta forma, perde-se parte da agilidade em prol do modelo de maturidade, o que pode introduzir desperdício dependendo de como é implementado.

Implementação do nível F

Os processos de nível F não são inerentes aos métodos ágeis, motivo pelo qual sugere-se manter equipes organizacionais de apoio, responsáveis pela execução das atividades correspondentes em todos os projetos (em especial para gerência de configuração, garantia da qualidade e medição). Tal abordagem reduz o impacto destas atividades sobre o time do projeto, que pode continuar realizando seu trabalho de forma ágil. Tais equipes devem participar das reuniões de planejamento da *sprint* visando garantir o alinhamento ao processo e verificar potenciais conflitos. Pelo que pôde ser observado, a existência de auditoria de Garantia da Qualidade despertou o compromisso da equipe com a aderência às atividades realizadas e aos artefatos gerados segundo o processo existente, facilitando sua institucionalização. A implantação da disciplina de Medição mostrou que a existência de métricas pode auxiliar na identificação de impedimentos e ajudar no planejamento realizado pela equipe. Para a Gerência de Configuração, a principal dificuldade apontada foi definir o momento certo de geração de *baselines*, uma vez que métodos ágeis enfocam a integração contínua. Outra dificuldade apontada, tanto para a Garantia da Qualidade quanto para a Gerência de Configuração, diz respeito ao momento correto para realizar auditorias e ao conteúdo a ser verificado em cada uma.

Implementação do nível E

Para que a maturidade alcançada seja sustentável e agregue valor ao negócio, o processo padrão adaptável deve refletir a forma real da empresa trabalhar e não uma aplicação “à risca” do modelo ou método de referência. Para a definição deste processo padrão é imprescindível balancear as necessidades da empresa, os preceitos ágeis e os requisitos do modelo de referência em busca de uma solução que satisfaça necessidades até certo ponto divergentes e alinhadas à cultura da empresa. Nesta direção, empresa, consultores e avaliadores necessitam ter mente aberta para novas abordagens de implementação e interpretações do modelo. Visando facilitar este desafio e aumentar a sinergia, sugere-se manter no SEPG uma pessoa com conhecimentos arraigados de metodologias ágeis e outra com fortes conhecimentos no modelo de maturidade, para que a definição dos processos passe por

ricas discussões alinhadas ao objetivo do programa de melhoria.

Entre as lições aprendidas na definição de processos, destacam-se: não se deixar levar por soluções prontas, manter-se aberto a soluções inovadoras; não se “apaixonar” pelos processos e aceitar mudanças mesmo após uma avaliação bem sucedida; permitir que o processo se estabilize, pois é necessário executá-lo nos projetos e coletar medidas consistentes; preservar a adaptabilidade e flexibilidade da abordagem ágil, enfocando atividades de adaptação do processo (alguns *sprints* necessitam gerar mais artefatos do que outros, cabendo esta decisão ao time nos *sprint plannings*); o uso de ferramenta para documentar o processo facilita sua evolução e rápida liberação de novas *releases*.

Implementação do nível D

Abordagens ágeis são criticadas por realizarem o detalhamento o mais próximo possível do desenvolvimento e por apresentarem pouco design de arquitetura, pouca documentação e formalização. Todavia, times ágeis geram documentação quase tanto quanto os demais, porém buscam a documentação enxuta (somente com elementos essenciais) e escolhem arquiteturas descomplicadas, que aceleram a programação (Ambler, 2010). Entre as principais lições aprendidas relacionadas aos processos de engenharia, destacam-se: a propriedade coletiva do código possibilitou disseminar o conhecimento pela equipe, uma vez que todos necessitam conhecer (um pouco) de tudo; a pressão de tempo dificulta o desenvolvimento dirigido por testes e, conseqüentemente, a integração contínua; além de possibilitar evoluir a aplicação de forma incremental, a liberação frequente de *releases* aumentou a expectativa dos clientes por software rodando, o que pode gerar frustração por ainda não estar completo; passar a usar as práticas do XP não é fácil, mas seu retorno pode ser muito proveitoso; é difícil gerar evidências das “boas práticas” do XP, tais como programação por pares, *refactoring*, integração contínua, propriedade coletiva e desenvolvimento orientado a testes.

Implementação do nível C

Os valores e práticas ágeis ajudam a gerenciar e mitigar riscos, uma vez que: a busca pela simplicidade diminui a complexidade; o *feedback* antecipa a detecção de erros; a comunicação aberta minimiza problemas de informação; a quebra em iterações e o planejamento constante ajudam a controlar prazo e custos; o cliente disponível e a entrega em *releases* diminuem o risco de se obter produtos inadequados; reuniões diárias possibilitam identificar mais cedo a iminência de um risco, permitindo atuar a tempo de minimizar suas conseqüências. Decisões são tomadas por equipes ágeis ao longo do projeto, e o cuidado adicional necessário está em registrar não só as questões e seu tratamento, mas também as alternativas de solução e os critérios para avaliação destas alternativas. Em relação ao desenvolvimento para reutilização, ainda existe pouca experiência prática. Uma iniciativa de reutilização sistemática ágil interessante e aparentemente simples emprega, de forma disciplinada, metáforas e palavras-chave na especificação de histórias e na geração de classes e métodos. Estes são posteriormente varridos por uma ferramenta que identifica, registra e disponibiliza os relacionamentos existentes visando apoiar desenvolvimentos futuros (Lima et al, 2010).

4. Conclusões

Nos últimos anos muito se discutiu sobre a possibilidade de implementar modelos de maturidade seguindo uma abordagem ágil, com muitas opiniões radicais. Hoje já existem alguns relatos com iniciativas de aproximação bem sucedidas, como é o caso da Powerlogic - MPS Nível C (Oliveira et al, 2010). O próprio *Software Engineering Institute* (SEI) publicou um relatório a respeito (Glazer et al, 2008). Mais recentemente, empresas também tem se apoiado em metodologias ágeis para guiar o próprio programa de melhoria de processo da empresa (Salgado et al, 2010).

Nestes relatos, fica evidente que algumas percepções a respeito dos modelos de maturidade acabaram definindo uma meia verdade sobre a viabilidade da proposta para o desenvolvimento de software nos dias de hoje. E isto é recíproco, ou seja, percepções tendenciosas ocorreram de ambos os lados. Mas se isto for analisado com cuidado, se perceberá que a falta de informação tem sido um dos grandes fatores de ruído neste processo. David Hussman comenta que os profissionais devem sempre se questionar sobre o valor que estão agregando, independente da metodologia sendo usada. David entende que valor é medido pela razão entre “por que” e “como”. Não adianta ter diversos “como” e poucos “porquês”, pois isto diminuir o valor agregado (<http://devjam.com/dudesblog/dudes-law>).

Além disso, a falta de informação motivou Alistair Cockburn, um dos signatários do Manifesto Ágil, a criar o “Juramento de Não-Lealdade”. Segundo ele, nenhuma idéia deve ser desconsiderada na sua origem: idéias provenientes de todas as escolas e tradições devem ser consideradas com o objetivo de encontrar aquelas que melhor se encaixem em cada situação (<http://alistair.cockburn.us/oath+of+non-allegiance>). Cockburn manifestou assim sua insatisfação em relação a “pessoas de uma escola de pensamento atacando idéias de outra” e sua ansiedade por “pessoas que não se importem com a origem das idéias, mas com o seu significado e o que elas são capazes de produzir”. É isto que se espera da integração entre o MR-MPS e a abordagem ágil, e foi isto que se buscou neste artigo.

Agradecimentos

Para o desenvolvimento deste artigo contou-se com a experiência de diversos profissionais que atuam na área de melhoria de processo de software e já tiveram experiências com metodologias ágeis. São eles: Alexandre Vasconcelos (UFPE), Ana Marcia D. Duarte e Nikolai Dimittri (Innovit), Ana S. Marçal (UNIFOR), Anne Elise Katsurayama e Analia Irigoyen (Promove), Cristiano Schwening (EngSoft), Eliane Collins (INdT), Fernando K. Kamei (UFPE), Gleison Santos (COPPE/Unirio), José M. Monteiro (UFC), Ludmila Roizenbruch (Hermes Pardini), Marcello Thiry e Alessandra Zoucas (Incremental), Márcia Alves e Isabella Campos (PowerLogic), Marcilio Ferreira S. Júnior (IFAL), Maria das Dores Resende e Renato Sales (AAIS), Maria Istela Cagnin (UFMS), Odisnei Galarraga (Software Process), Teresa Maciel (UFRPE)

Referências

Ambler, S. Modeling and Documentation Practices on IT Projects Survey Results. Disponível em <http://www.ambysoft.com/surveys/modelingDocumentation2008.html>. Acesso em Agosto de 2010.

Cohn, M., "Succeeding with Agile: Software Development Using Scrum", Addison-Wesley Professional, 1a edição, 2009.

Glazer, H., Dalton, J., Anderson, D., Konrad, M., Shrum, S., "CMMI or Agile: Why Not Embrace Both", Software Engineering Institute, 2008.

Larman, C., "Agile and Iterative Development: A Manager's Guide", Addison-Wesley Professional, 2003.

Lima, E., Oliveira, F., Murta, L., Werner, C. "CodeKeySearch: aliando reutilização de software com métodos ágeis". Disponível em www.ic.uff.br/~leomurta/papers/lima2008.pdf.

Manifesto Ágil. Disponível em <http://www.manifestoagil.com.br/>. Acesso em Agosto de 2010.

Oliveira, A. C. G. Guimarães, F. A., Fonseca, I. A., "Utilizando Metodologias Ágeis para Atingir MPS.BR nível F na Powerlogic". Disponível em www.softex.br/mpsbr.

Salgado, A., Melcop, T., Acchar, J., Rego, P. A., Ferreira, A. I. F., Kasturayama, A. E., Montoni, M., Zanetti, D., "Aplicação de um Processo Ágil Para Implantação de Processos de Software baseado em Scrum

Padrões de Processo para Atributos de Processo

Juliano Lopes de Oliveira¹, Paulo Henrique R. Faleiro¹, Adriana Silveira de Souza¹

¹Instituto de Informática - Universidade Federal de Goiás (UFG) – Caixa Postal 131 - Campus II – CEP 74001-970 – Goiânia – GO – Brasil

juliano@inf.ufg.br, paulofaleiro@gmail.com, adriana@inf.ufg.br

Abstract. *Process Attributes are critical elements in Software Process Improvement (SPI) initiatives. In the MR-MPS context, these attributes define the processes capacity in a given maturity level. In spite of their importance, and of the impact they cause in SPI initiatives, Process Attributes do not receive much attention from the SPI research community. This paper proposes the use of process patterns to organize the knowledge about process attributes in order to minimize the barriers to their implementation in SPI initiatives.*

Resumo. Atributos de processo são elementos críticos em iniciativas de Melhoria de Processos de Software (MPS). No MR-MPS esses atributos definem a capacidade dos processos em um determinado nível de maturidade. Apesar de sua importância, e do impacto que causam em iniciativas de MPS, atributos de processo, e resultados de atributos de processo, não recebem muita atenção da comunidade de pesquisa em MPS. Este artigo propõe o uso de padrões de processo para sistematizar o conhecimento sobre atributos de processo de modo a diminuir as barreiras para sua implementação em iniciativas de MPS.

1. Introdução

A qualidade dos processos de software tem cada vez mais se tornado um fator crítico para o sucesso de organizações produtoras de software. No Brasil, em particular, muitas iniciativas de Melhoria de Processos de Software (MPS) surgiram após o lançamento do MPS.BR, um programa mobilizador, coordenado pela Softex, com o objetivo de melhorar o processo de software brasileiro [Softex 2009].

Apesar dos avanços notáveis que já foram conquistados com a implantação do MPS.BR [Travassos e Kalinowski, 2009], é comum as organizações encontrarem grandes dificuldades quando estão implementando iniciativas de MPS [Rocha et. al. 2005]. Muitas dessas dificuldades estão relacionadas com o domínio, por parte da equipe técnica da organização que executa as atividades de MPS, dos conceitos teóricos e das práticas recomendadas pelos modelos de qualidade, tais como o MPS.BR e o CMMI [Mendes et. al. 2007].

A falta de conhecimento sobre o modelo adotado para a iniciativa de MPS é apontado como um dos fatores que levam à resistência às mudanças decorrentes de ações de MPS [Brietzke e Rabelo 2006].

Em geral, consultores de implementação do modelo adotado são alocados junto às equipes técnicas das organizações que executam projetos de MPS para minimizar os problemas decorrentes da sua inexperience e das dificuldades que estas equipes tipicamente demonstram para compreender os propósitos e os resultados esperados pelos modelos de qualidade. O trabalho de [Fernandes e Oliveira 2010] destaca a presença desses consultores de implementação como um fator que contribui para a implantação e para a institucionalização de processos de software em iniciativas de MPS.

O presente trabalho propõe uma abordagem para reduzir a dificuldade que as organizações produtoras

de software apresentam com relação à compreensão dos requisitos do modelo de qualidade de processo MPS.BR. Esta dificuldade de compreender as bases do modelo leva as iniciativas de MPS a serem exageradamente dependentes dos consultores de implementação, retardando significativamente a institucionalização dos processos melhorados.

A ideia básica da abordagem aqui proposta é o uso de padrões de processo para representar, de forma semi-estruturada, o conhecimento e a experiência que os implementadores de MPS possuem, e que é essencial para o sucesso de uma iniciativa de MPS.

De fato, padrões de processo têm sido usados nesse sentido há algum tempo [Ambler 1998; Coplien 1998]; porém, a abordagem aqui proposta estende esse uso para contemplar um aspecto de MPS que não tem recebido atenção suficiente da comunidade de padrões de processo: os atributos de processos.

Praticamente todos os trabalhos relacionados com padrões de processo focam em aspectos de um processo específico, deixando de lado os atributos de processo, que são igualmente essenciais para o sucesso de uma iniciativa de MPS.

O restante desse artigo descreve a abordagem proposta para utilizar padrões de processo no contexto de atributos de processo e está organizado da seguinte forma. A Seção 2 sintetiza os conceitos de padrão de processo e atributo de processo, que formam a base teórica da proposta. A Seção 3 discute a forma proposta para organizar o conhecimento sobre a implementação de atributos de processo com base em padrões de processo. A Seção 4 apresenta um exemplo simplificado do uso dessa proposta para descrever um padrão de processo para um atributo de processo do MPS.BR. Finalmente a Seção 5 traz as considerações finais deste trabalho e aponta direções para trabalhos futuros.

2. Padrões de Processo e Atributos de Processo

Um **Padrão de Processo** define uma solução geral e comprovada, ou abordagens de sucesso para um problema específico e recorrente em processos de desenvolvimento e/ou manutenção de software [Ambler 1998]. Uma característica importante de padrões de processos é que eles descrevem o que deve ser feito, e não detalhes sobre a maneira como deve ser feito algo.

Assim, pelo fato de não especificarem como realizar uma determinada tarefa, os padrões de processo podem ser empregados como blocos de construção reutilizáveis que podem ser aplicados na construção de blocos mais complexos. Padrões de processo podem ser considerados, portanto, como componentes de processo reutilizáveis.

Resultados pioneiros na utilização de padrões de processo para auxiliar nas atividades de melhoria de processos de software foram apresentados em [Ambler 1998; Ambler 1999; Coplien 1998].

Apesar de ser uma área de pesquisa já consolidada, não há um consenso na comunidade científica com relação a definições e taxonomias envolvidas na área de padrões de processo [Tran et al 2006]. No presente trabalho são adotados os conceitos e classificações propostas em [Ambler 1998], por se tratar de um trabalho seminal, que é amplamente citado na literatura de padrões de processo.

Vale notar que o termo “Padrão” (ou “*Pattern*”, em inglês) é amplamente utilizado pela comunidade de Computação, como um todo. O presente trabalho restringe o uso desse termo ao contexto de processos e, mais especificamente, a padrões de processo.

Os Padrões de Processo de Software podem ser classificados em três tipos principais [Ambler 1998], de acordo com a granularidade e a complexidade do problema abordado:

- Padrões relacionados com Tarefa (*task process pattern*): descrevem detalhadamente a sequência de ações que deveriam ser realizadas para resolver um problema específico de granularidade baixa;
- Padrões relacionados com Estágio (*stage process pattern*): definem os passos que devem ser realizados e, em geral, a ordem em que devem ser executados, para resolver um problema de maior granularidade. Tipicamente são padrões compostos por um conjunto de padrões de processos do tipo *tarefa*;
- Padrões relacionados com Fase (*phase process pattern*): representam a interação entre dois ou mais padrões de processo do tipo *estágio* para resolver um problema complexo.

Essa taxonomia de padrões de processo é similar à divisão de processos definida na norma ISO 12207, que considera as atividades de um processo e as tarefas de uma atividade [ISO, 2008]. O exemplo a seguir, baseado em [Asadi 2009], ilustra, de forma resumida, a definição de um padrão de processo composto que utiliza essa taxonomia:

- Fase: Implantação
 - o Estágio: Teste Sistemico
 - Tarefa: Verificação / Validação
 - Tarefa: Verificar a completude das tarefas
 - o Estágio: Implantação
 - Tarefa: Preparar Documentos
 - Tarefa: Treinar Desenvolvedores
 - Tarefa: Definir equipe de suporte

A descrição de um padrão de processo deve conter os elementos estruturais necessários para a compreensão da ideia do padrão, de forma clara, ainda que abstrata. Em geral, as descrições de padrão encontradas na literatura contemplam pelo menos os seguintes aspectos estruturantes de um padrão de processo:

1. o **Problema**, que expressa qual é o propósito (ou objetivo) do padrão de processo;
2. o **Contexto Inicial**, que caracteriza as condições ou situações que levam à aplicação do padrão de processo;
3. o **Contexto Resultante**, que define quais resultados que devem ser obtidos após a aplicação do padrão de processo;

4. a **Solução**, que descreve um modelo de solução para o problema apresentado.

A Figura 1 apresenta um exemplo

Nome:
Definição do Tamanho do Cronograma.
Problema:
Quanto tempo um projeto deveria durar?
Contexto:
O problema foi entendido e o tamanho do projeto precisa ser estimado.
Forças:
Se você define um cronograma muito generoso, os desenvolvedores se tornam acomodados, e você pode perder oportunidades de negócio. Por outro lado, se você define um cronograma com um curto período para a conclusão do projeto, os desenvolvedores provavelmente não irão concluir o projeto a tempo e, conseqüentemente, haverá prejuízos perante o cliente.
Solução:
Recompensar os desenvolvedores pelo cumprimento do cronograma com bônus financeiros ou com hora extra.

Figura 1. Padrão de processo – Definição do Tamanho do Cronograma. Adaptado de [Coplien 1998]

O Modelo de Referência do MPS.BR (MR-MPS) define **Atributo de Processo** da seguinte forma: “Atributo de processo: Uma característica mensurável da capacidade do processo aplicável a qualquer processo” [Softex 2009].

Os atributos de processo (AP) são usados no MR-MPS para representar a capacidade dos processos, de acordo com a norma ISO 15504 [ISO/IEC 2004]. Essa capacidade dos processos determina o seu grau de refinamento e institucionalização na organização que os utiliza [Softex 2009].

No MR-MPS, os atributos de processo são avaliados a partir dos resultados esperados de atributo de processo (RAP). Todos os processos de um determinado nível devem apresentar os RAP correspondentes a este nível. Assim, a implementação de RAP (e de Atributos de Processo) é determinante para o sucesso de uma iniciativa de MPS baseada no MR-MPS.

3. Compreendendo Atributos de Processos por meio de Padrões de Processos

Apesar dos benefícios já observados da aplicação de padrões de processo em iniciativas de melhoria de processo de software, os atributos de processo ainda são pouco explorados nas pesquisas realizadas nesse contexto.

Uma análise detalhada da correlação entre atributos de processo (segundo a definição do MPS.BR e da ISO 15504) e Padrões de Processo está sendo conduzida pelos autores sob a forma de uma revisão sistemática da literatura. Os resultados preliminares dessa revisão mostram a exigüidade de pesquisas e publicações relacionadas especificamente com atributos de processo.

Por outro lado, há uma quantidade significativa de publicações sobre padrões de processos que buscam solucionar problemas que, no contexto do MPS.BR, seriam relacionados com Resultados Esperados de processos específicos, ou com grupos de Resultados Esperados que, em alguns casos, podem formar um processo completo.

Portanto, pode-se perceber que não há resultados de pesquisas em padrões de processos que propõem soluções para problemas relacionados a Atributos de Processos e a Resultados de Atributos de Processos. Este artigo busca contribuir para esta área de pesquisa pouco explorada, que é a elaboração de padrões de processos para atributos de processo. A ideia é encontrar problemas recorrentes em iniciativas de melhoria de processo de software relacionados a implementação de atributos de processos do MPS.BR, e catalogar as respectivas propostas de soluções para esses problemas.

O exemplo apresentado a seguir descreve um padrão de processo relacionado a um resultado de atributo de processo do MR-MPS. Para esta descrição, além das informações propostas em [Ambler 1998], duas novas seções foram acrescentadas ao padrão:

1. Forças, que definem as motivações para a aplicação do padrão de processo; e

2. Padrões Relacionados, que tem como propósito listar os padrões de processos relacionados ao padrão descrito.

Com relação à seção sobre padrões relacionados, em [Washizaki 2005] é apresentada uma classificação que define três tipos de padrões. Porém para este trabalho somente foi considerado relevante o padrão relacionado do tipo Contexto Resultante – Contexto Inicial. Esse tipo de padrão ocorre quando o contexto resultante de certo padrão $p1$ leva ao contexto inicial de outro padrão $p2$. Isso implica que quando se aplicar o padrão $p1$ deve-se aplicar em seguida o padrão $p2$.

O Resultado de Atributo de Processo do MR-MPS escolhido para exemplificar a utilização de padrões de processos no contexto de atributos de processos foi o RAP 2 (Existe uma política organizacional estabelecida e mantida para o processo), conforme mostra a Figura 2.

Há alguns resultados de atributos de processo que já são contemplados em pesquisas sobre padrões de processo, embora no contexto de processos específicos, e não de atributos de processos, como define a ISO 15504.

Nome:
Definição de Políticas Organizacionais
Problema:
Como as políticas organizacionais devem ser estabelecidas e mantidas por uma organização?
Forças:
Através das políticas organizacionais a organização (direção) indica quais são suas expectativas quanto à execução de um processo. Assim, após a institucionalização das políticas, os colaboradores sabem exatamente quais orientações eles devem cumprir, seja para um processo específico, seja com relação à organização como um todo.
Contexto Inicial:
Em iniciativas de melhoria de processo de software, a organização determina quais processos devem ser utilizados para seus projetos. Para uma efetiva definição ou evolução desses processos as políticas organizacionais deveriam ser estabelecidas ou atualizadas pela direção, ou por pessoas com devidas autorizações para este fim.
Solução:
Políticas devem ser definidas para cada processo utilizado na organização, pela direção ou por pessoas delegadas por ela. Em paralelo devem ser definidas políticas organizacionais que expressem princípios ou restrições aos colaboradores perante a organização.
Um planejamento periódico de revisão dessas políticas deve ser elaborado, para que fatores como mudanças estratégicas, ou alterações no contexto operacional da organização, entre outros aspectos, sejam contemplados nas políticas organizacionais.
Não basta a definição e o planejamento de revisões serem elaborados de maneira apropriada se os colaboradores da organização não tiverem conhecimento dessas políticas. Para que isso não ocorra, as políticas definidas devem ser publicadas em local de fácil acesso a todos, assim como devem haver ampla divulgação a todos os interessados.
Contexto Resultante:
As políticas organizacionais foram definidas pela direção, ou por pessoas delegadas para esse fim. Um plano de revisão periódica das políticas foi definido. As políticas foram institucionalizadas e são amplamente divulgadas na organização.
Padrões de Processos Relacionados:
Garantia da Qualidade – Deve assegurar que as políticas definidas estão sendo cumpridas em todos os processos.
Gerência de Projeto – Deve aplicar as políticas definidas no planejamento e no acompanhamento de atividades do projeto.

Figura 2. Padrão de Processo para o RAP 2 – Definição de Políticas Organizacionais

Por exemplo, o RAP 3 (A execução do processo é planejada) e o RAP 7 (As pessoas que executam o processo são competentes em termos de formação, treinamento e experiência), estão contemplados em padrões de processos relacionados à Gerência de Projetos [Välimäki 2009].

Portanto, não é necessária a definição de um novo padrão de processo específico para estes RAP, mas apenas a revisão dos padrões já propostos para avaliar a sua aderência aos propósitos do MR-MPS.

4. Conclusões

Este artigo discute uma área de pesquisa ainda pouco explorada pela comunidade científica, que é a definição de padrões de processos relacionados a atributos de processos. Uma investigação profunda sobre esta área está sendo conduzida como parte de um projeto de pesquisa conduzido pelos autores no Instituto de Informática da Universidade Federal de Goiás.

Pelos resultados já obtidos desse projeto, pode-se observar que padrões de processo podem contribuir significativamente para a área de pesquisa em melhoria de processos de software. Por exemplo, a estrutura definida para descrever um padrão de processo propicia àqueles que o utilizam um maior esclarecimento sobre aspectos importantes relacionados a um objetivo específico de melhoria de processo. Alguns desses aspectos são:

- A definição clara dos motivos para se realizar determinada melhoria;
- O momento em que se deve realizar tal melhoria;
- O que deve ser feito para solucionar o problema; e
- Qual é o contexto resultante após a aplicação da solução proposta pelo padrão.

Além disso, a investigação de padrões de processos sob o ponto de vista de atributos de processo busca diminuir as dificuldades recorrentes encontradas na implementação de atributos de processos em iniciativas de MPS.

A utilização da nomenclatura utilizada no MR-MPS e na ISO 15504, como Atributos de Processo, e Resultado de Atributo de Processo, não impede que a abordagem e os resultados aqui propostos sejam utilizados em outros contextos de MPS. Por exemplo, em iniciativas de MPS baseadas em CMMI poderiam ser utilizados os objetivos genéricos deste modelo para estabelecer a relação com os atributos de processo.

O objetivo da pesquisa que está sendo desenvolvida é que não haja limitações para a utilização dos padrões de processo definidos, ou seja, todos os padrões de processo elaborados podem ser utilizados independentemente do modelo de processo adotado.

No estágio atual dessa pesquisa está sendo conduzido um estudo de caso para avaliar preliminarmente a aplicação da solução proposta, ou seja, a contribuição de padrões de processo para a implementação e a institucionalização de atributos de processo. Concorrentemente, o projeto continua identificando e formalizando novos padrões de processo aplicáveis a atributos de processo.

Bibliografia

Ambler, Scott. W. *Software Process Patterns*. Cambridge University Press (1998).

Ambler, Scott. W. *More Software Process Patterns*. Cambridge University Press (1999).

Asadi, Mohsen, and Ramsin, Raman. Method engineering process patterns. *Proceedings of the 2nd India Software Engineering Conference (ISEC)*, 2009, 143-144.

Brietzke, Josiane, and Rabelo, Abraham. Resistance Factors in Software Process Improvement. *CLEI Electronic Journal*, 9(1):Artigo4, 2006.

Coplien, James O.. A Generative Development Process Pattern Language. In Linda Rising Editor. *The Patterns Handbook: Techniques, Strategies, and Applications*, pp. 243-300, Cambridge University Press, 1998.

Fernandes, Patrícia G., and de Oliveira, Juliano L.. Perùl Cultural e Institucionalização de Processos de Software: Estudo de Caso em Duas Organizações de Software. In: VI Workshop Olhar Sociotécnic sobre a Engenharia de Software, Belém-PA. *Anais do IX Simpósio Brasileiro de Qualidade de Software*. SBC, 2010.

ISO/IEC - International Organization for Standardization/ International Electrotechnical Commission. *ISO/IEC 15504-1: Information Technology - Process Assessment – Part 1 - Concepts and Vocabulary*, Geneve: ISO, 2004.

ISO/IEC - International Organization for Standardization/ International Electrotechnical Commission. *ISO/IEC 12207: Systems and Software Engineering - Software Life Cycle Processes*, Geneve: ISO, 2008.

Mendes, Fabiana F., de Oliveira, Juliano L., Fernandes, Patrícia G., and de Souza, Adriana S.. Análise de Riscos na Implantação de Melhorias de Processos de Software. *Proqualiti – Qualidade na Produção de Software*, 3(3):25–31, 2007.

Rocha, Ana R., Montoni, Mariano, Santos, Gleison, *et. al.* Fatores de Sucesso e Dificuldades na Implementação de Processos de Software Utilizando o MR-MPS e o CMMI. *Proqualiti – Qualidade na Produção de Software*, 1(2):13–18, 2005.

Softex, Associação para Promoção da Excelência do Software Brasileiro. *MPS.BR – Guia Geral - 2009*, Disponível em: <http://www.softex.br/mpsbr>, setembro de 2009.

Tran, Hanh N., Coulette, Bernard, and Dong, Bich T.. A UML based process metamodel integrating a rigorous process patterns definition. In: *PROFES 2006, Lecture Notes in Computer Science*, 4034, 2006.

Travassos, Guilherme H., and Kalinowski, Marcos. *iMPS 2009 - Caracterização e Variação de Desempenho de Organizações que Adotaram o Modelo MPS*. Campinas, SP: SOFTEX, 2009. 27 p.

Välimäki, Antti, Kääriäinen, Jukka, and Koskimies, Kai. Global software development patterns for project management. *Communications in Computer and Information Science*, 2009, 42, 137-148.

Washizaki, Hironori, Kubo, Atsuto, Takasu, Atsuhiko, and Fukazawa, Yoshiaki. Relation analysis among patterns on software development process. *Lecture Notes in Computer Science*, 2005, 3547, 299-313.

Uso do Ambiente WebAPSEE na Implementação do Nível G do MPS.BR na Equilibrium Web

Ernani Sales^{1,2}, Luciana Nascimento¹, Sebastião Antônio Farias Júnior³,
Carla Lima Reis^{1,2}, Rodrigo Quites Reis^{1,2}

¹Laboratório de Engenharia de Software (LABES) - Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA) – Belém – PA – Brasil

²QR Consultoria e Serviços LTDA – Belém – PA – Brasil

³Equilibrium Web LTDA – Belém – PA – Brasil

{ernani,luciana,clima,quites}@webapsee.com; sebastiao@equilibriumweb.com

Abstract. *WebAPSEE environment is one of the current alternatives to support Software Process Management concerning current standards and quality models. This paper shows a successful experience about the use of this PSEE on the MPS.BR level G implementation at Equilibrium Web company. The observed positive aspects and improvement opportunities are related as well as some lessons learned.*

Resumo. *O ambiente WebAPSEE é uma das alternativas atuais para apoiar a gerência de processos de software com objetivo de atender padrões e modelos de qualidade existentes. Este artigo apresenta uma experiência de sucesso na utilização dessa ferramenta na implementação do nível G do MPS.BR na empresa Equilibrium Web. Os pontos positivos e as oportunidades de melhoria do uso desse ambiente bem como as lições aprendidas nessa experiência são relatados.*

1. Introdução

A melhoria de processos de software em empresas brasileiras tem ocorrido cada vez mais em consonância com o modelo de Melhoria do Processo de Software Brasileiro (MPS.BR) [Softex 2009]. Com 233 empresas avaliadas em todo Brasil, esse modelo tem cumprido as suas metas estabelecidas e contribuído com o aumento da qualidade de software e com o uso de ferramentas especializadas.

Segundo Travassos e Kalinowski (2008; 2009), os resultados de desempenho de organizações que adotaram o MPS.BR indicam que estas empresas alcançaram maior satisfação dos seus clientes, maior produtividade, capacidade de desenvolver projetos maiores e satisfação com o MR-MPS. Contudo, um número muito pequeno de empresas adotou o modelo na região Norte do Brasil.

A IOGE (Instituição Organizadora de Grupos de Empresas) RioSoft organizou um grupo de empresas em 2009 do qual participou a empresa paraense Equilibrium Web. A QR Consultoria, Instituição Implementadora sediada em Belém-PA foi a responsável pela consultoria de implementação na Equilibrium Web. Essa iniciativa de melhoria dos processos e produtos da Equilibrium Web teve como meta a implantação do nível G do modelo, além de verificar a utilização do ambiente WebAPSEE Pro [Lima Reis e Reis 2007], ferramenta desenvolvida pela QR Consultoria para apoio a gestão de processos de software.

Este artigo apresenta a experiência de implantação do nível G do MPS.BR na Equilibrium Web sob o

ponto de vista do uso da ferramenta, focando em lições aprendidas e procurando fornecer subsídios para discussões em torno de tecnologia de processos de software para implantação do modelo MPS.BR.

Na seção 2, a articulação para o investimento em melhoria de processos na Equilibrium Web é descrita, bem como a estratégia de implementação utilizada na implementação do nível G do MPS.BR. Na seção 3 é apresentado resumidamente o ambiente WebAPSEE. Na seção 4 é detalhado o foco principal do artigo, indicando como a ferramenta foi utilizada em termos dos dois processos avaliados: Gerência de Projetos e Gerência de Requisitos. Na seção 5, uma análise sobre o uso do ambiente WebAPSEE na implementação do nível G na Equilibrium Web é apresentada. Na seção 6 são apresentadas as lições aprendidas e na seção 7 as considerações finais do artigo.

2. Melhoria de Processos de Software na Equilibrium Web

Nesta seção é apresentada a empresa Equilibrium Web com suas particularidades e a estratégia de implementação utilizada durante a melhoria de processos da empresa.

2.1. Empresa

A Equilibrium Web é uma empresa de pequeno porte fundada em 2004 que atua como provedora de soluções em Tecnologias da Informação (serviços cliente/servidor, web, outsourcing, entre outros), sendo especialista em desenvolvimento de websites e sistemas web, além de ministrar treinamentos. A estrutura organizacional é formada por uma Diretoria a qual estão subordinadas três divisões: Gerência de Treinamentos, Gerência de Sites Web e Gerência de Desenvolvimento de Software.

A implementação do nível G do modelo MPS ocorreu na divisão Gerência de Desenvolvimento de Software, que é formada por 8 pessoas que exercem os papéis de Gerente de Projeto, Analista de Requisitos, Revisor, Desenvolvedor e Testador (alguns exercem mais de um papel).

2.2. Estratégia de Implementação

A implantação de processos de software em uma organização é fortemente baseada em conhecimento. Para a implantação de uma iniciativa de melhoria, é fundamental que os responsáveis por executar os processos sejam capacitados. Além disso, uma iniciativa de melhoria é normalmente considerada cara por muitas organizações, pois são necessários recursos significativos durante certo período de tempo. Desta forma, torna-se necessário desenvolver estratégias efetivas para implementar processos de software com sucesso [Montoni *et al* 2008].

Especificamente, no projeto de implementação do nível G do MR-MPS na Equilibrium Web foi utilizada a estratégia de implementação da QR Consultoria. Tal estratégia é baseada no modelo de implantação IDEAL [Mcfeeley 1996] e de Covre *et al* (2008), com a definição de cinco estágios (iniciação, diagnóstico, estabelecimento, ação e aprendizagem). Possui procedimentos, *templates* de documentos, material de treinamento, entre outros voltados para a implementação do MR-MPS. Além disso, essa estratégia tem como ferramental de apoio o ambiente WebAPSEE [Lima *et al* 2006]. Tanto a metodologia quanto a ferramenta foram desenvolvidos pela QR Consultoria.

3. Ambiente WebAPSEE

O WebAPSEE¹ permite a modelagem (visual) e execução de processos de software, e tem na **flexibilidade** de definição e execução (modificações *ad-hoc* no processo em tempo de execução) o seu principal diferencial. Esse ambiente evoluiu em termos de funcionalidades com o apoio à reutilização de processos [Costa et al. 2007], controle de versões de artefatos [Sales et al. 2007], gerência de requisitos [Sales et al. 2010] e gerência de conhecimento [Oliveira et al., 2010]. Além destas, outras funcionalidades características de ferramentas de gerenciamento de projetos estão disponíveis, como visualização do Gráfico de Gantt, caminho crítico, Estrutura Analítica do Projeto e emissão de relatórios gerenciais, como acompanhamento de atividades com desvio de esforço e prazo, plano de custos e recursos humanos, dentre outros.

Atualmente, o WebAPSEE é implantado com um servidor, que centraliza o repositório de processos e serviços de modelagem, execução, reutilização, gerência de artefatos e acesso a dados, e dois tipos de clientes: o WebAPSEE *Manager Console* e a *WebAPSEE Task Agenda*. As figuras 1 e 2 mostram os componentes citados enquanto a figura 3 mostra um exemplo de relatório de acompanhamento de um projeto na ferramenta.

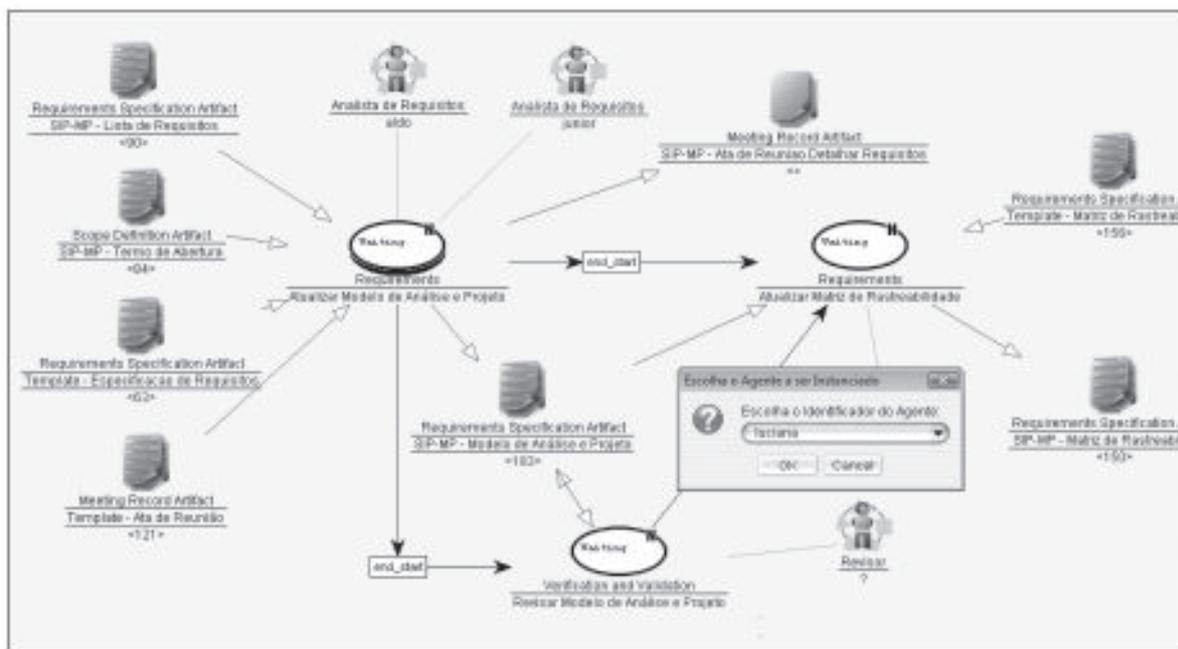


Figura 1. Funcionalidades do ambiente WebAPSEE – Manager Console – Modelagem de processos

¹ O WebAPSEE é um ambiente de desenvolvimento de software centrado em processos que possui uma versão livre (www.webapsee.com) e uma versão proprietária com maior número de funcionalidades licenciada pela empresa QR Consultoria (www.qrconsult.com.br).

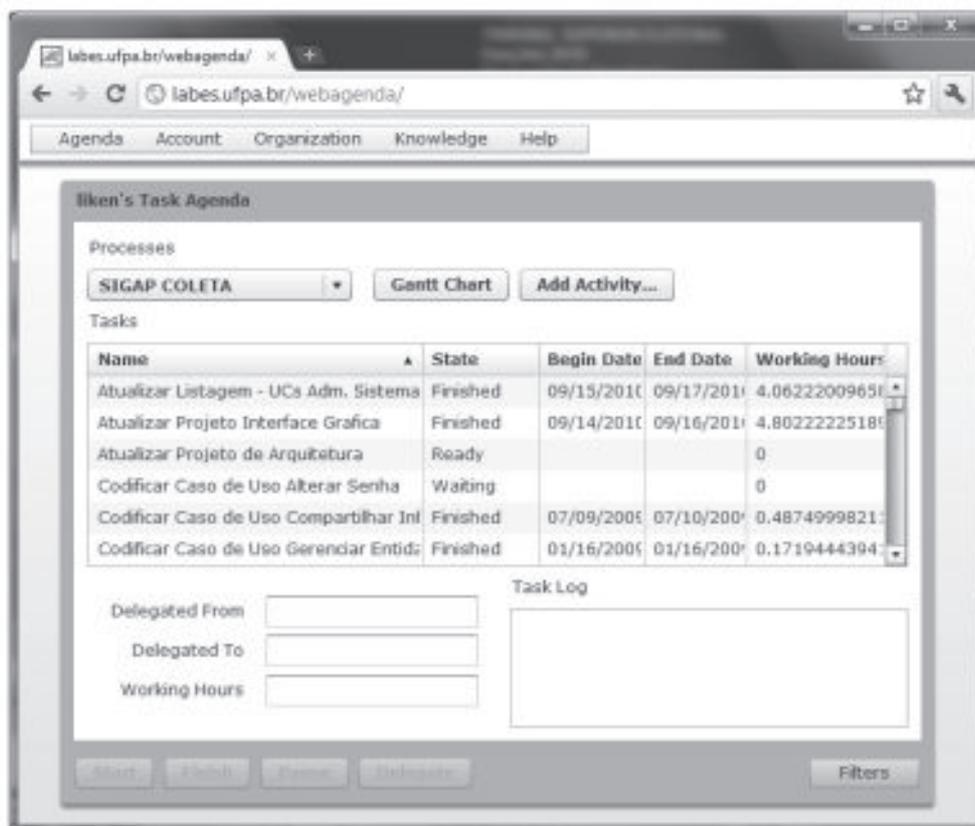


Figura 2. Agenda de atividades do WebAPSEE – WebAgenda

relatório de acompanhamento (Modo de Computador) - Microsoft Excel

03/10/2018 20:10:02

Relatório de Acompanhamento de Atividades

Processo: SIGAP COLETA

Atividade	Estado	Início Planejado	Término Planejado	Estimativa de Horas	Início Real	Término Real	Horas Trabalhadas	Desvio de Me
Subsistema de Manual-DI	Waiting			0,00			0,00	
Projeto de Arquitetura - Subsistema de Manual	Waiting			0,00			0,00	
Construção e Testes - Subsistema de Manual-DI	interrupted			0,00			0,00	
Planejar Testes - Subsistema de Manual				0,00			0,00	
Desenvolver e Testar - Subsistema de Manual-DI	interrupted			0,00			0,00	
Desenvolver Documentação Preliminar do Usuário	Waiting			0,00			0,00	
Revisar Manual de Usuário	Waiting			0,00			0,00	
Integração - Controle de Acesso	Waiting			0,00			0,00	
Realizar Matriz de Restrições	Waiting			0,00			0,00	
Controlar - Controle de Acesso-DI	interrupted			0,00			0,00	
Codificar Caso de Uso Solicitar Nova Senha	Waiting			0,00			0,00	
Codificar Serviços de Acesso ao Lattes	Waiting			0,00			0,00	
Testar Caso de Uso Alterar Senha	Waiting			0,00			0,00	
Codificar Caso de Uso Solicitar Acesso ao Sistema	Waiting			0,00			0,00	
Testar Caso de Uso Solicitar Nova Senha	Waiting			0,00			0,00	
Testar Caso de Uso Solicitar Acesso ao Sistema	Waiting			0,00			0,00	
Codificar Caso de Uso Alterar Senha	Waiting			0,00			0,00	
Testar Caso de Uso Realizar Login	Waiting			0,00			0,00	
Testar Caso de Uso Realizar Logout	Waiting			0,00			0,00	
Codificar Caso de Uso Realizar Login	Waiting			0,00			0,00	
Codificar Caso de Uso Realizar Logout	Waiting			0,00			0,00	
Testes Finais - Controle de Acesso	Waiting			0,00			0,00	
Revisar o Plano de Projeto - SIGAPCO	Waiting			0,00			0,00	
Planejar Integração - Subsistema de Manual	Finished	10/09/2018	10/09/2018	0,00	10/09/2018	20/09/2018	71,00	36
Análise dos Requisitos e Projeto de GUI - Subsistema de	Finished	10/09/2018	11/09/2018	0,00	20/09/2018	23/09/2018	0,00	
Subsistema de Manual - Elaboração Lista de Requisitos	Finished	10/09/2018	11/09/2018	1,00	20/09/2018	23/09/2018	0,11	
Subsistema de Manual - Elaboração Lista de Requisitos	Cancelled	20/09/2018	30/09/2018	1,00	30/09/2018	23/09/2018	0,00	

Figura 3. Exemplo de Relatório obtido na ferramenta – Acompanhamento de atividades

No *Manager Console*, o gerente e o engenheiro de processos podem cadastrar informações organizacionais (sobre pessoas, recursos de apoio, cargos/papéis, habilidades, entre outras), modelar processos, acompanhar e interagir (visualmente) com a execução dos processos, alocar pessoas e recursos, gerar relatórios e assim por diante. Já a *Task Agenda* é o aplicativo cliente utilizado pelos desenvolvedores, através do qual estes podem visualizar suas atividades (com estados e instruções de execução), acessar os artefatos necessários para a execução de cada tarefa e enviar os artefatos produzidos em cada atividade, além de fornecer *feedback* sobre o andamento de suas tarefas.

4. Implantação da ferramenta

Nesta seção o uso da ferramenta é caracterizado com relação aos resultados esperados do nível G do MPS.BR. Cabe ressaltar que os envolvidos no processo utilizaram necessariamente a *Task Agenda* para registrar o andamento de suas tarefas. Isso foi primordial para a correta utilização dos relatórios de monitoramento da ferramenta.

4.1. Gerência de Projetos

As principais fontes de evidência para os resultados esperados do processo de Gerência de Projeto (GPR) com a utilização da ferramenta são:

- **Modelo do Processo:** a ferramenta permitiu a descrição visual do processo sendo executado para um projeto contendo (ver Figura 4):
 - as atividades e seus respectivos estados de execução e dependências;
 - os artefatos de entrada e saída por atividade, o que permite visualização e manipulação eficiente do plano de gerência de dados e de comunicação;
 - as pessoas (e seus papéis) alocadas em atividades, o que permite visualizar e manipular o plano de recursos humanos;
 - os recursos de apoio alocados para cada atividade (com quantidades e valores quando pertinente), o que permite visualizar e manipular o plano de recursos de apoio.
- **Relatórios gerenciais** que evidenciam o planejamento e a situação do projeto para fins de monitoração;
- **Métricas coletadas** durante a execução do processo para acompanhamento, como, por exemplo, o total de horas trabalhadas por pessoa em cada tarefa;
- **Gerência do acesso** controlado aos artefatos do processo.

Atividades específicas de GPR ocorrem em diferentes momentos da execução do processo, em marcos e situações de exceção. Na descrição do processo é possível determinar quando as atividades de planejamento devem ocorrer e quais são os marcos do projeto. Além disso, os planos de ação podem ser modelados como *templates* de processos [Costa *et al* 2007] que são instanciados pelo gerente como fragmentos do processo quando uma exceção ocorre na execução do processo.

Como exemplo de implementação de resultados esperados, a gerência de dados é realizada através do Plano de Gerência de Dados: um relatório gerado a partir de informações extraídas do processo instanciado, onde é possível saber **quem** utiliza, **qual** artefato, **em qual** atividade e qual **artefato** será produzido. Além disso, esses artefatos ficam sob controle da ferramenta através da integração com uma ferramenta de controle de versões (externa, CVS ou SVN), sendo que os usuários terão acesso aos artefatos através da *Task Agenda*, acessada através de senha com permissão para leitura e escrita de artefatos dependente da situação do processo em andamento.

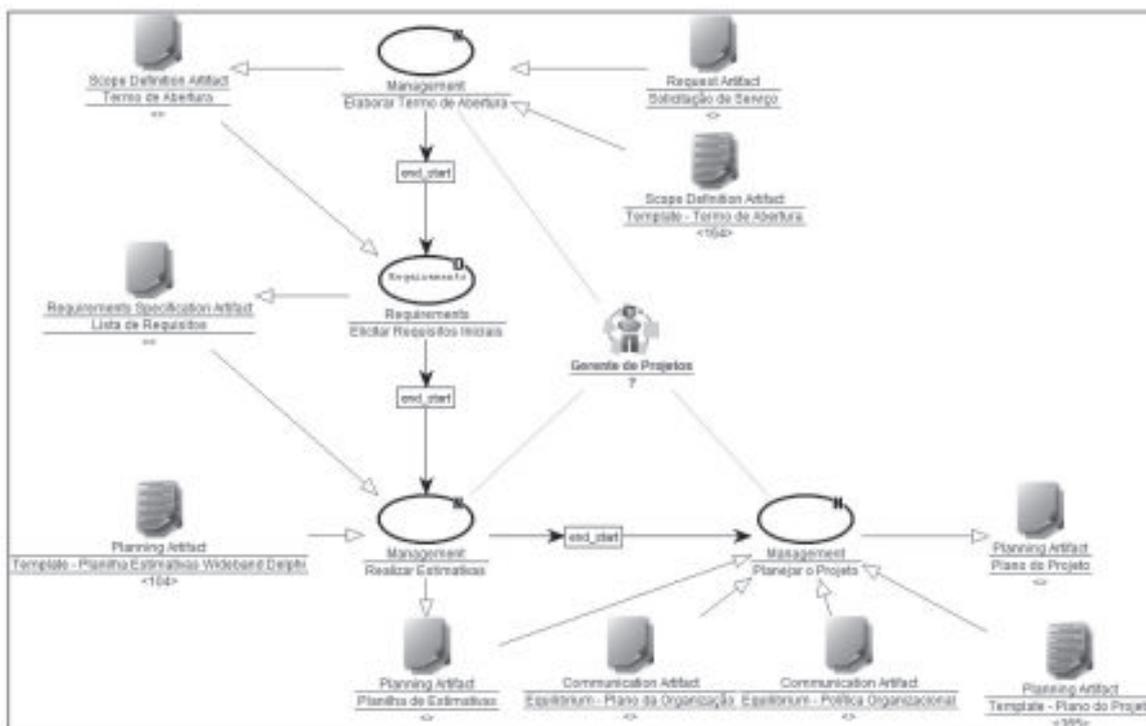


Figura 4. Parte do modelo de processo adotado na Equilibrium Web

Outro exemplo de resultado esperado é a monitoração do projeto através do acompanhamento visual da execução do processo e do relatório de acompanhamento de atividades com desvio de esforço e prazo. Em tais casos, relatórios de cronograma com desvio de prazo e geração do gráfico de Gantt do projeto puderam evidenciar atrasos.

4.2. Gerência de Requisitos

Da mesma forma que ocorre com o processo GPR, o processo de Gerência de Requisitos (GRE) é modelado de forma integrada ao processo do projeto em andamento. Apesar de GRE ter suas atividades ocorrendo em momentos pré-definidos, existem também atividades que ocorrem em situações que dependem de eventos externos, como uma solicitação de mudança nos requisitos, por exemplo.

Na ocorrência de uma mudança de requisitos, as atividades do processo de mudança são controladas e acompanhadas na ferramenta da mesma forma que o fluxo normal do processo. Isto significa que os artefatos impactados, as pessoas e recursos alocados durante a mudança também são visualizados como um trecho do processo.

No período da implantação do nível G na Equilibrium Web, o WebAPSEE ainda não possuía uma ferramenta que apoiasse a rastreabilidade de Requisitos integrada ao ambiente. Dessa forma, foram utilizados apenas *templates* para guiar a construção de artefatos como a Matriz de Rastreabilidade, Solicitação de Mudança de Requisitos, Avaliação da Mudança de Requisitos, entre outros. Entretanto, o ambiente permite estabelecer a dependência entre os artefatos através da definição dos relacionamentos de derivação e pertinência entre artefatos, além da ordem em que os artefatos são consumidos e produzidos pelas atividades na descrição visual do processo. A partir de setembro de 2010 a ferramenta integrada de gerência de requisitos [Sales et al. 2010] começou a ser usada na empresa.

5. Aspectos coletados sobre o uso do WebAPSEE na implementação do Nível G do MPS.BR

A partir da experiência de utilização da ferramenta tanto por parte dos usuários, funcionários (Gerentes de Projeto, Analistas, Desenvolvedores) da Equilibrium Web, quanto dos consultores de implementação da QR Consultoria, pode-se levantar aspectos relevantes para o processo de implementação de melhoria de processos realizado (Ver tabela 1). Tais aspectos não se restringem apenas à Gerência de Projetos e Gerência de Requisitos visto que o ambiente WebAPSEE apoia diversas áreas da Gerência de Processo de Software.

Tipo	Descrição	Fonte
Pontos Positivos	Definição dos processos (tarefas, prazos, agentes, recursos, etc.) através de uma linguagem visual	Gerentes de Projeto Equipe de Implementação
	Funcionalidades que permitem criar <i>templates</i> de processos de desenvolvimento de software e instanciá-los nos projetos da organização (reutilização de processos)	Gerentes de Projeto
	Possibilidade de modificação do fluxo de execução dos processos mediante eventos significativos (atrasos, mudanças nos requisitos, etc.)	Gerentes de Projeto
	Facilidade para cadastrar e recuperar as evidências dos processos de GPR e GRE	Equipe de Implementação
	Relatórios gerados automaticamente facilitam a monitoração dos prazos do projeto	Gerentes de Projeto
	Armazenamento, controle de acesso e de versão dos artefatos gerados durante a execução dos projetos	Gerentes de Projeto Equipe de Implementação
	Acesso às atividades diárias para as quais se está alocado	Desenvolvedores
Oportunidades de Melhoria	Ferramenta para GRE que auxilie na definição, consulta e atualização da rastreabilidade entre requisitos	Equipe de Avaliação
	Melhoria nas interfaces de cadastro do ambiente quanto à Usabilidade	Gerente de Projetos

Tabela 1. Pontos Positivos e Oportunidades de Melhoria no Uso do WebAPSEE na implementação do Nível G do MPS.BR na Equilibrium Web

6. Discussão sobre as Lições Aprendidas

Algumas situações ocorridas durante a experiência contribuíram para a melhoria da ferramenta ou para melhoria da estratégia de implementação da equipe. A seguir são apresentados os casos que levaram a algumas lições aprendidas do ponto de vista do uso da ferramenta na estratégia de implementação MPS.BR:

- O comprometimento da equipe e o entendimento dos objetivos pretendidos com a iniciativa de melhoria de processos, além da alta coesão dos funcionários da Equilibrium Web, foram fundamentais para a implementação das mudanças na forma de trabalho dos envolvidos nos projetos e para superar a dificuldade inicial em se adotar um PSEE [Fuggetta 2000] em um ambiente organizacional;
- O processo definido (e ajustado durante os projetos) foi seguido à risca com o uso da ferramenta, podendo ser auditado através do *log* de eventos. A equipe percebeu que não possui apenas um guia de processo para consulta na web, e sim a coordenação das etapas do processo a partir do *feedback* fornecido. Do ponto de vista da consultoria de implementação houve facilidade em acompanhar o andamento dos projetos de forma não presencial;
- Como esta é a segunda avaliação oficial de empresa que utilizou a ferramenta, a equipe de implementadores optou por criar *templates* de documentos que traziam algumas redundâncias em relação ao conteúdo da ferramenta. Tais redundâncias faziam com que o gerente gerasse um relatório no WebAPSEE e copiasse o conteúdo do mesmo para os *templates* fornecidos. Isso gerou uma sensação maior de segurança para efeito de avaliação. Nas próximas experiências pretende-se usar mais as evidências fornecidas pela ferramenta para efeito de avaliação;
- A falta de uma ferramenta específica de apoio a rastreabilidade de requisitos dificultou o andamento de alguns projetos, pois o preenchimento da planilha trouxe atrasos durante o aprendizado;
- A criação de planos de ação para solucionar problemas durante os projetos foi agilizada devido a monitoração constante fornecida pela ferramenta, obtida a partir do *feedback* da equipe. Em vários casos, o marco atingido serviu apenas pra checar e documentar para a próxima fase se todos os problemas estavam resolvidos. Ou seja, o gerente não necessitava reunir-se com tanta frequência para ficar ciente dos problemas.

7. Considerações Finais

Este artigo descreveu a implementação do MR-MPS em uma empresa paraense utilizando o ambiente WebAPSEE. A partir dessa experiência foi possível analisar as situações ocorridas e transformá-las em lições aprendidas tanto para desenvolvimento da ferramenta quanto para estratégia de implementação.

A empresa foi avaliada em junho de 2010 e obteve o nível pretendido com o uso do ambiente WebAPSEE. As lições aprendidas com a experiência são importantes para melhorar a estratégia de implementação adotada e estão gerando novas demandas de funcionalidades para a ferramenta que

atendam melhor o nível G e também os níveis superiores. Como exemplo, a ferramenta de gerência de requisitos integrada ao Webapsee já está sendo adotada na empresa.

Atualmente, a empresa Equilibrium Web utiliza a ferramenta para gerenciar todos os seus projetos, seja para construção de software ou outros produtos da empresa (sites, diagramação de layouts, etc.), e está no planejamento da iniciativa de melhoria para alcançar o nível F do MPS.BR.

Referências

Costa, A.; Sales, E.; Lima Reis, C.; Reis, R. (2007) "Apoio a Reutilização de Processos de Software através de Templates e Versões", In: 6º Simpósio Brasileiro de Qualidade de Software (SBQS'07). Porto de Galinhas, PE, Brasil.

Covre, V.; Lima Reis, C.; Favero, E. (2008) "Metodologia para Implementação do MPS.BR Utilizando o Ambiente WebAPSEE", In: 7º Simpósio Brasileiro de Qualidade de Software (SBQS'08). Florianópolis, SC, Brasil.

Fuggetta Alfonso. (2000) "Software Process: a roadmap", In: Conference on the future of Software engineering (ICSE '00). Limerick, Ireland.

LIMA, Adailton Magalhães; FRANÇA, Breno Bernard Nicolau de; COSTA, Anderson; REIS, C. A. L. . Gerência Flexível de Processos de Software com o Ambiente WebAPSEE. In: Simpósio Brasileiro de Engenharia de Software - Sessão de Ferramentas, 2006, Florianópolis. XIII Sessão de Ferramentas do SBES. Florianópolis : SBC, 2006. p. 97-102.

Lima Reis, C. A.; Reis, R. Q. (2007) Laboratório de Engenharia de Software e Inteligência Artificial: Construção do ambiente WebAPSEE. In *ProQuality* (UFLA), v. 3, p. 43-48.

Lima, Liken; Das Dores, Silvia N., Oliveira, Jadielly; Sales, Ernani; Andrade, Gabriela; Lima Reis, Carla A. (2010) "Ferramenta de Apoio à Gerência de Conhecimento Integrada a um Ambiente de Desenvolvimento de Software Centrado em Processos". In: Sessão de Ferramentas do I Congresso Brasileiro de Software (CBSOFT 2010). Salvador, BA, Brasil.

Mcfeeley, B. (1996), IDEALSM: A User's Guide for Software Process Improvement. Carnegie Melon University, Software Engineering Institute, Pittsburgh.

Montoni, M.; Santos, G.; Katsurayama, A.; Cabral, R.; Figueiredo, S.; Natali, A. (2008) "Aplicação da Estratégia SPI-KM para Apoiar a Implementação do MPS.BR Níveis G e F em Pequenas e Médias Empresas do Rio de Janeiro", In: 7º Simpósio Brasileiro de Qualidade de Software (SBQS'08). Florianópolis, SC, Brasil.

Softex (2009) "MPS.BR – Melhoria de Processo do Software Brasileiro – Guia Geral", ISBN 978-85-99334-15-7, setembro.

Sales, E.; Lima Reis, C.; Lima, A. (2007) "Gestão de Configuração integrada a Gerência de Processos

de Software no Ambiente WebAPSEE”, In: 33ª Conferência Latino Americana de Informática (CLEI’07). San José, Costa Rica.

Sales, Murilo; Sales, Ernani; Lima Reis, Carla; Reis, Rodrigo (2010) “Uma Ferramenta de Gerência de Requisitos Integrada a um Ambiente de Desenvolvimento de Software Centrado em Processos”. In: Sessão de Ferramentas do I Congresso Brasileiro de Software (CBSOFT 2010). Salvador, BA, Brasil.

Travassos, G.; Kalinowski, M. (2008) “iMPS: resultados de desempenho de organizações que adotaram o modelo MPS”, Relatório Técnico. Campinas, SP, Brasil.

Travassos, G.; Kalinowski, M. (2009) iMPS 2009 – “Caracterização e Variação de Desempenho de Organizações que Adotaram o Modelo MPS”. Campinas, SP, Brasil: SOFTEX, 2009. 27 p.

Aplicação de Jogos Educativos para Aprendizagem em Melhoria de Processo e Engenharia de Software

Marcello Thiry¹, Alessandra Zoucas¹, Rafael Q. Gonçalves¹, Clenio Salviano²

¹LQPS – Laboratório de Qualidade e Produtividade de Software/CTMAR– Universidade do Vale do Itajaí (UNIVALI)

Rod. SC 407, Km 04 - 88122-000– São José – SC – Brazil

thiry@univali.br, alessandra.zoucas@gmail.com, rafael.q.g@hotmail.com

²Centro de Tecnologia da Informação Renato Archer (CTI)

Divisão de Melhoria de Processo e Qualidade de Software (DMPQS)

Rodovia D. Pedro I, km 143.6, Campinas – SP - Brasil

clenio.salviano@cti.gov.br

Abstract. *This paper presents how games can help the coaching of professionals in concepts related to Process Improvement and Software Engineering. It also presents three educational games that focus on areas of Software Process Improvement. The paper shows the results of experiments conducted with over 40 students, to prove the efficacy of these games for teaching.*

Resumo. *Este artigo apresenta como jogos podem auxiliar na capacitação de profissionais em conceitos relacionados à Melhoria de Processo e Engenharia de Software. São apresentados três jogos educativos que focam em áreas da Melhoria de Processo de Software. O trabalho traz o resultado de experimentos conduzidos, com mais de 40 alunos, para comprovar a eficácia destes jogos para o ensino.*

1. Introdução

Pesquisas têm demonstrado que uma alternativa viável para a melhoria das empresas é a participação em projetos específicos ou cooperados de implementação de Melhoria de Processo de Software (MPS) alinhados a modelos de referência como o MPS.BR [Softex, 2009] e CMMI-DEV [SEI, 2006]. As informações para acompanhar e evidenciar a variação de desempenho nas empresas que adotaram o Modelo MPS.BR obtidos pelo projeto iMPS apontam que estas empresas alcançaram maior satisfação de seus clientes, além de maior produtividade e capacidade em desenvolver projetos maiores [Travassos e Kalinowski, 2008]. Dados coletados em 2009 pela segunda rodada do iMPS mostraram que as empresas relatam ter alcançado o retorno do investimento e que há uma tendência a melhoria de custo, qualidade, prazo e produtividade, principalmente para aquelas empresas que evoluíram ou internalizaram o modelo MPS.BR em seus processos [Travassos e Kalinowski, 2009].

Entretanto, ainda é atual a discussão sobre os fatores de sucesso na implementação de melhoria de processo de software. Um dos principais fatores, centro desta discussão, permanece sendo a falta de capacitação em Engenharia de Software (ES) da equipe das organizações envolvidas nos projetos de melhoria de processos. Em 2005, já se discutia a necessidade de realizar treinamentos em tópicos como engenharia de requisitos, planejamento de projetos, entre outros conteúdos de ES para nivelar as competências dos recursos humanos da organização alvo da melhoria de processos [Rocha et al., 2005].

Em outubro de 2008, foram publicadas lições aprendidas nos projetos de melhoria de processo de

software, onde uma consistia em estabelecer e implementar estratégias adequadas de capacitação dos envolvidos na iniciativa de melhoria [Montoni, 2008]. Ainda, segundo Montoni (2008), a transferência de conhecimento sobre métodos e técnicas de ES tem um impacto direto tanto no tempo, quanto no custo das iniciativas de melhoria. A lição aprendida supracitada está diretamente relacionada a uma possível causa levantada sobre a deficiência na capacitação das equipes das empresas em ES, que é o nível muito baixo de formação em informática (universidades fracas e maioria de não graduados) [Rocha, 2005]. Um agravante para esta dificuldade é o fato das iniciativas de treinamento em métodos e técnicas utilizadas como apoio ao processo de desenvolvimento muitas vezes possuem um enfoque por demais teórico e raramente envolve os estudantes em projetos reais [Romero; Vizcaino; Piattini 2008].

Contudo, há uma abordagem à prática no ensino de ES que se refere à aplicação de jogos educativos [Navarro, 2006; Benitti e Molléri, 2008; Ludewig, 2003; Wangenheim et al, 2008]. Os jogos, de maneira geral, compreendem um conjunto de atividades que podem envolver um ou mais participantes, e comumente desafiam as habilidades físicas ou mentais dos jogadores, exigindo que estes sigam um conjunto específico de regras para atingir um determinado objetivo, previamente estabelecido [Dempsey et al, 1993]. É possível verificar na literatura a pequena quantidade de experimentos realizados com o intuito de verificar a efetividade dos jogos no apoio ao ensino de ES [Wangenheim et al, 2008]. Além disso, há pesquisas na área de treinamento e educação que sugerem o uso de jogos no ensino, pois estes podem engajar o estudante, reforçando conceitos através da prática, e aprofundando os conhecimentos [El-shamy 2008]. O envolvimento emocional de um estudante aumenta conforme o entretenimento, levando a uma variação de estímulos que o ajuda a reter novos conhecimentos, mas para isto, aulas palestradas não são suficientes [Millbower 2003, apud Alexander e Beatty, 2008].

Neste sentido, o LQPS (Laboratório de Qualidade e Produtividade de Software), da Univali/SC, em parceria com a Instituição Implementadora MPS.BR Incremental Tecnologia, tem trabalhado na elaboração de jogos educativos como estratégia de ensino e aprendizagem de conceitos e técnicas de ES. Conceitos estes diretamente relacionados às práticas dos principais modelos de capacidade da maturidade de processo utilizados como referência para melhoria de processo de software, ou seja, conceitos aplicados na interpretação e implementação de tais modelos. Desta forma, os conceitos considerados abrangem práticas destes modelos e o ensino/aprendizagem pode ser considerado tanto como de ES, quanto dos modelos ou ainda ambos. Até o momento, três jogos educativos já foram elaborados: (1) "A Ilha dos Requisitos"; (2) "U-TEST" e (3) "X-MED". Estes jogos fazem uso de aspectos lúdicos, desafios, dicas e rápido *feedback* ao jogador, com o objetivo de contribuir para a compreensão das práticas destes processos de desenvolvimento de software. A avaliação da efetividade de aprendizagem dos três jogos para o ensino foi realizada através de experimentos controlados pela equipe do LQPS que envolveu alunos de graduação na área da Computação. Estes alunos forneceram dados para a avaliação dos jogos, tanto de forma quantitativa quanto qualitativa.

Este artigo apresenta a aplicação de jogos educativos como estratégia de capacitação de pessoas em conceitos relacionados à ES e MPS e está organizado em quatro seções: a seção 2 descreve a seleção e a análise de alguns trabalhos correlatos; a seção 3 mostra as características dos jogos educativos desenvolvidos, bem como os resultados obtidos com a avaliação da eficácia dos jogos para o ensino; e, a última seção apresenta considerações sobre o trabalho desenvolvido e aponta para pesquisas futuras.

2. Trabalhos Relacionados

Para identificar trabalhos relacionados com a pesquisa aqui apresentada, foram realizadas sessões de revisão sistemática da literatura. Estas revisões foram conduzidas por pesquisadores de ES da UFSC e da Univali. Numa destas revisões, buscou-se identificar todos os tipos de jogos ou simuladores com objetivos educacionais, incluindo jogos de computador ou não (p.ex: jogos de tabuleiro, cartas, etc.) [Wangenheim e Shull, 2009]. Como não foi encontrado um conjunto relevante de jogos para ensino de ES, especialmente desenvolvidos no Brasil, outra revisão sistemática da literatura foi realizada ampliando o domínio de buscas para a Web em geral, visando encontrar um maior número de estudos que abordam jogos desenvolvidos no Brasil e publicados em Português. Como resultado destas pesquisas foi identificado que ainda existem poucas publicações sobre jogos educacionais para ensino de ES no Brasil e entre as áreas da ES, a mais abordada pelos jogos é a de Gerenciamento de Projetos. Em relação à eficiência educacional dos jogos brasileiros para ES a análise aponta que a grande maioria das publicações não apresenta detalhamento suficiente sobre uma avaliação dos efeitos de aprendizagem dos jogos [Wangenheim, Kochanski e Savi, 2009].

A seguir são apresentados, em ordem cronológica de publicação, os trabalhos identificados: (1) **SimSE** [Navarro e Hoek, 2009]: jogo interativo e de simulação que proporciona aos estudantes uma experiência como gerente de projetos no processo de ES; (2) **RE-O-Poly** [Smith e Gotel, 2008]: jogo de tabuleiro para ensinar, através de um conjunto de perguntas, conhecimentos básicos sobre boas práticas de Engenharia de Requisitos; (3) **Software Quantum** [Knauss, Schneider e Stapel, 2008]: jogo de simulação que enfatiza a importância da Engenharia de Requisitos para atender as expectativas do cliente; (4) **"Guess what we want"** [Alexander e Betty, 2008]: jogo não digital para ensinar estudantes sobre a importância de requisitos bem definidos; (5) **SE•RPG** [Benitti e Molléri, 2008]: jogo que combina elementos de RPG e Simulação para que jogador atue como um gerente de projetos; (6) **SimulES** [Figueiredo et al. 2006]: jogo de cartas onde os jogadores devem conceber, construir e testar uma solução; (7) **Planager** [Rosa e Kieling, 2006]: jogo para apoiar o ensino em gerência de projetos, abordando duas áreas: escopo e tempo; (8) **The Incredible Manager** [Dantas, 2003]: o jogador assume o papel de um gerente de projetos de software, atuando no planejamento e controle do projeto; (9) **SESAM** [Ludewig, 2003]: jogo de simulação e aventura, onde o jogador assume o papel de um gerente em um projeto de software e apresenta a verificação da qualidade do software desenvolvido.

Os jogos foram analisados através de sete critérios: (1) **Possui Definição dos Objetivos Educacionais**; (2) **Classificação do jogo quanto ao gênero**: ação, aventura, puzzle, simulação, estratégia, etc.; (3) **Fornecer o feedback ao aluno sobre o seu desempenho**; (4) **O jogo é disponibilizado livremente para uso**; (5) **O jogo possui uma avaliação da efetividade de seu uso**; (6) **Classificação do jogo quanto à plataforma**: não digital, web, desktop, etc. e (7) **Área de conhecimento da ES** com base no SWEBOOK [IEEE 2004]. A caracterização dos trabalhos avaliados segue a seguinte descrição: (T) Totalmente presente; (P) Parcialmente presente e (N) Não está presente. A Tabela 1 apresenta os critérios e o resultado da avaliação.

Critério	SimSE	RE-O-Poly	Sw. Quant.	G. w. w. w.	SE•RPG	SimulES	Planager	TIM	SESAM
1	T	T	T	N	P	P	P	P	T
2	Simul.	Estrat. Simul.	Simul.	Simul.	Simul. RPG	Simul.	Simul	Simul	Simul Avent
3	T	T	T	N	T	T	T	P	T
4	T	T	T	T	T	T	T	T	T
5	T	T	T	N	T	P	N	T	T
6	Desktop	Não digital	Web	Não digital	Web	não digital	desktop	desktop	desktop
7	Ger. de Proj. Sw.	Requisitos de Sw.	Requisitos de Sw.	Requisitos de Sw.	Ger. de Proj. Sw.	Processo de Sw.	Ger. de Proj. Sw.	Ger. de Proj. Sw.	Ger. de Proj. Sw.

Tabela 1. Comparação dos jogos selecionados

Há uma preocupação na maioria dos trabalhos estudados quanto à importância de experimentos para a avaliação da efetividade da aplicação de jogos no ensino, ainda que não aplicado a todos os jogos avaliados. Com base nos resultados apresentados pelos jogos analisados pouco se pode concluir sobre a efetividade do uso desta abordagem no ensino, sendo identificado como um ponto ainda a ser explorado por outros trabalhos futuros [Navarro, 2006]. Outra característica observada é que a maioria dos jogos é voltada para gerência de projetos ou para engenharia de software de um modo geral. Assim, jogos específicos voltados para os resultados esperados de diferentes processos abordados em modelos de referência de processo podem realmente ampliar o conteúdo didático e oferecer um complemento à capacitação tradicional.

3. Jogos Educativos para Ensino de Engenharia de Software

A seguir são apresentadas as iniciativas do LQPS em desenvolvimento de jogos educativos alinhados a conceitos encontrados em resultados esperados dos processos Desenvolvimento de Requisitos, Gerência de Requisitos, Medição, Verificação e Validação do MR-MPS [Softex, 2009]. Além disso, os resultados dos experimentos também são apresentados e discutidos.

Cada jogo foi analisado a partir de experimentos com o objetivo de avaliar a efetividade da aprendizagem, definidas a partir de hipóteses. Cada turma foi dividida em dois grupos, um de controle e outro experimental. A sequência de execução do experimento foi: 1) aplicação de um questionário de perfil; 2) aplicação de um pré-teste; 3) aplicação do jogo (apenas para o grupo experimental); 4) aplicação de um pós-teste (mesmo grau de dificuldade do pré-teste); e 5) aplicação de um questionário qualitativo. Uma vez que a quantidade de alunos não era suficiente para um teste paramétrico, foi aplicado o teste de hipótese não paramétrico de Mann-Whitney para a comparação entre os resultados do pré-teste com o do pós-teste.

3.1 Jogo Educativo para Ensino de Engenharia de Requisitos de Software

O processo de Engenharia de Requisitos é considerada por alguns autores como a parte mais crítica no desenvolvimento software, pois a qualidade do produto final depende da qualidade dos requisitos [Ferguson; Lami, 2006]. Existem trabalhos demonstrando que 85% dos problemas de software, têm origem na atividade de elicitação de requisitos [Fernandes; Machado; Seidman, 2009].

Para auxiliar no ensino e aprendizagem desta disciplina, foi desenvolvido um jogo chamado “A Ilha dos Requisitos”. O jogo é contextualizado em um ambiente lúdico, com o objetivo de entreter os jogadores, através de seu enredo e das personagens. Durante uma partida, o jogador deverá realizar alguns desafios, onde são avaliados seus conhecimentos em Engenharia de Requisitos. Porém, os conteúdos da disciplina estão imersos no contexto lúdico do jogo, para minimizar o impacto da interação do jogador com o conteúdo formal da disciplina. O objetivo do jogo é ajudar a personagem Jack Reqs, a fugir da “A Ilha dos Requisitos”, uma ilha habitada por tribos canibais e terra de um vulcão que esta prestes a entrar em erupção, e para isto o jogador terá que enfrentar um conjunto de desafios. A Figura 1 apresenta uma tela do jogo com mapa onde o jogador navega entre os desafios.

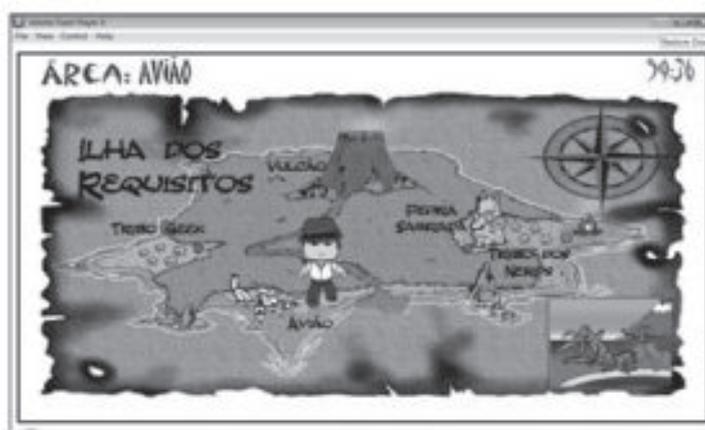


Figura 1. Tela do jogo "A Ilha dos Requisitos"

3.1.1 Desafios

Durante o jogo “A Ilha dos Requisitos” o jogador enfrenta desafios relacionados com o conteúdo da disciplina de Engenharia de Requisitos. Os desafios são executados visando proporcionar ao jogador uma passagem completa por todas as fases do processo de Engenharia de Requisitos. A Tabela 2 apresenta o conteúdo de cada desafio e sua relação com os resultados esperados dos processos Desenvolvimento de Requisitos (DRE) e Gerência de Requisitos (GRE) do MR-MPS.

Nº	Conteúdo abordado	Resultado Esperado
1	Processo de Engenharia de Requisitos	DRE.8, DRE.1 e GRE.1
2	Os papéis e responsabilidades do analista de requ.	DRE.6 e DRE.7
3	Análise do problema	DRE.1
4	Classificação dos requisitos	DRE.2, DRE.3 e DRE.4
5	Validação dos requisitos	GRE.1 e DRE.8
6	O processo de mudança dos requisitos	GRE.3 e GRE.4
7	Gerência dos requisitos	GRE.3 e GRE.4

Tabela 2. Desafios do jogo X Resultado Esperado MPS.BR

3.1.2 Avaliação do jogo

O experimento contou com a participação de 31 alunos (16 no grupo experimental e 15 no grupo de controle). Com os resultados do experimento foi possível afirmar, com 95% de confiança, que houve aprendizado entre o pré e o pós-teste (utilizando o teste não paramétrico de Mann-Whitney). Os testes avaliaram a capacidade do jogo em ensinar Engenharia de Requisitos nos níveis de entendimento e compreensão. O teste qualitativo foi obtido através dos questionários respondidos pelos alunos do grupo experimental. A partir das respostas foi possível inferir que a maioria dos alunos estava motivada para jogar e tiveram a percepção que o jogo foi relevante para o aprendizado.

3.2 Jogo Educativo para Ensino de Verificação e Validação de Software

Apesar do teste de software se fazer presente no processo de desenvolvimento, pesquisas demonstram que suas técnicas e práticas não são totalmente empregadas pelas organizações de desenvolvimento de software [ROCHA et al, 2006]. Ainda, segundo Rocha et al (2006), embora as organizações admitam que aplicam testes de software, 48,5% de suas práticas são consideradas não-aplicadas e não-importantes.

Portanto, visando dar suporte para o ensino e aprendizagem da disciplina de Testes foi desenvolvido o "U-Test" que é um jogo de simulação para apoio ao ensino de teste de software, com foco específico em teste de unidade, abordando questões teóricas e práticas. O jogo é baseado num case onde o jogador se vê como candidato a uma vaga de emprego numa empresa de software. Ele tem como desafio preparar os testes de unidade para as funções já implementadas no sistema. Desta forma, é apresentado ao jogador o case, comentando brevemente sobre a empresa e o projeto ao qual o jogador participará. Em seguida, o jogador deve construir os casos de teste para as funções que lhe serão apresentadas. Ao final o jogador executa os testes no simulador disponibilizado pelo jogo. Como resultado ao seu desempenho o jogador é informado de sua colocação no ranking geral de jogadores. A Figura 2 apresenta uma tela do jogo.

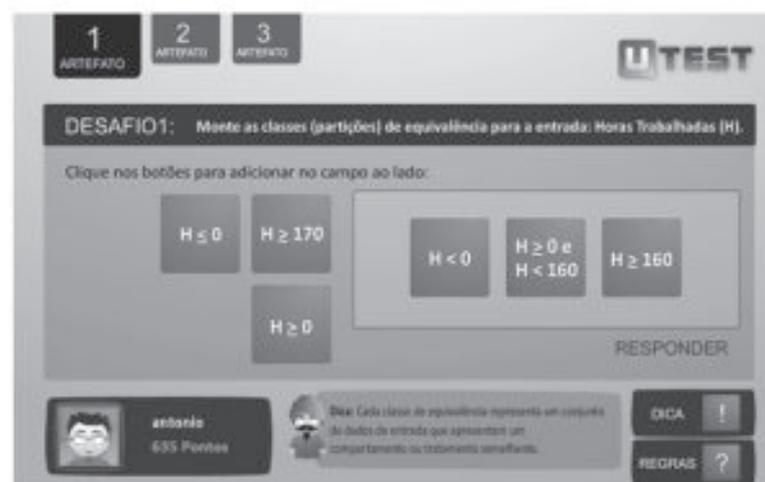


Figura 2. Tela do desafio 1 do "U- Test" com a dica e a resposta do jogador

3.2.1 Desafios

Durante o jogo, o jogador será submetido a dez etapas, sendo que em quatro destas etapas há desafios onde ele terá de aplicar técnicas de teste de software, como particionamento em classes de equivalência e análise de valor limite. Segundo o SWEBOK o teste de software consiste em uma verificação *dinâmica* de um comportamento ou programa. Assim, a Tabela 3 apresenta o conteúdo abordado em cada desafio e sua relação com os resultados esperados do processo Verificação (VER) do MR-MPS.

Nº	Conteúdo abordado	Resultado Esperado
1	Etapa 5 – Apresentação do Artefato	VER 1
2	Etapa 6 - Montar as classes de equivalência	VER 3
3	Etapa 7 - Definir valores limites para as classes identificadas	VER 3
4	Etapa 8 - Selecionar valor correspondente os valores identificados	VER 4
5	Etapa 9 - Montar um grafo de causa-efeito ou árvore de decisão	VER 5
6	Etapa 10 - feedback final do projeto	VER 6

Tabela 3. Desafios do jogo X Resultado Esperado MPS.BR

3.2.2 Avaliação do jogo

O experimento contou com a participação de 12 alunos (6 no grupo experimental e 6 no grupo de controle). Com os resultados deste experimento também foi possível afirmar, com 95% de confiança, que houve aprendizado entre o pré e o pós-teste (utilizando o teste não paramétrico de Mann-Whitney). O teste avaliou a capacidade do jogo em ensinar Testes de Software nos níveis de entendimento, compreensão e aplicação. As respostas ao questionário qualitativo permitiram inferir que a maioria dos alunos estava motivada para jogar e tiveram a percepção que o jogo foi relevante para o aprendizado.

3.3 Jogo Educativo para Ensino de Medição de Software

A pesquisa conduzida pelo SEI (*Software Engineering Institute*) acerca do uso das práticas de medição, apresentou fatores que demonstram uma grande distância entre o que há e o se espera neste contexto [SEI, 2006]. A falta de capacitação na área é um dos maiores causadores desta realidade e da Medição ainda não fazer parte do conteúdo da grade curricular de cursos da área de Computação e Informática. Isto pode ter contribuído também durante a execução deste trabalho não se encontrar jogos educativos na ES com foco no ensino de medição de software.

Portanto, foi construído o X-MED, um jogo que tem como objetivo principal exercitar a aplicação de medição de software voltada para a gerência de projetos alinhada ao nível F do modelo MSP.BR [SOFTEX, 2009] ou o respectivo nível 2 de maturidade do CMMI-DEV v1.2 [CMMI, 2006]. O processo de medição adotado no jogo é baseado no GQM – *Goal/Question/Metric* [Basili et al., 1994] e incorpora elementos do PSM – *Practical Software and Systems Measurement* [McGarry et al., 2001]. O público alvo do jogo é formado por alunos de pós-graduação em cursos de Computação e/ou profissionais de Engenharia de Software.

3.3.1 Desafios

O jogo foi projetado para ser adotado como um complemento às aulas tradicionais fornecendo um ambiente para exercitar os conceitos de Medição. Ao jogar, o aluno faz o papel de um analista de medição. Durante o jogo X-MED o jogador segue as tarefas de um programa de medição, por meio da seleção de soluções adequadas às tarefas de medição, como desenvolver ou selecionar objetivos de medição, planos GQM, planos de coleta de dados e a verificação, análise e interpretação de dados. Foram desenvolvidos sete passos, que são executados de maneira que proporciona ao jogador uma passagem completa por todas as fases do processo de Medição. A Tabela 4 apresenta o conteúdo abordado em cada desafio e sua relação com os resultados esperados dos processos Medição (MED) do MR-MPS.

Nº	Conteúdo abordado	Resultado Esperado
Passo 1	Caracterização de contexto.	MED 1
Passo 2	Identificação do objetivo de medição	MED1
Passo 3	Desenvolvimento do plano GQM	MED 2 E MED 4
Passo 4	Desenvolvimento do plano de coleta de dados	MED 3
Passo 5	Verificação de dados	MED 5
Passo 6	Análise de dados	MED 6
Passo 7	Interpretação de dados	MED 7

Tabela 4. Desafios do jogo X Resultado Esperado cujos conceitos envolvidos são abordados pelo desafio

3.3.2 Avaliação do jogo

Com a execução do experimento, muitos participantes relataram que o jogo os auxiliou a aprender sobre os conceitos e processo de Medição. Entretanto, os dados coletados nos testes estatísticos não confirmaram tais afirmações. Uma explicação para isto pode ser o fato da versão atual do jogo não estar adequada para apoiar efetivamente a aprendizagem, ou o grau de dificuldade do jogo requer um entendimento maior do que o adquirido numa aula com 4 horas de duração. Sendo assim, uma nova versão do jogo foi desenvolvida visando permitir a evolução de sua complexidade para torná-lo um jogo de simulação mais avançado. Espera-se que tais alterações garantam uma experiência de aprendizado mais dinâmica e possibilitar a execução de várias sessões do jogo sem repetição de cenários. A próxima etapa será reaplicar os experimentos na nova versão.

6. Conclusão e Trabalhos Futuros

Este trabalho apresentou três iniciativas de jogos educativos para apoiar o ensino de processos da Engenharia de Software e os resultados da aplicação dos mesmos em experimentos realizados em curso de graduação em Ciência da Computação.

A avaliação dos jogos foi feita por meio de experimentos envolvendo mais de 40 alunos. Os experimentos demonstraram que os jogos contribuem para a aprendizagem e que podem ter um efeito de aprendizagem similar ao de exercícios mais tradicionais. Entretanto, mesmo o efeito sendo similar ao de exercícios, a pesquisa qualitativa demonstra que os alunos ficaram altamente motivados durante as sessões com os jogos. Este efeito poderá auxiliar na fixação do conteúdo, uma vez que os alunos demonstraram interesse em jogar novamente. Vale ressaltar que os resultados obtidos até o momento, mesmo que dentro do rigor estatístico, não podem ser generalizados. Os experimentos foram realizados dentro de duas instituições de ensino apenas, onde os professores eram do mesmo

grupo de pesquisa. Desta forma, é necessário ampliar o conjunto de experimentos para que seja possível avaliar as tendências de aprendizagem.

A utilização dos jogos com alunos permitiu avaliar a possibilidade de eles serem aplicados em um contexto profissional, onde os participantes seriam funcionários de empresas interessadas na implementação de melhorias em seus processos e no alinhamento com modelos como o MPS.BR e CMMI. Os experimentos mostram que os jogos ajudam que as pessoas entendam os conceitos envolvidos nos resultados esperados do MPS.BR. Estes conceitos são necessários para interpretar e implementar o modelo MR-MPS. Portanto, os jogos não seriam utilizados para demonstrar como implementar estes modelos, mas eles poderão apoiar o entendimento de conceitos e técnicas da ES e da MPSe, como consequência, apoiar uma interpretação mais adequada das práticas e resultados esperados.

Assim, jogos educativos podem ser aplicados como um complemento à capacitação de profissionais, considerando que principalmente para auto-estudo, um jogo é muito mais motivador do que um conjunto de exercícios ou de aulas teóricas. Neste sentido, um dos próximos passos desta pesquisa é a realização de experimentos com empresas que já tenham realizado ciclos de melhoria de processo e obtido um nível de maturidade com avaliação oficial MPS.BR ou CMMI. O objetivo será avaliar se o nível de conhecimento atende realmente às necessidades das organizações e também se o mesmo nível de motivação será observado. Estes experimentos podem ser realizados em parceria com outras instituições que apóiam melhoria de processo de software em empresas desenvolvedoras de software. Outras iniciativas de desenvolvimento de jogos estão sendo realizadas. O objetivo não é apenas a ampliação de processos envolvidos nos jogos, mas também a ampliação do conteúdo abordado pelas versões atuais.

Referências

Alexander, M.; Beatty, J. (2008) "Effective Design and Use of Requirements Engineering Training Games". In: REET '08 - Barcelona, Setembro.

Dantas, A. R. (2003) "Jogos de Simulação no Treinamento de Gerentes de Projetos de Software". Dissertação em Engenharia de Sistemas e Computação UFRJ.

Dempsey, J.; Gilley, B.; Rasmussen, W. (1993) "Since Malone's theory of intrinsically motivating instruction: What's the score in the gaming literature?" Journal of Educational Technology Systems, v22 n2 p173-83 ISSN-0047-2395.

El-shamy, S. (2001) "Training Games: Everything You need to Know About Using Games to Reinforce Learning". Stylus Publishing, Sterling, Virginia.

Ferguson, R.; Lami, G. (2006) "An Empirical Study on The Relationship Between Defective Requirements and Test Failures". ISBN: 0-7695-2624-1 In. Software Engineering Workshop, v0 p7-10 IEEE Computer Society, april.

Fernandes, M.; Machado, R.; Seidman, S. (2009) "A Requirements Engineering and Management Training Course for Software Development Professionals". 22th Conference on Software Engineering Education and Training.

Figueiredo, E., Lobato, C. A., Dias, K. L., Leite, J. C. e Lucena, C. J. P. (2007) "Um Jogo para o Ensino de

Engenharia de Software Centrado na Perspectiva de Evolução". Congresso da Sociedade Brasileira de Computação, 2007, p. 37-46. RJ.

IEEE Computer Society. (2004) "SWEBOOK - Guide to the Software Engineering Body of Knowledge".

Knauss, E.; Schneider, K.; Stapel, K. (2008) "A Game for Taking Requirements Engineering More Seriously". 3º Intern. Workshop on (MERE 08), Spain, Barcelona.

Ludewig, J. (2003) "Models in software engineering – an introduction. In Software and Systems Modeling". 5-14. Springer Berlin / Heidelberg. Volume 2, Number 1.

Millbower, L. (2003) "Show Biz Training: Fun and Effective Business Training Techniques from the Worlds of Stage, Screen, and Song". AMACOM/NY.

Montoni, M. (2008) "Lições Aprendidas com Implementação do Modelo MPS em Empresas" p. 31-44 In: MPS.BR: Lições Aprendidas. Campinas, SP, Brasil.

Navarro, E.; Hoek, A. (2009) "Multi-Site Evaluation of SimSE". 40th ACM technical symposium on Computer science education p326-330 Chattanooga, TN, USA.

Rocha, A. R. et al. (2005) "Fatores de Sucesso e Dificuldades na Implementação de Processos de Software Utilizando o MR-MPS e o CMMI", In: I Workshop de Implementadores MPS.BR, Brasília, Brasil.

Rocha, A. R. C. Dias-neto, A. C. Natali, A. C. C. Travassos, G. H. (2006) "Caracterização do estado da prática das atividades de teste em um cenário de desenvolvimento de software brasileiro" SBQS, p. 27-41 Vila Velha – ES, Brasil.

Romero, M.; Vizcaíno, A.; Piattini M. (2008) "A Simulator for Education and Training in Global Requirements Engineering: a Work in Progress". Proceedings of the Eighth IEEE International Conference on Advanced Learning Technologies p123-125.

Rosa, R. e Kieling, E. (2006) "Planager - Um Jogo para Apoio ao Ensino de Gerência de Projetos de Software". TCC Bacharelado em Sistemas de Informação PUC RS.

SEI (2006) Performance Results of CMMI-Based Process Improvement. <<http://www.sei.cmu.edu/publications/documents/06.reports/06tr004.html>>.

SOFTEX. MR-MPS - Melhoria de Processo do Software Brasileiro, Guia Geral: Versão 2009. Brasília: Softex, 2009.

Smith, R.; Gotel O. (2008) "RE-O-Poly: A Customizable Game to Introduce and Reinforce Requirements Engineering Good Practices". Pace University, New York.

Travassos, G.; Kalinowski, M. (2008) "iMPS: resultados de desempenho de organizações que adotaram o modelo MPS", Campinas, SP, Brasil.

Travassos, G.; Kalinowski, M. (2009) "iMPS: Caracterização e Variação de Desempenho de Organizações que Adotaram o Modelo MPS", Campinas, SP, Brasil.

Wangenheim, C. G., Kochanski, D., Savi, R. (2009) "Revisão Sistemática sobre Avaliação de Jogos Voltados para Aprendizagem de Engenharia de Software no Brasil". Fórum de Educação em Engenharia de Software, p. 41-48, 2009.

Wangenheim, C. G. e Shull, F. (2009) "To Game or Not to Game?" Software, IEEE 26, no. 2, p. 92-94.

MPS.BR Nível A: Experiência da Stefanini

Washington Souza, Marcelo Ramasco, Andrea Mattos, Elaine Pinheiro

Stefanini IT Solutions – Fábrica de Software – Jaguariúna, SP – Brasil
wasouza@stefanini.com, mramasco@stefanini.com, amattos@stefanini.com, evpinheiro@stefanini.com

Abstract: *This paper describes the experience of Stefanini IT Solutions – Software Factory with the MPS.Br Level A implementation. It's a long and arduous journey, however the benefits outweighed the investment. The article presents an overview of what is MPS.BR level A, it's main activities and what should be done to get it.*

Resumo: *Este artigo descreve a experiência da Fábrica de Software da Stefanini It Solutions com a implementação do MPS.BR Nível A. Esta é uma jornada longa e trabalhosa, todavia os resultados compensam o investimento. O artigo apresenta uma visão geral do que é o MPS.BR nível A, suas principais atividades e o que deve ser feito para obtê-lo.*

1. Introdução

Desde 1996, a Stefanini It Solutions vem investindo nas atividades de melhoria de processos de software. Com uma metodologia voltada para a prática, bem definida e institucionalizada, as conquistas dos níveis de maturidade aconteceram de um modo natural.

Um dos maiores saltos de qualidade foi a introdução do Six Sigma [SIX SIGMA, 1981]. Este marco de qualidade alterou a cultura organizacional fixando no pensamento dos profissionais o “senso de melhoria” e foi o principal motivador e acelerador do programa de melhoria contínua na empresa.

Nos tópicos abaixo podemos destacar, além do programa de melhoria contínua utilizando Six Sigma, os desafios, adaptações as necessidades do MPS.BR [SOFTEX, 2007], obtenção da alta maturidade, principais benefícios e resultados.

2. Maturidade da Fábrica de Software

Em meados do ano de 2001 a Fábrica de Software Stefanini iniciou o trabalho de estruturação de um Grupo de Engenharia de Processos de forma a analisar a atual Metodologia Stefanini implementada em 1996 e certificada na norma ISO 9001 [NBR ISO 9001, 2000]. Esta metodologia continha práticas de gestão de projetos e de engenharia de software já com um bom nível de documentação de projetos.

Em outubro de 2001 a Stefanini contratou uma consultoria com o objetivo principal de adaptar e complementar os processos da Metodologia Stefanini ao modelo SW-CMM Nível 2 [SEI, 2001]. Os treinamentos nas práticas foram realizados em novembro de 2001 e os ajustes e as definições de

processos foram realizados de dezembro de 2001 a abril de 2002. Os projetos pilotos foram selecionados e todos os profissionais da Fábrica de Software foram treinados nos novos processos. Neste período foram criados os processos para as Metodologias de Fábrica de Projetos e de Fábrica de Programas (codificação). Em Outubro de 2002 foi realizada avaliação oficial e a Stefanini atingiu a maturidade no modelo SW-CMM Nível 2 [SEI, 2001].

Depois de ter utilizado os novos processos com sucesso em vários projetos, a meta era atingir um nível mais alto. O planejamento para atingir a maturidade no modelo SW-CMM Nível 3 [SEI, 2001] ocorreu em dezembro de 2003. De Janeiro a Abril de 2004 os processos foram definidos e ajustados. Em Maio de 2004 todos os profissionais da Fábrica de Software foram treinados. Os projetos pilotos iniciaram em 1º de Junho de 2004. E a avaliação oficial ocorreu com sucesso em Dezembro de 2004.

Desta vez o desafio seria maior, pois a Stefanini teria que ajustar os processos do modelo SW-CMM [SEI, 2001] para o modelo CMMI [SEI, 2005] e definir e institucionalizar os processos para alta maturidade.

Uma Análise de Conformidade dos processos foi realizada em Janeiro de 2005. Logo em seguida foram feitos o planejamento e os treinamentos oficiais. Entre Março e Maio de 2005 os processos foram adequados para atender ao modelo CMMI [SEI, 2005] e foram criados novos processos para atender a alta maturidade e a integração entre as fábricas de software. No fim de Maio de 2005 todos os profissionais foram treinados e elegemos o dia 1º de Junho como a data de corte para os projetos pilotos. A Análise de Prontidão foi realizada em Novembro de 2005 e a Avaliação Oficial com a obtenção da alta maturidade foi realizada em Dezembro de 2005.

2.1 Six Sigma

Em meados de 2006 a Stefanini iniciou os estudos no conjunto de práticas Six Sigma [SIX SIGMA, 1981]. Inicialmente através de auto-estudo pelo Grupo de Engenharia de Processos e depois através de treinamentos externos e cursos do SEI com o objetivo de aprofundar os conceitos estatísticos.

2.2 MPS.BR Nível A

No fim de 2008 a Stefanini iniciou o planejamento para atingir o Nível A de maturidade no modelo MPS.BR [SOFTEX, 2007]. Sabendo da importância do modelo para o cenário nacional, o Grupo de Engenharia de Processos da Stefanini iniciou em Janeiro de 2009 os contatos com a SOFTEX e com os avaliadores.

As análises e estudos dos guias do modelo MPS.BR [SOFTEX, 2007] também foram iniciados neste período. Destas análises foram detectados que os ajustes pontuais deveriam ser feitos nos processos e que as práticas de reutilização deveriam ser documentadas e estruturadas em um novo processo, novas medições e base de componentes reutilizáveis. Em Julho de 2009 foi realizada a Avaliação Inicial. Desta avaliação saíram os pontos que deveriam ser ajustados. O Grupo de Engenharia de Processos atuou rapidamente nas recomendações bem direcionadas pela equipe avaliadora. Em Setembro de 2009 foi realizada a Avaliação Final em que Stefanini obteve o Nível A do MPS.BR.

3. Adaptações ao MPS.BR

Foram necessários ajustes em boa parte dos processos, porém os mais afetados foram os processos de alta maturidade, gerenciamento de projetos e medição, treinamento organizacional, engenharia de software e reutilização.

Apesar da Fábrica de Software da Stefanini trabalhar com conceitos de componentes reutilizáveis desde 2003, os principais pontos de ajustes ocorreram na Gerência de Reutilização e no Desenvolvimento para Reutilização. Foi criado o Processo de Reutilização que contempla os seguintes subprocessos: Revisar e Manter os Critérios para Avaliação de Ativos Reutilizáveis, Avaliar os Ativos Reutilizáveis e Manter a Base de Reutilização, Identificar as Oportunidades de Reutilização de Software, Desenvolvimento com Reutilização e Desenvolvimento para Reutilização.

4. Desafio: Nível B

A maior barreira para se chegar ao MPS.BR Nível A, é a da existência do Nível B no caminho. O Nível B é provavelmente o mais complexo de todo o modelo, pois é neste momento que a implementação como um todo é colocada a prova. O maior ponto de atenção é que os processos estratégicos ou críticos devem estar estáveis e capazes para atingir os objetivos de negócio. Somente com estes processos estáveis e capazes é possível ir para a próxima etapa que é desenvolver modelos de desempenho de processo. Com modelos de desempenho podemos prever o comportamento futuro de um processo e tomar ações antecipadas para atender aos objetivos de negócio.

O Nível B se torna ainda mais complexo e trabalhoso quando as implementações dos níveis anteriores não foram feitas adequadamente, em especial dos Processos de Medições e Análise, Planejamento e Acompanhamento de Projetos. Nestes casos, a organização passa muito tempo corrigindo e adaptando os processos anteriores para só depois estabilizá-los e torná-los capazes. Estas atividades demoram muito tempo e desestimulam a empresa a avançar nos níveis de maturidade.

Na Stefanini tivemos problemas de estabilização e capacitação de processos durante o desenvolvimento dos modelos de desempenho, pois não encontrávamos a correlação entre as causas e as medições. Nestes casos tivemos que aprimorar o sistema de medição coletando informações contextuais sobre o processo. Isto nos fez adiar o programa de obtenção da alta maturidade por pelo menos seis meses.

4.1 Principais Desafios

Durante a implantação dos processos na organização, os principais desafios foram identificados e estão listados a seguir:

- Compreender o que é esperado e os detalhes da alta maturidade;
- Implementar o pensamento estatístico na organização;
- Desenvolver os projetos de melhoria de alta maturidade;
- Implementar o uso consistente de Análise de Causa de Problemas e Resolução;
- Atingir a estabilidade e a capacidade dos processos.

4.2 Projetos de Melhoria Six Sigma

O Six Sigma [SIX SIGMA, 1981] é um conjunto de práticas que foi criado inicialmente para melhoria sistemática de processos e redução de defeitos. Define-se como um defeito uma anomalia em um produto ou serviço contra suas especificações iniciais.

O planejamento estratégico é altamente utilizado para prover mudanças significativas nas organizações fazendo com que se alcancem melhorias nos processos, nos produtos e nos serviços visando a satisfação dos clientes.

O Six Sigma trabalha com dados reais dos processos da organização e possui um conjunto de práticas que orienta os projetos de melhoria de forma sistemática e clara, e para isso, utiliza-se um conjunto de ferramentas estatísticas que auxiliam no aumento de qualidade através de dados e fatos.

Os projetos são normalmente desenvolvidos utilizando o método DMAIC [SIX SIGMA, 1981] (vide figura 1) que possui um conjunto de práticas organizadas de modo a analisar de fato as causas dos problemas e propor soluções efetivas.



Figura 1. Modelo DMAIC utilizado pela Stefanini

Foram desenvolvidos diversos projetos de melhoria em alta maturidade utilizando o Six Sigma, dentre eles destacam-se:

- Densidade de defeitos;
- Atendimento a prazo;
- Remoção de defeitos;
- Attrition;
- Variação da produtividade;
- Retrabalho;

Ao longo dos anos, as necessidades dos projetos Six Sigma na Fábrica de Software foram mudando de acordo com os objetivos de negócio e, neste sentido, alguns novos serão criados e outros desativados.

4.3 Estabilização dos Processos

A estabilização dos processos é uma tarefa complexa e trabalhosa. Entende-se por um processo estável um processo que rotineiramente tem o mesmo comportamento.

Um ponto chave para este trabalho é o uso do processo ACP – Análise de Causas de Problemas e Resolução [SOFTEX, 2007]. O foco é identificar as causas especiais no desempenho dos processos e tratar para que estas causas não aconteçam novamente.

Exemplificando. Imagine que uma pessoa deve andar 10 km para ir do ponto A até o ponto B. No primeiro dia ele leva 15 minutos, no segundo dia 16 minutos e ao final de uma sequência de 30 dias a média é de 15 minutos variando entre 14 a 17 minutos. Realizando alguns testes estatísticos, verifica-se que este processo está estável. Porém no dia 31 esta pessoa levou 23 minutos para fazer este percurso. Analisando o que poderia ter acontecido, descobre-se que uma das faixas da rua estava em manutenção, e isto fez o trânsito ficar mais lento. Isto é considerada como uma causa especial, e conseqüentemente deve ser removida de sua amostra. Ela é uma causa especial porque é uma situação atípica ao processo e provavelmente não acontecerá novamente.

Na Stefanini diversos processos não demonstravam estabilidade e após análises identificamos que a maior parte desta instabilidade era a presença de causas especiais nas amostras. O processo ACP – Análise de Causas de Problemas e Resolução [SOFTEX, 2007] nos ajudou a identificar e remover estas causas especiais.

Após verificar que os processos estavam estáveis (vide figura 2), o próximo passo foi o de verificar se estavam capazes. Entende-se como um processo capaz, um processo que atende as suas necessidades. Exemplificando: Suponhamos que sua empresa fechou um contrato onde os chamados simples devem ser atendidos em até 30 minutos. A melhor equipe de profissionais da organização atende normalmente em 26 minutos com variações entre 19 minutos a 37 minutos.

Chegamos a conclusão que este processo não é capaz, pois apesar de atender seu cliente na maioria das vezes, alguns atendimentos não serão realizados dentro do prazo definido em contrato. O processo seria capaz se todos os atendimentos levassem até no máximo 30 minutos.

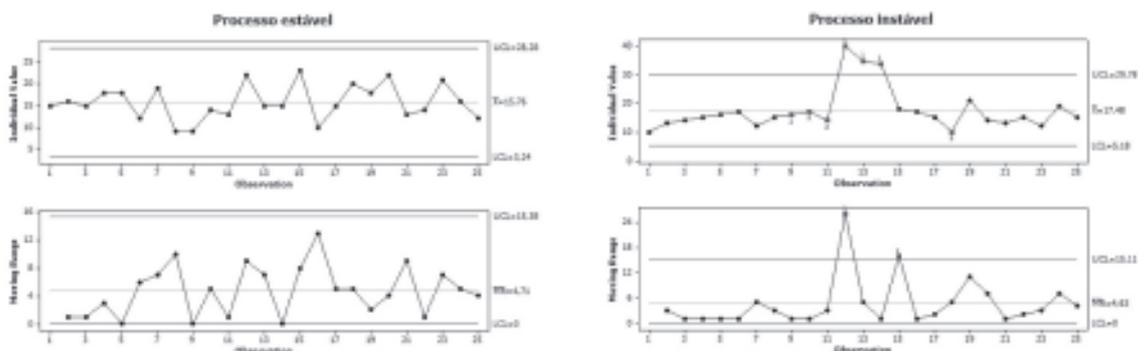


Figura 2. Processo estável x Processo instável

4.4 Baselines de Desempenho

Após verificar que os processos são estáveis e capazes, os Baselines de Desempenho devem ser definidos. O Baseline é a variação de desempenho do seu processo. Seguindo o exemplo anterior, seria estabelecido um Baseline de Desempenho para atendimento dos chamados simples em 19, 26 e 37 minutos. É o que o processo consegue produzir.

É necessário ter tanto os Baselines dos indicadores do processo (Y's) quanto os Baselines dos fatores que influenciam o processo (X's).

4.5 Modelos de Desempenho de Processos

O objetivo dos modelos de desempenho de processos é prever o comportamento futuro dos processos para a tomada de ações antecipadamente. São excelentes ferramentas de gestão preventiva para os Gerentes de Projetos e ajudam a melhorar a qualidade, reduzir o custo e ter maior agressividade nos prazos dos projetos. Todavia o desenvolvimento de modelos de desempenho não é uma tarefa simples, pois é necessário um profundo conhecimento dos processos em questão como coleta de dados (Y e X's) e análises estatísticas.

Os modelos de desempenho (vide figura 3) são normalmente gerados por projetos de melhoria de processos Six Sigma voltados alta maturidade com o objetivo de gerar um modelo para o gerenciamento quantitativo dos processos.

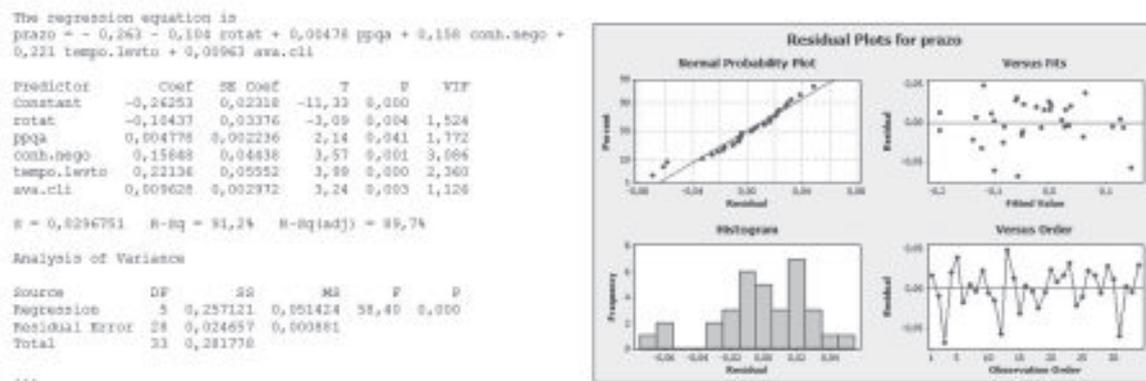


Figura 3. Análise estatística para identificação do melhor modelo de desempenho

Os primeiros modelos de desempenho gerados na Stefanini utilizavam extensas fórmulas matemáticas em planilhas eletrônicas, isto os tornavam demorados e de alto custo de manutenção. Após este período inicial todos os modelos foram refeitos utilizando Six Sigma e MiniTab [Brook, 2006] e desta forma conseguimos reduzir o tempo e custo em menos da metade e, ainda, obtivemos resultados mais concretos e estatisticamente válidos.

Projetos de melhoria utilizando Six Sigma atingem grandes benefícios para a organização.

4.6 Gerenciamento Quantitativo de Processos

As ferramentas para o gerenciamento quantitativo são os modelos de desempenho de processos e simuladores (vide figura 4). Através deles, os gerentes de projetos podem dizer o comportamento futuro de um processo. Exemplificando, um gerente de projetos pode identificar com antecedência de semanas situações de atraso de “n” dias no projeto, ou de um índice de erros acima do esperado em contrato, ou ainda o índice de defeitos produzidos acima do esperado entre diversos outros fatores.



Figura 4. Simuladores de desempenho utilizados na Stefanini

Os processos que serão controlados quantitativamente são derivados dos objetivos de negócio da organização. Recomenda-se que todos os processos críticos ou estratégicos sejam gerenciados quantitativamente.

O gerenciamento quantitativo é realizado através dos ajustes nos fatores (x's) que influenciam o processo (y). O modelo de desempenho é uma equação onde conforme os fatores vão sendo ajustados, o resultado final é alterado. Neste sentido, o gerente de projetos pode identificar qual o melhor cenário para se atender ao objetivo de negócio.

Na figura 5 temos um modelo de desempenho para variação de atendimento a prazo em um processo. Ele mostra que a variação de prazo é influenciada pelos fatores: turn over, defeitos encontrados, conhecimento do negócio, tempo de levantamento de dados e disponibilidade do usuário.

$$\text{Schedule variation (in functional specification) is a function of} \\ - 0,263 - 0,104 \text{ Turnover rate} + 0,00478 \text{ Defects} + 0,158 \text{ Business} \\ \text{Knowledge} + 0,221 \text{ Time of Gathering informations} + 0,00963 \\ \text{Customer availability}$$

Figura 5. Um modelo de desempenho de processos válidos

Em um projeto onde a entrega dentro do prazo é um SLA crítico, pode-se fazer uma simulação ajustando estes fatores para identificar qual o melhor cenário para atender a esta necessidade. A figura 6 demonstra na Simulação 1 que a variação de prazo será entre -12% a 1%, logo a probabilidade de atendimento deste SLA é mínima. Já na Simulação 3, com os fatores ajustados podemos ver que a

variação de prazo seria entre 0% e 10%, o que atenderia as necessidades do negócio.

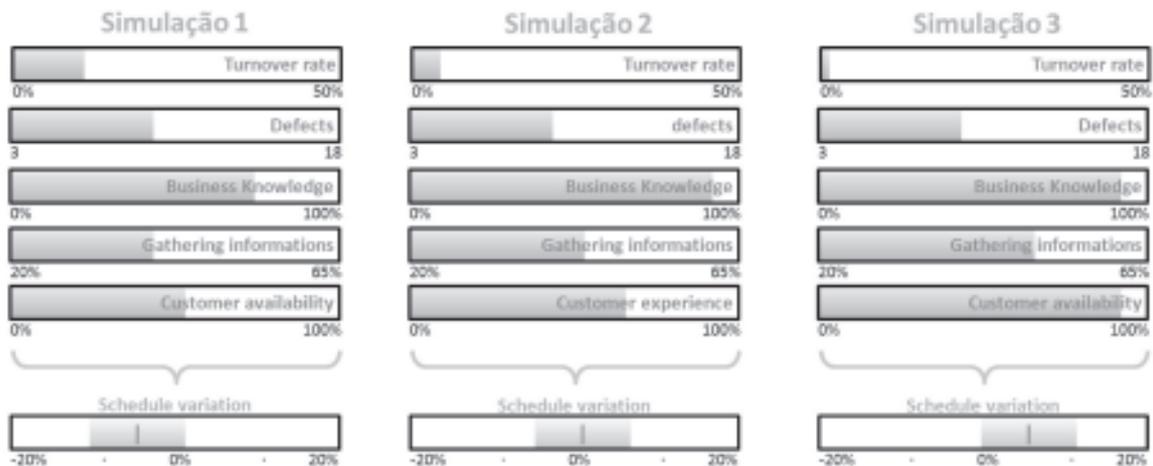


Figura 6. Simulações – ajustes nos fatores para se encontrar o melhor cenário para atendimento dos objetivos de negócios

5. MPS.BR Nível A

A alta maturidade de uma organização é o momento onde o Grupo de Engenharia de Processos possui um profundo conhecimento sobre os processos e os fatores que os influenciam. Neste momento os processos e subprocessos são submetidos constantemente a projetos de melhoria Six Sigma visando a melhoria de seus desempenhos.

O processo ACP – Análise de Causas de Problemas e Resolução [SOFTEX, 2007], apesar de estar localizado no nível A, é muito útil no nível B, em especial nos projetos Six Sigma. Na Stefanini implementamos este processo antecipadamente e recomendamos seu uso durante a implementação do nível B.

A figura 7 apresenta o desempenho de um processo em melhoria contínua: antes do projeto de melhoria, durante o piloto (com as melhorias), após o piloto e após um novo projeto de otimização do processo.

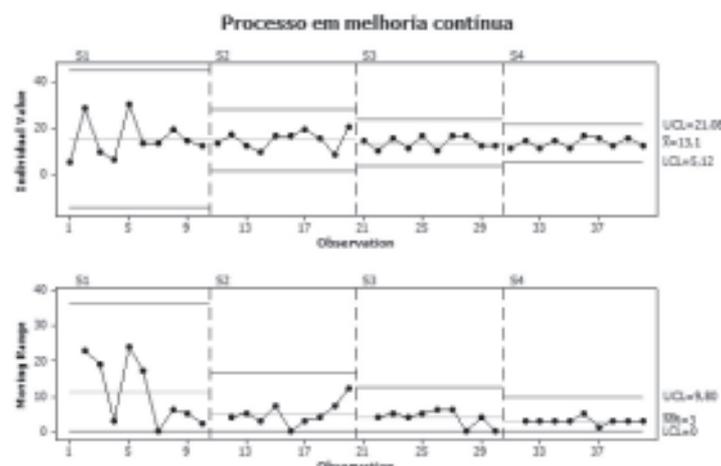


Figura 7. Processo em melhoria contínua ao longo do tempo

6. Recomendações para atingir ao MPS.BR Nível A

Com base na experiência da Stefanini, adquirida ao longo do tempo, recomendamos estas ações para facilitar a conquista do MPS.BR nível A:

- É necessário ter forte apoio da alta direção;
- O processo de Medição e Análise é o ponto chave para o sucesso no Nível A, e portanto, deve ser bem implementado desde os níveis iniciais do MPS.BR;
- Planejar e executar projetos de melhoria utilizando Sigma;
- Treinamentos focados em modelos de desempenho e Six Sigma;
- Os objetivos estratégicos, táticos e operacionais devem estar bem definidos;
- Os indicadores e as medições devem estar vinculadas aos objetivos estratégicos, táticos e operacionais;
- Conhecimento estatístico é primordial para estruturação do Nível B;
- Medições contextuais;
- Apoio de profissionais com experiência em alta maturidade;
- Equipe comprometida com o programa de melhoria de processos;
- Treinamento, consultoria ou coaching focados em alta maturidade e Six Sigma;
- Nos estágios iniciais a organização deve praticar EVM [PMI, 2005] para habilitar o início da compreensão da gestão baseada em dados;
- Criar o “pensamento estatístico” entre os gestores da organização.

7. Principais Benefícios

Os principais benefícios obtidos com o MPS.BR nível A foram:

- Aumento da produtividade;
- Redução do custo de desenvolvimento;
- Redução do retrabalho em todo ciclo de desenvolvimento;
- Aumento significativo na satisfação dos clientes;
- Agilidade no gerenciamento de projetos;
- Rapidez na identificação de problemas e suas causas reais;
- Aumento da qualidade em todo ciclo de desenvolvimento através da redução de defeitos;
- Melhoria contínua dos processos;
- Maior competitividade.

8. Principais Resultados obtidos

Os principais resultados obtidos com a implementação do MPS.BR nível A:

- Redução do Attrition de mais de 30% para menos de 5%;
- Aumento na produtividade em mais de 60%;
- Assertividade nas estimativas de prazos próximo a 100%;

- Redução no custo de desenvolvimento em média de 30%;
- Índice de satisfação dos clientes próximo a 100%.

9. Conclusão

Os Projetos de Melhoria Six Sigma, Modelos de Desempenho de Processos e Análise de Causa de Problemas fazem com que o desempenho da organização melhore como um todo tornando-a mais produtiva e com mais qualidade em seus produtos e serviços.

A implementação de Projetos Six Sigma e o “pensamento estatístico” mudam o jeito que a organização compreende o gerenciamento de projetos e faz com que as decisões sejam tomadas baseadas em dados e fatos e de forma preventiva.

No caso da Stefanini o retorno sobre o investimento foi rápido e podemos dizer que o investimento em melhoria de processos, em especial, MPS.BR níveis B e A valeu cada centavo.

10. Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS - ABNT. Sistemas de Gestão de Qualidade - Requisitos, NBR ISO 9001:2000, Rio de Janeiro, 2000.

ASSOCIAÇÃO PARA PROMOÇÃO DA EXCELÊNCIA DO SOFTWARE BRASILEIRO - SOFTEX. MPS.BR - Melhoria do Processo de Software Brasileiro - Guia Geral, V1.2, 2007. Disponível em: <http://www.softex.br/mpsbr/_guias/default.asp>.

BROOK, Q. Six Sigma and Minitab® - A Complete Toolbox Guide for all Six Sigma Practitioners. QSB Consulting, 2006.

SIX SIGMA. Motorola, USA, 1981.

SOFTWARE ENGINEERING INSTITUTE - SEI. Capability Maturity Model for Software (SW-CMM®), version 1.1, Carnegie Mellon, USA, 2001.

SOFTWARE ENGINEERING INSTITUTE - SEI. Capability Maturity Model Integration (CMMI® SE/SW) for Systems Engineering and Software Engineering, version 1.1, Carnegie Mellon, USA, 2005.

SIVIY, J. M.; PENN, M. L.; STODDARD, R. W. CMMI® and Six Sigma – Partners in Process Improvement. The SEI Series, Institute of Carnegie Mellon University and Addison-Wesley, 2007.

PROJECT MANAGEMENT INSTITUTE - PMI. Practice Standard for Earned Value Management. Project Management Institute, Newtown Square, PA, USA, 2005

Resultados de um Estudo Qualitativo sobre a implementação do modelo MPS em empresas do programa AmazonSoft

Davi Viana¹, Jacilane Rabelo², Carlos Mar², Elisângela Aguiar², Paulino Wagner², Dalton Vilela³, Tayana Conte¹

¹USES – Grupo de Pesquisa em Usabilidade e Engenharia de Software, Programa de Pós-Graduação em Informática – Universidade Federal do Amazonas (UFAM)
Manaus – Amazonas – Brasil

²AmazonSoft – Pólo de Software do Amazonas, Centro de Incubação e Desenvolvimento Empresarial (CIDE) – Manaus – Amazonas

³FUCAPI – Fundação Centro de Análise, Pesquisa e Inovação Tecnológica Manaus – Amazonas
{davi.viana,tayana}@dcc.ufam.edu.br, jacy@doctortech.com.br, carlos.mar@fabriq.com.br, elisangela@infosigma.com.br, wagner_palheta@yahoo.com.br, dalton.vilela@fucapi.br

Abstract. *Nowadays, companies are looking for directions in the adoption of appropriate programs for software process improvement (SPI). It is therefore important to develop research that enables the identification and understanding of the various factors that directly influence the initiatives for software process improvement. This paper has two main goals: (1) present the results of a qualitative study, which used procedures of the Grounded Theory method to identify critical success factors for the point of view of the collaborators, and (2) disseminate knowledge about qualitative studies in order to encourage its adoption by other Software Engineering researchers.*

Resumo. *Atualmente, as empresas procuram por direcionamentos adequados na adoção de programas de melhoria de processo de software (MPS). Por isto é relevante o desenvolvimento de pesquisas que possibilitem identificar e compreender os diversos fatores que influenciam diretamente as iniciativas de melhoria de processo de software. Este artigo possui dois objetivos: (1) apresentar os resultados de um estudo qualitativo, na qual se usou procedimentos do método Grounded Theory (GT) para identificar fatores críticos de sucesso do ponto de vista dos colaboradores das organizações; e (2) disseminar o conhecimento sobre estudos qualitativos, de forma a incentivar seu uso por outros pesquisadores de Engenharia de Software.*

1. Introdução

Quando há uma preocupação maior com a qualidade do processo é possível ter mais produtividade na organização, maior competitividade em relação às outras empresas do mesmo segmento, diminuir o esforço e custo dos projetos e lidar com questões críticas relacionadas ao tempo de lançamento de produtos comerciais [PFLEEGER, 2001 apud ROCHA *et al.*, 2005].

Porém, o sucesso de programas de MPS depende de diversos fatores técnicos e fatores sociais que, segundo SANTANA (2007), são questões socio-técnicas, relativas à condução das iniciativas de melhoria e à interação entre seus participantes. FERREIRA e SILVA (2008) destacam a falta de envolvimento e comprometimento dos envolvidos entre os motivos de insucesso em MPS.

Para auxiliar no direcionamento de adoção de programas de MPS, é de grande relevância o

desenvolvimento de pesquisas que apresentem conduções de estudos qualitativos que possibilitem identificar e compreender, em detalhes, os diversos fatores que influenciam diretamente iniciativas de melhoria de processo de software e da relação dos colaboradores com esses fatores, pois todo o processo de desenvolvimento é dependente do comprometimento humano [SEAMAN, 2008].

Apesar de haver diversos estudos abordando fatores em MPS [MONTONI e ROCHA, 2007], o número de pesquisas que visam analisar qualitativamente as influências nos programas de melhoria ainda está aquém do necessário para permitir uma ampla compreensão desses fatores em diversos contextos [MATOS *et al*, 2010].

Este trabalho descreve um estudo qualitativo em uma iniciativa de MPS no Estado do Amazonas, com o propósito de identificar quais os fatores de influência no programa de MPS, do ponto de vista dos colaboradores das organizações envolvidas. Este artigo possui dois objetivos: (1) discutir os resultados do estudo, na qual se usou procedimentos do método de análise qualitativa *GT* [STRAUSS e CORBIN, 1998]; e (2) disseminar o conhecimento sobre estudos qualitativos, de forma a incentivar seu uso como ferramenta para aprofundar o conhecimento em Programas de MPS.

O restante do artigo está organizado da seguinte forma: a Seção 2 descreve outras pesquisas qualitativas realizadas em programas de melhoria de processo de software. A Seção 3 apresenta a descrição da estratégia de implementação do programa de melhoria nas cinco empresas do Amazonas. A Seção 4 apresenta como foi conduzido o estudo qualitativo para verificar os fatores de influência no programa de melhoria. A Seção 5 discute os principais resultados sobre os fatores de influência identificados. Por fim, a Seção 6 apresenta as conclusões finais e possíveis extensões deste trabalho.

2. Pesquisas qualitativas em programas de melhoria de processo de software

As questões que exercem influência sobre as iniciativas de implementação de melhoria de processos de software vêm sendo objeto de estudos nas últimas décadas [MONTONI e ROCHA 2007]. O propósito desses estudos é obter um melhor entendimento sobre as questões que influenciam iniciativas de melhoria de processo de software, bem como suas interações, causas, efeitos e formas de tratamento. Essas questões são comumente denominadas de Fatores Críticos de Sucesso (FCS).

MONTONI e ROCHA (2007) apresentam uma metodologia para identificação de fatores críticos de sucesso em iniciativas de MPS, na qual são identificados, no contexto nacional: (1) os fatores de ambiente (a capacidade dos ambientes organizacionais de estabelecer e manter iniciativas de MPS, onde essa medida pode ter dois pontos de vista: o indivíduo e a organização); (2) o fator de eficiência da estratégia de implementação do MPS (garantir que os colaboradores da organização tenham conhecimento a respeito dos potenciais benefícios que podem ser alcançados); (3) o fator correspondente do quão sólido é o programa de MPS implementada (se realmente houve mudança na cultura da organização); (4) o fator do quanto a alta gerência está comprometida com o programa de MPS; e (5) é medido o fator de aceitação do programa de MPS e o quanto os colaboradores estão motivados com o mesmo.

Em FERNANDES e OLIVEIRA (2010) é discutida a importância da cultura organizacional como fator de influência nos processos de software em organizações. Na literatura, cultura organizacional é

reconhecida como um fator crítico de sucesso, porém pouco explorada qualitativamente. Segundo o estudo de caso realizado em OLIVEIRA (2010), “o perfil cultural de uma organização influencia a institucionalização dos processos de software”. O mesmo estudo identificou e avaliou a aplicação de recomendações para mudança de cultura organizacional de forma a aprimorar a institucionalização de processos de software.

A aplicação prática de processo de software e a adoção modelos de melhoria de processo de software são discutidas por COLEMAN e O’CONNOR (2008). Os pesquisadores utilizam GT para apresentar que os processos de software empregados na indústria Irlandesa são processos proprietários e, normalmente, são adaptados ao contexto da empresa. Por fim, é observado que há certo tipo de resistência na implementação de programas de MPS, por parte dos gerentes, devido aos custos que essas atividades podem gerar, além de “aumentar a documentação e burocracia”.

A investigação de fatores socioculturais de influência em programas de melhoria é abordada em MONTONI e ROCHA (2010), do ponto de vista dos consultores das organizações de consultoria. Utilizando GT, foi possível identificar três contextos que exercem diferentes tipos de influência no programa de MPS, são eles: Contexto individual; Contexto Organizacional e Contexto Tecnológico. Por fim, também foram identificadas as ações que representam as interações entre os colaboradores e os diferentes contextos abordados no trabalho.

Pesquisas recentes mostram a importância de estudos qualitativos para investigar os diversos FCS que podem influenciar os programas de MPS. Apesar dos resultados possuírem grande relevância para a área de melhoria de processo de software, os mesmos dependem do contexto no qual as organizações estão inseridas e normalmente não podem ser generalizados [MONTONI e ROCHA, 2010]. Por esta razão, faz-se necessária a realização de análises qualitativas em diferentes estudos de caso, que mostrem quais os fatores críticos de sucesso em diferentes contextos.

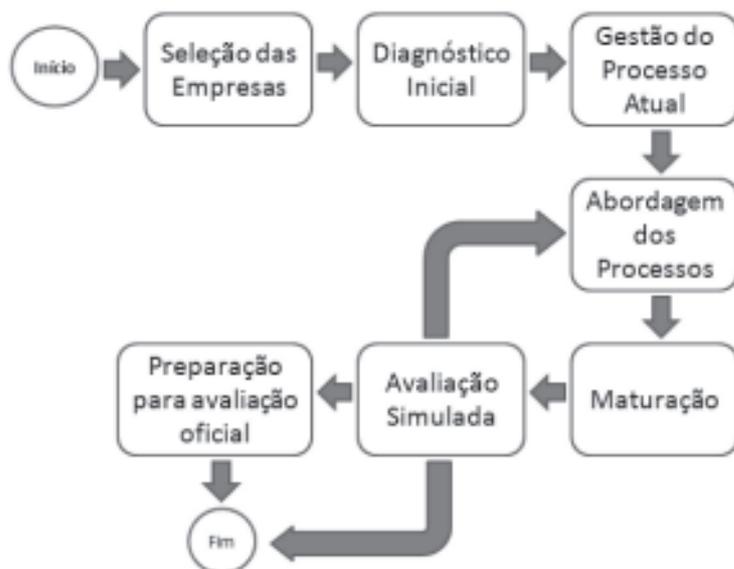


Figura 1. Etapas do programa de implementação do MPS.BR pelo AmazonSoft

3. Estratégia de implementação das iniciativas de MPS no Amazonas

Embora no Brasil já existam mais de 200 empresas avaliadas pelo Programa MPS.BR, somente em 2010 o estado do Amazonas teve suas três primeiras organizações avaliadas, através de incentivos do Programa AmazonSoft¹. Este trabalho relata um estudo feito no contexto do programa de melhoria que guiou a implementação do MPS.BR nestas três organizações. A estratégia de implementação do programa de melhoria foi definido de acordo com as etapas descritas na Figura 1.

Inicialmente foram selecionadas cinco empresas que iriam compor o grupo do programa de implementação do MPS, entre as empresas incubadas no Centro de Desenvolvimento e Incubação Empresarial (CIDE) pertencente ao programa prioritário AmazonSoft. A etapa do diagnóstico inicial visou verificar a aderência do processo de desenvolvimento atual das empresas em relação ao exigido pelo nível G do MPS.BR. Em seguida foi realizada uma apresentação da situação atual de todas as empresas, de modo a analisar quais os principais resultados que deveriam ser trabalhos.

Na fase de gestão do processo atual, os consultores definidos pelo programa de melhoria juntamente com os responsáveis pelas empresas verificaram a melhor forma de adaptar o processo corrente para contemplar todos os requisitos necessários, de maneira que o impacto na forma de trabalho conhecida fosse o mínimo possível. Para que os processos fossem empregados por todos os colaboradores das empresas foi necessário fornecer treinamentos para os mesmos a fim de diminuir o esforço na abordagem dos processos nas empresas. Nesta etapa os consultores também realizavam visitas periódicas as empresas para verificar se a adequação e geração dos artefatos estavam sendo realizadas para o projeto de desenvolvimento piloto.

Na etapa de maturação, a empresa possuía um acompanhamento mais direcionado dos consultores, pois todas as atividades realizadas iriam compor a planilha de avaliação e o processo de institucionalização do processo em toda a organização. Cada empresa definiu seu processo de desenvolvimento, desta maneira foi possível diminuir a resistência por parte dos colaboradores.

A avaliação simulada foi realizada com a finalidade de verificar quais empresas possuíam indicadores suficientes para uma avaliação oficial. Nesta fase, foi possível verificar que duas empresas necessitavam de mais um projeto, pois indicadores importantes foram coletados de maneira equivocada, logo uma empresa voltou para a fase de abordagem dos processos e a segunda empresa decidiu não realizar a avaliação até o momento. A empresa que havia voltado para a fase de abordagem dos processos também decidiu por sair do programa de melhoria pelo fato de faltar projeto para utilizar na avaliação. Por fim, apenas três empresas continuaram a preparação para a avaliação oficial. Na fase final, foram realizados os últimos ajustes necessários para a avaliação oficial, tais como: contratação da avaliação e preparação da planilha de indicadores.

Na próxima seção é apresentado como foi conduzido o estudo qualitativo que teve a finalidade de verificar os possíveis fatores de influência no programa de melhoria realizado nas três empresas que realizaram a avaliação oficial.

¹ AmazonSoft – www.amazonsoft.br

4. Estudo Qualitativo

Através de um estudo qualitativo identificou-se os fatores críticos de sucesso deste tipo de iniciativa com a finalidade de obter uma maior compreensão das influências desses fatores para um programa de melhoria. Este tipo de análise se faz necessária, pois a implementação de iniciativas de melhoria de processo de software representa um grande desafio devido aos diversos fatores que influenciam no processo [SANTANA 2007].

O estudo qualitativo foi realizado junto aos colaboradores das três empresas. Os papéis desempenhados pelos mesmos foram de programadores, analistas e gerentes dos projetos utilizados na avaliação. Analistas e gerentes de projeto tinham mais potencial de exploração de fatores devido aos processos que foram implementados nas empresas (gerência de projetos e gerência de requisitos). Por fim, o estudo foi dividido em três grandes etapas: Planejamento, Execução e Análise/Resultados, conforme descrito na Figura 2.

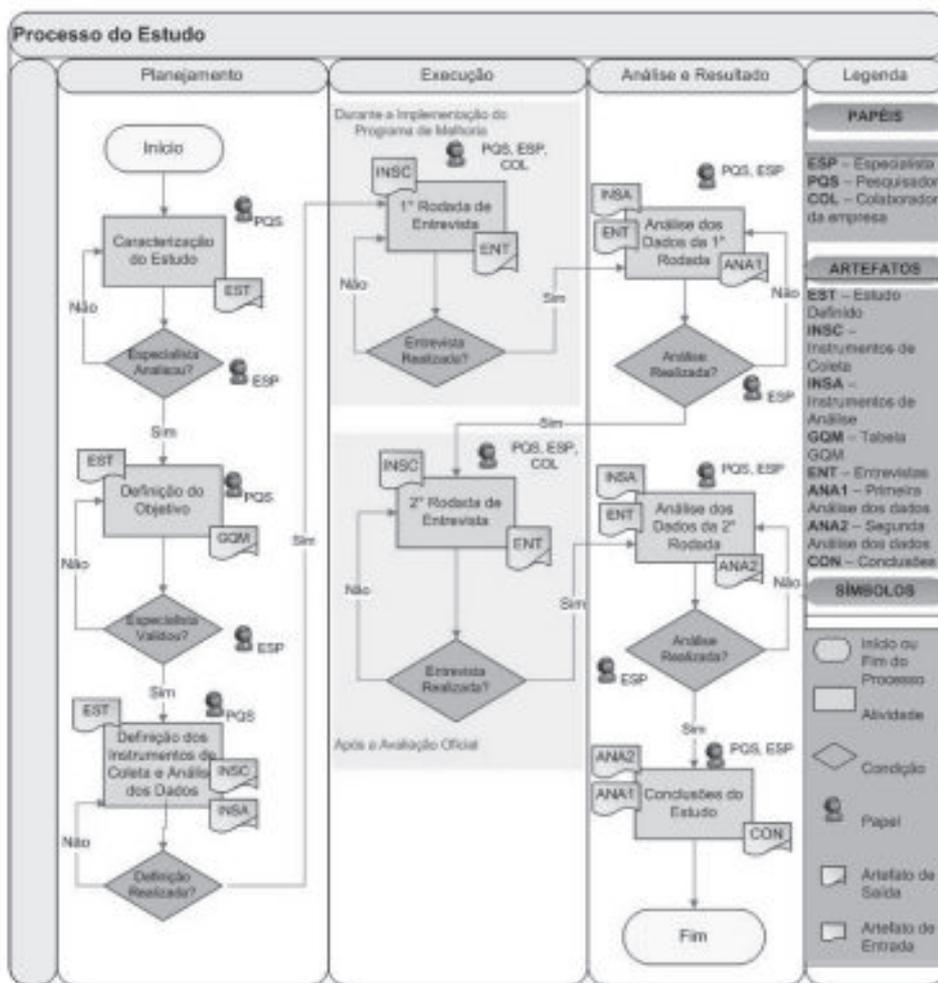


Figura 2. Processo seguido para Planejamento, Execução e Análise do Estudo Qualitativo

Na primeira etapa, tem-se a atividade de caracterização do estudo qualitativo que corresponde à análise dos fatores de influência no primeiro programa de melhoria realizado no Amazonas. Os resultados deste estudo podem servir de auxílio para futuras implementações de MPS que possuem um contexto semelhante.

No Planejamento também é especificado o objetivo do estudo. Para um melhor direcionamento desta pesquisa, o objetivo desse estudo está descrito de acordo com o paradigma GQM (*Goal-Question-Metric* – BASILI e ROMBACH, 1988) na Tabela 1. A última atividade desta etapa, estabelece que deve-se definir os instrumentos de coleta e análise de dados. Para realizar a coleta de dados optou-se por utilizar entrevistas semi-estruturadas baseadas questões abertas, pois segundo YIN (2001), “Entrevistas semi-estruturadas tem como objetivo principal compreender os significados que os entrevista-dos atribuem às questões e situações relativas aos temas de interesse”. Desta forma, pode-se investigar o ponto de vista do colaborador a partir de suas afirmações e compreender os fatores que influenciaram o programa de melhoria. Nesta atividade, definiram-se as questões iniciais do questionário a ser utilizado nas entrevistas.

Analisar	Programas de melhoria de processo de software
Com o propósito de	caracterizar
Em relação à	Influência dos fatores críticos de sucesso
Do ponto de vista	Dos colaboradores da empresa
No contexto de	Em empresas pioneiras na implementação do nível G do MPS.BR no Estado do Amazonas

Tabela 1. Objetivo do Estudo Qualitativo segundo o paradigma GQM1

O método de análise qualitativa utilizado está baseado nos procedimentos do método *Grounded Theory* (ou Teoria Fundamentada em Dados), que visa criar uma teoria a partir dos dados coletados. O método *GT* 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). Seus autores, Glaser e Strauss, afirmam que existem dois tipos básicos de teorias: as formais e as substantivas (BIANCHI e IKEDA 2006). O primeiro tipo é composto das teorias conceituais e abrangentes, enquanto que o segundo tipo é específico para determinado grupo ou situação e não visa generalizar além da sua área substantiva. O foco em pesquisas utilizando *GT* é o segundo tipo básico de teoria.

A etapa de execução corresponde à condução das entrevistas nas empresas. Esta etapa foi dividida em duas fases: “Durante a implementação do programa de melhoria” e “Após a avaliação oficial”. Desta forma, foi possível obter dados referentes aos dois momentos importantes do programa de melhoria.

Para as entrevistas, selecionaram-se colaboradores que estavam envolvidos com o programa de melhoria. Foi criado um Termo de Consentimento Livre e Esclarecido (TCLE) para informar a respeito do procedimento e confidencialidade da pesquisa e foram estimulados a falar o mais livremente possível. Os mesmos foram entrevistados em relação à definição e implementação do programa de melhoria do processo de software em suas organizações. As entrevistas foram realizadas individualmente e gravadas. As questões utilizadas nas entrevistas deste estudo foram baseadas nas questões definidas em Matos *et al.* (2010). Na 2ª rodada de entrevistas, o questionário foi modificado com o propósito de aperfeiçoar o questionário para obter novos dados.

Após a execução das entrevistas, foi realizada a transcrição na íntegra das mesmas. Foram gerados documentos individuais com todos os conteúdos gravados nas entrevistas, esses documentos foram descaracterizados para preservar a identidade dos colaboradores e utilizados na etapa de análise e resultado.

Na última etapa do processo, os dados qualitativos desta pesquisa foram analisados. Utilizou-se um subconjunto das fases do processo de codificação sugerido por STRAUSS e CORBIN (1998) para o método *Grounded Theory* – as codificações aberta (*open coding*) e axial (*axial coding*). Após transcrever as entrevistas, foi iniciada a etapa de codificação aberta, onde trechos relevantes que representavam dados específicos para o estudo foram relacionados a códigos. Na fase de codificação aberta os incidentes ou eventos são agrupados em códigos através da comparação incidente–incidente (BANDEIRA-DE-MELLO e CUNHA 2003). Em seguida aplicou-se a codificação axial que examina as relações entre as categorias que formam as proposições da teoria sub-tantiva (BANDEIRA-DE-MELLO e CUNHA 2006).

5. Análise dos dados qualitativos

Para a análise dos dados, foi eleito um questionário como fonte inicial. Ao se analisar os dados deste questionário, criou-se códigos relacionados a trechos do texto referentes à implementação do programa de MPS (Figura 3). Várias interações de comparações foram realizadas para a seleção de códigos que indicavam relatos representativos em citações no texto. Esta análise refere-se à segunda rodada de entrevistas realizada, ocorrendo após a avaliação oficial do modelo MPS.

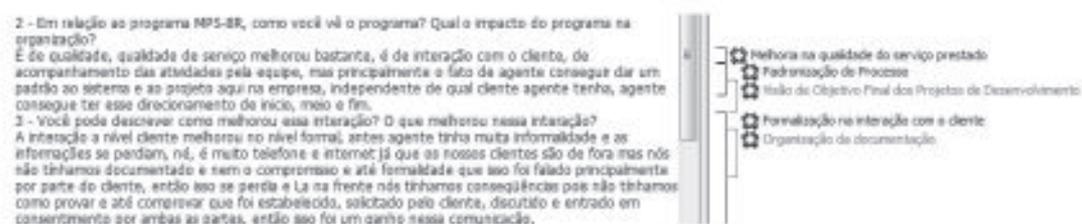


Figura 3. Codificação aberta de trecho de uma entrevista

Iniciou-se então a identificação dos relacionamentos entre códigos na fase de codificação axial, gerando inter-relações que agrupam a natureza dos fatores envolvidos em programas de MPS.

Nas Figuras 4 e 5 são apresentados os agrupamentos dos códigos de acordo com suas propriedades, formando assim conceitos que representam as categorias (assinalados com [FCS]). Estas categorias foram analisadas em conjunto com outros pesquisadores do grupo de pesquisa USES com o objetivo de proporcionar uma maior clareza sobre o fenômeno em questão.

Em relação aos fatores positivos (Figura 4), pode-se observar que houve melhoria no processo de software a partir da padronização deste, pois todas as atividades para o programa de melhoria e a documentação necessária foram bem definidas pelos responsáveis do programa de melhoria e do apoio da consultoria externa. Esta padronização possibilitou a visualização mais completa dos objetivos a serem alcançados com os projetos de software.

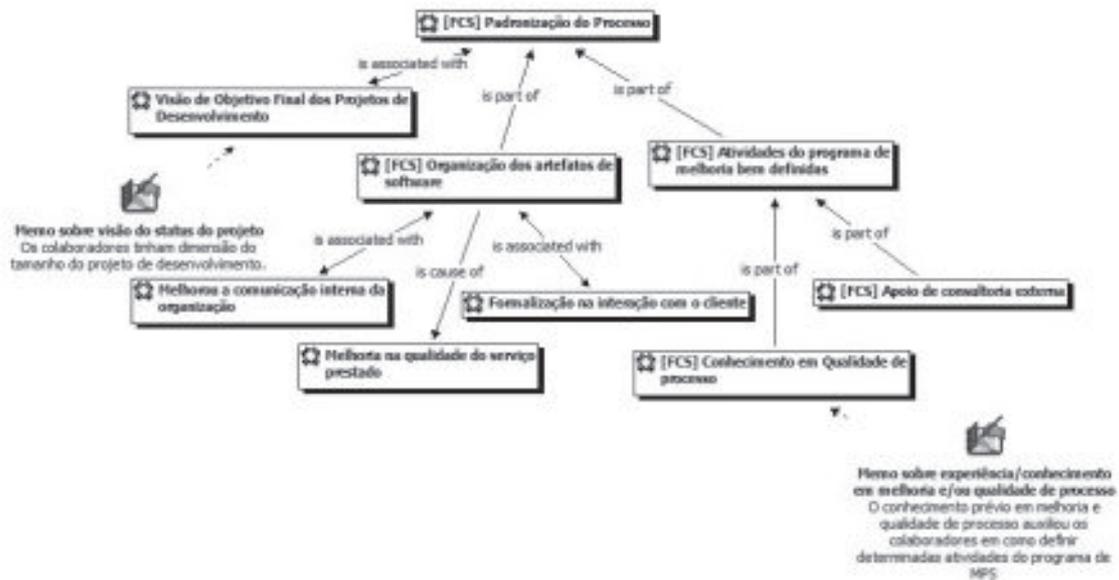


Figura 4. Representação gráfica de fatores de influência positivos

Também pode ser observado que houve melhoria na qualidade do serviço prestado a partir da organização dos artefatos de software, assim como melhoria na comunicação interna e na interação com o cliente. Isto foi percebido pelo cliente a partir do momento que ele precisava validar informações para dar seguimento ao processo.



Figura 5. Representação gráfica de fatores de influência negativos

Como fator de influência negativo encontrado, destaca-se a resistência em relação ao programa de melhoria (Figura 5). Os colaboradores julgaram as atividades do programa de melhoria muito trabalhosas e complicadas de forma a necessitar de treinamento em diversas atividades necessárias para o programa de MPS. Um ponto importante a ser notado é que algumas decisões da diretoria desagradavam aos colaboradores envolvidos com o programa de melhoria, como a definição estratégica dos projetos/clientes utilizados para ser apresentados na avaliação. Por fim, os colaboradores também tinham que desenvolver as atividades do dia-a-dia da empresa, desta forma algumas atividades do programa de melhoria se tornavam atividades secundárias, como por exemplo, ter que atender uma solicitação de mudança de requisito do cliente sem antes realizar todo o registro de solicitação de mudança. Estas influências negativas foram superadas com a determinação dos colaboradores em alcançar o objetivo do programa de melhoria.

O método *GT* compreende ainda a fase de codificação seletiva, que diz respeito à identificação da categoria central da teoria, porém esta fase ainda não foi executada, deve-se ao fato de que uma regra na utilização do método *GT* é a circularidade entre as fases de coleta e análise (BANDEIRA-DE-MELLO e CUNHA 2006), na qual as propriedades de uma categoria conceitual são validadas pelo pesquisador através da análise de novos dados coletados. Cabe salientar que nem todos os códigos apontados nos questionários foram relacionados às categorias até então criadas, além disso, ainda não foi possível validar as propriedades das categorias identificadas. Como o processo de codificação ainda não foi dado como concluído, pois este estudo faz parte de uma pesquisa maior com o objetivo de verificar aspectos humanos em programas de melhoria de processo de software. Desta forma, optou-se por aguardar a coleta e análise de novos dados, o que pode implicar a criação de novas categorias que podem ser associadas a estes códigos. As demais categorias identificadas encontram-se em [Santos *et al.*, 2010].

6. Conclusões e Trabalhos Futuros

Esta pesquisa apresentou a condução de um estudo qualitativo com a finalidade de identificar FCS no programa de MPS. Para a realização deste trabalho, foram conduzidas entrevistas em organizações de software pioneiras na implementação do nível G do MPS.BR no Amazonas.

Os procedimentos de análise qualitativa são úteis no sentido de buscar a essência de determinado evento que, neste trabalho, refere-se aos fatores relacionados a iniciativas de programa de melhoria. Um programa de melhoria apresenta diversos fatores que podem influenciar no sucesso ao final da implantação. Neste trabalho foi possível obter, através da análise qualitativa, a compreensão de alguns fatores que contribuíram para uma avaliação bem sucedida do modelo adotado. Verificou-se que a padronização do processo é um importante fator crítico de sucesso, pois é necessário um direcionamento do que precisa ser feito e como precisa ser feito. Para isso necessitou-se de conhecimento em processos/qualidade de software.

É importante ressaltar que a generalização dos resultados dos trabalhos qualitativos é limitada pelo fato dos resultados encontrados estarem relacionados diretamente ao contexto onde o estudo foi aplicado, porém é relevante para o entendimento mais preciso dos fatores de influência estudados.

Uma possível extensão deste trabalho é replicar o processo do estudo qualitativo proposto nos mais diversos programas de MPS do país com a finalidade de criar uma base de conhecimento que torne possível a extração de similaridade dos fatores críticos de sucesso que influenciam nestas iniciativas.

Agradecimentos

Os autores agradecem ao Amadeu Neto e Vinícius Secatti por seu auxílio em relação ao uso dos procedimentos qualitativos e aos colaboradores das empresas avaliadas. Agradecemos também o apoio financeiro da FAPEAM e do CNPq.

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.
- BANDEIRA-DE-MELLO, R., CUNHA, C. (2006). "Grounded Theory". Em Godoi, C. K., Bandeira-de-Mello, R., Silva, A. B. d. (eds), Pesquisa Qualitativa em Estudos Organizacionais: Paradigmas, Estratégias e Métodos, cap. 8, São Paulo, Saraiva.
- BASIL, V. H ROMBACH (1988). "The tame project: towards improvement-oriented software environments". IEEE Transactions on Software Engineering, pg 758-773.
- BIANCHI, E. M. P. G., IKEDA, A. A. (2006). "Analisando a Grounded Theory em Administração". IX SEMEAD - Seminários em Administração. São Paulo, Brasil.
- COLEMAN, G., O'CONNOR, R. (2008). "Investigating software process in practice: A grounded theory perspective", Journal of Systems and Software, v. 81, n. 5
- FERNANDES, P. G. (2010). "Cultura Organizacional como Apoio à Institucionalização de Processos de Software". (Dissertação de Mestrado - UFG)
- FERNANDES, P. G., OLIVEIRA, J. L. (2010). "Perfil Cultural e Institucionalização de Processos de Software: Estudo de Caso em Duas Organizações de Software". WOSES 2010.
- FERREIRA, P., SILVA, F. (2008), "Fatores humanos que influenciam a utilização de processos de software", In: Anais do VII SBQS, pp. 123 – 138.
- MATOS, O., SECATTI, V., SANTOS, D., OLIVEIRA, H., CONTE, T. (2010). "Aplicando Grounded Theory para Compreender os Fatores Críticos de Sucesso em Iniciativas de Melhoria de Processo de Software". WOSES 2010.
- MONTONI, M., ROCHA, A. (2007). "A Methodology for Identifying Critical Success Factors that Influence Software Process Improvement Initiatives: An Application in the Brazilian Software Industry", EuroSPI 2007, LNCS 4764, pp. 175-186.
- MONTONI, M., ROCHA, A. (2010). "Aplicação de Grounded Theory para Investigar Iniciativas de Implementação de Melhorias em Processos de Software". In: Anais do IX SBQS, pp 167-181
- ROCHA, A. R. et al. (2005). "Fatores de Sucesso e Dificuldades na Implementação de Processos de Software Utilizando o MR-MPS e o CMMI", Pro Quality– Qualidade na Produção de Software.
- SANTANA, A. (2007). "Problemas em Iniciativas de Melhoria de Processos de Software sob a Ótica de uma Teoria de Intervenção". Dissertação de Mestrado. Universidade Federal de Pernambuco. Março 2007.
- SANTOS, D., MATOS, O., RABELO, J., SECATTI, V., VIANA, P., VILELA, D., CONTE, T., 2009. "Análise Qualitativa do programa de Melhoria de Processo em empresas do Amazonas". Relatório técnico RT-DCC-USES003/2010. DCC/ UFAM.
- SEAMAN, C. B., 2008, "Qualitative Methods". In: SHULL et al. (eds.), Guide to Advanced Empirical Software Engineering, Chapter 2, Springer
- STRAUSS, A., CORBIN, J. (1998). "Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory". 2 ed. London, SAGE Publications.
- YIN, R. K. (2001). "Estudo de caso: Planejamento e métodos. Porto Alegre: Bookman.

Análise de Ferramentas para Apoio à Gerência de Projetos e Gerência de Requisitos de Software

Fabiana Freitas Mendes¹, Patrícia Gomes Fernandes¹, Juliano Lopes de Oliveira², Caroline da Cunha Mota¹, Maria Dalva S. Martins¹, Rosângela da Silva Nunes¹.

¹Centro de Recursos Computacionais– Universidade Federal de Goiás (UFG)

Caixa Postal 131 – 74.001-970 – Goiânia– GO – Brazil

²Instituto de Informática – Universidade Federal de Goiás (UFG)

Caixa Postal 131 – 74.001-970 – Goiânia– GO – Brazil

{fabiana,patricia}@cercomp.ufg.br, juliano@inf.ufg.br, {carolinecmota, dalvinha, rdsnunes}@gmail.com

Abstract. *To be effective, software process improvement needs tools to support the activities described in the processes. Without agility in execution, obtained through the adoption of tools, activities that enable visibility and control of software processes can become a bottleneck and thus hinder the institutionalization of these processes. This article examines tools for requirements tracking and projects scheduling to support the processes of Project Management and Requirements Management in software projects. The results of this analysis were used in practice, driving the acquisition of tools according to the needs established in the software process of the organization responsible for developing and maintaining software in a Federal University.*

Resumo. *Para ser efetiva, a melhoria de processo de software precisa contar com ferramentas que apoiem as atividades previstas nos processos. Sem a agilidade na execução, obtida por meio da adoção de ferramentas, as atividades que permitem visibilidade e controle dos processos podem se tornar um gargalo e, conseqüentemente, dificultar a institucionalização dos processos. Este artigo analisa ferramentas de rastreabilidade e de cronogramação para apoio aos processos de Gerência de Projetos e Gerência de Requisitos em projetos de software. Os resultados desta análise foram utilizados na prática, orientando a aquisição de ferramentas adequadas às necessidades estabelecidas no processo de software da organização responsável pelo desenvolvimento e manutenção de software em uma Instituição Federal de Ensino Superior.*

1. Introdução

O CERCOMP (Centro de Recursos Computacionais) é o órgão responsável pelo atendimento das necessidades de software de toda a Universidade Federal de Goiás (UFG). Nesse sentido, o CERCOMP desenvolveu e mantém mais de quarenta sistemas de informação corporativos que automatizam processos administrativos e acadêmicos da UFG. Com a aceleração da expansão universitária, houve um aumento significativo da demanda por novos sistemas, e pela evolução dos sistemas que já apoiam os processos da UFG. Para atender essa demanda, o CERCOMP iniciou, em setembro de 2007, um programa de Melhoria de Processos de Software (MPS). Este programa, que está em seu terceiro ciclo de melhoria, tem como principal desafio a institucionalização dos processos definidos [Mendes et al 2010].

Um dos fatores críticos para superar esse tipo de desafio é a adoção de ferramentas que apoiem a execução dos processos de software definidos [Souza e Oliveira 2005]. Este artigo discute um projeto de aquisição de ferramentas, conduzido dentro do programa de MPS do CERCOMP, visando a análise e a seleção de ferramentas para apoiar a institucionalização dos seus processos de Gerência de Projetos e Gerência de Requisitos.

O restante deste documento analisa os principais aspectos desse projeto de aquisição de ferramentas e está organizado da seguinte maneira. A Seção 2 resume o planejamento do projeto de aquisição, discutindo as atividades dos processos de software que deveriam ser apoiadas por ferramentas, e a escolha dos critérios para análise e comparação dessas ferramentas. A Seção 3 discute os resultados obtidos da comparação de ferramentas selecionadas no projeto. A Seção 4 sintetiza as lições aprendidas da execução desse projeto. A Seção 5 apresenta considerações finais sobre aquisição de ferramentas para apoio a MPS.

2. Planejamento do Projeto de Aquisição de Ferramentas

As atividades de aquisição de ferramentas de apoio aos processos de software foram conduzidas de acordo com duas referências principais: o Guia de Aquisição do modelo MPS.BR [SOFTEX 2009]; e a Norma IEEE 1062 – Recommended Practices for Software Acquisition (Práticas Recomendadas para Aquisição de Software) [IEEE 1998].

Estas referências definem um conjunto de atividades para que a aquisição de software seja feita com qualidade e atenda as necessidades da organização adquirente. Estas atividades podem ser classificadas em cinco grupos:

- **Planejamento:** atividades relacionadas à definição de requisitos, a critérios de análise e de aceitação do produto, e a identificação de potenciais fornecedores.
- **Análise e Escolha do Software:** atividades relacionadas a avaliação das propostas (ou oportunidades) e, de acordo com esta avaliação, a escolha da opção mais adequada aos critérios e requisitos estabelecidos.
- **Contratação ou Obtenção do Software:** atividades que envolvem a formalização da opção escolhida junto ao fornecedor selecionado (negociando um contrato, por exemplo).
- **Acompanhamento da Aquisição:** atividades para acompanhar o desenvolvimento ou adaptação do produto selecionado, caso este ainda não esteja pronto no momento da aquisição.
- **Aceitação do software:** atividades que tratam da aplicação dos critérios de aceitação ao produto final, e da disponibilização deste produto para uso pela organização adquirente.

Por questões de espaço, o presente artigo foca na discussão das atividades dos dois primeiros grupos: planejamento, e análise e escolha de ferramentas para apoio aos processos de software do CERCOMP.

Em relação ao planejamento da aquisição, inicialmente foram selecionados os aspectos dos processos de software do CERCOMP que deveriam ser apoiados por ferramentas. Neste momento estavam definidos os seguintes processos de software: gerência de projeto, gerência de requisitos, gerência de configuração e solução técnica (que engloba atividades técnicas relacionadas ao desenvolvimento de requisitos, design, construção e teste de software).

Dentre estes processos, foram considerados críticos aqueles que estavam relacionados aos objetivos

de visibilidade e controle de projetos de software: Gerência de Projeto e Gerência de Requisitos. Especificamente, foram escolhidas para automatização as atividades relacionadas à cronogramação de projetos e ao controle da rastreabilidade de requisitos, pois estas atividades foram consideradas pelas equipes técnicas do CERCOMP como as mais trabalhosas.

Os critérios para análise das ferramentas foram selecionados e adaptados daqueles propostos no Anexo A da Norma IEEE 1062 [IEEE 1998]. Esses critérios foram divididos em seis grupos, de acordo com o tipo de dados envolvidos:

- Forma de obtenção da ferramenta;
- Configuração de software da ferramenta;
- Histórico de melhorias e correções sobre a ferramenta;
- Informações sobre o uso da ferramenta;
- Indicadores de qualidade da ferramenta; e
- Dados sobre a transferência da ferramenta.

Para esses grupos foram definidos 57 critérios de análise. A lista completa dos critérios de cada um dos seis grupos citados pode ser obtida em [GPS 2010].

Além desses critérios genéricos, também foram avaliadas as necessidades específicas para cada tipo de ferramenta. Essas necessidades foram identificadas com base na análise do processo definido e na indicação das equipes técnicas sobre a dificuldade encontrada para realizar atividades de cronogramação e de controle de rastreabilidade nos projetos de software realizados no CERCOMP. Estas necessidades são apresentadas, respectivamente, nas Tabelas 1 e 2.

O próximo passo do projeto de aquisição foi identificar e selecionar as ferramentas que seriam analisadas com base nos critérios definidos. Para tanto, foi feita uma busca ad hoc na Internet por ferramentas que poderiam atender esses critérios.

3. Análise de Ferramentas

A análise das ferramentas foi fundamentada na documentação encontrada nos seus respectivos sites, assim como em versões de demonstração ou nos produtos propriamente ditos, quando disponibilizados pelos seus fornecedores.

Em relação à documentação dos produtos de software, para nenhuma das ferramentas analisadas foi encontrado um documento formal de especificação de requisitos; a maioria das ferramentas disponibiliza apenas uma lista das suas principais funcionalidades.

Por outro lado, todas as ferramentas não proprietárias analisadas disponibilizaram seus códigos-fonte para análise.

As seções 3.1 e 3.2 sintetizam, respectivamente, os resultados obtidos da análise de ferramentas de cronogramação e de rastreabilidade. Os relatórios completos dessas análises podem ser obtidos em [GPS 2010].

3.1. Análise de Ferramentas de Cronogramação

Inicialmente foram identificadas nove ferramentas relacionadas a atividades de cronogramação de projetos de software: GanttProject, Project Open, Microsoft Project, Primavera, RationalPlan, Open Project, DotProject, JxProject e RedMine. Dentre essas ferramentas, três são proprietárias: Microsoft Project, RationalPlan e Primavera. A ferramenta OpenProj tem versões proprietária e livre.

O relatório detalhado da análise de cada uma destas ferramentas, contendo inclusive as referências onde podem ser obtidas mais informações sobre cada ferramenta, pode ser encontrado em [GPS 2010].

Dentre as ferramentas analisadas, não foi possível obter versões de demonstração para Primavera e OpenProj. Além disso, não foi encontrada documentação adequada para análise dessas ferramentas, e também para a ferramenta RationalPlan.

Todas as ferramentas analisadas apresentaram algum tipo de manual de usuário (ou documento similar). A maioria delas também forneceu documentos de apoio a instalação, ou são ferramentas que podem ser executadas sem instalação direta (tipicamente via arquivos do tipo jar). Apenas para as ferramentas GanttProject e Primavera não foi possível encontrar essas informações (embora elas possam tê-las).

Quanto aos sistemas operacionais em que as ferramentas podem ser executadas, Microsoft Project e Primavera são restritas ao Ambiente Windows. Além disso, a versão Server do Microsoft Project só pode ser executada no Sistema Windows Server. Já as ferramentas RationalPlan, OpenProj, GanttPV, GanttProject e Project-Open podem ser executadas tanto em ambiente Windows quanto em Linux.

Com relação aos requisitos estabelecidos, a Tabela 1 apresenta um resumo da cobertura apresentada pelas ferramentas.

Devido ao tempo limitado para a análise das ferramentas, ou à falta de acesso a uma dada versão da ferramenta, nem sempre foi possível identificar se uma determinada característica estava presente ou não. Nestes casos, a característica foi classificada como “Não Encontrada (NE)”. Isto significa que não foi possível identificar a característica na ferramenta, embora a análise feita não permita afirmar que a ferramenta não a possua.

Requisito	1	2	3	4	5	6	7	8	9
Possui os dados: Fase/Atividade/Tarefa, esforço estimado, esforço real, data início previsto, data fim previsto, data início real, data fim real, recurso previsto, recurso real	P	P	P	S	N	N	P ⁶	P	S
Permite várias versões de um cronograma, gravando as versões antecedentes	N	NE	S	S	N	N	NE	N	NE
Calcula o esforço total previsto e o total realizado em um determinado período de tempo, sendo este período informado pelo usuário OU determinado por atividades selecionadas aleatoriamente pelo usuário	N	NE	NE	S	N	N	N ⁷	N	P ¹⁰
Informa ou destaca as atividades que estão programadas para serem feitas na semana e/ou dia corrente	N	NE	NE	S	N ⁵	N	S	N ⁴	S
Informa ou destaca as atividades que, segundo a data de fim previsto, estão atrasadas	S ¹	NE	S ⁴	S	N	N	S	N	S
Oferece suporte a multi-projetos, permitindo que a alocação de recursos seja controlada com base em um conjunto de projetos	N ²	S ³	S	S	S	S	S	N	P ¹¹
Gera o caminho crítico do projeto	N	NE	S	S	P ⁴				
Permite a quebra de uma atividade em sub-atividades	S	S	S	S	S	S	S ⁸	S	S
Permite a comparação do esforço previsto x realizado	N	S	NA	S	N	N	S	N	N
Permite a visualização do cronograma dos projetos através de Gráficos de Gantt	S	S	S	S	S	S	S	S	S

Tabela 1. Cobertura dos Requisitos de Cronogramação

Legenda:

Ferramentas – (1) GanttProject; (2) Project Open; (3) Microsoft Project; (4) Primavera; (5) RationalPlan; (6) Open Project; (7) DotProject; (8) JxProject; (9) RedMine.

Avaliação – (S) Satisfaz o requisito em questão; (N) Não atende o requisito; (P) Atende Parcialmente o requisito; (NA) O requisito em questão Não pode ser Avaliado; (NE) A informação para avaliação do requisito Não foi Encontrada.

Observações:

1. Aparentemente há como fazer isto, mas esta funcionalidade não foi encontrada, nem mesmo com a ajuda dos manuais.
2. Não é multi-projetos. Há modificações que, aparentemente, permitem o gerenciamento multi-projetos, mas são para versões antigas da ferramenta (Ganttjector e GanttProject Integrator). Além disso, uma destas modificações é paga.
3. Esta funcionalidade existe no software, embora não tenha sido percebida a possibilidade de evitar a alocação de recursos sobrecarregados (ou função similar).
4. É possível gerar apenas gráfico de Gantt.
5. Pode ser acompanhada no gráfico de Gantt que no dia corrente aparece uma linha.
6. Não foi encontrada a possibilidade de se registrar recursos além dos humanos e não é possível registrar recursos reais utilizados.
7. É possível visualizar esta informação por tarefa, mas não agregar tarefas e visualizar desta forma.
8. Isso é realizado na ferramenta por meio da possibilidade de se cadastrar o pai de uma atividade ou de um conjunto de atividades.
10. não foram encontradas evidências de que seja possível calcular o esforço selecionando atividades aleatoriamente.
11. Não foram encontradas evidências de que a ferramenta acusa quando um recurso está sendo alocado em dois projetos diferentes no mesmo período de tempo. A ferramenta não acusou conflito em um teste realizado.

Requisito	1	2	3	4	5	6	7	8	9	10	11
Estabelece relacionamentos entre cada requisito e suas fontes (Fornecedores de Requisitos associados)	NA	P	P	N	NA	NA	NA	NA	NA	S	S ⁶
Estabelece dependências entre Requisitos Funcionais (incluindo Regras de Negócio), Casos de Uso, Requisitos Não-Funcionais, Requisitos de Dados e Telas	S	P	S	S	S	S	S	S	S	S	S ⁶
Estabelece dependências entre os requisitos e os artefatos de design	S	N	S	S	S	NA	S	S	S	S ⁴	S ⁶
Estabelece dependências entre os requisitos e os artefatos de implementação	S	S	S	S	NA	NA	S	S	S	S ⁴	S ⁶
Dado um requisito, ou qualquer outro artefato, mostra as dependências do artefato em questão com o restante dos artefatos	S	N	S	S	NA	NA	S	S	S	S	S ⁶
Permite definir requisitos e casos de uso na própria ferramenta (verificar se há template para definição destes artefatos)	S ¹	S	S	P	S	S	NA	S ¹	S	S	N
Estabelece e indica dependências entre produtos gerados (dado um sistema, mostra as suas dependências com os demais sistemas)	NA	N	S	S	NA	NA	NA	NA	NA	P	S ⁷
É fácil identificar na Matriz de Rastreabilidade o requisito em questão	NA	S	S	S	NA	S	NA	NA	S	S	N
Divulga a Matriz de Rastreabilidade para todos os interessados	S	N	S	N	NA	NA	S	S	NA	S ³	N
Evidencia requisitos que não estão sendo implementados em qualquer artefato de projeto ou de implementação	NA	N	S	N	NA	NA	NA	NA	NA	N	S ⁶
Evidencia artefatos que não estão associados a requisitos	NA	N	S	N	NA	NA	NA	NA	NA	S	S ⁶
Verifica a existência de integração entre ferramentas de design e de implementação	S ²	N	S	N	NA	NA	S ³	S ²	NA	N	S ⁸

Tabela 2. Cobertura dos Requisitos de Rastreabilidade

Legenda:

Ferramentas – (1) Caliber;(2) Controla; (3) Enterprise Architect; (4) ReqManager; (5) Doors; (6) RequisitePro; (7) Nant; (8) Star Team; (9) Team System; (10) Avenq; (11) OpenShore.

Avaliação – (S) Satisfaz o requisito em questão; (N) Não atende o requisito; (P) Atende Parcialmente o requisito; (NA) O requisito em questão Não foi Avaliado; (NE) A informação para avaliação do requisito Não foi Encontrada.

Observações:

¹ No caso de usar a ferramenta Caliber DefineIT em conjunto com o CaliberRM

² No caso há integração com o Eclipse 3.4

³ No caso há integração com o VisualStudio

⁴ A ferramenta não foi feita para isso, mas é possível criar um tipo de item e anexar à ferramenta os artefatos de design criados.

⁵ Pode ser gerado e divulgado um arquivo da Matriz de Rastreabilidade. Além disso, existe meio de comunicação através da ferramenta.

⁶ Mediante busca.

⁷ Mediante busca avançada, e desde que todos os documentos envolvidos sejam importados para o OpenSHORE.

⁸ Em <http://openshore.sourceforge.net/screenshots/> são mostradas telas sobre utilização da ferramenta mediante instalação de plugins.

A ferramenta de cronogramação escolhida foi a RedMine, pois, além de atender grande parte dos requisitos também é uma ferramenta livre. Ela já foi instalada no ambiente da organização e já está sendo utilizada por alguns projetos.

3.2. Análise de Ferramentas de Rastreabilidade

Foram identificadas para análise inicial onze ferramentas que apoiam o controle de rastreabilidade de requisitos, sendo quatro livres (Controla, ReqManager, Avenço e OpenShore) e sete proprietárias (Caliber, Enterprise Architect, Doors, RequisitePro, Nant, Star Team, e Team System).

A análise detalhada de cada uma destas ferramentas, incluindo referências onde podem ser obtidas mais informações sobre elas, pode ser encontrada em [GPS, 2010]. Com relação aos requisitos estabelecidos, a Tabela 2 apresenta um resumo da cobertura apresentada pelas ferramentas.

Dentre as ferramentas inicialmente identificadas não foi possível encontrar de forma livre na Internet, ou não foram disponibilizadas pelos fornecedores, versões de demonstração para Doors, RequisitePro, e TeamSystem.

Além disso, não foi possível instalar a versão de demonstração da ferramenta Caliber, devido à inexistência, no ambiente de avaliação utilizado, da licença da plataforma necessária para esta instalação. Desse modo, a análise foi feita apenas com base na documentação sobre esta ferramenta.

Para as ferramentas Controla, ReqManager, RequisitePro e Doors não foram encontrados manuais de usuário, ou documentos similares. Adicionalmente, com exceção de Controla, ReqManager e RequisitePro, as ferramentas disponibilizaram documentos de apoio a instalação, ou puderam ser executadas sem instalação direta (tipicamente via arquivo jar).

Não foi encontrado material de treinamento, e tampouco indicativos de que os fornecedores ofereciam treinamento no local do adquirente, para as ferramentas Caliber, Controla, Enterprise Architect, Doors, RequisitePro, NANT, StarTeam e TeamSystem.

Quanto à plataforma de sistema operacional, Controla, ReqManager, Doors e RequisitePro, puderam ser executadas apenas em Ambiente Windows. Além disso, a versão Server de Caliber e StarTeam só pode ser executada no Sistema Windows Server. Já as ferramentas Enterprise Architect, NANT e TeamSystem puderam ser executadas tanto em ambiente Windows quanto em Linux.

A ferramenta de rastreabilidade escolhida foi a Enterprise Architect, pois foi a que melhor atendeu os requisitos propostos. Os procedimentos de aquisição da ferramenta ainda estão sendo executados e, portanto, ela ainda não está sendo utilizada pelos projetos do CERCOMP.

4. Lições Aprendidas

A experiência de execução de um projeto de aquisição de ferramentas para apoio a MPS permitiu o aprendizado de diversas lições, as quais são aqui sintetizadas:

- **Definição de critérios de análise.** Foi definido um conjunto de critérios que deveriam ser utilizados durante a análise de ferramentas. Alguns deles não foram devidamente avaliados porque nem sempre as informações foram encontradas, ou porque os critérios não foram bem compreendidos pelos avaliadores. Recomenda-se, portanto, que seja elaborada uma lista de critérios mais simples e concisa (em relação a que foi utilizada), porém completa em relação aos requisitos.
- **Pesquisa de ferramentas candidatas.** O processo de busca por ferramentas para análise não foi definido de maneira sistematizada. Desta forma, os responsáveis por identificar as ferramentas o fizeram de maneira ad hoc. Poderiam ter sido definidos procedimentos passíveis de repetição para a condução desta busca utilizando-se, por exemplo, de técnicas propostas para a busca de trabalhos em uma revisão sistemática [Kitchenham, 2004].
- **Escolha das ferramentas a serem analisadas.** Algumas ferramentas não foram criadas com o propósito de atender as necessidades detectadas no projeto de aquisição. Isso, entretanto, só foi detectado quando a análise da ferramenta já havia sido iniciada. Poderia ter sido definida uma lista de verificação para auxiliar na seleção inicial das ferramentas. Assim, apenas aquelas que atendessem os critérios estabelecidos nesta lista passariam para a fase de análise.
- **Dificuldade de contatar os fornecedores.** Durante a análise de ferramentas, muitas vezes fez-se necessário contatar os fornecedores a fim de obter informações, ou mesmo a versão de demonstração da ferramenta. Entretanto, muito deles não retornaram o contato e, dessa forma, pontos importantes deixaram de ser verificados na análise.
- **Dificuldade de analisar profundamente as ferramentas.** Alguns itens propostos na lista de critérios necessitam de um conhecimento avançado da ferramenta. Esse conhecimento, entretanto, só poderia ser obtido depois de certo tempo de uso. Neste caso, é importante contar com avaliadores que conhecem a ferramenta para minimizar o tempo e o esforço para sua análise.
- **Falta de material relacionado às ferramentas.** Algumas análises foram prejudicadas pela falta de documentação sobre as ferramentas. Por exemplo, faltaram manuais do usuário para poder orientar o teste de algumas ferramentas.
- **Falta de versões recentes.** Para algumas ferramentas livres não foram encontradas versões recentes. Em alguns casos fez-se necessário analisar uma versão antiga da ferramenta.
- **Análise superficial.** Em alguns casos não foi disponibilizada uma versão de demonstração da ferramenta. Nestes casos a análise foi feita com base na documentação da ferramenta, notadamente por meio de manuais de usuário ou vídeos de demonstração. Entretanto, a análise feita com versões de demonstração é muito mais completa e confiável do que aquela feita com base apenas na documentação, pois permite que o avaliador explore com maior rigor as características da ferramenta, averiguando a sua capacidade de satisfazer as necessidades da organização adquirente.

- **Existência de diferentes perfis de avaliadores.** A equipe de avaliadores do projeto contou com pessoas que possuem conhecimentos, habilidades e disponibilidades diversificadas. Isso significa que a qualidade da análise variou e, além disso, por causa de restrições na disponibilidade de alguns participantes, houve atraso na sua finalização. Para minimizar a diversidade de conhecimentos e habilidades poderiam ter sido aplicados outros métodos que não fossem baseados apenas no uso de uma lista de verificação, mas que incluíssem também técnicas de inspeção [Laitenberger, 2002]. As análises individuais poderiam ser feitas em iterações, de forma que ao final de cada iteração todos os avaliadores se reunissem para avaliar conjuntamente os resultados de cada análise individual. Cada avaliador apresentaria a análise da ferramenta pela qual ficou responsável, enquanto os demais fariam a verificação da qualidade da análise realizada. O problema deste método seria o aumento do custo das análises, mas o benefício em termos de homogeneização dos resultados seria compensatório.

5. Considerações Finais

O presente trabalho discutiu uma experiência de planejamento, análise e seleção de ferramentas de cronogramação e de rastreabilidade para apoio aos processos de software definidos para o CERCOMP/UFG. Foram identificados critérios para comparação desses tipos de ferramentas e esses critérios foram utilizados para analisar diversas ferramentas disponíveis, tanto livres quanto proprietárias. Os resultados do projeto de aquisição de ferramentas foram sintetizados na forma de lições aprendidas.

A ferramenta de cronogramação selecionada foi adquirida e está em uso pela equipe do CERCOMP há cerca de um ano. Um dos próximos passos em relação a esta ferramenta é a avaliação do quanto ela tem atendido às necessidades do CERCOMP. Já a ferramenta de rastreabilidade ainda está sendo adquirida, apesar de já ter sido realizado um treinamento no seu uso para toda a equipe.

A análise destas ferramentas foi executada no contexto de um programa de melhoria de processos de software (MPS) que está em execução no CERCOMP desde setembro de 2007. Durante estes anos de iniciativa de MPS, foi produzido um Manual de Produção de Software [GPS, 2009] que descreve atividades, padrões e diretrizes que devem ser consideradas durante um projeto de desenvolvimento ou manutenção de software no CERCOMP.

Um dos grandes desafios atuais do programa de MPS é a institucionalização dos processos de software. Espera-se que a definição e utilização das ferramentas selecionadas contribua para a superação desse desafio.

Agradecimentos

Os autores agradecem os profissionais que contribuíram para a análise de ferramentas no CERCOMP: Fernando César Silva da Mota e Hugo Alexandre Dantas do Nascimento, por meio do apoio institucional; Jánison Calixto dos Santos e Marcello Henrique, pela preparação do ambiente para avaliação das ferramentas; e Wantuir Coelho de Brito Junior, pela realização da análise de algumas ferramentas.

Referências

GPS, Grupo de Processos de Software do Cercomp (2010). Análise de ferramentas para apoio dos processos de desenvolvimento de Software. Disponível em http://www.cercomp.ufg.br/?noticia=1276613398&site_id=17. Último acesso em julho de 2010.

GPS, Grupo de Processos de Software (2009). Manual de Produção de Software do Cercomp. Disponível em <http://www.cercomp.ufg.br/uploads/files/17/manual.pdf>. Último acesso em julho de 2010.
IEEE, Institute of Electrical and Electronics Engineers (1998). IEEE 1062: IEEE Recommended Practices for Software Acquisition. Piscataway, NJ, 1998.

Kitchenham, B. Procedures for performing systematic reviews. Technical Report Keele University Technical Report TR/SE-0401, Software Engineering Group, Department of Computer Science, July 2004.
Laitenberger, O. A Survey of Software Inspection Technologies. In Handbook on Software Engineering and Knowledge Engineering, volume 2, pages 517-555. World Scientific Publishing, 2002.

Mendes, F. F., Nascimento, H. A. N. D., Fernandes, P. G., Nunes, R. S., Mota, C. C. (2010). Implantação de Melhoria de Processos em um Setor de Produção de Software de uma Universidade Federal . IX Simpósio Brasileiro de Qualidade de Software, p. 359-365.

SOFTEX, Associação Para Promoção da Excelência do Software Brasileiro. Melhoria de Processo de Software Brasileiro (MPS.BR): Guia de Aquisição: 2009. Disponível em: http://www.softex.br/mpsbr/_guias/default.asp. Último acesso em Julho de 2010.

Souza, A. S., Oliveira, J. L. (2005). Identificando Fatores Críticos para o MPS.BR através de Experiências de Implantação de Processo de Software em Goiás. ProQualiti – Qualidade na Produção de Software – Edição Especial MPS.BR. ISSN 1807-5061. Vol. 1 No. 2, p. 19-26, Novembro de 2005.

Um Estudo Experimental sobre Abordagens de Apoio à Rastreabilidade de Requisitos

Anna Beatriz Marques, Jacilane Rabelo, Sergio Costa Vieira, Tayana Conte
USES – Grupo de Usabilidade e Engenharia de Software/ Universidade Federal do Amazonas (UFAM),
Manaus – Amazonas – Brasil
{annaa.marques, jaci.rabelo}@gmail.com, sergio_ans@yahoo.com.br, tayana@dcc.ufam.edu.br

Abstract. *Traceability is a fundamental activity of Requirements Management, a process that is part of the first level of the MPS model. This paper presents an experimental study that compares different approaches to support traceability of requirements using quantitative and qualitative analysis. This article aims to contribute to the dissemination of knowledge about the execution of experimental studies to evaluate technologies that can be used in Software Process Improvement.*

Resumo. *A rastreabilidade é atividade fundamental da Gerência de Requisitos, processo que faz parte do primeiro nível do modelo MPS. Este artigo apresenta um estudo experimental que compara diferentes abordagens de apoio à rastreabilidade de requisitos utilizando análise quantitativa e qualitativa. Este artigo tem o intuito de contribuir com a disseminação do conhecimento sobre a execução de estudos experimentais para avaliar tecnologias que possam ser utilizadas em Melhoria de Processo de Software.*

1. Introdução

A Gerência de Requisitos é apresentada no modelo MPS [SOFTEX 2009] como processo a ser adotado para se alcançar o nível G, seu nível inicial, destacando sua importância para a qualidade do processo de software. Este processo permite a compreensão e controle das mudanças que ocorrem nos requisitos ao longo do processo de desenvolvimento [SOMERVILLE 2003]. Para auxiliar no controle de mudanças, é fundamental que seja estabelecida a rastreabilidade entre os requisitos e outros produtos de trabalho [SOFTEX 2009]. Apesar de reconhecida sua importância, a rastreabilidade ainda é vista como um desafio pelas empresas desenvolvedoras, pois sua complexidade tende a aumentar significativamente de acordo com o escopo do software. Assim, as empresas buscam abordagens de apoio que possam reduzir o esforço empregado na sua implementação. Existem diversas soluções disponíveis no mercado, mas optar por uma delas não é uma decisão trivial.

Na adoção de tecnologias para o processo de software, é importante ter evidências disponíveis relativas aos benefícios, funcionalidades e dificuldades apresentadas nas mesmas [CONTE *et al.* 2010]. Estudos experimentais são os “blocos de construção do conhecimento” necessários para construir evidências e determinar quais as melhores situações para empregar determinada tecnologia.

O foco deste trabalho está em analisar, através de um estudo experimental, diferentes abordagens de apoio à rastreabilidade no contexto do modelo MPS, objetivando disseminar junto à comunidade de pesquisadores em Engenharia de Software os métodos utilizados neste estudo e, ainda, fornecer subsídio às empresas em como prosseguir na escolha por abordagens de apoio mais apropriadas a seus processos.

A análise dos dados obtidos neste estudo foi feita quantitativamente e qualitativamente, buscando investigar a aceitação de cada abordagem pelos participantes do estudo, baseando-se no modelo de aceitação de tecnologia (Technology Acceptance Model - TAM) [CONTE *et al.* 2010]. Além disso, foram utilizados procedimentos baseados no método *Grounded Theory* [STRAUSS e CORBIN, 1998] para extrair as dificuldades e os níveis de aceitação das abordagens.

O restante do artigo está organizado da seguinte forma: a Seção 2 apresenta um breve referencial sobre Gerência de Requisitos com foco na rastreabilidade. A Seção 3 apresenta as abordagens de apoio à rastreabilidade analisadas neste estudo experimental. A Seção 4 descreve o processo segundo o qual este estudo foi conduzido. A Seção 5 apresenta os resultados obtidos através da análise quantitativa dos dados. A Seção 6 discute os resultados obtidos com a análise qualitativa dos dados, através do uso do modelo TAM e dos procedimentos do método *Grounded Theory*. Por fim, a Seção 7 discute as conclusões e lições aprendidas com a realização deste estudo.

2. Gerência de Requisitos e Rastreabilidade

Mudanças durante o processo de desenvolvimento de software ocorrem comumente e os requisitos são o centro dessas mudanças, pois definem o escopo do produto final de desenvolvimento [SOFTEX 2009]. Por esta razão é essencial que a gerência de requisitos seja implantada, pois é a área de processo que tem como objetivo controlar a criação e evolução destes requisitos [SOMERVILLE 2003]. É fundamental definir e manter um mecanismo de rastreabilidade para manter sob controle a evolução dos requisitos durante o processo de desenvolvimento [SOFTEX 2009].

De acordo com o modelo MPS, entre os cinco resultados esperados do processo Gerência de Requisitos, está o GRE3: “A rastreabilidade bidirecional entre os requisitos e os produtos de trabalho é estabelecida e mantida”. A rastreabilidade esperada é tanto horizontal quanto vertical. A rastreabilidade horizontal estabelece a dependência entre os requisitos ou produtos de trabalho em um mesmo nível, por exemplo, a rastreabilidade dos requisitos entre si. A rastreabilidade vertical estabelece a rastreabilidade entre diferentes níveis de produtos de trabalho, por exemplo, rastreabilidade entre requisitos e casos de uso [SOFTEX, 2009].

As organizações com iniciativas em melhoria de processo adotam diferentes abordagens de apoio à rastreabilidade na tentativa de reduzir o esforço requerido para tal atividade. A próxima seção apresenta as abordagens analisadas neste estudo.

3. Abordagens para Rastreabilidade de Requisitos

As abordagens utilizadas neste estudo foram selecionadas levando em consideração as funcionalidades necessárias para estabelecer a rastreabilidade bidirecional de forma horizontal e vertical entre os produtos de trabalho de um projeto de desenvolvimento de software, conforme requerido pelo modelo MPS. Para o estudo foram escolhidas abordagens representativas entre as opções existentes: um aplicativo que pode ser utilizado como matriz de rastreabilidade e dois softwares com funcionalidades específicas para rastreabilidade, sendo um software livre e o outro proprietário.

a) Matriz em Planilha Eletrônica - é a abordagem mais utilizada em organizações que buscam estabelecer a rastreabilidade, utiliza aplicativos de Planilha Eletrônica e consiste em construir a matriz de rastreabilidade através das células da planilha eletrônica.

b) OSRMT [SOURCEFORGE, 2010] - o OSRMT, um acrônimo para Open Source Requirements Management Tool, é um software gratuito, projetado para auxiliar na ge-rência de requisitos. Está licenciado sob os termos da GPL (*General Public License*).

c) Enterprise Architect [SPARXSYSTEM, 2010] - é uma ferramenta de análise, de-sign e desenvolvimento de aplicações em UML (*Unified Modeling Language*). Permite a modelagem de negócios, softwares e sistemas. Sua versão *trial* está disponível em *sparxsystems.com*.

4. Estudo Experimental

De acordo com Shull *et al.* (2001), o primeiro estudo que deve ser realizado para a ava-liação de uma tecnologia de software é um estudo de viabilidade, que visa avaliar se é viável empregar esta tecnologia. Neste estudo, buscou-se comparar o resultado das três abordagens selecionadas, identificando qual das abordagens apresenta melhor resultado. O processo de realização deste estudo experimental compreendeu três fases, apresenta-das na Figura 1 e descritas resumidamente a seguir.

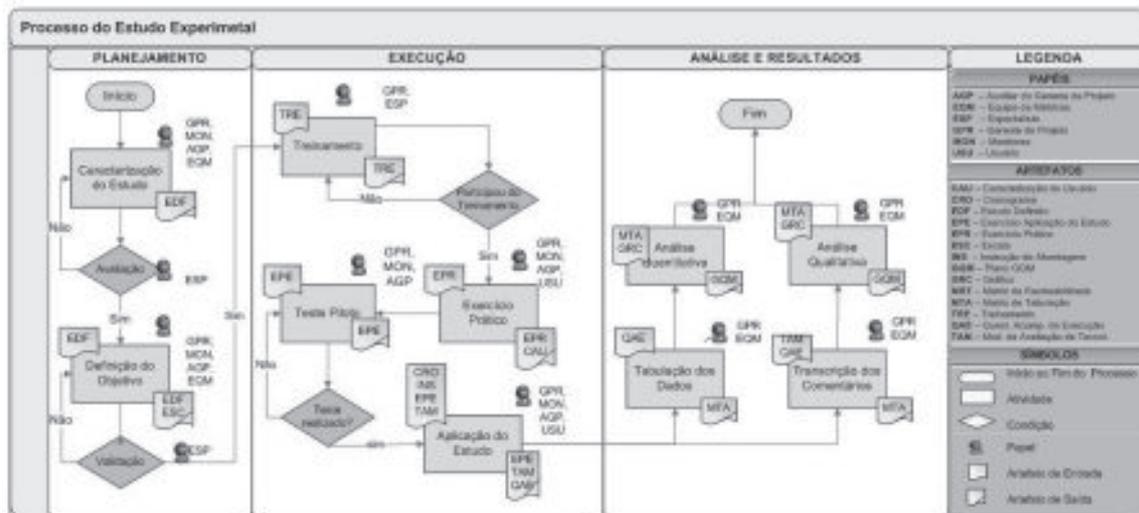


Figura 1. Processo do Estudo Experimental

Planejamento: O objetivo do estudo foi estruturalmente definido segundo o paradigma GQM (*Goal-Question-Metrics*) [BASILI *et al.*, 1988], conforme mostrado na Tabela 1. As atividades, recursos e treinamento que seriam necessários foram levantados em uma reunião com duração média de 2 horas e foram elaborados o formulário de Caracterização do Perfil e o Termo de Consentimento Livre e Esclarecido (TCLE).

Analisar	Abordagens de apoio à rastreabilidade de requisitos e produtos de trabalho
Com o propósito de	Caracterizar
Em relação à	Percepção sobre: Facilidade de Uso e Utilidade; Esforço para: criar, alterar e revisar matriz de rastreabilidade;
Do ponto de vista de	Profissionais de desenvolvimento de software
No contexto do	Processo de Gerência de Requisitos

Tabela 1. Objetivo do Estudo de Observação segundo o paradigma GQM

Execução: Os participantes do estudo foram 30 alunos do curso de Ciência da Computação da UFAM, matriculados na disciplina Engenharia de Software. Os participantes receberam treinamento sobre Gerência de Requisitos, com foco na rastreabilidade e foram informados da realização e do objetivo do estudo, sem entrar em detalhes sobre o que seria analisado. Cada abordagem foi utilizada por 10 participantes com níveis de experiência balanceados de acordo com os formulários de caracterização. Os observadores tomavam nota em formulários de acompanhamento do tempo gasto e das dificuldades apresentadas. Os participantes responderam a um questionário que seguia o modelo TAM contendo afirmações referentes à facilidade de uso e utilidade das abordagens utilizadas. Foram empregadas em média 14 horas para conclusão desta fase.

Análise e Resultados: Os dados dos formulários de acompanhamento foram analisados quantitativamente, conforme apresentado na seção seguinte. Como estas medidas quantitativas não provêm elementos suficientes para compreender os fatores relacionados às dificuldades apresentadas pelos participantes, os formulários de acompanhamento e os questionários TAM, foram analisados qualitativamente através de procedimentos da *Grounded Theory*, conforme será descrito na Seção 6.

5. Análise Quantitativa

Para avaliar o esforço empregado, a partir dos dados dos formulários de acompanhamento foram gerados gráficos de dispersão.

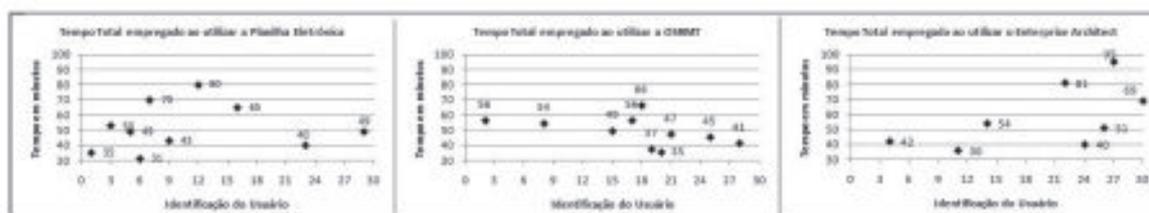


Figura 2. Gráfico do tempo total empregado pelos participantes

Na análise do tempo total empregado, dois participantes que utilizaram o EA foram desconsiderados, pois não concluíram a execução das tarefas.

Analisando a Figura 2, nota-se que dois participantes que utilizaram o EA empregaram os maiores tempos (respectivamente 95 e 81 minutos). Pôde-se observar que a média do tempo total empregado

também foi maior no EA (59 min.), em compa-ração com a OSRMT (49 min.) e a Planilha Eletrônica (52 min.). Para melhor compre-ensão do uso do tempo, foram analisadas separadamente cada tarefa e sua respectiva média de tempo empregado, conforme ilustra a Tabela 2. Assim, verificou-se que o EA apresentou a maior média apenas na TF01. A OSRMT teve a maior média na TF04, enquanto a Planilha Eletrônica teve a maior média nas demais tarefas.

Descrição da Tarefa	Planilha Eletrônica	OSRMT	Enterprise Architect (EA)
Estabelecer rastreabilidade horizontal (TF01)	19	21	22
Estabelecer rastreabilidade vertical (TF02)	14	11	13
Atualizar sistema de rastreabilidade (TF03)	13	7	8
Revisar sistema de rastreabilidade (TF04)	6	9	6

Tabela 2. Média do Tempo empregado por Tarefa

Com o intuito de compreender a razão destes resultados, foi realizada a análise qualitativa descrita na seção seguinte.

6. Análise Qualitativa

Esta análise consistiu em: (1) analisar os questionários TAM e, (2) analisar, através de *Grounded Theory*, estes questionários e os formulários de acompanhamento.

Segundo Laitenberger *et al* (1998) investigar a aceitação dos usuários para uma tecnologia requer um modelo que explique as atitudes e comportamentos das pessoas. Baseando-se no modelo de aceitação de tecnologia (Technology Acceptance Model - TAM) proposto por Davis (1989 apud CONTE

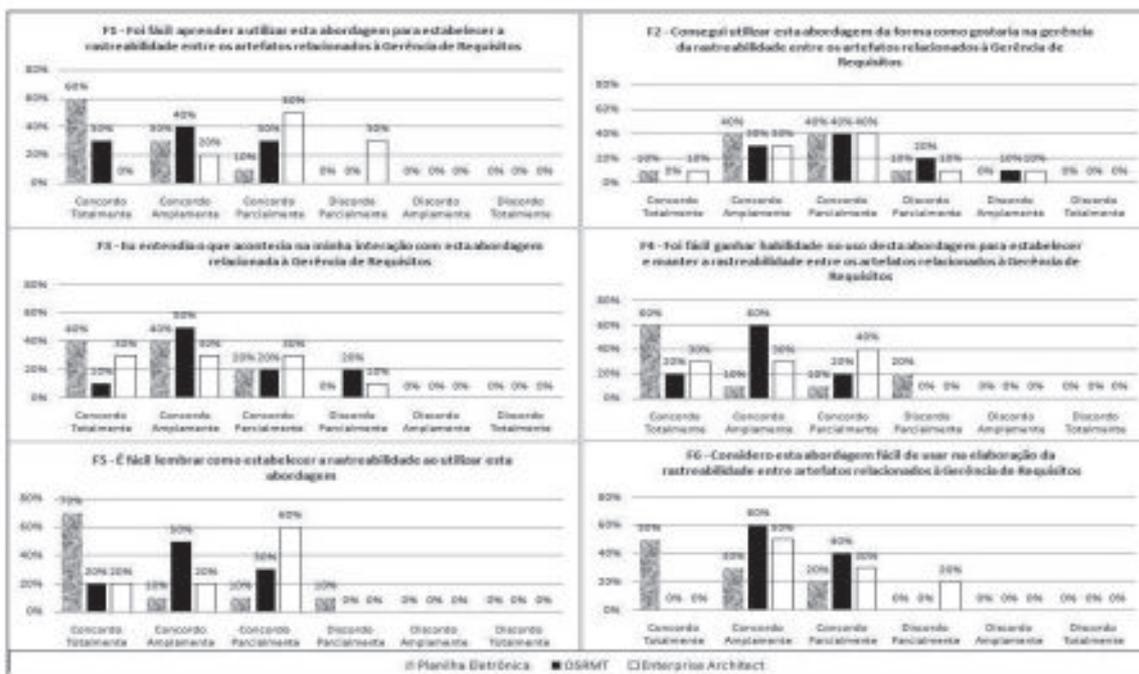


Tabela 2. Média do Tempo empregado por Tarefa

et al. 2010), utilizaram-se dois fatores: percepção sobre utilidade e percepção sobre facilidade de uso. As respostas dos participantes em relação à percepção sobre facilidade de uso estão representadas na Figura 3, nos fornecendo indícios de possíveis fatores negativos nas abordagens, pois em todas as afirmativas houve respostas discordando.

As respostas relacionadas à percepção sobre utilidade das abordagens estão re-presentadas na Figura 4.

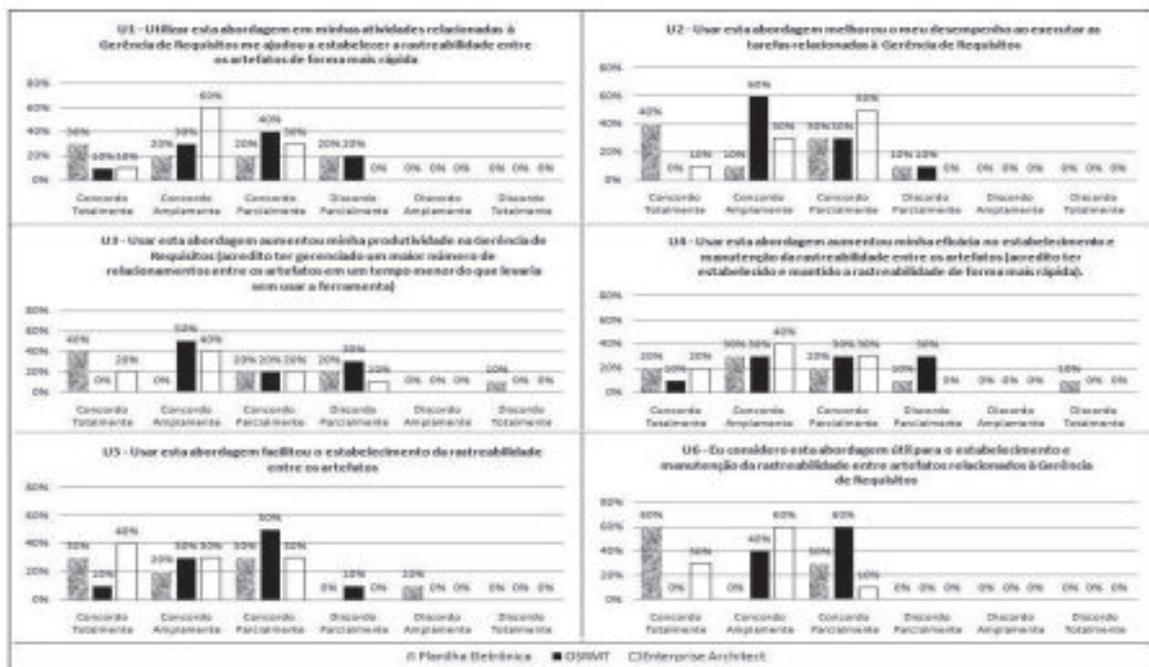


Figura 4. Respostas sobre utilidade das abordagens de apoio analisadas

Para aprofundar a compreensão dos resultados obtidos, utilizou-se procedimentos baseados no método *Grounded Theory* [STRAUSS e CORBIN, 1998], um método de pesquisa qualitativo que utiliza 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 [STRAUSS e CORBIN, 1998]. Este método tem por base a codificação, que é o processo de analisar os dados, durante o qual são identificados códigos e categorias.

A Figura 5 apresenta os códigos e categorias obtidos. Os códigos são descritos seguidos de dois números que representam respectivamente o grau de fundamentação (nro. de citações do código) e o de densidade teórica (nro. de relacionamentos do código com outros códigos). Foi criada apenas uma categoria com a qual foram associadas as subcategorias e os códigos encontrados.

A subcategoria "Dificuldades apresentadas pelos participantes" foi associada a quatro subcategorias e a estas foram associados ao todo dezesseis códigos. A subcategoria "Dificuldades Comuns a todas as Abordagens" relaciona os códigos que foram citados em todas as abordagens. Os participantes que apresentaram esforço superior à média na planilha eletrônica citaram os códigos: "Dificuldade na manipulação das células da planilha" e "Construir a matriz em planilha é trabalhoso", este último também foi citado pelos que discordaram das afirmações relacionadas à utilidade. Os demais códigos associados à subcategoria "Dificuldades na utilização da matriz em planilha" foram citados pelos que mantiveram o esforço abaixo da média.

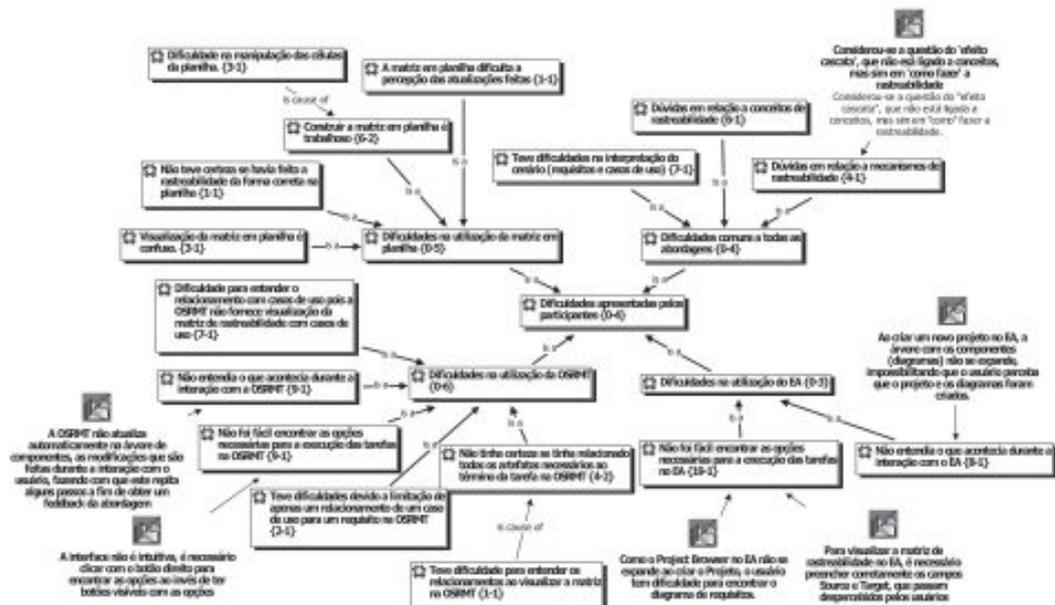


Figura 4. Respostas sobre utilidade das abordagens de apoio analisadas

Em relação à OSRMT, os que apresentaram esforço acima da média citaram to-dos os códigos associados à subcategoria “Dificuldades na utilização da OSRMT”, sendo que o código “Não foi fácil encontrar as opções necessárias para a execução das tarefas” também foi citado pelos que discordaram das afirmações sobre facilidade de uso. E os que discordaram das afirmações sobre utilidade, citaram o código “Dificuldade para entender o relacionamento com casos de uso, pois a OSRMT não fornece visualização da matriz de rastreabilidade com casos de uso”.

Dentre os participantes que utilizaram o EA, os que apresentaram esforço acima da média citaram os códigos “Não foi fácil encontrar as opções necessárias para a execução das tarefas no EA” e “Não entendia o que acontecia durante a interação com o EA”. O primeiro também foi citado pelos que discordaram das afirmações sobre facilidade de uso e utilidade.

7. Conclusões e Lições Aprendidas

De acordo com os resultados, pôde-se observar que algumas abordagens requerem treinamento apropriado para que o usuário tenha maior facilidade ao utilizá-las e, assim reduzir o esforço empregado, como o EA e a OSRMT nas quais os participantes tiveram dificuldades para encontrar as funcionalidades desejadas. Além disso, no caso do EA, por ser uma abordagem comercial, é necessário recursos para obtê-la. No caso da Planilha Eletrônica, a dificuldade está mais relacionada ao esforço na manipulação das células, o que torna a tarefa trabalhosa. Assim, orientados pela disponibilidade de recursos para tais atividades, as organizações podem optar pela abordagem mais apropriada.

Os procedimentos da *Grounded Theory* foram determinantes para a obtenção destes resultados, pois embora não seja possível generalizar os resultados desta análise, ela revelou quais os fatores relacionados às dificuldades apresentadas. A utilização do modelo de aceitação de tecnologia (TAM) contribui para avaliar o nível de aceitação das abordagens, o que pode encorajar as empresas desenvolvedoras a adotar este modelo na avaliação de abordagens de apoio a seus processos, embora algumas afirmações deste modelo exijam que o participante tenha uma base anterior para comparação.

Um dos objetivos deste artigo é compartilhar o conhecimento sobre como realizar estudos experimentais para avaliar tecnologias para melhoria de processo e Engenharia de Software de um modo geral. Embora este tenha sido um estudo *in-vitro*, cujos participantes foram alunos de graduação, o mesmo pode ser replicado em organizações de software. Ao descrever as atividades e procedimentos do estudo, espera-se incentivar a indústria de software a realizar estudos como este para apoio à tomada de decisões em casos de seleção de ferramentas ou outras tecnologias de software.

Entre as lições aprendidas para os próximos estudos, está o destaque de importância do observador, pois ele tem a responsabilidade de capturar as dificuldades apresentadas pelo participante. Portanto é importante que o observador esteja comprometido com o objetivo do estudo e disposto a acompanhar atentamente a execução das tarefas para que não afete negativamente a relevância dos dados obtidos, deixando de capturar aspectos importantes. -

Agradecimentos

Os autores agradecem a todos os participantes do estudo e aos que colaboraram durante a sua execução: Bruno Bonifácio, Davi Viana, Mário Filho, Rafael Rodolfo, Thiago Souto, Úrsula Campos e também aos revisores deste artigo por suas valiosas sugestões.

Referências

CONTE, T. *et al.* "Aplicação do Modelo de Aceitação de Tecnologia para uma Técnica de Inspeção de Usabilidade". Anais do SBQS 2010.

LAITENBERGER, O., DREYER, H.M., Evaluating the Usefulness and the Ease of Use of a Web-based Inspection Data Collection Tool, IESE, 1998.

ROCHA, A. R. *et al.* "Fatores de Sucesso e Dificuldades na Implementação de Processos de Software Utilizando o MR-MPS e o CMMI", Pro Quality, 2005.

SHULL, F., CARVER, J., TRAVASSOS, G. H., "An empirical methodology for introducing software processes." ACM SIGSOFT Software Eng. Notes, v. 26, n. 5, pp. 288-296, 2001.

SOFTTEX "MPS.BR – Guia de Implementação - Parte 1: Nível G: 2009: Fundamentação para Implementação do Nível G do MR-MPS ". Disponível em: www.softex.br, 2009.

SOMMERVILLE, Ian. Engenharia de Software. São Paulo: Addison Wesley, 2003.

SOURCEFORGE, Open Source Requirements Management Tool. Disponível em: <http://sourceforge.net/projects/osrmt/>, acessado em 01/08/2010

SPARXSYSTEM, Enterprise Architect. Disponível em: sparxsystems.com, acessado em 01/08/2010.

STRAUSS, A., CORBIN, J. (1998). "Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory". 2 ed. London, SAGE Publications.

CPqD Developer Suite

Gustavo L. de M. Chaves¹ e André Luiz de C. Villas-Boas²

¹Gerência de Tecnologia da Informação – CPqD

Rod. Campinas-Mogi-Mirim, km 118,5 – CEP 13086-902 – Campinas – SP – Brasil

²Assessoria da Qualidade – CPqD

Rod. Campinas-Mogi-Mirim, km 118,5 – CEP 13086-902 – Campinas – SP – Brasil

{gustavo,villas}@cpqd.com.br

Abstract. *In 2008 CPqD began to invest in the integration of a suite of free or cheap software development tools and in the structure of a team to support, maintain, and evolve such suite. The tools were deployed with the help of the institutional process groups, in order to guarantee its adherence to software process reference models. The new suite met the customer's expectations requiring a financial investment 20 times less than what would have been required to keep the proprietary suite previously used.*

Resumo. *Em 2008, o CPqD começou a investir na integração de uma suíte de ferramentas livres e de baixo custo e na estruturação de uma equipe para suportá-la, mantê-la e evoluí-la. O projeto contou com o apoio dos grupos de processo, garantindo a adequação aos processos institucionais de software. A nova suíte atendeu às expectativas dos usuários, exigindo um investimento financeiro 20 vezes menor do que teria sido necessário para manter a suíte proprietária de ferramentas usada anteriormente.*

1. Introdução

As atividades de desenvolvimento de software, quando realizadas por grandes equipes, exigem pessoas treinadas, processos bem definidos, ferramentas e infraestrutura de suporte adequadas. A partir da última década, a adoção de processos alinhados aos modelos de referência, como o MPS.BR (SOFTEX, 2009) e o CMMI (SEI, 2006), passou de fator de diferenciação a requisito de negócio.

Contudo, para uma pequena empresa implementar os processos básicos do modelo MPS.BR, por exemplo, estima-se que sejam necessários um tempo de maturação de um ano e um gasto de 0,5% do faturamento bruto (TRAVASSOS; KALINOWSKI, 2008). Os ganhos em produtividade advindos dessa implantação são notórios (KALINOWSKI et al., 2010), mas o tempo de retorno do investimento, medido em anos, é um ponto de consideração às vezes mais importante que o custo.

A dificuldade em adquirir e implantar ferramentas de software adequadas às necessidades e restrições das organizações é um fator que tem impacto direto na infraestrutura de processos. A definição prévia das ferramentas de apoio aos processos minimiza o retrabalho na definição dos procedimentos e diminui a resistência a mudanças na forma atual de trabalho. (MONTONI, 2008, p. 37).

Já existem iniciativas nacionais para atender às necessidades de automação dos processos de gestão (de projetos, de recursos, de conhecimento etc.), como a Estação TABA (ROCHA et al., 2005), desenvolvida pela COPPE/UFRJ. Contudo, não encontramos uma iniciativa consistente para a construção de uma suíte de ferramentas que automatize os processos operacionais de desenvolvimento de software. Este artigo contém, além desta introdução, mais cinco seções. A Seção 2 apresenta o CPqD e o problema

que a instituição enfrentava para manter uma suíte proprietária de ferramentas. A Seção 3 descreve o projeto CDS, concebido para implantar uma solução sustentável para o problema. A Seção 4 identifica os rumos da evolução futura do CDS. A Seção 5 apresenta as conclusões e as lições aprendidas durante a execução do projeto. A Seção 6 contém os agradecimentos dos autores.

2. Desenvolvimento de software no CPqD

O CPqD é uma fundação privada independente, focada na inovação em Tecnologias da Informação e Comunicação (TICs), e visa a contribuir para a competitividade do País e para a inclusão digital da sociedade. Desenvolvendo um amplo programa de pesquisa e desenvolvimento, o maior da América Latina em sua área de atuação, o CPqD gera soluções em TICs que são utilizadas nos setores de telecomunicações, financeiro, de energia elétrica, industrial, corporativo e de administração pública. Mais de 700 pessoas trabalham no CPqD em atividades diretamente ligadas ao desenvolvimento de software, o que faz do CPqD um dos maiores produtores do País. Um projeto abrangente de qualidade de software, iniciado há mais de dez anos, faz com que os processos de desenvolvimento desse produto sigam uma arquitetura baseada na norma ISO/IEC 12207 (ABNT, 1997) e sejam aderentes à norma ISO 9001 (ABNT, 2008) e ao nível 3 do CMMI-DEV v1.2 (SEI, 2006), conforme a Figura 1.

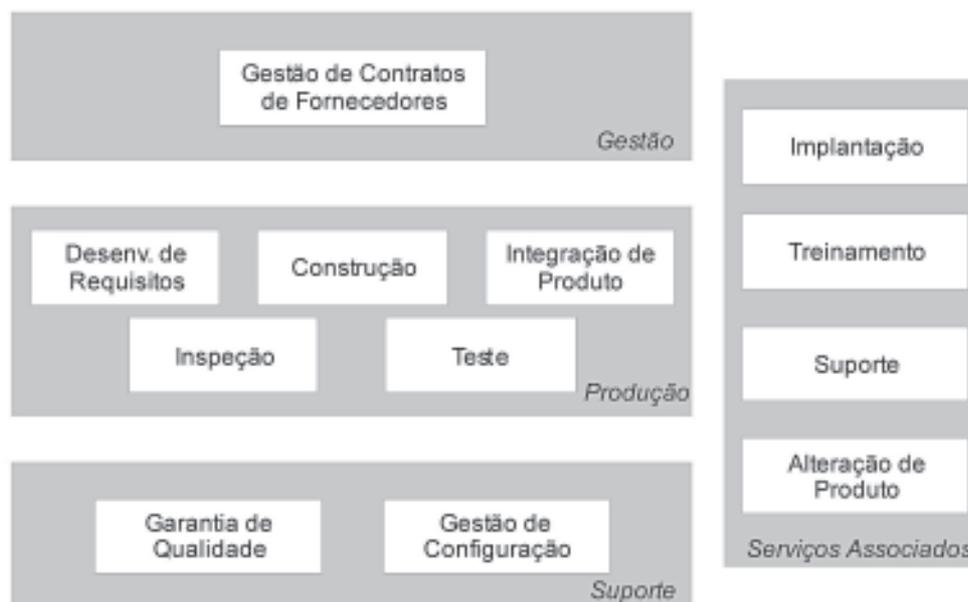


Figura 1. Processos de software do CPqD

Em 2000, o CPqD adotou uma suíte proprietária de ferramentas de desenvolvimento de software.

Em 2003, o contrato de manutenção e suporte foi suspenso, o que resultou em uma situação insustentável alguns anos mais tarde, com o número de licenças insuficiente, ferramentas desatualizadas e sem contrato de suporte para resolução de eventuais problemas. A análise da situação mostrou que a raiz do problema estava no alto custo das licenças e da manutenção da suíte, o que tornava esse contrato um forte candidato aos cortes que precisam ser feitos em tempos de crise.

Para encontrar uma solução adequada ao problema, em 2007 foram avaliados três cenários alternativos. Os dois cenários baseados na utilização de suítes comerciais indicaram que a renovação das licenças ou a adoção de uma suíte comercial equivalente custaria ao CPqD cerca de 20 milhões de reais no primeiro ano, e que a manutenção das licenças e do suporte custaria mais de quatro milhões de reais a cada ano subsequente.

A direção do CPqD apostou na alternativa de maior risco, que, no entanto, prometia uma redução de mais de 90% no investimento inicial e de mais de 80% nos anos subsequentes. A ideia foi investir na construção de uma suíte de ferramentas baseada em componentes de software livre ou de baixo custo, aliada à criação de uma equipe focada na evolução e na manutenção da suíte de ferramentas.

3. Projeto CDS

O projeto CPqD Developer Suite (CDS) ocorreu de março de 2008 a junho de 2009. O seu objetivo era implantar uma infraestrutura de ferramentas e um modelo de serviços que suportassem os processos de desenvolvimento de software do CPqD e que garantissem a sua sustentabilidade. A premissa básica do projeto era a de que deveriam ser adotadas ferramentas livres ou de baixo custo, de modo a minimizar a dependência de fornecedores e os custos de gestão de licenças, possibilitar a atualização frequente de versões e garantir o fornecimento de ferramentas e serviços sempre atualizados.

As metas do projeto foram as seguintes:

- estruturar uma equipe responsável pela evolução e manutenção do CDS;
- implantar os recursos de hardware e de software que suportariam o CDS;
- selecionar e homologar as ferramentas que constituiriam o CDS;
- definir e operacionalizar os procedimentos de provisionamento e suporte das ferramentas do CDS para os desenvolvedores do CPqD;
- disponibilizar o Portal do Desenvolvedor, um site para a consolidação de todas as informações e a documentação sobre as ferramentas;
- adaptar os processos de desenvolvimento de software ao CDS;
- migrar um grupo de projetos e usuários para o CDS.

A Equipe de Ferramentas foi estruturada na Gerência de Tecnologia da Informação, área responsável pelo suporte computacional interno do CPqD. A equipe foi composta por um time de seis pessoas, quatro das quais tiveram dedicação exclusiva, com a responsabilidade de selecionar, implantar, integrar e suportar as ferramentas da suíte, além de coordenar o projeto.

O acompanhamento do projeto foi realizado semanalmente por um grupo *ad hoc* composto por representantes das diretorias técnicas do CPqD. Esse grupo influenciava as prioridades do projeto a partir das necessidades de cada diretoria, procurando fomentar e facilitar a adoção das ferramentas dentro de suas respectivas áreas.

Algumas ferramentas foram pré-selecionadas, por já serem utilizadas no CPqD como alternativas às ferramentas da suíte comercial ou por estarem sendo usadas com sucesso pelo Instituto Atlântico de

Fortaleza¹, uma organização parceira do CPqD. A seleção das outras ferramentas e a especificação de sua configuração básica foram realizadas pela Equipe de Ferramentas com o apoio dos grupos institucionais de processo, garantindo que as ferramentas selecionadas atendessem aos requisitos básicos de automação das atividades específicas.

Atualmente, a suíte é constituída pelas seguintes ferramentas:

- **Subversion** (<http://subversion.apache.org/>): Software livre de controle de versões centralizado com grande penetração no mercado corporativo.
- **JIRA** (<http://atlassian.com/jira>): Software proprietário para gestão de mudanças, muito utilizado em projetos de software livre por oferecer uma licença gratuita para esse tipo de uso. Ele também é utilizado na gestão de requisitos e de inspeções, bem como na gestão de projetos ágeis.
- **Enterprise Architect** (<http://www.sparxsystems.com.au/products/ea/index.html>): Software proprietário de modelagem UML e de banco de dados. O EA Também foi adotado como ferramenta de rastreabilidade por oferecer a visibilidade gráfica dos relacionamentos entre requisitos, classes e tabelas.
- **Selenium** (<http://seleniumhq.org/>): Software livre para geração de testes funcionais automatizados para aplicações Web.
- **Maven** (<http://maven.apache.org/>) e **Ant** (<http://ant.apache.org/>): Ferramentas livres mais usadas na automação do processo de construção (*build*) de sistemas Java.
- **Hudson** (<http://hudson-ci.org/>): Software livre para automação de integração contínua.
- **Eclipse** (<http://eclipse.org/>): Software livre mais utilizado como ambiente integrado de desenvolvimento de software Java.
- **Archiva** (<http://archiva.apache.org/>): Software livre. Gerenciador de repositórios de artefatos do Maven.
- **Sonar** (<http://sonarsource.org/>): Plataforma livre de gestão de qualidade de código.
- **BrOffice.org** (<http://broffice.org/>): Software livre. Suíte de aplicativos de escritório composta por editores de texto, planilha, apresentação, desenhos e banco de dados.
- **Foswiki** (<http://foswiki.org/>) e **Wordpress** (<http://wordpress.org/>): Plataformas livres de *wikis* e de *blogs*, adotadas como ferramentas de colaboração.

Além da seleção e implantação das ferramentas, o projeto CDS realizou outros resultados bastante significativos:

¹ <http://www.atlantico.com.br/>

- a centralização do suporte às ferramentas trouxe benefícios tanto em relação à padronização dos procedimentos quanto à otimização dos recursos;
- a integração de algumas ferramentas exigiu o desenvolvimento de alguns módulos, alguns dos quais foram disponibilizados como software livre, como um *framework* para o desenvolvimento de *hooks* para o Subversion², e um módulo para automação do JIRA³;
- o JIRA foi adaptado para implementar alguns sistemas de informação de suporte à área de Qualidade, como o Repositório de Itens de Conhecimento (RiCo) e o Repositório de Propostas de Melhoria (RPM);
- o JIRA foi adotado na gestão de projetos baseados em metodologias ágeis;
- os grupos de processo tiveram a oportunidade de rever e simplificar alguns processos de software em função da consolidação do uso de uma só ferramenta e da melhoria geral do suporte.

Em junho de 2009, ao término do projeto inicial do CDS, uma pesquisa informal demonstrou que o nível de satisfação dos usuários com os seus resultados era muito bom. Além disso, mostrou-se alto o nível de expectativa com as evoluções futuras da suíte.

A Figura 2 exibe o número de usuários distintos que utilizam algumas das ferramentas do CDS por mês, deixando clara a forte adesão do JIRA, que, juntamente com o Subversion, tem cerca de 700 usuários ativos no CPqD. O ClearCase é a ferramenta de controle de versão que foi substituída pelo Subversion. A expectativa é que os poucos projetos que ainda dependem dela sejam migrados para o Subversion até o final de 2010. O EA é uma ferramenta utilizada por arquitetos de software e de banco de dados, e o número esperado de usuários é menor.

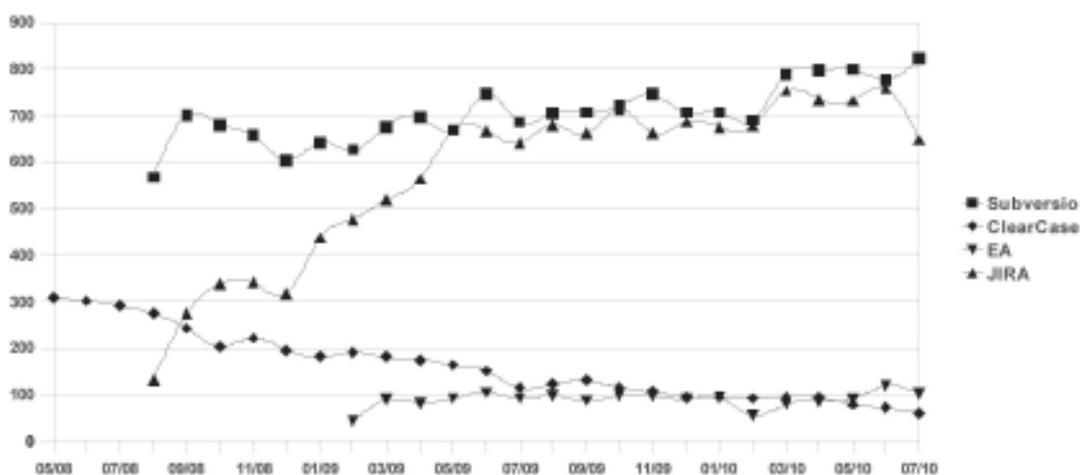


Figura 2. Evolução da adesão às ferramentas do CDS

¹ <http://www.atlantico.com.br/>

² <http://search.cpan.org/perldoc?SVN::Hooks>

³ <http://search.cpan.org/perldoc?JIRA::Client>

Vale ressaltar que durante o projeto uma diretoria técnica do CPqD foi destacada e transferida para a Trópico¹. Apesar disso, seu corpo técnico decidiu continuar a utilizar o CDS e contratar os serviços do CPqD para prestar suporte à sua utilização.

Financeiramente, o projeto CDS proporcionou uma grande economia, como se pode observar na Figura 3. Os gastos previstos no primeiro ano, relativos à aquisição das duas suítes comerciais avaliadas, ultrapassariam 20 milhões de reais. A manutenção dessas licenças consumiria quase 5 milhões de reais nos anos subsequentes. O projeto CDS, como planejado, pretendia gastar pouco mais de um milhão de reais no primeiro ano e meio milhão de reais nos anos subsequentes, basicamente para manter a Equipe de Ferramentas. O gasto realizado ficou abaixo do planejado.

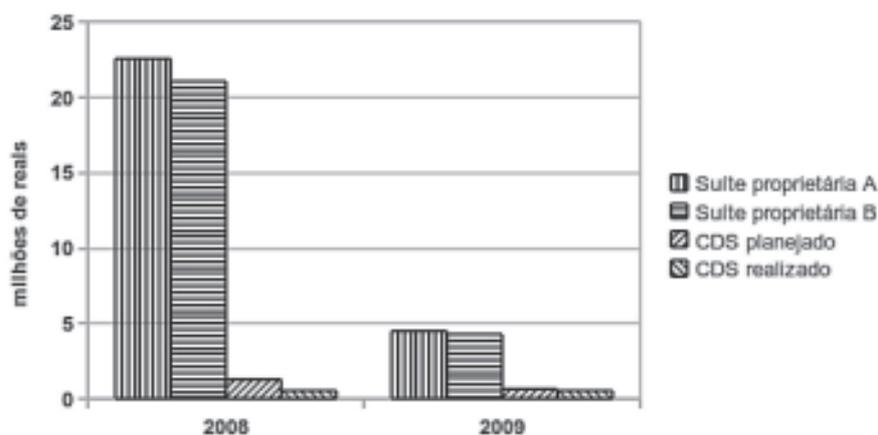


Figura 3. Resultados financeiros do projeto CDS

4. Futuro do CDS

Com o término do projeto, em junho de 2009, as ferramentas básicas do CDS e a estrutura de suporte e manutenção estavam definidas. A Equipe de Ferramentas continua trabalhando no suporte, na manutenção e na evolução dessas ferramentas, em projetos semestrais cujas metas são definidas em conjunto com o Grupo Institucional de Ferramentas, responsável pelos padrões de ferramentas de software.

Algumas atividades são constantes, como o suporte ao usuário e a atualização das ferramentas. O objetivo do CDS, em curto prazo, é produzir treinamentos para todas as ferramentas e buscar novas ferramentas para agregar valor aos processos de teste, documentação e inspeção de código.

A visão do CDS, em médio e longo prazo, passa pela concretização do conceito mais geral de integração contínua (FOWLER, 2006), automatizando a cadeia completa que começa no *commit* de uma mudança no Subversion e pelas etapas de construção, inspeção, testes, *deployment* e comunicação do resultado.

¹<http://www.tropiconet.com.br/>

5. Conclusões e lições aprendidas

Os objetivos do projeto CDS foram plenamente realizados, ou seja, conseguiu-se implantar uma suíte básica de ferramentas e estruturar um modelo economicamente sustentável para manter o seu suporte e a sua evolução. A história de problemas com a adoção de ferramentas proprietárias e de sucesso com a adoção de ferramentas livres ou baratas demonstra que é possível melhorar a qualidade geral dos processos, substituindo uma fração do investimento financeiro na aquisição de ferramentas proprietárias pelo investimento em pessoas e processos. Está claro para todos os envolvidos no projeto que, além da boa qualidade técnica das ferramentas adotadas, o esforço de uniformização de configuração e o investimento em suporte e manutenção foram elementos-chave, sem os quais os resultados não teriam sido bons.

De fato, conjecturamos que o alto investimento na aquisição de uma suíte “pronta” iniba o investimento em pessoas e em processos, o que cria um ambiente propício ao surgimento de procedimentos divergentes e ferramentas alternativas. Ao se restringir a seleção de ferramentas de acordo com o custo, torna-se mais óbvia a necessidade de se investir na estruturação de uma equipe interna dedicada ao suporte e à manutenção da suíte, possibilitando um melhor balanceamento do tripé ferramentas, pessoas e processos.

Um projeto de risco como esse é bastante suscetível às barreiras culturais e à natural aversão a mudanças, e certamente não teria tido sucesso sem um conjunto de fatores devidamente alinhados:

- o apoio firme e constante da alta gerência garantiu a participação ativa das diretorias técnicas;
- a participação dos grupos de processo foi decisiva para garantir a melhor configuração das ferramentas e a estruturação adequada do suporte;
- o acompanhamento semanal do projeto, feito pelos representantes das diretorias, garantiu o fluxo constante de informações, facilitando e estimulando a adoção do CDS pelos projetos em andamento;
- a estruturação da Equipe de Ferramentas, como um time coeso e dedicado, garantiu a qualidade do suporte e da manutenção das ferramentas.

Contudo, apesar do sucesso do Subversion e do JIRA, a adoção das outras ferramentas ainda precisa ser consolidada. Faltam também ferramentas importantes na área de testes e de inspeção, além de um conjunto completo de treinamento em vídeo.

O CDS é uma solução real e viável para as atividades relacionadas ao desenvolvimento de software em uma organização como o CPqD. É composto por um conjunto de ferramentas de software livre ou de baixo custo que podem ser adotadas por pequenas e médias empresas porque não exigem grande investimento. Toda a documentação, configuração e integração desenvolvida pelo projeto pode ser disponibilizada livremente para outras instituições que tenham interesse em usar e colaborar na evolução do CDS.

6. Agradecimentos

O sucesso do projeto CDS se deve ao trabalho e ao apoio de dezenas de pessoas em todo o CPqD. O apoio do patrocinador, Antonio Vivaldi, foi fundamental para vencer as resistências naturais à mudança. Os facilitadores, Maurício Vianna e Mario Cortes, garantiram os recursos necessários à execução do projeto. Os grupos de processo institucionais auxiliaram na seleção e na configuração das novas ferramentas.

Agradecemos especialmente aos colegas da Equipe de Ferramentas do CPqD: André Giraldi, Andreyev D. de Melo, João S. O. Bueno, Luciana A. R. Levanteze e Mario Celso Teixeira.

Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS (ABNT). NBR ISO/IEC 12207: Tecnologia de Informação – Processos de Ciclo de Vida de Software. ABNT, maio de 1997.

_____. NBR ISO 9001: Sistemas de Gestão da Qualidade – Requisitos. ABNT, dezembro de 2008.

FOWLER, M. Continuous Integration. 2006. Disponível em: <<http://martinfowler.com/articles/continuousIntegration.html>>. Acesso em: 11 aug. 2010.

KALINOWSKI, M. et al. MPS.BR: Promovendo a Adoção de Boas Práticas de Engenharia de Software pela Indústria Brasileira. In: CIBSE – VIII CONGRESSO IBERO-AMERICANO SOBRE ENGENHARIA DE SOFTWARE, Universidad del Azuay, 2010, Cuenca, Ecuador. Anais... Disponível em: <http://www.softex.br/mpsbr/_artigos/artigo.asp?id=3012>. Acesso em: 28 set. 2010.

MONTONI, M. Lições Aprendidas com Implementação do Modelo MPS em Empresas. In: MPS.BR: Lições Aprendidas. Campinas, SP: Associação para Promoção da Excelência do Software Brasileiro – SOFTEX, 2008. Disponível em: <http://www.softex.br/mpsbr/_livros/licoes/mpsbr_pt.pdf>. Acesso em: 28 abr. 2010.

ROCHA, A. R. et al. Estação TABA: Uma Infra-estrutura para Implantação do Modelo de Referência para Melhoria de Processo de Software. In: SBQS'05 – SIMPÓSIO BRASILEIRO DE QUALIDADE DE SOFTWARE, 4., 2005, Porto Alegre. Anais... Porto Alegre, Brasil, 2005.

SOFTEX. MPS.BR – Guia Geral: 2009. Disponível em: <<http://www.softex.br>>. Acesso em: 28 set. 2010.

SOFTWARE ENGINEERING INSTITUTE (SEI). CMMI for Development, version 1.2. USA: Carnegie Mellon University, 2006. Disponível em: <<http://www.sei.cmu.edu/library/abstracts/reports/06tr008.cfm>>. Acesso em: 12 aug. 2010.

TRAVASSOS, G.; KALINOWSKI, M. iMPS – Resultados de Desempenho de organizações que adotaram o modelo MPS. Campinas, SP: Associação para Promoção da Excelência do Software Brasileiro – SOFTEX, 2008. Disponível em: <http://www.softex.br/mpsbr/_livros/imps/imps.pdf>. Acesso em: 12 ago. 2010.

Brechó 2.0: Uma Ferramenta para Apoiar a Gerência de Reutilização

Rodrigo Santos¹, Anderson Marinho¹, Marlon Silva¹, Cláudia Werner¹, Leonardo Murta²

¹ COPPE/UFRJ, Universidade Federal do Rio de Janeiro, Brasil

² Instituto de Computação, Universidade Federal Fluminense, Niterói, RJ, Brasil
{rps,andymarinho,marlon,werner}@cos.ufrj.br, leomurta@ic.uff.br

Resumo. *Bibliotecas de componentes e serviços de software são importantes para alcançar as metas de custo efetivo e de produtividade em Reutilização. Estas bibliotecas precisam fornecer mecanismos diversificados para apoiar a gerência de serviços e de componentes, explorando aspectos relativos à organização, evolução e movimentação de artefatos, e características de um ambiente subjacente. Este artigo apresenta a Brechó 2.0, uma evolução da biblioteca Brechó para apoiar o processo de gerência de reutilização.*

1. Introdução

A incorporação de aspectos econômicos ao desenvolvimento de software foi apontada como um dos desafios da Engenharia de Software (ES) para os próximos anos [1] e a Reutilização se apresenta como uma disciplina estratégica [2]. Nesse sentido, o Projeto Brechó [3] foi criado para pesquisar tópicos relacionados a repositórios de componentes, a fim de desenvolver a biblioteca Brechó [21]. A versão 1.0 consiste de um sistema de informação Web com uma base de dados de componentes, fornecedores e consumidores, e com mecanismos de armazenamento, publicação, busca e recuperação.

No entanto, a existência de um repositório por si só não promove a reutilização [13]. Um pré-requisito para se explorar o seu potencial é a emergência de um mercado ou ambiente colaborativo de componentes [20] apoiado por um processo de gerência de reutilização, bem como a disponibilização de componentes como serviços [19]. Visando evoluir a biblioteca Brechó do estereótipo de “implementação de um repositório” para “implementação de um ambiente de reutilização”, este artigo apresenta a Brechó 2.0, uma ferramenta para apoiar processo de gerência de reutilização. A Seção 2 apresenta a evolução da Brechó 1.0 para a Brechó 2.0; a Seção 3 discute a aderência do processo de gerência de reutilização do Modelo MPS à Brechó 2.0; os trabalhos relacionados são expostos na Seção 4; e a Seção 5 conclui o artigo apresentando alguns trabalhos futuros.

2. Evolução da Biblioteca Brechó

Na versão 1.0, a Brechó apresenta mecanismos de publicação e documentação que consideram um conceito flexível de componente, que inclui todos os artefatos produzidos no desenvolvimento, permitindo a aquisição de conjuntos de artefatos em-pacotados a partir de licenças personalizadas e configuráveis [21]. A estrutura de documentação é baseada em categorias e formulários dinâmicos e configuráveis a elas associados. Isso possibilita buscas por categorias ou por palavras-chave. A Brechó também gerencia um mapa de reutilização (contratos entre produtores e consumidores

para o uso de componentes) e o esta-belecimento e infe-rência de dependências entre componentes. Novas características foram adicionadas com o intuito de evoluir a sua es-trutura rumo à implementação de um mercado ou ambiente de reutilização, o que origi-nou a *Brechó 2.0*. A organiza-ção interna da biblioteca foi modificada para contemplar novos produtos (ser-viços) [21]. Com isso, após cadastrar um *componente* e respec-tiva *distribui-ção*, a partir do nível de *release*, os compo-nentes passam a ter infor-mações concretas sobre seus arte-fatos em diferentes ní-veis de abstração e estes podem ser agrupados em diferentes *pa-cotes* ou disponibilizados na forma de *servi-ços*, atrelados a *licenças* especí-ficas.

Além disso, foram implementados: (i) mecanismos de dependência física (entre *licenças*) e lógica (entre *releases*) de com-ponentes; (ii) me-canismos de controle de pu-blicação de componentes pelo administrador da biblioteca, de acordo com o processo de gerência de reutilização; (iii) mecanismos para facilitar a publicação de compo-nentes, incluindo conceitos de entidades *default* (devido à organi-zação in-terna) e de empacota-mento automático de artefatos; e (iv) desen-volvimento de uma API de serviços para acesso às funcionalidades da biblioteca para sua integração com outras ferramentas. Por fim, em sua arquitetura, a *Brechó 2.0* apresenta um núcleo básico (*Brechó 1.0*) e acres-centa uma infra-estrutura dividida em três partes: (i) avaliação, busca e clas-sificação; (ii) apoio a SOA; e (iii) apoio ao mer-cado de compo-nentes. Por sua vez, cada uma dessas par-tes é composta por mecanismos e módulos de apoio, conforme as Seções 2.2-2.4.

2.1. Implementação

A *Brechó 2.0* é uma aplicação *Web* implementada com a tecnologia *Java J2EE*, usando o *framework Struts* para implementar uma arquitetura que segue o padrão MVC. Para a persistência de objetos, foi utilizado o *framework Hibernate* e o banco de dados *MySQL*. Com relação à infra-estrutura de serviços, o componente *Apache Axis2* foi usado tanto na construção da API de serviços quanto no mecanismo de publicação e de geração auto-mática de serviços da *Bre-chó 2.0*. Além disso, o componente *Apache ODE* foi utili-zado para apoiar a construção e o provimento de serviços compostos. Por se tratar de um pro-tótipo de pesquisa, questões relacionadas a escalabilidade e segurança deveriam ser alvos de avaliação, uma vez que ainda não foi executado um estudo de caso formal na indús-tria para verificar e validar a ferramenta como um produto de mercado.

2.2. Infra-estrutura de Apoio à Avaliação, Busca e Categorização

A *Brechó 2.0* apresenta uma infra-estrutura integrada por mecanismos avançados de avaliação, busca e categorização de componentes e serviços. O *mecanismo de avaliação* é dividido em dois módulos: (i) módulo de avaliação de componentes, que permite ao consumidor emitir sua opinião a respeito do componente que ele recuperou da biblioteca (grau de satisfação e *feedback* textual); e (ii) módulo de visualização de avaliações, que per-mite a consumidores e a produtores observar graficamente (percentual e opiniões) as avaliações feitas por consumidores. O *mecanismo de busca* é composto por dois mó-dulos: (i) módulo de refinamento e flexibilização de buscas, que possibilita o uso combi-nado das técnicas de busca por palavras-chave e por categorias, e ainda reali-zar a filtra-gem dos componentes por elementos presentes em sua documentação; e (ii) módulo de busca sintática, que alerta aos usuá-rios sobre prováveis erros de digitação no mo-mento da busca por palavras-chave e sugere palavras com grafias similares.

O *mecanismo de categorização* apresenta dois módulos: (i) módulo de sugestão de categorias, que permite que o produtor, ao publicar um componente, sugira uma categoria de forma manual e/ou semi-automática, com base em sugestões da *Web*; e (ii) módulo de organização de categorias e sugestões, que apóia o administrador na tarefa de reorganizar as categorias ao realizar pesquisas na *Web* para identificar si-milaridades entre sugestões e categorias existentes, além de notificar produtores cuja sugestão foi aceita ou que possuem componentes em categorias passíveis de exclusão. Um refinamento desse mecanismo, o *mecanismo de hierarquização de categorias*, utiliza o conceito de grafos acíclicos direcionados (*Directed Acyclic Graphs*) para apoiar a existência de *super* e *subcategorias* (similares a *classes* em orientação a objetos), em duas situações, dependendo da classificação adotada: uma (*sub*) categoria pode pertencer a mais de uma (*super*) categoria (de nome único na biblioteca), ou pode-se trabalhar como diretórios em sistemas de arquivos, onde categorias com o mesmo nome em diferentes hierarquias são distintas. O mecanismo ainda permite duas visualizações: *horizontal* (visão linear e na-vegacional das categorias hierarquizadas, re-latva à *manutenção de categorias*) e *vertical* (visão global e em forma de árvore, relativa à *manutenção de componentes*) [14].

2.3. Infra-estrutura de Apoio à Arquitetura Orientada a Serviço

A *Brechó 2.0* possui uma infra-estrutura que disponibiliza a publicação e utilização de serviços provendo *mecanismos de controle de acesso* e *de registro de uso*, que permitem que a biblioteca participe não somente da etapa de descoberta dos serviços, mas da sua utilização, similarmente ao que ocorre com componentes. Três estratégias de publicação de serviços foram definidas: (i) serviços externos, para serviços já existentes; (ii) serviços internos simples, para serviços gerados a partir de componentes previamente publicados na *Brechó 2.0*; e (iii) serviços internos compostos, para serviços constituídos da composição de serviços externos e internos simples ou de outros compostos. A primeira estratégia assume que os produtores já possuem serviços hospedados em um servidor e querem apenas publicá-los na *Brechó 2.0*. A segunda e terceira estratégias assumem que os produtores não possuem recursos para prover serviços e necessitam que a *Brechó 2.0* realize esse papel. Na segunda estratégia, porém, os serviços são gerados a partir de componentes previamente publicados, ao passo que, na terceira estratégia, os produtores constroem as especificações dos serviços compostos (a partir da linguagem de composição de serviços *BPEL*) e publicam na biblioteca para que esta realize a geração, hospedagem e orques-tração dos mesmos [10]. Na criação de serviços internos, a *Brechó 2.0* faz uso de reflexão para detectar, em função da linguagem utilizada, as operações disponíveis no componente que podem ser disponibilizadas como serviços (Figura 1.a).

Além disso, essa infra-estrutura provê uma API de serviços das próprias funcionalidades da *Brechó 2.0*, permitindo que usuários possam usar a biblioteca remotamente e que possíveis integrações com outras ferramentas possam ser realizadas. Um exemplo dessas ferramentas é o ambiente *Odyssey* [11] que utiliza a *Brechó 2.0* para armazenar e recuperar *plugins* que serão carregados em tempo de execução no ambiente [6].

2.4. Infra-estrutura de Apoio a um Mercado de Componentes

A *Brechó 2.0* apresenta uma infra-estrutura que estende a *Brechó* para uma biblioteca de referência para um mercado ou ambiente colaborativo baseado em valor [14]. Essa infra-estrutura foi desenvolvida

a partir do *mecanismo de tarifação*, responsável por coordenar diferentes formas de aquisição ou uso de artefatos ou serviços de componentes publicados. O mecanismo de tarifação combina os modelos *pré* e *pós-pago* do domínio de telecom e alguns conceitos do domínio bancário, no qual usuários têm um *limite de crédito* para gastar em compras, mas eles podem comprar *créditos* para compensar o débito (modelo pós-pago) ou para acumular créditos (modelo pré-pago). Esta informação é configurada em cada *conta de usuário* pelo administrador da biblioteca. Além disso, a *Brechó 2.0* coordena o processo de *transferência de créditos* durante a recuperação de componentes e serviços, utiliza o conceito de *carrinho de compras* para auxiliar o consumidor e permite a visualização de um *histórico de compras*. O mecanismo de tarifação adotado é composto pelo *mecanismo de precificação*, que visa analisar como o valor de componentes e de serviços pode ser percebido de forma quantitativa, considerando perspectivas distintas (*produtores* e *consumidores*) e uma instância de cadeia de valor (*precificação* e *aquisição* de artefatos) [15]. Devido ao conceito de *crédito*, a *Brechó 2.0* apóia cenários *colaborativos* e *com-petitivos*, além de auxiliar na precificação de componentes ao considerar informação extraída de ações de produtores e consumidores.

Entre os mecanismos desenvolvidos, denominados gerentes de negócio (semi-) automatizados, estão: precificação, marketing, avaliação e negociação [14]. A Figura 1.b ilustra o modelo de precificação de mercado do mecanismo de precificação, o qual realiza uma busca por artefatos similares a fim de prover uma análise de mercado

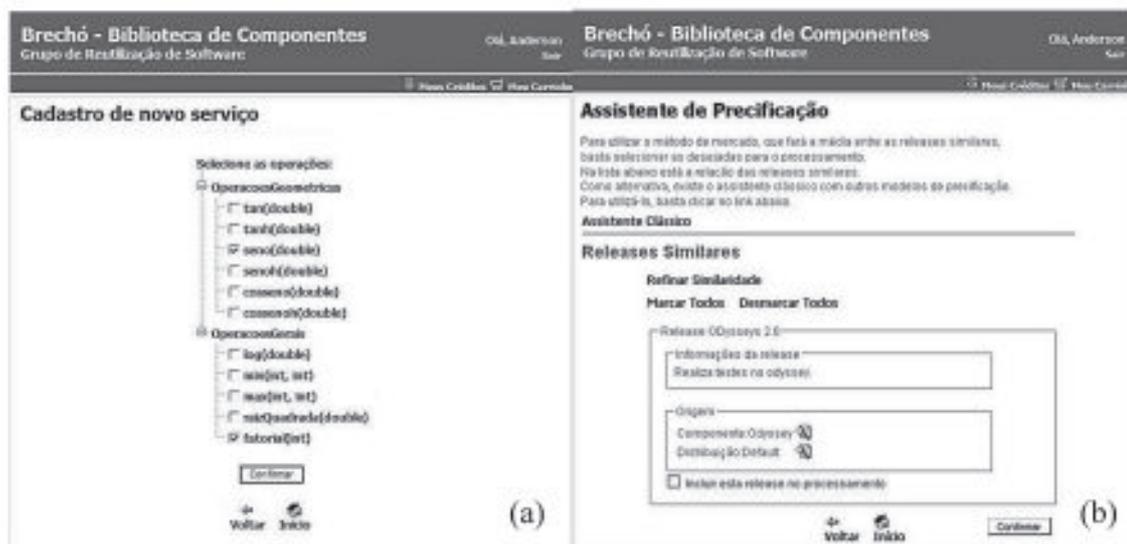


Figura 1. Mecanismo de Apoio a Serviços (a) e Mecanismo de Precificação (b)

Uma análise da infra-estrutura em relação à sua aderência às orientações para implementação de um processo de reutilização mostra que a sua concepção pode apoiar o Processo de Gerência de Reutilização (GRU) do Modelo MPS [18], que tem o propósito de gerenciar o ciclo de vida dos ativos reutilizáveis e que apresenta cinco resultados esperados. Assumindo a definição de componente como o encapsulamento de um conceito, deve-se estabelecer e manter uma estratégia para o gerenciamento desses ativos, bem como do conjunto de seus artefatos, assim como o ferramental que torne este procedimento viável. Isso vai ao encontro do resultado esperado GRU1, por contemplar

a definição do que deve, então, ser um ativo reutilizável para a biblioteca *Brechó 2.0*. Esta definição pode possuir uma semântica mais ampla ou especializada, devido à flexibilidade oferecida pela *Brechó 2.0* de ser tratada como um catálogo para componentes de diferentes repositórios distribuídos.

Devem ser definidos critérios que regem a manutenção e a qualidade do conteúdo armazenado na *Brechó 2.0*. Esta definição deve refletir as proposições de valor do gerente de ativos reutilizáveis e dos produtores, minimamente e dinamicamente, ou seja, não pode ser um documento de caráter global nem estático, e deve ser periodicamente revisto. Assim, os critérios correspondem às exigências para verificar e validar a aceitação, a certificação, a classificação, a descontinuidade e a avaliação dos ativos. Para definir estes critérios, *workshops* podem ser realizados, presenciais ou à distância, e uma ferramenta do tipo editor de texto pode ser utilizada. Com este documento, pode-se criar um fórum de discussão (e.g., GoogleGroups) ou uma categoria na *Brechó 2.0* visando divulgá-los, de forma a torná-los explícitos e colaborar para a confiabilidade através da transparência. Esse procedimento independe da *Brechó 2.0* enquanto ferramental, mas pode utilizar as informações extraídas de dados históricos, manipuladas e visualizadas [16], tais como perfis dos produtores mais ativos, oferta e demanda de ativos, e gráficos de categorias com alto volume de transações, com grande fluxo de publicação de componentes e volume de aquisição por componente etc. Dessa forma, parte-se para o estabelecimento e manutenção de um mecanismo de armazenamento e recuperação dos ativos reutilizáveis e, neste ponto, a *Brechó 2.0* atende ao resultado esperado GRU2.

Além disso, a *Brechó 2.0* apresenta um conjunto de gerentes de negócio (semi-) automatizados (Seção 2.4) que apoiam a sua manutenção por auxiliarem a verificação de inconsistências com relação aos critérios estabelecidos (e.g., verificação de componentes mal avaliados e pouco adquiridos, que rompem critérios de avaliação). Estes gerentes provêem condições para que decisões sejam tomadas a respeito da situação dos ativos reutilizáveis na base. A *Brechó 2.0*, funcionando como a base de ativos reutilizáveis, dá suporte à definição da frequência dessas verificações ao ser parametrizável em suas unidades temporais. Ela permite, ainda, que os produtores externem necessidades de catalogação de certos ativos através do campo de sugestões de demanda específica do mecanismo de avaliação (i.e., mecanismo para apoiar a avaliação de componentes via graus de satisfação e justificativas textuais), e do módulo de sugestão de novas categorias para classificação de componentes [16].

Por fim, uma vez definida essa base de ativos reutilizáveis, a *Brechó 2.0* apoia a realização das atividades e das relações entre produtores e consumidores, o que contribui diretamente para os resultados esperados pelo GRU3, pelo GRU4 e pelo GRU5, que tratam da manutenção do mapa de reutilização (i.e., contratos entre produtores e consumidores sobre o uso de componentes), da manutenção e controle do ciclo de vida dos ativos, e da notificação de *stakeholders* sobre suas ações, respectivamente.

Na perspectiva do produtor, a publicação de ativos reutilizáveis engloba os procedimentos para que determinado ativo (aceito e certificado após critérios verificados) seja incluído na *Brechó 2.0*. Entre os procedimentos requeridos e atendidos pela biblioteca, estão:

- (1) *Solicitação de publicação*: procedimento realizado pelo produtor, que preenche um formulário informando o nome, a descrição e as categorias do ativo a ser publicado mediante uso do documento de definição de ativos reutilizáveis. Esta solicitação gera uma tarefa para o gerente de ativos reutilizáveis;

(2) *Análise de solicitação*: com base nos critérios de aceitação, o gerente de ativos reutilizáveis analisa o ativo a ser publicado, rejeitando-o ou aceitando-o, e a *Brechó 2.0* notifica o produtor sobre o estado do ativo via *e-mail*;

(3) *Classificação do ativo reutilizável*: após o ativo ter sido aceito, o gerente de ativos reutilizáveis analisa a classificação fornecida pelo seu produtor, utilizando os critérios de classificação. Eles podem acionar o módulo de organização de categorias e sugestões, a fim de verificar se devem cadastrar novas categorias sugeridas pelos produtores e/ou reestruturar as categorias existentes. Assim, o ativo passa do estado *aceito* para *publicado*. Dado que o formulário de solicitação de publicação de ativos na *Brechó 2.0* já requer a sua classificação, os procedimentos (2) e (3) são executados de uma só vez e cor-respondem à mudança do estado *não avaliado* para o estado *aceito*;

(4) *Solicitação de inclusão de release*: com a efetivação dos procedimentos anteriores, o produtor pode fazer o *upload* dos artefatos que compõem a *release* do ativo aceito e publicado na *Brechó 2.0*. Para isso, o produtor deve acessar a entidade componente e cadastrar uma nova distribuição (ou utilizar a distribuição *default* pré-cadastrada) para, então, cadastrar a *release*, agrupando seus artefatos em pacotes específicos ou mantendo o pacote *default* (completo), seguindo a organização interna da *Brechó 2.0* (Seção 2);

(5) *Certificação de Release*: como a *Brechó 2.0* não apresenta o estado *aguardando certificação*, pois o processo de publicação poderia se tornar muito burocrático, a certificação é realizada quando o produtor decide tornar pública a *release* cadastrada. Dessa forma, o gerente de ativos reutilizáveis utiliza o documento dos critérios de certificação para verificar a *release* e notificam o produtor, caso esta seja reprovada no procedimento. Isso é apoiado por outros processos, como a Garantia de Qualidade e a Gerência de Configuração [18], visando atender ao GRU4;

(6) *Notificação de interessados*: como consequência de aplicar os critérios definidos a um ativo reutilizável ou a alguma *release*, produtores e consumidores são notificados a respeito de problemas detectados, modificações realizadas (corretivas ou evolutivas, cuja substituição é requerida ou opcional, respectivamente), novas versões disponibilizadas e descontinuidade de ativos. Assim, a *Brechó 2.0* atende ao GRU5 ao utilizar notificações via *e-mail*, divulgando inclusive componentes de interesse do consumidor.

Na perspectiva do consumidor, por sua vez, a aquisição (ou consumo) de ativos reutilizáveis engloba os procedimentos necessários para que determinado ativo seja recuperado da *Brechó 2.0*. Entre os procedimentos requeridos e atendidos, estão:

(1) *Busca de ativos reutilizáveis*: o consumidor utiliza os mecanismos de busca da *Brechó 2.0* (palavras-chave e categorias hierarquizadas) e aplica filtros extraídos da documentação dos ativos e das categorias comuns;

(2) *Recuperação de ativo reutilizável*: a partir do ativo selecionado, o consumidor seleciona uma distribuição, uma *release* e um pacote de interesse para realizar o *download*, bem como a licença de uso apropriada. Após o *download*, um mapa de reutilização é gerado ou atualizado, registrando o consumidor que recuperou a *release* na *Brechó 2.0*, o que atende ao GRU3. Isso

auxilia a realização da manutenção e evolução da biblioteca por seu gerente (e.g., identificação de tendência que pode ajudar a calibrar os critérios estabelecidos), além de prover os gerentes de negócio com dados históricos para notificações e/ou recomendações;

(3) *Adaptação de ativo reutilizável*: após adquirir o ativo, o consumidor analisa e documenta as necessidades de adaptação que tornem este ativo adequado para sua utilização (e.g., desenvolvimento de adaptadores para os componentes se adequarem às interfaces existentes, e formatação de um documento de plano de projeto segundo os padrões adotados pelo consumidor). A *Brechó 2.0* não auxilia o consumidor com um ferramental de apoio (editor para adaptação de artefatos), mas fornece um canal de comunicação com os produtores e com o gerente de ativos reutilizáveis, via *e-mail*. Assim, a implantação de um processo de Projeto e Construção do Produto é interessante para o consumidor [18];

(4) *Integração de ativo reutilizável*: correspondendo a um procedimento próprio do consumidor, como o anterior (3), testes devem ser planejados e executados para verificar a adequação do ativo reutilizável adaptado ao seu processo de desenvolvimento. Nova-mente, a *Brechó 2.0* não apóia o consumidor com um ferramental (editor de integração). Neste caso, os artefatos do pacote passam por um ciclo entre (3) e (4), de modo que os problemas sejam consolidados em um documento de integração;

(5) *Avaliação de Ativo Reutilizável*: de posse do documento de critérios de avaliação propostos, os consumidores podem avaliar os ativos que foram recuperados, adaptados e integrados. Esta avaliação, opcional, consiste em atribuir um grau de satisfação que explicita a opinião do consumidor, além de comentários textuais, conforme provido pelo mecanismo de avaliação da *Brechó 2.0* [14].

Percebe-se que a infra-estrutura *Brechó 2.0* contempla os resultados esperados pelo processo GRU, tratando questões técnicas do processo. Isso se deve ao fato da biblioteca (semi-) automatizar alguns gerentes de negócio e apoiar o gerente de ativos reutilizáveis a visualizar informações que possibilitem mediar a evolução dos documentos sobre o conceito de ativo e sobre os critérios relacionados. Existem pontos que foram adaptados (e.g., a amenização da burocracia envolvida na publicação de ativos), ou mediados por mecanismos de comunicação e coordenação da biblioteca (e.g., procedimentos de adaptação e integração de ativos).

Trabalhos Relacionados

Existem bibliotecas de componentes disponíveis para uso na *Web* [4] [5] [7] [12] [22], caracterizadas por catalogar componentes já empacotados ou de prateleira. Logidex [9] e Select Asset Manager [17] são gerenciadores comerciais que mantêm informações sobre os componentes reutilizáveis de uma organização. O Logidex apóia a publicação de serviços, mas não a sua geração e controle de acesso. O trabalho de Lee *et al.* [8] apóia a geração de serviços, porém não se preocupa com o controle de acesso, além de ser limitado para componentes Java e C++. As abordagens apresentadas focam em combinar mecanismos de busca, classificação e recuperação, mas carecem da junção de mecanismos de precificação, de exploração de dados históricos e de publicação de serviços.

Conclusões

Este artigo apresentou a *Brechó 2.0*, uma evolução da biblioteca Brechó para apoiar processo de gerência de reutilização. A *Brechó 2.0* busca potencializar alguns mecanismos: (i) flexibilização do processo de publicação e tratamento de dependências; (ii) de-finição de interface para integração com outras ferramentas; (iii) geração e publicação de serviços; (iv) suporte a movimentações de mercado (avaliação, *marketing*, tarifação e precificação) com foco no valor de componentes e de serviços. A *Brechó 2.0* tem por objetivo apoiar diretamente a implementação do processo GRU e, indiretamente, o processo De-senvolvimento para Reutilização (DRU) [18], dado que a *Brechó 2.0* dá suporte à gerência dos artefatos construídos com base neste processo, quando a organização atingir um maior nível de maturidade (o GRU está no nível E, e o DRU está no nível C). Ou seja, a *Brechó 2.0* funciona como um catálogo e disponibiliza o uso das *releases* de ativos sob processo Gerência de Configuração (nível F), que requer o estabelecimento e manutenção de um sistema de apoio como CVS, Subversion etc. Sendo assim, somente o ativo contido na *Brechó 2.0* pode ser reutilizado, pois está atrelada a ele uma liberação do repositório de gerência de configuração.

Novos mecanismos estão sendo desenvolvidos, relacionados ao desenvolvimento de um mecanismo de suporte a uma plataforma para apoiar ecossistemas de software [14]. A meta é explorar dados históricos e visualização de informação produzida a partir das ações dos *stakeholders* em um ambiente colaborativo ou um mercado competitivo baseado em valor. Isto é fundamental para se desenvolver modelos confiáveis que descrevam o comportamento de componentes e serviços em uma biblioteca. A Brechó tem sido utilizada pelo Laboratório de Engenharia de Software da COPPE/UFRJ nos últimos três anos para fins acadêmicos e está disponível no link <http://reuse.cos.ufrj.br/brecho>. Por fim, estudos de caso na indústria estão sendo planejados, bem como pesquisas sobre a definição de valor para serviços de software.

Agradecimentos.

Os autores agradecem ao CNPq e à FAPERJ pelo apoio financeiro.

Referências

1. Biffl, S. *et al.* (2006) "Value-Based Software Engineering". Springer-Verlag, 388p.
2. Boehm, B.; Sullivan, K. J. (2000) "Software Economics: A Roadmap", In: The Future of Software Engineering, 22nd ICSE, Limerick, Ireland, 319-344.
3. Brechó (2010) "Projeto Brechó", <http://reuse.cos.ufrj.br/brecho>.
4. Brereton, P. *et al.* (2002) "Software Components – Enabling a Mass Market", In: 10th International Workshop on Software Technology and Engineering Practice, Washington, USA, 169-176.
5. ComponentSource (2010) "ComponentSource", <http://www.componentsource.com>.

6. Fernandes, P. C. C. *et al.* (2007) "Carga Dinâmica de Componentes via Biblioteca Brechó", In: I SBCARS, Sessão de Ferramentas, Campinas, Brasil, 1-8.
7. JIDE (2010) "JIDE Software", <http://www.jidesoft.com/>.
8. Lee, R.; Kim, H.; Yang, H. S. (2004) "An Architecture Model for Dynamically Converting Components into Web Services", In: 11th APSEC, Busan, Korea, 648-654.
9. Logidex (2010) "Logic Library Logidex", <http://www.logiclibrary.com>.
10. Marinho, A.; Murta, L.; Werner, C. (2009) "Extending a Software Component Repository to Provide Services". In: 11th ICSR, Falls Church, VA, USA, 258-268, Sep.
11. Odyssey (2010) "Projeto Odyssey", <http://reuse.cos.ufrj.br/odyssey>.
12. Overhage, S.; Thomas, P. (2004) "A Business Perspective on Component Trading: Criteria, Immaturities, and Critical Success Factors", In: 30th Euromicro Conference, Rennes, France, 108-117.
13. Poulin, J. S. (1995) "Populating Software Repositories: Incentives and Domain-Specific Software". *Journal of Systems and Software* 30, 3 (September), 187-199.
14. Santos, R. P. (2010) "Brechó-VCM: Uma Abordagem Baseada em Valor para Mercados de Componentes". Dissertação (Mestrado), COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.
15. Santos, R.; Werner, C., Silva, M. (2009) "Incorporating Information of Value in a Component Repository to Support a Component Marketplace Infrastructure", In: 10th IEEE IRI, Las Vegas, USA.
16. Santos, R.; Werner, C.; Silva, M. (2010) "Brechó-VCM: A Value-Based Approach for Component Markets". *International Transactions on Systems Science and Applications*. To appear.
17. Select (2009) "Select Asset Manager", <http://www.selectbs.com/adt/>.
18. SOFTEX (2009) "Guia Geral do MPS.BR – Modelo MPS e Modelo de Referência (MR-MPS)". Sociedade SOFTEX, Maio de 2009.
19. Stal, M. (2002) "Web Services: Beyond Component-Based Computing". *Communications of the ACM* 45, 10 (October), 71-76.
20. Szyperski, C.; Gruntz, D.; Murer, S. (2002) "Component Software – Beyond Object-Oriented Programming". Addison-Wesley & ACM Press, 2. ed., 589p.
21. Werner, C. M. L. *et al.* (2009) "Towards a Component and Service Marketplace with Brechó Library". In: IADIS International Conf. WWW/Internet, Rome, Italy, 567-574, Nov.
22. Xtras (2010) "Xtras.Net", <http://xtras.net/>.

Ferramenta Colaborativa para Apoio à Gestão do Conhecimento em Implementações MPS

Luiz Leandro dos Reis Fortaleza¹, Gleison Santos², Tayana Uchôa Conte¹

¹USES – Grupo de Pesquisa em Usabilidade e Engenharia de Software - Departamento de Ciência da Computação – Universidade Federal do Amazonas (UFAM)
Manaus – AM – Brasil

² Programa de Pós-Graduação em Informática - Universidade Federal do Estado do Rio de Janeiro (UNIRIO) - Av. Pasteur, 458 - Urca - Rio de Janeiro - RJ - CEP 22290-240
leandro.fort87@gmail.com, gleison.santos@uniriotec.br, tayana@dcc.ufam.edu.br

Abstract. *Adopting a quality model is important to produce software with quality and adequate to market expectation. This paper presents a collaborative case tool based on wiki to the MR-MPS implementation with knowledge management support. We expect its use to be complementary to the model's official guides when used by members of Implementation Institutions and members of organizations implementing the model. Thus, we expect this tool to contribute to the success of software process improvement initiatives.*

Resumo. *Adotar um modelo de qualidade é importante para produzir software de qualidade e que atenda às expectativas do mercado. Este artigo apresenta uma ferramenta colaborativa baseada em wiki para a implementação do MR-MPS com apoio à gestão de conhecimento. Espera-se que seu uso seja complementar aos guias oficiais do modelo ao ser utilizada por membros de uma Instituição Implementadora e por membros de organizações que estejam implementando o modelo, contribuindo, assim, para o sucesso de iniciativas de melhoria de processos de software.*

1. Introdução

Para as organizações de desenvolvimento de software, alcançar competitividade pela qualidade, implica tanto na melhoria da qualidade dos produtos de software e serviços correlatos, quanto na melhoria dos processos de produção e distribuição [Szimanski *et al.* 2006]. A adoção de um modelo de qualidade é um passo importante para se conseguir produzir software de maneira eficiente e que atenda às necessidades do mercado. O MPS.BR [SOFTEX 2009a] é um programa criado em dezembro de 2003 que tem por objetivo a Melhoria do Processo de Software Brasileiro e é coordenado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX), que conta com o apoio do MCT, da FINEP, do SEBRAE e do BID. Este programa tem por base os conceitos de maturidade e capacidade de processo para avaliação e melhoria da qualidade e produtividade de empresas de software e serviços correlatos.

O Modelo de Negócio do MPS.BR descreve o papel das Instituições Implementadoras (IIs) e Instituições Avaliadoras (IAs). As primeiras são instituições autorizadas a executarem implementações baseadas no Modelo de Referência do MPS.BR (MR-MPS) em organizações de software e as últimas são instituições responsáveis pela avaliação das organizações que adotaram o modelo em seus processos organizacionais. O MPS.BR possui atualmente 18 Instituições Implementadoras, responsáveis pela maioria das implementações nas 228 organizações que obtiveram sucesso na avaliação MPS.BR

atualmente [SOFTEX 2010]. Esses números demonstram que o programa MPS.BR tem tido grande aceitação por parte das organizações de desenvolvimento de software brasileiras.

O MR-MPS possui sete níveis de maturidade, que são representados por letras de G (menor nível) a A (maior nível). Esses níveis são cumulativos, então para se obter uma avaliação positiva em um nível, é preciso atender a todos os resultados esperados dos níveis anteriores também. Visando auxiliar o processo de implementação do modelo, a Softex disponibiliza um Guia de Implementação [SOFTEX 2009b] que é uma série de dez documentos que fornecem orientações para se implementar cada um dos níveis de maturidade do MPS.BR. Tal guia tem se mostrado útil nas implementações realizadas ao longo dos últimos anos e sofreu modificações nas suas diferentes versões buscando adequar-se às necessidades das empresas brasileiras.

No entanto, embora o Guia de Implementação [SOFTEX 2009b] descreva orientações gerais, cada II possui uma estratégia de implantação, o que muitas vezes implica em ter orientações específicas para implementar determinados processos. Nota-se, então, a necessidade de se ter um apoio que permita distribuir o conhecimento relativo às implementações de uma determinada II de maneira que seus diferentes implementadores tenham acesso rápido a estas orientações específicas. Desse modo, a criação de um mecanismo de apoio à gestão do conhecimento pode ser um facilitador para o sucesso de implementações, pois pode agregar orientações específicas, exemplos, modelos e relatos de experiências que podem ser úteis para uma implementação bem sucedida.

Este artigo descreve a criação de uma base de conhecimento sobre o nível G do MR-MPS e boas práticas de implementação deste nível de maturidade. Esta base foi implementada como um wiki, que pode ser visto como uma ferramenta colaborativa de apoio à Gestão do Conhecimento em Implementações MPS. A principal motivação foi a percepção da possível contribuição ao se agregar uma ferramenta colaborativa, de fácil acesso e com conteúdo dinâmico, ao guia de implementação do referido nível [SOFTEX 2009b]. Espera-se que esta abordagem possa resultar em um maior número de implementações bem sucedidas baseadas no MR-MPS.

Além dessa introdução, este artigo se divide nas seguintes seções: a Seção 2 apresenta brevemente diferentes ferramentas para apoio à implementação do MPS.BR. A Seção 3 discute a importância do apoio ferramental à Gestão do Conhecimento, com enfoque na utilização de wikis para compartilhamento do conhecimento. A Seção 4 mostra o wiki proposto para o nível G, com detalhamento das funcionalidades e características. Por fim, a Seção 5 apresenta as considerações finais.

2. Ferramentas para apoio ao MPS.BR

O MPS.BR, desde o seu início, contou com a colaboração de instituições de pesquisa, portanto fomentou trabalhos de pesquisa relacionados a ele, além de se beneficiar de ferramentas computacionais trazidas do meio acadêmico. Neste último quesito, pode-se citar os ambientes de desenvolvimento de software como, por exemplo, a Estação Taba [Santos *et al.* 2008] e o WebAPSEE [França *et al.* 2009].

A Estação Taba [Santos 2008] [Santos *et al.* 2008] é um meta-ambiente de desenvolvimento de software capaz de gerar, através de instanciação, ambientes de desenvolvimento de software (ADS) centrados

em processos e adequados a projetos com diferentes características. O WebAPSEE [Rocha 2009] [França *et al.* 2009] fornece apoio automatizado para a gerência de processos de software e foi projetado para permitir a integração de vários serviços relacionados com uma visão abrangente do meta-processo de software.

Santos (2008) apresenta um estudo baseado em revisão sistemática destacando o apoio ferramental em fatores de apoio a iniciativas de melhoria de processos de software. A Figura 1 mostra a abrangência do apoio ferramental citada em publicações mencionadas pelas publicações identificadas no escopo desta revisão sistemática. Pode-se perceber a presença de ferramentas/repositório para Gerência de Conhecimento além de guias eletrônicos para processos. Ambos os tipos de ferramentas podem ser implementados com o auxílio de wikis.

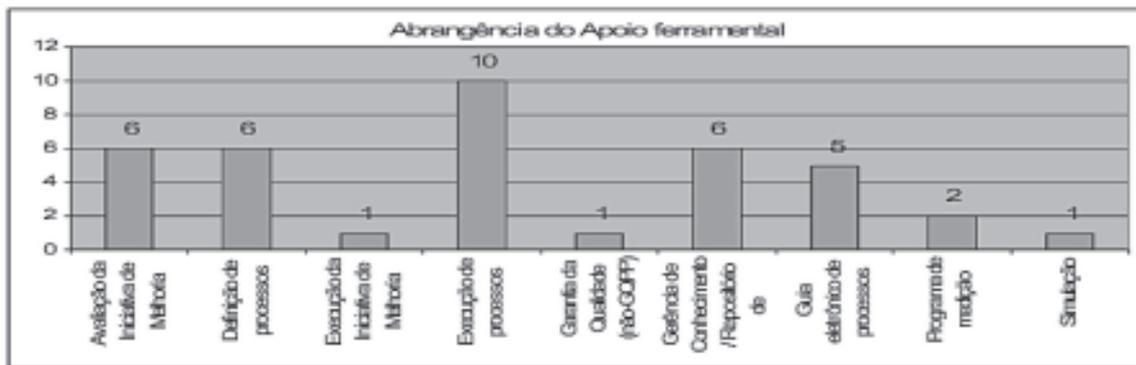


Figura 1. Abrangência do Apoio Ferramental em iniciativas de Melhoria de Processos de Software [Santos 2008]

Também existem ferramentas de apoio ao processo de avaliação, que é o processo no qual as Instituições Avaliadoras verificam se os resultados esperados para um nível do MR-MPS foram alcançados. Como exemplo, é possível citar a FAPS-INT [Tristão 2009] e a MPS-TOOL [Heineck 2008]. Existem também pesquisas e ferramentas de avaliação voltadas para grupos de implementação cooperados como se

Como se observou nos parágrafos anteriores, muitas ferramentas são utilizadas como apoio ao MPS.BR, todavia, ferramentas criadas com o objetivo explícito de auxiliar o processo de implementação são poucas, e mais raras ainda são as que fazem uso de um formato que tem se tornado comum para os usuários: o wiki. A simplicidade de edição, aliada ao caráter colaborativo do wiki, pode constituir um diferencial positivo ao se efetuar uma implementação, pois com tais recursos existe a possibilidade de a implementação poder ser feita de forma mais dinâmica e eficiente.

3. Apoio Colaborativo para Gestão do Conhecimento

A gestão do conhecimento é um processo corporativo, focado na estratégia empresarial e que envolve a gestão das competências, a gestão do capital intelectual, a aprendizagem organizacional, a inteligência empresarial e a educação corporativa [Santos *et al.* 2001].

O papel principal da tecnologia da informação na gestão do conhecimento consiste em ampliar o alcance e acelerar a velocidade de transferência do conhecimento [Carvalho 2000]. Existem diversas ferramentas voltadas à gestão do conhecimento, e de acordo com Carvalho (2000), essas ferramentas

podem ser classificadas nas seguintes categorias: voltadas para intranet, gerenciamento eletrônico de documentos, sistemas de *Groupware*, Sistemas de *Workflow*, sistemas para construção de bases inteligentes de conhecimento, *Business Intelligence*, sistemas de mapas do conhecimento e ferramentas de apoio à inovação.

A Gestão do Conhecimento atua como uma atividade transversal, pois pode ser utilizada para apoiar atividades em todas as fases do desenvolvimento de software, bem como para implantação e avaliação de processos de melhoria da qualidade [Oliveira 2009].

Por definição, um wiki é um conjunto de páginas web *linkadas*, criadas incrementalmente por um grupo de usuários e o software utilizado para gerenciar esse conjunto de páginas [Wagner 2004]. Por seu caráter colaborativo, os wikis contribuem para acelerar a velocidade de transferência e compartilhamento de conhecimento.

Um dos principais problemas da gestão do conhecimento, segundo afirmam Santos *et al.* (2001) é a tendência das pessoas de reter seus conhecimentos. Mesmo as que não o fazem intencionalmente podem simplesmente não estar motivadas a mostrar o que sabem. Um wiki pode servir para estimular o compartilhamento do conhecimento, dentre os membros de uma organização, pois, com a sua utilização, o conhecimento individual é compartilhado com todos, formando um grupo, e a partir daí novamente compartilhado entre os diferentes grupos, desencadeando um ciclo constante [Schnons *et al* 2007].

4. Projeto do Wiki para Apoio a Implementações do Nível G do MR-MPS

O wiki construído tem por objetivo agregar conhecimento sobre a implementação do nível G do MR-MPS. Possuindo detalhes de cada um dos resultados esperados para os processos de Gerência de Projetos (GPR) e Gerência de Requisitos (GRE), além dos resultados dos atributos de processo para os dois processos considerados no Nível G. Este wiki pode ser acessado através do endereço: <http://projfinal.provisorio.ws/wiki>.

Este wiki pode ser visto como uma ferramenta colaborativa, uma vez que alguns de seus usuários participam ativamente da elaboração de conteúdo e este pode ser editado por outros usuários, de modo que as informações contidas nele acabam por ter diversos autores colaborando entre si em prol da construção do conhecimento. Além disto, o wiki descrito neste artigo também é uma ferramenta de apoio à gestão do conhecimento, pois possibilita estruturar de forma encadeada as atividades organizacionais ligadas às implementações do Nível G do MR-MPS. Dessa forma, o conhecimento de uma Instituição Implementadora pode ser externalizado e gerenciado através dele.

O wiki traz um conteúdo genérico sobre a implementação, tendo por base o Guia de Implementação do nível G [SOFTEX 2009b] e conhecimento coletado junto à implementadores. Porém é permitido a cada Instituição Implementadora que faça uso do wiki, editá-lo, se julgar necessário, fazendo com que ele se adeque às suas necessidades. Desse modo, cada Instituição Implementadora pode criar o seu componente wiki como uma instância do wiki genérico, a ser disponibilizado para as Empresas nas quais a II atua, deste modo cada II teria um novo meio de compartilhar conhecimento com as Empresas.

Para tornar a navegação mais dinâmica, fez-se uso de árvore hiperbólica [Lamping et al. 1995], que é uma estrutura gráfica para representação do conhecimento, na forma de hipérbole, cujo centro representa a informação desejada e de onde partem eixos radiais em direção aos nós, de onde, por sua vez, partem novos eixos e assim por diante. Algumas estruturas hierárquicas são muito grandes para serem exibidas por completo na tela do computador. A exibição convencional mapeia toda a hierarquia numa região que é maior que a área disponível para visualização e então usa barras de rolagem para se mover pela região. O problema dessa abordagem é não permitir que o usuário veja o relacionamento da parte visível da árvore com o resto da estrutura sem visões auxiliares. Seria muito útil ver toda a hierarquia enquanto se tem o foco em uma parte particular. Desse modo, as relações entre as partes que compõem o todo podem ser vistas em sua totalidade e o foco pode ser alterado de forma suave e contínua [Lamping et al. 1995] facilitando sua visualização [Santos 2003].

A utilização de uma árvore hiperbólica neste wiki fornece maior interação com usuário. Com isto, buscou-se atender às necessidades de dois tipos distintos de usuários: os que preferem a navegação por meio de textos e aqueles que têm preferência por uma navegação mais gráfica. A árvore hiperbólica construída tem como eixo central o Nível G e como ramificações os processos de Gerência de Projetos e de Gerência de Requisitos, além dos atributos de processos. Tal recurso permite a localização de determinadas informações de forma mais eficiente.

Esta árvore pode ser vista na Figura 2, onde também se observa o mecanismo de busca que destaca o caminho até o nó onde se encontra a informação desejada, neste exemplo, ao se digitar “ciclo de vida” destaca-se o caminho até o GPR3.

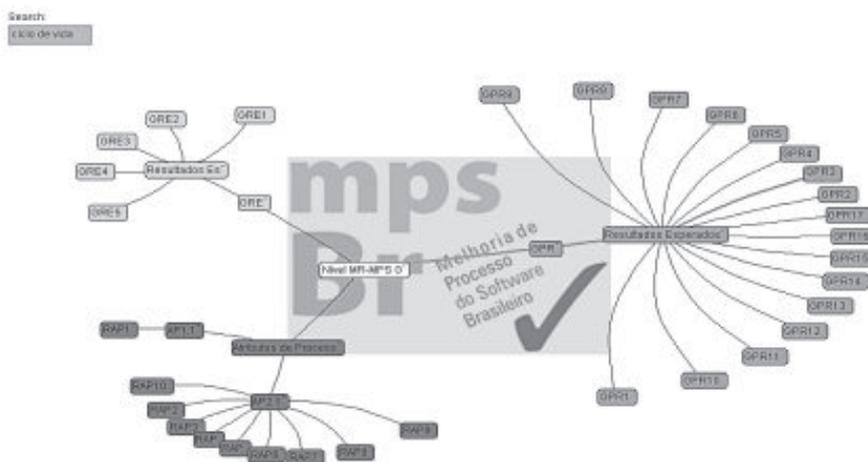


Figura 2. Visualização da árvore hiperbólica referente ao Nível G do MR-MPS

Quando um usuário passa o cursor sobre o código de um resultado esperado ou de um atributo de processo, na árvore hiperbólica, é exibida a descrição de seu conteúdo. Para acessar este conteúdo deve-se clicar duas vezes sobre o código correspondente. O usuário implementador pode editar o conteúdo do wiki, já o colaborador tem acesso apenas a consultas. Na Figura 3 é exibida a página inicial referente ao processo Gerência de Projetos, contendo texto sobre o propósito deste processo, exemplos de atividades envolvidas e resultados esperados, sendo estes últimos em formato de *link* para outras páginas do wiki. Esta página pode ser acessada a partir de um *link* na página inicial do wiki ou por meio do nó correspondente na árvore hiperbólica.



Figura 3. Página wiki referente ao processo Gerência de Projetos

Para cada resultado esperado de um processo ou de um atributo de processo, quando se clica no código correspondente na árvore hiperbólica ou em um *link* no formato texto, é visualizada uma página contendo definição, indicadores diretos e indiretos, links ao longo do texto para outras páginas relacionadas e é possível acrescentar outras informações que o usuário implementador julgue necessárias. A Figura 4 mostra, como exemplo, a página referente ao resultado esperado GPR3, contendo a descrição deste resultado e exemplos de Indicadores Diretos e Indiretos relacionados a ele, bem como algumas informações a mais sobre modelos de ciclo de vida, indo além do que se encontra no guia de implementação.

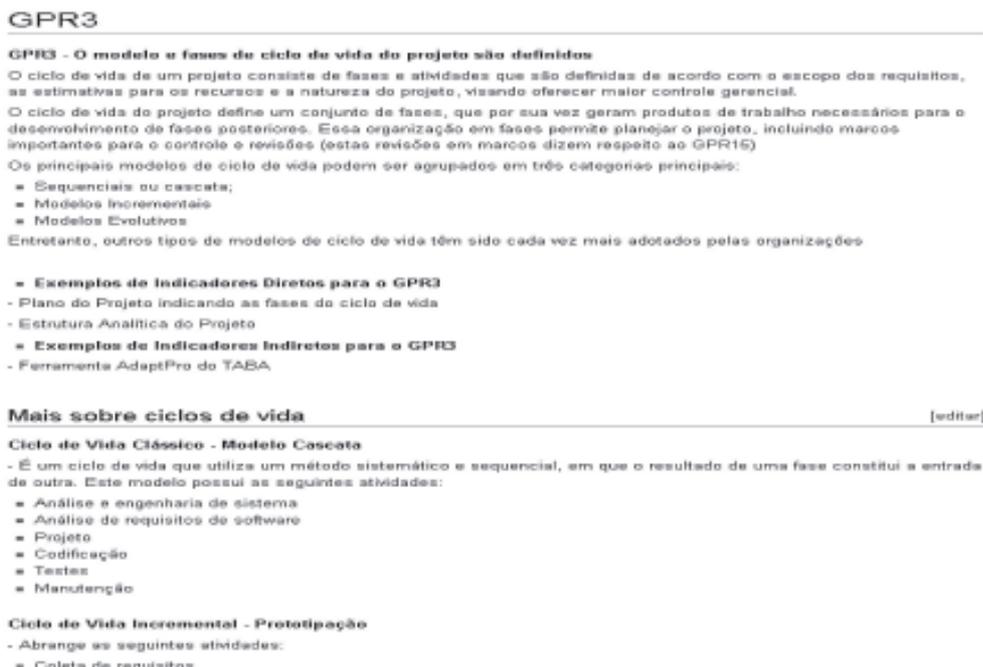


Figura 4. Página do wiki referente ao resultado GPR3

Pensou-se em dois perfis básicos para utilização do wiki: implementador (que também pode ser considerado como os participantes do grupo de processos da organização, responsáveis pela definição e manutenção dos processos em uso) e colaborador da empresa. O implementador pode acrescentar informações ao wiki utilizado pela sua Instituição Implementadora: definições, exemplos de evidências, etc., bem como efetuar consultas. Já o colaborador da Unidade Organizacional na qual o nível G está sendo implementado pode consultar as informações cadastradas pelo implementador.

5. Conclusão

O wiki apresentado neste trabalho tem por objetivo auxiliar o processo de implementação do nível G de MPS.BR, podendo tornar-se uma complementação ao guia oficial. Com seu uso, cada Instituição Implementadora pode ter acesso ao wiki e criar sua própria instância, com suas orientações específicas para seus implementadores. Conforme as informações aumentem, a Instituição Implementadora que o utiliza, assim como as empresas, podem consultar tais informações melhorando suas práticas organizacionais.

Como trabalho futuro pretende-se estender o conteúdo do wiki aos demais níveis do MR-MPS. Espera-se que a utilização deste wiki facilite a implementação e que a ferramenta contribua para a institucionalização do nível implementado, uma vez que o implementador pode utilizá-lo de forma a se adequar às especificidades da Organização e os profissionais podem consultá-lo para entender o modelo e como ele pode ajudá-los. Outra extensão, inclui a expansão do conteúdo do wiki com a criação de outras seções relacionadas a implementações, como por exemplo: implantação do nível G em organizações que fazem uso de métodos ágeis ou que adotem Desenvolvimento Distribuído de Software (DDS). Além disso, vislumbra-se a possibilidade de definir o processo de software das organizações associando-o ao conhecimento sobre implementação do modelo.

O impacto trazido por esta ferramenta, no entanto, só poderá ser mensurado após ser utilizada pelas Instituições Implementadoras e análise da contribuição deste nos resultados das implementações que o utilizaram, pode-se pensar em versões futuras, onde possa haver um perfil de usuário avaliador.

Referências

Carvalho, R.B. (2000). *Aplicações de Softwares de Gestão do Conhecimento: Tipologia e Usos*. Dissertação de Mestrado. UFMG, Belo Horizonte, MG, Brasil.

França, B.B., Sales, E.O., Reis, C.A. L & Reis, R.Q. (2009). Utilização do Ambiente WebAPSEE na implantação do nível G do MPS . BR no CTIC-UFPA. In *Anais do VIII Simpósio Brasileiro de Qualidade de Software - SBQS*. Ouro Preto - MG, pp. 310-317.

Heineck, A. (2008). *Uma Ferramenta para Apoio na Avaliação de Grupos Cooperados de Empresas que Implementam o Modelo MR MPS*. Trabalho de Conclusão de Curso. PUC-RS, Porto Alegre, RS, Brasil.

Lamping, J., Rao, R. & Pirolli, P. (1995). A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *SIGCHI - Conference on Human Factors in Computing Systems*. ACM Press, pp. 401-408.

Oliveira, J. D. (2009). *Abordagem para implantação de Gerência do Conhecimento com apoio de um Ambiente de Desenvolvimento de Software centrado em processos*. Dissertação de Mestrado. Programa de Pós-Graduação em Ciência da Computação da UFPA, Belém, PA, Brasil.

Prikladnicki, R. & Galarraga, O. (2009). MA-MNC: Modelo de Avaliação para Grupos Cooperados de Empresas que Implementam o Modelo MR-MPS. In *VIII Simpósio Brasileiro de Qualidade de Software*. Ouro Preto - MG.

Prikladnicki, R., Furlanetto, T.M. & Becker, C.A. (2007). Uma Análise Quantitativa da Implantação do MR MPS em Projetos Cooperados: Avaliação sob a ótica das Empresas. In *VIII Simpósio Internacional de Melhoria de Processo de Software e de Sistemas*. Santo Amaro - SP.

Rocha, V.C. (2009). *Metodologia para implementação do MPS.BR utilizando o ambiente WebAPSEE*. Dissertação de Mestrado. UFPA, Belém, PA, Brasil.

Santos, A.R., Pacheco F.F, Pereira, H.J & Bastos Júnior, P. (2001). Gestão do Conhecimento como Modelo Empresarial. In *Gestão do Conhecimento: Uma experiência para o sucesso empresarial*. Curitiba: Champagnat.

Santos, G. et al. (2008). Aplicação da Estratégia SPI-KM para Apoiar a Implementação do MPS.BR Níveis G e F em Pequenas e Médias Empresas do Rio de Janeiro. In *VII Simpósio Brasileiro de Qualidade de Software*. Florianópolis - SC, pp. 311-319.

Santos, G. (2008). *Ambientes de Engenharia de Software Orientados a Corporação*. Tese de Doutorado. COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

Santos, G. (2003). *Representação da Distribuição do Conhecimento, Habilidades e Experiências Através da Estrutura Organizacional*. Dissertação de Mestrado. COPPE/UFRJ, Rio de Janeiro, RJ, Brasil.

Schon, C., da Silva, F. & Molossi, S. (2007). "O uso de wikis na gestão do conhecimento em organizações". In: *Biblios: Revista de Bibliotecología y Ciencias de la Información*. Lima, Peru.

SOFTEX (2009a). "*MPS.BR – Guia Geral*", maio 2009. Disponível em www.softex.br

SOFTEX (2009b). "*Guia de Implementação – Parte 1: Fundamentação para Implementação do Nível G do MR-MPS*", maio 2009. Disponível em www.softex.br

SOFTEX (2010). *Mps.Br - Melhoria de Processo de Software Brasileiro*. Disponível em: www.softex.br

Szimanski, F., Rouiller, A.C. & de Souza, A.D. (2006). Aplicando MPS.BR nível G na melhoria do processo de desenvolvimento de software numa pequena empresa. In *VII Congresso de Qualidade na Produção de Software – ProQualiti*. Lavras, MG, Brasil

Tristão, L.R. (2009). *FAPS-INT: Uma ferramenta para apoiar avaliações integradas de processo de software*. Trabalho de Conclusão de Curso. UNIVALI, São José, SC.

Wagner, C. (2004). Wiki: A Technology for Conversational Knowledge Management and Group Collaboration. *Communications of the Association for Information Systems*, 13, 265-289.

Project Builder: uma Ferramenta de Apoio a Implementação do Processo Gerência de Projetos do MPS.BR

Bernardo Grassano¹, Eduardo Carvalho ², Analia Irigoyen Ferreira³, Mariano Montoni³

¹Project Builder

Av. Rio Branco 123, grupo 612, Centro - CEP 20040-005
Rio de Janeiro, RJ, Brasil

²Informal Informática

Rua do Catete 311, Grupo 1311, Catete,
Rio de Janeiro, RJ, Brasil

³ProMove – Business Intelligence Solutions

Rua Marechal Mascarenhas de Moraes, 120, Copacabana, 22030-040
Rio de Janeiro, RJ, Brasil

bernardo.grassano@projectbuilder.com.br, euardo@informal.com.br, {analia,
mmontoni}@promovesolucoes.com

Abstract. *This work presents the experience in implementing MPS.BR level G Project Management 's process area in several organizations, using Project Builder. Besides describing how to use Project Builder to implement project management practices, this paper describes lessons learned, success factors, difficulties and weaknesses observed during this experience. The acquired knowledge during this work helped us to conclude that Project Builder supports largely the Project Management process area - Level G.*

Resumo. *Este trabalho apresenta a experiência na implantação do processo de Gerência de Projetos aderente ao nível G do MPS.BR em diversas organizações, utilizando a ferramenta Project Builder. Além de detalhar como implementar cada prática na ferramenta, este trabalho destaca as lições aprendidas, os fatores de sucesso, as dificuldades e os pontos fracos desta experiência. O conhecimento adquirido ao longo deste processo permitiu concluir que a ferramenta Project Builder apóia largamente a área de processos de Gerência de Projetos do MPS.BR nível G.*

1. Introdução

A Project Builder, Informal e ProMove são empresas parceiras em um projeto comum cujo principal objetivo é facilitar a implementação dos processos de gerência de projetos do nível C do modelo MPS em pequenas e médias empresas de software por meio da adoção de uma ferramenta de gerenciamento de projetos denominada Project Builder (PB). Também é objetivo desse projeto contribuir para o aumento da maturidade de empresas de software do Brasil e, conseqüentemente, a qualidade dos produtos de software, pois além do apoio para implementação dos processos de gerenciamento, conhecimentos estratégicos serão disponibilizados na ferramenta.

A ferramenta Project Builder tem como objetivo principal a Gestão Corporativa de Projetos de forma a atender a grande diversidade de contextos organizacionais. Ainda assim, a maioria dos seus clientes são empresas desenvolvedoras de produtos de software dos quais 38% (5 empresas) foram avaliados no nível G, F ou E do MPS.BR.

Este artigo tem por objetivo apresentar os resultados de uma avaliação da aderência da ferramenta Project Builder ao processo de Gerência de Projetos do modelo MPS (SOFTEx, 2009). Essa avaliação foi conduzida por especialistas com experiência na utilização da ferramenta Project Builder para apoiar diferentes implementações do processo Gerência de Projetos (GPR) em organizações de software. A avaliação da aderência foi realizada por meio de oficinas de projetos conduzidas para discutir e consolidar a forma com melhor relação custo x benefício para implementar os resultados esperados do processo GPR na ferramenta. Nessas oficinas, um membro da Project Builder apresentou os conceitos principais da ferramenta, enquanto um membro da Informal e outro da ProMove levantavam questões sobre as alternativas de implementação de cada um dos resultados esperados do processo GPR. Os resultados das discussões foram documentados em um relatório contendo a descrição do como a ferramenta Project Builder implementa o resultado, bem como os pontos fracos e oportunidades de melhoria.

Este trabalho está estruturado conforme a seguir: na seção 2 é realizada uma breve descrição das funcionalidades existentes no Project Builder; detalhes de como a ferramenta Project Builder (Versão 8.210 do PBCompleto) foi utilizada para gerar os resultados esperados do processo Gerência de Projetos estão descritos na seção 3; e a seção 4 apresenta as considerações finais e perspectivas futuras.

2. As Funcionalidades do Project Builder

Existem três tipos de versões relacionadas ao conjunto de funcionalidades disponíveis no Project Builder, a saber: PB Mínimo, PB Intermediário e PB Completo.

A versão PB Mínimo oferece funcionalidades de apoio ao gerenciamento de portfólio e de projetos, bem como apoio ao gerenciamento de documentos. O apoio ao gerenciamento de projetos envolve as seguintes disciplinas: escopo, tempo, comunicação, recursos humanos e integração. Esta versão permite a criação de Estrutura Analítica do Projeto na forma textual e gráfica, além do cronograma do projeto, definição de calendários específicos, modelos de projetos, configuração de mensagens por projeto/pessoa, notificações programadas, curva S de escopo, alocação diária de recursos, registro de atividades e o envio de relatórios por e-mail. Esta versão do PB oferece também recursos de integração com outras ferramentas de gerência de projetos como o MS-Project¹ e WBS Chart Pro².

Além de todas as funcionalidades do PB-Mínimo, a versão PB Intermediário oferece funcionalidades de apoio à gestão de pessoas (CHA – Conhecimentos, Habilidades e Atitudes), biblioteca de lições aprendidas e gerenciamento de programas. No contexto de projeto, esse tipo de versão fornece também apoio ao gerenciamento de custos de homem/hora e outros custos por projetos e receitas, curva S de custos, alocação de recursos por dia e múltiplos calendários.

Além de todas as funcionalidades do PB-Intermediário, a versão PB Completo oferece funcionalidades de apoio ao planejamento estratégico, propostas de projetos, seleção e priorização de projetos. No contexto de projeto, esse tipo de versão fornece também apoio ao gerenciamento de riscos e qualidade, gerenciamento de questões e fórum de projetos.

¹ www.microsoft.com/project

² www.criticaltools.com/wbsmain.htm

3. A Aderência do Project Builder ao Processo de Gerência de Projetos

A tabela 1 demonstra para cada resultado esperado do processo GPR, o grau de aderência da ferramenta Project Builder ao processo. As possíveis situações definidas foram: Totalmente implementado (☺) – quando o resultado é facilmente apoiado; Largamente implementado (☹) – quando o resultado é apoiado largamente; e Não implementado (⊗) – quando o resultado não é apoiado pela ferramenta.

Resultados Esperado			
GPR 1, 5,6, 7 12 e 14	☺	GPR 2,3,4,8,9, 10,11, 13, 15, 16, 17, 18 e 19	☹

Tabela 1. Avaliação da aderência dos resultados esperados do processo GPR com a utilização da ferramenta Project Builder

Além dos 17 resultados esperados no nível G para o processo GPR (1 a 17), a avaliação conduzida neste trabalho considerou também os resultados esperados GPR 18 e GPR 19 do nível E do modelo MPS.

3.1. GPR 1 - O escopo do trabalho para o projeto é definido

Neste resultado, é importante definir todo o trabalho necessário para entregar um projeto. Sendo assim, uma das ferramentas comumente utilizadas pelos gerentes de projeto para definir o escopo de um projeto é a EAP (Estrutura Analítica do Projeto) que organiza os “pacotes de trabalho” para serem gerenciados. O escopo do trabalho (EAP) pode ser definido na própria ferramenta PB. Além disso, existem campos descritivos na ferramenta para incluir a descrição do escopo, as premissas e restrições do projeto. O resultado GPR 1 foi, então, avaliado como totalmente implementado. A Figura 1 apresenta um exemplo de uma EAP extraída automaticamente da ferramenta PB.

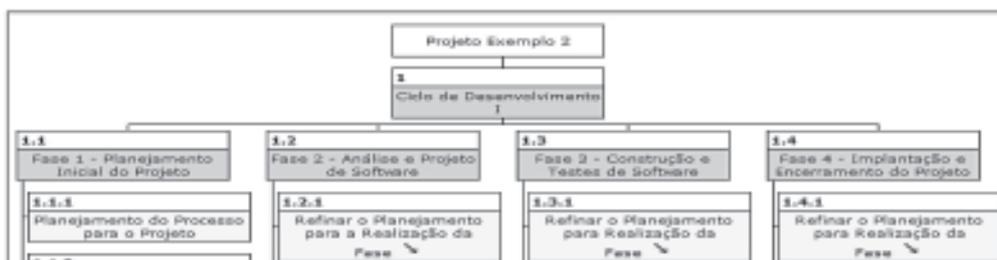


Figura 1. Estrutura Analítica do Projeto (EAP)

3.2. GPR 2 - As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados

Nesta prática, o escopo deve ser dividido em componentes menores, com o objetivo de dimensionar o tamanho do projeto com mais facilidade. No nível G, esta dimensão pode ser feita a partir de características como complexidade, número de requisitos e/ou EAP juntamente com dados históricos (SOFTEX, 2009b). Com o aumento da maturidade, técnicas para dimensionamento, como Análise de Pontos e Função (Vazquez *et al.*, 2010) e /ou metodologias próprias devem ser utilizadas. Neste sentido, a ferramenta PB pode apoiar a consulta ao histórico dos projetos e, também, o armazenamento do

tamanho e da classificação de um determinado componente, utilizando campos específicos. Mesmo que o cálculo da estimativa não seja realizado pela ferramenta, é possível anexar ao projeto, o arquivo contendo os procedimentos do cálculo realizado e garantir que as informações possam ser consultadas a qualquer momento. Portanto, o resultado GPR 2 foi avaliado como largamente implementado.

3.3. GPR 3 - O modelo e as fases do ciclo de vida do projeto são definidos, e GPR 4 - 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

No resultado GPR 3, é importante que a estrutura das fases, atividades e as dependências entre elas sejam definidas. Além disso, a escolha deste ciclo de vida deve ser baseada nas características do projeto (SOFTEX, 2009b). Conforme ilustrado pela Figura 2, a ferramenta PB permite que os modelos e as fases do ciclo de vida sejam definidos no projeto, mas não permite que a justificativa desta escolha seja registrada direto na ferramenta. No entanto, é possível anexar um arquivo no projeto com esta justificativa.

No resultado GPR 4, o esforço e o custo devem ser estimados com base em dados históricos (empresas de nível G, geralmente não possuem) e/ou opinião especializada (SOFTEX, 2009b). O Project Builder permite a consulta do histórico de projetos anteriores com base no esforço. O registro desta ação pode ser feito no próprio campo de descrição do projeto ou por meio de um arquivo anexo. As referências técnicas também podem ser anexadas, por meio de um arquivo, no projeto. Os resultados GPR 3 e GPR 4 foram avaliados como largamente implementados na ferramenta PB.

Componente: 1 -> 2	Início prev.	Dur. prev./real	Fim prev.
Projeto: Projeto Exemplo	06/13/2005	128/1705	12/164/2005
1) Ciclo de Incremento I -	06/12/2005	120/1705	12/164/2005
+ 1.1) Fase 1 - Planejamento Inicial do Projeto	06/12/2005	18/1705	22/12/2005
+ 1.1.1) Planejamento do Processo para o Projeto	06/12/2005	4/4	09/12/2005
+ 1.1.2) Monitoração do Projeto ao Longo da Fase	09/12/2005	5/0	12/12/2005
+ 1.1.3) Avaliação dos Produtos da Fase pelo GQPP	23/12/2005	1/0	23/12/2005
+ 1.1.4) Planejamento Inicial	09/12/2005	12/1702	20/12/2005
+ 1.1.5) Encerramento da Fase	21/12/2005	2/0	22/12/2005
+ 1.1.6) Avaliação de Aderência aos Processos	22/12/2005	1/0	22/12/2005
+ 1.2) Fase 2 - Requisitos e Arquitetura	10/12/2005	25/1692	12/161/2005
+ 1.2.1) Refinar o Planejamento para a Realização da Fase	10/12/2005	2/0	20/12/2005

Figura 2. Modelo e as fases do ciclo de vida definidos

3.5. GPR 5 - O orçamento e o cronograma do projeto, incluindo a definição de marcos e pontos de controle, são estabelecidos e mantidos

Neste resultado, as dependências entre as tarefas devem ser estabelecidas e o caminho crítico identificado. Desta forma, é possível estabelecer o cronograma e o orçamento (é possível registrar outros custos para cada item de EAP, além dos custos de HH e receitas) para o projeto (SOFTEX, 2009b). O Project Builder permite a definição do orçamento, cronograma, marcos e pontos de controle do projeto, respectivamente. Portanto, o resultado GPR 5 foi avaliado como totalmente implementado.

#	Valor	Typo	Descrição
1	100,00	Alimentação	Alimentação durante o planejamento no cliente
2	200,00	Hospedagem	Hospedagem no hotel diária

Figura 5. Planejamento de outros recursos

3.9.GPR 9 - 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, e GPR 10 - Um plano geral para a execução do projeto é estabelecido com a integração de planos específicos

O Project Builder permite a identificação e o planejamento dos dados relevantes ao projeto. Da mesma forma, o Project Builder permite a geração do plano do projeto e todos os seus parâmetros pela própria ferramenta, caso todo o planejamento tenha sido realizado nela. Para que isso seja possível, cada plano considerado deve ser definido e controlado com um item de EAP no projeto. Além disso, tipos de componente e envolvimento devem ser criados para garantir todo o detalhe necessário. Como esta utilização não é nativa da ferramenta, os resultados GPR 9 e GPR 10 foram avaliados como largamente implementados.

3.11.GPR 11 - A viabilidade de atingir as metas do projeto, considerando as restrições e os recursos disponíveis, é avaliada. Se necessário, ajustes são realizados, e GPR 12 - O Plano do Projeto é revisado com todos os interessados e o compromisso com ele é obtido

Conforme ilustrado pela Figura 6, o Project Builder possibilita o registro da análise de viabilidade do projeto por meio da ferramenta de notificação. O Project Builder permite também que os envolvidos revisem o plano do projeto e se comprometam por meio de notificação na própria ferramenta. A utilização de questões e metas deve ser considerada. Como esta utilização não foi definida originalmente para este fim, os resultados GPR 11 e GPR 12 foram avaliados, respectivamente, como largamente e totalmente implementados.

Questão 1 de 1		Início real	Fim real	Situação
IPR	<input type="checkbox"/> Cronograma é viável?	23/08/2018	23/08/2018	Concluído
IPR	<input type="checkbox"/> Orçamento é viável?			
IPR	<input type="checkbox"/> Os peritos alocados são adequados?			
Informações consolidadas				

Figura 6. Registro da análise de viabilidade do projeto

3.13.GPR 13 - O projeto é gerenciado utilizando-se o Plano do Projeto e outros planos que afetam o projeto e os resultados são documentados

O Project Builder (Figura 7) possibilita a gerência dos parâmetros do projeto. Além disso, outros relatórios podem apoiar as diversas monitorações necessárias e o registro desta monitoração pode ser feita por meio de notificações na atividade de monitoração. Somente é possível gerenciar os planos pela ferramenta caso cada plano tenha sido considerado como um item de EAP. O Plano de Gerência de Dados, por exemplo, deve ser incluído com um item da EAP e os artefatos gerados podem ser gerenciados por meio do status do componente: Concluído, em andamento, aceite, aprovado e etc. Portanto, o resultado GPR 13 foi avaliado como largamente implementado.



Figura 7. Monitorações básicas

3.14.GPR 14 - O envolvimento das partes interessadas no projeto é gerenciado, e GPR 15 - Revisões são realizadas em marcos do projeto e conforme estabelecido no planejamento

O Project Builder possibilita o gerenciamento dos envolvidos no projeto por meio de relatórios específicos como relatório de esforço realizado/previsto, relatório de desempenho de projeto, relatório de capacidade de realização das pessoas e mapa de alocação das pessoas. Além disso, a ferramenta possibilita a geração de relatórios específicos para apoiar as revisões nos marcos e pontos de controle.

O registro da realização destas atividades pode ser feito por meio de notificações na própria ferramenta. Não existe um relatório específico para que o gerente monitore o envolvimento das partes interessadas. Além disso, para que a monitoração completa seja possível, o planejamento de comunicação deve ser cadastrado como um item da EAP. Portanto, os resultados GPR 14 e GPR 15 foram avaliados, respectivamente, como largamente e totalmente implementados.

3.16.GPR 16 - Registros de problemas identificados e o resultado da análise de questões pertinentes, incluindo dependências críticas, são estabelecidos e tratados com as partes interessadas, e GPR 17 - Ações para corrigir desvios em rela-

ção ao planejado e para prevenir a repetição dos problemas identificados são estabelecidas, implementadas e acompanhadas até a sua conclusão

O Project Builder possibilita que ações sejam incluídas como componentes ou questões do projeto. O planejamento dessas ações pode ser realizado como qualquer componente da EAP. Também é possível inserir a análise da pertinência na descrição do próprio componente, que pode ter a sua situação cancelada, caso seja necessário. As questões (problemas) e componentes podem ser cadastrados na própria ferramenta. Como esta utilização não foi definida originalmente para este fim, os resultados GPR 16 e GPR 17 foram avaliados como largamente implementados.

3.18.GPR 18 - Um processo definido para o projeto é estabelecido de acordo com a estratégia para adaptação do processo da organização, e GPR 19. Produtos de trabalho, medidas e experiências documentadas contribuem para os ativos de processo organizacional.

O Project Builder possibilita que modelos de componentes sejam cadastrados e importados para o projeto. Na descrição destes modelos podem ser detalhados todos os campos relevantes na definição de um processo como: artefatos de entrada e saída, critérios de entrada e saída, documentos de entrada e saída e as dependências entre as atividades.

Para o GPR 19, o Project Builder possibilita o cadastro de lições aprendidas ao longo da execução do projeto. O projeto também contribui para o histórico de projetos com o esforço e custos planejados e realizados. No entanto, as definições de processo e a justificativa de adaptação do processo padrão por meio do registro de questões são informadas em um único campo da ferramenta. Considerando que esse campo não foi originalmente definido para este fim, os resultados GPR 18 e GPR 19 foram avaliados como largamente implementados.

4. Conclusão e Perspectivas Futuras

Este trabalho apresentou o resultado da avaliação da aderência da ferramenta Project Builder (PB) aos resultados esperados do processo Gerência de Projetos (GPR) do modelo MPS.

Apesar da ferramenta PB não apoiar totalmente todos os resultados esperados do processo GPR, os usuários da ferramenta apontam diversos benefícios como maior alinhamento na organização (os planos estratégicos podem ser desmembrados em projetos e acompanhados dinamicamente pela alta gerência diretamente no Project Builder); maior sucesso nos projetos (a simplicidade do PB facilita o gerenciamento de portfólios, programas e projetos, bem como promove o ganho de maturidade em gerenciamento de projetos na organização); e maior motivação da equipe (o ambiente colaborativo do sistema facilita a integração dos envolvidos e o entendimento das prioridades da organização).

Os esforços na evolução da aderência do Project Builder serão continuados por meio da coleta de lições aprendidas e implementação de melhorias na ferramenta até que todos os processos gerenciais do nível C do MPS.BR (gerência de configuração, gerência de requisitos, etc.) sejam apoiados completamente pela ferramenta.

Referências

Vazquez, C. E., Simões, G. S., Albert, R. M., 2010, *Análise de Pontos de Função: Medição, Estimativas e Gerenciamento de Projetos de Software*, 9ª Edição, Editora Érica.

Softex, 2009, "MPS.BR – Melhoria de Processo do Software Brasileiro, Guia Geral (v. 2009)". In: http://www.softex.br/mpsbr/_guias/default.asp..

Softex, 2009, "Guia de Implementação – Parte 1: Fundamentação para Implementação do Nível G do MR-MPS". In: http://www.softex.br/mpsbr/_guias/default.asp..

Implantação dos Processos Gerência de Projeto e Medição com Auxílio de Ferramenta Baseada em Planilhas

Carlos A. Simões¹, Claudia L. Sodré², Gleison Santos^{3,4}

¹ Synapsis Brasil Ltda. – Av. das Américas nº 3434, Bloco 2, Sala 403, Barra da Tijuca – CEP 22640-102 – Rio de Janeiro – Brasil

² Ampla Energia e Serviço S.A. – Praça Leoni Ramos nº 1, Bloco 1, 2º andar, São Domingos – CEP 24210-205 – Niterói – Brasil

³ Programa de Pós-Graduação em Informática - Universidade Federal do Estado do Rio de Janeiro (UNIRIO) - Av. Pasteur, 458 - Urca - Rio de Janeiro - RJ - CEP 22290-240

⁴ COPPE/UFRJ - Programa de Engenharia de Sistemas e Computação - Av. Horácio Macedo, 2030, CT Bloco H sl 319 - Cx Postal 68511 - Rio de Janeiro - CEP 21941-914

cs@synapsisbrasil.com.br, clasmar@endesabr.com.br, gleison.santos@uniriotec.br

Abstract. *This paper describes how we developed a case tool to support the implementation of the processes Measurement and Project Management in accordance with the MR-MPS reference model. The implementation was done in Synapsis Brazil with the aim of supporting the development of information systems projects and the software process improvement initiative compliant with MR-MPS maturity level C and CMMI maturity level 3. Excel spreadsheets were used due to the flexibility, integration capabilities and ease of learning, implementation and maintenance.*

Resumo. *Este artigo descreve como foi desenvolvida uma ferramenta de apoio à implementação dos processos Medição e Gerência de Projeto em conformidade com o MR-MPS. A implementação foi realizada na Synapsis Brasil com o objetivo de apoiar a melhoria de processos de software e, em especial, o desenvolvimento dos projetos de sistemas de informação aderentes ao nível de maturidade C do modelo MPS e também ao nível 3 do CMMI. A adoção de planilhas Excel deveu-se à flexibilidade, capacidade de integração e facilidade de aprendizado, implementação e manutenção.*

1. Introdução

A melhoria de processo de software assume que uma organização bem gerenciada com indicadores e processos definidos tem mais possibilidades de produzir produtos que atendem mais adequadamente aos requisitos dos clientes, no prazo e no orçamento do que organizações mal gerenciadas e sem processos [SOLINGEN E BERGHOUT 1999]. Controlar qualidade, produtividade e prazo sem a adoção de processo de software é uma tarefa muito difícil.

A Synapsis Brasil adotou como um de seus objetivos estratégicos o investimento contínuo na melhoria da qualidade de seus produtos e serviços e, recentemente, implementou simultaneamente o nível de maturidade C do MR-MPS [SOFTEX 2009] e o nível de maturidade 3 do CMMI [CMMI 2006], sendo avaliada com sucesso em junho e julho de 2009, respectivamente, nestes dois modelos de referência para as duas unidades (RJ e CE). Seu plano estratégico tem como particularidade a implantação ao mesmo tempo de processos comuns e duas unidades com separadas geograficamente. Isto torna necessária uma grande integração das duas equipes, gerenciamento centralizado e eficaz de modo a

fazer com que as unidades caminhem em sintonia. A motivação, o comprometimento e o conhecimento precisam ser disseminados de forma igual nas duas unidades.

Este artigo tem como objetivo apresentar a abordagem adotada pela Synapsis na implementação das áreas de processo Medição e Gerência de Projeto em conformidade com o MR-MPS, utilizando um conjunto de planilhas Excel desenvolvida para apoiar o desenvolvimento dos projetos de sistemas de informação aderentes ao nível de maturidade C do modelo MPS e também ao nível de maturidade 3 do CMMI. O artigo é organizado em três seções, além desta introdução. A Seção 2 comenta sobre a construção de ferramentas enquanto que a Seção 3 descreve como foi realizada a implementação de resultados esperados de Gerência de Projetos (GPR) e Medição (MED) em planilhas. Por último, a seção 4 apresenta as considerações finais.

2. Escolha de Ferramentas para apoiar a Iniciativa de Melhoria de Processos

Na implantação de melhorias de processo fundamentada em modelos como o MPS, a utilização de ferramenta que apóie a implementação de projetos facilita significativamente o trabalho da equipe de projeto apoiando a elaboração dos artefatos necessários e garantindo a integridade de informações entre estes. Avaliando os resultados esperados do modelo de referência MPS.BR e os artefatos gerados para atender as necessidades dos projetos, percebe-se que existe um relacionamento de dependência considerável entre as áreas de processo e também entre artefatos gerados na execução dos resultados esperados. A adoção de uma ferramenta que integre os resultados esperados tem como vantagens evitar duplicidade de informação, garantir a coerência entre os artefatos, minimizar o esforço de treinamento e facilitar a institucionalização dos processos.

A procura por ferramentas integradas para apoiar a implementação de melhoria de processo de software depara com alguns problemas, tais como: (i) elevado custo de aquisição e de manutenção de licenças; (ii) reduzido número de opções; (iii) alto custo para customização de ferramentas na adequação às necessidades da organização; (iv) elevado tempo de customização; (v) dependência de fornecedor externo; (vi) dificuldade na implementação de melhorias; (vii) alto custo de treinamento; (viii) dificuldade de integração com outras ferramentas utilizadas; (ix) elevado custo e prazo para desenvolver internamente uma ferramenta em linguagem tradicional com, por exemplo, Java; (x) dificuldade em manter a equipe responsável pela ferramenta, entre outras.

Dessa forma, a Synapsis optou por desenvolver um conjunto de planilhas adotando o Excel devido à flexibilidade, capacidade de integração, facilidade de aprendizado, velocidade de implementação e manutenção. O uso de planilhas Excel também havia se mostrado eficiente durante a implantação dos processos Gerência de Reutilização [SANTOS et al., 2009], quando planilhas foram utilizadas, em conjunto com um repositório de arquivos controlado pelo Subversion, como uma Biblioteca de Ativos Reutilizáveis.

Além disso, aproveitou-se o fato de que na empresa já se adotava uma planilha customizada e parametrizada para cálculo de estimativa de esforço e custo. Em adição, levou-se em consideração a necessidade de uso intensivo de parametrização para validação, integridade e apoio ao uso das planilhas, criando uma planilha específica para tal. Dessa forma, o desenvolvimento objetivou apoiar e agilizar a

execução das atividades de planejamento e gerenciamento de projeto de seu processo de desenvolvimento. Um dos problemas identificados na adoção do conjunto de planilhas foi não ser um sistema tradicional e sim planilhas, que deveriam possibilitar validar informações inseridas, evitar perda de fórmulas, possuir parâmetros de validação e facilitassem o uso, além de permitir a edição e inclusão de informação somente em locais preestabelecidos. Tais características são padrões em sistemas de informação e deveriam de alguma forma ser implementadas nas planilhas. Além disto, um mecanismo de versionamento deveria ser utilizado de modo a garantir a integridade dos dados e reduzir o risco de perdas de informação.

3. Implantação dos resultados esperados GPR e MED em planilhas

A seguir é apresentado o conjunto de planilhas elaboradas:

- **Escopo:** Tem como objetivo apoiar a descrição da análise de necessidades de negócio do cliente (Figura 1) em conformidade com os resultados esperados GPR1 e GRE1. O escopo é o ponto de partida para o planejamento do projeto, devendo estabelecer o que está e o que não está incluído no projeto, contendo a definição do objetivo e da motivação, os limites e restrições, todos os produtos que serão entregues e gerados pelo projeto, a viabilidade de atendimento, entre outras informações.

5	Nome projeto: Desenvolvimento de Ferramenta de Estimativas		6	Sigla: Teste		7	Data elaboração: 10/08/10	
7	Líder de Projeto: Carlos Simões			8	Anexo:			
9	Cliente: Synapsis Brasil			10	Anexo:			
11	1. Controle de versão do documento de escopo:							
12	Revisão	Item	Data	Autor	Comentário			
13	V01	Definição das necessidades do cliente	10/08/10	Carlos Simões	Definição da estrutura das planilhas.			
14	V02							
15								
16	2. Objetivo do projeto							
17	Implementar um conjunto de planilhas para apoiar o planejamento de projeto, em particular as estimativas necessárias.							
19	3. Análise das necessidades de negócio do cliente							
20	Desenvolver um conjunto de planilhas com a finalidade de apoiar o planejamento do projeto e as estimativas de tamanho, esforço, prazo e custo.							
21	Atender a algumas das práticas de GPR e de MED do modelo de referência NPS.BR.							
22	Disponibilizar funções para: Definir escopo; Definir processo; Elaborar estimativa de tamanho usando análise de pontos de função; Consultar estimativas das funções; Calcular esforço e custo em função do perfil e do nível de experiência deste; Consultar dados para elaborar cronograma e monitorar desempenho do projeto; Gerar planos para Garantia da Qualidade; Verificação, Validação, Gerência de Comunicação, Gerência de Documento e Gerência de Configuração. Contendo também a relação dos artefatos a serem gerados pela execução de cada atividade do processo; Apresentar resumo das estimativas; Apresentar resumo com visão cliente; Possibilitar gerar estimativa de prazo em função do tamanho da equipe e da quantidade de incrementos do projeto; Apresentar o GQT; Apresentar o plano de Medição adaptado para o projeto; Disponibiliza informação sobre o perfil responsável pela análise e pela coleta das medições para cada medida e momento (componentes de processo) em que estas são realizadas; apresentar função de ajuda.							
24	4. Premissas							
25	Ser aderente aos modelos CPRI e NPS.BR.							
26	Ser integrada e fácil de usar.							
27	Usar Análise de Pontos de Função para apoiar a estimativa de tamanho							
29	5. Restrições							
30	Ter a primeira versão pronta para uso em prazo compatível como o cronograma de implantação do projeto de melhoria para avaliação NPS.BR.							
32	6. Limites do projeto							
33	Atender às práticas GPR1, GPR2, GPR3, GPR4, GPR5, MED1, MED2, MED3 e MED4.							
35	7. Análise de impacto nos processos de negócio							
36	Não há.							
39	8. Outros itens necessários ao projeto							
40	Treinamento	Planejar treinamento para os usuários da ferramenta.						
41	9. Viabilidade de atendimento da necessidade:							
42	Não se aplica							
44	9.1 Análise da área de soluções:							
45	Não se aplica							
47	9.2 Análise da área de Operações:							
48	Não se aplica							
50	9.3 Análise da área de Telecomunicações:							
51	Não se aplica							
53	9.4 Análise da área de Arquitetura:							
54	Não se aplica							
56	9.5 Análise da área de testes:							
57	Não se aplica							

Figura 1. Escopo do projeto

- Processo:** Possibilita caracterizar o processo de desenvolvimento adequado à necessidade de negócio do cliente (Figura 2), incluindo itens relacionados ao ambiente desejado, o tipo de processo desejado, restrições e premissas pré-definidas, informações sobre tipos de testes de software, aquisição, entre outras. Dá-se início à adaptação do processo padrão para o projeto (GPR18 parcial). Possibilita selecionar o ciclo de vida (GPR3) em função das características da necessidade de negócio. Pode-se também indicar se parte do desenvolvimento do software será realizado por um fornecedor externo, integrando assim, o processo de desenvolvimento de software com o processo de aquisição [SIMÕES *et al.*, 2009] [SIMÕES *et al.*, 2010].

4	Nome do Projeto / Requisito		Responsável estimativa		Data da estimativa
5	Desenvolvimento de Ferramenta de Estimativas		Carlos Simões		Resp. válido
6	Sugestão: Modelo referência MPS BR nível C - CHMI ML3		Ciclo de vida sugerido		Ciclo de vida adotado
7	Processo projeto de sistema de informação adaptado		Multincremental		Multincremental
8	Processo instanciado adaptado adotado		Mensagens:		
9	Processo projeto de sistema de informação adaptado		Caracterização de projeto com 6 indicações de (S)		
11	Processo projeto de sistema de informação adaptado	S	N	Processo fábrica de especificação de requisitos	
12	Processo manutenção de sistema de informação	N	N	Processo fábrica de código	
13	Processo extração / modificação de dados	N	N	Processo fábrica de testes	
15	Grupo de manutenções em sistema de informação	N	N	Adição de funcionalidade ao projeto	
16	Estimativa já aprovada pelo cliente	N	S	Apresentar valores monetários	
17	Monitorar requerimentos de um projeto como banco de horas	N	N	Venda para empresas que não pertençam ao grupo	
20	Conteúdo desenvolvimento para DW	N	N	Conteúdo desenvolvimento para SAP	
22	Usado em mais de uma instalação / unidade	N	N	Necessita fazer testes de performance	
23	Testes de regressão com outras funções já existentes	N	N	Mecanismos / ferramentas que facilitam os testes	
25	Aquisição de especificação de requisitos	N	S	Desenvolver considerando múltiplos incrementos	
26	Aquisição de especificação técnica	N	N	Especificação de requisitos fornecidos pelo cliente	
27	Aquisição de codificação	N	N	Projeto técnico (especificações) fornecido pelo cliente	
28	Aquisição de testes	N	N	Plano de testes (casos e itens) fornecido pelo cliente	
30	Fácil acesso ao fornecedor de requisitos	N	S	Entregar mais de um incremento no projeto	
31	Necessita fazer refinamento do protótipo	S	N	Restrição de custo	
32	Ambiente Web	S	N	Restrição de prazo	
33	Ambiente cliente servidor	N	N	Restrição de disponibilidade de recursos humanos	
34	Orientação a objeto (UML)	S	N	Negócio fora do domínio da organização	
35	Análise estruturada / Essencial	N	N	Tecnologia fora do domínio da organização	
36	For parte de um sistema maior	N	N	Dificuldade em levantar os requisitos	
37	Justificativa adoção processo	Justificativa adoção ciclo de vida			

Figura 2. Caracterização do processo para o projeto

- Estimativa:** Possibilita registrar a análise funcional (transacional) e a análise de dados em conformidade à métrica Análise de Ponto de Função, com quantidades, classificação, peso e linguagem de programação de cada função (Figuras 3). O produto resultante da execução de GPR1 e GRE1 fornece as referências necessárias à execução de GPR2 no qual é descrito que "As tarefas e os produtos de trabalho do projeto são dimensionados utilizando métodos apropriados". O escopo do projeto deve agora ser decomposto em componentes menores (produtos e tarefas), mais facilmente gerenciáveis e possíveis de serem dimensionados. Isto ajuda a garantir que os relacionamentos entre estes componentes e os dados de medição sejam mantidos e possam ser coletados a um nível adequado de granularidade. As medições devem ter uma granularidade tal que permitam avaliar o comportamento e o desempenho de uma única atividade [KITCHENHAM *et al.* 2006]. Esta planilha possibilita definir percentual de reaproveitamento funcional e de codificação para cada funcionalidade do projeto.

requerimento: Desenvolvimento de Ferramenta de Estimativas	Qtd de Funções	Classificação	Qtd Elementos de Dados	Qtd Tipos Registro	Peso PF	Método utilizado: Análise de Pontos de Função		Nível de Reaproveitamento
						QTD PF	QTD PF	
Análise de pontos de função						200		
Análise Funcional: Nome / Descrição	10					100		Identificação
Elaborar conceito de planilha:								
Definir escopo	1	AL				100		100%
Definir processo	1	AL				100		100%
Calcular quantidade de funções com base no valor do produto de esforço	1	AL				100		100%
requerimento: Desenvolvimento de Ferramenta de Estimativas								
1 Análise de Dados: Nome / Descrição								
2 Escopo	1	AL				7	7	N
3 Processo padrão	1	AL				7	7	N
4 Análise funcional e análise de dados	1	AL				33	10	N
							49	

Figura 3. Estimativa de Pontos de Função

- **Função:** Apresenta as funções referentes à análise transaccional, o esforço e custo correspondente à execução de algumas das principais atividades do processo adaptado (Figura 4). Consolida informações calculadas nas planilhas: Estimativa, Esforço e Processo. Tem como finalidade apoiar a elaboração do cronograma onde alguns dos componentes de processo são decomposto por função, como por exemplo, especificação de requisitos e construção. Apóia a decisão de selecionar, priorizar e decidir pela implementação ou não de função a partir da avaliação de esforço e custo.

projeto: Desenvolvimento de Ferramenta de Estimativas		Pontos de Função				REquisitos		Código e programação		Testes		Totais		Custo	
Requisição	Processo	Função	Complexidade	Experiência	Complexidade	Experiência	Complexidade	Experiência	Complexidade	Experiência	Total	Por Função	Por Função	Por Função	Por Função
0001	0001	0001	1	1	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0	1,0
0002	0002	0002	1	1	2,0	2,0	2,0	2,0	2,0	2,0	2,0	2,0	2,0	2,0	2,0
0003	0003	0003	1	1	3,0	3,0	3,0	3,0	3,0	3,0	3,0	3,0	3,0	3,0	3,0
0004	0004	0004	1	1	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0	4,0
0005	0005	0005	1	1	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0	5,0
0006	0006	0006	1	1	6,0	6,0	6,0	6,0	6,0	6,0	6,0	6,0	6,0	6,0	6,0

Figura 4. Funções referentes à análise transaccional

- **Esforço:** Apresenta o conjunto de componentes do processo padrão da organização, possibilitando adaptar para o processo escolhido a partir de exclusão de componente de processo que pelas características do projeto não serão executadas (Figura 5), complementando a adaptação do processo padrão para o projeto (GPR18). A partir da base histórica de produtividade por componente de processo e do tamanho em termos de quantidade de pontos de função, calcula o esforço de desenvolvimento em homem hora e o custo para cada um dos componentes do processo de desenvolvimento de software adaptado ao projeto (GPR4 e GPR5 parcial), relacionando esforço, taxa hora associada ao perfil executor do componente de processo, nível de experiência do perfil, linguagens de programação utilizadas para cada funcionalidade e margem de risco para o projeto.

Componentes do Processo		Execução		Perfil		Produzida		Estimada		771		Custo R\$	
Requisição	Processo	Função	Complexidade	Experiência	Complexidade	Experiência	Complexidade	Experiência	Complexidade	Experiência	Total	Por Função	Por Função
projeto: Desenvolvimento de Ferramenta de Estimativas													
Fase 1 Planejamento do projeto													
Planejar processo para o projeto													
				LP	MD	17,5	17,5	60,0	60,0	60,0	92,2	70,00	6.891,25
				QA	MD	75,0	75	16,1	16,1	16,1	16,1	70,00	1.237,47
				QA	MD	190,2	190,2	5,4	5,4	5,4	5,4	70,00	444,58
				LP	MD	212,9	212,9	5,7	5,7	5,7	5,7	70,00	397,18
Planejar projeto													
				LP	MD	71,3	71,3	16,9	16,9	16,9	35,3	70,00	2.510,90
				LP	MD	125,8	125,8	9,3	9,3	9,3	9,3	70,00	652,47
				QA	MD	225,0	225,0	5,4	5,4	5,4	5,4	70,00	375,82
				QA	MD	725,0	725,0	1,7	1,7	1,7	1,7	70,00	116,63
Auditar configuração - Fase 1													
				LP	MD	1450,0	1450,0	0,8	0,8	0,8	4,3	70,00	289,30
				CH	MD	680,0	680,0	1,8	1,8	1,8	1,8	70,00	124,35

Figura 5. Estimativa de esforço e custo do projeto

- **Cronograma:** Disponibiliza dados detalhados de esforço e prazo para cada componente do processo adaptado (Figura 6) com o objetivo de alimentar o cronograma padrão elaborado no MS-Project (GPR7 parcial e GPR5). Este cronograma padrão possui o mesmo conjunto de atividades do processo padrão. Tornando o processo de alimentação do cronograma no MS-Project a partir da planilha uma simples operação de "copiar" dados estimados de esforço / prazo e colar no MS-Project. As horas realizadas são apontadas no MS-Project Server (PWA) e copiadas para a planilha.

Objetivo Negocio	Objetivo	Questões	Medida
Incrementar a nível atual de satisfação dos usuários	Monitorar o processo de Garantia de Qualidade do Processo e Produto	Qual a densidade de defeitos encontrados nas auditorias de PPOA?	DQLQA
Incrementar a nível atual de satisfação dos usuários	Monitorar o processo de Garantia de Qualidade do Processo e Produto	Qual a densidade de defeitos encontrados nas checklists de aderência de PPOA?	DCCQA
Assegurar um número de horas em capacitações técnicas e motivacionais do pessoal	Monitorar o processo de Gestão de Recursos Humanos	Qual a quantidade de horas de capacitação Técnica e Motivacional?	QRTSC
Cumprimento de projetos em prazo e reatibilidade: ponderación de los proyectos en base al ingreso contractual, se dará inicio a la medición del cumplimiento con los proyectos informados en esta (al fin de junio) y se debe tener el 100% evaluado al cierre de diciembre.	Monitorar o processo de Monitoração e Controle	Qual a Densidade de defeitos encontrados nas relatórios de monitoramento?	DCCQA
Cumprimento de projetos em prazo e reatibilidade: ponderación de los proyectos en base al ingreso contractual, se dará inicio a la medición del cumplimiento con los proyectos informados en esta (al fin de junio) y se debe tener el 100% evaluado al cierre de diciembre.	Monitorar o processo de Planejamento do Projeto	Qual o Índice de Desempenho de Prazo do Projeto?	IP

Figura 8. Plano de Medição: Objetivos, questões e medidas

- PM:** Disponibiliza o plano de medição adotado para o projeto conforme o processo adaptado. Apresenta o conjunto de medidas identificadas (MED2), a definição de procedimentos de coleta e armazenamento de medidas (MED3) e os procedimentos para análise das medidas (MED4) em conformidade com MED1 e o processo adaptado. As informações disponibilizadas referentes à definição operacional das medidas são: nome da medida; descrição; mnemônico; valor base; limite inferior; limite superior; equação de cálculo; unidade; procedimento de análise; procedimento de coleta; entidade; frequência de coleta; defeito (S/N); ativa (S/N); obrigatória (S/N); automática (S/N); atômica (S/N).
- RH:** Apresenta a estimativa de prazo e equipe de desenvolvimento, com percentuais de alocação por perfil e etapa do projeto de software (Figura 9). Possibilita fazer simulação de tamanho de equipe e quantidade de incrementos para estimativa de prazo. Disponibiliza informação sobre distribuição de esforço entre casos de uso ou incrementos.

Requisitos: Desenvolvimento de Ferramenta de Estimativas	Escalar	RH	Qtd de RH alocados	% de utilização de RH	Qtd de RH alocados referente ao No. de alocação	Fase 1 Planejamento do requerimento	Fase 2 Especificação de requisitos e construção e teste	Fase 3	Fase 4 Encerramento do requerimento	Total
Líder de Projeto	LP		1	100%						0,0
Consultor	CS	126,8	2	100%	2,8	14,8	30,2	14,8	0,8	85,3
Analista de Qualidade	QA		1	100%						0,0
Gestora de Configuração	CG		1	100%						0,0
Líder de Processo	LP	43,7	1	100%	1,8	16,7		38,2	2,8	43,7
Analista Programador	AP	104,8	2	100%	2,8			82,4		82,4
Líder de Teste	LT	5,8	1	100%	1,8			5,4		5,4
Analista de testes	AT	118,4	1	100%	1,8			118,4		118,4
Direção e Gerência	DG		1	100%						0,0
Cliente	CL		1	100%						0,0
Documentador	DOC		1	100%						0,0
Analista de Banco de Dados	DBA		1	100%						0,0
		218,8				26,3	30,2	261,1	3,6	313,3
Quantidade envolvidos:		7	Total RH considerando qtd RH		26,3	Quantidade de incrementos:		1	2	
Fator de influência da equipe no prazo:		DL	Total RH considerando qtd RH e qtd incrementos		26,3	30,2	126,8	3,6	168	
Considerar fator influência caminho crítico?		s	Qtd dias com correção e sem fator de influência		4,4	5,8	22,8	0,8	33	
Considerar fator de influência equipe?		s	Qtd dias do fator influência tamanho equipe		2,2	3,7	1,1	0,0	2	
			Qtd dias com fator de influência		4,2	5,4	28,2	0,1	35	
Fase 1 Planejamento do requerimento (dias)		Fase 2 Especificação de requisitos e Fase 3 Descrição, construção e teste (dias)		Fase 4 Encerramento e requerimento (dias)						
4,9		8,4		24,2		8,7				
36,1 dias - 1,8 meses										

Figura 9. RH, prazo, equipe e incremento

- Tabelas:** Apresenta as tabelas padrão de contagem utilizada para apoiar o uso da métrica análise de pontos de função.
- Resumo:** Disponibiliza o resumo executivo do projeto de software.
- CP&M:** Contém as associações entre os componentes de processo e as medidas. Para aplicação do

controle estatístico de processo, os limites de controle (valor base, limite inferior e limite superior) estão associados aos relacionamentos entre componente de processo e medida e não somente às medidas.

- **C&A:** Disponibiliza informação sobre o perfil responsável pelas coletas e pelas análises das medições para cada medida selecionada em MED2 e os momentos (componentes de processo) que estas devem ser realizadas.
- **Cliente:** Disponibiliza a visão do cliente com as principais informações calculadas para acompanhamento do projeto.
- **Ajuda:** Disponibiliza informações de ajuda ao uso do conjunto de planilhas, inclusive alguns conceitos necessários à aplicação do método de estimativa.
- **Validar:** Disponibiliza informações para parametrização e validação das planilhas.
- **Produto:** Contém o processo padrão da organização, algumas alternativas de adaptação de processo previamente disponibilizadas, as produtividades históricas para cada atividade do processo padrão e dos processos previamente adaptados e o perfil responsável pela execução de cada atividade. Esta planilha possibilita adaptação para uso em outras organizações a partir de adaptações do processo padrão.

A Figura 10 apresenta a integração do conjunto de planilhas com o repositório de medidas da Synapsis, feita a partir de macros do Excel que gravam os dados diretamente nas tabelas do repositório de medidas. Esta integração é fundamental para o armazenamento de dados estimados de esforço, custo, prazo, produtividade, características do projeto e do processo, entre outros. Os laudos de avaliação de produtos (GQA e VER) também foram desenvolvidos em Excel, possibilitando também a exportação e armazenamento das informações de defeitos diretamente no repositório de medidas.

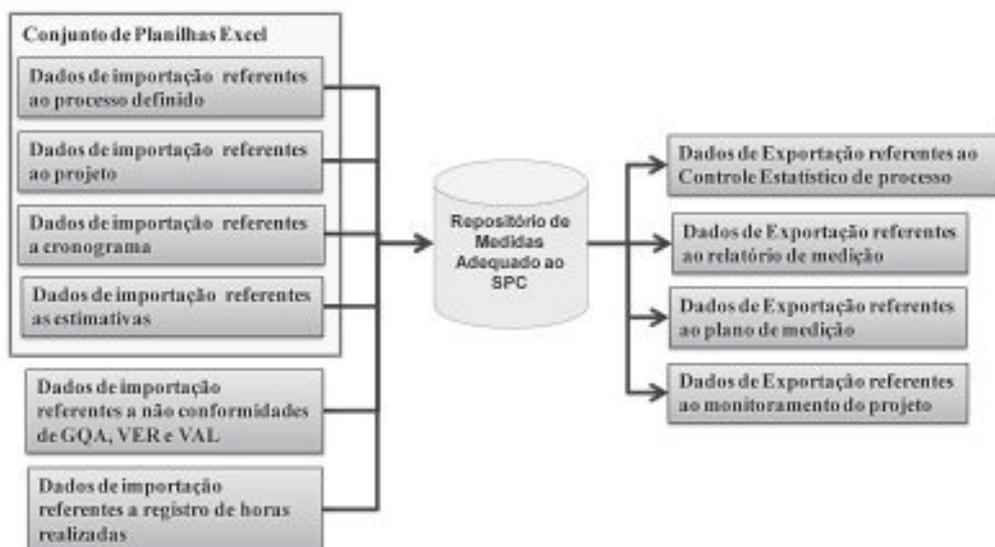


Figura 10. Modelo de exportação e importação de dados

4. Considerações Finais

A utilização da planilha auxiliou a Synapsis a alcançar o nível C de maturidade do modelo MPS.BR e o nível 3 de maturidade do modelo CMMI. Visando alcançar níveis de maturidade mais elevados, como o nível B do MPS.BR, atualmente a Synapsis trabalha na iniciativa de complementar a implementação dos extratores de dados para laudos, relatórios e aplicação em controle estatístico de processo de software, inclusive adequação do repositório de medida para ser capaz de armazenar informações sobre *baselines* relacionadas aos desempenhos dos componentes de processo. A Synapsis planeja alcançar o nível A do MR-MPS e Nível 5 do CMMI em Junho de 2012.

Referências

CMMI-DEV, V1.2 - CMMI® for Development, Version 1.2 CMU/SEI-2006-TR-008 ESC-TR-2006-008 Improving processes for better products CMMI Product Team August 2006

KITCHENHAM, B., KUTAY, C., JEFFERY, R., CONNAUGHTON, C., 2006, "Lessons learnt from the analysis of large-scale corporate databases," 2006 Proceedings - International Conference on Software Engineering 2006, pp. 439-444

SANTOS, G., ZANETTI, D., MACIEL, M., SIMÕES, C., WERNER, C., ROCHA, A.R., 2009, "A Experiência de Implantação dos Processos Gerência de Reutilização e Desenvolvimento para Reutilização na Synapsis-Brasil". In: V Workshop Anual do MPS - WAMPS 2009 (WA-MPS.BR), Campinas, SP.

SIMÕES, C., SANTOS, G., ROCHA, A.R., 2009, "Implantação do Processo Aquisição na Synapsis Brasil". In: V Workshop Anual do MPS - WAMPS 2009, Campinas, SP.

SIMÕES, C., SANTOS, G., ROCHA, A.R., 2010, "Implantação do Processo Aquisição no Contexto do Processo de Desenvolvimento de Software na Synapsis Brasil". In: IX Simpósio Brasileiro de Qualidade de Software – SBQS 2010, Belém, PA.

SOFTEX, 2009 "MPS.BR – Melhoria de Processo do Software Brasileiro – Guia Geral". <http://www.softex.br/mpsbr>. Acesso em: abril/2010

SOFTEX, 2009a, MPS.BR – Melhoria de Processo do Software Brasileiro – Guia de Implementação – Parte 2. <http://www.softex.br/mpsbr>. Acessado em: abril/2010

SOFTEX, 2009b, MPS.BR – Melhoria de Processo do Software Brasileiro – Guia de Implementação – Parte 6. <http://www.softex.br/mpsbr>. Acessado em: abril/2010

SOLINGEN, R., BERGHOUT, E., 1999, The Goal/Question/Metric Method: A Practical Guide for Quality Improvement of Software Development, McGrawHill

Uma Implementação do Processo de Gerência de Projetos Usando Ferramentas de Software Livre

Ewelton Yoshio C. Yoshidome¹, Maurício Ronny de A. Souza¹, Wallace Michel P. Lira¹, Sandro Ronaldo Bezerra Oliveira¹

¹Programa de Pós-Graduação em Ciência da Computação – Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA), Belém –PA – Brasil

mauricio.ronny@uol.com.br, wallace.lira@gmail.com, srbo@ufpa.br **Abstract.** *This paper presents a systematized implementation of the Project Management process using the free software tools OpenProj, Redmine, Spider1.* Introdução

Abstract. *This paper presents a systematized implementation of the Project Management process using the free software tools OpenProj, Redmine, Spider-UCP, Spider-APF, Spider-CL and Subversion, based on the best practices suggested by MPS.BR.*

Resumo. *Este artigo apresenta uma implementação sistematizada do processo de Gerência de Projetos com o uso das ferramentas de software livre OpenProj, Redmine, Spider-UCP, Spider-APF, Spider-CL e Subversion, com base nas boas práticas sugeridas pelo MPS.BR.*

1. Introdução

Modelos de qualidade de software tendem a melhorar os processos organizacionais e mensurar a capacidade da organização em gerenciar o desenvolvimento, aquisição e manutenção dos produtos e serviços de software [Softex, 2009a]. Neste contexto, surgiu o MPS-BR (Melhoria de Processo de Software Brasileiro), que se trata de um programa desenvolvido pela SOFTEX (Associação para Promoção da Excelência do Software Brasileiro) voltado para a melhoria de qualidade em software. Este programa divulga um modelo de referência MR-MPS [Softex, 2009a].

Dentre os processos adotados pelo MR-MPS, destaca-se o processo de Gerência de Projetos (GPR). O objetivo deste processo 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, 2009b]. Gerir projetos inclui um conjunto de práticas gerenciais e técnicas que permite à equipe de software definir um roteiro, enquanto ela se move em direção à sua meta estratégica e seus objetivos táticos [Pressman, 2006]. Ainda, Gerência de Projetos é a disciplina de planejar, organizar e executar o Projeto SPIDER [Oliveira, 2009], contexto no qual este artigo está inserido, agrega subprojetos que analisam a aderência de cada processo do MPS.BR às ferramentas de software livre existentes, adaptando-as ou desenvolvendo novas ferramentas conforme necessidade. No contexto deste artigo, a ferramenta OpenProj, solução de software livre amplamente difundida para Gerência de Projeto, teve funcionalidades adicionadas para suportar as exigências do MPS.BR. Ainda, foi implementada a ferramenta Spider-CL com o objetivo de sistematizar a criação e aplicação de critérios objetivos nas análises promovidas pela organização.

Adicionalmente, foram desenvolvidas as ferramentas Spider-UCP e Spider-APF com o intuito de sistematizar a coleta e armazenamento de estimativas/métricas utilizando técnicas de Pontos Por Caso de Uso (*Use Case Points*) e Pontos por Função (*Function Points*). A versão customizada do OpenProj, a Spider-CL, a Spider-APF e a Spider-UCP encontram-se disponíveis em <http://www.spider.ufpa.br/>, acessando o item de menu Resultados.

Este trabalho objetiva analisar uma proposta de sistematização do processo de Gerência de Projetos do MR-MPS, utilizando um conjunto de ferramentas de software livre (todas as ferramentas utilizadas possuem licença GPL). O trabalho apresenta, na Seção 2, a metodologia de implementação deste processo com apoio das ferramentas OpenProj (versão customizada pelo Projeto SPIDER), Redmine, Subversion, Spider-CL, Spider-UCP e Spider-APF; em seguida, a Seção 3 faz uma análise da aderência das ferramentas ao MR-MPS. Por fim, a Seção 4 realiza uma visão geral do trabalho, apontando resultados.

2. Trabalhos Relacionados

Na linha de trabalhos apresentando propostas de implementação do MPS.BR a partir da utilização de ferramentas, Araújo e Mello (2009) propõe uma metodologia de uso da ferramenta Team Foundation Server para implementar o nível F do MR-MPS, (processo de gerência de projetos incluso), porém o artigo aborda a experiência de uso de uma ferramenta proprietária. Com foco em software livre, Cardias Junior (2010), Estácio (2010) e Souza (2010) apresentam estratégias de utilização de ferramentas de software livre para a implementação dos processos de Gerência de Requisitos, Medição e Gerência de Configuração, respectivamente, analisando, em seguida, a aderência das ferramentas elencadas com os resultados esperados do MPS.BR.

Este artigo aborda uma estratégia para a implementação do processo de Gerência de Projetos do MPS.BR baseada em boas práticas de uso de ferramentas de software livre, adaptadas e desenvolvidas no contexto de um projeto acadêmico, e analisa a aderência destas ferramentas semelhantemente aos artigos supracitados. Conjuntos de ferramentas para apoio à implementação MPS.BR também são vistos nas pesquisas [Rocha, 2005], que apresenta a estação TABA – um ambiente de desenvolvimento de software, e [Lima, 2006], sobre o WEbapsee- um ambiente de automação de processo de software. A presente pesquisa analisa uma estratégia de utilização de ferramentas já existentes de software livre (Redmine e OpenProj) em conjunto com ferramentas desenvolvidas no Projeto SPIDER (Spider-CL, SPIDER-UCP e Spider-APF) para apoio à implementação do MPS.BR.

3. Metodologia de Implementação do GPR com Apoio de Ferramentas Livre

Esta seção apresenta a metodologia proposta para implementação sistematizada do processo de GPR do MR-MPS, através do uso das ferramentas de software livre: (1) OpenProj (versão customizada no Projeto SPIDER), para definição e planejamento do projeto; (2) Redmine (disponível em <http://www.redmine.org/>), para o acompanhamento e revisão; (3) o Subversion (ou SVN, disponível em <http://subversion.tigris.org/>), como repositório de dados do projeto; (4) Spider-CL, para definição e aplicação de critérios objetivos, a partir de *checklists*; e (5) Spider-APF ou Spider-UCP, para coleta e

armazenamento de estimativas/métricas. Neste trabalho entende-se metodologia como sendo a codificação de um conjunto de práticas recomendadas, às vezes acompanhada de material de treinamento, programas de educação formal, planilhas, diagramas, tomando como parte um método (processo com uma série de passos, para construir um software) [Pressman, 2006]. Esta metodologia procura agregar boas práticas para o uso de ferramentas livres de apoio ao processo de GPR.

As ferramentas Spider-APF e Spider-UCP foram desenvolvidas pelo Projeto SPIDER. A primeira apresenta uma abordagem de coleta de estimativas/métricas a partir de Ponto por Função [Pressman, 2006], enquanto a segunda apresenta uma abordagem através de Pontos por Caso de Uso [Pressman, 2006]. A organização deve escolher a abordagem mais adequada às suas especificidades. Para mais detalhes acerca do uso destas ferramentas, consultar <http://www.spider.ufpa.br/index.php?id=resultados>.

3.1 Planejamento do projeto

Considerando um ambiente onde estejam devidamente instaladas e configuradas ferramentas de apoio ao processo de GPR, a metodologia inicia com a definição do Plano de Projeto. Este plano deve informar: (1) o escopo do projeto; (2) o dimensionamento dos produtos de trabalho; (3) a definição do modelo e das fases do ciclo de vida; (4) estimativas de esforço e custo; (5) definição do cronograma e orçamento; (6) análise dos riscos do projeto; (7) planejamento de recursos necessários; (8) identificação dos dados relevantes do projeto. Estas informações devem estar contidas no relatório gerado pela ferramenta OpenProj. É necessário, portanto, registrar estas informações na ferramenta, conforme visto na Figura 1.

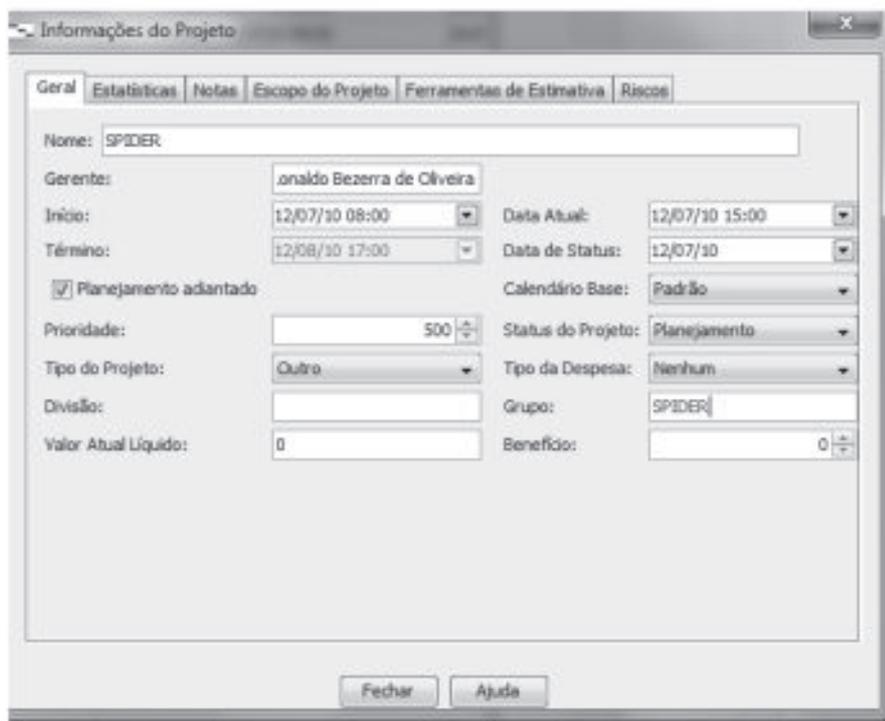


Figura 1. Informações do projeto: Descrição e detalhamento do projeto na ferramenta OpenProj

Inicialmente, devem ser registrados os recursos disponíveis para o projeto. O OpenProj possibilita planejar os recursos, utilizando a funcionalidade “Recursos”, permitindo identificar recursos humanos ou de infra-estrutura. Vale ressaltar que os recursos humanos envolvidos incluem todos os interessados no projeto, ou *stakeholders*.

É necessário, posteriormente, registrar no OpenProj as tarefas que devem ser realizadas, alterando a tabela de tarefas em sua interface inicial. Recursos necessários para as tarefas devem ser alocados nesta etapa. No ato do registro das tarefas, informações importantes para cada tarefa são inseridas no Plano de Projeto, tais como: data de início; data de finalização; duração; e prioridade. Quando as tarefas forem registradas, a ferramenta OpenProj é capaz de gerar um diagrama de EAP (Estrutura Analítica do Projeto), o qual pode ser utilizado para visualizar o escopo da aplicação.

Para dimensionar as tarefas adequadamente, deve ser consultado dos projetos anteriores: (1) o histórico de Planos de Projeto gerados no OpenProj; (2) o acompanhamento de cada tarefa na ferramenta de Gerência de Mudança; e (3) o histórico de estimativas/métricas no Spider-APF/Spider-UCP. Com base nestas informações, as estimativas/métricas para o novo projeto devem ser registradas no Spider-APF/Spider-UCP ao passo que a duração das tarefas deve ser feita no OpenProj, favorecendo, ainda, a definição do custo para realização de cada tarefa previamente especificada.

Após o registro dos recursos e as tarefas no OpenProj, deve ser preenchida a tabela referente à análise de riscos do projeto. Esta funcionalidade foi implementada pela equipe de desenvolvimento do Projeto SPIDER na ferramenta OpenProj. Para cada risco identificado, deve ser especificada uma descrição, probabilidade de ocorrência, impacto no projeto, as tarefas relacionadas ao risco e sua prioridade.

Finalmente, é necessário identificar os dados relevantes do projeto, que são as várias formas de documentação necessárias para a execução do mesmo, como atas de reuniões, relatórios, dados informais e demais artefatos importantes [Softex, 2009b]. Para tal, deve ser utilizada a nova funcionalidade “Informações do Projeto”, implementada pela equipe do Projeto SPIDER, no guia “Dados Relevantes”. Devem ser informados o nome do artefato e a tarefa ao qual o mesmo está relacionado. O tratamento sobre estes dados importantes inclui definir formas de armazenamento, disponibilidade, responsáveis e nível de controle apropriado, tarefas normalmente planejadas pelo processo de Gerência de Configuração [Souza et al, 2010]. A utilização de ferramentas de controle de versão e repositório, como o Subversion, contribui para a administração destes artefatos, de forma que todos os dados relevantes do projeto sejam colocados em um repositório e seu controle seja estabelecido através de metodologias de uso, definidas para a ferramenta durante o planejamento do projeto. Uma metodologia de uso da ferramenta Subversion é descrita em [Souza et al., 2010].

Depois do registro destas informações, o Plano de Projeto está completo. No entanto, é necessário disponibilizá-lo. Para facilitar este procedimento, a equipe de desenvolvimento do Projeto SPIDER implementou a funcionalidade Gerar Relatório na ferramenta OpenProj. Esta funcionalidade gera um arquivo no formato *pdf* (*Portable Document Format*) contendo os dados do Plano de Projeto.

3.2 Execução e Acompanhamento do Projeto

Antes de iniciar o projeto, o Plano de Projeto deve ser revisado. Em seguida, é necessário realizar um

estudo de viabilidade considerando o escopo do projeto e aspectos técnicos, financeiros e humanos da organização. Estas revisões têm como objetivo verificar a viabilidade de inicialização do projeto. Em caso de aceitação, o Plano de Projeto deve, posteriormente, ser revisado com todos os *stakeholders* para assegurar seu comprometimento com o planejamento. Uma possível estratégia para documentar este comprometimento é divulgar o Plano de Projeto gerado na ferramenta OpenProj através de um fórum na ferramenta Redmine, evidenciando o comprometimento através da resposta dos interessados em mensagens no fórum. É importante observar que o Plano de Projeto deve ser previamente revisado em uma reunião com a equipe.

No contexto de GPR, recomenda-se utilizar critérios objetivos para proceder a análise de viabilidade do projeto. Para tal, pode ser utilizada a ferramenta Spider-CL para registro de critérios objetivos, criação de *checklists* contendo os critérios definidos e posterior aplicação do *checklist*. Um checklist de Análise de viabilidade de projeto criado na ferramenta pode ser visualizado na Figura 2b.

Durante a execução do projeto, este deve ser acompanhado de forma a garantir que todos os planos sejam seguidos, a aderência ao cronograma, analisar a resolução de cada tarefa e o esforço empregado. Para contemplar essas práticas, as tarefas definidas no cronograma da ferramenta OpenProj podem ser instanciadas através de *Issues* (registros na ferramenta Redmine para retratar tarefas a serem executadas ou problemas a serem resolvidos), e então acompanhar o que fora planejado com o que é realizado, como visualizado na Figura 2a.

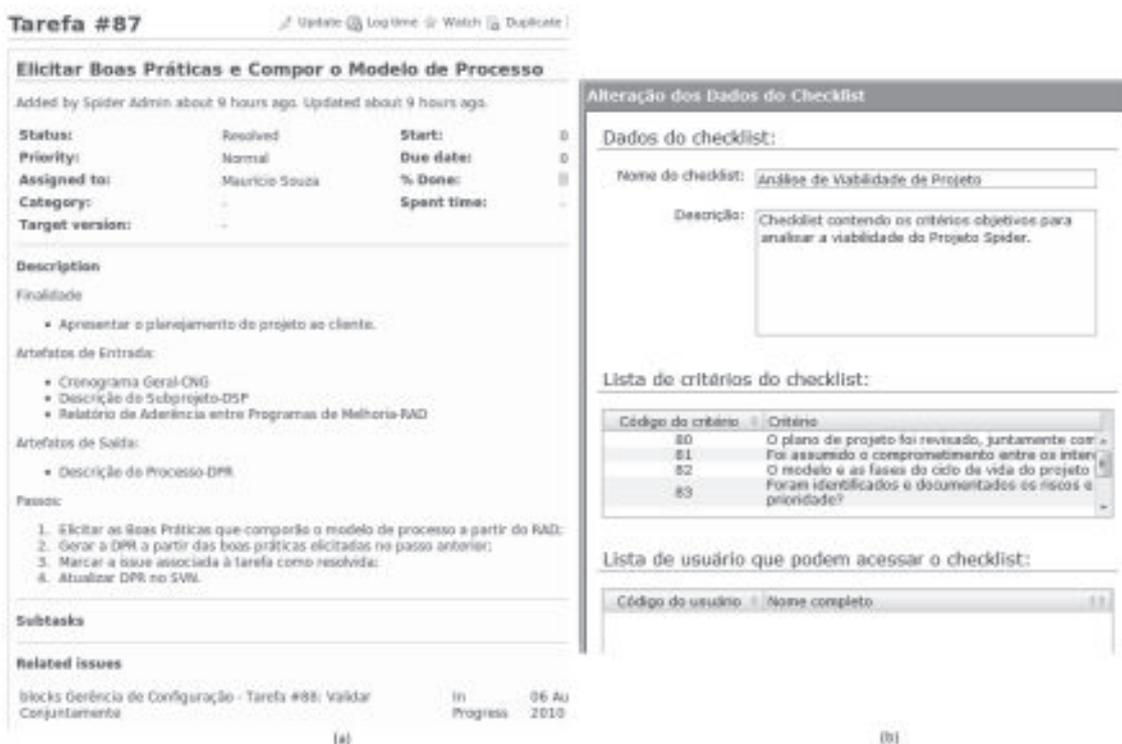


Figura 2. (a) Tarefa instanciada no Redmine.
(b) Checklist para análise de viabilidade criado na Spider-CL

É necessário registrar, na ferramenta Redmine, os *stakeholders* (como usuários) e as tarefas (como *issues*), ambos previstos no Plano de Projeto. Durante a definição dos interessados, é possível atribuir papéis que irão impactar nas permissões que os mesmos terão na ferramenta. Ao registrar *issues* de

tarefas no Redmine é possível definir responsáveis, especificar datas, descrever passos da atividade, definir relações entre tarefas (ordem de execução ou dependência) e acompanhar a evolução das tarefas. Os *issues* contam com um ciclo de vida customizável na ferramenta, baseado em estados, e um histórico registrando qualquer atualização realizada. Os *issues* registrados podem ser visualizados em forma de gráfico de Gantt, atualizado automaticamente conforme o registro, execução ou término de tarefas. Para verificar se o planejamento está aderente à realidade, deve-se comparar o cronograma definido na ferramenta OpenProj (planejado) com o definido na ferramenta Redmine (instanciado). Finalmente, o registro de *stakeholders* como usuários, com sua consequente associação aos *issues* do Redmine, tem como objetivo gerenciar a atuação destes atores no projeto.

3.3 Revisar e Gerir Problemas

Ao longo do projeto, revisões devem ser feitas em marcos e conforme estabelecidas no Plano do Projeto, para verificar o andamento do projeto e gerar insumos para decisões relacionadas à viabilidade de continuidade do mesmo. Os marcos de projeto podem ser definidos como “*Roadmaps*” na ferramenta Redmine, e uma descrição pode ser associada a este marco explicitando informações pertinentes para a caracterização do marco. As revisões podem ser instanciadas como *issues* no Redmine, definindo no sistema um novo *tracker* (tipo de *issues*) “Revisão”, que pode ser relacionada a um marco específico. Os resultados das revisões devem ser adicionados no próprio *issue* criado, como notas ao atualizar seu estado para resolvido [Souza *et al.*, 2010].

Durante as revisões do projeto, problemas devem ser identificados, analisados e registrados para uma posterior ação corretiva e a devida gerência destas ações até a conclusão das mesmas. Estes problemas podem ser registrados como *issues* no Redmine para a devida gerência, e estes problemas registrados devem ser definidos como oriundos do *issue* de revisão ao qual estão relacionados.

Com base na análise dos problemas identificados, ações corretivas devem ser estabelecidas para a solução dos problemas que podem impedir que o projeto alcance seus objetivos, e estas ações devem ter sua execução monitorada até sua conclusão. O sistema de *issues* do Redmine fornece apoio a esta atividade, visto que as não-conformidades registradas têm um ciclo de vida que descreve a sua solução desde quando foi criada até ser concluída. À medida que o estado do *issue* sofre alterações, informações podem ser adicionadas, como: as ações corretivas; responsáveis; e relatos das atividades realizadas. Associado aos *issues*, o Redmine dispõe de históricos os quais permitem acompanhar a execução das atividades relacionadas aos problemas [Souza *et al.*, 2010].

4. Aderência da Metodologia Proposta ao MR-MPS

A análise de aderência entre a metodologia proposta ao modelo MR-MPS é feita através do mapeamento das práticas sugeridas na Seção 2 com os resultados esperados do processo de Gerência de Projetos no Guia Geral do MPS.BR [Softex, 2009a], como visto na Tabela 1. É importante ressaltar que a metodologia se restringe a apresentar boas práticas para que seja possível contemplar os resultados esperados do processo de GPR em conjunto com práticas definidas no processo de software da organização. Resultado Esperado (RE) é “um resultado observável do sucesso do alcance do propósito do processo” [ISO/IEC, 2004]. No do processo de GPR do MPS.BR (até o nível F), são dezessete resultados esperados.

RE	Ferramentas	Funcionalidades/Práticas
GPR 1	OpenProj	Diagrama EAP gerado automaticamente.
GPR 2	OpenProj/ Spider-UCP/APF	Estimativas/Métricas registradas no Spider-UCP/APF; Duração das atividades registradas no OpenProj.
GPR 3	OpenProj	Estrutura do Gráfico de Gantt gerado automaticamente.
GPR 4	OpenProj	Consulta do histórico de Planos de Projeto.
GPR 5	OpenProj/ Redmine	Registro de tarefas no OpenProj; Registro de <i>issues</i> do tipo Tarefa no Redmine.
GPR 6	OpenProj	Tabela de riscos do projeto.
GPR 7	OpenProj / Redmine	Registro de recursos humanos no OpenProj; Adição dos usuários (registrados como recursos no OpenProj) como membros do projeto no Redmine.
GPR 8	OpenProj	Registro dos recursos de infra-estrutura.
GPR 9	OpenProj / Redmine / Subversion	Tabela de dados relevantes do projeto no OpenProj; Acesso ao repositório no SVN pelo Redmine; Repositório de dados do projeto disponibilizado pelo SVN.
GPR 10	OpenProj	Relatório gerado no formato <i>pdf</i> , contendo o plano do projeto.
GPR 11	OpenProj / Spider- CL	Revisão do relatório gerado; Aplicação do <i>checklist</i> pela Spider-CL.
GPR 12	Redmine	Criar fórum para publicar plano do projeto e obter comprometimento dos interessados.
GPR 13	Redmine	Instanciar e acompanhar tarefas planejadas no cronograma a partir de <i>issues</i> .
GPR 14	OpenProj / Redmine	Registro de <i>stakeholders</i> no OpenProj; Registro de <i>stakeholders</i> no Redmine.
GPR 15	Redmine	Registro de marcos como <i>Roadmaps</i> e revisões como <i>issues</i> .
GPR 16	Redmine	Registrar problemas identificados nas revisões como <i>issues</i> , consequentes dos <i>issues</i> de revisão.
GPR 17	Redmine	Registrar um plano de ação para a solução dos problemas no próprio <i>issue</i> , e acompanhar a sua resolução pelo ciclo de vida e histórico dos <i>issues</i> .

Tabela 1. Mapeamento da Metodologia com os Resultados Esperados de GPR do MPS

5. Conclusões

Como observado em lições aprendidas de implementações realizadas no programa de melhoria do processo, o emprego de ferramentas sistematizadas pode reduzir consideravelmente o tempo de implementações do MR-MPS, e, tratando-se de ferramentas livres, há também uma redução de custos para a organização.

Com o objetivo de agilizar o processo de implementação do programa MPS-BR, este trabalho apresentou uma solução para implementação do processo de GPR, utilizando ferramentas de software livre. Apenas o uso da metodologia não é suficiente para atender o MPS-BR, porém define boas práticas para uso do ferramental de forma aderente ao MR-MPS. Vale ressaltar que as boas práticas apresentadas neste trabalho encontram-se institucionalizadas no Projeto SPIDER (www.spider.ufpa.br), que objetiva a concepção, elaboração e construção de ferramentas de software livre para apoiar a implementação dos processos constantes no MR-MPS.

6. Agradecimentos

Este trabalho está recebendo o apoio financeiro do CNPq e CAPES a partir de bolsas de estudo de Mestrado.

Referências

Araujo, L.L., Mello, A. B., "Implantação do MPS.BR (Melhoria do Processo de Software Brasileiro), Nível F, com TFS (Team Foundation Server) no Desenvolvimento Eficiente de Sistemas". V Workshop Anual do MPS (WAMPS 2009). Campinas, SP, 2009.

Cardias Junior, A. B. et al. (2010) "Uma Análise Avaliativa de Ferramentas de Software Livre no Contexto da Implementação do Processo de Gerência de Requisitos do MPS.BR". Anais do WER10 - Workshop em Engenharia de Requisitos (pp 75-84), Cuenca, Equador, Abril, 2010.

Estácio, B. Oliveira, S. R. B. (2010) "Spider-MPlan: Uma Ferramenta para Apoio ao Processo de Medição do MPS.BR". XI Workshop de Software Livre (WSL 2010). Porto Alegre, RS, Brasil, 2010.

the International Organization for Standardization and the International Electrotechnical Commission (2004) "ISO/IEC 15504-2: Information Technology - Process Assessment - Part 2: Performing an Assessment". Geneve.

Lima, A. et al. (2006) "WebAPSEE: Um Ambiente Livre e Flexível Para Gerência de Processos de Software". VII Workshop de Software Livre (WSL 2006). Porto Alegre, RS, Brasil, 2006.

Oliveira, S.R. B. (2009) "SPIDER - Uma Proposta de Solução Sistêmica de um SUITE de Ferramentas de Software Livre de apoio à implementação do modelo MPS.BR". Projeto de Pesquisa. ICEN - Universidade Federal do Pará, Belém.

Pressman, R. S. Engenharia de software. Tradução José Carlos Barbosa dos Santos. Rio de Janeiro: McGraw-Hill, 2006. 1056 p.

Rocha, A. R. et al. "Estação TABA: Uma Infra-estrutura para Implantação do Modelo de Referência para Melhoria de Processo de Software". IV Simpósio Brasileiro de Qualidade de Software (SBQS'05). Porto Alegre, Brasil, 2005. Disponível em < [http://ramses.cos.ufrj.br/ta/](http://ramses.cos.ufrj.br/ta/ta/) >. Acesso em 20 mar. 2009.

SEI (2006) "Capability Maturity Model Integration for Software Engineering/Systems Engineering/Integrated Product and Process Development/Acquisition". Version 1.2, Staged Representation, Carnegie Mellon, USA.

SOFTEX - Associação Para Promoção Da Excelência Do Software Brasileiro. (2009a) "MPS.BR – Guia Geral:2009". Disponível em: www.softex.br.

SOFTEX - Associação Para Promoção Da Excelência Do Software Brasileiro. (2009b) "MPS.BR – Guia de Implementação – Parte 1:2009". Disponível em: www.softex.br.

Souza, M. R. A., Lira, W. M. P., Oliveira, S. R. B. (2010) "Uma Implementação do Processo de Gerência de Configuração usando o Redmine, o SVN e a Spider-CL", Anais do III Workshop de Gerenciamento de Projetos de Software (WGPS).

O Uso do dotProject no Processo Prodepa de Desenvolvimento de Software

Renata W. Monteiro¹, Leila Daher¹, Cláudio R. de Lima Martins¹

¹Diretoria de Desenvolvimento de Sistemas - Empresa de Processamento de Dados do Estado do Pará (PRODEPA) – Belém – PA – Brasil

{renata.monteiro, leila.daher, claudio.martins}@prodepa.pa.gov.br

Abstract. *The paper presents the PRODEPA's customization of the open source software dotProject, which is a project management application. The customization that has been developed by PRODEPA aims to make dotProject compatible with PPDS (Processo Prodepa de Desenvolvimento de Software). The paper shows the new features included in the application and aims to support the PPDS, whose methodology is based on the practices presented in the MPS-BR.*

Resumo. *O artigo relata as adaptações feitas pelo PRODEPA no software livre para gerenciamento de projetos, dotProject, a fim de torná-lo compatível com o PPDS (Processo Prodepa de Desenvolvimento de Software). O relato apresenta as novas funcionalidades inseridas na ferramenta dotProject e visa auxiliar o PPDS, cuja metodologia é baseada nas práticas presentes no MPS.BR.*

1. Introdução

Uma metodologia para desenvolvimento de software visa formalizar os processos inerentes a cada etapa da produção do software, estabelecendo padrões, gerando e gerenciando documentação de apoio e garantindo qualidade em cada etapa isoladamente e na linha de produção como um todo.

Desde 2007, após a implantação do processo de software aderente ao MR-MPS, que resultou na avaliação nível F do MPS-BR (MPS.BR, 2010), a Empresa de Processamento de Dados do Estado do Pará (PRODEPA) vem implantando sua fábrica de software. Para isso, vem adotando uma metodologia de desenvolvimento de sistemas baseada em boas práticas de engenharia de software, aperfeiçoando as atividades e padronizando o uso de ferramentas que auxiliam os profissionais na produção de software, buscando diminuir o esforço demandado para o controle do processo. Essa metodologia faz parte do Processo Prodepa de Desenvolvimento de Software (PPDS).

Após pesquisas realizadas em uma variedade de ferramentas disponíveis no mercado, a PRODEPA optou em adotar o dotProject (DOTPROJECT, 2010), versão 2.1.2. Dentre os critérios utilizados para a sua adoção, estão: ser uma ferramenta web, utilizar tecnologias abertas e baseada em software livre, e, conseqüentemente, permitir realizar as personalizações necessárias para adequá-la à metodologia da empresa. Além disso, a ferramenta deve possuir facilidades de integração com outros softwares adotados no PPDS.

Este artigo está organizado em seis seções. A seção 2 comenta sobre a ferramenta dotProject e seu objetivo; na seção 3, são abordadas as novas funcionalidades inseridas na ferramenta; na seção 4, é apresentada a

utilização do DotProject na fábrica de Software da Prodepa; na seção 5, são apresentadas as considerações finais.

2. O dotProject

O dotProject (DOTPROJECT, 2010) é uma ferramenta baseada em software livre para apoio à gerência de projetos. Possui funcionalidades que são aderentes aos resultados esperados e às boas práticas descritas no nível F de maturidade do MR-MPS, entre elas: (a) o planejamento do projeto com seus marcos estabelecidos; (b) planejamento de recursos; (c) cronograma; (d) acompanhamento “planejado x realizado”; (e) gráfico de Gantt, para acompanhamento e controle dos projetos; (f) detalhamento do processo e suas atividades; (g) reutilização do processo padrão; (h) as informações do projeto são comunicadas pela ferramenta; (i) histórico do projeto; (j) controle de acesso dos projetos; (k) coleta e armazenamento de medidas.

O foco principal da ferramenta é o apoio ao gerenciamento de projetos e possui, dentre as várias qualidades, uma de grande valor para o PPDS: os recursos para comunicação. Os processos de gerenciamento das comunicações do projeto fornecem as ligações críticas entre pessoas e informações que são necessárias para comunicações bem sucedidas (PMBOK, 2003). A ferramenta dotProject possui a opção de enviar à equipe de um projeto e aos interessados, todas as informações necessárias para que os responsáveis tomem conhecimento de suas atividades, assim como compartilhem as informações na própria ferramenta, evitando desperdício de tempo no emprego de ferramentas e práticas informais de comunicação, como o uso de email.

3. Descrição das novas funcionalidades

Segundo ROCHA et al. (2005), o apoio ferramental são fatores que influenciam positivamente quando estão fortemente presentes durante a implementação de processo. O dotProject foi customizado de forma a ser utilizado como ferramenta para a gerência dos projetos e coleta das informações para geração de métricas. Para aumentar a produtividade de suas atividades, a Prodepa implementou na ferramenta recursos que agregassem tanto o acompanhamento de seu processo padrão (PPDS), quanto o planejamento de cada um dos projetos desenvolvidos nesse processo. Além disso, padronizou ferramentas de análise e projeto, *bug tracking* e prototipação, adotando, respectivamente, o *Enterprise Architect* (EA, 2010), *Bugzilla* (BUGZILLA, 2010) e *Pencil Project* (PENCIL, 2010).

Diante da necessidade de novas funcionalidades que facilitassem o dia-a-dia dos líderes de projeto e integrassem as ferramentas, a Prodepa optou em criar, inicialmente, o requisito funcional de importação dos dados da ferramenta EA para o dotProject. Ao utilizar esta funcionalidade, os casos de uso criados no EA são importados para acompanhamento no dotProject com as atividades filhas estabelecidas (vide Figura 3), sendo elas: análise, projeto, construção e teste. Como resposta desta personalização, a Prodepa diminuiu o esforço de alteração do processo.

Para acesso à importação, o colaborador tem que ser integrante da equipe e estar autenticado no projeto que está definido no dotProject. Após a validação do acesso, deve-se selecionar o pacote com os casos de uso que deseja importar. Na Prodepa, o desenvolvimento de sistemas é realizado por iterações. Para a importação é criado um pacote no EA com o nome da iteração, e dentro deste é colocado todos os casos

de uso da respectiva iteração, como ilustrado na figura 1. Depois da escolha do pacote, a importação é realizada e apresentada na tela do projeto, como visto na figura 2.

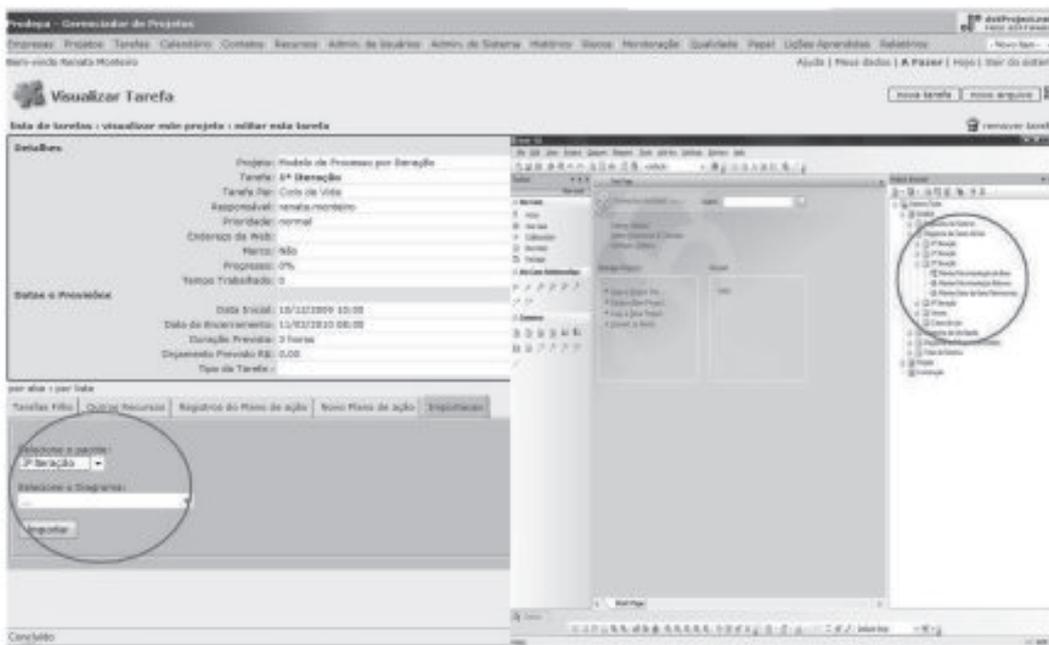


Figura 1. Tela de escolha dos Casos de Uso a serem importados no dotProject e Tela do EA com os pacotes organizados por iteração

Nome	Ramo	Resultado	Nome da Tarefa	Atividade	Local de Dependência	Data Inicial	Duração	Data Final	Nome Atualizado	Ativo
✓	---	100%	FASE 1 - PLANEJAMENTO DO PROJETO			10/03/2008	28 semanas	10/03/2008		✓
✓	---	100%	Estabelecimento Geral do Projeto			10/03/2008	28 semanas	10/03/2008		✓
✓	Requisitos	100%	Estabelecer o Termo de Abertura do Projeto	Termo de Abertura do Projeto	dependência (100%)	10/03/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Identificar o Escopo do Projeto	Definição do Escopo	dependência (100%)	10/03/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Elaborar Requisitos de Cliente	Especificação de Requisitos	dependência (100%)	10/03/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Realizar Estimativas	Manilha de Estimativas	dependência (100%)	10/03/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Maneja o Processo BAC O Projeto	Plano do Processo	dependência (100%)	10/03/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Maneja o Projeto	Plano do Projeto	dependência (100%)	10/03/2008	18 semanas	10/03/2008		✓
✓	Requisitos	90%	Crear Logonama para o sistema	Logonama	metadependência (100%)	03/04/2008	1 semana	10/03/2008		✓
✓	---	64%	UML de UML			03/04/2008	19 semanas	10/03/2008		✓
✓	---	64%	UML Iteração			03/04/2008	40 semanas	10/03/2008		✓
✓	Requisitos	100%	Produzir Diagrama de Classe	Diagrama de Classe	dependência (100%) (x-1)	03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	100%	Crear Base de	Base de		03/04/2008	1 semana	10/03/2008		✓
✓	Requisitos	8%	Implementar Funcionalidade	Atividade do Produto	dependência (100%) (x-1)	03/04/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Produzir Modelo de Dados	Modelo	dependência (100%) (x-1)	03/04/2008	18 semanas	10/03/2008		✓
✓	---	100%	UML Master Control			03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	100%	Análise	Caso de Uso	dependência (100%)	03/04/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Desenvolvimento	Código Fonte	dependência (100%)	03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	100%	Teste	Relatório de teste	dependência (100%)	03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	100%	Projeto	Diagrama de Sequência UML/UML	dependência (100%) (x-1)	03/04/2008	1 semana	10/03/2008		✓
✓	---	100%	UML Master Control			03/04/2008	40 semanas	10/03/2008		✓
✓	Requisitos	100%	Análise	Caso de Uso	dependência (100%)	03/04/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Desenvolvimento	Código Fonte	dependência (100%)	03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	100%	Teste	Relatório de teste	dependência (100%)	03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	100%	Projeto	Diagrama de Sequência UML/UML	dependência (100%) (x-1)	03/04/2008	1 semana	10/03/2008		✓
✓	---	100%	UML Master Control			03/04/2008	40 semanas	10/03/2008		✓
✓	Requisitos	100%	Análise	Caso de Uso	dependência (100%)	03/04/2008	1 semana	10/03/2008		✓
✓	Requisitos	100%	Desenvolvimento	Código Fonte	dependência (100%)	03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	90%	Teste	Relatório de teste	dependência (100%)	03/04/2008	18 semanas	10/03/2008		✓
✓	Requisitos	100%	Projeto	Diagrama de Sequência UML/UML	dependência (100%) (x-1)	03/04/2008	1 semana	10/03/2008		✓

Figura 2. Apresentação dos Dados Importados da Ferramenta EA na ferramenta dotProject

Após a implementação desse requisito na ferramenta, percebeu-se que outras necessidades deveriam ser implementadas na personalização do dotProject. A motivação para a personalização da ferramenta é tentar automatizar algumas atividades do PPDS, e atender as melhores práticas abordadas no MR-MPS e aos padrões da empresa. Tais funcionalidades não existem na versão original do dotProject, por esse motivo foram implementadas na personalização da ferramenta. As novas funcionalidades são brevemente descritas na tabela 1.

Funcionalidade	Descrição
Módulo de Garantia da Qualidade	Tem a finalidade de sistematizar os laudos de avaliação dos produtos de trabalho, garantindo a qualidade do que é produzido através da utilização de <i>checklist</i> de verificação. Em caso de correções, é necessário o registro de ação, conforme descrito no próximo item.
Registrar Planos de Ação	Visa registrar as ações para as correções de não-conformidades ou desvios identificados. A notificação ao responsável é feita pela própria ferramenta e o acompanhamento é realizado através da situação de cada ação que tem como opções: definido, em execução, finalizado e cancelado.
Registrar mudanças	Tem como objetivo guardar as informações relativas as solicitações de mudança e comunicar aos interessados os artefatos que devem ser alterados.
Monitoração do Projeto	Acompanhamento do projeto durante o período, com a opção de geração de <i>baseline</i> do projeto, assim como a geração do gráfico de esforço planejado x esforço realizado.
Módulo Risco do Projeto	São registrados os possíveis riscos qualitativos do projeto com seu impacto, probabilidade de ocorrência e contingência. Caso ocorra o risco, deve-se fazer a anotação do acontecimento para registro e acompanhamento.
Módulo de Papéis	A criação deste módulo tem o intuito de maior aderência da ferramenta ao PPDS, já que a PRODEPA trabalha com o modelo de fábrica de software e tem seus papéis bem definidos para cada atividade.

Tabela 1. Novas Funcionalidades implementadas no dotProject

As funcionalidades **Monitoração do Projeto** e **Módulo Risco do Projeto**, foram desenvolvidos com base em um trabalho realizado com o objetivo de atender os resultados esperados de planejamento, monitoração e controle de projetos dos modelos CMMI-DEV e MPS.BR [DotProjectDEMO, 2010].

4. Exemplo da utilização do DotProject na fábrica de software da Prodepa

Em termos operacionais, é importante demonstrar-se o quanto a ferramenta agiliza e controla o processo de desenvolvimento do software dentro da Prodepa. Para isso, será dada uma breve descrição do fluxo operacional e da utilização da ferramenta na fábrica de software.

Hoje, a fábrica de software da Prodepa trabalha de forma organizada, pois todas as etapas do desenvolvimento de um projeto são padronizadas obedecendo ao PPDS. O processo inicia da forma descrita a seguir.

O líder do projeto instancia o projeto no dotProject, reutilizando o modelo do processo padrão e inicia o planejamento.

Após a instanciação do processo, o responsável pela configuração é notificado a criar os ambientes do projeto, como: EA para modelagem do sistema e Subversion (também conhecido por SVN) para controle de versão, de forma a permitir que sejam armazenados todos os artefatos referentes ao projeto. Os diretórios possuem uma estrutura padrão pré-definida que é adotada por todos os projetos, como pode ser visto na figura 3. Após a criação do ambiente, o líder acompanha o planejamento do projeto e o analista inicia o trabalho conforme o PPDS.

Projeto	Usuário
COBEN-Sistema de concessão benefícios hansenianos	Alon Anxari
	Bernardo Lima
	Edivaldo Santana
Framework Pluraquôô 1.3	Adriana Muniz
	Adriana Teles
	Antonio Junior
	Emerson Bonneterre
	Evandro Feres
	Heber Cunha
	José Fernandes
Garantia Site SCIR	Thiago Soares
	Valtoni Soavevura
	Clávia Machado

Figura 4. Tela do dotProject com as pessoas alocadas nos projetos

Data	Iní	Fim	Nome	Uff	Local	Hora	Valor de Custo	Comentários
20/05/2018 08:00	+	Desenvolvimento		adriana.muniz		8,00	36,000	Implementação do método de pesquisa
21/05/2018 14:00	+	Desenvolvimento		adriana.muniz		8,00	36,000	Correção de pagamento nas entidades
22/05/2018 18:45	+	Desenvolvimento		adriana.muniz		8,00	36,000	Implementação do método de pesquisa e cadastro
23/05/2018 08:00	+	Desenvolvimento		adriana.muniz		8,00	36,000	Correção de pagamento nas entidades
24/05/2018 18:45	+	Desenvolvimento		adriana.muniz		8,00	36,000	Implementação dos métodos de pesquisa, cadastro e alteração
25/05/2018 08:00	+	Desenvolvimento		adriana.muniz		8,00	36,000	Realizar vários ajustes no formulário de cadastro
24/05/2018 08:00	+	Desenvolvimento		adriana.muniz		8,00	36,000	Trabalhar na implementação do cadastro e alteração.
26/05/2018 08:00	+	Desenvolvimento		adriana.muniz		8,00	36,000	Ajustes no Mapeamento das classes que tinham relacionamento com o município criado a classe Município(ouco). Alteração na lógica de pesquisa de centros e alteração nos forms de pesquisa e alteração
21/05/2018 18:45	+	Desenvolvimento		adriana.muniz		8,00	36,000	Implementação da tela de alteração de centro.
24/05/2018 14:00	+	Desenvolvimento		adriana.muniz		8,00	36,000	Implementação da tela de alteração de centro.
25/05/2018 08:00	+	Desenvolvimento		adriana.muniz		3,00	12,000	Ajustes na alteração de centro.
25/05/2018 08:00	+	Desenvolvimento		maria.rita		8,00	36,000	Atualização de layout.
HORA TOTAL =							53,00	

Figura 5. Tela do dotProject com os registros de horas de trabalho

Durante o desenvolvimento do projeto podem ocorrer desvios e solicitações que não foram planejadas na concepção do projeto. Para manter o controle e gerenciá-las ao longo do ciclo de vida do projeto, caso haja solicitação de mudança, estas ficam registradas com a situação (solicitado, aprovado ou rejeitado), compondo um histórico de mudanças solicitadas pelo cliente, como pode ser visto na figura 6 (tela de Registro de Solicitação de Mudança). Porém, para o acompanhamento de problemas e desvios, ações para corrigir e prevenir são necessárias. Para isso há o registro do plano de ação, como mostra a figura 7.



Figura 6. Registro de Solicitação de Mudança do Projeto

O que será feito	Como executar	Por que isso é necessário	Como será realizado	Onde	Criado	Situação atual	Salvar
No item 5, adicionar o Controle de Acesso será lido com o BOM	Lorraine Zani 05/04/2008 05/04/2008	Para manter a consistência de descrição do projeto	Na documentação de declaração de escopo	Na local de trabalho	Lorraine Zani	Finalizado	[Icon]
No item 8 - instalação, retirar a restrição informada	Lorraine Zani 05/04/2008 05/04/2008	Porque não se trata de restrição	Na documentação de declaração de escopo	Na local de trabalho	Lorraine Zani	Finalizado	[Icon]
No item 9 - premissas, considerar apenas a primeira premissa como premissa e restante pode ser exclusão	Lorraine Zani 05/04/2008 05/04/2008	Porque não se trata de premissa	Na documentação de declaração de escopo	Na local de trabalho	Lorraine Zani	Finalizado	[Icon]

Figura 7. Registro do Plano de Ação para Correção de Desvios

O monitoramento do Projeto também é facilitado, pois com a opção de geração de *baseline* do projeto, pode-se ter uma visão mais acurada do "planejado x realizado". Na Figura 8 é mostrada a tela com um exemplo da aplicação da *baseline* no relatório de monitoração do projeto.

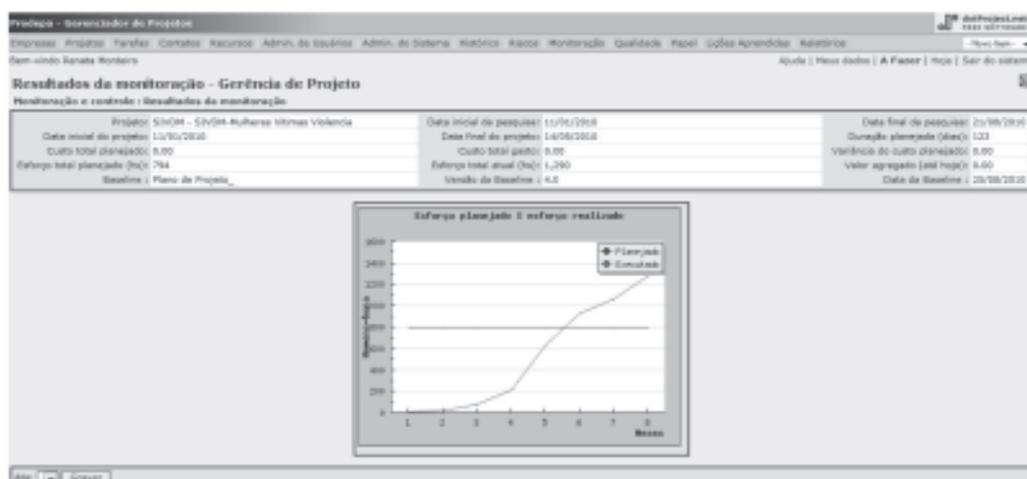


Figura 8. Relatório de Monitoração do Projeto

O acompanhamento dos projetos pela gerência da fábrica é simplificada, pois o dotProject permite o acompanhamento geral dos projetos através de consultas na tela de forma resumida. A Figura 9 exemplifica o relatório de acompanhamento dos projetos.

	Progresso(%)	Projeto	Usuario	Papel	Previsão Fim
1	95.6	Reformulação Portal Forpecap	Adeleide Evangelista		06/06/2011
2	0.8	Framework Mairaquilã 1.3	Adriana Muniz		23/07/2010
3	22.7	Manutenção eProtocolo			11/06/2010
4	94.3	STVDH Mulheres Víctimas Violencia			13/06/2010
5	0.8	Framework Mairaquilã 1.3	Adriana Talas	Desenvolvedor	23/07/2010
6	22.7	Manutenção eProtocolo		Desenvolvedor	11/06/2010
7	86.0	SCPMSS-Sistema de Cadastro de Propriedades Rurais		Desenvolvedor	30/08/2010
8	0.1	CTUPOL-Módulo de Operar Utilizar Utilizante		Parametrizador	13/06/2010

Figura 9. Acompanhamento do Projeto

5. Considerações Finais

A aplicação do MPS-BR nas empresas é bastante trabalhosa e até inviável sem o uso de ferramentas adequadas que auxiliem na realização das atividades previstas nos processos para desenvolvimento de um software. Na fábrica de software da Prodepa, a utilização da ferramenta DotProject aponta como uma alternativa para resolução de grande parte desses problemas.

Mesmo após a inclusão de novas funcionalidades no dotProject, há a necessidade de alguns aperfeiçoamentos para evolução da ferramenta. A ênfase é a integração com outras ferramentas, e a tentativa de automatizar e semi-automatizar algumas atividades do PPDS que ainda são realizadas manualmente. Para isso, a Prodepa vem realizando estudos de viabilidade, de forma a identificar pontos de melhoria, e propor a construção de módulos no dotProject que solucionem as dificuldades observadas. Entre as melhorias, podemos citar: (i) a integração com a ferramenta Mantis bug tracker (MANTIS, 2010); (ii) criação de consultas e relatórios que apoiem a análise do processo de medição, visto que a coleta de medida é registrada, no dotProject, na atividade que esta sendo desenvolvida, proporcionando um maior controle e acompanhamento do que é produzido pela equipe; (iii) relatórios gerenciais que facilitem a distribuição da equipe nos projetos; (iv) integração com o LDAP, para utilização do serviço de autenticação de usuário.

As customizações efetuadas na ferramenta dotProject, aliadas com a utilização de uma metodologia bem definida, solidificaram o PPDS – Processo Prodepa Desenvolvimento de Software.

A ferramenta está disponível para acesso no endereço <http://homologar.prodepa.gov.br/dotproject/> utilizando o usuário e senha wamps2010.

Referências

BUGZILLA. Disponível em <<http://www.bugzilla.org/>>. Acesso em 17/ago/2010

DOTPROJECT. Disponível em <<http://www.dotproject.net/>>. Acesso em 17/ago/2010.

DotProjectDEMO. Disponível em: <<http://projetos.telemedicina.ufsc.br/demo/>>. Acesso em: 17/ago/2010.

EA. Disponível em: <<http://www.sparxsystems.com.au/>>. Acesso em: 17/ago/2010.

MANTIS. Disponível em: <<http://www.mantisbt.org/>>. Acesso em: 17/ago/2010.

MPS.BR (2010) “Mps.Br – Melhoria de Processos do Software Brasileiro”, <http://www.softex.br/mpsBr/>, Fevereiro.

PENCIL. Disponível em <<http://pencil.evolus.vn/en-US/Home.aspx>>. Acesso em 17/ago/2010.

PROJECT MANAGEMENT INSTITUTE. A Guide to the Project Management Body of Knowledge – PMBOK® Guide 2004 Edition. Third Edition.ed.

Rocha, A. R. et al. (2005). Fatores de Sucesso e Dificuldades na Implementação de Processos de Software Utilizando o MR-MPS e o CMMI. In: I Workshop de Implementadores MPS.BR. Brasília-DF: COOPE/UFRJ.

WebAPSEE Pro: Um Ambiente de Apoio a Gerência de Processos de Software

Ernani Sales, Anderson Costa, Murilo Sales, Carla Lima Reis, Rodrigo Reis
Laboratório de Engenharia de Software (LABES) - Instituto de Ciências Exatas e Naturais – Universidade Federal do Pará (UFPA) – Belém – PA – Brasil
QR Consultoria e Serviços LTDA – Belém – PA – Brasil
{ernani,anderson,murilo,clima,quites}@webapsee.com

Abstract. *WebAPSEE Pro environment is one of the current alternatives to support Software Process Management concerning current standards and quality models. This paper presents an overview about this environment and describes the automated support MR-MPS level G processes (Project Management and Requirements Management) implementation.*

Resumo. *O ambiente WebAPSEE é uma das alternativas atuais para apoiar a gerência de processos de software com objetivo de atender padrões e modelos de qualidade existentes. Neste artigo é apresentada uma visão geral do uso do ambiente e é detalhado o apoio fornecido aos processos requeridos pelo Nível G do MR-MPS: Gerência de Projetos e Gerência de Requisitos.*

1. Introdução

Atualmente, existem diversos modelos e normas de qualidade definidos para atender as necessidades das empresas em melhoria de processos de software [ABNT NBR/ISO/IEC 12207, 2009] [ISO/IEC 15504 2003] [SEI 2010]. Baseado nisso foi definido o Modelo de Referência para Melhoria de Processo do Software Brasileiro (MR-MPS) [SOFTEX 2009]. Contudo, muitas empresas possuem dificuldades em produzir software de qualidade alinhado a esses modelos de referência [Wangenheim 2007] [MCT 2005].

Segundo Rocha *et al.* [2006], os principais fatores que influenciam positivamente, quando fortemente presentes, e os que influenciam negativamente, quando fracos ou ausentes, na implantação de processos de software são: comprometimento da empresa, grau de acompanhamento dos processos, disponibilidade de recursos, motivação da empresa, apoio ferramental e treinamento.

Diante dos fatores citados anteriormente, é importante considerar ações que facilitem a maneira como estas normas e modelos de qualidade de software são implantados, por exemplo: o apoio de abordagens e de ferramentas automatizadas que tornem viável a implantação e melhoria de processos de software sem aumento de prazo e custo de desenvolvimento.

Este artigo apresenta o ambiente WebAPSEE Pro e de que forma fornece apoio automatizado aos processos do nível G do MPS.BR. Vale ressaltar que, apesar do foco deste artigo evidenciar o nível inicial do modelo, outros resultados de níveis superiores podem ser apoiados com a utilização desse

ambiente, mas a análise dos mesmos está fora do escopo deste artigo.

A organização do texto é dada como segue: na seção 2, é apresentado o ambiente WebAPSEE Pro, em termos de funcionalidades e potencialidades. Nas seções 3 e 4, é apresentado como o ambiente apoia, respectivamente, as atividades inerentes aos processos de Gerência de Projetos e Gerência de Requisitos. Por fim, na seção 5, são apresentadas as considerações finais.

2. WebAPSEE Pro

O ambiente WebAPSEE Pro (Figura 1), evolução do software livre WebAPSEE [Lima *et al.* 2006], tem como objetivo fornecer apoio automatizado para a gestão de processos de software. Dessa forma, o ambiente foi projetado para permitir a integração de vários serviços interrelacionados que auxiliem na execução das tarefas dos envolvidos durante todo o ciclo de vida do processo de software. Isto é, o ambiente apoia desde a concepção e levantamento de requisitos até a geração de análise *post mortem* dos processos, passando pelo controle da execução dos processos de forma bastante flexibilizada - implementando um modelo de execução descrito por Lima Reis [2003].

Com relação à execução de processos, o ambiente foi desenvolvido com base no entendimento que as soluções deste contexto devem estar integradas com a linguagem de modelagem de processos. Assim, a modelagem e execução de processos são atividades que ocorrem freqüentemente de forma intercalada. Isto ocorre devido à possibilidade de se executar processos parcialmente definidos [Lima Reis 2003]. Esse requisito representa uma situação na qual o gerente pode iniciar a execução de um processo enquanto decide como uma atividade futura será realizada.

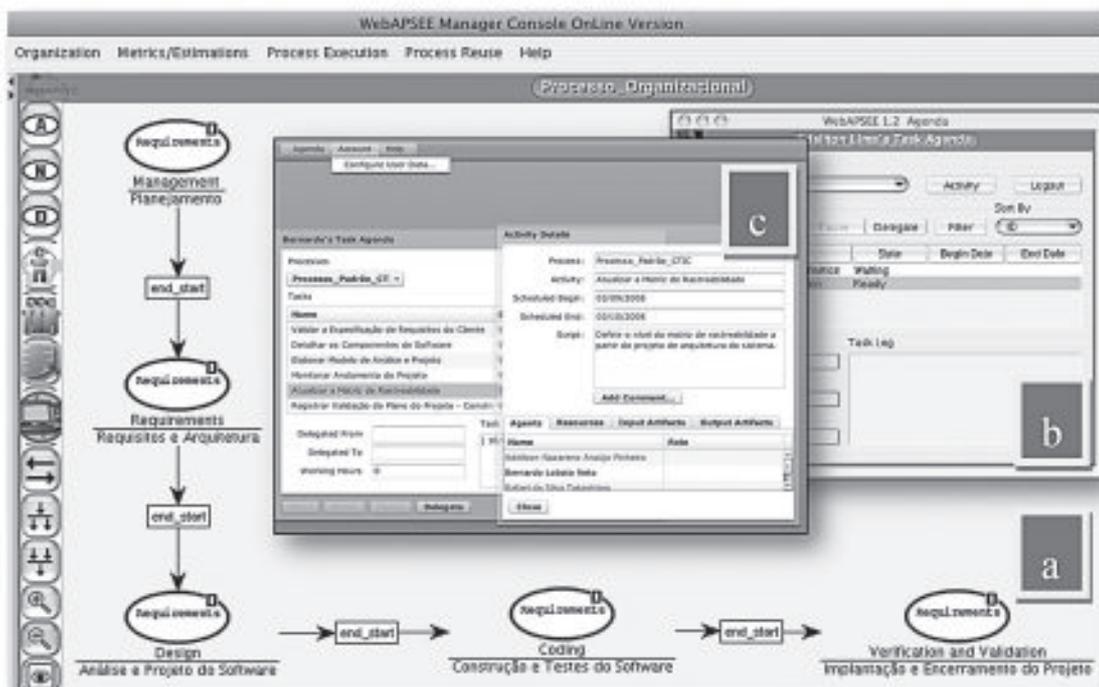


Figura 1. Ambiente WebAPSEE Pro: Manager Console, Task Agenda e Web Agenda

O ambiente WebAPSEE Pro implementa uma arquitetura cliente/servidor, que contém três clientes: (a) *Manager Console*, direcionado aos gerentes, que permite a definição, planejamento e acompanhamento da execução de processos de software, além do gerenciamento dos dados organizacionais, coleta de métricas, geração de relatórios, etc.; (b) *Task Agenda Desktop*, que provê às pessoas alocadas todas as informações necessárias para execução de suas atividades (prazos, artefatos de entrada e de saída, outros envolvidos, estimativa de horas, dentre outras informações), além de permitir o *feedback* sobre o andamento das tarefas a partir da interação (ações de iniciar, pausar, delegar, finalizar) com a máquina de execução do ambiente; e (c) *Task Agenda Web*, similar a *Task Agenda Desktop*, entretanto desenvolvida utilizando tecnologia web.

3. Apoio à Gerência de Projetos

As funcionalidades do ambiente WebAPSEE Pro podem ser agrupadas em seis macro-funcionalidades: Gerência da Organização, Definição e Implantação de Processos, Controle e Monitoração de Projetos, Gerência de Configuração, Medição e Análise do Projeto, e Gerência Integrada de Projeto. Cada macro-funcionalidade agrupa um conjunto de funcionalidades fornecido pelo ambiente. A seguir serão apresentados de forma mais detalhada essas macro-funcionalidades.

A **gerência da organização** contempla o gerenciamento de habilidades, papéis, recursos humanos (agentes) e materiais, grupos, ferramentas, artefatos, sistemas, projetos e organizações (tanto a própria organização que utiliza o ambiente quanto seus clientes). Além disso, é possível gerenciar uma hierarquia de tipos, visto que todos os componentes supracitados são tipados. O ambiente já fornece uma hierarquia de tipos pré-definida, no entanto é possível criar novos tipos adequando o ambiente à cultura da organização. Esses tipos permitem tratar componentes de processo de forma não contextualizada, tornando possível, por exemplo, o uso de políticas automáticas que executam regras definidas pelo usuário de forma abstrata. Um exemplo disso é uma política de alocação que poderia ter o seguinte significado: “toda vez que uma atividade do tipo codificação estiver atrasada, avisar ao gerente por email para tomar providências”. Portanto, o uso de tipos permite raciocinar acerca de processos e seus componentes, aumentando o apoio automatizado fornecido pelo ambiente. Na Figura 2 é mostrado o formulário de cadastro de agentes e o formulário de cadastro de projetos.

No WebAPSEE Pro é possível definir alguns relacionamentos entre os componentes gerenciados pelo ambiente, tais como: agentes com habilidades (para cada habilidade deve-se indicar o grau dessa habilidade), agentes com papéis (um agente pode assumir inúmeros papéis), agentes com grupos (um agente pode participar de vários grupos), afinidades (indicando um grau de afinidade entre dois agentes), dependência entre recursos, composição e derivação de artefatos, sistemas com projetos (cada sistema pode ser desenvolvido por um ou mais projetos), sistemas e organizações (um sistema pode pertencer a uma organização). Além disso, todos os agentes, recursos materiais, ferramentas e artefatos podem ser alocados às atividades de um modelo de processo definido no ambiente.

O ambiente permite a **definição e implantação de processos** a partir de uma linguagem de modelagem gráfica de processos própria, a WebAPSEE-PML [Lima Reis 2003]. Tal linguagem é baseada em redes de atividades e é formalmente definida através de gramática de grafos, com objetivo de evitar inconsistências de modelagem e de execução. Com essa linguagem pode-se definir modelos de processos para um determinado projeto ou modelos de processos reutilizáveis (isto é, modelos que

não possuem características específicas de um projeto e podem ser instanciados para diferentes projetos). A Figura 3 mostra os elementos básicos da WebAPSEE-PML.

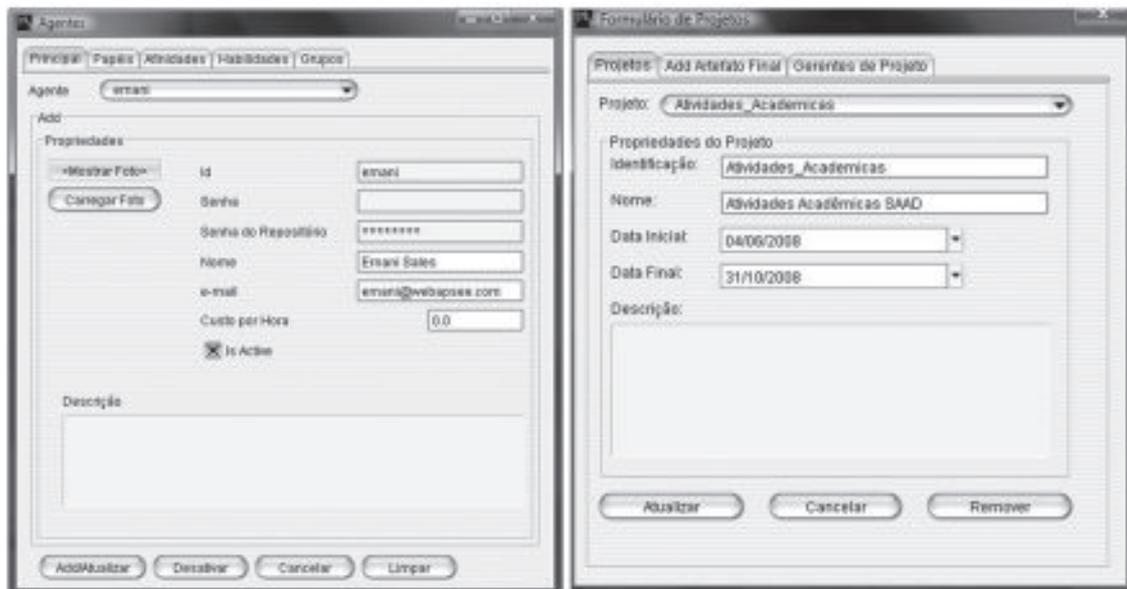


Figura 2. Formulários Organizacionais: Agentes e Projetos

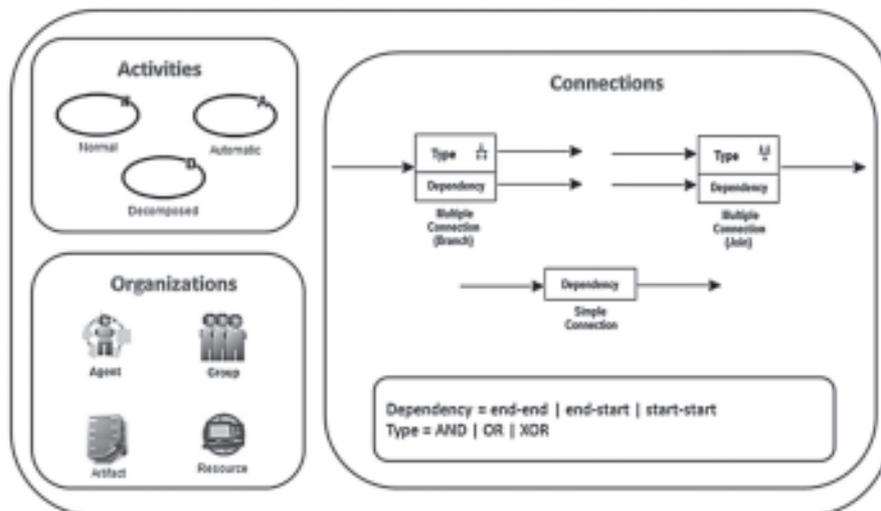


Figura 3. Símbolos da WebAPSEE-PML (Linguagem de modelagem de processos do WebAPSEE)

Após a definição e planejamento de um projeto, o ambiente permite o **controle e monitoração de projetos** a partir do uso de um editor gráfico (Figura 4) que exibe todas as atividades do processo e seus respectivos estados de execução (*Waiting, Ready, Active, Paused, Finished, Canceled, Failed*). O estado da atividade é proveniente da interação dos agentes (pessoas) com sua agenda de tarefas (*Task Agenda*). O acompanhamento dos projetos é realizado em tempo real e modificações dinâmicas em toda a estrutura do processo são permitidas e controladas.

Além das atividades, o editor de processos do WebAPSEE Pro permite visualizar os agentes (pessoas) que executam as atividades, os artefatos de entrada e saída de cada atividade, os recursos materiais ou de apoio alocados às atividades, grupos alocados em atividades, além das conexões de dependências (*end-start*, *start-start*, *end-end*) entre as atividades (indicando o fluxo de execução).

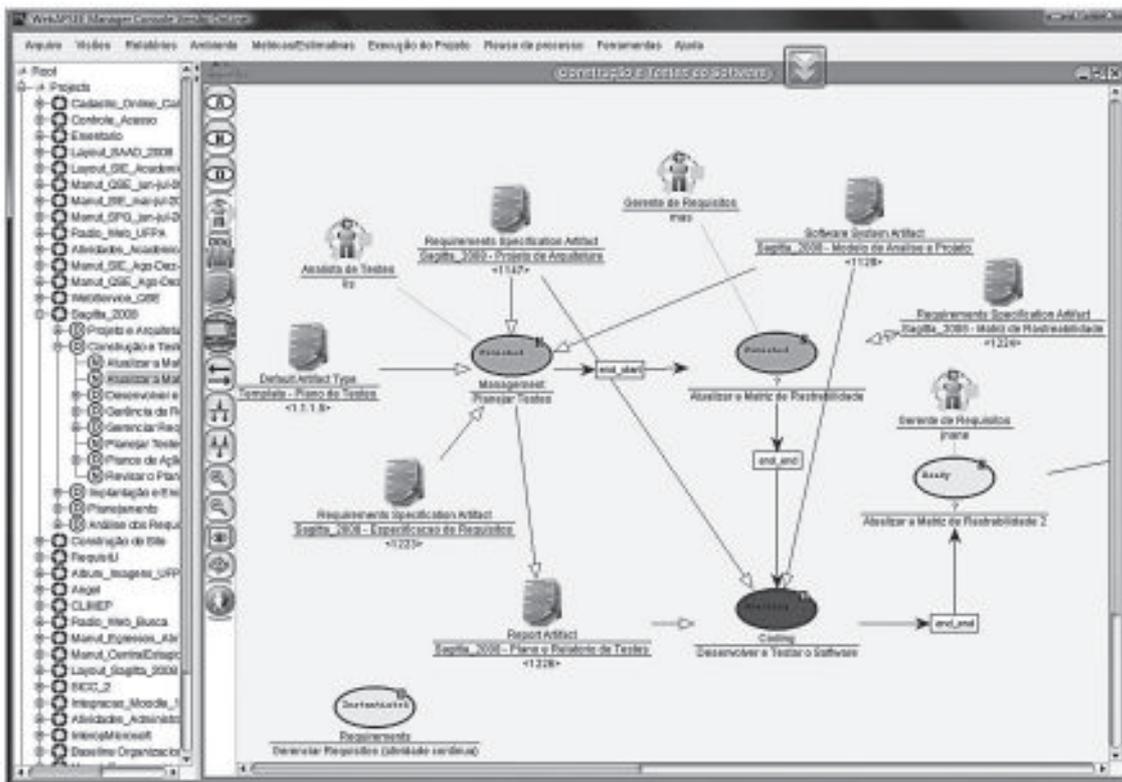


Figura 4. Editor de Modelagem e Execução de Processos

O WebAPSEE Pro implementa o controle de versões (*Check-in*, *Check-out* e *Log*) dos artefatos gerenciados pelo ambiente (**Gerência de Configuração**) a partir de uma integração com as ferramentas CVS e SubVersion. Dessa forma, é possível visualizar todas as versões de um dado artefato dentro do ambiente, bem como baixar qualquer uma das versões e criar uma nova versão de um artefato. Além disso, é implementado um mecanismo de controle de acesso sobre o controle de versões de artefatos, que permite aos agentes alocados em uma atividade baixar apenas os artefatos de entrada dessa atividade e gerar apenas versões de artefatos de saída dessa atividade e enquanto a mesma está em execução.

Outra macro-funcionalidade importante do ambiente é a **medição e análise do projeto**, que permite a definição de métricas e a coleta de estimativas e medidas associadas a alguns componentes gerenciados pelo ambiente (Agentes, Artefatos, Atividades, Grupos, Organizações, Projetos e Recursos). O ambiente já fornece algumas métricas pré-definidas, sendo que algumas dessas são coletadas automaticamente pelo ambiente. Por exemplo, a duração da execução de todas as atividades de um projeto é medida e armazenada para análises posteriores. O ambiente permite, ainda, a geração de relatórios a partir das informações de projetos e a partir das medidas/estimativas coletadas. Uma proposta mais completa de apoio integrado ao processo de medição está sendo integrada ao ambiente e foi baseada em [Nascimento, 2007].

Por fim, o WebAPSEE Pro permite a **gerência integrada de projeto** a partir da utilização de uma agenda de tarefas (*Task Agenda*) pelos envolvidos alocados em atividades de um projeto. O usuário da agenda pode iniciar, pausar, finalizar ou delegar uma atividade através da interação com a agenda de tarefas. Essa agenda permite também a visualização do roteiro para realização da atividade, da estimativa de prazo para sua execução, dos agentes envolvidos, dos recursos requeridos, dos artefatos de entrada e saída. Além disso, é possível ver o estado atual do processo (visão gráfica) e baixar/carregar os artefatos de entrada e saída, respectivamente.

4. Gerência de Requisitos

O WebAPSEE Pro possui um módulo para apoiar o processo de Gerência de Requisitos (GRE) denominado *WebAPSEE Requirement Manager (WARM)*. Esse módulo agrega as seguintes funcionalidades ao ambiente:

Gerenciar Requisitos: criação, recuperação, edição e remoção de requisitos; **Gerenciar Casos de Uso:** criação, recuperação, edição e remoção de casos de uso; **Gerenciar Rastreabilidade de Requisitos:** criação, edição e remoção de elos de rastreabilidade horizontal (rastros entre requisitos, Figura 5-A) e de rastreabilidade vertical (rastros entre requisitos e casos de uso, requisitos e artefatos, requisitos e atividades, e requisitos e agentes, Figura 5-B). Os rastros com indicativo verde são rastros validados pelo próprio usuário e rastros com indicativo em vermelho são rastros suspeitos (herdados de uma versão anterior do requisito e que precisam ser validados). Existe também um símbolo de rastro que pode ser visto na Figura 5-B que não contém a figura de usuário, trata-se de um rastro criado automaticamente pela ferramenta a partir das associações existentes no processo.

Gerenciar Mudanças de Requisitos: registro de mudança de um requisito e versionamento do requisito; **Gerenciar Baselines de Requisitos:** criação de baseline de requisitos e controle de versões sobre *baselines* (Figura 5-D); **Visualizar Árvore de Impacto:** visualização da árvore de impacto de um requisito (contemplando todos os seus rastros existentes) para análise do impacto de mudanças (Figura 5-C); **Visualizar e Editar Matriz de Rastreabilidade:** visualização e edição da matriz de rastreabilidade (sendo uma para cada par de componentes relacionados) e **Emitir Relatórios:** geração da Lista de Requisitos com os requisitos associados com um determinado sistema dentro do ambiente, e geração de um Relatório de Impacto de Mudança para um dado requisito.

Vale ressaltar que além das diversas funcionalidades presentes para apoio ao processo de Gerência de Requisitos, um dos grandes diferenciais do apoio oferecido pela WARM na GRE está no fato dela estar totalmente integrada ao ambiente WebAPSEE Pro. Dessa forma, possibilita-se a associação automatizada de requisitos com componentes presentes nos modelos de processos, tais como: atividades, artefatos (documentos) produzidos/consumidos e pessoas envolvidas.

5. Considerações Finais

Foi apresentado o ambiente WebAPSEE Pro e como suas funcionalidades contribuem para facilitar a implantação dos processos do MPS.BR com foco maior em Gerência de Projetos e Gerência de Requisitos. A ferramenta vem sendo utilizada com sucesso em implementações do nível G do MPS.BR

em organizações de desenvolvimento de software. Existem inúmeros trabalhos acadêmicos ligados ao ambiente WebAPSEE Pro e que visam fornecer apoio a outros processos do MR-MPS, tais como: Gerência de Configuração [Sales 2009], Medição [Nascimento 2007], Adaptação de Processos [Costa 2010], Gerência do Conhecimento [Oliveira 2010], dentre outros. Atualmente estudos estão sendo realizados visando a incorporação de funcionalidades para apoio a níveis mais altos de maturidade no MPS.BR.

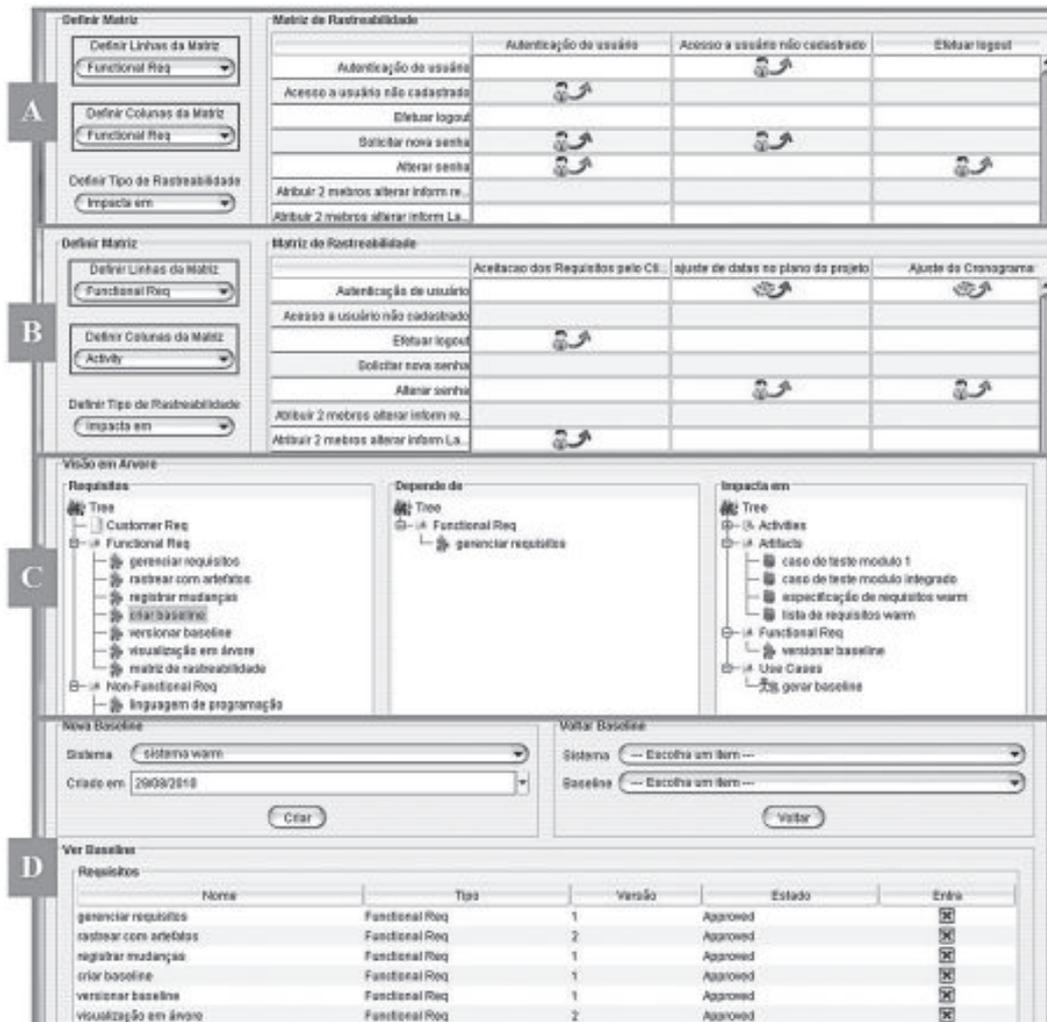


Figura 5. Rastreabilidade Horizontal, Rastreabilidade Vertical, Árvore de Impacto e Baseline de Requisitos

Referências

Costa, A. J.; Sales, E. O.; Lima Reis, C. A.; Reis, R. Q. (2007) Apoio a Reutilização de Processos de Software através de Templates e Versões. In Simpósio Brasileiro de Qualidade de Software. Trabalho Técnico. Porto de Galinhas, PE.

Costa, Anderson J. S. (2010) "Um Mecanismo de Adaptação de Processo de Software". Dissertação de Mestrado. Belém: PPGCC-UFGA. Disponível em www.labes.ufpa.br.

CVS. (2010) "Concurrent Versions System", Disponível em: <http://nongnu.org/cvs/>. Acesso em: março de 2010.

ABNT NBR/ISO/IEC 12207:2009 – Sistemas e Engenharia de Software – Processos de Ciclo de vida de Software (2009). ISBN: 978-85-07-01427-0.

ISO/IEC 15504 –1 Information Technology – Process Assessment, - Part 1: Concepts and Vocabulary (2003).

LIMA, Adailton M.; FRANÇA, Breno B.N.; COSTA, Anderson; REIS, C. A. L. . Gerência Flexível de Processos de Software com o Ambiente WebAPSEE. In: Simpósio Brasileiro de Engenharia de Software - Sessão de Ferramentas, 2006, Florianópolis. XIII Sessão de Ferramentas do SBES. Florianópolis : SBC, 2006.

Lima Reis, Carla A. (2003) “Uma Abordagem Flexível para Execução de Processos de Software Evolutivos”. Tese de Doutorado. Porto Alegre: PPGC da UFRGS.

MCT – Ministério de Ciência e Tecnologia. (2005) “Pesquisa sobre Qualidade e Produtividade no Setor de Software Brasileiro”. Disponível em: www.mct.gov.br

Nascimento, Luciana M. (2007) “Uma abordagem para Medição em um Ambiente de Desenvolvimento de Software Centrado em Processos”. Dissertação de Mestrado. Belém: PPGEE-UFPA.

Oliveira, Jadielly F. (2010) “Infraestrutura de Gerência do Conhecimento em um Ambiente de Desenvolvimento de Software Centrado em Processos”. Dissertação de Mestrado. Belém: PPGCC-UFPA.

Rocha, A. R.; Montoni, M.; Santos, G.; Oliveira, K.; Natali, A. C.; Mian, P.; Conte, T.; Mafra, S.; Barreto, A.; Albuquerque, A.; Figueiredo, S.; Soares, A.; Bianchi, F.; Cabral, R.; Net, A. D. (2006) “Success Factors and Difficulties in Software Process Deployment Experiences based on CMMI and MR MPS-BR”, In: 6th International Workshop on Learning Software Organizations (LSO’2006), Rio de Janeiro, Brasil.

Sales, Ernani O. (2009) “Gerência de Configuração integrada a Execução de Processos de Software”. Dissertação de Mestrado. Belém: PPGCC-UFPA.

SEI. (2010) “CMMI Site”, Disponível em: www.sei.cmu.edu/cmmi/.

Softex (2009) “MPS.BR – Melhoria de Processo do Software Brasileiro – Guia Geral”, ISBN 978-85-99334-15-7, Setembro.

Softex (2009) “Guia de Implementação – Parte 1: Fundamentação para Implementação do Nível G do MR-MPS”. ISBN 978-85-99334-16-4.

SVN. (2010) Subversion. Disponível em: <http://subversion.tigris.org>. Acesso em: março de 2010.

Wangenheim C. Gresse Von, Ita Richardson. (2007) “Why Are Small Software Organizations Different?” IEEE Software, vol. 24, no. 1, 2007, pp. 18-22.

WebAPSEE (2010). “WebAPSEE: Flexible Process Management”, Disponível em <http://www.webapsee.com>

Weber, K.C., Araújo, E., Machado, C.A.F., Scalet, D., Salviano, C.F., Rocha, A.R.C. (2005) “Modelo de Referência e Método de Avaliação para Melhoria de Processo de Software – versão (MR-MPS e MA-MPS)”. In: IV Simpósio Brasileiro de Qualidade de Software. Porto Alegre, Brasil.

Process Publisher: Um Gerador de Guias Eletrônicos de Processos de Software

Adailton Lima^{1,2}, Breno França¹, Marcelo Pereira¹, Carla Lima Reis^{1,2}, Rodrigo Reis^{1,2},

¹ Laboratório de Engenharia de Software (LABES)

Universidade Federal do Pará (UFPA) – <http://www.labes.ufpa.br>

Belém, Pará, Brasil

² QR Consultoria e Serviços - <http://www.qrconsult.com.br>

Belém, Pará, Brasil

{adailton,clima,quites,breno,marcelo.pereira}@webapsee.com

Abstract. *This paper presents the main characteristics of the tool named Process Publisher with respect to its functionalities related to process modeling and electronic process guide generation. We also discuss the main expected benefits of the use of this tool on the context of software process improvement.*

Resumo. *Este artigo apresenta as principais características da ferramenta Process Publisher, como a modelagem de processos e publicação do Guia Eletrônico de Processos. Também são discutidos os benefícios esperados com a utilização da mesma no contexto de melhoria de processos de software.*

1. Introdução

No contexto de tecnologias de processos de software, o nível de automação provido para apoio aos processos de software varia desde ferramentas especializadas, chamadas PSEEs (*Process-centered Software Engineering Environment*) [Grhun 2002], à guias de processos publicados em uma intranet. Apesar da maior automação no acesso e manipulação de informações dos processos provida pelos PSEEs, existe a necessidade de conhecimento especializado (com custos de treinamento e licenças de uso) para todos os interessados no acesso às informações. Já os chamados guias de processos apresentam as informações dos processos na forma de conteúdo interativo, o que pode facilitar o acesso pelos interessados sem a necessidade de treinamentos especializados e licenças de acesso.

Para a institucionalização dos processos organizacionais é necessário um mecanismo que permita a ampla divulgação de informações sobre os processos de uma organização. Neste contexto, a utilização de guias de processos representa um apoio para divulgação das informações dos processos entre todos os interessados e sua institucionalização.

Os guias de processos podem ser usados como mecanismos para a gradual adoção de ferramentas mais avançadas para gestão automatizada de processos de software. Assim, a partir do aumento da maturidade e capacidade de gestão dos processos de software, a organização pode adotar soluções integradas de simulação, execução e reutilização de processos, evoluindo os modelos descritos inicialmente com os guias eletrônicos. No entanto, quando não há uma política de atualização constante do guia de processos, refletindo as mudanças naturais dos processos em execução, ele tende a ficar obsoleto.

A ferramenta **Process Publisher** fornece um ambiente para modelagem gráfica de processos e um gerador de guias eletrônicos, baseado nos processos modelados neste ambiente. Esta ferramenta visa atender tanto a necessidade de publicação da documentação eletrônica sobre processos de software de uma organização, quanto a visualização do estado de um projeto em execução. Sua utilidade é percebida também no início de uma política para melhoria de processos de software, pois provê um meio simples e amplo para divulgação de informações sobre seus processos organizacionais a todos os interessados. Além disso, a partir da sua integração com PSEEs, as informações sobre execuções anteriores de processos podem ser recuperadas na seleção de processos em situações futuras.

Este artigo apresenta as principais características da ferramenta **Process Publisher**, além dos benefícios esperados com a utilização da mesma no contexto de melhoria de processos de software. O artigo está estruturado como a seguir: na seção 2 são apresentados os principais objetivos da ferramenta; na seção 3 são descritas as duas principais funcionalidades da ferramenta; na seção 4 são apresentados os principais requisitos de usabilidade que influenciaram a implementação do Guia Eletrônico de Processos gerado pela ferramenta; na seção 5 são apresentados exemplos de funcionalidades presentes no Guia Eletrônico de Processos gerado; na seção 6 são discutidos os principais benefícios relacionados com o modelo MPS.BR; por fim, na seção 7 são apresentadas as considerações finais deste artigo.

2. Objetivos da Ferramenta

O objetivo geral da ferramenta **Process Publisher** é apoiar a modelagem, recuperação e distribuição de processos de software de uma organização. Além disso, esta ferramenta possui os seguintes objetivos adicionais:

- ser minimamente invasiva, isto é, procura não implicar em dramáticas alterações nas ferramentas e técnicas utilizadas pela organização-cliente;
- permitir a integração com ferramentas externas tradicionalmente usadas no desenvolvimento de software, através da utilização do padrão XML para intercâmbio de dados;
- possuir independência em relação ao modelo de processo a ser descrito, ou seja, permitir a modelagem de qualquer modelo de processos existentes no mercado;

A ferramenta **Process Publisher** foi desenvolvida para explorar a integração com PSEEs, facilitando a geração de conteúdo a partir dos processos que estiverem em execução. Esta característica permite uma visualização mais adequada da situação dos projetos em andamento e facilita a coleta de evidências de implementação de melhoria de processos para o caso de uma avaliação oficial (explicado com mais detalhes na seção 6). Atualmente a ferramenta está integrada ao ambiente WebAPSEE [Lima 2006], a partir do qual é gerado conteúdo acerca dos processos em execução.

3. Funcionalidades Gerais

A seguir são descritas as duas principais funcionalidades gerais providas pela ferramenta **Process Publisher**.

3.1 Modelagem de Processos

Considerando, em primeiro lugar, a flexibilidade para que haja independência ao modelo de processo a ser descrito, um Guia de Processos deve permitir que o gerente possa definir modelos de processos específicos, voltados às necessidades e ao contexto enfrentado na ocasião pela sua organização.

Na ferramenta **Process Publisher** o gerente de processos pode definir informações relacionadas à organização (como por exemplo cargos, pessoas e habilidades) e às atividades dos processos (como por exemplo tarefas, prazos, artefatos envolvidos e fluxo de atividades).

Atualmente, o estado da arte tecnológico na área é representado por linguagens de modelagem de processos visuais, que permitem a descrição dos elementos do processo através de ícones e arcos gráficos. Na versão atual da ferramenta **Process Publisher** é disponibilizado um editor gráfico de processos baseado na linguagem de definição de processos do ambiente WebAPSEE [Lima 2006].

3.2 Publicação de Processos

A funcionalidade básica fornecida por qualquer Guia de Processos consiste na geração de páginas Web estáticas ou dinâmicas que permitam a navegação no modelo.

A publicação de processos através da ferramenta **Process Publisher** consiste na exportação das informações cadastradas a respeito do modelo de processo em um formato Web. Neste caso, o primeiro passo realizado é a geração de um documento XML que contém todas as informações relacionadas ao processo escolhido para exportação. De posse do conteúdo XML do processo, é realizada a geração do conteúdo estático e dinâmico que integrará o Guia Eletrônico de Processos. Por fim, o conteúdo gerado deve ser disponibilizado em um servidor Web acessível por todos os interessados no conteúdo do processo.

4. Requisitos de Usabilidade do Guia Eletrônico de Processos

A ferramenta **Process Publisher** possui alguns requisitos de usabilidade que influenciaram fortemente a escolha da tecnologia a ser adotada. Tais requisitos são listados na Tabela 1 a seguir:

REQ1	O guia de processo gerado deve ter alta disponibilidade dentro da organização. O conteúdo deve ser facilmente acessível e mudanças devem ser facilmente propagadas;
REQ2	O guia de processo gerado deve ser acessado através de ligações através das quais o usuário possa navegar facilmente;
REQ3	O guia de processo gerado deve permitir o acesso rápido a um conteúdo específico sem que seja necessário percorrer um longo caminho;
REQ4	O modelo de processos deve ser visualizado graficamente. Cada componente gráfico do processo deve ter uma ligação com seu respectivo conteúdo.

Tabela 1. Principais Requisitos de Usabilidade

O requisito **REQ1** diz respeito à disponibilidade do guia eletrônico de processos. Através da adoção de tecnologias Web e redes de comunicação de computadores, é possível que seja facilmente permitido acesso ao conteúdo do guia eletrônico de processos a todos os interessados dentro da organização. Ainda como consequência da adoção de tecnologia Web e da centralização do conteúdo em um servidor de páginas, é possível propagar alterações no processo alterando-se apenas o conteúdo no servidor e tornar esta informação disponível automaticamente a todos os interessados.

O requisito **REQ 2** diz respeito à facilidade de navegação no conteúdo. Uma tecnologia bastante difundida, baseada no modelo Web, são as páginas HTML. Páginas HTML são como páginas comuns quem agregam em si conteúdo multimídia e possuem ligações com outras páginas relacionadas. A utilização de páginas HTML foi adotada na construção da ferramenta de maneira a atingir o requisito de navegação do conteúdo.

O requisito **REQ3** é atendido através da construção de uma árvore onde os itens do processo são listados e categorizados por classe de item. O usuário pode visualizar um determinado conteúdo através de uma ligação entre o item na árvore e sua respectiva página HTML.

Por fim, o requisito **REQ4** é atendido através da construção de um visualizador gráfico de processos que “desenha” o modelo do processo na página HTML e monitora qualquer interação do usuário com seus componentes gráficos. Uma interação do usuário com algum componente gráfico do modelo de processos pode levá-lo a uma página HTML com o conteúdo relativo ou pode levá-lo a outro modelo de processos.

Dentre os requisitos supracitados é importante destacar os requisitos **REQ3** e **REQ4**, que exigem um alto nível de interatividade no conteúdo gerado no Guia Eletrônico de Processos. Este nível de interatividade não é atendido de maneira natural pelo modelo da Web ou pelas páginas HTML, que originalmente foram concebidas para apresentação de conteúdo estático. Daí, se fez necessária a integração com outras tecnologias emergentes que complementam tais limitações. Na implementação atual da ferramenta a tecnologia Adobe Flex [Flex 2010] é utilizada para criação de desenhos gráficos e árvores de navegação de conteúdo nas páginas HTML geradas.

5. Características do Guia Eletrônico de Processos

A seguir são descritas as principais características do Guia Eletrônico de Processos gerado pela ferramenta.

5.1 Navegação interativa

Foi adotada uma categorização das informações do processo para atender aos requisitos de usabilidade já citados. Desta forma, a interação do usuário com o Guia Eletrônico de Processos é realizada a partir da navegação entre estas categorias. Conforme destacado como item 1 na Figura 1, as categorias definidas são: tipos, pessoas, artefatos, medidas, recursos e atividades.

A partir da seleção da categoria, é possível interagir com os itens desta através de uma estrutura de árvore, como destacado no item 2 na Figura 1. Para cada item listado na árvore existe uma página correspondente, contendo detalhes e ligações para outras páginas relacionadas (como o item 3 destacado na Figura 1).



Figura 1. Visualização textual de atividades

5.2 Descrição do processo

O processo alvo do Guia Eletrônico de Processos gerado é descrito através de formulários, tabelas e listagem dos atributos dos componentes que foram modelados.

Como exemplo, a Figura 2 apresenta um cargo selecionado na árvore interativa, e as suas informações representadas na página associada. Nesta página existem informações que detalham a hierarquia e descrição do cargo, além de apresentar uma listagem de todas as atividades onde este cargo atua no processo.

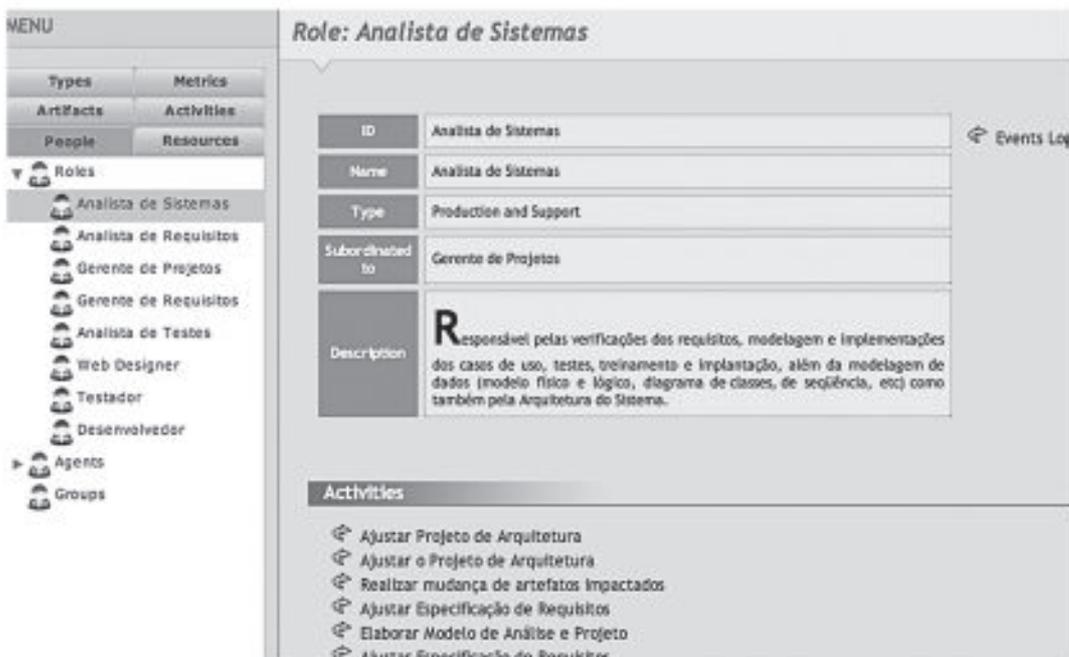


Figura 2. Visualização dos cargos do processo

5.3 Notação gráfica do processo

A partir da notação gráfica disponível no Guia Eletrônico de Processos é possível visualizar o fluxo existente entre as atividades do processo e o contexto geral no qual a atividade está inserida.

Como ilustrado na Figura 3, a notação gráfica segue a linguagem adotada na ferramenta WebAPSEE [Lima 2006] e possui informações dos artefatos, recursos, cargos, pessoas e dependência entre as atividades. A representação gráfica também permite a interatividade do usuário, que pode clicar nas atividades e visualizar submodelos de processos quando estas atividades forem decompostas em subatividades.

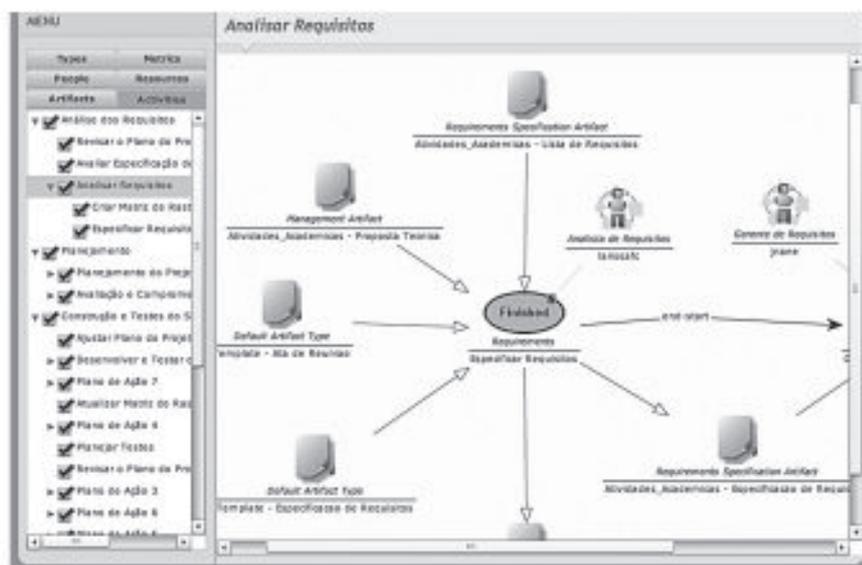


Figura 3. Visualização gráfica do Modelo de Processos

5.4 Dados da execução do processo

A partir da integração atual com o ambiente WebAPSEE [Lima 2006] é possível obter informações relacionadas a um processo em execução ou até mesmo finalizado. Neste caso, detalhes da execução do processo fazem parte do modelo disponibilizado no Guia Eletrônico de Processos.

Como exemplo, a Figura 4 apresenta a listagem do *log* de eventos ocorridos no processo. Estes eventos estão ordenados pela data de ocorrência, e servem como um histórico das interações das pessoas envolvidas com a realização das atividades definidas no processo. Além do *log* de eventos, são disponibilizadas informações de pessoas envolvidas nas atividades, além do histórico e conteúdo de todas as versões dos artefatos – extraídos de repositórios SubVersion ou CVS - que foram utilizados e produzidos nas atividades do processo.

The screenshot shows a software interface titled "Event Log". At the top, there are several tabs: "Agenda", "Connection", "Global Activity", "Modeling Activity", "Process", "Process Model", and "Resource". Below the tabs is a table with the following columns: "Occurrence Date", "What", "Event Dispatcher (WHO)", "Why", and "Activity". The table contains five rows of event data.

Occurrence Date	What	Event Dispatcher (WHO)	Why	Activity
2008-10-10 11:08:37.0	ToActive	Usuario Exemplo 10	Rule 3.4	Avaliar a Mudança de Requisitos e OBTEN COMPROMETIMENTO
2008-10-10 09:04:34.0	ToActive	Usuario Exemplo 14	Rule 3.4	Codificar Requisitos Orientação Graduação (RF01 e RF04)
2008-10-09 19:02:38.0	ToPaused	Usuario Exemplo 10	Rule 3.8	Avaliar a Mudança de Requisitos e OBTEN COMPROMETIMENTO
2008-09-16 07:30:16.0	ToActive	Usuario Exemplo 10	Rule 3.4	Ajustar Plano do Projeto e OBTEN COMPROMETIMENTO
2008-09-15 18:10:07.0	ToPaused	Usuario Exemplo 10	Rule 3.8	Ajustar Plano do Projeto e OBTEN COMPROMETIMENTO

Figura 4. Visualização do log de eventos do processo

6. Benefícios Esperados

A seguir são descritos os dois principais benefícios esperados com a adoção da abordagem baseada na publicação de Guias Eletrônicos de Processos da ferramenta **Process Publisher**.

6.1 Apoio à Institucionalização de Processos

Além da definição e melhoria contínua do processo, para sua efetiva institucionalização é necessário que haja um amplo acesso e conhecimento por parte dos interessados e envolvidos. Neste caso, os resultados de uma ação para institucionalização dos processos de desenvolvimento de software pode ser afetada diretamente pelos meios que serão utilizados para que a informação seja acessada e absorvida pelos interessados.

No caso da ferramenta **Process Publisher**, o Guia Eletrônico de Processos gerado pode servir como uma referência rápida e constantemente atualizada se disponibilizado a todos os interessados. Desta forma, pode colaborar positivamente para a institucionalização dos processos de desenvolvimento de software de uma organização.

6.2 Apoio ao Processo de Avaliação do MPS.BR

No guia de avaliação do MPS.BR [Softex 2010], a atividade "Preparar a avaliação" possui o objetivo de preencher a planilha com as evidências que comprovem a implementação dos processos e que será utilizada na avaliação. Neste momento, normalmente, a planilha deve ter *links* para os documentos dos projetos que serão alvo da avaliação.

A partir da utilização da ferramenta **Process Publisher** e também da sua integração com o ambiente WebAPSEE [Lima 2006], é possível prover as informações necessárias para preenchimento desta planilha de maneira simplificada. Para tanto, basta que o Guia Eletrônico de Processos seja gerado com as

informações de execução do processo, juntamente com as versões dos artefatos utilizados nas atividades do processo. Assim, o avaliador responsável pelo preenchimento da planilha com as evidências pode navegar nas páginas e obter acesso aos dados almejados de maneira direta.

Para preenchimento da planilha de avaliação, tanto os indicadores diretos quanto os indicadores indiretos podem ser obtidos, desde que tenham sido disponibilizados no processo.

7. Considerações Finais

A ferramenta **Process Publisher** foi desenvolvida e é comercializada pela empresa **QR Consultoria e Serviços** em parceria com a UFPA a partir de projeto financiado pela Fundação de Amparo a Pesquisa do Pará (FAPESPA) e da FINEP.

No contexto de melhoria de processos de software esta ferramenta é útil principalmente para apoiar a divulgação das informações sobre os processos organizacionais. A ferramenta diferencia-se dos guias eletrônicos de processos existentes porque mostra uma “fotografia” de um projeto em execução, devido a sua integração com um ambiente de execução de processos, contendo todas as informações da organização e do projeto incluindo artefatos que possam ser publicados. Como benefício adicional, derivado da sua integração com o PSEE WebAPSEE [Lima 2006], a ferramenta **Process Publisher** pode servir como apoio à identificação e registro das evidências necessárias para avaliação da aderência dos processos da organização a algum modelo de maturidade, como, por exemplo o MPS.BR.

Esta ferramenta está em contínua evolução, e como trabalhos futuros considera-se a integração com outras ferramentas de processos para obtenção automática das informações nela já presentes através da utilização do padrão XML para intercâmbio de dados.

Referências

Flex. Página oficial da tecnologia Adobe Flex. Disponível em <http://www.adobe.com/flex>. Acesso em agosto de 2010.

Softex. Guia de Avaliação do MPS.BR. Disponível em http://www.softex.br/mpsbr/_guias/guias/MPSBR_Guia_de_Avaliacao_2009.pdf. Acesso em agosto de 2010.

Gruhn, Volker (2002) “Process-Centered Software Engineering Environments: A Brief History and Future Challenges”. In: *Annals of Software Engineering* 14, 363–382. Kluwer Academic Publishers.

Lima, Adailton M.; Costa, Anderson; França, Breno B. N.; Reis, Carla A. L.; Reis, Rodrigo Q. **Gerência Flexível de Processos de Software com o Ambiente WebAPSEE**. 19º. Simpósio Brasileiro de Engenharia de Software – Sessão de Ferramentas, Outubro, 2006.

WKM: Uma Ferramenta para Auxiliar a Gerência de Conhecimento Integrada a um ADS Centrado em Processos

Jadielly Oliveira^{1,2}, Liken Lima^{1,2}, Silvia Nunes¹ das Dores, Ernani Sales^{1,2}, Gabriela Andrade¹, Carla Lima Reis^{1,2}

¹Laboratório de Engenharia de Software – Universidade Federal do Pará (UFPA)

Caixa Postal 479 – 66.075 -110 – Belém – PA – Brasil

²QR Consultoria e Serviços Ltda. – Rua Santo Antônio, 316

66.010-090 – Belém – PA – Brasil

{jadielly,liken,ernani,gabriela}@webapsee.com,silvia.cnd@gmail.com, clima@ufpa.br

Abstract. *This paper presents the WebAPSEE Knowledge Manager (WKM), an extension of the Process centered Software Engineering Environment WebAPSEE, which aims support for planning and execution of Knowledge Management strategies. The tool was developed from the collection of requirements arising from software quality models, literature and a study with five software development organizations realized in Belém of Pará.*

Resumo. *Este artigo apresenta a ferramenta WebAPSEE Knowledge Manager (WKM), uma extensão do Ambiente de Engenharia de Software Centrado em Processos WebAPSEE, que tem como objetivo auxiliar no planejamento e execução de estratégias de Gerência do Conhecimento. A ferramenta foi elaborada a partir da coleta de requisitos advindos de modelos de qualidade de software, pesquisa bibliográfica e da realização de um estudo com cinco organizações de desenvolvimento software realizado em Belém do Pará.*

1. Introdução

O êxito de um projeto de software depende muito da tomada de decisões e das soluções aplicadas para os problemas que ocorrem durante o desenvolvimento. Essas decisões normalmente são feitas com base no conhecimento e experiência que os engenheiros de software obtiveram durante a realização de suas atividades passadas ou mesmo nas atividades correntes [Fischer e Ostwald, 2001] ou através de treinamentos [Rus e Lindvall, 2002]. No entanto, muitas vezes esse conhecimento é retido na mente do indivíduo que solucionou o problema e em nenhum momento documentado. Com isso, as organizações de software passam a depender, de forma crítica, de seus profissionais. Diante deste contexto, saber preservar e utilizar o conhecimento organizacional com eficiência tornou-se um fator decisivo para que as organizações possam atingir seus objetivos com qualidade.

Gerenciar o conhecimento é um processo que auxilia organizações a encontrar, selecionar, organizar, disseminar e transferir as informações importantes e as experiências necessárias para as várias atividades desenvolvidas, como, resolução de problemas, aprendizagem dinâmica, planejamento estratégico e tomada de decisões [Probst, Raub e Romhardt, 1999].

A área de Gerência do Conhecimento (GC) tem se consolidado cada vez mais no mercado de desenvolvimento de software, prova disso foi a atualização de modelos de maturidade de software como o proposto pelo Programa MPS.BR - Melhoria do Processo de Software Brasileiro [Softex, 2009], People-CMM [Curtis, Hefley e Miller, 2009] e normas como a ISO/IEC 12207 [ISO/IEC, 2008] com recomendações e práticas que evidenciam a GC como atividade obrigatória para que uma organização atinja maior maturidade e capacidade de desenvolvimento de software.

Visando auxiliar organizações de desenvolvimento de software na implantação de estratégias de gestão do conhecimento de forma aderente a normas e modelos de qualidade, este artigo apresenta a ferramenta *WebAPSEE Knowledge Manager* (WKM), que é uma ferramenta de apoio à GC integrada ao Ambiente de Engenharia de Software Centrado em Processos (PSEE) *WebAPSEE* [Lima Reis e Reis, 2007]. O WebAPSEE é um PSEE que permite a modelagem e execução de processos, acompanhamento dos prazos das atividades, alocação de recursos, além de promover a reutilização de boas práticas gerenciais em diferentes projetos e a auxiliar na coordenação de atividades de equipes geograficamente dispersas. A ferramenta WKM foi desenvolvida a partir da evolução dos trabalhos [Oliveira *et al.*, 2009] e [Oliveira e Reis, 2009].

O restante do texto está estruturado da seguinte forma: a seção 2 apresenta os requisitos da ferramenta e um breve mapeamento entre os requisitos e a fonte que originou o requisito; a seção 3 descreve a WKM através de uma visão geral da arquitetura e das suas funcionalidades; a seção 4 apresenta as considerações finais.

2. Requisitos da Ferramenta

A elaboração da ferramenta teve como base um conjunto de requisitos para implantação de GC que foram coletados de várias fontes, a fim de proporcionar completude às funcionalidades de apoio a GC no ambiente WebAPSEE.

Os requisitos foram obtidos de quatro fontes: (i) Bibliografia acerca das necessidades para implantação de Gerência do Conhecimento, dentre as quais se podem citar [Rus e Lindvall, 2002], [Probst *et al.*, 1999], [O'Leary, 1998], [Fischer e Ostwald, 2001]; (ii) Atividades descritas na NBR ISO/IEC 12207 [ISO/IEC, 2008] e no Guia de Implementação do nível E do modelo MPS [SOFTEX, 2009]; (iii) Aspectos sócio-culturais tratados no P-CMM [Curtis, Hefley e Miller, 2009]; e (iv) Estudo qualitativo realizado com cinco empresas de Belém do Pará [Oliveira, 2010].

Foram definidos no total quinze requisitos, dentre os quais alguns são diretamente atendidos pela ferramenta, enquanto que outros são atendidos indiretamente, pois a ferramenta apenas fornece o insumo para que o mesmo possa ser realizado, como, por exemplo, o REQ 15. Os requisitos são apresentados na Tabela 1.

Identificador	Descrição do Requisito
REQ 1	Promover ações de conscientização sobre a importância da realização da gerência de conhecimento na organização, tanto com a alta direção quanto com os funcionários.
REQ 2	Seleção de uma estratégia adequada às características da organização.
REQ 3	Estabelecer políticas para criação, compartilhamento e utilização de ativos de conhecimento por toda a organização.
REQ 4	Definir papéis e alocar pessoas responsáveis para trabalhar na execução do processo de Gerência do Conhecimento.
REQ 5	Permitir o planejamento das ações relacionadas à gerência do conhecimento, de forma que as atividades detalhadas no plano sejam realizadas pela organização.
REQ 6	Definir os tipos de conhecimento estratégicos para a organização.
REQ 7	Permitir a aquisição de diversos tipos de conhecimento de membros da organização, tanto relacionados a projetos específicos como em nível organizacional estabelecendo padrões e classificações de tipos de conhecimento a fim de facilitar a recuperação e disseminação dos ativos.
REQ 8	Permitir a avaliação de conhecimento antes de armazená-lo e disponibilizá-lo para a organização, com o objetivo de filtrar conhecimento de valor, garantindo que o mesmo esteja correto e claro o suficiente para ser reutilizado.
REQ 9	Garantir que o conhecimento está disponível para reutilização por outros membros da organização.
REQ 10	Permitir a busca de conhecimento pelos membros da organização.
REQ 11	Formar uma rede de especialistas na organização, permitindo a busca de pessoas (especialistas) na organização, de forma que se tenham estabelecidas as habilidades de cada membro da organização e uma quantificação de experiência do indivíduo.
REQ 12	Desenvolver soluções para disseminação do conhecimento aos funcionários da organização.
REQ 13	Permitir que os usuários possam avaliar a utilidade do conhecimento, bem como a própria estratégia adotada na organização, para que seja possível obter resultados em relação aos benefícios obtidos com a implantação da mesma.
REQ 14	Permitir a manutenção de conhecimento, de forma que conhecimento defasado possa ser atualizado ou desabilitado da organização.
REQ 15	Promover formas de compensar os funcionários pela criação de conhecimento de valor à organização.

Tabela 1. Requisitos da Ferramenta

Cada requisito foi encontrado em uma fonte principal, e alguns se repetem em várias referências. A Tabela 2 apresenta um mapeamento entre os requisitos e suas respectivas fontes de pesquisa.

Esse mapeamento representa a aderência da ferramenta aos modelos de qualidade, bem como às melhores práticas descritas na literatura referente ao assunto. É importante ressaltar também que, dependendo da estratégia adotada na organização, alguns desses requisitos podem ser desconsiderados ou adaptados mediante a necessidade da mesma.

Requisitos	Fonte	MPS.BR	CMM-People	ISO-IEC 12207	Estudo Qualitativo	Pesquisa Bibliográfica
REQ 01						X
REQ 02					X	
REQ 03			X			
REQ 04	X	X	X			X
REQ 05	X	X	X	X		
REQ 06				X	X	X
REQ 07	X			X		X
REQ 08	X					X
REQ 09	X			X		
REQ 10	X			X		
REQ 11	X			X	X	X
REQ 12	X			X	X	
REQ 13	X	X	X			X
REQ 14	X			X		X
REQ 15			X		X	X

Tabela 2. Mapeamento entre os requisitos e suas fontes de pesquisa

3. WebAPSEE Knowledge Manager

A ferramenta WKM foi desenvolvida de forma integrada ao *WebAPSEE*, permitindo assim, que a WKM utilize as funcionalidades providas pelo *WebAPSEE*, como por exemplo, a manutenção de informações organizacionais e a máquina de execução de processos.

A seguir são apresentadas uma breve descrição da arquitetura da WKM e uma visão geral de algumas funcionalidades.

3.1. Arquitetura

A arquitetura da WKM é baseada na arquitetura do *WebAPSEE*, cliente-servidor, e cada conjunto de funcionalidades possui um componente específico para implementar e disponibilizar as funcionalidades. A Figura 1 ilustra um visão geral da arquitetura do ambiente *WebAPSEE* e da WKM.

A WKM é integrada aos clientes *Manager Console* e a *Task Agenda Web*, disponibilizando as

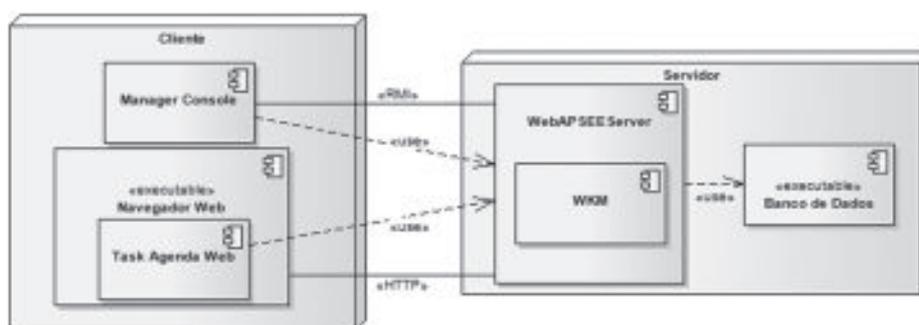


Figura 1. Visão Geral da Arquitetura

funcionalidades de GC, implementadas através do componente WKM no *WebAPSEE Server*, associado aos componentes existentes do *WebAPSEE*, como a máquina de execução de processos e a gerência organizacional.

O *WebAPSEE Server*, implementado utilizando Java, disponibiliza interfaces de invocação de procedimento remoto RMI, e executado no contexto do container de aplicação Apache Tomcat, permitindo assim que o *Manager Console*, desenvolvido em Java e interface gráfica *Swing*, e a *Task Agenda Web*, desenvolvido em Adobe Flex e Java para fornecer a interface gráfica Web, se comuniquem com o servidor do WebAPSEE, via RMI e HTTP respectivamente.

3.2. Utilização da Ferramenta

As principais funcionalidades da WKM são: (a) definição/manutenção da estrutura organizacional (cadastro de organização, unidades organizacionais, criação de redes de especialistas), (b) definição/manutenção do Plano de GC (definição de planos para coleta e disseminação de conhecimento, definição das estruturas de conhecimento a serem coletadas, dentre outros.), (c) gerência dos itens de conhecimento (aquisição, busca, avaliação, homologação, disseminação e manutenção do conhecimento) e (d) geração de relatórios de acompanhamento relacionados à GC.

A partir da interface do *Manager Console* é possível definir a estrutura organizacional, através do cadastro de organizações (Figura 2.A) e das unidades organizacionais que a compõem (Figura 2.B). Estes cadastros permitem a gerência dos dados que caracterizam a organização como razão social, endereço, sigla, além de permitirem a configuração de um organograma da mesma, que é exibido na *Task Agenda Web* (Figura 2.C).

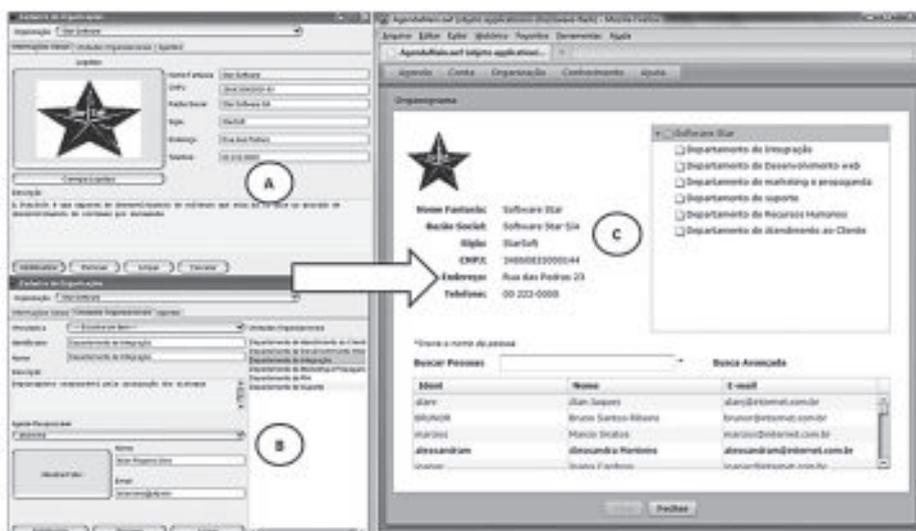


Figura 2. Definição da Estrutura Organizacional

Além das informações sobre a organização, a WKM permite a gerência dos dados dos membros da organização, seus papéis, afinidades, habilidades, grupos dos quais faz parte, o que possibilita a

identificação de especialistas em determinadas áreas e as afinidades desses especialistas dentro do grupo de trabalho, estes dados podem também serem visualizados através da *Task Agenda Web*, como mostra a Figura 3.

Outra importante funcionalidade que o *Manager Console* permite ao gerente é definir e manter Planos de GC nos quais são estabelecidas as estruturas de conhecimento (por exemplo, lições aprendidas, padrão arquitetura etc.) que devem ser coletadas e disseminadas em um projeto. Durante a definição dos Planos de GC também é possível definir marcos no processo para a realização de coleta de conhecimento indicando qual o tipo de conhecimento deve ser coletado em determinado momento do processo (Figura 4.A). Um marco para coleta é definido através da seleção de um evento associado a uma atividade (início, término ou delegação da atividade), o que é possível devido à integração com a máquina de execução do *WebAPSEE*. As Figuras 4.B e 4.C apresentam a visão do usuário da agenda ao receber uma notificação (sugestão) para cadastrar um novo item de conhecimento quando um marco foi alcançado.

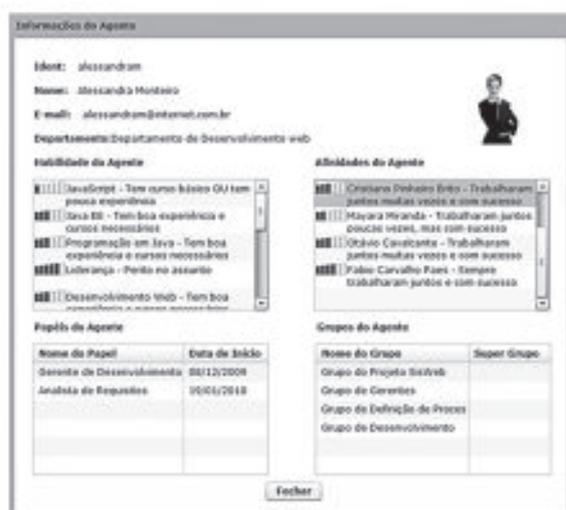


Figura 3. Informações sobre os membros da organização

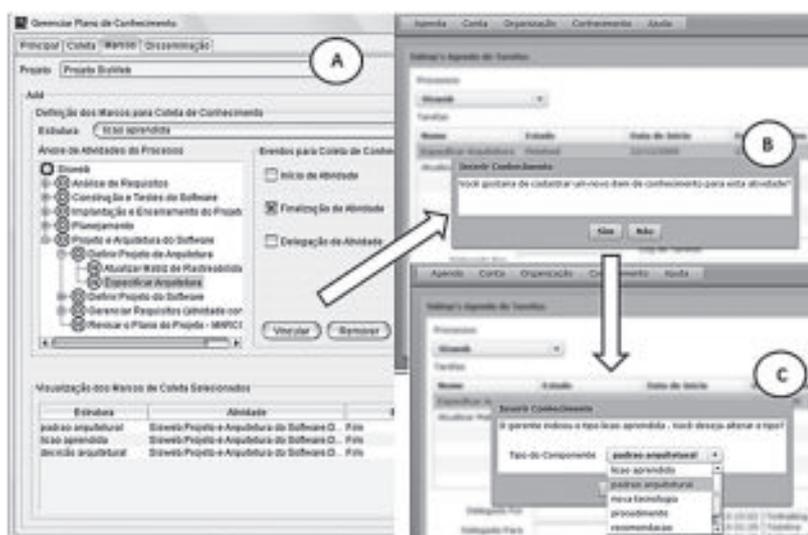


Figura 4. Coleta de Conhecimento nos Marcos definidos no plano de GC

Alternativamente à coleta de requisitos a partir da definição de marcos, um usuário da *Task Agenda Web* pode inserir diretamente um item de conhecimento, que após homologado estará disponível para consulta de outros usuários.

Ao ser inserido, o item de conhecimento recebe o estado de salvo (estado “rascunho”), permitindo que este seja enviado para homologação (estado “em homologação”). O homologador, ao receber o conhecimento pode: disponibilizar o conhecimento (estado “disponível”), mandar para avaliação (estado “em avaliação”), enviar para o autor realizar uma revisão (estado “revisão”) ou desabilitar o conhecimento (estado “desabilitado”), caso em que o item se torna inútil. A visão do homologador é apresentada na Figura 5. Quando o item de conhecimento está “disponível” é possível buscá-lo na base do sistema a partir de uma série de filtros, como tipo de componente do WebAPSEE (atividades, ferramentas, processos), por data, autor ou palavra-chave. Além disso, o sistema passa a disseminar os itens de conhecimento relacionados a atividades dos processos a partir de uma comparação entre atividades semelhantes em processos diferentes.

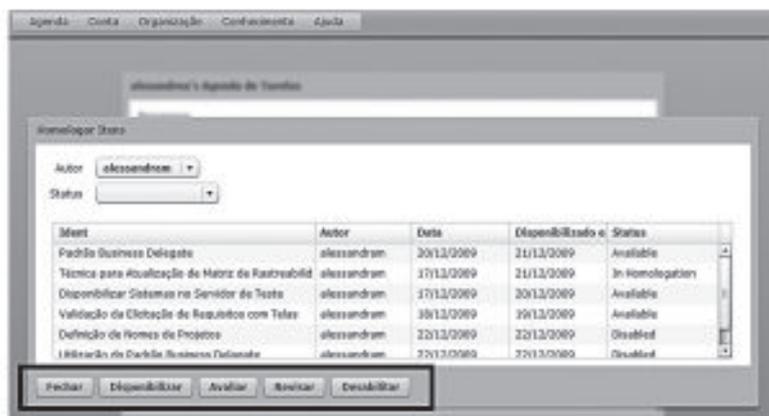


Figura 5. Homologação dos Itens de Conhecimento

Através da interface do *Manager Console*, o gerente pode mandar gerar relatórios dos itens de conhecimento cadastrados, indicando o número de itens cadastrados por um agente, o número de itens existentes por estrutura de conhecimento e o número de avaliações positivas por item de conhecimento.

4. Considerações Finais

Este artigo apresentou a ferramenta *WebAPSEE Knowledge Manager*, ferramenta desenvolvida com o objetivo de apoiar a o processo de Gerência de Conhecimento. A WKM utiliza algumas funcionalidades do ambiente WebAPSEE, como: (i) a máquina de execução de processos para definir marcos no processo para coleta de conhecimento; e (ii) a consulta dos dados organizacionais para auxiliar na escolha de especialistas para avaliar os conhecimentos cadastrados.

Alguns sistemas [Montoni, 2003] [Natali, 2003] [Holz, 2003] foram desenvolvidos para apoiar as atividades de Gerência de Conhecimento (GC), sobretudo, integrados a Ambientes de Desenvolvimento

de software (ADS). Entretanto, tais ferramentas não atendem todo o espectro de funcionalidades de GC. Além disso, existe a necessidade de fornecer apoio a GC no ambiente WebAPSEE. Algumas das principais contribuições da ferramenta são: (i) possibilitar o planejamento de gerência de conhecimento integrado à máquina de execução do WebAPSEE; (ii) realizar coleta, manutenção e disseminação ativa do conhecimento; (iii) permitir o gerenciamento e acesso de informações sobre especialistas e (iv) geração de relatórios.

Atualmente, a ferramenta está sendo utilizada no Centro de Tecnologia da Informação e Comunicação da Universidade Federal do Pará (CTIC-UFPA), avaliado no nível G do MPS.BR, e no Laboratório de Engenharia de Software da UFPA, para auxiliar na implantação do processo de Gerência de Conhecimento. Espera-se como trabalhos futuros a realização de estudos de caso avaliando a utilização da ferramenta, bem como contribuir com resultados para uma possível avaliação oficial no nível E do Modelo MPS no CTIC-UFPA.

A ferramenta WKM, o manual de instalação e de uso podem ser encontrados no link "<http://labes.ufpa.br/likem/wkm>" e a mesma encontra-se sob a licença BSD (*Berkeley Software Distribution*).

Referências

Curtis, B., Hefley, W. E., and Miller, S. A. (2009) "The People CMM: a Framework for Human Capital Management". 2nd. Addison-Wesley Professional.

Fischer, G., Ostwald, J., (2001), "Knowledge Management: Problems, Promises, realities, and Challenges", IEEE Intelligent Systems, v. 16, n. 1 (Jan/Feb), pp. 60-72.

Holz, H., (2003) "Process-Based Knowledge Management Support for Software Engineering", Doctoral Dissertation, University of Kaiserslautern.

ISO/IEC 12207 International Organization for Standardization and International Electrotechnical Commission (2008) "ISO/IEC 12207:2008 Systems and Software Engineering -Software Life Cycle Processes". Genebra: ISO, Março.

Lima Reis, C. A.; Reis, R. Q. (2007) "Laboratório de Engenharia de Software e Inteligência Artificial: Construção do Ambiente WebAPSEE". ProQualiti – Qualidade na Produção de Software. v. 3, n. 1, junho de 2007. p. 43-48.

Montoni, M. A.; (2003) "Aquisição de Conhecimento: Uma Aplicação no Processo de Desenvolvimento de Software". Dissertação de Mestrado COPPE/UF RJ.

Natali, A. C., (2003) "Uma Infra-Estrutura para Gerência de Conhecimento em um Ambiente de Desenvolvimento de Software". Dissertação de Mestrado.UFES.

Oliveira, J. F. ; Andrade, G. F. ; Tavares, L. C. ; Reis, C. A. L. (2009). "Planejamento e Execução de Gerência do Conhecimento em um Ambiente de Desenvolvimento de Software". In: VIII Simpósio Brasileiro de Qualidade de Software, Ouro Preto.

Oliveira, J. F. ; Reis, C. A. L. (2009) . "Apoio Automatizado à Elaboração de Planos de Gerência de Conhecimento para Processos de Software". In: XII Conferência Iberoamericana de Ingenieria de Requisitos y Ambientes de Software, 2009, Medelín.

Oliveira, J.F.; (2010) "Abordagem para Implantação de Gerência do Conhecimento com Apoio de um Ambiente de Desenvolvimento de Software Centrado em Processos". Dissertação de Mestrado. PPGCC - UFPA. Janeiro de 2010, Belém.

O'Leary, D. E., (1998), "Knowledge-Management Systems: Converting and Connecting", IEEE Intelligent Systems, v. 13, n. 3 (May/Jun), pp. 30-33

Probst, G. J. B.; Raub, S.; Romhardt, K. (1999) "Managing Knowledge: Building Blocks for Success", 368 pp, Ed. Wiley.

Rus, I.; Lindvall, M., (2002) "Knowledge management in software engineering", Software, IEEE, vol.19, no.3, pp.26-38.

SOFTEX. Associação para Promoção da Excelência do Software Brasileiro (2009) "MPS.BR – Guia de Implementação do Nível E", Maio de 2009.

Integração da Gerência de Configuração com a Gerência de Projetos e de Requisitos em um Ambiente Colaborativo

Jonnathan Carvalho, Mario Amaral, Mara Barcelos, Simone Vasconcelos, Aline Vasconcelos
Projeto Quali-EPT – Instituto Federal Fluminense (IFF)
Rua Dr. Siqueira, 273 – Parque Dom Bosco – CEP 28030-130 – Campos dos Goytacazes – RJ – Brasil
{joncarv,mamaral,mrbarcelos,simonevs,apires}@iff.edu.br

Abstract. *This paper presents a tool to support Software Configuration Management, integrated in a collaborative environment, making easier the work of distributed teams. From its use, it is possible versioning the documents produced in this environment, generating baselines, and to create and to follow maintenance requests. Its main goal is, through its integration with other development processes, to ensure that the generated products are established and maintained, increasing their final quality.*

Resumo. *Este artigo apresenta uma ferramenta para apoiar a Gerência de Configuração de Software, integrada a um ambiente colaborativo, facilitando o trabalho de equipes distribuídas. Com seu uso, é possível versionar os documentos produzidos neste ambiente, gerar baselines e criar e acompanhar pedidos de manutenção. Seu principal objetivo é, através da integração com outros processos do desenvolvimento, garantir que os produtos gerados sejam estabelecidos e mantidos, aumentando a qualidade final destes produtos.*

1.Introdução

No contexto de um projeto de software, o gerente que começa sem um plano de projeto sólido compromete o sucesso do produto final [Pressman 2007]. Assim como o plano de projeto, outros documentos devem ser produzidos para uma gestão efetiva, considerando as pessoas envolvidas, o processo adotado e o produto a ser desenvolvido.

Todos estes documentos são produzidos ao longo do ciclo de vida de desenvolvimento de um software, desde sua concepção até entregas ao cliente. Porém, apesar do estabelecimento destes documentos representar um passo importante para a gestão de projetos, assim como para outras etapas do desenvolvimento, se as modificações nestes documentos não forem controladas, informações essenciais ao projeto podem se perder, prejudicando a qualidade do produto final.

Assim, para que haja um controle efetivo dos documentos produzidos, a Gerência de Configuração de Software (GCS) permite, através de técnicas e ferramentas de apoio, facilitar a manutenção, viabilizando o aumento da qualidade final do produto. A GCS é “uma disciplina que aplica procedimentos técnicos e administrativos para identificar e documentar as características físicas e funcionais de um Item de Configuração (IC), controlar as alterações nessas características, armazenar e relatar o processamento das modificações e o estágio da implementação e verificar a compatibilidade com os requisitos especificados” [IEEE 1990].

Todos estes aspectos apontam para um cenário em que a busca pela melhoria contínua constitui um fator crítico de sucesso. Neste contexto, o MPS.BR (2010a) – Melhoria de Processo do Software Brasileiro – define um modelo de melhoria do processo de desenvolvimento de software, que envolve os diversos processos da Engenharia de Software, como a Gerência de Projetos, de Requisitos e de Configuração, organizados em uma série de níveis incrementais para a sua implementação. O Projeto de Qualidade de Software Aplicada à Educação Profissional e Tecnológica – QUALI-EPT (2010) visa à implantação da qualidade de software nos projetos da Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais – RENAPI (2010), segundo as diretrizes do MPS.BR. Os projetos da RENAPI envolvem equipes distribuídas em todo o Brasil, gerando a necessidade da integração de todos os produtos desenvolvidos em um ambiente colaborativo. Com este objetivo, o QUALI-EPT atua junto a cada projeto, implementando os Níveis G e F do MR-MPS, mais especificamente os processos Gerência de Projetos (GPR) e Gerência de Requisitos (GRE) no Nível G, e Gerência de Configuração (GCO) no Nível F.

Apesar do MPS.BR estabelecer resultados esperados para adequação a cada processo, os meios para o seu atendimento não são determinados. Baseado nisso, o QUALI-EPT sugere metodologias e desenvolve ferramentas de apoio à adequação a estes processos. Assim, para apoiar a fase de Engenharia de Requisitos (ER), parte do processo GRE, a ferramenta *Fermine* apresentada em [Almeida *et al.* 2010], permite a criação e manutenção dos itens que contemplam a ER como: requisitos, regras de negócio, casos de uso, etc. Permite também definir a rastreabilidade entre casos de uso, requisitos, regras de negócio, etc. Com o mesmo propósito, foi desenvolvida uma ferramenta para apoiar a adequação ao processo GPR, através de funcionalidades como, por exemplo, definição da Estrutura Analítica do Projeto – EAP.

Para garantir que os documentos gerados nos processos GPR e GRE permaneçam íntegros durante todas as fases do desenvolvimento, este artigo apresenta a ferramenta *RedSCoM* (*Redmine Software Configuration Management*) para integração do processo GCO com os processos GPR e GRE em um ambiente colaborativo, de forma que os documentos produzidos sejam efetivamente estabelecidos e mantidos. A ferramenta apresentada possibilita: a) versionamento automático dos documentos gerados nos processos GPR e GRE; b) planejamento e geração de *baselines*; c) criação e acompanhamento de solicitações de manutenção em *baselines*, etc.

O fato da ferramenta *RedSCoM* estar integrada às demais em um ambiente colaborativo é importante, considerando que geralmente as soluções encontradas para satisfazer estes processos não estão integradas, tornando a manutenção dos documentos um trabalho complexo e, muitas vezes, manual. A integração de ferramentas em um ambiente colaborativo possibilita um ganho de produtividade em função do trabalho padronizado, além de disponibilizar informações centralizadas e documentos compartilhados, representando assim uma vantagem considerável.

O restante do artigo está organizado da seguinte forma: a Seção 2 apresenta o ambiente colaborativo de desenvolvimento; a Seção 3 apresenta a ferramenta de apoio à GCS integrada à GRE e GPR; a Seção 4 apresenta os trabalhos relacionados; e, a Seção 5 apresenta as conclusões e os trabalhos futuros.

2. O Ambiente Colaborativo de Desenvolvimento

A RedSCoM, assim como as ferramentas mencionadas, foi desenvolvida como *plugin* para o *Redmine* (2010), que é uma aplicação *web open-source* para gerenciamento de tarefas, desenvolvida em *Ruby on Rails*. Dentre as características que o torna um ambiente colaborativo, destacam-se: suporte a múltiplos projetos, compartilhamento de documentos e arquivos, notificações por e-mail, *wiki*, fórum e possibilidade de integração com SVN – *Subversion* (2010). Este ambiente, bem como seus *plugins*, é desenvolvido e mantido sob a licença GPL (*GNU General Public License v2*).

As aplicações desenvolvidas com o *framework Rails* são baseadas no padrão *Model-View-Controller* – MVC. A Figura 1 caracteriza os elementos presentes na arquitetura de aplicações *Rails*. O *Action Mailer* e *Action WebServices* atuam no envio de e-mails e comunicação com serviços *web*, respectivamente. A junção do *Action View*, responsável pela renderização de *templates*, com o *Controller*, compõe o *Action Pack* que, por sua vez, é o centro do *framework Rails*. O mapeamento objeto-relacional e a camada de persistência são providos pelo *Active Record*. O *Dispatcher* é responsável pelas requisições ao *Controller* apropriado e por recarregar a aplicação se houver dependências que necessitem de carregamento.

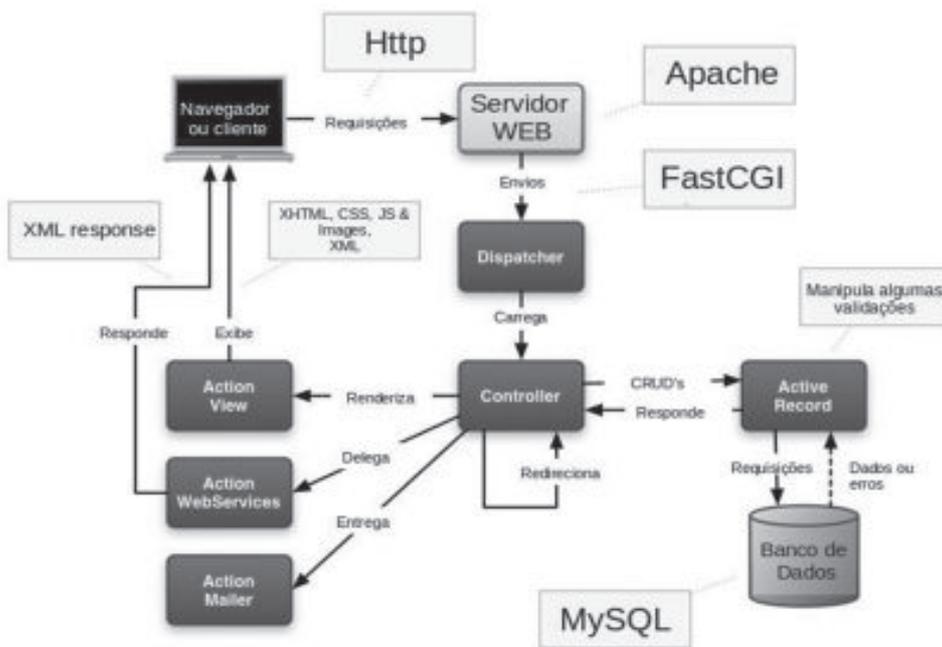


Figura 1. Arquitetura de aplicações *Rails* (adaptado de [VVN 2010])

3. RedSCoM: Ferramenta de Apoio à Gerência de Configuração

A motivação para o desenvolvimento de um *plugin* para apoiar as atividades da GCS no *Redmine* se deu pelo fato deste ambiente, apesar de possuir a opção de integração com o SVN, estar limitado apenas à visualização do repositório, não apresentando opções para seu manuseio direto. Desta forma, todos os documentos gerados na aplicação eram versionados através do uso de alternativas externas à aplicação. Neste sentido, o principal objetivo da RedSCoM é prover apoio automatizado aos resultados esperados do processo GCO em um ambiente colaborativo, assim como apoiar os demais processos do MPS.BR no que diz respeito à manutenção dos documentos gerados por estes processos, mantendo a sua integridade.

Pelo fato da RedSCoM ser desenvolvida incrementalmente, ela ainda não apóia o cumprimento de todos os resultados esperados do processo GCO, como, por exemplo, auditoria dos ICs que compõem uma *baseline*.

3.1. Principais Funcionalidades

Nas subseções a seguir são apresentados os resultados esperados do processo GCO e como as funcionalidades desta ferramenta auxiliam no atendimento a cada resultado.

3.1.1. GCO1 - Um Sistema de Gerência de Configuração é estabelecido e mantido

É necessário que sejam definidos os subsistemas que compõem o Sistema de Gerência de Configuração, a saber: a) sistema de controle de versões; b) sistema de controle de modificações e; c) sistema de gerenciamento de construção. Neste contexto, são definidos, respectivamente para cada sistema, as soluções: SVN, integrado ao *Redmine*; e o próprio *Redmine*, através da criação de solicitações de manutenção. A integração com uma ferramenta de gerenciamento de construção ainda é uma limitação, porém, considerando a realidade da RENAPI, este suporte é dado pelo ambiente de codificação adotado, também integrado ao SVN, permitindo versionamento de código fonte, como a IDE Eclipse (2010), por exemplo.

3.1.2. GCO2 - Os itens de configuração são identificados com base em critérios estabelecidos

Para o cumprimento deste resultado, a ferramenta permite a identificação dos ICs como parte do Plano de GCS, que também é disponibilizado na ferramenta de forma resumida. A Figura 2 apresenta o Plano de GCS.



Figura 2. Plano de Gerência da Configuração na RedSCoM

Formam o Plano de GCS: a) Introdução, onde é possível descrever o objetivo geral do plano; b) Itens de Configuração, onde são identificados, com base nos documentos gerados pelos *plugins* para GPR e GRE; c) *Baselines*, onde são planejadas e geradas, e; d) Infra-estrutura, onde é definida a estrutura

de diretórios do repositório em que cada tipo de artefato será mantido. É importante ressaltar que, nesta versão da RedSCoM, o Plano de GCS é resumido, considerando que não abrange todos os itens que podem estar presentes em um Plano de GCS.

Como mencionado, esta ferramenta permite o versionamento dos ICs, portanto, para isso, é necessário que eles sejam identificados, através da padronização de sua nomenclatura, como apresentado na Figura 3. Cabe ressaltar ainda que a lista de ICs disponíveis é carregada de acordo com o tipo dos documentos gerados pelo *plugin* Fermine (GRE) e pelo *plugin* para Gerência de Projetos, uma vez que a RedSCoM apóia o versionamento dos documentos gerados por estas ferramentas, integradas ao *Redmine*.



Figura 3. Identificação e padronização da nomenclatura dos ICs

3.1.3. GCO3 - Os itens de configuração sujeitos a um controle formal são colocados sob baseline

Em determinados momentos, no decorrer do projeto, é necessário que os ICs sejam agrupados formando uma *baseline* do software. Para isso, a ferramenta possibilita o planejamento e geração automática de *baselines*, conforme a Figura 4.



Figura 4. Planejamento e geração de *baselines*

Para criar uma *baseline*, a ferramenta solicita o *login* e *password* de acesso ao repositório. Após verificar estas informações, é recuperado o conteúdo do *trunk*, pasta onde são encontrados os ICs referentes à versão corrente de desenvolvimento, e, então, uma nova *tag* é gerada com este conteúdo, através do mecanismo de *tags* do SVN. Assim, a aplicação passa a reconhecer esta nova versão do software e um controle formal sobre ela passa a ser necessário.

3.1.4. GCO4 - A situação dos itens de configuração e das baselines é registrada ao longo do tempo e disponibilizada

Para alcançar este resultado, o *Redmine* disponibiliza meios de acesso a situação dos ICs, bem como das *baselines*, através da visualização do repositório, permitindo a verificação dos comentários das operações de *commit*, realização de *diff*, etc.

3.1.5. GCO5 - Modificações em itens de configuração são controladas

Toda modificação em um IC, quando ele passa a fazer parte de uma *baseline*, deve passar por um controle formal de modificações, em que é gerada uma solicitação de modificação a ser analisada pelo Comitê de Controle da Configuração (CCC), que é um grupo de pessoas responsável por avaliar e aprovar ou rejeitar modificações propostas em ICs [MPS.BR 2010b]. Para uma nova solicitação, o usuário, que pode ser um cliente ou um membro da equipe de desenvolvimento, utiliza a funcionalidade de criação de Nova Tarefa, adaptada para a RedSCoM, como mostra a Figura 5.

A imagem mostra a interface de criação de uma nova tarefa no sistema Redmine. O formulário contém os seguintes campos:

- Tipo:** Pedido de Manutenção
- Título:** Modificar geração da matriz de avaliação
- Descrição:** A matriz de avaliação deve recuperar todos os membros imediatamente subordinados ao papel estabelecido como Avaliador na pesquisa. Da forma está sendo feita, está recuperando todos os membros de todos os papéis subordinados, não apenas os imediatos.
- Situação:** Sob avaliação
- Prioridade:** Urgente
- Atribuído para:** (campo vazio)
- Versão:** AVALIA-RC-0.0.1
- Início:** 2010-08-26
- Data prevista:** (campo vazio)
- Tempo estimado:** Horas
- % Terminado:** 0 %

Figura 5. Nova solicitação de modificação

Uma solicitação deve ser feita descrevendo a modificação, a qual versão (*baseline*) do software ela é correspondente e o estado inicial da solicitação, que por padrão se encontra sob avaliação. Cabe ressaltar que os estados são definidos no *Redmine*, assim como a relação entre eles (*workflow*), com base em um processo pré-definido. Ao ser criada uma solicitação de modificação, os membros da equipe com o papel CCC, definido no próprio *Redmine* através da funcionalidade Papéis e Permissões, são notificados por e-mail para avaliar e classificar a modificação.

3.1.6. GCO6 - O armazenamento, o manuseio e a liberação de itens de configuração e baselines são controlados

Este resultado esperado é contemplado através da integração da RedSCoM, que trata do armazenamento dos ICs, com o *Redmine* e *SVN*, que possuem políticas de permissões e níveis de acesso com a definição de diferentes papéis para usuários, através de protocolos de autenticação. Além disso, o *SVN* trata o acesso concorrente à informações de forma satisfatória. É importante ressaltar que o *Redmine* é um ambiente que permite a integração com *SVN* apenas para visualização do repositório. O *plugin* RedSCoM, integrado ao *Redmine*, permite que o repositório seja manipulado diretamente através de ações de *check-out* e *commit*, quando um documento é selecionado para versionamento.

3.1.7. Integração com os processos GPR e GRE

Um dos objetivos da RedSCoM é a integração com os processos GPR e GRE. Esta integração se dá através do versionamento dos documentos gerados nestes processos, através dos *plugins* para GPR e GRE, como pode ser visto na Figura 6. Os ICs disponíveis para versionamento que são listados na RedSCoM são baseados nos artefatos gerados por estes *plugins*, como Documento de Requisitos, Especificação de Casos de Uso, EAP, etc. Por este motivo, inicialmente, a RedSCoM está limitada ao versionamento dos artefatos gerados nestes processos, porém, à medida que soluções para apoiar os demais processos forem implementadas, a RedSCoM está preparada para se adaptar também a elas.

Para versionar um documento, a ferramenta realiza *check-out* do repositório, criando uma cópia local, gera o documento a ser versionado em formato XML (*eXtensible Markup Language*), adiciona (*add*) o documento e o envia para o repositório através de um *commit*, de forma transparente para o usuário.



Figura 6. Versionamento de artefatos pela RedSCoM

4. Trabalhos Relacionados

Dentre as ferramentas pesquisadas, com o objetivo de verificar o suporte aos resultados esperados do processo GCO, integrado aos processos GPR e GRE, destacam-se: *Bugzilla*, *Trac* e *Enterprise Architect*. A *Enterprise Architect* (2010), que é uma ferramenta de modelagem proprietária da *Sparx Systems* que disponibiliza funcionalidades para integração com a Gerência de Requisitos, atende ao resultado

esperado GCO3 do processo GCO. O *Bugzilla* (2010) é uma ferramenta *free* que atende aos seguintes resultados esperados do processo GCO: GCO1, GCO4, GCO5 e GCO6. A *Trac* (2010) também é *free*, porém é mais completa que o *Bugzilla*, no contexto desta comparação, já que além de atender aos mesmos resultados, possui ainda integração com a Gerência de Projetos. A Tabela 1 mostra a comparação entre as ferramentas

Critérios	Ferramentas				
	Bugzilla	Trac	Enterprise Architect	Redmine	RedSCoM
GCO1	✓	✓	x	x	✓
GCO2	x	x	x	x	✓
GCO3	x	x	✓	x	✓
GCO4	✓	✓	x	✓	✓
GCO5	✓	✓	x	x	✓
GCO6	✓	✓	x	x	✓
GCO7	x	x	x	x	x
Integração com GPR	x	✓	x	x	✓
Integração com GRE	x	x	✓	x	✓
Solução livre	✓	✓	x	✓	✓

Tabela 1. Comparação entre as ferramentas

5. Conclusões e Trabalhos Futuros

Este artigo apresentou uma ferramenta de apoio ao processo GCO do MPS.BR, integrado aos processos GPR e GRE em um ambiente colaborativo. Esta integração se dá através do versionamento dos documentos gerados neste ambiente, permitindo centralizar os produtos de trabalho em apenas um ambiente de desenvolvimento. Além disso, destacam-se também: definição resumida do Plano de GCS; planejamento e geração de *baselines*; criação e acompanhamento de solicitações de modificação, etc. Como trabalhos futuros, encontram-se: apoio a realização de auditorias em *baselines* (resultado esperado GCO7); disponibilização de *checklist* para definição dos ICs; integração com outros processos, em formato de *plugins*; disponibilização no portal de software público brasileiro.

6. Referências

Almeida, G.T., Ramos, B.A., Vasconcelos, A.P.V., *et al.*, (2010), Ferramenta de Apoio à Engenharia de Requisitos Integrada a um Ambiente Colaborativo de código Aberto. *In*: Congresso Brasileiro de Software: Teoria e Prática (CBSOFT), Sessão de Ferramentas, Salvador, Ba, Brasil, setembro (a ser publicado).

Bugzilla, (2010). Disponível em: <http://www.bugzilla.org>. Acessado em 26/08/2010.

Eclipse, (2010). Disponível em: <http://www.eclipse.org>. Acessado em 26/08/2010.

Enterprise Architect, (2010). Disponível em: <http://sparxsystems.com/products/ea>. Acessado em: 26/08/2010.

IEEE, (1990), Std 610.12 - IEEE Standard Glossary of Software Engineering Terminology, Institute of Electrical and Electronics Engineers.

MPS.BR, (2010a), Melhoria de Processo do Software Brasileiro. Disponível em: http://www.softex.br/mpsbr/_home/default.asp. Acessado em 26/08/2010.

MPS.BR, (2010b), Guia de Implementação - Parte 2: Fundamentação para Implementação do Nível F do MR-MPS: 2009. Disponível em: <http://www.softex.br>.

Pressman, R. S., Engenharia de Software, Makron Books, 6ª edição. São Paulo. 2007

QUALI-EPT, (2010), Projeto de Qualidade de Software Aplicada à Educação Profissional e Tecnológica. Disponível em: <http://www.renapi.org/qualidade/conheca-o-projeto/apresentacao>. Acessado em 26/08/2010.

Redmine, (2010). Disponível em: <http://www.redmine.org>. Acessado em 26/08/2010.

RENAPI, (2010), Portal da Rede Nacional de Pesquisa e Inovação em Tecnologias Digitais. Disponível em: <http://www.renapi.org>. Acessado em 26/08/2010.

Subversion, (2010). Disponível em: <http://subversion.tigris.org>. Acessado em 26/08/2010.

Trac, (2010). Disponível em: <http://trac.edgewall.org/>. Acessado em 27/08/2010.

VVN, (2010). Disponível em: <http://vvn.net/wp/tag/ruby/>. Acessado em: 27/08/2010.

