

Geração Automática do Cronograma de Projeto de Software a partir da Especificação de Requisitos

Erik Alexander Landim RAFAEL (1); Joyce Miranda dos SANTOS (2)

(1) CEFET-AM, Av. Sete de Setembro, 1975 – Centro, Manaus-AM, (92)3621-6768, e-mail: erikrafael@cefetam.edu.br

(2) CEFET-AM, e-mail: joyce_miranda1986@yahoo.com.br

RESUMO

A escassez de tempo e de recursos são dificuldades inerentes ao contexto do planejamento, execução e controle de projetos de software. Identificar quais atividades devem ser executadas, quais os recursos serão requeridos e organizá-los em uma linha de tempo no cronograma de trabalho é responsabilidade do Gerente de Projeto. Neste trabalho, uma solução tecnológica desenvolvida integrando informações para a elaboração do cronograma de trabalho de projetos de engenharia de software é apresentada. Esta pesquisa levantou e desenvolveu um software no qual os requisitos para elaboração do cronograma de trabalho são exportados de uma base de dados relacional desenvolvida em MySQL para um arquivo XML (Extensible Markup Language – Linguagem de Marcação Extensível). Este arquivo XML tem uma estrutura que possibilita o MS-Project da Microsoft importar e produzir um arquivo de projeto com extensão .mpp. Utilizando a abordagem de *metadado*, diferentes informações da base de dados podem ser disponibilizadas para a geração do cronograma do projeto em XML. *Templates* que indicam o tipo de ciclo de vida a ser usado nos cronogramas de projeto também podem ser especificados. A metodologia utilizada no desenvolvimento deste trabalho pode ser descrita em cinco etapas: levantamento das necessidades e especificação de uma base de requisitos, definição da estrutura XML usada pelo MS-Project, definição da estrutura do *metadado*, definição da estrutura dos *templates* e implementação/teste do sistema produzido. Como contribuição oferecida por este trabalho, é possível observar: a criação de uma ferramenta que possibilita manter os requisitos do software a serem desenvolvidos, a combinação dos requisitos com estimativas da realização das atividades de forma a gerar automaticamente o cronograma de trabalho em *templates* de ciclos de vida, a integração de ferramentas gerenciais e a redução do esforço do gerente de projetos na confecção e manutenção de cronogramas do projeto.

Palavras-chave: projeto de software, cronograma, metadado, templates, ferramenta.

1. INTRODUÇÃO

Ao se tentar gerenciar projetos de *software*, diversas dificuldades são encontradas, como por exemplo, a escassez de tempo e de recursos para a execução do projeto. Quando uma estimativa que não considera essas limitações é feita, têm-se como consequência projetos que não cumprem seu planejamento e que extrapolam os prazos e os custos estimados (GRANDCHAMP, 2002). Por esse motivo, a elaboração de estimativas realistas de prazo (cronogramas), recursos e custos são tão importantes para o sucesso de um projeto (GOMES, 1999), em especial, projetos de *software*.

Um cronograma realístico de um projeto é obtido através da análise das atividades definidas para alcançar os objetivos estabelecidos, levando em consideração a duração e os recursos necessários à execução de cada atividade (PMBOK, 2000). Direcionando este contexto para projetos de *software*, essas atividades são identificadas na fase da especificação do escopo do projeto, onde os requisitos levantados direcionam as fases do ciclo de vida do projeto, do seu início à sua conclusão (PRESSMAN, 1995).

Atualmente, existem no mercado aplicações específicas para análise de requisitos e aplicações específicas para gerenciamento de projetos, sendo ambas bastante funcionais, possuindo um alto nível de maturidade e com grande aceitação pelo mercado. Contudo, essas ferramentas, na maioria das vezes, trabalham de forma independente e sem comunicação entre os dados armazenados. A falta de integração entre essas ferramentas causa retrabalho e até mesmo limitações quanto ao uso das informações geradas por parte dos gestores de projetos. Com base nessas informações, foi percebida a necessidade de se buscar uma solução capaz de integrar as informações geradas por esses tipos de ferramentas sem que seja necessária a mudança das tecnologias e ferramentas utilizadas no projeto.

2. ESPECIFICAÇÃO DO PROBLEMA

A idéia de fazer este trabalho surgiu a partir do cenário observado em uma empresa de desenvolvimento de sistemas. Nesta empresa, eram utilizadas duas ferramentas de trabalho distintas em atividades que, se integradas, poderiam trazer uma maior otimização do trabalho realizado. Uma das ferramentas era utilizada para armazenar os dados obtidos na fase de especificação de requisitos e a outra ferramenta era utilizada para realizar a confecção do cronograma do projeto.

No caso da empresa mencionada, os requisitos são armazenados em uma base de dados relacional e, quando existe a necessidade da elaboração do cronograma, os requisitos são transferidos (manualmente) para a ferramenta de gerenciamento de projetos – MS-Project. Esta ferramenta é usada para a geração de cronogramas. Feito o cronograma, caso exista a necessidade de se obter as atividades ali lançadas sob uma visão diferente, se faz necessária a criação de um novo modelo, onde os requisitos precisam ser registrados novamente. Nesta abordagem, não há comunicação entre as ferramentas e também não há um procedimento automatizado para a realização da migração de dados entre elas. Uma vez que os requisitos são passados para a ferramenta de gestão, não existe a possibilidade de mudar a forma de reorganizar ou reordenar as informações que estão na base de dados, sem que isso demande esforço e retrabalho.

3. ESPECIFICAÇÃO DA SOLUÇÃO

Um cronograma realístico é obtido através da análise das atividades definidas para alcançar os objetivos requeridos. Em projetos de *software*, essas atividades são identificadas a partir dos requisitos especificados. Baseando-se nessas duas afirmativas, chegou-se à conclusão de que a integração entre as fases de especificação de requisitos e de gerenciamento de tempo do projeto auxiliaria a obtenção de um cronograma com qualidade e conduziria à redução do esforço envolvido em sua construção e manutenção.

Através da análise do problema descrito, foi verificado que a integração entre as atividades de análise de requisitos e de geração/manutenção do cronograma de projetos de *software* poderia ser obtida através da utilização da linguagem XML. Isto porque, esta linguagem permite a descrição e estruturação de dados, o que possibilita a troca de informações entre plataformas/aplicações distintas (ALMEIDA, 2002). Dentro do contexto deste projeto, o XML é utilizado como a tecnologia que possibilita o transporte dos requisitos armazenados em uma estrutura específica de base de dados relacional (MySQL) para uma estrutura de dados proprietária (MS-Project), oferecendo assim a integração entre as plataformas. Além disso, o XML é utilizado para a criação de metadados e *templates*.

Segundo Ikematu(2001), a principal finalidade dos metadados é documentar e organizar de forma estruturada os dados, com o objetivo de minimizar duplicação de esforços e facilitar a manutenção destes. No sistema, os metadados armazenam a forma como a base de dados de requisitos está estruturada.

A tradução para o termo template mais apropriada ao uso em computação é modelo. Assim sendo, pode-se definir template como um modelo de documento contendo um conjunto de informações de formato padrão usado como base para elaboração de novos documentos. No modelo estão contidas as instruções sobre onde e qual informação deve entrar em cada parte do layout do documento a ser produzido; por exemplo, conteúdos que devem aparecer no início e conteúdos que só devem aparecer no final. Neste trabalho, templates são usados para especificar a forma como o gestor deseja organizar ou ordenar, no MS-Project, as informações no cronograma.

Enfim, a proposta da ferramenta é permitir que o gestor tenha a flexibilidade de escolher quais informações da base de requisitos serão apresentadas no cronograma, bem como, criar a partir das informações escolhidas, *templates* (padrões) para a apresentação dessas informações sob a perspectiva de diferentes ciclos de vida (Figura 1). Será possível aplicar um *template* a diferentes tipos de projetos. Assim, os cronogramas obtidos a partir da aplicação de um *template*, possuirão a mesma organização para a apresentação de suas atividades.

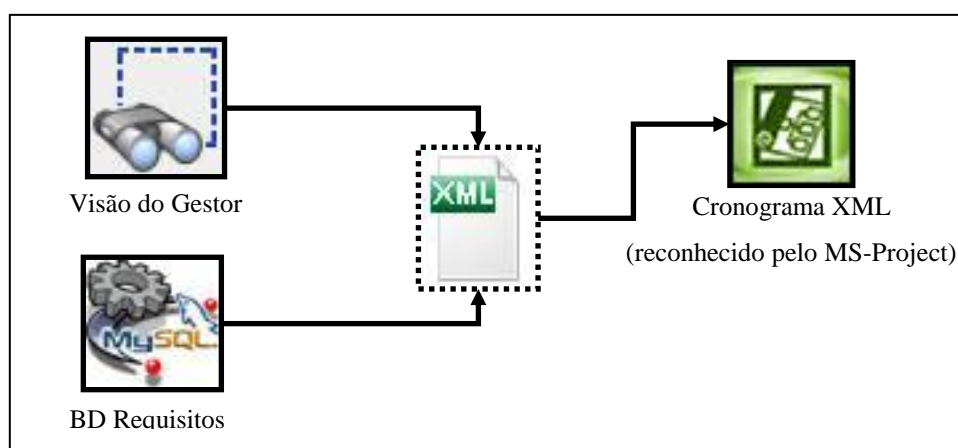


Figura 1: Utilização da tecnologia XML na geração do cronograma

3.1 Processo Utilizado

O processo utilizado para alcançar o objetivo deste trabalho consistiu em cinco etapas, identificadas na Figura 2. Nas próximas seções serão descritas as atividades realizadas em cada etapa.

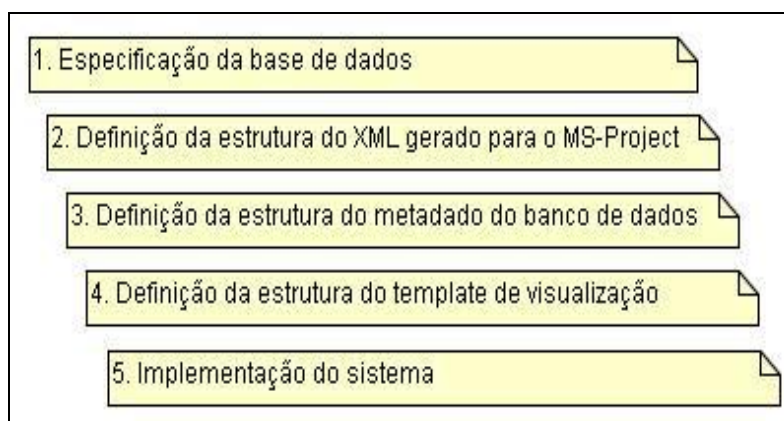


Figura 2: Processo utilizado para a geração do cronograma.

3.1.1 Especificação da Base de Dados

Inicialmente, foi feita a modelagem de uma base de dados fundamentando-se nos conceitos da Engenharia de Requisitos (PRESSMAN, 1995) e do Gerenciamento de Projetos (PMBOK, 2000). A base foi especificada, de forma a conter informações relevantes à atividade de especificação de requisitos de *software*. A relevância desta atividade deve-se ao fato de que, a partir desta base, são extraídas as informações que o gestor irá utilizar para realizar a estimativa de tempo, recursos e custos necessários ao projeto. As informações desta base irão compor o conteúdo do arquivo XML de criação do cronograma do projeto que será reconhecido pela ferramenta MS-Project.

3.1.2 Definição da estrutura do XML gerado para o MS-Project

O MS-Project é um *software* de propriedade da Microsoft, desenvolvido para apoiar a atividade de gerenciamento de projetos, permitindo o planejamento e acompanhamento de tarefas e recursos. O MS-Project possibilita a importação e exportação de dados no formato XML, de forma que outras aplicações possam trabalhar com estes dados (MICROSOFT®).

Uma análise da estrutura do arquivo XML gerado pelo MS-Project foi realizada com o objetivo de identificar a padronização utilizada por ele no armazenamento das informações. Foram identificadas as marcações (*tags*) relevantes, assim como seu significado e as informações nelas contidas. Uma vez realizada esta tarefa, foi possível estabelecer como deveria ser a estrutura mínima do arquivo XML a ser gerado pelo Sistema de Geração de Cronogramas (SGC). Definida a estrutura mínima do arquivo XML, foi analisada quais seriam as relações semânticas entre as informações da base de requisitos e as *tags* XML geradas pelo MS-Project. Foi identificada na base de dados, a origem das informações que serão o conteúdo das *tags* deste arquivo XML. O resultado desta etapa pode ser visualizado na Tabela 1.

Tabela 1 – Estrutura mínima do arquivo XML gerado pelo SGC

Tag	Conteúdo	Exemplo
<UID> <ID>	Número de identificação único da tarefa.	1 (primeira tarefa).
<Name>	Nome da tarefa.	Manter perfis.
<OutlineLevel>	Nível ocupado.	3 (terceiro nível).
<WBS> <OutlineNumber>	Número de identificação de acordo com o nível ocupado.	1.1.2 (3º nível, 2º tarefa).
<Priority>	Prioridade de execução da tarefa.	1.
<ConstraintType>	Identificação do tipo de calendário.	0 ou 1.
<IsSubproject>	Define se é ou não subprojeto.	0 ou 1.
<FixedCost>	Identifica se é custo fixo.	0 ou 1.
<FixedCostAccrual>	Campo fixo para custo de projetos simples	3.
<Notes>	Descrição da tarefa.	Cadastro, alteração e exclusão de recursos dos projetos.

3.1.3 Definição da estrutura do metadado do banco de dados

Nesta etapa, foi estudada a forma como o SGC permitiria a flexibilidade para o gestor escolher as informações que seriam colocadas no cronograma a ser gerado e como elas seriam organizadas ou ordenadas no layout do cronograma do projeto. Esta etapa enfrentou uma problemática que foi a seguinte: como permitir essa flexibilidade ao gestor, implementando restrições, sem que isso implicasse na reprogramação da geração do cronograma (implementação de um novo código fonte) cada vez que a base de dados fosse alterada ou que uma outra forma de apresentação fosse desejada pelo gestor?

Como proposta de solução para este problema, foi escolhido utilizar o conceito de metadados e XML *Schema*. *Metadados* são definidos como ‘*dados sobre dados*’, ou seja, um repositório estruturado de propriedades, características e informações sobre um ou mais recursos. Por exemplo, os metadados poderiam indicar se uma base de dados existe em uma corporação, quais atributos compõem uma determinada tabela e quais são as características (tamanho e formato) de certo atributo (SANTOS, 2003). O sistema trabalha com dois padrões de metadado. O primeiro define as informações contidas no banco de dados, o relacionamento entre essas informações e qual a ligação existente entre estas informações e as *tags* do XML (MS-Project). Este metadado poderá ser definido uma única vez. Caso haja mudanças na estrutura do banco de dados, como por exemplo, nome das tabelas ou colunas, inserção de novos dados, não será necessário alterar o código fonte que produz o cronograma. O segundo (template de apresentação) define como as informações serão estruturadas visualmente no cronograma.

O XML *Schema* tem como propósito definir e descrever o conteúdo de documentos XML através da utilização dos componentes da linguagem (XML) para restringir e documentar o significado, a utilização e as relações entre as partes constituintes do documento: tipos de dados, elementos e seu conteúdo, atributos e valores. Em resumo, um XML *Schema* define um vocabulário para uma classe de documentos XML (SAUDE, 2003). No sistema, para cada metadado, foram definidos XML *Schemas* cujo conteúdo define a estrutura que os metadados escritos em XML devem possuir para serem validados e utilizados pelo sistema. O resultado desta etapa pode ser consultado na Tabela 2.

Tabela 2 – Conteúdo do metadado do banco de dados

Tag	Conteúdo	
<table>	Identificação da tabela do banco de dados.	
	<id>	“Alias” a ser usado na construção das consultas SQL.
	<name>	Nome da tabela.
<primaryKey>	Nome do campo que consiste a chave primária da tabela.	
<project>	Nome do campo da tabela que identifica o projeto para o qual o cronograma será gerado.	
<name>	Nome do campo de onde serão extraídas as informações que irão popular a tag <Name> do arquivo XML gerado pelo SGC	
<notes>	Nome do campo de onde serão extraídas as informações que irão popular a tag <Notes> do arquivo XML gerado pelo SGC	
<priority>	Nome do campo de onde serão extraídas as informações que irão popular a tag <Priority> do arquivo XML gerado pelo SGC	
<fields> / <item>	Nome dos campos (informações) que poderão ser trabalhados pelo gestor, em termos de apresentação. Ex: seleção e ordenação.	
	<value>	Texto que será mostrado ao gestor como informação que ele poderá escolher.

<tblRelated>	Nome da tabela que se relaciona à tabela especificada na tag <table> deste documento.	
	<select>	Definição da consulta SQL de relacionamento.
	<from>	
	<where>	
	<group>	

3.1.4 Definição da estrutura do *template* de apresentação

O *template* de apresentação contém a definição de quais informações serão utilizadas e como estas serão organizadas no cronograma. O conteúdo deste arquivo pode ser definido pelo gestor. Desta forma, foi especificada uma estrutura que suportasse o conteúdo de informações relevantes para a apresentação do cronograma. Este *template*, juntamente com o metadado do banco de dados, são utilizados pelo SGC para montar a estrutura XML do cronograma a ser reconhecido pelo MS-Project. A aplicação combinará as *tags* XML, definidas pela Microsoft para exportação e importação de arquivos, com os dados extraídos do banco de dados. Estes dados são buscados na base de dados a partir de consultas SQL montadas de acordo com as informações contidas no metadado do banco de dados e no *template* de apresentação. O resultado desta etapa pode ser consultado na Tabela 3.

Tabela 3 – Conteúdo do *template* de apresentação

Tag	Conteúdo	
<metaDado>	Nome identificador do <i>template</i> .	
<descricao>	Descrição curta de o que se trata o <i>template</i> .	
<task>	Identifica as informações e a forma como estas serão apresentadas no cronograma.	
	<level>	Identifica o nível de apresentação no cronograma. Ex: 1º nível corresponde à primeira informação que será apresentada.
	<fixedText>	Texto fixo a ser apresentado anteriormente a cada tarefa, pode ser utilizado como identificador da informação que está sendo apresentada. Ex: Requisito.
	<type>	Identifica a origem do dado que está sendo apresentado. Pode ser de dois tipos: banco de dados ou tarefa avulsa (não está associada à base de dados).
	<table>	Caso seja de origem do banco de dados, identifica o nome da tabela de onde a informação será extraída.
	<condition>	Condição de seleção da informação. Ex: prioridade maior que 100.
	<order>	Condição de apresentação da tabela. Ex: ordem crescente de prioridade.

3.1.5 Implementação do sistema

O sistema desenvolvido apresenta funcionalidades para a geração automática dos metadados: de banco de dados e de apresentação. Além das rotinas de geração dos metadados, o sistema realiza a geração do arquivo XML do cronograma do projeto a partir dos metadados criados. A implementação da rotina para geração do cronograma consiste nas seguintes etapas (Figura 3):

- Verificação da existência do metadado do banco de dados definido (Figura 3. Item 1);
- Verificação da existência do *template* de apresentação definido (Figura 3. Item 2);
- Validação da estrutura dos metadados: no código implementado existe um procedimento onde é utilizado o XML *Validator* que verifica se a estrutura e dados contidos nesses arquivos estão de acordo com a definição imposta no XML *Schema* criado (Figura 3. Item 3.1);
- Geração das consultas SQL através da associação das informações contidas em ambos metadados. Exemplo: no *template* de apresentação, o gestor escolheu organizar no cronograma os Módulos do sistema associados aos Casos de Uso, nessa ordem. Primeiramente, a rotina verifica no metadado do banco de dados quais são as informações existentes sobre Módulos. No metadado estarão as informações da tabela e campos referentes a Módulos que deverão ser buscados para preencher o arquivo XML. A partir dessas informações, o sistema montará as consultas SQL para mostrar estas informações. Como neste exemplo os Módulos devem ser mostrados com seus respectivos Casos de Uso, o sistema irá verificar no metadado do banco de dados qual o relacionamento existente entre Módulos e Casos de Uso e a partir das informações obtidas irá criar as consultas SQL;
- Consulta ao banco de dados de requisitos: A partir das consultas SQL criadas o sistema busca as informações no banco de dados (Figura 3. Item 3.2);
- O sistema salva as informações em um arquivo XML no padrão exigido pelo MS-Project (Figura 3. Item 4).

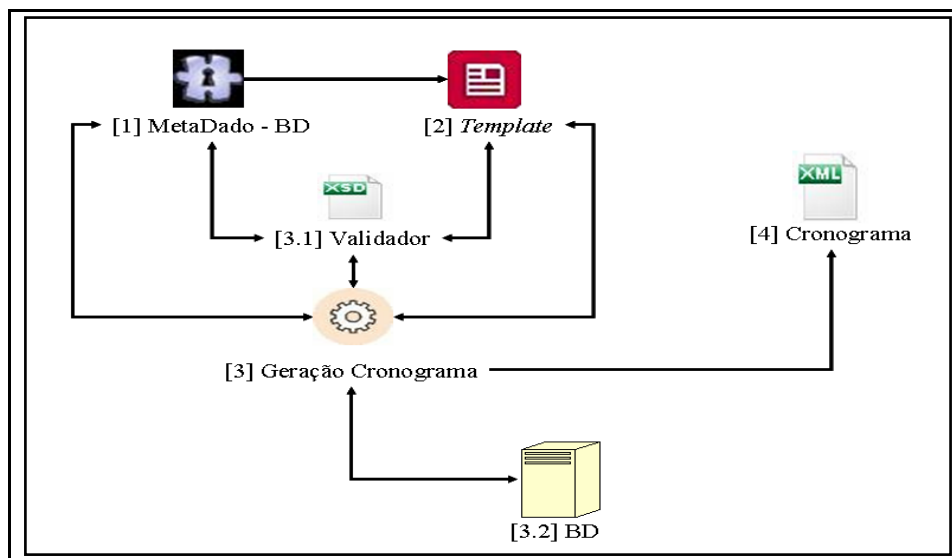


Figura 3 – Esquema da rotina de geração do XML do cronograma

4. CONCLUSÃO

A existência de diversas dificuldades na execução de projetos de *software* faz com que diversos empreendimentos neste segmento de atividade não sejam finalizados com o sucesso esperado em sua fase de inicialização. Como em *software*, os requisitos levantados direcionam as fases do projeto, este trabalho se propôs a interligar as atividades de especificação de requisitos com a elaboração de cronograma. A pesquisa feita neste trabalho resultou em uma aplicação que realiza esta integração através da tecnologia XML, exportando dados de uma base de dados relacional para uma ferramenta que possibilita a geração de

cronogramas de projeto. Além de permitir essa integração, a aplicação permite a criação de *templates* de apresentação para a aplicação em diferentes escopos de projeto. Assim, pôde-se concluir que os objetivos do trabalho foram alcançados.

Contudo, algumas limitações foram percebidas ao final do desenvolvimento da aplicação, o que possibilita a realização de novos trabalhos futuramente. Uma limitação existente é a restrição de permitir apenas três níveis na apresentação do cronograma, essa limitação foi imposta devido ao fato de que à medida que o número de níveis de detalhes do cronograma aumenta, mais complexo fica o procedimento de geração das consultas SQL. Uma outra limitação identificada, no qual se pode propor melhorias, diz respeito a atualizações automáticas, ou seja, uma vez detectadas modificações feitas no cronograma exportado pelo MS-Project, uma aplicação permitiria que essas modificações fossem refletidas no banco de dados de requisitos nos atributos que possuem correlação com os dados do cronograma. De outra forma, também seria desejável que modificações feitas no banco de dados de requisitos fossem refletidas ou combinadas com um XML importado e, portanto, um novo XML poderia ser gerado, resultado de um merge de atualizações realizadas.

REFERÊNCIAS

ALMEIDA, Maurício Barcellos. **Uma introdução ao XML, sua utilização na Internet e alguns conceitos complementares**. Revista Ciência da Informação, Brasília, v. 31, n. 2, p. 5-13, maio/ago. 2002.

GOMES, A. E., 1999, “**Métricas e Estimativas de Software: o Início de um Rallye**”, Revista Developers Magazine, n. 39, pp. 50-53.

GRANDCHAMP, R. E. **Gerenciamento de Projetos de Software**. 2002. 82 f. Monografia (Especialização – MBA em Gerência de Produção e Tecnologia) Departamento de Economia, Contabilidade, Administração e Secretariado, Universidade de Taubaté, Taubaté, 2002.

IKEMATU, R. S. **Gestão de metadados: sua evolução na tecnologia da informação**. DataGramaZero - Revista de Ciência da Informação, 2(6), IASI - Instituto de Adaptação e Inserção na Sociedade da Informação. 2001.

PMBOK 2000. Tradução livre disponibilizada através da Internet pelo PMI MG em janeiro de 2002 do **A Guide to the Project Management Body of Knowledge (PMBOK® Guide)**; 2000 Edition, V 1.0. Disponível em: <<http://www.pmimg.org.br>>. Acesso em: 25 set. 2007.

PRESSMAN, Roger S. **Engenharia de Software** / Tradução: José Carlos Barbosa dos Santos; revisão técnica: José Carlos Maldonado, Paulo César Masiero, Rosely Sanches. São Paulo: Makron Books, 1995.

SANTOS, A. C. **Introdução a metadados**. Revista CienteFico, Salvador, v.2. Faculdade Ruy Barbosa. 2003.

SAUDE, A. V. **Gerador automático de editores XML baseado no XML Schema**. Dissertação (Mestrado) Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, 2003.

MICROSOFT®. **Microsoft Office Online – Microsoft Office Project**. Disponível em:

< <http://office.microsoft.com/pt-br/project/CH010220021046.aspx>>. Acesso em: 8 ago. 2008.