

VI CONGRESSO BRASILEIRO DE SOFTWARE • TEORIA E PRÁTICA

CBSOFT

MARINGÁ 2016

Trilha da Indústria

CBSOFT.ORG

REALIZAÇÃO:



EXECUÇÃO:



ORGANIZAÇÃO:



APOIO:



FOMENTO:



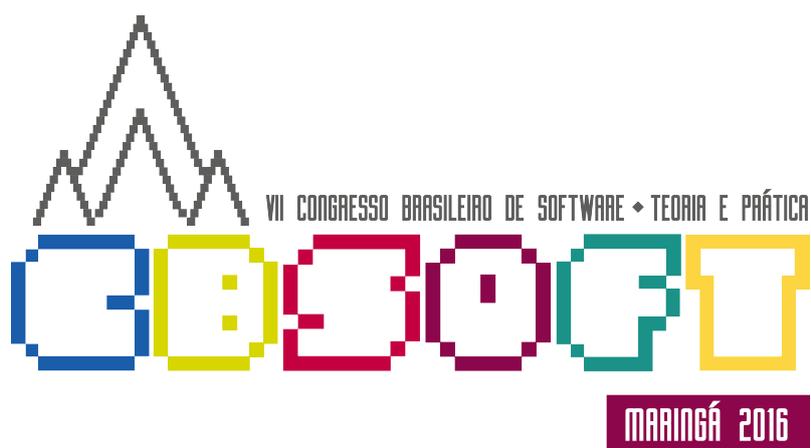
PATROCINADORES GIGA:



PATROCINADORES MEGA:



ThoughtWorks



TRILHA DA INDÚSTRIA | *INDUSTRY TRACK*

19 a 20 de setembro de 2016 | *September 19 - 20, 2016*
Maringá, PR, *Brazil*

ANAIS | *PROCEEDINGS*

Sociedade Brasileira de Computação – SBC

COORDENADOR DO COMITÊ DE PROGRAMA | *PROGRAM COMMITTEE CHAIR*

Auri M. R. Vincenzi (UFSCar)

EDITORES | *PROCEEDINGS CHAIRS*

Marco Aurélio Graciotto Silva (UTFPR)
Willian Nalepa Oizumi (IFPR)

COORDENADORES GERAIS | *GENERAL CHAIRS*

Edson Oliveira Júnior (UEM)
Thelma Elita Colanzi (UEM)
Igor Steinmacher (UTFPR)
Ana Paula Chaves Steinmacher (UTFPR)
Igor Scaliante Wiese (UTFPR)

REALIZAÇÃO | *REALIZATION*

Sociedade Brasileira de Computação (SBC)

EXECUÇÃO | *EXECUTION*

Universidade Estadual de Maringá (UEM) – Departamento de Informática (DIN)
Universidade Tecnológica Federal do Paraná (UTFPR) – Câmpus Campo Mourão (UTFPR-CM)

Apresentação

A Trilha da Indústria faz parte do CBSoft e tem o objectivo de contribuir para estreitar a relação indústria-governo-academia. Ela pretende ser um fórum para apresentações de resultados de pesquisa e desenvolvimento no contexto da indústria, eventualmente, em cooperação com as universidades. Destina-se também a estimular a interação entre academia e indústria. Um dos objetivos da trilha Indústria é discutir as possibilidades de cooperação academia-indústria (e governo), visando estabelecer condições e financiamento para a investigação, desenvolvimento e inovação, bem como a formação de recursos humanos qualificados, de forma integrada. Nesse sentido, agências de fomento brasileiras oferecem algumas iniciativas, como o Programa INCT, e o Programa de Formação de Recursos Humanos em Áreas Estratégicas (RHAE) (apoiados pelo CNPq); e os Centros de Pesquisa, Inovação e Difusão (CEPID) e o programa , a Pesquisa Inovativa em Pequenas Empresas (PIPE) (apoiados pela FAPESP), dentre outras. Tais perspectivas favorecem a transferência de tecnologia e a tendência de estreitar a geração e o uso efetivos de resultados de pesquisa, desenvolvimento e inovação.

Este ano, a trilha conta com três seções técnicas com a apresentação de 11 trabalhos, relatando iniciativas interessantes no contexto da indústria, tais como desenvolvimento distribuído de software, métodos ágeis, melhoria de processos de software, dentre outros. Além disso, teremos uma palestra convidada do Prof. Dr. Fabio Kon (IME/USP), em cooperação com a trilha Ideias Inovadoras, com o título “Inovação Aberta, Ecossistemas de Startups e sua Evolução”.

Finalmente, eu gostaria de agradecer o apoio que os organizadores da trilha do ano anterior me deram durante a organização desta edição; os autores que apresentaram as propostas para esta edição da trilha; o comitê de programa e os revisores que avaliam as propostas submetidas; e a comissão organizadora do CBSoft’2016. Obrigado a todos e espero que apreciem o evento este ano.

Auri M. R. Vincenzi (UFSCar)
Coordenador da Trilha da Indústria do CBSoft’2016

Foreword

The Industry Track is part of CBSoft and has the aim of contributing to the enhancement of the academia-industry-government relationship. It intends to be a forum for presentations of research and development results in the industry context, eventually in cooperation with academia. It aims to stimulate the interaction between academia and industry. Brazilian Funding Agencies offers some initiatives in this direction, like the INCT Program, and the Training Program of Human Resources in Strategic Areas (RHAE) (supported by CNPq), the Innovative Research in Small Business (PIPE) and the Centers for Research, Innovation and Dissemination (CEPIDs) (supported by FAPESP), besides others initiatives.

One of the Industry Track goals is discussing the possibilities of cooperation between academia and industry (and government), aiming at establishing conditions and funding for research, development, and innovation, as well as the formation of qualified human resources in an integrated manner. Such perspectives favor the technology transfer and enhance the effective generation and use of results from research, development, and innovation.

This year, the track has three technical sections with the presentation of 11 papers, presenting exciting initiatives in the industry context, encompassing relevant topics such as distributed software development, agile methods, software process improvement. Moreover, we will have an invited talk of Prof. Dr. Fabio Kon (IME/USP), in cooperation with Insightful Ideas track.

I would like to thank the chairs for the previous edition for their support; the authors who submitted proposals for this edition of the Industry Track at CBSoft'2016; the program committee and the reviewers who evaluate the submitted proposals; and the organizing committee of CBSoft'2016. Thank you all and enjoy the event.

Auri M. R. Vincenzi (UFSCar)
Chair of the CBSoft'2016 Industry Track

Comitê técnico | *Technical committee*

Coordenador de comitê de programa | *PC chair*

Auri Marcelo Rizzo Vincenzi (UFSCar)

Comitê de programa | *Program committee*

Adenilso Simão (ICMC-USP)
Alexandre Alvaro (UFSCar)
Alexandre Gomes (UnB)
Avelino Zorzo (PUCRS)
Celso Gonçalves Camilo Junior (UFG)
Cesar Marcondes (UFSCar)
Daniel Lucrédio (UFSCar)
Eduardo Guerra (INPE)
Fabiano Ferrari (UFSCar)
Fernando Trinta (UFC)
Frederico Durão (UFBA)
Gleison Santos (UNIRIO)
Higor Souza (IME-USP)
José Carlos Maldonado (ICMC-USP)
Leandro M. Nascimento (UFRPE)
Marcelo Fantinato (EACH-USP)
Marcos Chaim (EACH-USP)
Neilson Ramalho (USP)
Paulo Merson (SEI, TCU e Alcenit)
Simone Souza (ICMC-USP)
Vanilson Burégio (UFRPE)
Vinicius Garcia (UFPE)

Revisores externos | *External reviewers*

Eduardo Freitas (IFG)
Jacson Rodrigues Barbosa (UFG)
Kenyo Faria (IFG)
Marllos Prado (UFG)
Viviane Malheiros (SERPRO)

Artigos técnicos | *Technical papers*

Sessão Técnica 1 (ST 1)

BigData de Música Usando Atores em Scala <i>Daniel Cukier (IME-USP)</i>	1
Distributed Development of Cognitive Systems at IBM Research - Brazil <i>Maíra Gatti de Bayser (IBM Research - Brazil), Alan Braz (IBM Research - Brazil), Claudio Pinhanez (IBM Research - Brazil)</i>	4
Os Desafios da Gestão de Projetos de Desenvolvimento Distribuído de Software <i>Fernanda Kussama Pellegrini (SIDI)</i>	6

Sessão Técnica 2 (ST 2)

SISCAPMobile (Sistema para Controle Administrativo Penitenciário na versão móvel) <i>Jobson A. Nascimento (Thoths - Soluções Inteligentes, CESMAC), Egydio Ricardo G. Junior (Mauricio de Nassau)</i>	10
SPIN-Campinas Promovendo a Troca de Experiências entre Empresas e Academia <i>Fernanda Pellegrini (SIDI - Brazil)</i>	14
Precificação de Sistemas com Código Complexo: Um Estudo de Caso com APF e COSMIC <i>Nadia Rafaela Costa (SERPRO), Francisco José Rêgo Lopes (SERPRO), Emerson Mendes Portela Costa (DATAPREV), Ana Luíza Bessa de Paula Barros (UECE), Paulo Henrique Maia (UECE)</i>	19
Projeto de um Ambiente de Desenvolvimento para Métodos Ágeis <i>Marcelo L. Ribeiro (UEM), Itana M. de S. Gimenes (UEM)</i>	23

Sessão Técnica 3 (ST 3)

Architectural Analysis for Improvement of Software Quality produced by SRBR <i>Marcia C. C. Costa (Samsung Research Center Brazil), Gizelle S. Lemos (Samsung Research Center Brazil), André Chiarelli (Samsung Research Center Brazil)</i>	27
Análise Qualitativa Entre os Processos da Instrução Normativa IN/SLTI/MPOG 04/2014 e o Modelo CMMI-ACQ <i>Luiz Sérgio P. Silva (UFPE), Renata T. Moreira (UFPE), Alexandre M. L. Vasconcelos (UFPE), Suzana C.B. Sampaio (UFPE)</i>	32

Implementação do Processo de Gerência de Risco usando o Método Ágil RisAgi para o Alcance do Nível de Maturidade 3 do CMMI-DEV: O Caso da EMREL

Petrônio A. de Medeiros (EMPREL), Moisés B. de Leal Junior (EMPREL), Homero A. Cavalcanti (EMPREL), Ana Cristina Rouiller (SWQuality Consultoria e Sistemas Ltda.), Sandro Ronaldo Bezerra Oliveira (SWQuality Consultoria e Sistemas Ltda., UFPA), Kleoson Bruno Corrêa dos Santos (UFPA)

36

Resultados e Lições Aprendidas na Implementação do Modelo CMMI-DEV, Nível de Maturidade 2, usando Metodologias Ágeis: Um Caso de Sucesso na Jambu Tecnologia

Marcelo Rocha de Sá (Jambu Tecnologia Consultoria e Engenharia Elétrica Ltda), Sandro Ronaldo Bezerra Oliveira (UFPA)

40

BigData de música usando atores em Scala

Daniel Cukier

Universidade de São Paulo / Playax
danicuki@ime.usp.br

Abstract

How to create a scalable architecture, robust and efficient to map data from thousands of sources like radio, TVs and music websites? Playax is a Startup that provides data intelligence to the music industry and uses, besides Ruby, Scala language and its actors model. We will show in this talk details and code from this architecture and how it is solving the problem of thousands of artists.

Resumo

Como criar uma arquitetura escalável, robusta e eficiente para mapear dados de milhares de fontes como rádios, TVs e sites de música? A Playax é uma startup que provê inteligência de dados para o mercado da música e usa, além de Ruby, a linguagem Scala e seu modelo de atores. Mostraremos nessa palestra detalhes e códigos dessa arquitetura e como ela está resolvendo o problema de milhares de artistas.

1.1. Resumo

Nesse projeto foram feitas otimizações no algoritmo opensource de busca de faixas por *fingerprint* (echoprint). Foi implementada a versão *multithread* do algoritmo e suporte para escalabilidade horizontal (utilizando múltiplas máquinas virtuais na nuvem). As melhorias implementadas no algoritmo de *audio-fingerprint* possibilitaram um ganho de performance de consumo de CPU de até 10 vezes. O algoritmo existente antes do projeto, permitia a execução de apenas uma chamada ou consulta de *fingerprint* por vez. As melhorias foram implementadas em 2 fases:

- Alteração para torna-lo thread safe e permitir múltiplas chamadas simultâneas
- Alterações nas configurações do solr, para que o sistema pudesse rodar de maneira distribuída, em vários servidores. Com essa nova configuração, a base de dados de fingerprints seria replicada automaticamente entre os servidores. A arquitetura usada foi master-slave, onde a escrita (inclusão de novos dados) era feita em um único servidor, e a leitura era feita através de um Load Balancer que apontava para todos os servidores do cluster.

- Configuração na nuvem de um Load Balancer usando o software haproxy, para distribuir a carga de chamadas ao sistema entre os vários servidores de áudio-fingerprint
- Mudança no algoritmo para otimização. O algoritmo anterior usava uma ordenação de array, que tinha complexidade $O(n \log n)$. A mudança feita trocou o código de ordenação *quicksort* por uma implementação sem necessidade de ordenação e complexidade $O(n)$. Isso causou uma redução no tempo de resposta para consultas. O tempo médio, que antes era de 4s, passou a ser de 800ms. Com isso, o gasto com recursos de CPU caíram 3x após a implementação dessa melhoria.

Nessa apresentação mostraremos a arquitetura do sistema de escuta escalável de milhares de rádios e sites e a integração dessa escuta com o algoritmo de audiofingerprint.

1.2. Tópicos

- Modelo de atores em Scala: nativo e Akka
- Audio fingerprint – implementação própria e escalável do algoritmo de identificação automática de músicas
- Chamadas de web services – integração com APIs públicas de vários serviços de música como YouTube, Soundcloud, Deezer, etc.
- Arquitetura distribuída para aferição fontes musicais / streamings – como ouvir milhares de streamings de rádios, TVs e sites de maneira escalável.
- Chamada de código nativo C a partir de Scala / JVM
- Padrão Throttler: controle de fluxo de chamadas
- Banco de dados chave valor distribuído: Aerospike.

1.3. Exemplos de tela do sistema

www.playax.com

BEM-VINDO A PLAYAX

Que tal explorar a plataforma? Veja o número de plays, envie músicas para as rádios, confira o desempenho e fique de olho nas oportunidades.

Última detecção em rádio 15/05 02:29
LÁGRIMA DE AMOR (DANIELLA ALCARPE)
 EDUCATIVA 97.5 FM - CONGONHAS/MG

Fale com a rádio

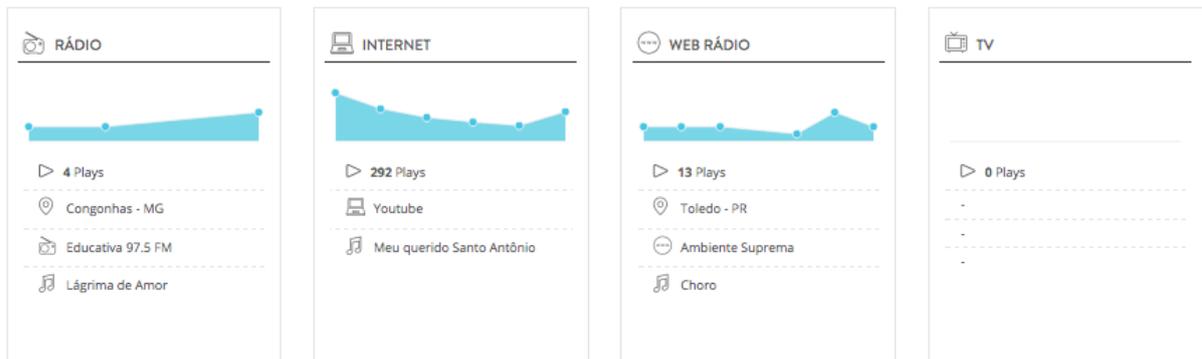
CRESCIMENTO EM
 Nova Iguaçu - RJ **+ 100%**

Desempenho por localidade

QUEDA EM
 Itamonte - MG **- 71%**

Faça uma campanha

ÚLTIMOS 7 DIAS



[Ver desempenho](#)

1.4. Audiência

Desenvolvedores de software de nível intermediário e avançados, conhecedores de paradigma de programação funcional. Pré-requisito: conhecimentos básicos de linguagem funcional, orientação a objetos.

1.2. Biografia

Daniel Cukier trabalha com desenvolvimento de software desde 1995, principalmente com Java, Scala e Ruby (Rails). Após 2 anos como CTO da Elo7, tornou-se conselheiro técnico da Monashees Capital. Atualmente é CTO da Playax, uma plataforma de inteligência de dados na área de música. Como aluno de doutorado, faz parte do grupo de pesquisas em empreendedorismo digital e Startups de Software do IME-USP.

Fez mestrado também no IME-USP com a dissertação Padrões para Introduzir Novas Ideias na Indústria de Software. Ativo na comunidade ágil e de desenvolvimento de software desde 2005, quando ingressou na Agilcoop, palestrando em dezenas de eventos, produzindo podcasts, artigos e ministrando cursos sobre agilidade. Introduziu o uso de Métodos Ágeis na Locaweb, maior empresa de hosting da America Latina, onde participou de vários projetos de software na área de SaaS e IaS. Organizou o evento DevOpsDays Brasil, um encontro de desenvolvedores com sysadmins.

Blog: www.agileandart.com

Distributed Development of Cognitive Systems at IBM Research – Brazil

Maíra Gatti de Bayser¹, Alan Braz¹, Claudio Pinhanez¹

¹Social Data Analytics Group
IBM Research – Brazil

{mgdebayser, alanbraz, csantosp}@br.ibm.com

Abstract. *Managing software development of distributed teams is not so easy, managing cognitive systems development is even harder; together these processes pose still more challenges. In this talk we will present the challenges that we face in our daily work on developing Cognitive Systems at IBM Research – Brazil, how we overcome them and the tools that we use to mitigate these challenges.*

Resumo. *Gerenciar o desenvolvimento de software de times distribuídos não é tão fácil, gerenciar o desenvolvimento de sistemas cognitivos é ainda mais difícil, juntos estes processos apresentam ainda mais desafios. Nesta apresentação, vamos apresentar os desafios que enfrentamos em nosso trabalho diário no desenvolvimento de Sistemas Cognitivos na IBM Research – Brazil, como nós o superamos e as ferramentas que usamos para minimizar os mesmos.*

1. Agenda

The talk is structured as follows:

- IBM Research – Brazil Presentation
 - Social Data Analytics Team Presentation
- Challenges on Distributed Development
- Challenges on Cognitive Systems Development
- Methods and Tools for Managing Distributed Development of Cognitive Systems

2. Audience

The talk targets all kinds of students from undergrad to PhD students that would like to learn the topics, and all kinds of professionals, from researchers, to professors or software engineers that could apply the lessons learned in their environment.

3. Biography

Maíra Gatti de Bayser¹ is a Research Staff Member (RSM) and Technical Leader in the Social Data Analytics group at IBM Research, Brazil since 2011. She was nominated Master Inventor in 2014 and she has achieved the 6th patent plateau in 2016. Maíra holds Ph.D. (2009) and M.Sc. (2006) in Software Engineering for Multi-Agent Systems from

¹M.G. de Bayser IBM Research page: <http://researcher.ibm.com/person/br-mgdebayseris>

Informatics Department of the PUC-Rio, Brazil. She has been a research fellow at the Agent and Intelligent Systems of Kings College London, England (2009), and before a visit student at the University of Waterloo, Canada (2008).

Alan Braz² is a Research Software Engineer (RSE) at IBM Research, Brazil working as part of the Social Data Analytics group with a focus on architecture and software development methodologies. Alan holds a Master (2013) in Computer Science from UNICAMP (State University of Campinas) researching about “Developing Reliable Component-based Software with Agile Methods”. Since 2005 working at IBM as a software architect, technical team leader, Scrum Master and Agile Coach. Certified Scrum Master by AgileAlliance and Enterprise Architect for Java EE 5.

Claudio Pinhanez³ is a researcher, professor, and media artist. He is the leader of the Social Data Analytics research group of IBM Research Brazil, where he works on Social and Human Data Analytics, Ubiquitous Computing, and Human-Computer Interfaces. Claudio has a PhD from MIT Media Laboratory (1999) and has been at IBM Research since 1999. He is a Senior Member of ACM, a member of the IBM Academy of Technology, and an associated editor of 3 scientific journals.

²A. Braz IBM Research page: <http://researcher.ibm.com/person/br-alanbraz>

³C.S. Pinhanez IBM Research page: <http://researcher.ibm.com/person/br-csantosp>

Os Desafios da Gestão de Projetos de Desenvolvimento Distribuído de Software

Fernanda Kussama Pellegrini

Samsung Instituto para Desenvolvimento da Informática – Rua Aguaçú, 171 – cep
13098-321 – Campinas – SP – Brasil

f.kussama@sidi.org.br

***Abstract:** project management discipline is directly affected by the distributed software development (DSD) making academia and industry continue researching and studying on the subject. In order to contribute to these researches and studies, this presentation aims to analyze the challenges and best practices that support project management identified by [COSTA, 2014] from the point of view of SIDI (Samsung Development Institute for Information Technology) besides indicating additional challenges in research and development projects carried out in partnership with other organizations.*

***Resumo:** a disciplina de gestão de projetos é diretamente afetada pelo desenvolvimento distribuído de software (DDS) fazendo com que academia e indústria continuem pesquisando e estudando sobre o tema. Com o objetivo de contribuir com estas pesquisas e estudos, esta apresentação se propõe a analisar os desafios e melhores práticas que suportam a gestão de projetos identificados por [COSTA, 2014] do ponto de vista do SIDI (Samsung Instituto de Desenvolvimento para Informática), além de indicar desafios adicionais enfrentados nos projetos de pesquisa e desenvolvimento realizados em parceria com outras organizações.*

1. Apresentação

O desenvolvimento distribuído de software (DDS) tem sido sistematicamente empregado em grandes corporações há décadas, transformando a cultura das organizações e a forma como seus colaboradores trabalham. As pessoas são cada vez mais capazes de se comunicar em uma linguagem comum e de trabalhar com ferramentas e tecnologias que favorecem a colaboração entre equipes. As empresas, por sua vez, têm incentivado seus colaboradores, através de treinamentos e orientações, a trabalharem de forma global, considerando e respeitando as diferenças culturais e a diversidade. Conseqüentemente, os mecanismos de desenvolvimento de software e gestão de projetos têm sido afetados por todas estas mudanças e, portanto, são objetos de pesquisas e estudos tanto no meio acadêmico quanto industrial.

[COSTA, 2010] e [COSTA, 2014] fizeram extensas revisões da literatura relacionada ao DDS, selecionando os trabalhos mais relevantes da área. No primeiro trabalho foram analisados 54 estudos primários e no segundo 224, buscando informações sobre desafios e melhores práticas que suportam a gestão de projetos de DDS. No trabalho mais recente, foram identificados 32 desafios, 28 melhores práticas,

assim como 31 ferramentas e 14 modelos de apoio aos projetos quando o desenvolvimento de software é distribuído, complementando o trabalho anterior.

Analisando estes desafios e melhores práticas do ponto de vista do SIDI (Samsung Instituto de Desenvolvimento para Informática) - um instituto privado de ciência e tecnologia brasileiro do setor de informática e telecomunicações que atua como interveniente técnico de projetos de software desenvolvidos em parceria com empresas incentivadas pela Lei de Informática (leis 8.248/91 [Brasil 1991], 10.176/01 [Brasil 2001], 11.077/04 [Brasil 2004] e 13.023/14 [Brasil 2014] e Decreto 5906/2006 [Brasil 2006]) - percebe-se que vários dos desafios já não criam dificuldades adicionais às atividades rotineiras de gestão de projetos, ou seja, são desafios que independem da distância física entre as equipes de projeto. Outros, porém, possuem grande impacto nos projetos, alguns com soluções que os endereçam parcialmente e outros sem solução sistemática.

Além do mais, a experiência do SIDI mostra que existem outros desafios enfrentados pelas organizações que gerenciam projetos de desenvolvimento distribuído de software. São eles:

- Colaborar para o aumento da capacidade das organizações da cadeia produtiva.
- Planejar os projetos adequadamente do ponto de vista financeiro.
- Colaborar para flexibilidade da cadeia produtiva.

Assim, os objetivos desta apresentação são:

- Discutir os desafios e melhores práticas identificados por [COSTA, 2014] do ponto de vista do SIDI.
- Apresentar desafios adicionais relacionados à Gerência de Projetos de Desenvolvimento Distribuído de Software baseados na experiência do SIDI.

Para atingir estes objetivos, os seguintes tópicos serão abordados na apresentação:

- Visão Geral do DDS;
- Lei de Informática;
- Apresentação do trabalho de [COSTA, 2014];
- Contexto do Estudo de Caso
- Desafios e Melhores Práticas dos ambientes DDS apontados por [COSTA, 2014] do ponto de vista do SIDI.
- Desafios Adicionais à Gerência de Projetos de Software distribuídos identificados pelo SIDI.

2. Audiência

Esta apresentação é voltada principalmente a gerentes de projetos. Porém, por ser um tema interessante a todos os envolvidos em projetos de desenvolvimento de software e

por não existirem pré-requisitos para assisti-la, a audiência é aberta a todos os envolvidos com software, seja do lado dos clientes, seja do lado dos fornecedores, seja quem assume funções técnicas, seja quem assume funções gerenciais.

3. Breve Biografia

Fernanda Kussama Pellegrini é Bacharel em Ciência da Computação (1998) pela Universidade Federal de São Carlos (UFSCar) e Especialista em Engenharia de Software (2001) e Engenharia da Qualidade (2006) pela Universidade Estadual de Campinas (UNICAMP). Iniciou sua carreira profissional como estagiária na IBM, em 1998. Depois disso, trabalhou com desenvolvimento de software no CPqD, ADC e Motorola. Nessas empresas teve oportunidade de vivenciar diferentes fases do desenvolvimento de software, como análise de requisitos, programação, testes e implantação de sistemas no cliente. Na Motorola começou a trabalhar com Processos e Qualidade de Software em 2003 e na Robert Bosch, a partir de 2009, expandiu a área de atuação trabalhando com qualidade e processos de desenvolvimento de hardware e sistemas. Desde setembro de 2012, trabalha no SIDI (Samsung Instituto para Desenvolvimento da Informática) como Coordenadora do PMO (Process and Project Management Office), implantando processos de gerenciamento de projetos e suportando os projetos na tomada de decisões. Tem experiência na área de garantia e controle da qualidade, definição e melhoria de processos, auditorias, lições aprendidas, ISO 9000, TL 9000, ISO/TS 16496, OHSAS 18000, ISO 14001, ISO 19011, CMMI e em metodologias e ferramentas Seis Sigma para desenvolvimento de processos e produtos. De 2004 a 2014, fez parte do grupo de coordenação do SPIN-Campinas e já realizou apresentações na Trilha da Indústria do Congresso Brasileiro de Software (CBSOFT) em 2013 e 2014.

4. Referências

Costa, C; Rocha R. G. C.; Silva F. Q. B. ; Prikladnicki, R. (2010) “Desafios e Boas Práticas para o Gerenciamento de Projetos no Desenvolvimento Distribuído de Software”. In: IV Workshop de Desenvolvimento Distribuído de Software (CBSOFT - SBES), 2010, Salvador - BA. Anais do IV Workshop de Desenvolvimento Distribuído de Software.

Costa, J. B. da (2014) “Um mapeamento sistemático de gerenciamento de projetos no desenvolvimento distribuído de software” Disponível em: <http://repositorio.ufpe.br/bitstream/handle/123456789/11555/DISSERTA%C3%87%C3%83O%20Juliana%20Braz%20da%20Costa.pdf?sequence=1&isAllowed=y> Acesso em: 4 de julho de 2016.

Brasil (1991) “Lei 8.248/91, de 23 de outubro de 1991”. Publicada no Diário Oficial da União em 24 de outubro de 1991. Disponível em: http://www.planalto.gov.br/ccivil_03/leis/L8248.htm. Acesso em: 20 de junho de 2016.

Brasil (2001) “Lei 10.176/01, de 11 de janeiro de 2001”. Publicada no Diário Oficial da União em 12 de janeiro de 2001. Disponível em: http://www.planalto.gov.br/ccivil_03/leis/LEIS_2001/L10176.htm. Acesso em: 20 de junho de 2016.

Brasil (2004) “Lei 11.077/04, de 30 de dezembro de 2004”. Publicada no Diário Oficial da União em 31 de dezembro de 2004; retificada no Diário Oficial da União em 14 de janeiro de 2005 e 16 de fevereiro de 2005. Disponível em:

http://www.planalto.gov.br/ccivil_03/_Ato2004-2006/2004/Lei/L11077.htm. Acesso em: 20 de junho de 2016.

Brasil (2006) “Decreto nº 5.906, de 26 de setembro de 2006”. Publicado pelo Diário Oficial da União em 27 de setembro de 2006. Disponível em:

http://www.planalto.gov.br/ccivil_03/_Ato2004-2006/2006/Decreto/D5906.htm. Acesso em: 23 de junho de 2016.

Brasil (2014) “Lei 13.023/14, de 8 de agosto de 2014”. Publicada no Diário Oficial da União em 11 de agosto de 2014. Disponível em:

http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2014/Lei/L13023.htm. Acesso em: 20 de junho de 2016.

SISCAPMobile(Sistema para Controle Administrativo Penitenciário na versão móvel)

J. A. Nascimento¹, Egydio Ricardo G. Junior²

¹Centro de Estudos Superiores de Maceió(Cesmac)

¹Rua Cônego Machado-918- Farol-Maceió-AL, Cep: 57051-160

¹Thoths- Soluções Inteligentes. CNPJ: 22.047.338/0001-31

²Agente de Segurança Penitenciária

{nascimento, ricardo}job.nascimento@gmail.com, egypcioricardo@gmail.com

Abstract. *One of the public sector, most critics, who have not yet experienced the acquisition of technological tools that help information of management, is the Brazilian prison system. To be carried out an investigation in such part, and with the help of managers of the Brazilian penitentiary system and software developers, SISCAPMobile technological tool, has the main objective practicality in the prison system information management, helping to significantly re-socialization convicted, Brazilian reality not demonstrated in statistical surveys. Given that many prisons example of public jail and prisons does not have computer equipment to such processing and when you treat them simply with the use of spreadsheets without the possibility of precise static surveys and difficult to share. The system was developed and published awaits the position of the entities that were previously linked to the Ministry of Communications.*

Resumo. *Um dos setores públicos, mais críticos, que ainda não experimenta a aquisição de ferramentas tecnológicas que auxiliem a gestão das informações, é o sistema prisional brasileiro. Ao ser realizado uma investigação em tal sistema, e com o auxílio de gestores do sistema penitenciário brasileiro e desenvolvedores de software, a ferramenta tecnológica SISCAPMobile, terá como objetivo principal a praticidade na gestão de informações do sistema prisional, ajudando de forma significativa a ressocialização dos apenados, realidade brasileira não demonstrada em levantamentos estatísticos concisos e substanciais. Tendo em vista que inúmeras unidades prisionais a exemplo de cadeias públicas e presídios não dispõe de equipamento de informática para tal processamento e quando tem as tratam de forma simples com uso de planilhas sem a possibilidade de levantamentos estáticos precisos e de difícil compartilhamento. O sistema foi desenvolvido e para publicação aguarda o posicionamento das entidades que anteriormente eram vinculadas ao Ministério das Comunicações.*

1. Informações Gerais

O SISCAPMobile é uma plataforma móvel que foi desenvolvida, após ganhar um concurso promovido pelo Ministério das Comunicações. O concurso foi o INOVApps, edital nº 54/2015/SEI-MC, vinculado ao ministério das comunicações. Neste concurso foram premiados cerca de 100 projetos, cujo intuito era fazer com que desenvolvedores

no caráter de pessoa física ou jurídica desenvolvessem aplicativos que seriam disponibilizados de forma gratuita por 02 anos a partir data de publicação no guia de aplicativos do governo federal, ou em outros portais vinculados ao governo federal. O aplicativo ganhou na área de segurança pública. E poderá ser utilizado em todas as unidades prisionais do país, desde que os agentes penitenciários, dentre outros correlacionados ao setor prisional, dispuserem de dispositivos móveis e internet móvel.

O aplicativo tem o intuito de facilitar o gerenciamento de informações nas unidades prisionais e até em delegacias do Brasil, podendo com estas informações compor um banco de dados abrangente com acesso aos órgãos administradores para estatísticas proativas de melhoramento do sistema prisional, além de garantir, no caso das unidades penais, a segurança no acesso das unidades prisionais. Uma vez que as delegacias podem ter o controle dos detidos que circulam pelas suas instituições.

Um dos grandes atrativos da criação deste aplicativo é a identificação imediata do apenado, uma vez que no momento do cadastro, são inseridas informações à respeito, do pavilhão, piso, bloco e cela, em que o recluso será recolhido. Através do conhecimento dessas informações, é possível descobrir rapidamente onde o recluso se encontra, bem como o tipo de crime cometido pelo apenado. O acesso a tais informações facilitaria os órgãos de fiscalização que tem interesse em realizar uma ressocialização e segregação no sistema penitenciário brasileiro, para evitar problemas como, apenados que cometeram crimes de alto grau de periculosidade, na mesma cela daqueles que cometeram infrações leves. Isso é plenamente realizável, pois pode-se observar a quantidade de apenados em cada cela, e a natureza das infrações que cada recluso cometeu, permitindo assim separar os mesmos por categoria de crime.

O aplicativo possui como diferenciais o cadastro de reclusos com captura de fotos e grau de periculosidade, além de dados básicos relativos ao recluso, a exemplo de nome e alcunha, para inclusão em prontuário. Bem como o cadastro de visitantes com captura de foto e inserção de grau de parentesco com o recluso, além de emissão de carteira de visita com QR Code para facilitação do trâmite nos dias de visita dos familiares e/ou companheira, tornando ágil a visualização de permissão de entrada, contribuindo para o bom fluxo de visitantes em dias de visita. Os dados inseridos podem ser enviados através de e-mail para controle da secretaria vinculada a unidade prisional. Este aplicativo visa melhorar a gestão de informação por parte do agente público, servindo ainda de base para os agentes públicos vinculados a polícia executiva ou mesmo ostensiva.

O aplicativo contém como características principais o cadastro de informações como:

- ✓ Informações pessoais: nome, nome da mãe, data de nascimento, grau de periculosidade, alcunha, crime, sexo, data de cadastro e o local em que o mesmo ficará detido;
- ✓ Foto de frente e perfil do recluso: Essas fotos tiradas com a câmera do *smartphone* do agente público;
- ✓ Informações dos visitantes: nome, vínculo do visitante com o recluso, CPF frente, situação(ativo/inativo);
- ✓ Foto de frente do visitante com a câmera do *smartphone* do agente público.
- ✓ Impressão da carteira de visitação e checagem via QR Code do status do visitante, se o mesmo está habilitado para visitação (ativo) ou não habilitado (inativo).

O aplicativo foi desenvolvido na plataforma *Android Studio 1.3.1*. como IDE de desenvolvimento e tem como intuito de ser disseminado em todo o território nacional nas instituições prisionais. E seguindo as etapas clássicas para desenvolvimento de software[Rafael, 2014],[Wilson, 2009].

2. Sumário

A apresentação da palestra sobre a ferramenta poderia ser subdivida nos seguintes tópicos:

- ✓ **Realidade do sistema prisional:** A idéia é neste tópico enfatizar a escassez de ferramentas de gestão no setor prisional, e como esse fato diminui a eficiência na gestão de informações, bem como mostrar a importância de se utilizar a tecnologia da informação para agregar valor;
- ✓ **Tecnologia agregada para o *SISCAPMobile*:** Neste tópico pretende-se discorrer de como foi construída a ferramenta de uma forma a que se perceba a grande sincronia a coleta dos requisitos e no desenvolvimento do software visando resolver os principais gargalos dos presídios em geral;
- ✓ **Principais interfaces com o usuário do *SISCAPMobile*:** No item serão descritas as principais telas do aplicativo, bem como suas funcionalidades.
- ✓ **Otimização das tarefas dos agentes penitenciários:** Neste tópico serão discutidos como o aplicativo pode otimizar as atividades cotidianas realizadas pelos agentes penitenciários;
- ✓ **Parcerias a serem estabelecidas:** No final da palestra serão discutidas como podem ser realizadas as parcerias entre desenvolvedores, iniciativa privada e pública desenvolvimento de aplicativos e/ou plataformas desktop para a área de segurança pública no Brasil.

3. Audiência

O principal público alvo, da palestra, seriam gestores públicos, principalmente ligados ao setor de segurança pública, desde; polícia militar, civil, federal. Demais setores correlacionados, como a *OAB* e *INSS*. Assim como desenvolvedores mais e menos experientes, visando a formação de parcerias com empresas e instituições de ensino para desenvolvimento de projetos ligados ao setor prisional. Não há a necessidade de pré-requisitos para o entendimento da palestra, tendo em vista que o tema é uma problemática atual e crescente, a reestruturação do sistema prisional, e como agregar a tecnologia da informação, para a melhor gestão das informações.

4. Referência

Rafael Prikladnicki; Renato Willi; Fabiano Milani; Colaboradores, “Métodos Ágeis para Desenvolvimento de Software”, bookman, 2014.

Wilson de Pádua Filho, Engenharia de Software Fundamentos, métodos e padrões LTC, 3ª ed., 2009.

5. Biografia

O autor¹ possui experiência na área de Engenharia Elétrica, com ênfase em Telecomunicações, Eletrotécnica atuando principalmente nos seguintes temas: Eletromagnetismo e Microondas Aplicados, Eletrotécnica, Eletrônica e Programação

aplicada utilizando Linguagem *C* e *Python*. E gerente de projetos na empresa de desenvolvimento de software THOTHS, atuando no desenvolvimento de plataformas voltadas para órgãos públicos e setores comerciais. E atuou como o coordenador do projeto *SISCAPMobile*.

Para acesso tem o link de acesso ao currículo *lattes*:

<http://buscatextual.cnpq.br/buscatextual/visualizacv.do?id=K4550453A6>

Link da empresa que desenvolveu o aplicativo

<http://www.thoths.com.br/>

Palestras correlacionadas ao tema abordado neste artigo:

12ª FETECH 2014:

Tema: “*Tecnologia e a lei das execuções penais: a eficácia na aplicabilidade*”

Expotec 2015:

Tema: “*Inovações tecnológicas paraibanas são destaque para a reestruturação do sistema penitenciário*”

Cbsoft 2015(Trilha da Indústria)

Tema: “**SISCAP®(Sistema para Controle Administrativo Penitenciário)**”

INFOBRASIL 2015

Tema: “**Tecnologia e a lei das execuções penais: a eficácia na aplicabilidade com a ferramenta SISCAP®**”

SPIN-Campinas promovendo a troca de experiências entre empresas e academia.

Fernanda Kussama Pellegrini

fkpellegrini@gmail.com

***Abstract:** One of the goals of the Industry Track of Brazilian Conference on Software is to establish cooperation between professionals and companies, as well as encourage interaction between academia and industry [CBSOFT, 2016]. The SPIN-Campinas (Software and System Process Improvement Network) has worked to achieve these same goals in Campinas-SP region. Thus, this presentation discusses how it has been done, the difficulties and challenges related to these activities.*

***Resumo:** Um dos objetivos da Trilha da Indústria do Congresso Brasileiro de Software é estabelecer cooperações entre profissionais e empresas, bem como estimular a interação entre academia e indústria [CBSOFT, 2016]. O SPIN-Campinas (Software and System Process Improvement Network) tem atuado para alcançar estes mesmos objetivos na região de Campinas-SP. Assim, esta apresentação aborda como isto tem sido feito, as dificuldades e desafios relacionados a estas atividades.*

1. Introdução

As empresas de desenvolvimento de software e tecnologia da informação vivem constantes mudanças devido ao surgimento cada vez mais rápido de novas tecnologias, ferramentas e metodologias. Estas mudanças consideram não apenas aspectos técnicos, mas também gerenciais e administrativos, que afetam diretamente a forma de trabalho dos profissionais e empresas destas áreas.

Para que os profissionais mantenham sua empregabilidade em alta e as empresas continuem competitivas num mercado cada vez mais acirrado e globalizado, todos devem se manter atualizados em relação às novas tendências, seja conhecendo novas práticas e formas de trabalho, seja adotando-as no dia a dia.

Entretanto, essa tarefa está longe de ser considerada simples. Muitas vezes o foco da atualização se volta para um determinado aspecto - por exemplo, uma nova tecnologia - enquanto outros aspectos também importantes são deixados de lado. É bastante comum quando se concentra em metodologias de desenvolvimento de software, mas não se atenta em acompanhar o surgimento de novas tecnologias; ou mantém-se o foco em aspectos técnicos, e se deixa de lado os aspectos gerenciais; ou ainda priorizam-se os projetos, mas não as pessoas envolvidas.

Outro fator importante a ser considerado é que cursos nem sempre mostram o lado prático das disciplinas abordadas, fazendo com que os alunos obtenham pouca informação sobre como aplicar os conceitos aprendidos em sua rotina de trabalho. Mais restrito ainda, é a informação de aplicação prática na indústria brasileira e em ambientes parecidos aos que os estudantes vivenciam.

Além disso, a grande quantidade de informação disponível principalmente na internet pode se tornar prejudicial ao estudante ao invés de aliada. O trabalho de filtrar eventos, publicações e até mesmo cursos de boa qualidade, não é tarefa fácil, principalmente quando a informação é gratuita. Por outro lado, muitos cursos e publicações são caros, fazendo com que nem todos os profissionais, e até mesmo empresas, tenham acesso a eles.

Neste contexto é que o SPIN-Campinas [SPIN-Campinas, 2013] atua, proporcionando eventos presenciais, gratuitos e de qualidade sobre temas relacionados e não limitados a processos, engenharia; desenvolvimento de sistemas e software, gestão de projetos e gestão de pessoas. Atua especialmente na região de Campinas-SP propiciando, principalmente, a troca de experiências entre profissionais, empresas, governo e instituições de ensino da região em que atua.

2. Apresentação

Esta apresentação explica a missão dos SPINs, como o SPIN-Campinas se organiza e funciona, além de seus benefícios e desafios. Os principais tópicos da apresentação são:

O que é o SPIN? - SPIN é uma rede composta por indivíduos de uma mesma região geográfica interessados em promover o aperfeiçoamento das práticas de Engenharia de Software e Sistemas e assuntos correlatos. Inicialmente proposto pelo SEI (*Software Engineering Institute*), há mais de 100 SPINs ativos ao redor do mundo atualmente. Os SPINs são formados por membros de qualquer setor, seja da academia, indústria ou governo e cada um tem sua própria forma de funcionamento e organização dependendo da visão dos fundadores e interesses da comunidade [SEI, 2016].

Campinas e região - A região de Campinas é denominada como "O Vale do Silício brasileiro" por ser reconhecida como o maior polo tecnológico da América Latina. Tal reconhecimento se dá, pois conta com 15 centros de pesquisa e desenvolvimento, 18 instituições de Educação Superior, com destaque para a Universidade Estadual de Campinas (Unicamp), cinco parques tecnológicos [Aquim, 2015] e 32 unidades das 500 maiores empresas do mundo, como por exemplo, Samsung, HP, Compaq e IBM [Wikipedia, 2016]. Consequentemente a região de Campinas é responsável por 15% da produção de tecnologia do Brasil. A cidade de Campinas conta ainda com incentivos fiscais concedidos pela prefeitura para startups (empresas jovens e inovadoras) e aceleradoras, que fornecem treinamentos sobre inovação e gestão, e, principalmente, investimentos em novos negócios. A região ainda é beneficiada pela sua localização geográfica, malha viária (interligada por cinco das grandes rodovias do Estado de São Paulo: Anhanguera, Bandeirantes, D. Pedro I, Adhemar de Barros e Santos Dumont), e pelo Aeroporto de Viracopos, tornando-a de fácil acesso para pessoas e empresas de grandes centros até mesmo de fora do Brasil [Brito, 2015].

SPIN-Campinas e Histórico - O SPIN-Campinas foi fundado em 2001 por representantes da Motorola, CPqD e Cenpra, em uma época em que muito se discutia sobre as práticas relacionadas ao modelo CMM para indústria de software. Havia um grande interesse da comunidade, tanto acadêmica quanto da indústria, sobre como implantar este modelo nas empresas brasileiras. Desde então, mais de 65 eventos foram realizados. No entanto, com o passar do tempo, o interesse da comunidade mudou e outros temas passaram a ser abordados, englobando desde assuntos bastante técnicos até

gerenciais. O grupo também se abriu para temas não necessariamente relacionados a software. Como exemplos, podemos citar: Seis Sigma; Design Thinking; Métodos Ágeis; Ferramentas da Qualidade; Técnicas e Ferramentas de Testes; e Gestão de Pessoas. Nestes 15 anos de existência do SPIN-Campinas, várias empresas se revezaram no grupo de coordenação. Além das empresas fundadoras do grupo, citadas anteriormente, fizeram ou fazem parte da coordenação: SIDI, Venturus, Instituto Eldorado, ASR, IMA, Bosch, SAVIS e Matera. Muitas outras empresas contribuíram com o grupo, cedendo auditórios para os eventos, patrocinando *coffee-breaks* e principalmente relatando suas experiências. Algumas poucas delas são: Sofist, CI&T, Dextra, Opusphere, Baita, e Adaptworks.

Forma de trabalho - O SPIN-Campinas sempre atuou de forma voluntária tanto por parte do grupo de coordenação, quanto por parte dos palestrantes e empresas que sediam e patrocinam os eventos; não possui propósito comercial e nem recursos financeiros para retribuir os palestrantes. A contrapartida aos voluntários é o networking com mais de 500 inscritos no grupo. O grupo de coordenação do SPIN-Campinas trabalha na definição dos temas e palestras (em geral, três por evento), busca por palestrantes, busca por empresas ou instituições que sediem os eventos e patrocinem os *coffee-breaks*. Divulgação dos eventos, inscrições, contabilização de participantes, abertura, acompanhamento e avaliação dos eventos, pesquisa de temas de interesse da comunidade, organização das salas/auditórios onde os eventos também são tarefas realizadas pelo grupo de coordenação. Palestrantes, empresas ou instituições patrocinadoras têm espaço para divulgar seus trabalhos, distribuir brindes e folders, se assim desejarem, mas enfatiza-se aos palestrantes que propagandas e autopromoção devem ser evitadas ao longo das apresentações para que o propósito da iniciativa não seja desvirtuado. Até então, o grupo de coordenação evita o recebimento de patrocínios em dinheiro nem organiza eventos pagos, pois pelo fato de não ser uma entidade jurídica não pode emitir recibos nem notas fiscais. Esta opção foi definida para evitar problemas de conflito de interesse com as empresas em que os representantes do grupo de coordenação trabalham.

Benefícios, dificuldades e desafios - O SPIN-Campinas traz uma série de benefícios para comunidade. Dentre elas destacam-se: palestras gratuitas e de qualidade; oportunidade de troca de experiências entre empresas; oportunidade de benchmarking; networking. Muitas vezes os eventos são oportunidades que os profissionais e empresas têm de se conhecerem para então estabelecerem parcerias e contatos de longo prazo, formais ou não. Dentre as dificuldades e desafios enfrentados pelo grupo de coordenação, estão:

- a) Devido ao fato do grupo não possuir recursos financeiros as atividades realizadas ficam bastante limitadas. Muitas vezes, a comunidade possui interesse em temas que nenhuma empresa da região adotou, mas trazer palestrantes de regiões distantes é difícil, pois não há recursos para cobrir os custos de viagem e hospedagem dos palestrantes.
- b) O grupo realiza atividades bastante operacionais que demandam grande esforço e concorrem com as atividades de trabalho e pessoais de cada um. Isto limita ainda mais as atividades realizadas pelo grupo de coordenação.

- c) A alta demanda por pessoas com bastante conhecimento nas áreas de atuação do SPIN-Campinas e atentas às mudanças e inovações que ocorrem no mercado. Mais do que isso, o grupo necessita de pessoas com muitos contatos nas empresas e com profissionais para conhecer suas áreas de atuação e inovações para, então, convidá-los a palestrar nos eventos organizados pelo grupo.
- d) Destaca-se também o desafio do grupo de manter e ampliar a participação das empresas, profissionais e estudantes nos eventos do grupo, assim como aumentar a participação acadêmica, principalmente como palestrantes.
- e) Manter o grupo de coordenação motivado, mesmo frente às dificuldades e desafios enfrentados.

3. Audiência

Profissionais da indústria ou academia interessados em conhecer um mecanismo de troca de experiências entre empresas e academia, ou que queiram compartilhar suas experiências num fórum relacionado à Engenharia de Sistemas e Software.

4. Biografia

Fernanda Kussama Pellegrini é Bacharel em Ciência da Computação (1998) pela UFSCar e Especialista em Engenharia de Software (2001) e Engenharia da Qualidade (2006) pela UNICAMP. Iniciou sua carreira profissional como estagiária na IBM, em 1998. Depois disso, trabalhou com desenvolvimento de software, qualidade e melhoria de processos no CPqD, ADC, Motorola e Robert Bosch. Desde setembro de 2012, trabalha no Samsung Instituto para Desenvolvimento da Informática como Coordenadora do Escritório de Projetos, implantando processos de gerenciamento de projetos e suportando os projetos na tomada de decisões. Tem experiência na área de garantia e controle da qualidade, definição e melhoria de processos, auditorias, ISO 9000, TL 9000, ISO/TS 16496, OHSAS 18000, ISO 14001, ISO 19011, CMMI e em metodologias e ferramentas Seis Sigma para desenvolvimento de processos e produtos. Já realizou apresentações na Trilha da Indústria do Congresso Brasileiro de Software em 2013 e 2014 e fez parte do grupo de coordenação do SPIN-Campinas por 10 anos.

5. Referências

CBSOFT (2016) “Chamada de Trabalhos Trilha da Indústria” Disponível em: <http://cbsoft.org:50880/cbsoft2016/trilha-industria//chamada-trilha-da-industria> Acesso em: 10 de agosto de 2016.

Spin-Campinas (2013) “Início” Disponível em: <https://sites.google.com/site/spincps/home> Acesso em: 10 de agosto de 2016.

SEI (2016) “Overview” Disponível em: <http://www.sei.cmu.edu/spin/> Acesso em: 10 de agosto de 2016.

Wikipedia (2016) “Vale do Silício Brasileiro” Disponível em: https://pt.wikipedia.org/wiki/Vale_do_Sil%C3%ADcio_brasileiro Acesso em: 10 de agosto de 2016.

Brito, Sarah (2015) “Campinas é apontada como maior polo de tecnologia da AL” Correio Popular. Disponível em: http://correio.rac.com.br/_conteudo/2015/06/

capa/campinas_e_rmc/278156-campinas-e-apontada-como-maior-polo-de-tecnologia-da-al.html Acesso em: 10 de agosto de 2016.

Aquim, Tatiane (2015) “Campinas: a força do setor de data center do Brasil”, Datacenter Dynamics. Disponível em: <http://www.datacenterdynamics.com.br/focus/archive/2015/06/campinas-for%C3%A7a-do-setor-de-data-center-do-brasil> Acesso em: 10 de agosto de 2016.

Precificação de sistemas com código complexo: um estudo de caso com APF e COSMIC

Nádia Rafaela Costa¹, Francisco José Rêgo Lopes¹, Emerson Mendes Portela Costa²
Ana Luiza Bessa de Paula Barros³, Paulo Henrique Mendes Maia³

¹ Serviço Federal de Processamento de Dados (SERPRO)

² Empresa de Tecnologia e Informações da Previdência (DATAPREV)

³ Universidade Estadual do Ceará (UECE)

{pos.nrafaela, lorespaz, emersonportela}@gmail.com,

analuiza@larces.uece.br, pauloh.maia@uece.br

Abstract. *This work makes a comparison between Function Points Analysis (FPA) and Common Software Measurement International Consortium (COSMIC) - the methods most consistently used today, through a case study on a system that has simple user interface, but complex code. This kind of systems often has low score in APF. From measurements, prices have been calculated using both techniques and compared, and COSMIC has showed better applicability for pricing systems with this characteristics.*

Resumo. *Este trabalho faz uma comparação entre Análise de Pontos de Função (APF) e Common Software Measurement International Consortium (COSMIC) - métodos mais constantemente utilizados atualmente, através de um estudo de caso com um sistema que tem interface de usuário simples, mas com código complexo. Sistemas dessa natureza costumam apresentar baixa contagem em APF. A partir de medições, foram calculados os preços usando ambas as técnicas e comparados entre si, tendo o COSMIC uma melhor aplicabilidade para precificação de sistemas com estas características.*

1. Resumo da Proposta de Apresentação

A crescente competitividade entre as empresas suscita a busca pela máxima eficiência de seus processos e produtos. Desta forma, a medição da produtividade e da qualidade no desenvolvimento e manutenção de *software* tornou-se obrigatória para a promoção da qualidade e para obtenção de destaque de uma instituição diante de seus concorrentes. Assim, a Engenharia de Software amadurece, a medição de sistemas passa a desempenhar um papel cada vez mais importante no entendimento e controle das práticas e produtos utilizados no desenvolvimento de *software*.

Uma questão premente para diversas organizações que desenvolvem sistemas é: qual deve ser o preço do *software*? No Brasil, de acordo com as recomendações do Tribunal de Contas da União para precificação de sistemas com base em métricas que não sejam homem/hora, o Governo contrata sistemas medidos utilizando pontos de função. Embora as empresas pratiquem valores diferentes para o preço do seu ponto de função, esse valor não pode exceder a alguns limites e também não pode ser tão baixo que não consiga

remunerar o esforço empreendido. Fatto (2016) publicou informações sobre licitações realizadas pelo Governo, no período de 2006 a 2012, para contratação de serviços de TI. Seu conteúdo mostra que as médias de preço estimado para um ponto de função variam, mas a amostragem exibida traz teto máximo de R\$ 1.333 por ponto de função.

A experiência com contagens de *software* utilizando Análise de Pontos de Função (APF) mostra que, para alguns tipos de sistemas com características específicas, tais como código complexo e baixo nível de interface com usuário, a APF não representa corretamente, em sua pontuação, a devida tradução do esforço empregado no desenvolvimento. Isso se dá em virtude das características que definem a especificação da APF, a qual analisa o *software* sob a perspectiva do usuário. Desta forma, por exemplo, uma página com a pesquisa padrão do Google, que exibe uma tela para que o usuário digite o texto a ser pesquisado e um botão de pesquisa, contada em APF terá, no máximo, apenas 6 pontos de função. Isso, obviamente, está longe de representar toda a complexidade envolvida no mecanismo implementado para que ela funcione. Utilizando o valor de R\$ 1.333 por ponto de função, a página inicial do Google teria seu valor precificado em apenas R\$ 7.998, o que claramente não condiz com o nível de complexidade do algoritmo de pesquisa implementado pela empresa.

Outras técnicas para medição do *software* têm surgido como alternativa à utilização de APF. Uma das que mais adquiriu notoriedade é Common Software Measurement International Consortium (COSMIC), que tem seu processo de medição baseado na observação das movimentações de dados, no qual cada movimento equivale a um ponto de função COSMIC (PFC). Neste processo, executam-se as fases de estratégia de medição, mapeamento e fase de medição. Na primeira fase, o medidor deve acordar e documentar com os patrocinadores da medição o escopo e as características da medição. Na segunda, é realizada a instanciação do modelo genérico de software a ser medido. Já é na terceira fase que, efetivamente, a contagem ocorre.

Nossa hipótese é que COSMIC é uma métrica mais adequada do que APF para precificação de um sistema que possui um código com alta complexidade, ou seja, com número significativo de métodos que realizam operações não triviais e que não produzem um volume significativo de elementos de interface com o usuário. Como forma de comprovar essa hipótese, pelo menos inicialmente, foi realizado um estudo comparativo entre as duas técnicas de medição para um sistema de controle de fluxo de caixa que visa, através de programação matemática utilizando a plataforma Java, determinar a aplicação ótima do saldo de caixa. O estudo considerou as quantidades de pontos obtidos em cada abordagem, o esforço empreendido no desenvolvimento do sistema e o valor do preço médio de um ponto de função.

Os resultados mostraram que com COSMIC foi possível contar quase quatro vezes mais pontos do que com APF, o que refletiu numa precificação do sistema três vezes mais cara, contudo, mais realista e justa.

Os trabalhos de Buglione (2008), Desharnais (2008), Cuadrado-Gallego (2008) e Ho (1999) serviram como referência para o estudo comparativo entre as técnicas APF e COSMIC. Outrossim, os conceitos descritos em Sommerville (2013), Xunmei (2006), IFPUG (2010), Abran (2007) e Fatto (2016) foram utilizados neste trabalho para melhor definir as características dessas técnicas.

Neste contexto, a proposta desta palestra é apresentar, de forma sucinta e consolidada, as medições feitas em APF e COSMIC de um *software* com características de alta complexidade. Pretende-se exibir as vantagens e desvantagens das técnicas de medição abordadas, assim como o comparativo entre elas. A apresentação será organizada da seguinte forma:

- *Introdução à Medição e Estimativa de Software*: breve definição sobre medição e estimativa de *software* apenas para contextualizar a audiência.
- *Técnicas de Medição*: breve explicitação acerca das técnicas de medição APF e COSMIC.
- *Descrição do Sistema*: breve descrição do sistema utilizado como estudo de caso.
- *Resultados das Medições*: explanação e comparação dos resultados obtidos entre as medições com as técnicas APF e COSMIC.
- *Considerações Finais*: breve conclusão sobre melhor cenário para utilização de cada técnica.

2. Audiência

Esta palestra tem como audiência qualquer profissional da área de desenvolvimento de *software* que se interesse em conhecer métodos de contagem e precificação de *software*, em especial a gerentes de projetos, analistas de requisitos, analistas de qualidade e professores da área.

Sugere-se que os participantes da palestra tenham conhecimento básico sobre as etapas do ciclo de desenvolvimento de um *software*.

3. Breve Biografia dos Co-Autores da Palestra

Nádia Rafaela Costa é Bacharel em Informática pela Universidade de Fortaleza, Pós Graduada em Engenharia de Requisitos pela Universidade Federal do Rio Grande do Sul e Pós Graduada em Estratégia e Gestão Empresarial pela Universidade Federal do Ceará. Possui certificação *Certified Function Point Specialist* no IFPUG. Atua como analista de pontos de função no Serviço Federal de Processamento de Dados - SERPRO, exercendo a função de coordenação do escritório de métricas da regional Fortaleza.

Francisco José Rêgo Lopes é Especialista em Engenharia de Software pela UFRGS e Bacharel em Ciências da Computação pela UFCG. Atua como analista de sistemas no Serviço Federal de Processamento de Dados - SERPRO, exercendo a função de líder de projeto.

Emerson Mendes Portela Costa é Bacharel em Sistemas de Informação pela Unichristus. Atua como analista de sistemas na Empresa de Tecnologia e Informações da Previdência Social - DATAPREV. Possui experiência como gestor, analista e desenvolvedor em outras empresas.

Ana Luiza Bessa de Paula Barros é Bacharel em Informática pela Universidade de Fortaleza, Mestre em Ciência da Computação pela Universidade Federal de Minas Gerais, e Doutora em Teleinformática pela Universidade Federal do Ceará. Atualmente é professora adjunta da Universidade Estadual do Ceará e pesquisadora do Programa de Pós-graduação em Administração dessa mesma universidade, tendo como principais áreas de pesquisa Gestão da Inovação e Engenharia de Software.

Paulo Henrique Mendes Maia é Bacharel e Mestre em Ciência da Computação pela Universidade Federal do Ceará e PhD em Computação pelo Imperial College London. Atualmente é professor adjunto da Universidade Estadual do Ceará e pesquisador do Mestrado Acadêmico em Ciência da Computação dessa mesma universidade, tendo como principal área de pesquisa Engenharia de Software.

Referências

- Abran, A., Desharnais, J.-M., Oligny, S., St-Pierre, D., and Symons, C. (2007). O método cosmic para medição de tamanho funcional. Disponível em: http://cosmic-sizing.org/wp-content/uploads/2015/04/Method_Overview_Portuguese.pdf. Versão 3.0.
- Buglione, L., Cuadrado-Gallego, J. J., and De Mesa, J. A. G. (2008). Project sizing and estimating: A case study using PSU, IFPUG and COSMIC. In *Software Process and Product Measurement*, pages 1–16. Springer.
- Cuadrado-Gallego, J. J., Buglione, L., Rejas-Muslera, R. J., and Machado-Piriz, F. (2008). Ifpug-cosmic statistical conversion. In *2008 34th Euromicro Conference Software Engineering and Advanced Applications*, pages 427–432. IEEE.
- Desharnais, J.-M., Abran, A., and Cuadrado, J. (2006). Convertibility of function points to cosmic-ffp: identification and analysis of functional outliers. *ENSUR A*, page 190.
- FATTO. Editais de serviços de software por ponto de função. Disponível em: <http://fattocs.com/pt/recursos/editais>. Acesso em: 30 jun. 2016.
- Ho, V. T., Abran, A., and Fetcke, T. (1999). A comparative study case of COSMIC-FFP, full function point and IFPUG methods. *Department of Informatics, University of Quebec at Montreal, Canada*.
- IFPUG (2010). Manual de práticas de contagem de pontos de função. Versão 4.3.1.
- Sommerville, I. (2013). Engenharia de software. pages 465–472.
- Xunmei, G., Guoxin, S., and Hong, Z. (2006). The comparison between FPA and COSMIC-FFP. In *Proceedings of Software Measurement European Forum (SMEF) Conference*, pages 113–114. Citeseer.

Projeto de um ambiente de desenvolvimento para métodos ágeis

Marcelo L. Ribeiro¹, Itana M. de S. Gimenes¹

¹Departamento de Informatica – Universidade Estadual de Maringá (UEM)
Caixa Postal 5790 – 87020-900 – Maringá – PR – Brasil

mlessaribeiro@gmail.com, itana@din.uem.br

Abstract. *Traditional software development methods were failing to deliver the final product to customers. Such failures generated negative economic and human impact, so it became necessary to find a new way to developing software. From this scenario agile methods emerged. These methods prescribe that human values and the way people work together are the key factors of the software development process. Another change occurred in the software construction with the emergence of new techniques to conceive, design, develop, test, deploy and evolve. To adapt to these changes the software industry has developed new approaches and tools, so it is desirable that we conduct exploratory studies combining the experiences of industry and academia. One of the difficulties with agile methods is to articulate the management methods to the data practices and tools in order to design a productive development environment. Thus, this presentation aims to expose a software development environment that articulates the use of Scrum as a management method and the techniques and tools to support software development.*

Resumo. *Os métodos de desenvolvimento de software tradicionais estavam falhando ao entregar o produto final aos clientes. Tais falhas geravam impacto econômico e humano negativo. Portanto tornou-se necessário encontrar uma nova maneira de desenvolver software. A partir deste cenário surgiram os métodos ágeis, que promovem a cooperação entre fornecedor e cliente durante a construção do software. Essa cooperação alterou drasticamente as técnicas para conceber, projetar, desenvolver, testar, implantar e evoluir o software. Para adequar-se a essas mudanças a indústria desenvolveu novas abordagens e ferramentas, por isso se fazem necessários estudos exploratórios combinando as experiências na indústria e na academia. Assim esta apresentação tem como objetivo expor a proposta de um ADS que articula o uso do Scrum como método de gerenciamento e as técnicas e ferramentas de apoio ao desenvolvimento de software promovendo a integração entre as ferramentas.*

1. Introdução

O uso de técnicas, ferramentas e processos de desenvolvimento de software surgiu no final dos anos 60 quando foi reconhecida a necessidade de estratégias rigorosas que pudessem ser replicadas. Neste contexto, surgiram vários modelos de processos, como por exemplo: Cascata, Espiral, Incremental e Prototipação [Bassi 2008]. Esses modelos definem etapas para a geração de artefatos desde o início até o fim de um projeto que produz um software executável. Exemplos de artefatos são: o modelo de casos de uso, o documento de requisitos, a arquitetura de software, os artefatos de teste, os planos de trabalho, as especificações, o manual de usuário e os relatórios. Esses processos, em geral, eram complexos e rígidos, portanto sua introdução em situações reais se tornou difícil. Observou-se que, mesmo com o uso de técnicas de engenharia de software, a entrega do produto final aos clientes falhava. Essas falhas geravam um impacto econômico e humano negativo, portanto tornou-se necessário encontrar uma nova maneira de desenvolver *software* [Beck 2004].

Neste cenário, surgiram as primeiras iniciativas de métodos ágeis (MA) as quais resultaram no Manifesto Ágil [Williams e Cockburn 2003]. As principais razões para adoção das práticas ágeis são: acelerar a entrega do produto, melhorar a capacidade de gerenciar mudanças de prioridades, aumentar a produtividade e melhorar a visibilidade do projeto.

Empresas de software precisam atingir uma alta agilidade e qualidade no desenvolvimento de projetos de software. No entanto, aplicar apenas os princípios de MA não é o suficiente para que os projetos possam ser desenvolvidos com agilidade [Beck 2004]. Também é imprescindível a utilização de ferramentas apropriadas, a fim de definir um gerenciamento ágil de projetos e impô-las aos participantes [Franky, 2011]. Baseado na literatura existente, Gandomani et al. (2013) dividiu os desafios na aplicação dos MA em quatro categorias: 1) organização e gestão 2) pessoas 3) processo e 4) ferramentas. O último desafio reflete que o uso de ferramentas não-flexíveis é uma barreira para adoção de MA. As empresas devem buscar ferramentas que podem apoiar a evolução incremental para ajudá-las a superar esta barreira e na manutenibilidade dos MA.

A cooperação entre desenvolvedores e clientes durante a construção de software vem alterando drasticamente as técnicas para conceber, projetar, desenvolver, testar, implantar e evoluir o software. Para adequar-se as essas mudanças a indústria desenvolveu novas abordagens e ferramentas. Tais mudanças demandam pesquisas e estudos exploratórios combinando as experiências na indústria e na academia, de modo a dar mais atenção ao que foi desenvolvido pela indústria [Fuggetta e Di Nitto 2014]. No contexto ágil, a relevância dos dados relatados pela indústria e o impacto prático é amplamente reconhecido [Gregory et al. 2015].

Assim, pesquisas são necessárias para reconhecer a importância das práticas de desenvolvimento de software Fuggetta e Di Nitto (2014). Tendo em vista que a configuração de um ADS em projetos ágeis pelas organizações não é trivial, bem como existem diversas opções de ferramentas e combinações entre elas. É importante investigar como as organizações podem escolher a melhor combinação de práticas e ferramentas baseada em suas necessidades [Abbas, Gravell e Wills, 2010].

2. Proposta de apresentação

Esta apresentação tem o objetivo de mostrar como é o Ambiente de Desenvolvimento de Software Baseado no *Scrum* (ADSBS) e como ele pode auxiliar no gerenciamento ágil, através da integração das ferramentas de gestão e desenvolvimento. Também será apresentado um panorama do ponto de vista da indústria, apoiando-se em pesquisas de questionários (*surveys*) disponíveis e que nos permite discutir sobre trabalhos futuros.

Realizou-se um trabalho de análise da literatura científica dos últimos 4 anos sobre o tema de ferramentas em projetos ágeis. A busca por artigos foi conduzida nas bibliotecas digitais da IEEE¹ e ACM², e selecionados segundo critérios pré-estabelecidos. Serão apresentados dados — por exemplo os *surveys* publicados recentemente em [Melo et al. 2012] e [VersionOne 2015] — sobre o desenvolvimento de software em projetos ágeis na indústria.

Por fim, será apresentado o projeto de um ADSBS, como as ferramentas disponíveis no mercado podem ser articuladas para permitir o gerenciamento da construção do software e do projeto. Exemplificando ferramentas de software livre que permitem atingir o conceito de *Application Lifecycle Management* (ALM). Além disso será evidenciado a arquitetura e integrações entre ferramentas que compõem um ADSBS, bem como um processo de desenvolvimento que segue o *Scrum* como base.

3. Agenda

A apresentação abordará os seguintes tópicos sobre ambiente de desenvolvimento de software em projetos ágeis:

- Revisão da literatura científica dos últimos 4 anos;
- Mostrar o ambiente de desenvolvimento de software proposto
 - Arquitetura;
 - Integrações entre o ambiente;
 - Processo de desenvolvimento;
- Discussão e trabalhos futuros.

4. Audiência

A apresentação destina-se a desenvolvedores e gestores – praticantes, ou não, de metodologias ágeis – entusiastas de projetos ágeis, consultores de engenharia de software e a comunidade científica interessada em pesquisas na área de metodologias ágeis.

4.1. Oportunidades

Apresentar o projeto de um ambiente de desenvolvimento que permite o gerenciamento do código e do projeto. Espera-se abrir um diálogo a respeito do ambiente proposto com

¹ IEEE Xplore Digital Library no endereço, o <http://ieeexplore.ieee.org/Xplore/home.jsp>

² ACM Digital Library no endereço, o <http://dl.acm.org>

empresas de desenvolvimento ou consultorias visando produzir conhecimento relevante a respeito da automatização do ambiente de desenvolvimento de software para academia e indústria.

5. Palestrantes

Marcelo Lessa Ribeiro: mestrando no programa de pós-graduação em Ciência da Computação no Departamento de Informática da Universidade Estadual de Maringá (UEM), cujo tema de pesquisa e ambiente de desenvolvimento de software em projetos ágeis. Possui título de bacharel em Informática pela Universidade Estadual de Maringá (UEM). Desenvolvedor de Software da Objective Solutions, tendo trabalhado em projetos ágeis. Possui experiência em configuração e palestras sobre o ferramental de projetos ágeis.

References

- Abbas, N.; Gravell, A. M. and Wills, G. B. (2010) Using Factor Analysis to Generate Clusters of Agile Practices (A Guide for Agile Process Improvement), In: Agile Conference (AGILE), United States of America, p.11-20.
- Bassi, L.B.F. (2008), Experiências com desenvolvimento ágil. Dissertação de Mestrado. São Paulo: IME – USP.
- Beck, K. (2004), Extreme Programming Explained: Embrace Change. Reading. Addison-Wesley.
- Franky, M. C. (2011), Agile Management and Development of Software Projects Based on Collaborative Environments. In: Software Engineering. Notes (SIGSOFT), 36(3):1-6.
- Fuggetta, A.; Di Nitto, E. (2014), Software process. ACM Press, p. 1-12.
- Gandomani, T.J.; Zulzalil, H.; Ghani, A. A. A.; Sultan, A. B. M.; Nafchi, M. Z. (2013), Obstacles in Moving to Agile Software Development Methods: At a Glance. Journal of Computer Science, v. 9, pp. 620-625.
- Gregory, P.; Barroca, L.; Taylor, K.; Salah, D.; Sharp, H. (2015), Agile challenges in practice: a thematic analysis. In: 16th International Conference on Agile Software Development, Finland.
- Melo, C. O., Santos, V. A., Corbucci, H., Katayama, E., Goldman, A., and Kon, F. (2012). Métodos Ágeis no brasil: Estados da prática em times e organizações. Relatório técnico RT-MAC-2012?03, Departamento de Ciência da Computação. IME – USP.
- VersionOne (2015). 10th annual state of agile survey.
- Williams, L.; Cockburn, A. (2003), Agile software development: it's about feedback and change. Computer, 36(6):39-43.

Architecture Analysis for Improvement of Software Quality Produced by SRBR

Marcia C. C. Costa, Gizelle S. Lemos, André Chiarelli

SRBR – Samsung Research Center Brazil
Campinas – SP – Brazil

{m.costa,g.lemos,a.chiarelli}@samsung.com

Abstract. *This paper describes the process of architecture analysis applied on software developed by SRBR (Samsung Research & Development Institute Brazil) and the improvements achieved in the final product due to its performance. During architecture analysis, the SRBR software engineering team has used supportive tools to extract information regarding the architecture from implemented source code, analyzed their outcomes and compared them with what was previous planned to be developed based on software architecture design documentation produced by development team. This practice has been useful to detect architectural problems such as high complexity or tight coupling, incorrect usage of components or APIs, high percentage of code cloning, and differences between architectural design and implementation.*

Resumo. *Este artigo descreve o processo da análise arquitetural aplicada a software desenvolvido pelo SRBR (Samsung Research & Development Institute Brazil) e as melhorias alcançadas no produto final devido a esta análise. Durante a análise arquitetural, o time de engenharia de software no SRBR tem usado ferramentas de suporte para extrair informações sobre a arquitetura a partir do código-fonte, analisado os resultados obtidos e os comparado com o que foi previamente planejado para se desenvolvido baseado na documentação da arquitetura do software gerada pelo time de desenvolvimento. Esta prática tem sido útil para detectar problemas relacionados à alta complexidade ou baixo acoplamento, uso incorreto de componentes ou APIs, alta porcentagem de código clonado e diferenças entre o design arquitetural projetado e a implementação.*

1. Introduction

“The software architecture of a system is the set of structures needed to reason about the system, which comprises software elements, relations among them, and properties of both” [Len Bass, et al 2013]. Software architecture design and documentation is useful in different aspects, the most known are:

- Plan open source components usage regarding their licenses and restrictions;
- Visualize the complexity of the software;
- Enhance the communication among stakeholders;

- Show the constraints and limits of the system;
- Disseminate knowledge about how the system works to a new development team member;
- Allow the visualization and optimization of the system regarding quality attributes.

Although architectural design is important to maintain the quality of the software, it is not valued by project team members and usually the implemented software does not reflect what has been designed.

Architecture analysis is a software engineering practice that rebuilds the software architectural design based on the implementation. It is useful to identify gaps between architectural design and implementation, to analyze lack of architecture information, in order to verify the quality of the attributes of product and if their goals have been satisfied in software design.

After running supportive tools that extract architectural information from the source code, software architects are able to verify different aspects of architectural design analysis:

- Differences between source code and its documented architecture;
- Mapping between requirements and architecture;
- Clones in the source code;
- Cyclical dependences in the source code;
- Incorrect usage of APIs and open source components;
- Problems of cyclomatic complexity, cohesion, coupling;
- Incorrect usage of architectural patterns.

2. Architecture analysis Experience at SRBR

SRBR software engineering team has performed architecture analysis [Muhammad Ali Babar et al 2013] for two years and the results have assisted development teams to improve the quality of Samsung's software systems.

SRBR performs architecture analysis within four steps:

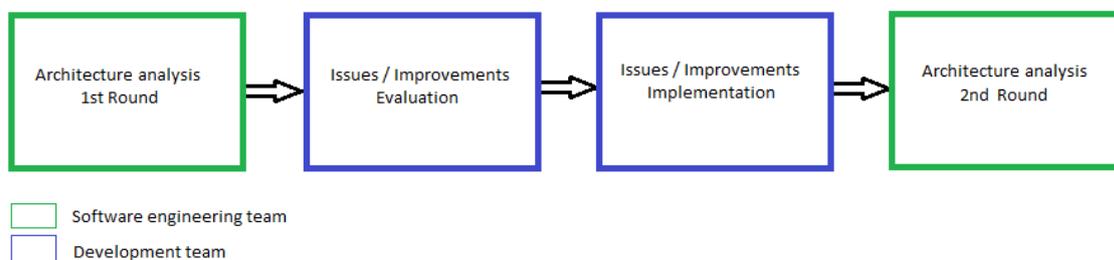


Figure 1. Steps of architecture analysis on SRBR

The SRBR software engineering team, which is detached from development team, is responsible for supporting the development team, conducting software process audits, software process coaching, verifying the quality of software project deliverables

and performing architecture analysis, which may be carried out on three different scenarios considering projects at SRBR:

1. When software source code is transferred from another (external) team to SRBR, architecture analysis shows how the source code is organized. In this case, the objective is to acquire better knowledge on the architectural design besides the documentation that comes with the source code.
2. Whenever SRBR receives the source code implemented by an outsourced partner, architecture analysis helps SRBR team to verify if the delivered source code matches the architectural design documentation and the requirements. In this case, the activity of architecture analysis has been performed to support the acceptance of deliverables.
3. During the implementation of software developed by SRBR development team. In this case, the objective is to verify the results of the architectural design regarding quality attributes, such as maintainability, performance, testability. In the first step of architecture analysis, the SRBR software engineering team defines, within a consensus of the project leader, what are the quality goals to be reached or if the software will be analyzed in a more generic way.

3. Steps and requirements to approach

In the first step, the SRBR software engineering team (SE) collects data about architecture of the implemented source code. Open source tools (such as Nitriq¹ and Atomiq²), commercial tools as Klocwork® Architect³ and Microsoft® Visual Studio⁴ and in-house developed tools have been used as tools of support. After running the tools to extract data from the source code, the SRBR software engineering team performs the analysis to create the “Architecture Analysis First Round Report” in which SE team records improvement opportunities, which may be recommended or mandatory depending on how critically each recommendation affects the goals or software quality.

All improvements raised are presented to the software development team who evaluates all issues and improvement opportunities recorded by the software engineering team. All points are discussed and the project leader defines a schedule of implementation for the development team, who organize themselves within this period to implement the solutions and update the necessary documentation.

After the period assigned to provide solutions to the improvements proposed, SE team rerun all tools and collects the results again comparing them with first step results in order to verify the solutions of improvements and to evaluate how effective these solutions were. While checking effectiveness, the SE team performs a qualitative and quantitative analysis about the general improvement resulting from the architecture analysis documenting it in the “Architecture Analysis Second Round Report”.

¹ Nitriq Code Analysis Tool. Available at: <http://www.nitriq.com/> - Accessed at July, 1st. 2016.

² Atomiq Code Clone Finder Tool. Available at: <http://www.getatomiq.com/> - Accessed at July, 1st. 2016.

³ Klocwork tools: <http://www.klocwork.com/> - Accessed at July, 1st. 2016.

⁴ Microsoft Visual Studio: <https://www.visualstudio.com/pt-br/visual-studio-homepage-vs.aspx> - Accessed at July, 1st. 2016.

The main aspects analyzed during the second round of architecture analysis are:

- Percentage of source code clones [Yoshiki Higo et al 2002] removed;
- Numbers of cycles removed;
- Percentage of improvement regarding architecture metrics.

4. Conclusion

Architecture analysis has been a valuable practice on SRBR projects and its results have shown the importance of a good architecture design, which is directly related to quality of the produced source code. Besides the initiatives of applying architecture analysis on projects, SRBR development teams have improved the quality of architectural design documents during the design phase.

Based on the results of architecture analysis, the SRBR Software Engineering Team is working to improve the maturity of SRBR SW Design Process, through the following initiatives:

- SRBR Software Development Process was reviewed and a defined Architecture Analysis Process was included,
- The SRBR software engineering team has been evaluating new tools for extending the support for this practice,
- Design training sessions and initiatives for sharing best architecture design practices and common issues identified during analysis have been conducted.

5. Audience

Software architects, software developers and software project leaders are the audience of this presentation. Basic knowledge about software architecture is recommended.

References

Len Bass, Paul Clements, and Rick Kazman. (2013). "Software Architecture in Practice". SEI series in Software Engineering. Addison-Wesley, 3rd edition.

Muhammad Ali Babar, Alan W. Brown, Kai Koskimies and Ivan Mistrik. (2013). "Agile Software Architecture: Aligning Agile Processes and Software Architectures". Elsevier Inc

Yoshiki Higo, Yasushi Ueda, Toshihiro Kamiya, Shinji Kusumoto, and Katsuro Inoue (2002). "On Software Maintenance Process Improvement Based on Code Clone Analysis. Volume 2559 of the series Lecture Notes in Computer Science pp. 185-197

About the authors

Marcia Cristina de Carvalho Costa – Samsung R&D – SW Engineering and Project Support Operations Manager, with 30 years of experience in software engineering, assuring the quality of software process and product in research and development organizations such as MCT/CTI, CPqD, Motorola, Ci&T and Samsung. - [linkedin.com/in/Márcia-costa-a6a887](https://www.linkedin.com/in/Márcia-costa-a6a887)

Gizelle Sandrini Lemos – PhD in Computer Science at Universidade Estadual de Campinas. Works as software engineering researcher since 2002. Nowadays is a member of SRBR software engineering team working as software quality specialist. – <http://lattes.cnpq.br/7389890812265291>

André Chiarelli - MSc in Computer Science at National Institute for Space Research. Nowadays is a member of SRBR software engineering team working as software quality analyst <http://lattes.cnpq.br/5002279890831176>

Análise Qualitativa Entre os Processos da Instrução Normativa IN/SLTI/MPOG 04/2014 e o Modelo CMMI-ACQ

Luiz Sérgio P Silva¹, Renata T. Moreira¹, Alexandre M. L. Vasconcelos¹,
Suzana C.B. Sampaio¹

¹ Centro de Informática (CIn)
Universidade Federal de Pernambuco (UFPE)
Caixa Postal 7851- Recife - PE - Brasil - CEP 50732-970.

{lsps, rtm, amlv, scbs2}@cin.ufpe.br

***Resumo.** Iniciativas têm surgido para melhorar os processos de software e serviços nos últimos anos. Apesar do crescimento das iniciativas no Brasil, quando refere-se ao Processo de Contratação de Soluções de Tecnologia da Informação (TI) para a Administração Pública Federal (APF), a aplicação das melhores práticas em organizações brasileiras encontra vários obstáculos. Entre eles destacam-se: a complexidade dos processos e a supervisão contínua dos órgãos de controle. Para minimizar esses obstáculos, o Tribunal de Contas da União (TCU), recomendou a elaboração da Instrução Normativa IN/SLTI/MPOG 04/2014, que contém diretrizes para o processo de contratação de soluções de TI. Este artigo apresenta o mapeamento entre o GCSTI e o Modelo CMMI-ACQ, a fim de identificar a maturidade e a aderência entre seus processos.*

1. Introdução

A Administração Pública Federal Brasileira (APF), atualmente é a maior compradora e consumidora de produtos e serviços de TI no Brasil (SLTI, 2014). Diversas ações têm sido realizadas pelos órgãos que compõem a APF para melhorar o desempenho do governo em relação ao gerenciamento das aquisições de TI, que tem demonstrado diversas dificuldades, entre elas, destaca-se o fato da legislação brasileira ser bastante complexa para garantir um processo eficiente de aquisição (TCU, 2010).

Com base nos gastos realizados pela APF, é inegável a complexidade dos processos de contratação que gerenciam a aquisição e prestação de serviços de TI, sendo este o principal desafio para as PMEs (micro e pequenas empresas), que representam 93% do setor no Brasil (ABES, 2015). Neste contexto, o presente estudo visa avaliar o processo de contratação de soluções de TI perante o modelo CMMI-ACQ, a fim de analisar qual é a maturidade exigida pela APF para execução do processo de contratação, que envolve a aderência à IN/SLTI/MPOG/2014 e ao Guia de Contratação de Soluções de TI (GCSTI). Este mapeamento é a base para a análise da adequabilidade do processo de contratação de soluções de TI em relação ao contexto das empresas brasileiras que prestam serviço.

Este trabalho destina-se aos profissionais envolvidos no fornecimento de produtos e

serviços a Administração Pública Federal Brasileira (APF), como gestores e prestadores de serviços, assim como todos os papéis que estão relacionados em editais de aquisição e prestação de serviços de TI à APF. Além disso, é recomendado aos Pesquisadores em que o tema do trabalho está relacionado com Aquisição, Desenvolvimento, Prestação de Serviços, Suporte, dentre outros.

O artigo está estruturado em 6 Seções, incluindo esta Introdução, as Seções 2 e 3 apresentam o referencial teórico; a Seção 4 apresenta as discussões e resultados do mapeamento entre o Guia de Contratação de Soluções de TI e o Modelo CMMI-ACQ, a Seção 5 apresenta as conclusões do trabalho e, a Seção 6 apresenta uma breve biografia do autor.

2. CMMI para Aquisição (ACQ)

O modelo CMMI para aquisição (CMMI-ACQ) fornece uma estrutura para facilitar a implementação de estratégias no processo de aquisição, eliminando barreiras existentes entre as partes interessadas, incluindo fornecedor de serviços, departamentos de negócios, áreas de sistemas, dentre outros (GALLAGHER, *et al.*, 2010). Todas as práticas do Modelo CMMI-ACQ concentram-se nas atividades do ponto de vista de quem faz aquisição. Essas atividades incluem o fornecedor de terceirização, o desenvolvimento e concessão de acordos com fornecedores e a gestão da aquisição, incluindo a aquisição de produtos e serviços (GALLAGHER, *et al.*, 2010).

O CMMI-ACQ integra o corpo de conhecimento essencial para realização de aquisições bem-sucedidas, fornecendo uma série de benefícios e oportunidade para as organizações de aquisição (GALLAGHER, *et al.*, 2010).

3. Instrução Normativa SLTI/MPOG 04/2014 e o Guia de Contratação de Soluções de Tecnologia da Informação (GCSTI)

A estrutura da IN/SLTI/MPOG 04/2014 (SLTI, 2014) provê mecanismos de governança para contratações de serviços e soluções TI de acordo com a estrutura definida na Figura 1:

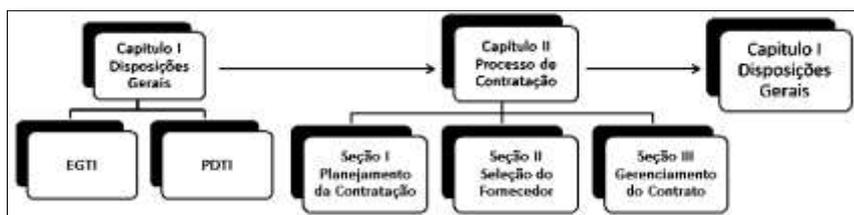


Figura 1 – Instrução Normativa IN/SLTI/MPOG 2014 (TCU, 2014)

As contratações públicas relacionadas às soluções de TI são orientadas pela IN/SLTI/MPOG 04/2014 e o Guia de Contratação de Soluções de Tecnologia da Informação (BRASIL, 2014). O Guia de Contratação de Soluções de TI (GCSTI) é um conjunto de processos, atividades e tarefas para contratação de soluções de TI pela APF, que implementa as definições previstas na IN/SLTI/MPOG 04/2014, através de fases que se desdobram durante todo o processo de contratação de soluções de TI (CAVALCANTI, 2015).

4. Discussões e Resultados do Mapeamento entre o GCSTI e o Modelo CMMI-ACQ

Esta seção apresenta os resultados do mapeamento realizado entre o GCSTI e o Modelo CMMI-ACQ. As fases do mapeamento foram realizadas conforme descrito em (SILVA, 2013). Na Tabela 1 são apresentados os resultados finais consolidados do mapeamento realizado entre o GCSTI e o CMMI-ACQ.

Considerando as Áreas relacionadas à Aquisição, que estão definidas no Nível 3 de Maturidade do Modelo CMMI ACQ, além das demais Áreas do Nível 2 e Áreas Compartilhadas com os outros Modelos CMMI, sendo elas: Desenvolvimento de Requisitos de Aquisição (ARD), Gestão da Aquisição (AM), Validação da Aquisição (AVAL), Solicitação e Desenvolvimento de Acordo de Fornecedor (SSAD) e Gestão Técnica da Aquisição (ATM) obtiveram 100% de cobertura do modelo. Somente a Área de Processo de Verificação da Aquisição (AVER) obteve 87,50%; uma vez que o GCSTI não define um método de revisão por pares como forma de verificação de artefatos do processo de aquisição.

Tabela 1. Resultados do Mapeamento

Áreas de Processos	(%) CMMI ACQ	Áreas de Processos	(%) CMMI ACQ
PP	100%	OT	28,57%
PMC	100%	IPM	70%
CM	85,71%	RSKM	85,71%
PPQA	100%	ARD	100%
MA	68,75%	AM	100%
REQM	80%	AVER	87,50%
OPF	11,11%	AVAL	100%
OPD	50%	SSAD	100%
DAR	100%	ATM	100%

Com base nestes resultados, nota-se que existe uma deficiência na sequência de execução dos processos do GCSTI, em relação ao Modelo CMMI-ACQ. Ao mesmo tempo em que as Áreas de Processos do nível 3 de Maturidade são atendidas em quase 100% e, estão definidas no nível 3 de maturidade, as áreas do nível 2, que definem o nível gerenciado não são atendidas completamente.

Analisando estes resultados perante a indústria de software brasileira, percebe-se que, para atender o GCSTI, as empresas precisam definir processos relacionados à aquisição (Nível 3 de maturidade), demonstrando algum nível de maturidade e capacidade em seus processos.

5. Conclusões

Em termos gerais, não há um alinhamento entre o GCSTI e as metodologias, normas e modelos que comumente são utilizados como padrão de qualidade pela indústria de software e serviços. Dessa forma, as empresas têm um grande desafio de alinharem a realização de suas atividades à forma de trabalho definida pelo governo federal, que está orientado a legislação brasileira e a Lei 8.666/93. Este desafio está relacionado a falta de maturidade e capacidade da indústria brasileira de *software* e serviços, formada em maior parte por empresas de micro e pequeno porte.

O GCSTI define processos e atividades, que em sua maioria, estão equivalentes ao nível 3 de maturidade do Modelo CMMI-ACQ. Neste nível de maturidade as empresas necessitam de processos definidos e especificados.

6. Breve Biografia do Autor

Luiz Sérgio P. Silva é PhD Candidate no Centro de Informática (CIn) da Universidade Federal de Pernambuco (UFPE) na Área de Prestação de Serviços de TI à APF. É Mestre em Ciência da Computação pelo mesmo local. Possui MBA em Marketing (FGV); Graduação em Ciência da Computação (UNIFOR) e Administração (UFC). Possui experiência em Projetos de Melhoria de Processos de Software e Serviços de TI, com a Implantação e Avaliação de Normas e Modelos e, Métodos Ágeis. Maiores detalhes pode ser acessado em: <http://lattes.cnpq.br/0396396403551589>.

Referências

ABES - Associação Brasileira de Empresas de Software. 2015. Disponível em: <<http://www.abes.org.br/>>. Acesso em: 15/06/2016.

BRAGA, R. Auditoria de Governança de TI. Brasília: TCU/ISC, 2009.

CAVALCANTI, S.C. O Novo Modelo de Contratação de Soluções de TI pela Administração Pública Federal. 2a. Ed. Belo Horizonte. Editora Fórum, 2015.

CHRISISS, M. D., KONRAD, M. E SHRUM S. “*CMMI: guidelines for process integration and product improvement*”. Addison-Wesley. 2010.

CRUZ, C. S. da. Governança de TI e Conformidade Legal no Setor Público: Um Quadro Referencial Normativo para a Contratação de Serviços de TI. 2008. Dissertação de Mestrado. Universidade Católica de Brasília, Brasília, 2008. Disponível em: <http://www.btdt.ucb.br/tede/tde_arquivos/3/TDE-2008-11-25T123713Z-687/Publico/TextoCompleto Cruz - 2008.pdf>. Acesso em: 15/06/2016.

FORRESTER, E., BUTEAU, B., SHRUM, S. *CMMI: Guidelines for Superior Service*. Addison-Wesley. 2010.

GALLAGHER, B., PHILLIPS, M., RICHTER, K., SHRUM, S. *CMMI: Guidelines for Improving the Acquisition of Products and Services*. Addison-Wesley. 2010.

ISO/IEC 20000 *Information Technology - Service Management*, Geneve: ISO, 2011.

SILVA, L. S. P. Modelo de Contratação de Soluções de TI: Uma Análise Comparativa para Identificar a Maturidade e a Aderência aos Modelos CMMI-ACQ, CMMI-DEV e CMMI-SVC. 2013. Dissertação (Mestrado em Ciência da Computação) - Centro de Informática – Universidade Federal de Pernambuco, Recife, Pernambuco, 2013.

SLTI, Guia de Boas Práticas de Contratação de Soluções de Tecnologia da Informação. Versão 2.0, Brasília: SLTI, 2014. Disponível em: <<http://cpsustentaveis.planejamento.gov.br/assets/conteudo/uploads/guia-de-boas-praticas-em-contratacao-de-solucoes-de-ti.pdf>>. Acesso em: 15/07/2016.

Implementação do Processo de Gerência de Risco usando o Método Ágil RisAgi para o Alcance do Nível de Maturidade 3 do CMMI-DEV: O Caso da EMREL

Petrônio A. de Medeiros¹, Moisés B. de Leal Junior¹, Homero A. Cavalcanti¹, Ana Cristina Rouiller², Sandro Ronaldo Bezerra Oliveira^{2,3}, Kleoson Bruno Corrêa dos Santos³

¹EMPREL – Empresa Municipal de Informática do Recife – Rua Vinte e Um de Abril, 3370 – Torrões – Recife – PE – Brasil

²SWQuality Consultoria e Sistemas Ltda., Rua do Apolo, 202 – Recife Antigo – Recife – PE – Brasil

³Programa de Pós-Graduação em Ciência da Computação – Universidade Federal do Pará (UFPA), Rua Augusto Corrêa, 1 – Guamá – 66075-110 – Belém – PA – Brasil

{petronioadm, moisesb, homero}recife.pe.gov.br, ana@swquality.com.br, srbo@ufpa.br

Abstract. *This presentation aims to show the strategy for the implementation of the Risk Management process area of CMMI-DEV from the adoption of an agile method to support the software development management, the RisAgi. The RisAgi has not focused on delivering software, but manage the uncertainties of project, aiming at the best possible quality of the final product. Along with the life cycle, this method provides a set of roles of human resources, ceremonies and work products able to support the implementation of an agile risk management.*

Resumo. *Esta apresentação tem por objetivo de mostrar a estratégia definida para a implementação da área de processo de Gerência de Riscos do CMMI-DEV a partir da adoção de um método ágil de apoio à gestão do desenvolvimento de software, o RisAgi. O RisAgi não tem como foco entregar software, mas gerenciar as incertezas dos projetos, visando a melhor qualidade possível do produto final. Juntamente com o ciclo de vida, este método fornece um conjunto de papéis de recursos humanos, cerimônias e produtos de trabalho capazes de apoiar a implementação de uma gestão ágil de riscos.*

1. Introdução

O software, devido a sua própria natureza, é abstrato e intangível [Sommerville 2010], podendo representar a quase inexistência de restrições para o que é produzido. Porém significa também que o software pode se tornar extremamente confuso e difícil de ser mantido. Existe uma boa disposição de desenvolvedores e empresários capazes de atender bem o mercado. Entretanto, um dos fatores que acarretam em produtos de software com níveis de qualidade inadequados é a carência de mão-de-obra especializada. Devido a isso, existem modelos de qualidade que sugerem boas práticas para a melhoria contínua no processo organizacional, como o modelo CMMI-DEV (*Capability Maturity Model Integration for Development*) [SEI 2010]. Empresas desenvolvedoras de software que desejam implementar estes programas de melhoria podem se deparar com o problema de conciliar entre a execução das atividades dos projetos em desenvolvimento com as constantes melhorias que estão sendo implementadas em seu ambiente. Somando-se a isto, existe o fato que o tempo de implementação de programas de melhoria geralmente é extenso [Morgado *et al.* 2007].

Adicionalmente, existe no mercado um número considerável de empresas desenvolvedoras de software que buscam por avaliação de melhoria de processo no modelo CMMI-DEV. Para que estas empresas consigam sucesso nestas avaliações é necessário que seja satisfeito um determinado conjunto de requisitos (práticas/resultados esperados) estabelecidos pelos modelos. Segundo Morgado *et al.* (2007), por esses modelos agregarem um grande número de conceitos em suas estruturas, empresas que desejam receber avaliações de melhoria, geralmente, contratam especialistas (implementadores). Mesmo para esses especialistas, definir a estratégia de como alguns processos contidos nestes modelos devem ser executados não é uma tarefa trivial, uma vez que muitos destes são classificados como organizacionais ao processo de desenvolvimento de software, ou seja, são empregados por uma organização para estabelecer e implementar uma estrutura subjacente, constituída de processos de ciclo de vida e pessoal associados, e melhorar continuamente a estrutura e os processos, sendo tipicamente empregados fora do domínio de projetos e contratos específicos.

Inserida nestes processos organizacionais no modelo CMMI-DEV [SEI 2010] está a área de processo *Risk Management* (Gerência de Risco) constante no Nível de Maturidade 3. Esta área de processo tem o propósito de identificar potenciais problemas antes que eles ocorram para que as atividades de tratamento de risco possam ser planejadas e invocadas quando necessário em toda a vida do produto ou projeto para mitigar os impactos adversos no alcance dos objetivos. A área de gerenciamento de riscos é uma das mais importantes dentro da gestão de projetos, sendo tratada como uma das nove áreas de conhecimento do guia específico de gestão de projetos, o PMBOK (*Project Management Body of Knowledge*) [PMI 2008].

Segundo Cohn (2010), um grande esforço em gestão de riscos de forma explícita torna-se desnecessário quando um projeto utiliza uma abordagem ágil. As iterações curtas, foco em software funcionando, resolução de impedimentos, forte ênfase em testes automatizados, entregas e *feedbacks* frequentes ao cliente ajudam as equipes a evitar o risco que a maioria dos projetos enfrentam, o de não entregar nada. Assim, não é de se estranhar que muitos projetos ágeis renunciem a qualquer forma de gestão de riscos explícita [Cohn 2010]. No entanto, essa ideia não é mais aceita como um consenso geral entre muitos praticantes das metodologias ágeis, principalmente quando projetos de software são complexos, de longa duração e com o escopo grande. Certamente, as incertezas serão maiores. Dessa forma, o uso de uma metodologia ágil para gestão de riscos surge como uma boa opção para as organizações que desejam gerenciar as incertezas dos projetos de forma mais ágil e em conjunto com outra metodologia ágil de gerência de projetos.

O RisAgi (Riscos Ágeis) [Oliveira *et al.* 2013] é um método ágil concebido para possibilitar o gerenciamento das incertezas envolvidas em projetos de desenvolvimento de software. Essas incertezas podem ser entendidas como riscos e/ou impedimentos. O RisAgi apresenta um ciclo de vida definido com base no ciclo de vida dos processos de gestão de riscos do PMBOK. O modo de execução nessas fases segue os valores das metodologias ágeis e se baseia, principalmente, nas práticas encontradas nos métodos ágeis Scrum e APM (*Agile Project Management*) [Highsmith 2004].

2. Proposta da Apresentação

Como solução para minimizar o cenário apresentado, a EMPREL – Empresa Municipal de Informática do Recife (empresa pública, integrante da Administração Indireta Municipal que atua, estrategicamente, no planejamento e implementação da política de Tecnologia da Informação e Comunicação necessárias aos órgãos e entidades da

administração direta e indireta da Prefeitura do Recife) iniciou em 2013 a implementação da meta estratégica “Contratação de Consultoria/Treinamento de Preparação para a Obtenção das Certificações CMMI 3/MPS.BR C”. Como empresa vencedora deste Edital, a SWQuality Consultoria e Sistemas Ltda. (empresa brasileira oferecendo serviços de consultoria, avaliação e capacitação em Gestão de Projetos, Preparação para Certificação em processos de Qualidade em Serviços, Tecnologia da Informação e desenvolvimento de software) tratou de definir as estratégias para a implementação dos processos constantes nos níveis de maturidade destes modelos tidos como meta para a EMPREL. Assim, durante este trabalho, a SWQuality preservou a cultura organizacional de desenvolvimento de software da EMPREL, incluindo novas práticas de maturidade.

Dentro da metodologia de implementação usada pela SWQuality, o RisAgi fornece apoio para todas as práticas constantes na área de processo de gerência de risco do modelo CMMI-DEV. Assim, este método foi instanciado para atender as seguintes demandas desta área de processo: determinar as fontes de riscos e categorias; definir parâmetros de riscos; estabelecer uma estratégia de gerência de riscos; identificar riscos; avaliar, categorizar e priorizar riscos; desenvolver planos de mitigação de riscos; e implementar planos de mitigação de riscos. Como resultado, definiu-se uma Guia de Implementação desta área de processo usando o RisAgi, cujo foco está em orientar a instanciação deste método ágil para atender as demandas especificadas. O guia pode atender principalmente as empresas de desenvolvimento de software na implementação do programa de definição e melhoria dos processos organizacionais na área de software, além da investigação e da disseminação do conhecimento nas práticas propostas pelo CMMI-DEV no contexto da área de processo de gerência de riscos.

Assim, entende-se que a apresentação permitirá a pesquisadores e empreendedores acesso a serviços em Qualidade de Software usando o RisAgi. Para isso, o foco da apresentação é direcionado a alguns segmentos principais: a) adequação das práticas da área de processo de gerência de risco para uso a partir do RisAgi, apresentando lições aprendidas e boas práticas obtidas; b) discussão do Guia para a Implementação desta área de processo usando o RisAgi; c) apresentação do Estudo de Caso realizado na EMPREL; e d) disseminação do conhecimento do RisAgi e de Melhoria do Processo de Software para atender ao mercado nacional e global.

3. Agenda

Assim, como sumário prévio da apresentação pode-se listar os tópicos:

- *Overview* sobre a Área de Processo de Gerência de Risco do CMMI-DEV;
- O Método Ágil RisAgi para a Implementação da Gerência de Riscos;
- A Cultura de Desenvolvimento de Software da EMPREL;
- O Estudo de Caso da Implementação do processo de Gerência de Riscos usando o RisAgi;
- Os Ferramentais (Cerimônias, Artefatos, etc.) de Apoio à Implementação;
- Pontos Fortes e Fracos, e Oportunidades de Melhorias na Implementação da Área de Processo de Gerência de Risco na EMPREL;
- Principais Pontos na Implementação do CMMI-DEV na EMPREL;
- A Avaliação Oficial da EMPREL no Modelo CMMI-DEV.

4. Audiência

Para esta apresentação o perfil dos profissionais que se beneficiarão com as informações a serem tratadas/discutidas pelos palestrantes é bem abrangente, a saber: desde

empresários do ramo de desenvolvimento de software que tenham interesse na implementação dos processos organizacionais usando práticas constantes em modelos de qualidade; passando por diferentes perfis (gerentes/líderes de projetos, equipes de desenvolvimento, testadores, etc.) envolvidos no processo de desenvolvimento de software; até alunos e professores de Universidades que trabalham com pesquisa/extensão no contexto da melhoria do processo de software organizacional.

Como pré-requisito de conhecimento técnico para o público interessado na apresentação, este deve ter o entendimento básico na área de processo tratada neste trabalho, bem como o entendimento da cultura presente no CMMI-DEV.

5. Breve Biografia do Palestrante

Como biografia do palestrante desta apresentação, tem-se:

- Sandro Ronaldo Bezerra Oliveira, é professor associado e pesquisador da Universidade Federal do Pará desde 2008, tendo orientado inúmeros alunos de doutorado, mestrado, conclusão de curso e iniciação científica. Mantém cooperação de pesquisa com a Universidade de Coimbra e o Centro de Informática da UFPE. Os seus interesses de pesquisa concernem a especificação de abordagens para a implementação de modelos e normas para a qualidade do processo de software organizacional. Concluiu seu estágio pós-doutoral no CIn/UFPE, sob a supervisão do Prof. PhD. Alexandre Marcos Lins de Vasconcelos. Vem ativamente participando de inúmeros projetos de pesquisa na área de Qualidade de Software, sendo coordenador de projetos financiados pelo CNPq, CAPES, FAPESPA, SECTI, dentre os quais destaca-se o SPIDER – *Software Process Improvement: DEvelopment and Research*, que recebeu em 2012 o Prêmio Dorgival Brandão Júnior referente ao Ciclo 2011 do PBQP Software, concedido pela SEPIN-MCTI. É editor associado de periódicos e membro do comitê de diversas conferências nacionais e internacionais. Currículo Lattes disponível em: <http://lattes.cnpq.br/2080791630485427>.

6. Agradecimentos

Este trabalho recebeu apoio financeiro da Prefeitura do Recife, bem como faz parte da metodologia de trabalho da SWQuality Consultoria e Sistemas Ltda. Além disso, está inserido no escopo do Projeto SPIDER (www.spider.ufpa.br), instituído na UFPA.

Referências

- Cohn, M. (2005) “Agile Estimating and Planning”, Prentice Hall.
- Highsmith, J. (2004) “Agile Project Management: Creating Innovative Products”. Addison-Wesley.
- Morgado, G., Gesser, I., Silveira, D., Manso, F., Lima, P., Schmitz, E. (2007) “Práticas do CMMI como Regras de Negócio”. São Paulo: Produção.
- Oliveira, S. R. B., Santos, B., Neto, B., Cardoso, P. (2013) “RisAgi: Uma Metodologia Ágil para Gestão de Riscos em Projetos de Desenvolvimento de Software”. In: IX WAMPS. Campinas – SP.
- PMI – Project Management Institute (2008) “A Guide to the Project Management Body of Knowledge”. Fourth Edition, Newton Square, USA.
- SEI (2010) “Capability Maturity Model Integration for Development – CMMI-Dev”. Versão 1.3. Carnegie Mellon. USA.
- Sommerville, I. (2010) “Software Engineering”. 9th edition. Addison-Wesley.

Resultados e Lições Aprendidas na Implementação do Modelo CMMI-DEV, Nível de Maturidade 2, usando Metodologias Ágeis: Um Caso de Sucesso na Jambu Tecnologia

Marcelo Rocha de Sá¹, Sandro Ronaldo Bezerra Oliveira²

¹Jambu Tecnologia – Travessa Alenquer, 131 – Cidade Velha – 66020-020 – Belém – PA – Brasil

²Programa de Pós-Graduação em Ciência da Computação – Universidade Federal do Pará (UFPA), Rua Augusto Corrêa, 1 – Guamá – 66075-110 – Belém – PA – Brasil

marcelo@jambu.com.br, srbo@ufpa.br

Abstract. *This presentation aims to show that the addition of an agile methodology with a traditional quality model can be very productive, though adaptations are necessary. The quality model chosen is part of the CMMI program, CMMI-DEV, to be able to meet the international requirements for evaluation, definition and improvement of software processes. As for agile methodologies, were used those that stand out the activities of project management practices, since the maturity level of the CMMI-DEV to be treated in this presentation is 2, containing the process areas of Project Planning, Project Monitoring and Control, Requirements Management, Configuration Management, Measurement and Analysis, and Product and Process Quality Assurance.*

Resumo. *Esta apresentação tem por objetivo mostrar que a junção de uma metodologia ágil com um modelo de qualidade tradicional pode ser muito produtiva, embora adaptações sejam necessárias. O modelo de qualidade escolhido faz parte do programa CMMI, o CMMI-DEV, por ser capaz de atender as exigências internacionais para avaliação, definição e melhoria dos processos de software. Quanto às metodologias ágeis, foram usadas as que se destacam pelas atividades práticas de gerenciamento de projetos, uma vez que o nível de maturidade do CMMI-DEV a ser tratado nesta apresentação é o 2, contendo as áreas de processo de Planejamento do Projeto, Monitoramento e Controle do Projeto, Gerência de Requisitos, Gerência da Configuração, Medição e Análise, e Garantia da Qualidade do Produto e Processo.*

1. Introdução

Cada vez mais empresas e organizações necessitam de produtos com qualidade, livres de erros e com soluções imediatas. Para atender a esta demanda, empresas de software visam, em seus projetos, a diminuição de conflitos, o aumento da confiabilidade e a redução dos custos, resultando em maior satisfação do cliente [Travassos e Kalinowski 2014]. Assim, pode-se perceber que o software tem estado cada vez mais presente no cotidiano das pessoas, porém a maioria dos projetos de software não atende aos objetivos traçados. Segundo Catunda *et al.* (2011), isso decorre da falta de processos adequados nas organizações em que eles são desenvolvidos.

Para entregar sistemas de software com alta qualidade, as empresas têm adotado modelos de maturidade que visam a melhoria da qualidade dos processos de

desenvolvimento. Neste contexto, modelos de qualidade são propostos a fim de medir e garantir a excelência do software produzido [Catunda *et al.* 2011].

Juntamente com a melhoria de processos, a partir do ano 2000 surgiu uma tendência para o desenvolvimento ágil de sistemas de software devido a um ritmo acelerado de mudanças e inovações na tecnologia da informação, em organizações e no ambiente de negócios. Assim, as metodologias ágeis preconizam menor burocratização e demandam maior esforço das pessoas envolvidas. Elas também permitem acolher mudanças mais naturalmente e adaptações em qualquer momento [Manifesto 2001].

Essa discussão baseia-se no fato de que governos e mercado, de maneira geral, esperam como diferencial, que as empresas comprovem a qualidade de seu processo de desenvolvimento e a maneira formal de se comprovar isso, atualmente, costuma ser opositora ao que preconizam as metodologias ágeis [Travassos e Kalinowski 2014]. As formas de desenvolvimento existentes dentro da área da Engenharia de Software são a origem dessa discussão. Isto ocorre devido às impedâncias existentes entre processos tradicionais, desenvolvidos no início do século baseados no modelo de produção Fordista para a indústria e os processos ágeis baseados no Toyotismo (produção enxuta).

O desenvolvimento ágil busca tratar o processo de produção de software de maneira diferenciada, uma vez que entende que se trata de um processo criativo e que produz um bem intangível. Porém, as normas e modelos tradicionais baseiam-se no pensar fordista, segundo o qual foram criadas, tratando a produção de software de maneira parecida com os produtos manufaturados.

O que se procura mostrar é que a implementação de um modelo de maturidade deve ser o máximo adaptativa possível à realidade da empresa/projeto. Assim, percebe-se a necessidade de ferramentais (procedimentos, práticas, produtos de trabalhos mais enxutos, softwares, etc.) que auxiliam as empresas a obter um melhor gerenciamento dos seus processos de software, com o objetivo de unir conceitos de metodologias ágeis e processos constantes em modelos como o MPS.BR – Melhoria do Processo de Software Brasileiro [SOFTEX 2016] ou o CMMI – *Capability Maturity Model Integration* [SEI 2010].

2. Proposta de Apresentação

Como solução para minimizar o cenário apresentado, a empresa Jambu Tecnologia Consultoria e Engenharia Ltda (empresa privada genuinamente paraense centrada na produção de soluções tecnológicas, inovadoras e regionalizadas, para gestão de conteúdo digital corporativo e gestão de processos de negócio BPM, utilizando sistemas de software e hardware livres, sendo incentivadora do uso destes sistemas como alternativa tecnológica qualificada para diminuição de desigualdades socioeconômicas) submeteu e aprovou em 2015 um projeto no Edital do programa Sebraetec (<http://www.sistemasebraetec.sebrae.com.br>) da Unidade Pará com foco na implementação da melhoria do processo de software usando o CMMI-DEV (*CMMI for Development*). Assim, durante o trabalho de implementação das práticas constantes neste modelo, a empresa manteve sua cultura ágil alinhada às práticas de maturidade.

Ao final deste projeto de implementação, definiu-se um Guia de Implementação das Áreas de Processo constantes no Nível 2 de Maturidade do CMMI-DEV usando Metodologias Ágeis (Scrum, XP, Kanban, *Planning Poker*, *Risk Agile*, *Checklists*, etc.), cujo foco está em orientar a adequação dos processos de software organizacionais aos

dois contextos. O guia atende principalmente as empresas de desenvolvimento de software na implementação do programa de definição e melhoria dos processos organizacionais na área de software, além da investigação e da disseminação do conhecimento nas práticas propostas pelo CMMI-DEV e metodologias ágeis, a partir da adaptação de cerimônias e evidências já usadas na implementação das metodologias ágeis, bem como na criação de outras a partir das orientações definidas no CMMI-DEV.

Vale ressaltar que em decorrência do sucedido processo de implementação do CMMI-DEV usando metodologias ágeis na Jambu Tecnologia, a empresa teve em Junho/2016 seus processos de software avaliados pela equipe da SWQuality Consultoria e Sistemas Ltda. alcançando a maturidade e capacidade no nível de maturidade 2 deste modelo, tornando-se a primeira empresa de software da região Norte do Brasil avaliada no modelo CMMI-DEV.

Desta forma, entende-se que a apresentação permitirá a pesquisadores e empreendedores acesso a serviços em Qualidade de Software e Metodologias Ágeis. Para isso, o foco da apresentação é direcionado a alguns segmentos principais: a) adequação das recomendações dos processos constantes no nível de maturidade 2 do CMMI-DEV para atendimento das práticas existentes em Metodologias Ágeis, apresentando lições aprendidas e melhores práticas obtidas; b) discussão do Guia para a Implementação da Melhoria do Processo usando características da Qualidade de Software e das Metodologias Ágeis; c) apresentação do Estudo de Caso realizado na Jambu Tecnologia com foco na análise de casos de uso da implementação da Qualidade de Software e Agilidade; d) adequação dos ferramentais de apoio com foco no atendimento das práticas constantes no modelo de qualidade do CMMI-DEV e metodologias ágeis; e e) disseminação do conhecimento em Fundamentos Ágeis e Fundamentos em Programas de Melhoria do Processo de Software para atender ao mercado nacional e global.

3. Agenda

Assim, como sumário prévio da apresentação pode-se listar os tópicos:

- *Overview* sobre Metodologias Ágeis e Modelos de Maturidade do Processo de Software;
- A Cultura de Desenvolvimento de Software da Jambu Tecnologia;
- O Estudo de Caso da Implementação do CMMI-DEV Nível de Maturidade 2 usando Metodologias Ágeis no Contexto de Software da Jambu Tecnologia;
- Os Ferramentais (Modelos, Procedimentos, Softwares, etc.) de Apoio à Implementação de Melhoria do Processo de Software na Jambu Tecnologia;
- A Avaliação Oficial da Jambu Tecnologia no Modelo CMMI-DEV;
- Pontos Fortes e Fracos, e Oportunidades de Melhorias na Implementação do Processo da Jambu Tecnologia;
- Principais Pontos de Atenção na Implementação do CMMI-DEV usando Metodologias Ágeis.

4. Audiência

Para esta apresentação o perfil dos profissionais que se beneficiarão com as informações a serem tratadas/discutidas pelos palestrantes é bem abrangente, a saber: desde empresários do ramo desenvolvimento de software que tenham interesse na implementação dos processos organizacionais usando práticas constantes em modelos

de qualidade e metodologias ágeis; passando por diferentes perfis (gerentes/líderes de projetos, equipes de desenvolvimento, testadores, etc.) envolvidos no processo de desenvolvimento de software; até alunos e professores de Universidades que trabalham com pesquisa/extensão no contexto da melhoria do processo de software organizacional.

Como pré-requisito de conhecimento técnico para o público interessado na apresentação, este deve ter o entendimento básico das filosofias constantes nas metodologias ágeis, bem como o entendimento da cultura presente no modelo de melhoria do processo de software CMMI-DEV.

5. Breve Biografia do Palestrante

Como biografia do palestrante desta apresentação, tem-se:

- Sandro Ronaldo Bezerra Oliveira, é professor associado e pesquisador da Universidade Federal do Pará desde 2008, tendo orientado inúmeros alunos de doutorado, mestrado, conclusão de curso e iniciação científica. Mantém cooperação de pesquisa com a Universidade de Coimbra e o Centro de Informática da UFPE. Os seus interesses de pesquisa concernem a especificação de abordagens para a implementação de modelos e normas para a qualidade do processo de software organizacional. Concluiu seu estágio pós-doutoral no CIn/UFPE, sob a supervisão do Prof. PhD. Alexandre Marcos Lins de Vasconcelos. Vem ativamente participando de inúmeros projetos de pesquisa na área de Qualidade de Software, sendo coordenador de projetos financiados pelo CNPq, CAPES, FAPESPA, SECTI, dentre os quais destaca-se o SPIDER – *Software Process Improvement: DEvelopment and Research*, que recebeu em 2012 o Prêmio Dorgival Brandão Júnior referente ao Ciclo 2011 do PBQP Software, concedido pela SEPIN-MCTI. É editor associado de periódicos e membro do comitê de diversas conferências nacionais e internacionais. Currículo Lattes disponível em: <http://lattes.cnpq.br/2080791630485427>.

6. Agradecimentos

Este trabalho recebeu apoio financeiro do SEBRAE Pará para a implementação do programa de melhoria do processo de software organizacional e da equipe de avaliação da SWQuality Consultoria e Sistemas Ltda. Além disso, está inserido no escopo do Projeto SPIDER (www.spider.ufpa.br), instituído na Universidade Federal do Pará.

Referências

- Catunda, E., Nascimento, C., Cerdeiral, C., Santos, G., Nunes, E., Shots, N., Shots, M., Rocha, A. R. (2011) “Implementação do Nível F do MR-MPS com Práticas Ágeis do Scrum em uma Fábrica de Software”. In: X SBQS. Curitiba – PR.
- Manifesto (2011) “Manifesto for Agile Software Development”. Disponível em: <http://www.agilemanifesto.org/>. Último Acesso em Julho/2016.
- SEI (2010) “Capability Maturity Model Integration for Development – CMMI-Dev”. Versão 1.3. Carnegie Mellon. USA.
- SOFTEX (2016) “Melhoria do Processo de Software Brasileiro– Guia Geral MPS de Software:2016”. Brasil.
- Travassos, G. H., Kalinowski, M. (2014) “iMPS 2013 - Evidências Sobre o Desempenho das Empresas que Adotaram o Modelo MPS-SW”. In: SOFTEX. Campinas - SP.