

CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico

Dados do Projeto e do Proponente

Sigla:	ESSL
Título do Projeto:	Usando a Engenharia de Software para Apoiar a Construção de Software Livre
Referência da Chamada:	Chamada CT-INFO: CNPq 2003 – SOFTWARE LIVRE Programa de Pesquisa e Desenvolvimento Tecnológico em Software Livre
Linha(s) de atuação em que se insere o projeto/Grupo:	<input checked="" type="checkbox"/> Pesquisa <input checked="" type="checkbox"/> Desenvolvimento de aplicações <input checked="" type="checkbox"/> Projetos de demonstração <input checked="" type="checkbox"/> Inovação tecnológica <input checked="" type="checkbox"/> Transferência de tecnologia <input type="checkbox"/> Capacitação tecnológica <input type="checkbox"/> Estudos prospectivos e outros
Coordenador do Projeto:	Julio Cesar Sampaio do Prado Leite
Instituição Executora:	Departamento de Informática – PUC-Rio
Data:	2003-10-28

1. Identificação e caracterização do problema

Descreva, objetivamente, com o apoio do referencial teórico, a questão ou problema focalizado.

A engenharia de software clássica tem procurado entender de que maneira a construção de software livre, que aparentemente não segue os princípios clássicos, tem tido sucesso. Um workshop específico sobre isso foi recentemente adicionado ao ICSE (Conferência Internacional de Engenharia de Software) e um artigo de Mockus, Fielding e Herbsleb, publicado no conceituado ACM TOSEM, Vol. 11, No. 3, July 2002, foi um dos primeiros artigos sérios a fazer uma análise cuidadosa do assunto. Um dos grandes problemas, identificados pelos próprios promotores de código livre é facilitar a entrada de novos colaboradores, ou seja como tornar o conhecimento existente no código e nos documentos de apoio (e-mail, listas de discussão) mais fáceis de ler e mais integrados ao documento base; o código.

Em nosso artigo publicado no SBES 2003 resumimos o problema da seguinte maneira. “O desenvolvimento de software com código aberto tem sido freqüente e tem mostrado resultados positivos. No entanto, um dos pontos considerados críticos é a qualidade das descrições, quer seja de código ou de informações acessórias. Isto gera dificuldades tanto na compreensão da estrutura do sistema como na identificação dos módulos que devem sofrer alterações quando há mudanças em requisitos. Entendemos que aqui os preceitos de Engenharia de Software podem ser úteis, sem, no entanto, alterar a filosofia de desenvolvimento do software livre. Propomos o uso de Cenários como uma maneira prática e fácil para enriquecer o código aberto. Cenários fornecem uma representação uniforme da informação, promovem o reuso facilitando a identificação de funcionalidades, agilizam a manutenção e a identificação de partes do código que seriam afetadas no caso de mudanças. Além disto a padronização oferecida pelos cenários permite a automação de parte destas tarefas.” (Enriquecendo o Código com Cenários, Lyrene Fernandes, Miriam Sayão, Julio Cesar Leite e Karin Breitman, XVII Anais do Simpósio Brasileiro de Engenharia de Software, Sociedade Brasileira de Computação, ISBN 85-7401-126-6, Out. 2003, pp.67-82. (2003).

Nosso artigo, resumido acima, demonstra a possibilidade de um método que resolva o problema apontado com isso contribuindo para que o desenvolvimento de software livre seja de melhor qualidade.

Um dos principais argumentos de nosso método está descrito no artigo acima da seguinte maneira. “Nossa equipe tem investigado a adoção do paradigma de software livre como alternativa para a elaboração e disseminação de software de qualidade a custos baixos. Temos focado no desenvolvimento de métodos, técnicas e ferramentas para facilitar questões relacionadas ao reuso e evolução do software livre. Dentre eles, uma nova abordagem para a construção de código com base em cenários, de maneira que cenários estejam encapsulados no código. Esta abordagem possui dois enfoques igualmente importantes: primeiro, o desenvolvimento de software livre baseado em cenários; e

segundo, o uso de cenários como apoio para o desenvolvimento de software livre. Neste trabalho tratamos o segundo caso, focando o uso de cenários como comentários do código e documentação que pode ser extraída automaticamente. Isto fornece a rastreabilidade entre requisitos (registrados nos cenários) e componentes, e as ligações de dependência entre requisitos. Esta simbiose entre cenários e código passa a ser uma espécie de guia para a construção de programas que também objetive o ideal de Knuth com a programação literária (Knuth, D. E. Literate Programming. The Computer Journal, vol. 27, nº 2, pp 97-111)".

O outro problema que abordamos, sob o ponto de vista de aplicação, é o problema da democratização do conhecimento. Hoje, grandes editoras mantêm um monopólio sobre o conhecimento científico que dificulta sua divulgação em função do custo de assinaturas de revistas e o alto custo dos livros. A sociedade da informação, através a rede, fornece uma rara oportunidade para que o conhecimento passe a ser um bem público. Esta tese é defendida pelo professor Willinsk criador do projeto PKP.

Na descrição sumária do projeto enfatizamos nosso duplo objetivo. "Avançar e implementar efetivamente o método de construção de software com base em cenários criando dois softwares livres. Um para ser aplicado no processo de construção de software livre, a implementação do método e outro para dar apoio a divulgação livre (democratização do conhecimento) de resultados científicos através de uma apoio efetivo para a editoração de revistas e livros de cunho científico nos moldes do projeto "Public Knowledge Project" (<http://www.pkp.ubc.ca/>)."

2. Justificativa

Faça, de forma sucinta, um relato da situação-problema abordada, citando dados ou informações significativas que possam delimitar seu contexto histórico. Fundamente sua defesa e linha de atuação/tema indicando a relevância de seu projeto para o País, seu caráter inovador e porque seu projeto se enquadra dentro das exigências do Edital.

O nosso grupo foi o primeiro grupo brasileiro de engenharia de software a estudar o problema de software livre sob a ótica de engenharia de software. Identificamos e situamos o processo de construção de software livre e relatamos sobre suas vantagens e problemas em artigo recentemente publicado nos anais do Simpósio Brasileiro de Engenharia de Software de 2003. Propomos uma maneira pioneira para sustentar de forma natural, dentro do princípio de que código é documento, e prática, o êxito da idéia de cenários aplicados em várias comunidades, o enriquecimento do código fonte com base em resultados pioneiros

de nosso grupo na área de cenários, léxico e de ontologias. Além disso, e principalmente estaremos dando início em língua Portuguesa ao princípio de democratização do conhecimento, como proposto no projeto PKP.

Nosso projeto ao propor a aplicação do nosso método na própria ferramenta do método e na construção de um PKP em português, contribui de forma decisiva para que primeiro possamos aprender melhor como apoiar o software livre e segundo para efetivamente, no futuro, possibilitar a divulgação de conhecimento sem que se tenha que pagar elevados royalties. Portanto economicamente, temos a vantagem de melhorar a maneira de fazer software livre, com isso diminuindo o impacto do item software na nossa balança de pagamentos como também na mesma rubrica poderemos no futuro economizar também ao não ter que pagar royalties as grandes editoras por conhecimento que deve, também, ser livre. Vide recentemente o caso do portal CAPES, a assinatura dessas edições eletrônicas são muito caras, mas no momento indispensáveis. No futuro, se a idéia do PKP se difundir, isso não será mais um problema. No caso nosso projeto é o primeiro a tratar disso do ponto de vista da língua portuguesa, pelo menos que saibamos.

3. Estratégias de crescimento do grupo de pesquisa no contexto do Plano Estratégico Institucional

Descrever ações de participação efetiva do grupo de tecnologia da informação (TI) no ambiente institucional visando os desenvolvimentos acadêmico, científico e/ou inovação tecnológica da instituição, destacando relacionamentos internos e externos de real interesse (parcerias, fornecedores, clientes, etc.)

Nosso grupo tem tido uma participação ativa em edição de revistas, conferências, workshops e construção e implementação de sítios de distribuição de conhecimento. Atualmente nosso grupo tem colaboração com várias universidades brasileiras e estrangeiras.

Estamos colaborando com universidades na Argentina, Espanha, Canadá, Alemanha, Estados Unidos em diferentes projetos. Na Argentina colaboramos com duas universidades (Tandil e La Plata), na Espanha colaboramos com duas universidades (Valencia e Valladolid), na Alemanha com Essen e Manheim e com o IESE, nos Estados Unidos com a CMU através do SEI, no Canadá com Toronto, Waterloo e York. Do ponto de vista brasileiro colaboramos com a UFSCar, com o IME, com a PUC-RS, com a UNB, e com a UFPE.

No contexto da PUC, participamos do LES (www.les.inf.puc-rio.br) e fazemos parte do projeto ESSMA que estuda os conceitos de agentes.

Nossos sítios: o portal de requisitos (www.er.les.inf.puc-rio.br) a série WER (www.inf.puc-rio.br/wer) e a ferramenta C&L (<http://139.82.24.189/cel/>) tem sido freqüentemente visitados.

Um dos primeiros usos do PKP em Português será a organização do conhecimento acumulado de engenharia de requisitos no contexto do workshop WER (www.inf.puc-rio.br/wer03) que irá agora para a sua sexta edição e será realizado em Piracicaba de modo a difundir ainda mais o conceito fundamental da engenharia de requisitos.

4. Objetivos e Metas

Indique os objetivos e metas a serem alcançados pelo projeto, explicitando os produtos resultantes.

Objetivo do Projeto: Avançar e implementar efetivamente o método de construção de software com base em cenários criando dois softwares livres. Um para ser aplicado no processo de construção de software livre, a implementação do método e outro para dar apoio a divulgação livre (democratização do conhecimento) de resultados científicos através de uma apoio efetivo para a editoração de revistas e livros de cunho científico nos moldes do projeto “Public Knowledge Project” (<http://www.pkp.ubc.ca/>).

Outros Resultados: Além dos dois software livres, um deles embutindo um ferramental de apoio para a metodologia, nossa principal contribuição sob a ótica de engenharia de software, vamos também produzir um relatório completo do projeto, onde através de nossa colaboração com IESE Fraunhofer empregaremos métodos comprovados para experimentação de software. Com isso teremos como reportar os sucessos ou obstáculos que vamos obter ou enfrentar com base em dados. Desse relatório pretendemos extrair dois artigos para revistas. Um voltado para revistas mais leves (tipo IEEE Software) e outro mais detalhado para um revista do tipo ACM TOSEM. Pretendemos também ao longo de 2004 apresentar um trabalho em workshop ou conferência relacionado.

Como fazer: No espaço de seis meses vamos implementar a primeira versão da ferramenta de apoio ao método. Essa ferramenta já tem um protótipo, (<http://139.82.24.189/cel/>), e portanto acreditamos que não teremos maiores problemas. Nos seis meses seguintes vamos fazer a re-engenharia da ferramenta PKP (<http://www.pkp.ubc.ca/demos/index.html>) e utilizando nosso método e ferramenta. Vale lembrar que utilizando nosso método todo o software fica em Português! Vamos, cuidadosamente, guardar informações sobre o processo (com dito acima). Esperamos hospedar nosso projeto, além de no nosso sítio, no sítio da Univates, que tem uma cópia do SourceForge. Se isso não for possível utilizaremos o sítio do SourceForge. Estaremos, como dito na descrição sumária, utilizando o esquema LAMP (Linux, Apache,

MySQL e PHP). Pretendemos seguir os padrões OSI para licenciamento. O software será construído todo em Português, claro fora os comandos dos software livres que utilizaremos. O uso do nosso método leva naturalmente a isso.

5. Equipe

Indique os nomes dos pesquisadores e função, contratados, a contratar (com perfil) e/ou associados a serem alocados no projeto, explicitando suas atividades.

Nossa equipe conta além do coordenador, de uma pesquisadora experiente e doutora na área de cenários, Karin Breitman. Duas doutorandas que estão estudando o problema ressaltado na Seção 1, Myriam Sayão (não listada na equipe por problemas com o CV Lattes) e Lyrene Fernandes. Uma aluna de mestrado, Carolina Felicíssimo que está estudando aspectos de ontologias e relacionamento com a ferramenta C&L, e três técnicos. Um desses técnicos será Roberto Cristoph, que defende sua dissertação de mestrado no início de 2004. O Roberto está escrevendo sobre software livre, com a ótica de evolução de software (vide <http://www-di.inf.puc-rio.br/~julio/evol.htm>). Os outros dois pretendemos recrutar em regime de dedicação parcial para dar apoio profissional a construção das sementes dos dois softwares.

Além desse equipe, procuraremos utilizar da melhor maneira possível nossos contatos com outras universidades e institutos de pesquisa (vide Seção 3).

Como sabemos todo software livre tem que partir de uma semente. Nossa idéia é que nosso projeto seja a semente da ferramenta de apoio a construção de software livre e do PKP em português.

6. Metodologia e Estratégia de Ação

Descreva a metodologia a ser empregada na execução do projeto e de que maneira a estratégia adotada contribuirá para alcançar os objetivos propostos.

Nosso método é sedimentar o que já conhecemos e empregar os resultados na construção dos dois softwares.

Vale lembrar de nossa experiência em engenharia reversa.(vide nossos trabalhos na área de reuso de software e engenharia reversa e nossa contínua colaboração com o Professor Prado da UFSCar que vem utilizando de maneira surpreendente o ferramental do Draco-PUC (<http://www.inf.puc-rio.br/~draco/>) em software de larga escala. Vide dissertação de mestrado de Adail Roberto Nogueira. Transformação de DataFlex Procedural para Visual DataFlex Orientado a Objetos Reusando um Framework (Banca:PRADO, A. F., BIAJIZ, M., LEITE, J. C. S. P.) defendida em 22/02/2002 que aplicou Draco-PUC a um sistema da ordem de 5 milhões de linha de código.

O projeto da ferramenta de apoio já está iniciado, conforme ressaltamos acima (<http://139.82.24.189/cel/>) e com nossa experiência em engenharia reversa, não cremos que teremos problemas em reverter o código, livre, do PKP.

Conforme enfatizamos vamos nos primeiros seis meses focar no ferramental de apoio e depois fazer a engenharia reversa do PKP reconstruindo-o em Português e com os cenários embutidos no código.

6. Orçamento e cronograma físico-financeiro

Apresente, graficamente, em seqüência cronológica, as etapas físicas do projeto, indicando seus prazos de execução e as detalhando as respectivas previsões de despesa para cada uma delas. Aloque a mão-de-obra necessária para o desenvolvimento na classificação de despesas de terceiros pessoa física.

Nosso orçamento é simples. Vinte e seis mil reais de equipamento (3 servidores e um laptop) para Janeiro de 2004. Três técnicos em tempo parcial, estimou-se R\$ 1.500,00 por técnico por mês, a partir de Dezembro (total de R\$ 54.000,00). Passagens, duas nacionais, provavelmente para o workshop de software livre brasileiro e uma passagem internacional ou para o ICSE no workshop de software livre ou para uma visita em loco ao projeto PKP. Total de R\$ 9.000,00. E quinze diárias (R\$ 1855,65). Os gastos de viagem estimamos para meados de 2004, mas não temos todas as datas, ainda. Estamos também solicitando seis mil reais para despesas gerais na rubrica de material de consumo.

7. Relevância dos resultados e impactos esperados

Descreva os resultados e/ou produtos esperados. Avalie a repercussão e/ou impactos (sócio-econômicos, técnico-científicos e ambientais) desses resultados na solução do problema focalizado.

Se conseguirmos o que planejamos estaremos com certeza fornecendo uma semente que pode ajudar a trazer grandes economias para o país. É claro que o sucesso do nosso trabalho não vai depender só dos resultados que alcançarmos, mas também da nossa capacidade de divulgação e motivação para que outros utilizem o que vamos produzir. Se além de ajudarmos a construir software livre (vide balanço de pagamentos brasileira onde o item software é da ordem de bilhões: contra nós), podermos iniciar um movimento de democratização do conhecimento, através das idéias do PKP, certamente estaremos trazendo um ROI (retorno do investimento) difícil de ser medido.

8. Riscos e Dificuldades

Refletir sobre possíveis dificuldades e riscos potenciais que poderão interferir na execução das ações propostas e comprometer o alcance das metas e objetivos preconizados. Explicitar as medidas previstas para contornar ou superar essas dificuldades.

Riscos principais são dois: falha na qualidade do produto produzido e dificuldade de atrair, após o projeto, usuários e desenvolvedores para darem continuação a semente.

É difícil alcançar níveis de qualidade em projetos onde os recursos são pesquisadores e programadores em tempo parcial. Esse é um risco, mas é um risco que correm todos os projetos de universidade. Vários são os casos de sucesso, mesmo que levem mais tempo que o esperado, no entanto vários são os casos de insucesso.

Cremos que temos muita chance de ter sucesso, uma equipe experiente e motivada e com conhecimento do que estamos propondo. No entanto, software é um produto extremamente complexo. Procuraremos ter o melhor nível de qualidade possível dentro das condições que delineamos.

A divulgação dos resultados pode ter sucesso, mas por outro lado existem uma série de concorrentes e uma pressão das editoras para que projetos como PKP não dêem certo. No entanto esse tipo de software pode ajudar uma série de casos onde a divulgação não interessaria editoras, mas que pode ser de interesse do público em geral.

9. Experiências e linhas de financiamento a projetos

Indique as melhores práticas e outros projetos de pesquisa em andamento dos quais participem membros da equipe proponente, incluindo o título, vigência, a dedicação em hora/homem/mês, a origem e o valor do financiamento.

Acreditamos que a história do grupo de requisitos da PUC é uma história de sucesso, apesar dos poucos recursos que utilizou ao longo de seus mais de cinco anos de existência. Antes o foco principal do coordenador era o projeto Draco-PUC, um sucesso absoluto em termos acadêmicos, 3 teses de doutorado e cerca de 20 dissertações de mestrado, principalmente na UFSCar com recursos irrisórios de financiamento.

No que concerne requisitos, os êxitos acadêmicos são ainda maiores, não só em termos da rede de colaboração que se formou, como também com o volume e a qualidade das publicações dos últimos anos.

No que se refere a projetos o grupo da PUC-Rio liderou um projeto de sucesso no âmbito do PADCT (Projeto Plataforma), que possibilitou a criação do portal de requisitos além de apoiar o WER de 2000 e possibilitar uma integração com a Espanha através do projeto WEST (<http://www.dsic.upv.es/~west2001/>). Com base em nossa experiência (PUC-Rio, UNB, UFPE), fomos pioneiros ao tratarmos dos assuntos de transferência de tecnologia no âmbito de engenharia de requisitos (The Journal of Technology Transfer: , v.28, p.159 - 165, 2003, Kluwer Academic Publishers, Holanda, Requirements Engineering Technology Transfer: An Experience Report. Pinheiro, F.A.C, Leite, J.C.S.P e Castro, J.F.B).

10. Atendimento aos Critérios da Chamada

Destaque os aspectos relevantes da proposta quanto aos critérios para avaliação constantes na Chamada.

Acreditamos que com as contribuições esperadas estamos atendendo explicitamente o que pede o edital que é gerar conhecimento para a nos ajudar a construir software livre e mais estamos também fazendo uma contribuição, não pedida no edital mas relacionada, que é a idéia da democratização do conhecimento. Ou seja faremos software para gerar software e software para gerar conhecimento.

Acreditamos que nosso projeto encaixa-se de maneira singular com os propósitos do governo brasileiro de aprimorar a competência brasileira no setor de software.

11. Considerações finais

Informe, caso julgue necessário, outros critérios que possam ser considerados na avaliação de sua proposta (além dos constantes da Chamada) e, sucintamente, alguma informação adicional que, a seu juízo, seja relevante para a elucidação, compreensão ou apreciação de seu projeto.

Acreditamos que o financiamento desse projeto trará uma motivação extra ao grupo que já vem tratando desse problema e cremos será um impulso na contribuição que os conhecimentos mais avançados de engenharia de software podem trazer para a construção de software livre onde o trabalho de colaboração é a peça chave.