

**UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ**  
**CAMPUS LUIZ MENEGHEL**

**LÉLIS GETÚLIO YAMAGUTI LIMA**

**ENGENHARIA DE REQUISITOS APOIADA POR  
MODELAGEM DE PROCESSOS DE NEGÓCIOS**

Bandeirantes

2011





**UNIVERSIDADE ESTADUAL DO NORTE DO PARANÁ**

***CAMPUS LUIZ MENEGHEL***

**LÉLIS GETÚLIO YAMAGUTI LIMA**

**ENGENHARIA DE REQUISITOS APOIADA POR  
MODELAGEM DE PROCESSOS DE NEGÓCIOS**

Bandeirantes

2011

**LÉLIS GETÚLIO YAMAGUTI LIMA**

**ENGENHARIA DE REQUISITOS APOIADA POR  
MODELAGEM DE PROCESSOS DE NEGÓCIOS**

Trabalho de Conclusão de Curso apresentado ao curso de Sistemas de Informação da Universidade Estadual do Norte do Paraná - *Campus* Luiz Meneghel, como requisito para obtenção do grau de Bacharel em Sistemas de Informação, orientado pelo Prof. José Reinaldo Merlin.

Bandeirantes

2011

**LÉLIS GETÚLIO YAMAGUTI LIMA**

**ENGENHARIA DE REQUISITOS APOIADA POR  
MODELAGEM DE PROCESSOS DE NEGÓCIOS**

Trabalho de Conclusão de Curso apresentado ao curso de Sistemas de Informação da Universidade Estadual do Norte do Paraná - *Campus* Luiz Meneghel, como requisito para obtenção do grau de Bacharel em Sistemas de Informação, orientado pelo Prof. José Reinaldo Merlin.

**COMISSÃO EXAMINADORA**

---

Prof. José Reinaldo Merlin.  
Universidade Estadual do Norte do  
Paraná – *Campus* Luiz Meneghel

---

Prof<sup>a</sup>. Me Daniela de Freitas G. Trindade.  
Universidade Estadual do Norte do  
Paraná – *Campus* Luiz Meneghel

---

Prof. Carlos Eduardo Ribeiro.  
Universidade Estadual do Norte do  
Paraná – *Campus* Luiz Meneghel

Bandeirantes, \_\_\_\_ de \_\_\_\_\_ de 2011.

## **DEDICATÓRIA**

*A você, que seja por qualquer motivo está em posse deste trabalho de conclusão de curso.*

## **AGRADECIMENTOS**

A Deus por toda incondicionalidade. A meu avô Getúlio Salles Bispo, avó Ivani Yamaguti Salles e mãe Marcela Yamaguti Salles que por nenhum momento duvidaram de minhas escolhas e participaram ativamente de todo o caminho percorrido sem medir esforços, a minha irmã Fernanda Yamaguti Tanaka por toda felicidade sincera que uma criança pode proporcionar. Vocês são as pessoas mais importantes da minha vida.

Aos professores do curso de Sistemas de Informação, em especial ao meu professor orientador José Reinaldo Merlin durante estágio e realização deste trabalho; também aos professores da comissão examinadora, Me. Daniela e Biluka pelas sugestões para enriquecimento do trabalho. A todos da minha família pela contribuição direta ou indireta, ao Paulo Tanaka que antes de qualquer coisa é meu amigo, aos grandes amigos que conheci durante essa jornada: Mateus, Saulo e Jéssica, significativos pré, durante e pós Rep. Caverna; a todos os companheiros de estágio no LAS, em especial Bruno e Jaime; aos integrantes nada convencionais da Rep. Thor oh Chinelo (Anderson, Andrei, Felipe, Ivan, João, Jonatã, Murilo, Rafael Castilho, Rafael Paixão e Renan); a Juliana e sua família pela significativa participação em minha vida; a todos da XIII turma de S.I. pela batalha conjunta; aos grandes amigos de Paraguaçu Paulista: Bruno, Celso, Guilherme, Juliano e Leonardo; e a todas as pessoas que de alguma maneira contribuíram para essa vitória.

Minha eterna gratidão a todos vocês.

*“Veni, vidi, vici.”*  
*(Júlio César)*

## RESUMO

A definição de requisitos de um software pode ser elaborada a partir das informações existentes sobre os processos de negócio da organização. Este trabalho realiza um levantamento bibliográfico sobre esses dois temas, enfatizando a intersecção existente entre eles, realizando a atividade de definição de requisitos por meio do modelo de processos de negócio. Uma abordagem é proposta para a aplicação em um estudo empírico, durante o desenvolvimento de um software. O completo entendimento do processo de negócio apoiou a fase de engenharia de requisitos por meio da Modelagem dos Processos de Negócio utilizando-se da notação BPMN. Analisou-se as contribuições que a modelagem de negócios pode trazer para a Engenharia de Requisitos.

**Palavras-chave:** Engenharia de Requisitos. Modelagem de Processos de Negócio. BPMN.

## **ABSTRACT**

The software requirements definition can be prepared from existing information about organization's business processes. This academic work does a literature review about these two themes, emphasizing the intersection between them, realizing the requirements definition through business processes model. An approach is proposed for application in an empirical study, during the software development. The complete understanding of the business processes supported the requirements engineering phase through the Business Processes Modeling using the BPMN notation. It examined what that business modeling can bring to the Requirements Engineering.

**Keywords:** Requirements Engineering. Business Processes Modeling. BPMN.

## LISTA DE FIGURAS

Figura 1: Foco da temática do trabalho.....	16
Figura 2: Processo de ER (Adaptado de Spínola e Ávila, 2007). ....	22
Figura 3: Hierarquia de Requisitos (Adaptado de Knight, 2004).....	23
Figura 4: Resumo SRS (Adaptado da IEEE Std 830-1998). ....	26
Figura 5: Diagramas UML 2.3 (Adaptado de OMG UML, 2010). ....	28
Figura 6: Exemplo de BPD (Adaptado de Valle; Oliveira, 2009).....	33
Figura 7: BPD Registro Novo Projeto Pesquisa. ....	36
Figura 8: Nome diagrama Casos de Uso.....	39
Figura 9: Atores Casos de Uso. ....	40
Figura 10: Casos de uso do diagrama. ....	41
Figura 11: Associações do diagrama. ....	42
Figura 12: Gateway exclusivo caracterizando relacionamento <i>extend</i> . ....	42
Figura 13: Gateway inclusivo caracterizando relacionamento <i>extend</i> . ....	43
Figura 14: Eventos de mensagem e caso de uso. ....	44
Figura 15: Diagrama de casos de uso gerado a partir do BPD. ....	45
Figura 16: Classe Processo. ....	46
Figura 17: Classe Projeto Pesquisa. ....	46
Figura 18: Classe Tarefa. ....	46
Figura 19: Classes derivadas das <i>lanes</i> . ....	47
Figura 20: Classe Formulários de Ética. ....	48
Figura 21: Classe Mensagem.....	48
Figura 22: Atributos e operações de tempo. ....	49
Figura 23: Atributos de status.....	49
Figura 24: Diagrama de classes gerado a partir do BPD. ....	50

# LISTA DE QUADROS

Quadro 1: Fluxograma de Registro de Projeto de Pesquisa .....	34
---	----

## LISTA DE SIGLAS

BPD	<i>Business Process Diagram</i>
BPMI	<i>Business Process Management Initiative</i>
BPMN	<i>Business Process Modeling Notation</i>
BPMS	<i>Business Process Management Systems</i>
EPN	Engenharia de Processos de Negócio
ER	Engenharia de Requisitos
ES	Engenharia de Software
OMG	<i>Object Management Group</i>
OO	Orientado a Objeto
PROPG	Pró-Reitoria de Pesquisa e Pós-Graduação
SI	Sistemas de Informação
SRS	<i>Software Requirements Specification</i>
UML	<i>Unified Modeling Language</i>

# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>15</b>
1.1	FORMULAÇÃO DO PROBLEMA	16
1.2	OBJETIVO	17
1.3	JUSTIFICATIVA	17
1.4	MÉTODO DE PESQUISA	17
1.5	ESTRUTURA DO TRABALHO	18
<b>2</b>	<b>ENGENHARIA DE SOFTWARE</b>	<b>19</b>
2.1	ENGENHARIA DE SOFTWARE E SISTEMAS DE INFORMAÇÃO	20
2.2	ENGENHARIA DE REQUISITOS	21
2.2.1	<i>Requisitos</i>	22
2.2.2	<i>Identificação de Requisitos</i>	24
2.2.3	<i>Análise e Negociação</i>	25
2.2.4	<i>Especificação e Documentação</i>	25
2.2.5	<i>Validação</i>	26
2.2.6	<i>Gerenciamento de Requisitos</i>	26
2.3	UNIFIED MODELING LANGUAGE - UML	27
2.3.1	<i>Extensão da UML para Modelagem de Negócio</i>	28
<b>3</b>	<b>ENGENHARIA DE PROCESSOS DE NEGÓCIO</b>	<b>30</b>
3.1	PROCESSOS DE NEGÓCIO	30
3.2	MODELAGEM DE PROCESSOS DE NEGÓCIO	31
3.2.1	<i>Business Process Modeling Notation - BPMN</i>	32
<b>4</b>	<b>MODELANDO UM PROCESSO DE NEGÓCIO</b>	<b>34</b>
4.1	APLICAÇÃO DA MODELAGEM	35
4.2	CONSIDERAÇÕES FINAIS	37
<b>5</b>	<b>TRANSIÇÃO MODELAGEM BPMN PARA UML</b>	<b>38</b>
5.1	ANÁLISE DE TRANSIÇÃO BPD X CASOS DE USO	39
5.1.1	<i>Geração do diagrama de Casos de Uso</i>	44
5.2	ANÁLISE DE TRANSIÇÃO BPD X DIAGRAMA DE CLASSES	45
5.2.1	<i>Geração do diagrama de Classes</i>	49
5.3	CONSIDERAÇÕES FINAIS	50
<b>6</b>	<b>CONSIDERAÇÕES FINAIS</b>	<b>52</b>
6.1	POSSIBILIDADES DE TRABALHOS FUTUROS	53
	REFERÊNCIAS	54

# 1 INTRODUÇÃO

Durante o processo de desenvolvimento de software, a Engenharia de Requisitos (ER) é a fase responsável pela descrição e especificação de um sistema. Atualmente a relação entre tecnologia da informação e os processos de negócio de uma organização está em ampla discussão. É crescente o uso de sistemas de informação para apoiar os processos de negócio, mas para que esses sistemas sejam desenvolvidos de maneira eficaz, o entendimento do processo de negócio é fator determinante para o sucesso da fase de ER. (CARVALHO, 2009).

Segundo Storch e Pêsoa (2008), sistemas workflow<sup>1</sup> foram os que deram origem aos conceitos de *Business Process Management Systems* (BPMS), que são sistemas que automatizam a gestão de processos de negócio. Alguns pontos de sistemas workflow são importantes de serem citados:

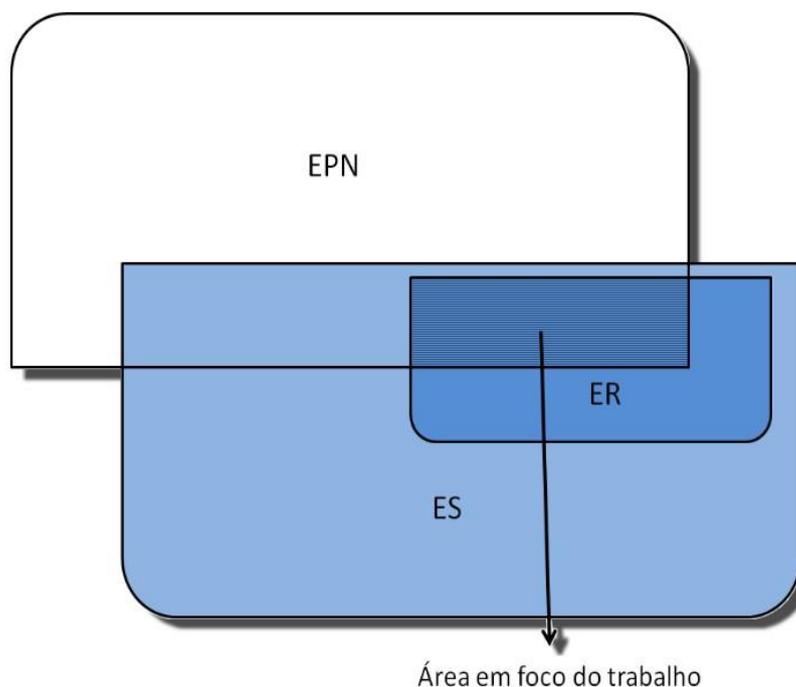
- Automatização de formulários;
- Uso de imagens digitalizadas, permitindo o gerenciamento de processos sem utilização de papel; e
- Regras de negócio que guiavam as alternativas de decisão dos operadores e permitiam automatizar alguns tipos de decisões mais triviais.

Muitas são as abordagens de identificação de requisitos para os sistemas de informação, mas poucas abordam as contribuições da Engenharia de Processos de Negócios (EPN) com técnicas de modelagem de processos de negócio para facilitar a comunicação entre analistas e clientes (KNIGHT, 2004). Essas abordagens são de valores motivacionais para se realizar estudos na área, visando contribuir para uma das etapas mais cruciais do desenvolvimento de sistemas de informação: a Engenharia de Requisitos.

---

<sup>1</sup> Sistemas workflow são projetados para auxiliar grupos de pessoas a realizar seus procedimentos de trabalho, armazenando o conhecimento de como o fluxo de trabalho deve ocorrer. São sistemas que auxiliam as organizações a especificar, executar e coordenar o fluxo de trabalho em determinado ambiente. (ELLIS, 1995 apud THOM, 2002).

Portanto, o presente trabalho possui o foco nestes temas citados, podendo ser visualizado na Figura 1 a intersecção dos mesmos.



**Figura 1: Foco da temática do trabalho.**

## 1.1 FORMULAÇÃO DO PROBLEMA

As soluções de Sistemas de Informação (SI) são utilizadas cada vez mais frequentemente por organizações para o gerenciamento do negócio (DIAS et al., 2006). A dificuldade, no entanto, é a falta de um alinhamento entre o domínio de negócio com o domínio do sistema que lhe fornece suporte (ODEH; KAMM, 2003; SHEN, 2004).

Segundo alguns autores, (KNIGHT, 2004; YU, 1995; BUBENKO, 1993; SILVEIRA et al., 2002; CAMPOS e SANTOS, 2001), existe a necessidade de integrar a visão organizacional por meio de seus processos de negócios para a melhor definição de requisitos que fornecerão subsídios para o desenvolvimento dos sistemas de informação.

Portanto, faz-se necessário investigar procedimentos que permitam ao engenheiro de software desenvolver um produto baseado no correto entendimento dos processos de negócio do cliente.

## **1.2 OBJETIVO**

O objetivo deste trabalho é realizar um estudo empírico sobre Engenharia de Requisitos baseada em modelos definidos pela Engenharia de Processo de Negócios.

## **1.3 JUSTIFICATIVA**

Entregar produtos de qualidade, no tempo prometido e dentro dos valores planejados é uma preocupação constante para os profissionais de Tecnologia da Informação (TI). A velocidade das transformações no mercado reflete em um consumidor de software exigente, cuja necessidade de sobrevivência de seu negócio encurta os prazos de entrega e reduz a disposição orçamentária dos projetos de software.

Para o cliente, software com qualidade é o que satisfaz as necessidades do negócio. Partindo dessa premissa, justifica-se a importância da identificação de requisitos realizada a partir do modelo de negócio, assim podendo ser desenvolvido um sistema que esteja alinhado com os objetivos da organização.

Uma engenharia de requisitos bem feita depende do completo entendimento do negócio do cliente, o que é uma grande dificuldade por parte dos analistas, que nem sempre entendem este domínio. Nesse sentido, um modelo de processo de negócio traz muitos benefícios, principalmente na comunicação de clientes com engenheiros de software, por se tratar de um modelo visual, justificando a importância de realizar a exploração dos métodos e abordagens de identificação de requisitos orientada por processos da organização.

## **1.4 MÉTODO DE PESQUISA**

Essa pesquisa é classificada como exploratória na medida em que contribui para aumentar os conhecimentos sobre o tema. A pesquisa será executada por meio de revisão de literatura sobre métodos de modelagem de processo de negócio seguido de um estudo empírico o qual se verifica a aplicação desses modelos durante a engenharia de requisitos, visando resultados qualitativos.

## 1.5 ESTRUTURA DO TRABALHO

O presente trabalho está dividido em seções como segue:

- Na seção 1, é feita uma breve introdução e levantado o problema a ser discutido neste trabalho com as devidas justificativas de realização do mesmo, expondo os objetivos a serem alcançados e estruturação lógica do trabalho;
- Na seção 2 e 3, faz-se a fundamentação teórica, abordando os principais conceitos existentes na área de Engenharia de Requisitos (ER) contida na Engenharia de Software (ES), também na área de Modelos de Processos de Negócio dentro da Engenharia de Processos de Negócio (EPN).
- A seção 4 constitui a base para o desenvolvimento do trabalho, sendo selecionado um processo e realizado a modelagem do mesmo com a técnica selecionada para este trabalho.
- Na seção 5 é apresentado o desenvolvimento da abordagem proposta para o trabalho, por meio de uma análise sobre o modelo realizado na seção 4.
- Por fim, a seção 6 apresenta a conclusão do trabalho.

## 2 ENGENHARIA DE SOFTWARE

O termo “Engenharia de Software” não possui uma definição universal e/ou única. Mas existem algumas definições bem aceitas da literatura, uma delas é de Maffeo (1992):

Engenharia de Software é a área interdisciplinar que engloba vertentes tecnológica e gerencial, visando abordar de modo sistemático os processos de construção, implantação e manutenção de produtos de software com qualidade assegurada, segundo cronogramas e custos previamente definidos.

A *interdisciplinaridade*, a que se refere o autor citado, provém dos fundamentos de:

- Ciência da Computação: disciplina que deve prover os fundamentos científicos da ES, assim como a Física e Química provêm os fundamentos para as disciplinas de Engenharia tradicionais;
- Administração de Projetos: essa por sua vez provê os fundamentos do gerenciamento de projetos de software, incluindo as atividades dos planejamentos envolvidos neste processo;
- Comunicação: muito importante para a ES e considerada crucial para o engenheiro de software, o mesmo deve possuir habilidades de comunicação interpessoal, tanto oral quanto escrita; e
- Técnica de Solução de Problemas: o engenheiro de software pode ser considerado um solucionador de problemas, um gestor de soluções inteligentes e integradas, utilizando-se de técnicas para prover a base para as atividades de planejamento, gerenciamento, análise sistêmica, *design*, fabricação criteriosa, implantação controlada, validação extensiva e manutenção contínua.

Outra definição é que a Engenharia de Software é uma disciplina de Engenharia que abrange todos os aspectos relacionados à produção de software, desde os primeiros estágios até a manutenção do software, após estar em operação (SOMMERVILLE, 2003).

Para Rezende (2002), Engenharia de Software constitui-se de uma metodologia para desenvolvimento e manutenção de sistemas modulares, possuindo as seguintes características:

- Processo dinâmico, integrado e inteligente de soluções tecnológicas;
- Alinhamento com requisitos funcionais do negócio do cliente;
- Efetivação de padrões de qualidade e produtividade em suas atividades e produtos;
- Fundamentação na TI disponível, viável e oportuna; e
- Planejamento e gestão de atividades, recursos, custos e datas.

As definições citadas levam em conta aspectos gerenciais do tema em questão, o que algumas outras definições acabam omitindo e concentrando-se apenas no aspecto tecnológico do problema, porém a vertente gerencial deve ter um destaque cada vez maior na disciplina de ES.

## **2.1 ENGENHARIA DE SOFTWARE E SISTEMAS DE INFORMAÇÃO**

Primeiramente deve-se definir isoladamente cada conceito contido em Sistemas de Informação (SI). Sistema pode ser definido como conjunto de partes que interagem entre si, integrando-se para atingir um objetivo ou resultado; e Informação é todo o dado trabalhado, útil, tratado, com valor significativo atribuído ou agregado a ele (REZENDE, 2002).

Ainda segundo Rezende (2002), todo sistema que manipule informação pode ser genericamente considerado um Sistema de Informação, independentemente se usa ou não recursos de Tecnologia da Informação (TI).

O SI com TI por sua vez pode tratar de maneira eficiente e eficaz grande volume de informação. No presente trabalho o termo “Sistemas de Informação” será usado de maneira a englobar o SI com utilização de TI, o que implica na utilização da disciplina de Engenharia de Software para seu desenvolvimento e manutenção.

Assim sendo, os SIs são os habilitadores do negócio de uma organização e, portanto, precisam estar alinhados com os objetivos deste negócio para que seja eficaz em sua totalidade (AZEVEDO JUNIOR E CAMPOS, 2008).

## 2.2 ENGENHARIA DE REQUISITOS

Segundo Pressman (2002), uma questão difícil de obter uma resposta é como que se pode garantir que foi especificado um sistema que atenda adequadamente às necessidades e satisfaça às expectativas dos clientes, pois não existe uma resposta concreta para tal. A melhor solução de que se dispõe atualmente é uma sólida Engenharia de Requisitos (ER).

A ER é um processo que envolve todas as atividades exigidas para criar e manter o Documento de Requisitos de software, sendo este o principal artefato trabalhado neste processo (SOMMERVILLE, 2003).

Existem quatro atividades genéricas de alto nível para este processo, são elas (SOMMERVILLE, 2003):

- Identificação;
- Análise e negociação;
- Especificação e Documentação; e
- Validação.

Pressman (2002) considera mais uma atividade que faz parte deste processo, a Gestão de Requisitos, pois os requisitos podem sofrer alteração dos mais diversos fatores, desde inovação tecnológica a mudanças na natureza do negócio, o que os afeta diretamente.

Alguns dos principais objetivos da ER, segundo Leite e Doorn (2003), são:

- Estabelecer uma visão comum entre o cliente e a equipe de projeto em relação aos requisitos que serão atendidos pelo projeto de software;
- Registrar e acompanhar requisitos ao longo de todo o processo de desenvolvimento;
- Documentar e controlar os requisitos alocados para estabelecer uma *baseline* para uso gerencial e da engenharia de software; e
- Manter planos, artefatos e atividades de software consistentes com os requisitos alocados.

Para alcançar esses objetivos, é importante a adoção de um processo de engenharia de requisitos bem definido, o que muitas organizações não possuem, lembrando que uma organização pode desenvolver um processo adequado à sua realidade.

Um modelo genérico de processo de Engenharia de Requisitos está representado na Figura 2.

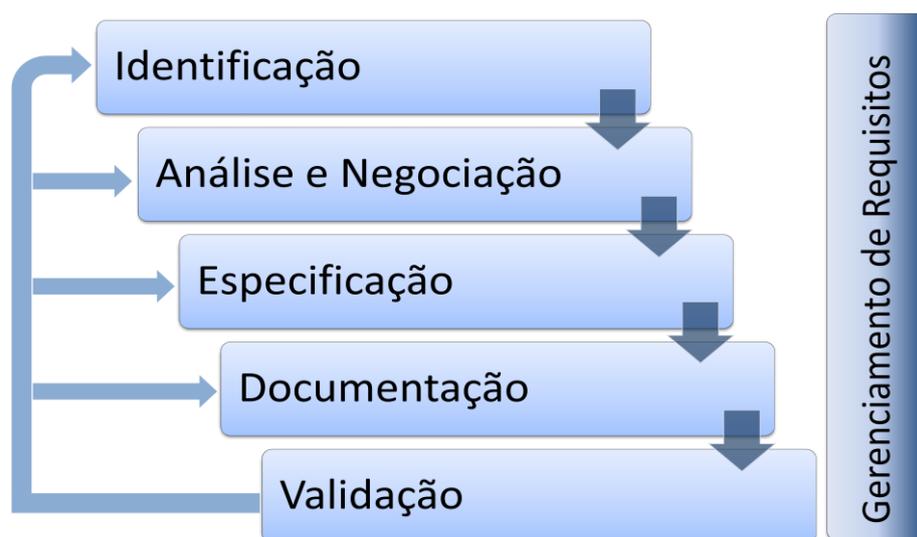


Figura 2: Processo de ER (Adaptado de Spínola e Ávila, 2007).

### 2.2.1 Requisitos

No contexto em questão, Pfleeger (2004) define requisitos como uma característica do sistema ou a descrição de algo que o sistema é capaz de realizar, para atingir os seus objetivos.

Os requisitos de sistema de software são classificados como funcionais e não-funcionais. Os requisitos funcionais devem declarar as funções que o sistema fornecerá, como se comportará com entradas específicas e como vai reagir em determinadas situações. Os requisitos não-funcionais são restrições sobre os serviços ou as funções oferecidos pelo sistema. Destacam-se restrições de tempo, desempenho, segurança, padrões, entre outros (SOMMERVILLE, 2003).

Ainda segundo Sommerville (2003), esses dois tipos de requisitos podem ser descrito em: requisitos de usuário e requisitos de sistema, o primeiro com intuito de ser compreensível ao usuário especificando somente o comportamento externo do

sistema, evitando as características técnicas do projeto de sistema, podendo ser escritos em linguagem natural, formulários e diagramas simples. Já os do segundo tipo, devem ser descritos mais detalhadamente que os requisitos de usuário, com uma especificação completa e consistente de todo o sistema para que os engenheiros de software possam realizar o projeto de sistema, este também pode servir como base para um contrato destinado à implementação do sistema.

Sommerville (2003) complementa que ainda há um nível superior aos citados anteriormente, é o chamado nível dos Requisitos do Negócio. Esses requisitos são as descrições das necessidades que existem no negócio executado pelos usuários, abordando os objetivos, processos, papéis e outras características da organização. Esses requisitos cumprem a finalidade do software a ser desenvolvido, que é atender às necessidades do negócio.

Essa divisão em níveis auxilia o gerenciamento de requisitos e sugere que a identificação de requisitos seja iterativa, iniciando-se pelos Requisitos de Negócio, seguido dos Requisitos de Usuário e por fim os Requisitos de Sistema. Esta hierarquia é demonstrada na Figura 3.



**Figura 3: Hierarquia de Requisitos (Adaptado de Knight, 2004).**

Assim é ressaltada a importância do total entendimento do negócio, por parte dos engenheiros de software, que o SI pretende atender, sendo os Requisitos de Negócio compondo o topo da pirâmide de requisitos.

### 2.2.2 Identificação de Requisitos

A priori, a identificação de requisitos parece ser uma atividade simples, realizando-a perguntando ao cliente, aos usuários e a outros interessados, quais são os objetivos do sistema, o que precisa ser conseguido, como o software se encaixa nas necessidades do negócio e como vai ser usado no dia-a-dia. Mas não é, pelo contrário, é muito difícil (PRESSMAN, 2002).

Christel e Kang (1992) citam alguns problemas para justificar a dificuldade desta atividade:

- Problemas de escopo: escopo mal definido e o cliente e/ou usuários especificam detalhes técnicos desnecessários, confundindo em vez de esclarecer;
- Problemas de entendimento: os clientes e/ou usuários não sabem exatamente o que é necessário, não possuem pleno entendimento do negócio, possuem dificuldade de comunicação, omitem informações que consideram “óbvias”, requisitos conflitantes entre os próprios usuários ou ainda requisitos ambíguos; e
- Problemas de volatilidade: os requisitos mudam ao longo do tempo.

Para otimizar essa atividade, Sommerville e Sawyer(1999) propõem o entendimento em quatro dimensões, são elas:

- Entendimento do domínio da aplicação: ter conhecimento geral da área onde o sistema é aplicado;
- Entendimento do problema: entender os detalhes do problema específico do cliente;
- Entendimento do negócio: para compreender como o SI pode contribuir para o desenvolvimento do negócio; e
- Entendimento das necessidades e restrições dos interessados no sistema.

### 2.2.3 Análise e Negociação

A análise categoriza os requisitos e os organiza de maneira relacionada, explora cada um em relação aos demais, verifica-os quanto à consistência, omissões, ambiguidade e ordena-os com base nas necessidades dos clientes e/ou usuários (PRESSMAN, 2002).

Pressman (2002), ainda cita que é comum que clientes e usuários peçam mais do que é possível ser conseguido, considerando os recursos limitados do negócio, também é relativamente comum que clientes ou usuários proponham requisitos conflitantes. Sendo assim, o engenheiro de software precisa reconciliar esses conflitos por intermédio de um processo de negociação. Usando uma abordagem iterativa, requisitos são eliminados, combinados e/ou modificados de modo que cada parte alcance algum grau de satisfação.

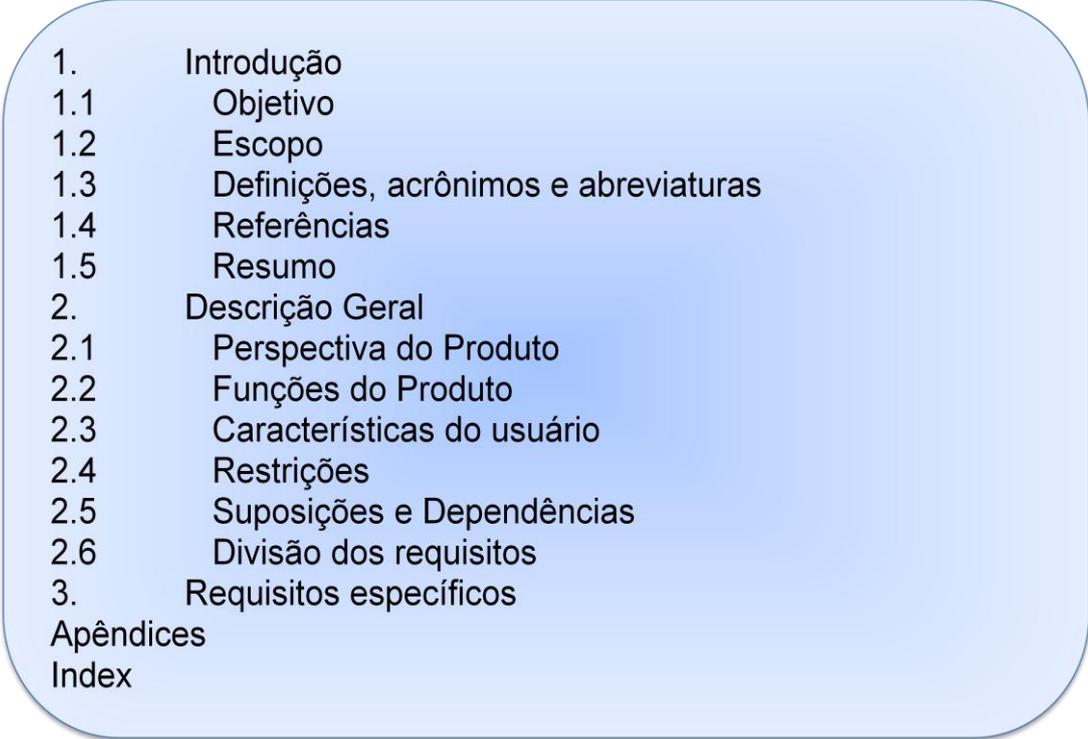
### 2.2.4 Especificação e Documentação

À medida que os requisitos são obtidos, procede-se sua documentação. É nesta atividade que os requisitos são registrados. O documento de requisitos é o artefato criado nesta fase.

Tanto Sommerville (2003) quanto Pfleeger (2004) concordam na separação dos requisitos em dois tipos de documentos. Um para que o cliente possa ler, chamado de documento de definição de requisitos e outro com os requisitos para que o engenheiro de software tome como base para seu trabalho, chamado documento de especificação de requisitos.

O padrão para o documento de especificação de requisitos, em inglês *Software Requirements Specification* (SRS), é o modelo proposto IEEE Std 830-1998, muito conhecido e utilizado.

Na Figura 4 é mostrado um resumo do SRS.



1.	Introdução
1.1	Objetivo
1.2	Escopo
1.3	Definições, acrônimos e abreviaturas
1.4	Referências
1.5	Resumo
2.	Descrição Geral
2.1	Perspectiva do Produto
2.2	Funções do Produto
2.3	Características do usuário
2.4	Restrições
2.5	Suposições e Dependências
2.6	Divisão dos requisitos
3.	Requisitos específicos
	Apêndices
	Index

**Figura 4: Resumo SRS (Adaptado da IEEE Std 830-1998).**

### **2.2.5 Validação**

A validação de requisitos, segundo (PRESSMAN, 2002), examina a especificação para garantir que todos os requisitos do software tenham sido declarados de modo não-ambíguo; que as inconsistências, omissões e erros tenham sido detectados e corrigidos e que os produtos de trabalho estejam de acordo com as normas estabelecidas para o processo, projeto e produto.

A importância dessa atividade é muito grande, pois a ocorrência de erros em um documento de requisitos pode levar a grandes custos relacionados ao retrabalho quando esses erros são descobertos durante o desenvolvimento ou depois que o software estiver em operação (SOMMERVILLE, 2003).

### **2.2.6 Gerenciamento de Requisitos**

Esta atividade se faz necessária, pois conforme citado anteriormente neste trabalho os requisitos são passíveis de alteração dos mais diversos fatores, mas não implicam necessariamente em falhas nas práticas de ER.

De uma forma geral, esta atividade envolve a utilização de técnicas e ferramentas para o gerenciamento de configuração e controle de versão, além de verificar inconsistências nas especificações, conforme evoluem (NUSEIBEH E EASTERBROOK, 2000).

Sommerville (2003) lembra que o gerenciamento de requisitos inclui o planejamento do gerenciamento, em que são especificados os procedimentos e as políticas para tal; e o gerenciamento de mudanças, em que as mudanças são analisadas e seu impacto é avaliado.

### **2.3 UNIFIED MODELING LANGUAGE - UML**

A UML é uma coletânea das melhores práticas de engenharia que tem apresentado sucesso na modelagem de sistemas (VICENTE, 2004).

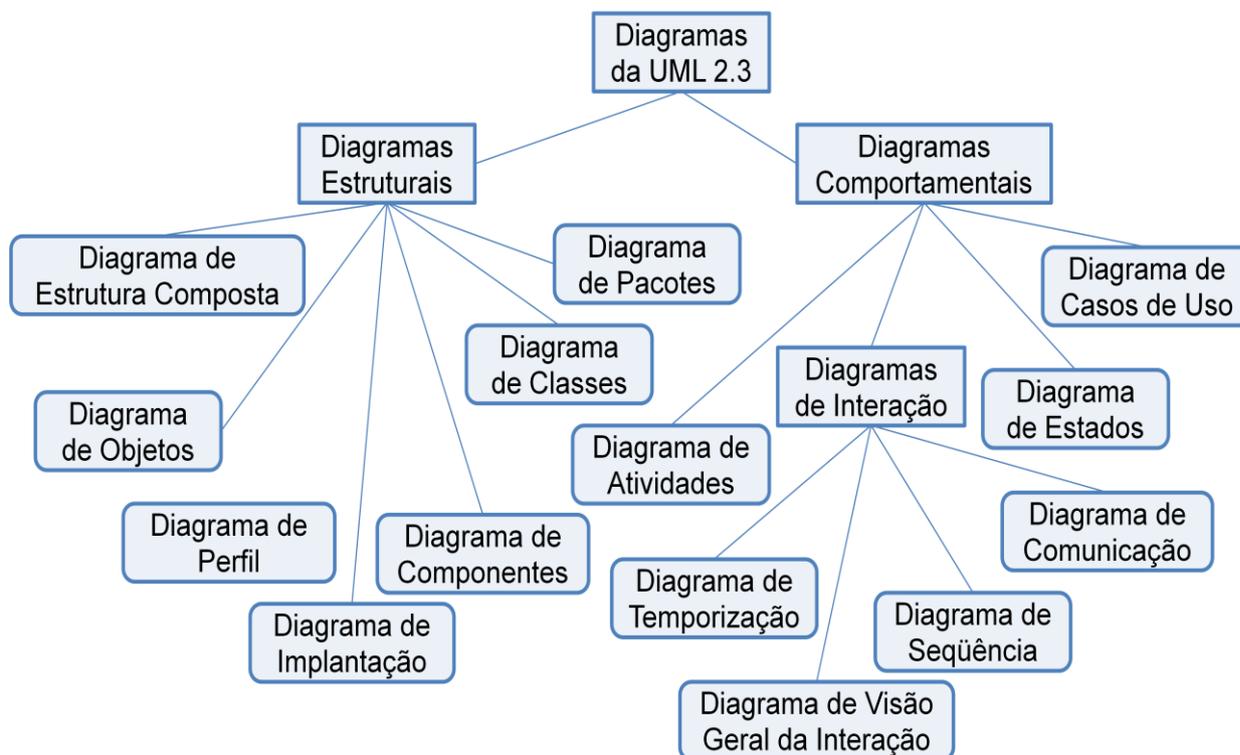
Os criadores da UML, Jacobson et al. (1999), a definem como uma linguagem padrão para especificar, visualizar, construir e documentar artefatos para o desenvolvimento de software.

A UML é independente do processo de desenvolvimento de projeto adotado pela organização. Qualquer que seja o projeto a ser desenvolvido é possível utilizar a UML para registrar as atividades e decisões de análise, projeto e implementação de software, pois não existe um único processo que pode utilizá-la, mas quanto mais iterativo, melhor (FOWLER; SCOTT, 1998 apud REZENDE, 2002).

A *Object Management Group* (OMG) é detentora da patente da UML, que atualmente está na versão 2.3, aprovada pela OMG em 2010.

Desde a versão 2.0 a UML divide os diagramas em duas categorias: estruturais e comportamentais.

O esquema completo dos diagramas é apresentado na Figura 5.



**Figura 5: Diagramas UML 2.3 (Adaptado de OMG UML, 2010).**

No contexto deste trabalho serão utilizados os diagramas em parte da elaboração do documento de requisitos, usufruindo das vantagens que a UML proporciona como linguagem padronizada para representação dos requisitos de um software.

### 2.3.1 Extensão da UML para Modelagem de Negócio

Algumas propostas já foram apresentadas na literatura como extensão da UML para Modelagem de Processos de Negócio. Entre elas pode-se citar a da OMG em 1997, a de Marshall em 1999 e a de Eriksson e Penker em 2000. Esta última é analisada por alguns autores como Azevedo Junior e Campos (2008) e Knight (2004), que apontam algumas desvantagens ou oportunidades de melhoria (CARVALHO, 2009).

Carvalho (2009), após extensa revisão bibliográfica dos métodos propostos pelos autores citados anteriormente, apresenta alguns resultados. Ela descreve que apesar de adotarem a modelagem de processos, os autores não explicitam os

princípios de modelagem aplicados. Também descreve que por utilizarem a UML como modelagem de processos de negócio, algumas limitações são apresentadas. Por utilizarem uma notação que originalmente não foi concebida para tal, e sim para sistemas de software, isso traz insuficiências de representatividade e ainda expõe que tais abordagens não citam a importância de se atentar para os níveis de detalhamento de processos de negócio.

### 3 ENGENHARIA DE PROCESSOS DE NEGÓCIO

A definição de Engenharia de Processos de Negócio (EPN) adotada neste trabalho é a de Cameira e Caulliraux (2000) que a definem como uma técnica utilizada para entender ou mapear como uma parte da organização, uma organização ou um conjunto de organizações funciona, como são realizados os processos, como a informação é transmitida através desses processos, suas interfaces, quais recursos são utilizados, quem realiza as diversas atividades, entre outros, permitindo entender as cadeias de valores<sup>2</sup> existentes.

Segundo Santos (2002), os objetivos da EPN são:

- Uniformização de entendimentos sobre a forma de trabalho;
- Melhoria do fluxo de informações;
- Padronização dos processos;
- Melhoria da gestão organizacional;
- Aumento da conceituação organizacional sobre processos; e
- Ganhos econômicos e de redução de tempo e custo dos processos.

Vernadat (1996) apud Santos (2002) coloca que a criação de modelos de processo é um dos elementos centrais da EPN, pois ela será a base para se alcançar os objetivos acima citados. Um modelo é, acima de tudo, um padrão esperado de comportamento, uma média de todos os casos possíveis.

O campo de estudo da visualização dos modelos de processo é representado por meio da Modelagem de Processos de Negócio, que será tratado com mais detalhes na subseção 3.2.

#### 3.1 PROCESSOS DE NEGÓCIO

Antes de tratar de Modelagem de Processos de Negócio, faz-se necessário definir o conceito de Processos de Negócio.

---

<sup>2</sup> Uma cadeia de valor representa o conjunto de atividades desempenhadas por uma organização desde as relações com os fornecedores e ciclos de produção e de venda até a fase da distribuição final. O conceito foi introduzido por Michael E. Porter (professor da *Harvard Business School*) em 1985.

Entre as definições existentes na literatura, Davenport (1994) descreve o Processo como:

... um conjunto de atividades estruturadas e medidas, destinadas a resultar num produto especificado para um determinado cliente ou mercado. É, portanto, uma ordenação específica das atividades de trabalho no tempo e no espaço, com um começo, um fim e *inputs* e *outputs* claramente identificados.

Uma definição bastante completa para Processos de Negócio é a de Santos (2007):

Uma estruturação-coordenação-disposição lógico-temporal de ações e recursos com o objetivo de gerar um ou mais resultados para a organização. Os processos podem estar em diferentes níveis de abstração ou detalhamento, relacionados às atividades gerenciais, finalísticas ou de apoio. Se forem finalísticos, os resultados gerados são produto(s)/serviço(s) para os clientes da organização, se forem gerenciais promovem o funcionamento da organização e seus processos, e se forem de suporte prestam apoio aos demais processos da organização. Também podem possuir um responsável por seu desempenho global e responsáveis locais direcionados aos andamentos de suas partes-constituintes e, comumente, são transversais a forma através da qual a organização se estruturou (por função, por produto, por eixo geográfico, etc.). [...] Os processos são objetos de controle e melhoria, mas também permitem que a organização os utilize como base de registro do aprendizado sobre como atua, atuou ou atuará em seu ambiente ou contexto organizacional. Os processos são a organização em movimento, são, também, uma estruturação para ação: para a geração e entrega de valor.

A informação é um recurso que está envolvido direta ou indiretamente com o processo de negócio, podendo ser materializada através de SI que informatizam ou apóiam as atividades dos processos. Devido a essa forte relação entre processos de negócio e SI, é desejável ter uma arquitetura de software adequada aos processos organizacionais, bem como ter novos processos que possam ser transformados e conseqüentemente melhorados, com apoio de novas tecnologias (ERIKSSON E PENKER, 2000 apud CARVALHO, 2009). Assim, no contexto deste trabalho, a Modelagem de Processos de Negócio é uma alternativa viável para servir como guia desta arquitetura.

### **3.2 MODELAGEM DE PROCESSOS DE NEGÓCIO**

A Modelagem de Processos de Negócio tem sido usada para usufruir da manipulação da informação contida nos diversos setores de uma organização.

O objetivo da modelagem é criar um modelo de processos por meio da construção de diagramas operacionais sobre o comportamento organizacional. A modelagem serve para validar o projeto, verificando se os requisitos globais estabelecidos são atendidos, testando suas reações sob diversas condições (SOUZA et al., 2009).

Torna-se assim, cada vez mais comum, as organizações modelarem seus processos de negócio obtendo um maior conhecimento interno de sua estrutura e necessidade, se mantendo competitivas no mercado.

Para a modelagem é necessário a adoção de uma técnica de modelagem. No presente trabalho a técnica selecionada é o BPMN, pois apresenta as seguintes vantagens (VALLE; OLIVEIRA, 2009):

- padronização e gestão feitas pelo OMG, um grupo de empresas-membros, consolidadas e com boa reputação no mercado de padrões abertos;
- oferece um padrão de notação com suporte em várias ferramentas de modelagem;
- permite evoluir para o padrão XPDL 2.0, que é explicitamente uma linguagem de descrição de *workflow*;
- visando reduzir a lacuna entre o desenho de processo de negócio e a sua implementação, o BPMN permite a conversão de seus DPN para a linguagem de execução de processo de negócio BPEL – *Business Process Execution Language*;
- o BPMN incorpora facilidades de técnicas consagradas de padrões de modelagem, como UML/AD e o IDEF.

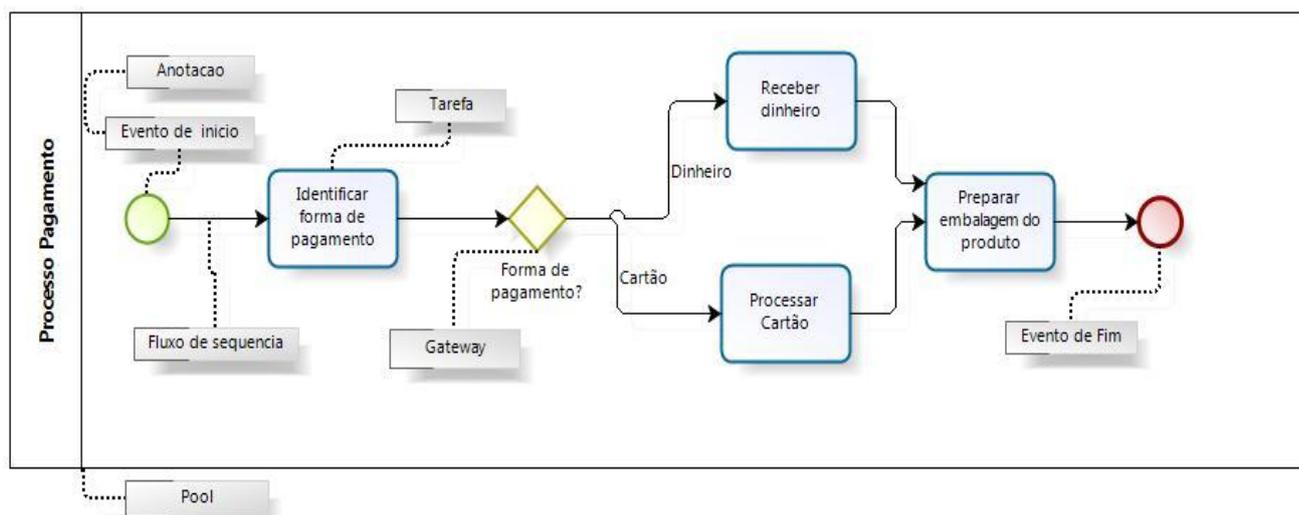
### **3.2.1 Business Process Modeling Notation - BPMN**

BPMN é um padrão de notação para modelagem de processos de negócio desenvolvido pelo *The Business Process Management Initiative* (BPMI) juntamente com a OMG. O objetivo principal do BPMN é prover uma notação compreensível por todos os envolvidos no negócio que criam os processos até os desenvolvedores

técnicos responsáveis pela implementação da tecnologia que executará esses processos (CARVALHO, 2009).

O diagrama gerado pela notação é o *Business Process Diagram* (BPD), nesse diagrama são dispostos os diversos elementos que formam o BPMN. Assim o BPD permite a representação gráfica dos elementos dos processos, facilitando e simplificando o desenvolvimento do modelo do processo de negócio através de elementos distintos e com significado próprio (CARVALHO, 2009).

Na Figura 6 é demonstrado um exemplo de diagrama de processo modelado, utilizando os principais elementos de modelagem do BPMN, por meio da ferramenta *BizAgi Process Modeler*<sup>3</sup>.



**Figura 6: Exemplo de BPD (Adaptado de Valle; Oliveira, 2009).**

<sup>3</sup> <http://www.bizagi.com/>

## 4 MODELANDO UM PROCESSO DE NEGÓCIO

Para o desenvolvimento da proposta do presente trabalho, é necessária a realização da modelagem de um processo de negócio. O processo adotado foi o de registro de novos projetos de pesquisa de uma universidade. Foi utilizado um fluxograma descritivo com os trâmites do processo em questão, propiciando o entendimento do mesmo e embasamento para a modelagem. Este documento foi elaborado e disponibilizado pela Pró-Reitoria de Pesquisa e Pós-Graduação (PROPG) da universidade e está resumido no Quadro 1.

**Quadro 1: Fluxograma de Registro de Projeto de Pesquisa**

Nº	Descrição	Prazo
1	O Coordenador elabora o projeto e o encaminha ao Conselho de Centro. Em havendo a necessidade de avaliação de ética, o coordenador anexa os formulários do respectivo comitê preenchidos.	-
2	O Conselho de Centro encaminha o projeto à Comissão de Pesquisa do Campus, ao Comitê de Ética e a consultor ad hoc, se necessário, para a emissão de pareceres. Se houver solicitação de recursos financeiros à administração do Campus, encaminha-se o projeto à Direção do Campus.	5 dias
3	As Comissões de Pesquisa, o(s) Comitê(s) de Ética, o consultor ad hoc e a Direção do Campus emitem pareceres e os encaminham ao Conselho de Centro.	20 ou 30 dias
4	Com base nos pareceres, o Conselho de Centro aprecia o mérito, emite parecer próprio e encaminha à PROPG.	5 dias
5	A PROPG revisa a documentação e encaminha o projeto à Câmara de Pesquisa e Pós-Graduação.	5 dias
6	A Câmara de Pesquisa e Pós-Graduação avalia o projeto, delibera e o encaminha à PROPG.	5 dias após a próxima reunião da Câmara
7	Uma vez aprovado, a PROPG registra o projeto e dá ciência aos interessados, inclusive o Diretor de Campus e o Diretor do Centro de Estudos respectivos, como forma de manterem registro de seus projetos de pesquisa.	5 dias
8	Não sendo aprovado em qualquer instância, o projeto é devolvido ao coordenador para diligência e/ou recurso ao CEPE.	-

#### 4.1 APLICAÇÃO DA MODELAGEM

Como já citado anteriormente, o padrão de modelagem adotado para o trabalho é o BPMN. Assim sendo, definiu-se a *pool* “Registro Novo Projeto de Pesquisa” para ser o processo a ser modelado. As entidades identificadas no fluxograma foram:

- Coordenador, que pode ser expresso por meio do termo Docente;
- Conselho de Centro;
- Comissão de Pesquisa;
- Comitê de Ética;
- Consultor *ad hoc*;
- Direção do Campus;
- PROPG; e
- Câmara de Pesquisa e Pós-Graduação.

Essas entidades representam as *lanes*, que são subpartições que atuam no processo. As entidades Comissão de Pesquisa, Comitê de Ética e Consultor *ad hoc* foram agrupadas em uma *lane* chamada “Comissões e consultor”.

Em seguida, as tarefas do processo foram identificadas conforme o fluxograma descritivo de registro de projetos de pesquisa e foram representadas graficamente com o elemento *task* da notação. Cada *task* relacionada com sua *lane*, em outras palavras, cada tarefa do processo está localizado na subpartição correspondente por quem a executa. Para uma modelagem mais rica, as tarefas foram especializadas conforme a notação permite.

Também foram utilizados os elementos de fluxo *gateways* e *events*. Os *gateways* para representar qual caminho o fluxo seguirá e os *events* para marcar o início do processo, eventualmente algum evento intermediário e os possíveis fins que o processo pode ter. Esses elementos também foram especializados conforme a notação BPMN para maior riqueza de detalhes e melhor entendimento do processo.

O resultado dessa modelagem pode ser visualizado no BPD da Figura 7.

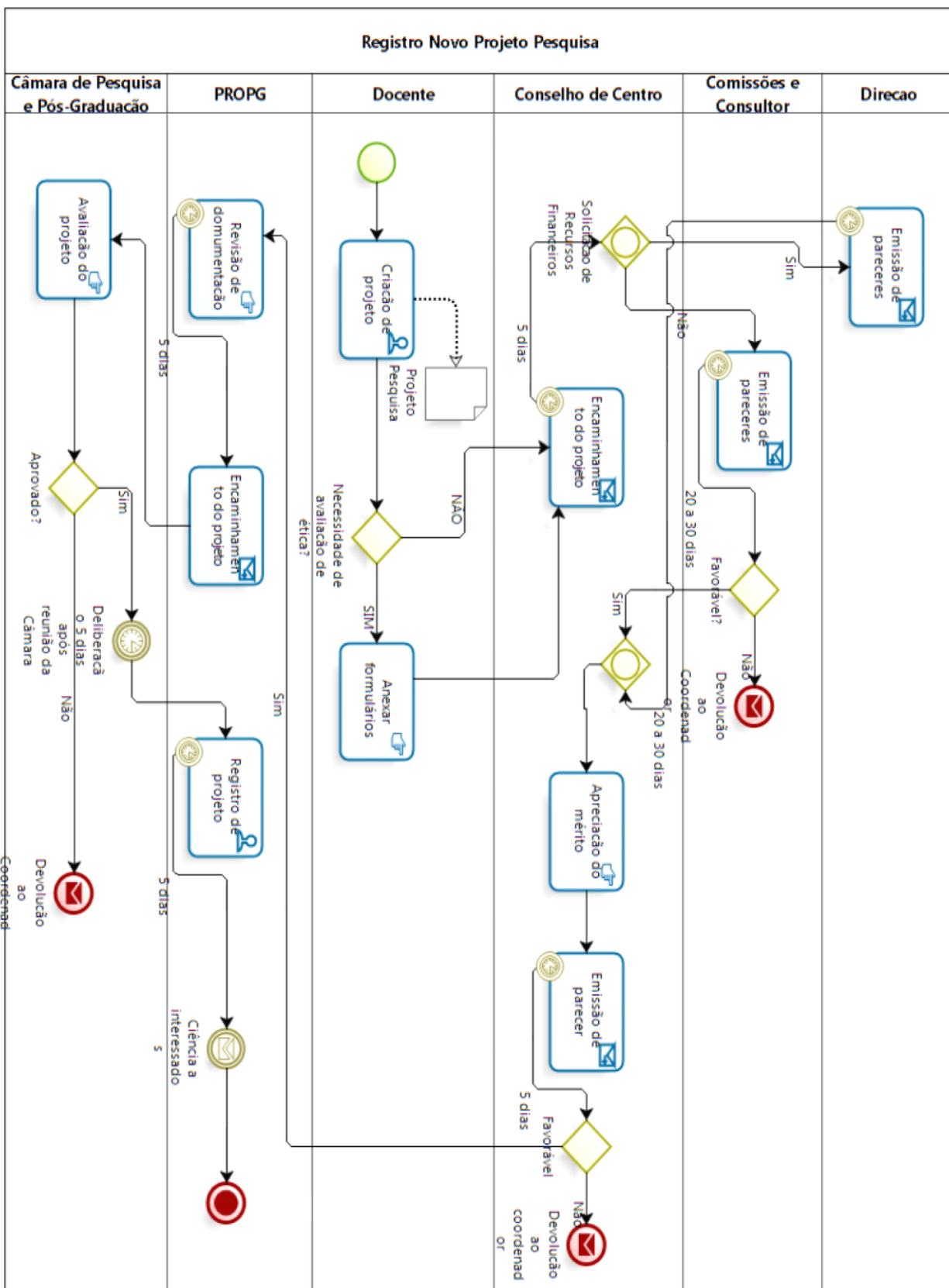


Figura 7: BPD Registro Novo Projeto Pesquisa.

## **4.2 CONSIDERAÇÕES FINAIS**

O processo de Registro de Projeto de Pesquisa descrito no fluxograma apresentado na subseção 4.1 foi modelado com sucesso a partir da notação BPMN. Com o modelo visual, o processo pode ser entendido de maneira completa, através da visualização do fluxo, das tarefas, responsabilidades e eventos. Como consequência, a contribuição para parte de uma ER eficaz pode ser feita de maneira direta, essa contribuição está descrita na próxima seção com uma análise da transição do BPD para diagramas UML.

## 5 TRANSIÇÃO MODELAGEM BPMN PARA UML

Eriksson e Penker (2000) apud Vicente (2004) colocam que a modelagem de software utiliza a modelagem de negócios para os seguintes aspectos:

- Identificação de informações de sistemas que possibilitem um suporte eficaz para o negócio;
- Identificação de requisitos funcionais desenvolvidos de maneira correta que o sistema fornecerá aos processos de negócios;
- Identificação de requisitos não funcionais que não estão descritos nos casos de uso;
- Identificação de possíveis classes de sistemas com base na modelagem de negócio; e
- Identificação de componentes de negócio que possam ser componentes de software.

O presente trabalho aborda a transição da modelagem BPMN com foco em dois diagramas UML, sendo um diagrama de cada categoria, comportamental e estrutural da UML 2.0.

Dentre a categoria comportamental, foi selecionado o diagrama de casos de uso, pois Jacobson et al. (1999) cita que esses diagramas possuem papel central para modelagem do contexto do sistema.

Para a categoria estrutural, foi selecionado o diagrama de classes, sendo que o mesmo autor cita que são os diagramas mais utilizados para modelagem de sistemas Orientados a Objetos (OO) e por serem base para outros diagramas estruturais.

## 5.1 ANÁLISE DE TRANSIÇÃO BPD X CASOS DE USO

Após a modelagem do processo de negócio iniciou-se o processo de análise dos componentes do diagrama BPD, sendo identificados e descritos os casos de uso por meio da modelagem UML.

Vicente (2004), em seu trabalho faz uma derivação de um modelo EPN realizado através da metodologia ARIS<sup>4</sup> para uma modelagem de casos de uso utilizando o elemento Executor e o elemento Atividade representando o Ator e o Caso de Uso do diagrama de Casos de Uso, respectivamente.

No atual trabalho, como já explicitado anteriormente, é utilizado a modelagem BPMN para tal. Assim, a primeira transição realizada foi para o nome do diagrama de Casos de Uso, no qual o nome do processo modelado descrito na primeira coluna do lado esquerdo do diagrama BPD representa diretamente o nome do diagrama a ser modelado.

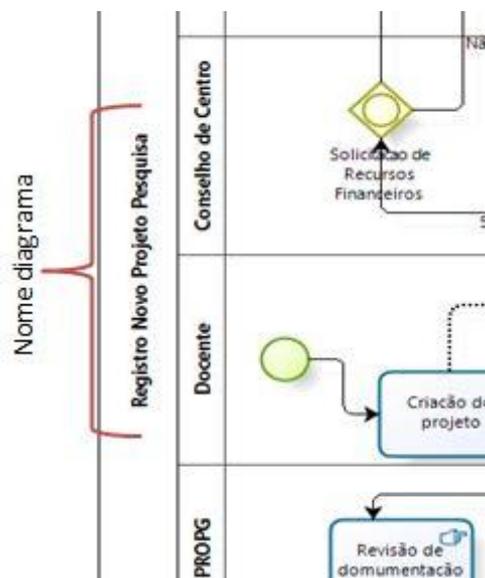
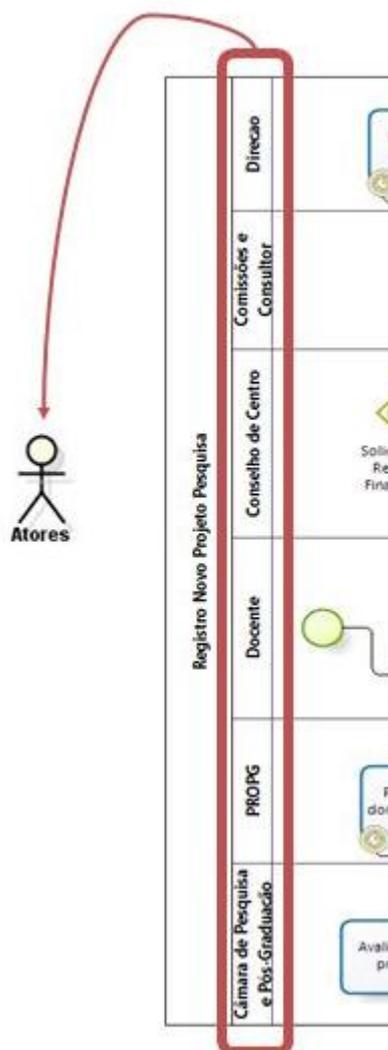


Figura 8: Nome diagrama Casos de Uso.

Em seguida foram analisadas as *lanes* do processo modelado, sendo que estas são entidades que atuam no processo, sendo assim, representam os possíveis atores do diagrama de casos de uso.

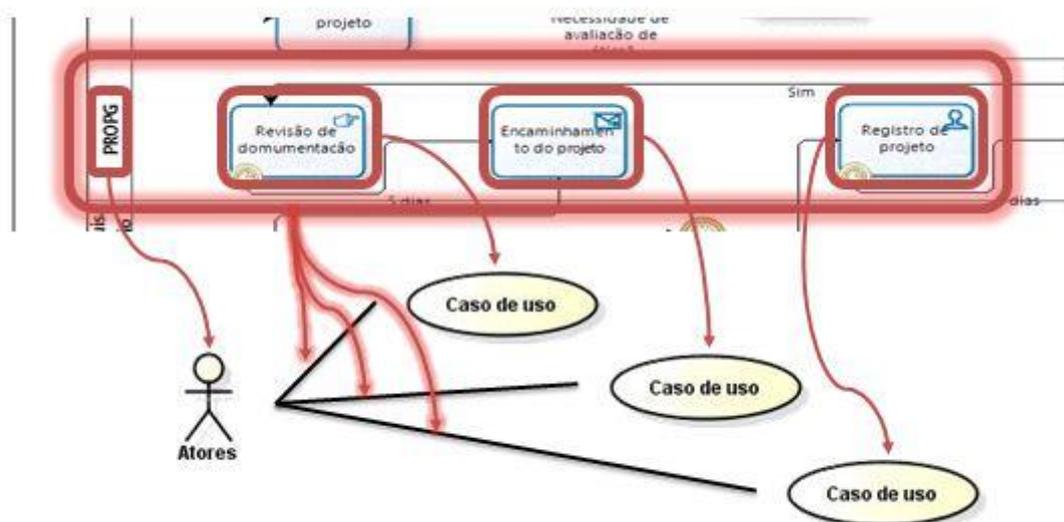
<sup>4</sup> A metodologia ARIS de Modelagem de Processos de Negócio está fundamentada na utilização de uma grande variedade de modelos e objetos através dos quais os processos de negócio de uma dada organização podem ser representados e analisados, com a utilização da ferramenta ARIS Toolset (SANTOS, 2002).



**Figura 9: Atores Casos de Uso.**

Um importante elemento identificado para essa transição é a *task*, que são as tarefas do processo de negócio. Essas tarefas podem representar o caso de uso em si do diagrama a ser modelado. Na Figura 10 essas *tasks* estão marcadas na cor vermelha e demonstram que representam os diversos casos de uso identificados.

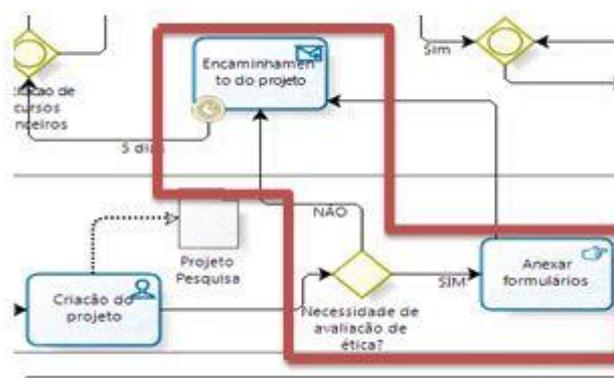




**Figura 11: Associações do diagrama.**

Analisando os *gateways* do modelo BPD, foram identificados que *gateways* que precederam duas *tasks* poderiam ser possíveis relacionamentos *extends* no diagrama a ser modelado. Isso se dá ao fato de que essas *tasks* representam casos de usos e pela existência de um *gateway* significa que essa tarefa pode ou não ser executada dentro do processo em questão, caracterizando assim essa associação com relacionamentos *extends*.

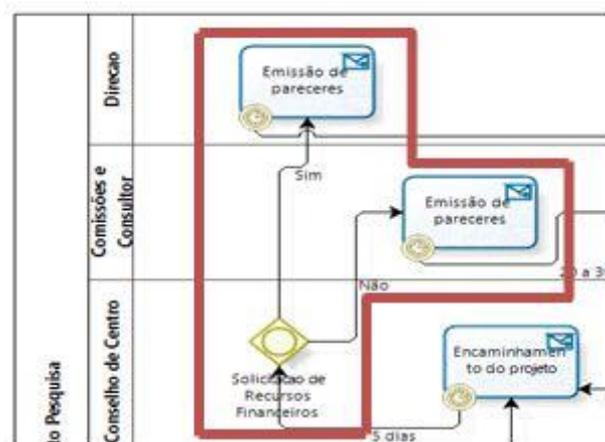
Na Figura 12 e a Figura 13 é demonstrado este caso identificado em dois *gateways* presentes no BPD modelado.



**Figura 12: Gateway exclusivo caracterizando relacionamento *extend*.**

Na Figura 12 pode-se visualizar que chegando ao *gateway* “Necessidade de avaliação de ética?” o processo seguirá apenas um caminho: no qual não serão

anexados os respectivos formulários, caso a resposta seja “não”; ou será necessário anexar caso a resposta seja “sim”. Essa decisão de executar uma ou outra *task*, que como visto anteriormente representa casos de usos, caracteriza um caso de uso *extend*, pois são relacionamentos que demonstram casos de uso que podem ou não serem executados.



**Figura 13: Gateway inclusivo caracterizando relacionamento *extend*.**

Já no caso do *gateway* “Solicitação de Recursos Financeiros?”, o processo pode seguir os dois caminhos e não necessariamente apenas um, pois se trata de um gateway inclusivo. Mas por sua característica de poder seguir apenas um dos fluxos, este é caracterizado como um relacionamento *extend* no diagrama de casos de uso pela não obrigatoriedade de executar umas das *tasks* envolvidas.

Outros elementos identificados no BPD que também podem gerar casos de uso do sistema são os eventos de mensagens. Estes fazem parte do processo, dependendo de qual o fluxo seguido esses eventos são acionados. Sendo assim, são importantes elementos que possuem relevância para a transição do diagrama BPD para diagrama de Casos de uso.

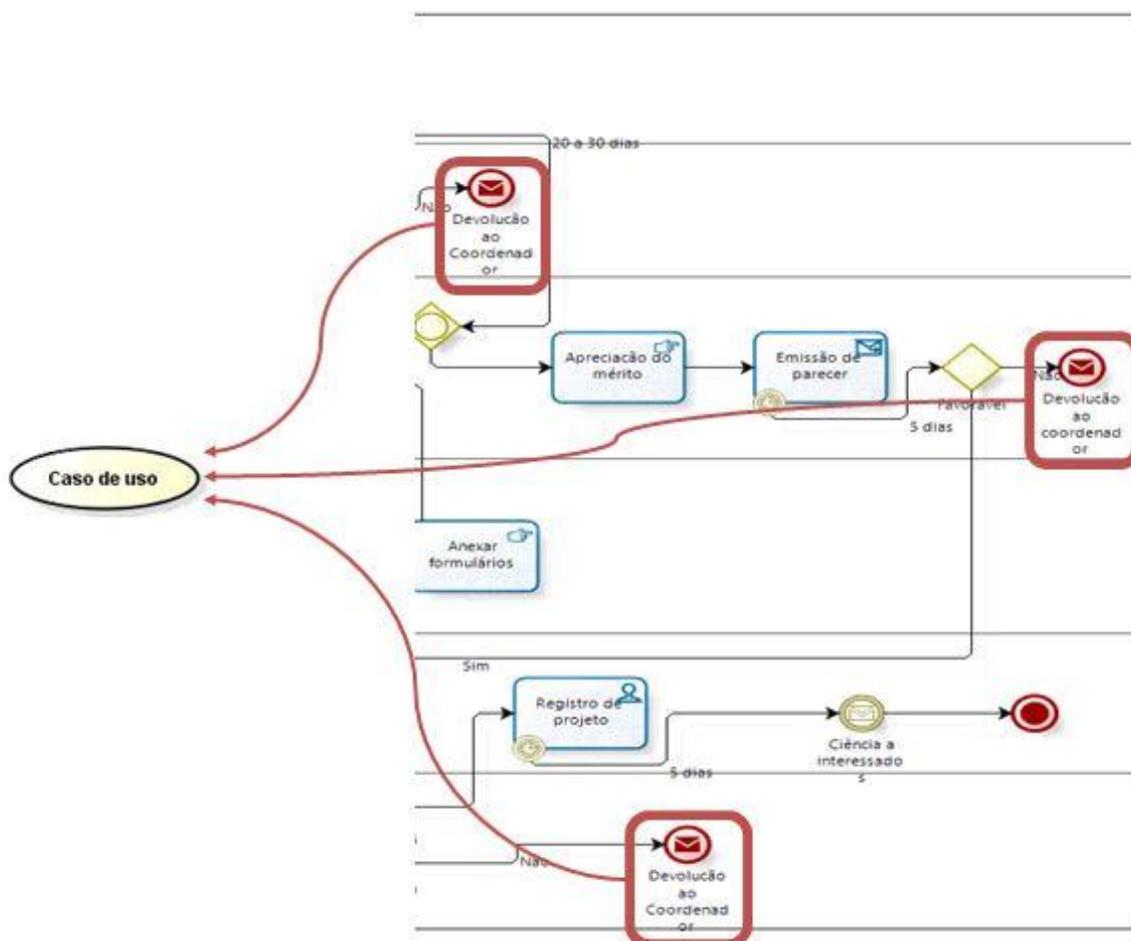


Figura 14: Eventos de mensagem e caso de uso.

### 5.1.1 Geração do diagrama de Casos de Uso

Após todo o estudo realizado sobre o BPD e análise de como seus elementos contribuiriam para uma modelagem UML de casos de uso, foi possível a identificação dos atores; casos de usos; associações; e relacionamentos. Com esses conceitos de transições definidos, o seguinte diagrama UML de Casos de uso foi modelado.

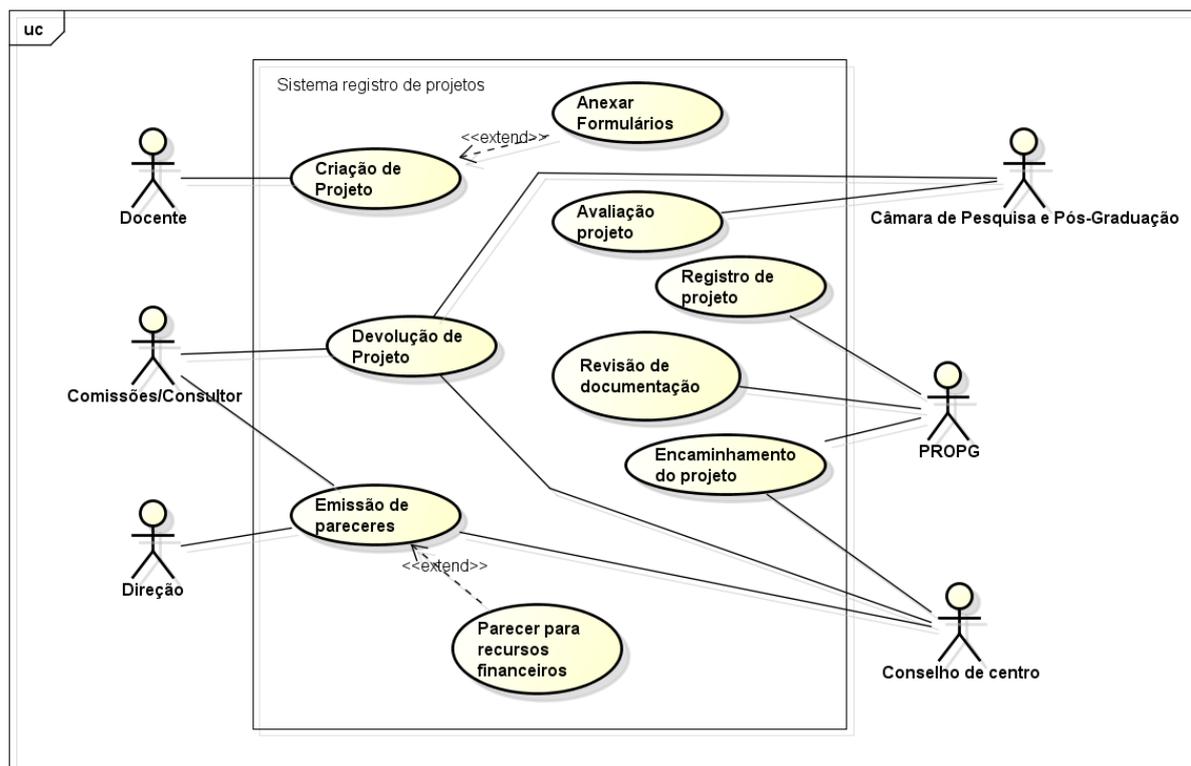
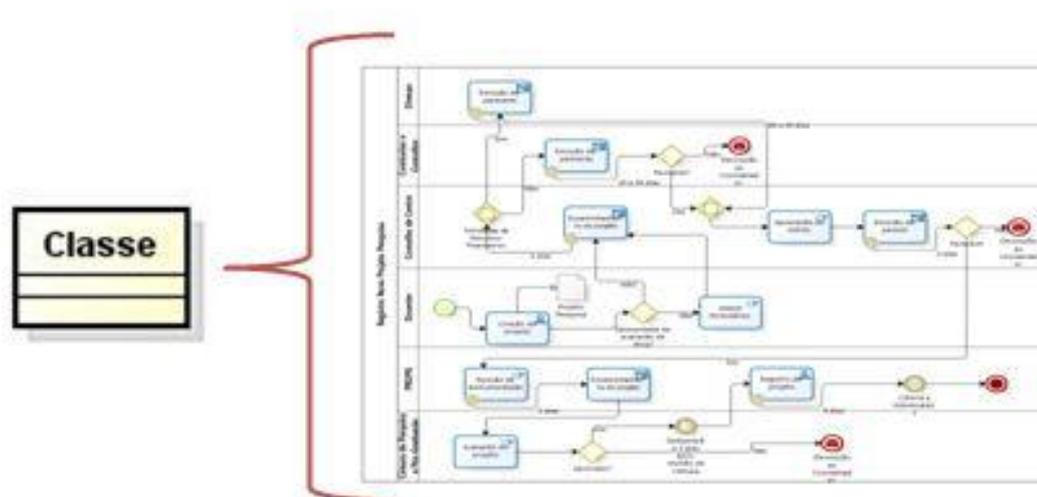


Figura 15: Diagrama de casos de uso gerado a partir do BPD.

## 5.2 ANÁLISE DE TRANSIÇÃO BPD X DIAGRAMA DE CLASSES

Como já citado anteriormente, outro diagrama UML tratado nesse trabalho é o diagrama de classes, que procura demonstrar uma estrutura OO do contexto geral do sistema a ser desenvolvido, independentemente da linguagem de programação a ser utilizada.

A primeira identificação de uma classe apresentada pelo BPD Registro Novo Projeto Pesquisa foi o processo como um todo, pois pensando em uma expansão para o sistema, ele poderia possuir outros processos com outras operações e atributos. Assim a primeira classe definida foi “Processo”.



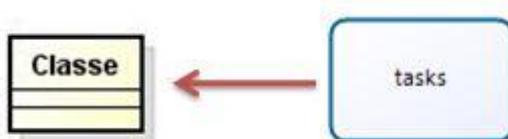
**Figura 16: Classe Processo.**

Para que exista um processo, é necessário que haja um objeto que o gere, esse objeto é o Projeto de Pesquisa, caracterizando outra classe do diagrama a ser modelado e também a existência de um relacionamento entre essas duas classes; Processo e Projeto Pesquisa.



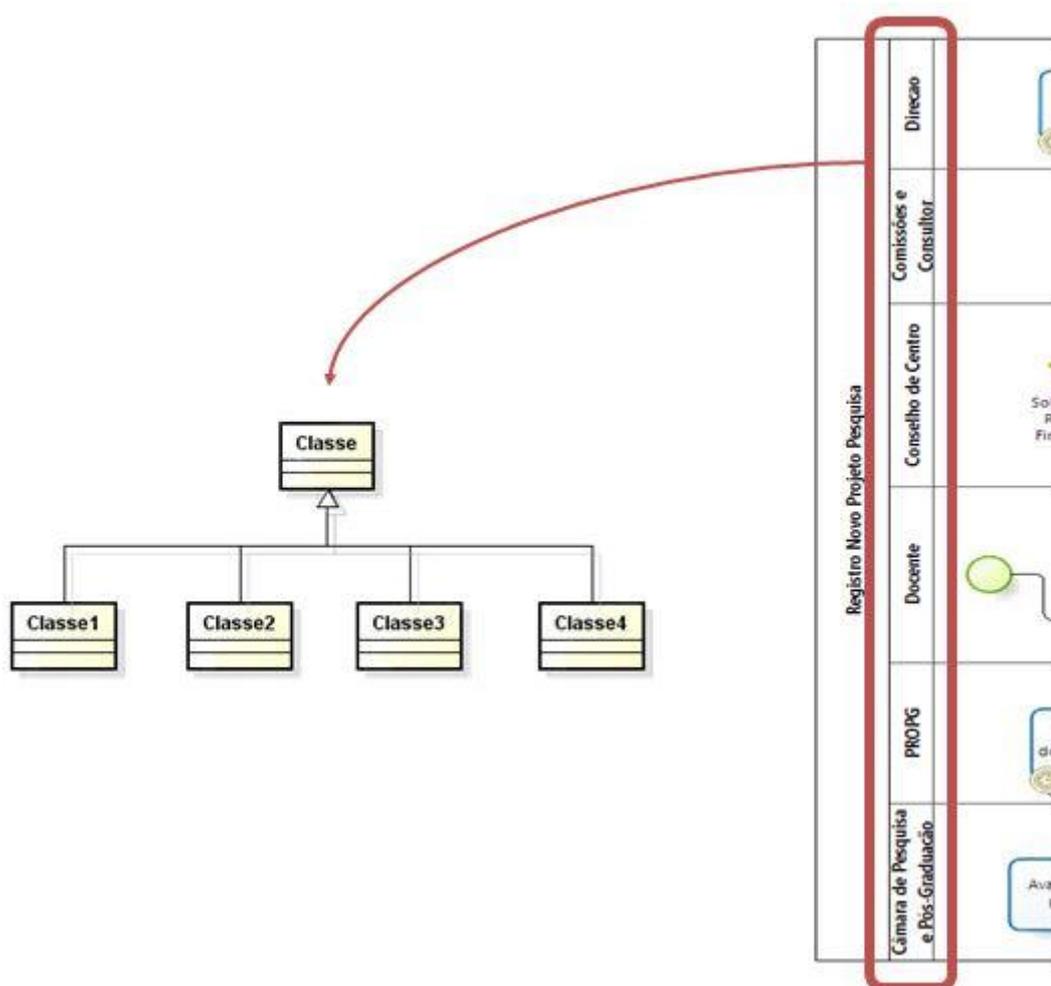
**Figura 17: Classe Projeto Pesquisa.**

As *tasks* são elementos indispensáveis para modelagem de processos com a notação BPMN, com isso é importante a representatividade desses elementos no projeto de desenvolvimento de software. Portanto elas também representam uma classe, a “Tarefa”, no diagrama de classes. Pelo fato das *tasks* comporem um processo, também é identificado um relacionamento entre Processo e Tarefa, que posteriormente poderá ser visualizada no diagrama de classes.



**Figura 18: Classe Tarefa.**

Como já citado anteriormente, as *lanes* representam entidades que atuam no processo, portanto caracteriza-se a existência da classe pai “Entidade” e suas classes filhas, que é respectivamente cada *lane* do processo modelado. Essas entidades são responsáveis por executar as *tasks* modeladas no BPD, caracterizando assim uma relação entre as classes dessas entidades e a classe Tarefa, que representa as *tasks*.

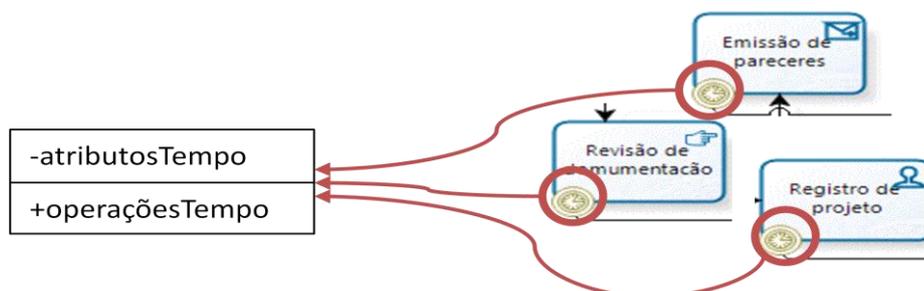


**Figura 19: Classes derivadas das *lanes*.**

Conforme o tema do Projeto de Pesquisa criado, pode ser que haja necessidade de uma avaliação de ética, para isso devem ser anexados os respectivos formulários. Essa tarefa está modelada no BPD e nomeada como Anexar Formulários. Esses formulários geram um objeto diferente do Projeto de Pesquisa, caracterizando assim uma nova classe e um relacionamento entre ambos.

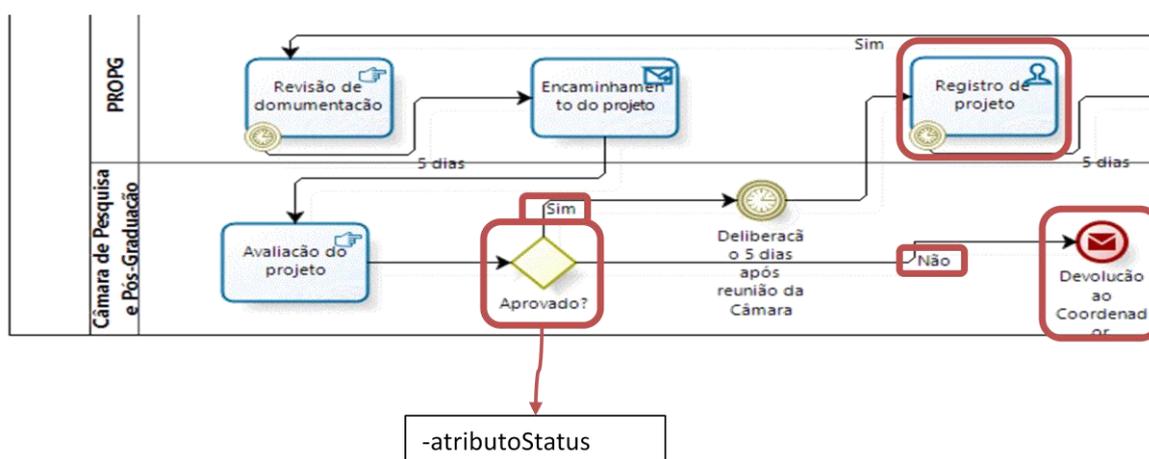


Outros eventos existentes no BPD também podem ser significativos na modelagem de classes, como por exemplo, os eventos de *timers* anexados a algumas *tasks*. Esses eventos explicitam a necessidade de atributos e operações de tempo na correspondente classe.



**Figura 22: Atributos e operações de tempo.**

Mais um elemento do BPD que auxilia na identificação de atributos das classes geradas são os *gateways*, por serem os definidores do caminho do processo estes podem ser considerados como atributos de status nas classes. Por exemplo, o processo ao atingir o *gateway* “Aprovado?” que está localizado na *lane* “Câmara de Pesquisa e Pós-Graduação” pode obter a resposta “Sim” ou “Não”, com isso alterando o status do Projeto de Pesquisa em questão para “Aprovado” ou “Devolvido” respectivamente. Este exemplo pode ser visualizado na Figura 23.



**Figura 23: Atributos de status.**

### 5.2.1 Geração do diagrama de Classes

Com a análise realizada sobre o BPD foram derivadas algumas classes para a construção do diagrama de classes UML, também definindo os relacionamentos

entre as classes geradas. Organizando as idéias apresentadas de transição de um modelo para outro, foi gerado o diagrama apresentado na Figura 24.

O diagrama de classes do projeto buscou demonstrar os principais conceitos do sistema sendo classificado na perspectiva conceitual, independente da linguagem de programação a ser utilizada para o desenvolvimento do software.

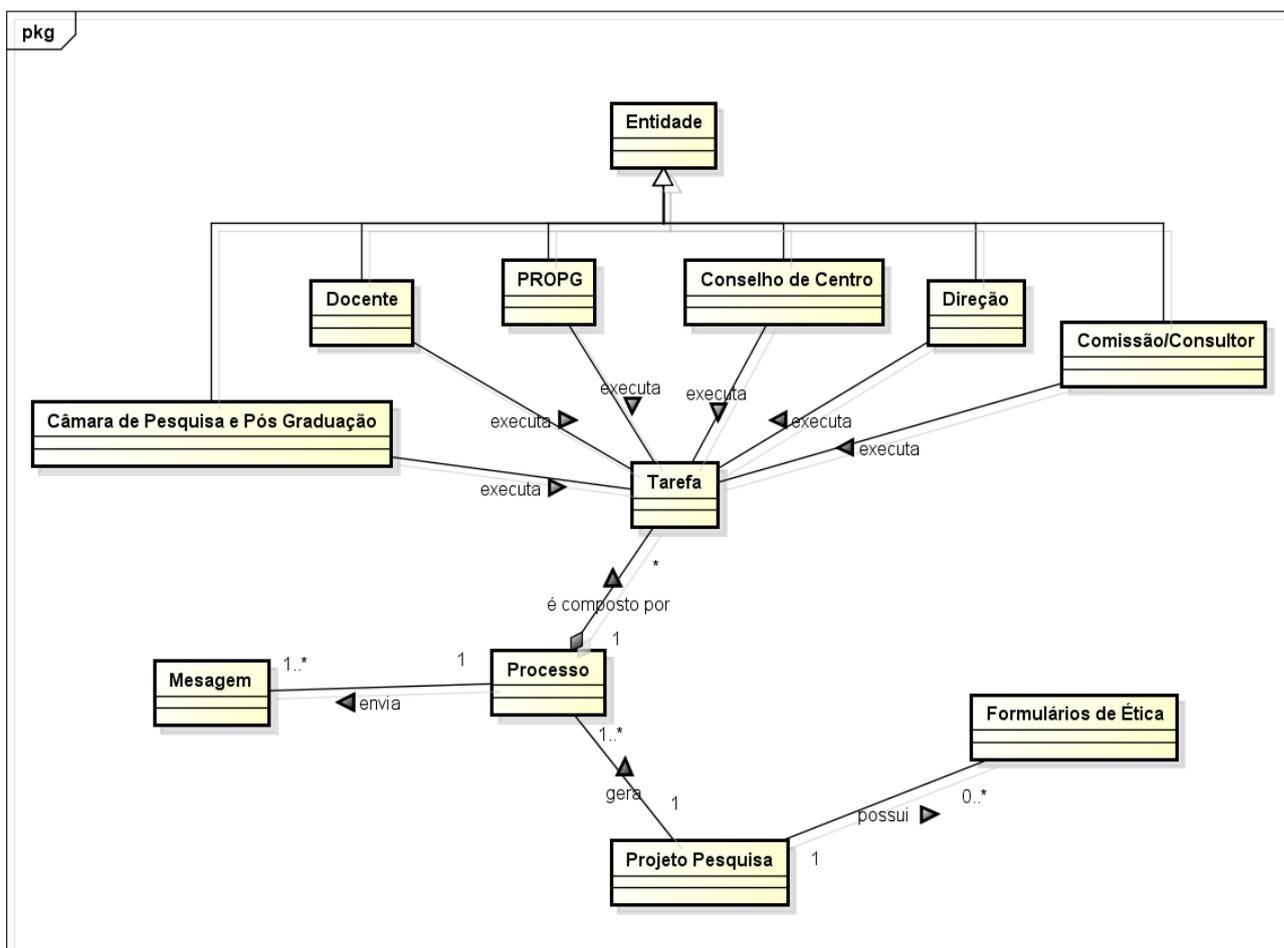


Figura 24: Diagrama de classes gerado a partir do BPD.

### 5.3 CONSIDERAÇÕES FINAIS

Esta seção demonstrou o método proposto para modelagem de sistemas da ER orientada à modelagem de processos, utilizando o processo de registro de novos projetos de pesquisa de uma Universidade procurando mostrar de forma clara e concisa de que maneira são feitas as transições de um modelo para outro, explicando cada artefato gerado individualmente.

Com o diagrama de casos de uso foi possível apresentar todos os requisitos funcionais para o desenvolvimento do software. Já o diagrama de classes foi modelado de maneira parcial, pois não foram detalhados os atributos e operações de cada classe. Alguns destes foram apresentados na abordagem (atributos e operações de tempo e atributos de status), porém são necessárias outras técnicas e/ou métodos para especificação de requisitos para maior riqueza de detalhes neste contexto.

## 6 CONSIDERAÇÕES FINAIS

O trabalho realizado foi motivado pelo problema já constatado na literatura e também pela prática de empresas de desenvolvimento de softwares, a deficiência em definir requisitos para construção de SIs não satisfazendo os objetivos de negócio, conseqüentemente frustrando os clientes.

A utilização do modelo BPMN para modelagem de processos de negócio foi eficaz no que diz respeito a proporcionar o entendimento do processo adotado para o desenvolvimento do trabalho. Utilizando-se desde elementos básicos até especializados da notação foi possível gerar um modelo visual de todo o processo de maneira simplificada, o que proporciona o entendimento para as diversas pessoas envolvidas, como por exemplo, clientes, analista de negócio e analista de sistemas.

Com o BPD realizado e o processo claramente entendido iniciou-se o desenvolvimento da parte central a fim de cumprir o objetivo estabelecido por esta pesquisa. Os modelos UML de casos de uso e classes foram gerados diretamente do BDP apresentado, concluindo assim que a utilização de modelos de processo de negócio em apoio à ER mostrou-se promissora. Os resultados mostram que pode haver significativa contribuição para o processo de software, auxiliando na construção de sistemas de qualidade, que realmente atendam às necessidades do cliente.

É importante lembrar que o método proposto não deve ser usado unicamente para a definição de requisitos e sim como um refinamento complementando outros métodos já conhecidos e mais tradicionais, tais como:

- Entrevistas realizadas por analista de sistemas com os *stakeholders* do negócio abordado, individuais ou em grupos;
- Questionários que deverão ser respondidos por pessoas envolvidas do negócio, tais como: proponentes, beneficiários, usuários, especialistas;

- *Brainstorming* envolvendo especialistas do domínio do problema (analista de negócio) e do domínio da solução (analista de sistemas);
- Prototipação de partes do sistema sendo proposto; e
- Outras técnicas ou métodos que sejam disponíveis ou convenientes.

A utilização de combinação de métodos aumenta o nível da qualidade do software.

## **6.1 POSSIBILIDADES DE TRABALHOS FUTUROS**

Algumas possibilidades de desdobramentos para trabalhos futuros serão apresentadas nesta subseção. A primeira delas é de verificar se existe a possibilidade de transição do BPD para outros modelos UML não abordados neste trabalho.

Outra sugestão é utilizar os modelos apresentados neste trabalho e criar um protótipo para maior consistência dos resultados apresentados.

Sendo o BPMN uma notação expansível, com o qual se pode criar novos elementos, outra possibilidade é de criar elementos pertencentes a UML para haver uma proximidade maior entre as duas notações, criando assim uma maior integração entre elas.

## REFERÊNCIAS

- AZEVEDO JUNIOR, D.P.; CAMPOS, R. **Definição de Requisitos de Software Baseada numa Arquitetura de Modelagem de Negócios**. Revista Produção, v.18, n.1, pp. 26-46, 2008.
- BUBENKO, J.A. **Extending the Scope of Information Modeling**. 4th International Workshop on the Deductive Approach to Information Systems and Databases, pp. 73-98. Lloret de Mar, Espanha, 1993.
- CAMEIRA, R.; CAULLIRAUX, H. **Engenharia de Processos de Negócios: considerações metodológicas com vistas à análise e integração de processos**. 3º Simpósio de Administração da Produção, Logística e Operações Internacionais. São Paulo: FGV, 2000.
- CAMPOS, R.; SANTOS, L.R.S. **Modelagem de Processos e Definição de Requisitos de Sistema de Informação para a Previsão de Demanda**. XXV Encontro Nacional da ANPAD. Campinas: ANPAD, 2001.
- CARVALHO, E. A. **Engenharia de Processos de Negócios e a Engenharia de Requisitos: análise e comparações de abordagens e métodos de Elicitação de Requisitos de sistema orientada por Processos de Negócio**. Dissertação (Mestrado em Engenharia de Produção). Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2009.
- CHRISTEL, M. G.; KANG, K. C. **Issues in Requirements Elicitation**. Software Engineering Institute, CMU/SEI-92-TR-12, 1992.
- DAVENPORT, T. **Reengenharia de Processos: como inovar na empresa através da tecnologia da informação**. Rio de Janeiro: Campus, 1994.
- DIAS, F.; MORGADO, G.; OSCAR, P.; SILVIERA, D.; JUAREZ, A.; LIMA, P. & SCHMITZ, E. **Uma Abordagem para a Transformação Automática do Modelo de Negócio em Modelo de Requisitos**. Anais do WER06 - Workshop em Engenharia de Requisitos, Rio de Janeiro, RJ, Brasil, pp 51-60, 2006.
- ERIKSSON, H. E.; PENKER, M. **Business Modeling with UML: business pattern sat work**. New York: Wiley Publishers, 2000.
- IEEE Std 830-1998. **IEEE Recommended Practice for Software Requirements Specifications**. IEEE Computer Society, 1998.
- JACOBSON, I.; BOOCH, G.; RUMBAUGH, J. **The Unified Modeling Language Reference Manual**. 1 ed. Boston (MA): Addison-Wesley Longman, 1999.

- KNIGHT, D. M. **Elicitação de Requisitos de Software a partir do Modelo de Negócio**. Dissertação (Mestrado em Informática). Núcleo de Computação Eletrônica (NCE), Universidade Federal do Rio de Janeiro. Rio de Janeiro, 2004.
- LEITE, J. C. S. P.; DOORN, J. H. **Perspectives on Software Requirements**. Kluwer Academic Press, 2003.
- MAFFEO, B. **Engenharia de Software e Especificação de Sistemas**. Rio de Janeiro: Campus, 1992.
- NUSEIBEH, B.; EASTERBROOK, S. **Requirements Engineering: A Roadmap**. ACM - Future of Software Engineering. 2000.
- ODEH, M.; KAMM, R. **Bridging the gap between business models and system models**. Information and Software Technology, v. 45, n. 15, p. 1053-1060, 2003.
- OMG Unified Modeling Language, **UML Superstructure Version 2.3**, 2010.
- PÊSSOA, M.; STORCH, S. **Escolhas tecnológicas para o gerenciamento por processos**. São Paulo: Atlas, 2008.
- PFLEEGER, S. L. **Engenharia de Software: Teoria e Prática**. 2<sup>o</sup> ed. Trad. Dino Franklin. São Paulo: Prentice Hall, 2004.
- PRESSMAN, R. S. **Engenharia de Software**. São Paulo: McGraw-Hill, 2002.
- REZENDE, D. A. **Engenharia de Software e Sistemas de Informação**. Rio de Janeiro: Brasport, 2002.
- SANTOS, R.P.C. **As Tarefas para Gestão de Processos**. Tese (Doutorado em Engenharia de Produção). COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2007.
- \_\_\_\_\_. **Engenharia de Processos: Análise do Referencial Teórico-Conceitual, Instrumentos, Aplicações e Casos**. Dissertação de Mestrado. COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2002.
- SHEN, H.; WALL, B.; ZAREMBA, M.; CHEN, Y.; BROWNE, J. **Integration of business modeling methods for enterprise information system analysis and user requirements gathering**. Computers in Industry, v. 54, n. 3, 2004.
- SILVEIRA, D.S.; CRUZ, P.O.; SCHMITZ, E.A. **Heurísticas para Extração dos Casos de Uso de Negócio a Partir da Modelagem de Processos**. XI Congresso Latino Ibero-Americano de Investigación de Operaciones (CLAIO2002). Concepción (Chile), 2002.
- SOMMERVILLE, I. **Engenharia de Software**. São Paulo: Addison Wesley, 2003.

SOMMERVILLE, I.; SAWYER, P. **Requirements Engineering: a good practice guide**. New York: John Wiley & Sons, 1999.

SPÍNOLA, R. O.; ÁVILA, A. L. **Introdução à Engenharia de Requisitos**. Engenharia de Software Magazine, Ed. Especial, 2007.

THOM, L. H. **Aplicando o Conhecimento sobre os Aspectos Estruturais da Organização no Processo de Modelagem de Workflow**. Dissertação (Mestrado em Ciência da Computação), Universidade Federal do Rio Grande do Sul, Porto Alegre, RS, 2002.

VALLE, R.; OLIVEIRA, S. B. (org). **Análise e Modelagem de Processos de Negócio: Foco na notação BPMN (Business Process Modeling Notation)**. São Paulo: Atlas, 2009.

VERNADAT, F.B. **Enterprise Modeling and Integration: principles and applications**. Londres: Chapman & Hall, 1996.

VICENTE, L. S.S. **Modelagem de Processos e Linguagem de Modelagem Unificada: uma análise crítica**. Dissertação (Mestrado em Engenharia de Produção) COPPE, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2004.

YU, E. **Modeling Strategic Relationships for Process Reengineering**. Tese de PhD, Universidade de Toronto, Toronto, 1995.