

ProjectSpace: Uma Ferramenta para Gerenciamento e Monitoração de Projeto de Software

Ana Cristina Rouiller
Silvio R. Lemos Meira
Sandro Ronaldo Bezerra Oliveira
Alexandre Marcos Lins de Vasconcelos
Centro de Informática
Universidade Federal de Pernambuco
Caixa Postal 7851, 50732-970,
Recife - PE, Brasil
acr, srlm, srbo, amlv@cin.ufpe.br

Gilmar de Souza
Wellington Silva
Paulo G. F. Amado
CESAR - Centro de Estudos Superiores do Recife
Caixa Postal 7851, 50732-970,
Recife - PE, Brasil
gilmar, wellington, amado@cesar.org.br

Resumo

Este artigo apresenta o ProjectSpace, um projeto de iniciativa do Centro de Informática da UFPE em parceria com o CESAR – Centro de Estudos Avançados do Recife e a Devex Tecnologia e Sistemas que objetiva a criação de um ambiente para monitoração e gerenciamento de projetos de software. O ProjectSpace vem sendo especificado e implementado a dois anos e atualmente se encontra com alguns módulos em fase de implementação enquanto outros já estão em fases de uso e validação. Este artigo trás a arquitetura do ProjectSpace e a primeira versão da ferramenta.

1. Introdução

De forma distinta dos processos de manufatura, que podem ser definidos e automatizados em operações de linha de produção, os processos de software são compostos por atividades que são realizadas por humanos e são atividades intelectuais e criativas envolvendo muita colaboração [1]. A engenharia de processo tem se esforçado no sentido de definir modelos e padrões para a construção de um efetivo processo de desenvolvimento de software para as organizações tais como [3] [4] [6] e [7]. O reconhecimento da importância dos processos e o crescimento da cultura de processo têm levado as organizações à criação de ambientes de engenharia de software mais eficientes. Um exemplo são os PSEEs - *Process-Centered Software Engineering Environment*. Os PSEEs integram tanto os requisitos do produto, que são o foco da engenharia de software, como os requisitos do processo, que são o foco do gerenciamento do projeto e da engenharia do processo.

Todavia, a maioria das organizações de software ainda sentem dificuldade em definir os seus processos padrões, muitas vezes por nem sequer saberem gerenciar os seus projetos de forma adequada [5] [7]. O gerenciamento de projeto objetiva, entre outras coisas, assegurar que processos particulares sejam seguidos, coordenando e monitorando as atividades da engenharia de software. Porém, o gerenciamento de projetos de software ainda é pouco abordado e praticado. Os ambientes tradicionais de engenharia de software têm suportado somente a engenharia de software, assumindo um processo implícito e tendo como foco principal o produto. Esta visão limita as organizações no que diz respeito à tomada de decisões, ao estabelecimento e arquivamento de metas organizacionais, à determinação de pontos para melhoria, à estipulação de prazos para entrega de produtos, à obtenção de uma certificação, entre outros. Além dessas limitações, os ambientes de gerenciamento de projetos disponíveis no mercado não são específicos para software impossibilitando a obtenção de métricas automáticas, sem exigir grande intervenção humana.

Enxergando estas problemáticas o Centro de Informática da UFPE em cooperação com o CESAR - Centro de Estudos Avançados do Recife e a Devex Tecnologia e Sistemas criaram o projeto *ProjectSpace*. O ProjectSpace tem como principal motivação o fato de que gerenciando os projetos de software, registrando e acompanhando as suas execuções torna-se possível a obtenção de métricas para introduzir melhorias no processo de software.

2. Arquitetura do ProjectSpace

O *ProjectSpace* é composto por oito subsistemas principais que cooperam entre si, como mostra a figura 1:

- *TimeSheet* – fornece apoio no processo de disciplina pessoal realizando o registro e avaliação do tempo gasto com cada atividade;
- *ResourceManager* – Gerencia os recursos da organização tais como: espaço físico, pessoal, treinamentos, ferramentas, habilidades e funções individuais e de equipes, etc. Além de proporcionar uma previsão da liberação destes recursos;
- *RequirementManager* – Gerencia os requisitos dos sistemas de software de cada projeto efetuando o controle dos requisitos por sistema de software, associando os requisitos às aplicações (ou atividades) a serem desenvolvidas, promove uma visão da realização destes requisitos;
- *ProcessManager* – Gerencia os elementos do processo de software como: atividades, métodos, técnicas, modelos de ciclo de vida, padrões e normas de qualidade, classe de problemas do desenvolvimento, etc. Este módulo tem por função, também, auxiliar a confecção de planos padrão de projetos e dar uma visão de problemas gerais do processo de software;
- *QualityManager* – Gerencia os atributos de qualidade do processo de software efetuando o cadastramento de modelos e normas e mapeando a dimensão de processo para as atividades da organização. Deve propiciar uma avaliação parcial do processo de software, no que se refere a gerência de projetos;
- *PlanEngine* – Gerencia os processos de desenvolvimento de software instanciados para um projeto específico. Deve ter funções como: auxílio na confecção do plano de projetos, acompanhamento da execução do plano, registro das versões e alterações do plano, registro das versões e alterações das atividades dos planos, visualização do andamento dos trabalhos, ativação das ferramentas CASE através do ToolManager, comunicação parcialmente automatizada, visualização das restrições do projeto e das atividades, registro de tempo gasto nas atividades de desenvolvimento, etc.;
- *ToolManager* – Gerencia e ativa as ferramentas CASE através do registro e controle das ferramentas CASE;
- *DataAccessLayer* – Realiza o acesso às informações armazenadas na base compartilhada por todos os subsistemas do ProjectSpace.

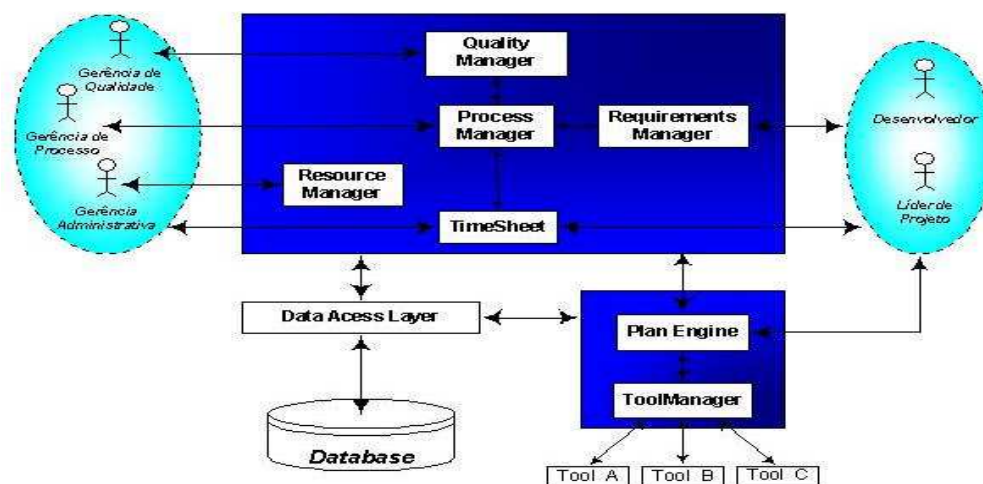


Figura 1– Arquitetura do ProjectSpace

3. Versão Inicial do ProjectSpace

As principais funções da versão inicial do *ProjectSpace* são: auxílio na confecção do plano de projetos, acompanhamento da execução do plano, registro das versões e alterações do plano, registro das versões e alterações das atividades dos planos, visualização do andamento dos trabalhos, ativação

das ferramentas CASE através do ToolManager, comunicação parcialmente automatizada, visualização das restrições do projeto e das atividades, registro de tempo gasto nas atividades de desenvolvimento, entre outras.

No *ProjectSpace* um projeto é categorizado por um software ou um serviço a ser realizado pela organização e pode conter diversos planos associados a ele. As especificações destes planos devem seguir os planos padrões definidos pela organização. Um plano padrão pode ser especificado pelo gerente de processo e deve compreender um conjunto de atividades, técnicas, treinamentos, habilidades, ferramentas, recursos necessários e possíveis para a realização de um projeto específico (de desenvolvimento de software ou não). O plano padrão é um instrumento que deve servir de base para a construção do plano do projeto, mas não necessariamente deve ser seguido à risca em sua definição. No caso dos planos de desenvolvimento, estes devem possuir uma metodologia e um modelo de ciclo de vida específico associado. Os planos padrões podem possuir finalidades muito diferenciadas tais como: treinamento, desenvolvimento de software, consultoria, avaliação de produto, etc. Dentro de cada tipo específico podemos encontrar vários padrões que dependem de outros fatores como: metodologia adotada, tipo de curso que será realizado, etc. A combinação de metodologia, modelo de ciclo de vida, atividades, etapas, (técnicas) e fases formam um plano padrão.

Um plano de projeto no *ProjectSpace* identifica as atividades, as restrições, os recursos, os períodos, os relacionamentos, as responsabilidades, cursos, etc. necessários para a realização de um projeto de software específico. Um plano de projeto pode possuir diversas versões dependendo das alterações que foram realizadas durante sua execução. Daremos o nome de *logs (alterações) leves* aos que não implicarem em mudança de versão e *logs (alterações) graves* os que implicarem em geração de uma nova versão do plano de projeto. Entre os logs leves estão: alterações de responsável pelo plano, data de aprovação, de prazos, projeto associado, prioridade de acesso a ferramentas etc. Entre os logs graves podemos citar: modificação da interdependência das atividades, pré-condições dinâmicas do fluxo de desenvolvimento, comunicação i/o etc. O elemento tipo de *log* de projeto irá especificar as alterações possíveis no plano. O plano do projeto não representa uma instância do plano padrão. As figuras 2 e 3 mostram visões do plano de projeto na primeira versão do *ProjectSpace*.

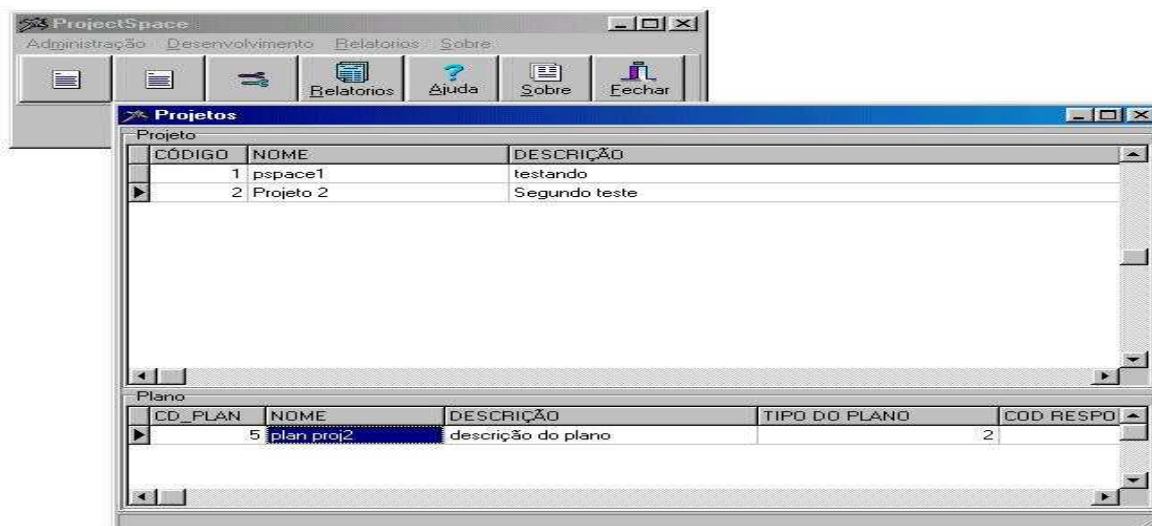


Figura 2 – Visão do cadastramento de projetos e planos associados no ProjectSpace

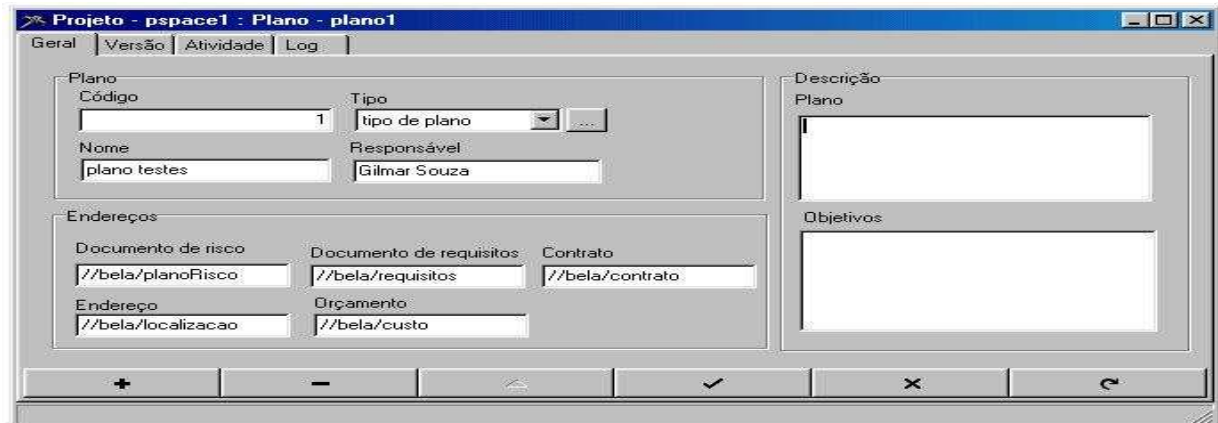


Figura 3 – Visão do cadastramento de plano de projetos, versões, atividades e logs

Uma atividade em plano de projeto no *ProjectSpace* representa uma tarefa atômica que pode ser executada por uma entidade computacional ou humana (ou equipe). Possui um cronograma de execução, restrições, pré-condições, pós-condições, atividades alternativas, ordem de execução (inter-relacionamento), recursos necessários (humanos, ferramentas, programas, etc.), artefatos (de entrada e saída), técnicas ou agrupamento de técnicas associadas, etc. Assim como o plano de projeto, uma atividade do plano pode possuir *logs* leves e graves. Exemplos de *logs* leves são: alteração de prazos, registro de problemas na execução das atividades, alteração de responsável, alteração de executores, alteração de ferramentas associadas, pré e pós-condições estáticas, os requisitos que satisfaz (parcialmente ou integralmente). Exemplos de *logs* graves são: tipo de atividade, o plano a qual pertence, alteração na ordem da execução, pré e pós-condições dinâmicas do fluxo do desenvolvimento. A figura 4 apresenta uma visão do cadastramento das atividades no plano de projeto.

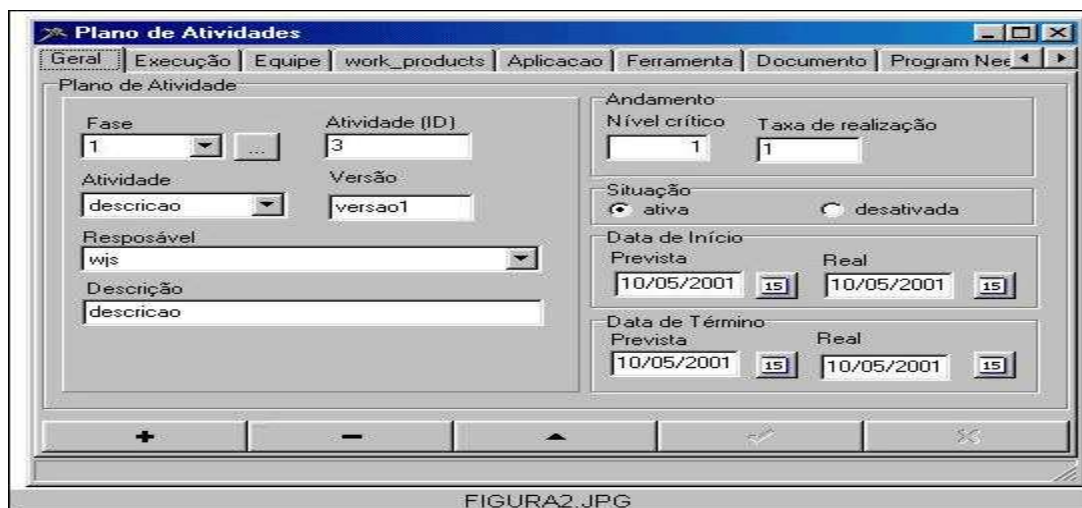


Figura 4 – Visão do cadastramento das atividades plano de projeto

O cadastramento geral das atividades da organização é de responsabilidade da gerência de processo, assim como sua adequação às práticas básicas de modelos e padrões de avaliação e melhoria do processo de software. Uma atividade na organização pode possuir outras atividades que são equivalentes, assim como necessitar de co e pré-requisitos de atividades para sua realização. As atividades também estão associadas às técnicas, às ferramentas, às habilidades, etc. necessárias para sua realização. A figura 4 mostra uma visão do cadastramento das atividades.

A execução de uma atividade em um plano de projeto sempre implica em cadastrar um início e término (que representa dados que serão implementados no *TimeSheet*) assim como a determinação de uma taxa de realização quando de sua finalização. Esta taxa de realização especifica o quanto o

executor acredita ter cumprido a atividade a que se propôs. No caso do executor especificar uma taxa de realização menor que a anterior, ele deve justificar. A figura 5 apresenta uma visão da execução de atividades.

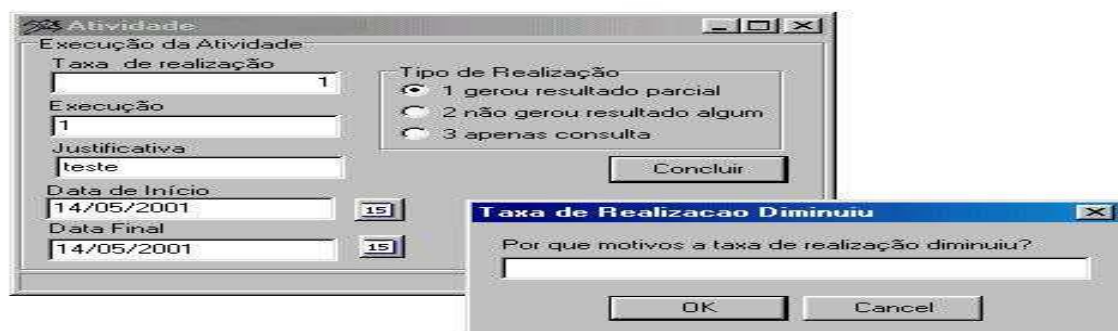


Figura 5 - Visão de apontamento de horas na execução de uma atividade do plano

A execução das atividades no *ProjectSpace* implica em ativação de ferramentas CASE. O ToolManager é o módulo que efetuará o gerenciamento das ferramentas CASE, contendo um conjunto de ferramentas CASE que trabalham juntas a um conjunto comum de dados e protocolos, tendo sido concebida para integrar-se ao *ProjectSpace* e, assim, ser capaz de gerenciar todas as questões inerentes ao uso, registro e controle dessas ferramentas durante a execução das atividades definidas em um plano de projeto. Maiores detalhes deste módulo pode ser visto em [10]. A figura 6 apresenta uma visão do ToolManager.

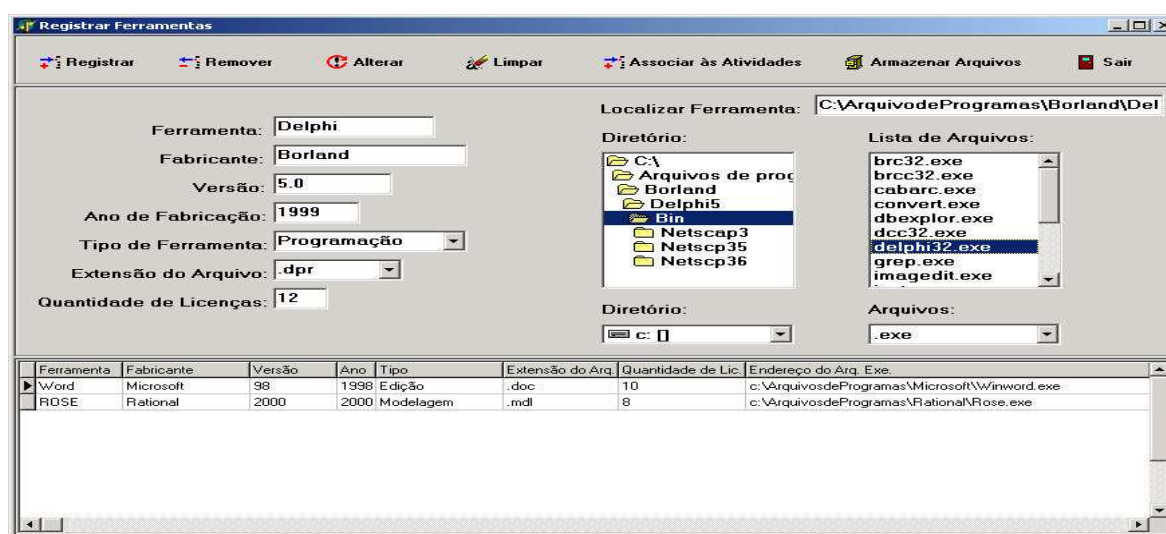


Figura 6 – Visão do cadastramento de ferramentas CASE no ToolManager

Todas as alterações realizadas no plano de projetos desde o nível mais simples até o mais complexo são registradas nesta versão do *ProjectSpace*.

4. Conclusão e Trabalhos Futuros

Atualmente estamos terminando a implementação do *PlanEngine*, do *ToolManager*, do *TimeSheet* e parte do *Data Access Layer*. Este conjunto de módulos iniciais estarão a disposição da Devex e do CESAR a partir do mês de junho quando será dado início a etapa de validação da ferramenta, ajustes e apontamento de melhorias. Os demais módulos do *ProjectSpace* já estão modelados e deverão ser implementados a partir da liberação da primeira etapa do projeto para o CESAR e para a Devex (junho/2001).

Como resultado de uma implementação bem sucedida do *ProjectSpace*, poderemos ter entre outros: o escopo do trabalho do projeto definido; a viabilidade de alcançar as metas do projeto baseada na verificação dos recursos disponíveis e restrições; as tarefas e recursos necessários para completar o trabalho medidas e estimadas; as interfaces entre os elementos do projeto, e com outros projetos e unidades organizacionais identificada e monitorada; os planos para execução do projeto desenvolvidos e implementados; o progresso do projeto monitorado e registrado; as ações para corrigir os desvios do plano e prevenir a recorrência de problemas identificados no projeto; etc.

Algumas métricas também são esperadas tais como: Quão bem distribuídos estão os recursos da organização (jornada de trabalho, espaço físico, licenças de ferramentas CASE, etc.); Quais as habilidades e funções mais comuns na organização; Qual a necessidade de um determinado treinamento; Quais as metodologias, os modelos de ciclo de vida, técnicas, recursos, ferramentas, etc. mais utilizados na organização; Quão maduro um indivíduo ou uma equipe está para utilizar determinada tecnologia; Quais os problemas que mais ocasionam atrasos em projetos; Em que fases há uma maior incidência de problemas nos projetos; Quais os problemas que mais levam a adaptação ou confecção de um novo plano de projeto; Quão bem é estipulado o término das atividades dos projetos; Quanto se realiza as atividades dentro dos prazos estabelecidos; Quanto se cumpre o que foi planejado; Quanto o projeto já evoluiu; Quais os problemas que mais ocorrem no desenvolvimento de sistemas de software; Quanto se cumpre as práticas básicas das normas em um dado projeto ou no desenvolvimento como um todo; etc.

5. Referências Bibliográficas

- [1] W. Gibbs. "Software's chronic crisis". Scientific American, September, 1994.
- [2] Pressman, R. S. Software Engineering: A Practitioner's Approach. McGraw-Hill, 3ª ed., 1992.
- [3] Cromer, T. & Horch, J. From the many to the one-one companys path to standardization. IEEE, 1999.
- [4] Machado, L. F. D. C. Modelo para Definição de Processos de Software na Estação Taba, Tese de M.Sc. COPPE/UFRJ, Rio de Janeiro, RJ.
- [5] Fernandes, Aguinaldo Aragon & Kugler, J. L. C. Gerência de Projetos de Sistemas. Rio de Janeiro, LTC, 1989.
- [6] Gates, L. P. How to Use the Software Process Framework. Special report CMU/SEI-97-SR-009, 1997.
- [7] Maidantchik, C., Rocha, A. R. C. & Xexeo, G. B Software Process Standardization for Distributed Working Groups. In Proceedings of the 4 th IEEE International Software Engineering Standards Symposium, Curitiba, Paraná, Brasil, maio de 1999.
- [8] Page-Jones, M. Gerenciamento de Projetos, São Paulo, Mc Graw-Hill, 1990.
- [9] Rouiller A. C, Meira, S. R. L., Alvarenga, B. B. & Nascimento, L. T. & Lysei, E. Gerenciamento Automatizado de Projetos de Software – Uma Experiência Prática na Devex Tecnologia. XII Conferência Internacional de Tecnologia de Software. Junho, 2001.
- [10] Oliveira, S. R. B., Rouiller, A. C., Vasconcelos, A. M. L. & Meira, S. R. L. *ToolManager*: Uma Camada de Gerenciamento de Ferramentas CASE em um Ambiente Centrado no Processo. XII Conferência Internacional de Tecnologia de Software, Curitiba-PR, Junho-2001.