

**UNIOESTE – Universidade Estadual do Oeste do Paraná**

**CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS**

**Colegiado de Ciência da Computação**

***Curso de Bacharelado em Ciência da Computação***

**BP2UC: TÉCNICA PARA DERIVAR CASOS DE USO A  
PARTIR DE MODELOS BPMN**

*Thiago Pessini*

**CASCADEL**

**2014**

**THIAGO PESSINI**

**BP2UC: TÉCNICA PARA DERIVAR CASOS DE USO A PARTIR DE  
MODELOS BPMN**

Monografia apresentada como requisito parcial  
para obtenção do grau de Bacharel em Ciência  
da Computação, do Centro de Ciências Exatas  
e Tecnológicas da Universidade Estadual do  
Oeste do Paraná - Campus de Cascavel

Orientador: Prof. Dr. Victor Francisco Araya  
Santader

CASCADEL

2014

**THIAGO PESSINI**

**BP2UC: TÉCNICA PARA DERIVAR CASOS DE USO A PARTIR DE  
MODELOS BPMN**

Monografia apresentada como requisito parcial para obtenção do Título de *Bacharel em Ciência da Computação*,  
pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos  
professores:

---

Prof. Dr. Victor Francisco Araya Santader

(Orientador)

Colegiado de Ciência da Computação,

UNIOESTE

---

Prof. Dr. Ivonei Freitas da Silva

Colegiado de Ciência da Computação,

UNIOESTE

---

Prof. Me. Elder Elisandro Shemberger

Colegiado de Sistemas para Internet,

UTFPR - Toledo

---

Prof. Me. Sidgley Camargo de Andrade

Colegiado de Sistemas para Internet,

UTFPR - Toledo

Cascavel, 21 de novembro de 2014.

## **DEDICATÓRIA**

Dedico este trabalho aos meus pais e minha família por sempre me incentivarem nesta árdua caminhada.

## **AGRADECIMENTOS**

Agradeço primeiramente aos meus pais por sempre me apoiarem nas decisões, não medindo esforços para fornecerem todos os recursos necessários para a realização desta caminhada. Agradeço aos meus irmãos pelos valiosos conselhos e orientações.

Ao meu orientador, professor Victor Francisco Araya Santader, pela gigantesca compreensão, conselhos, apoio e disponibilidade. Além de ser um grande pesquisador mostrou-se uma grande pessoa. Agradeço aos demais membros da banca, professor Ivonei Freitas da Silva, Elder Elisandro Schemberger e Sidgley Camargo de Andrade pelas valiosas observações, avaliações e sugestões no trabalho.

Agradeço também ao setor de Produto da Wealth Systems por fornecer recursos, pela disponibilidade e compreensão que tiveram durante a pesquisa.

Por fim, agradeço os meus amigos pelos conselhos, conversas, brincadeiras e cervejas divididas.

# LISTA DE FIGURAS

Figura 3.1: Exemplo de um Diagrama de Casos de Uso. ....	16
Figura 4.1: Exemplo de um processo de negócio de pagamento e entrega de produtos (retirado de [27]).....	20
Figura 4.2: Processo de negócio melhorado (retirado de [27]).....	21
Figura 4.3: Diagrama de Casos de Uso resultantes da aplicação do método (retirado de [27]). ....	22
Figura 4.4: Processo de negócio de admissão médica em uma clínica (retirado de [28]). ....	24
Figura 4.5: Casos de Uso gerais do processo de negócio de admissão médica (retirado de [28]).....	25
Figura 4.6: Casos de Uso específicos para tratar da segurança do processo de negócio de admissão médica (retirado de [28]). ....	26
Figura 5.1: Processo de negócio relacionado a entrega de pizza. ....	29
Figura 5.2: Parte inteira do Sub Process “Finalizar pedido”.....	30
Figura 5.3: Diagrama de atividades do BP2UC.....	31
Figura 5.4: Diagrama de Casos de Uso gerado a partir da aplicação das diretrizes.....	41
Figura 6.1: Visão geral da metodologia empírica. ....	44
Figura 6.2: Diagrama de Casos de Uso originado a partir da aplicação das diretrizes no processo de negócio. ...	47
A.1: Processo de negócio disponibilizado para o Estudo de Caso. ....	51
A.2: Parte interna do Sub Process “Report new project” da Figura A.1. ....	52
A.3: Parte interna do Sub Process “Planning Project” da Figura A.1. ....	52
A.4: Parte interna do Sub Process “Functional Specification” da Figura A.1. ....	52
A.5: Parte interna do Sub Process “Production” da Figura A.1.....	53
A.6: Parte interna do Sub Process “Create User Manuals” da Figura A.1.....	53
A.7: Parte interna do Sub Process “Closure Project” da Figura A.1.....	53

# LISTA DE TABELAS

Tabela 2.1: Events do conjunto simplificado disponíveis na versão 2.0 do BPMN. ....	8
Tabela 2.2: Activities disponíveis na versão 2.0 do BPMN.....	9
Tabela 2.3: Gateway disponível na atual especificação 2.0 do BPMN.....	10
Tabela 2.4: Connecting Objects disponíveis na atual especificação do BPMN.....	10
Tabela 2.5: Swimlanes disponíveis na versão 2.0 do BPMN.....	11
Tabela 2.6: Artifacts disponíveis na versão 2.0 do BPMN. ....	11
Tabela 2. 7: Data Object disponível na versão 2.0 do BPMN. ....	12
Tabela 3.1: Elementos disponível no diagrama de Casos de Uso [26]. ....	15
Tabela 5.1: template para a tabela de auxílio no 3º passo das diretrizes.....	37
Tabela 5. 2: template para a tabela auxiliar da Diretriz 4.2.1. ....	37
Tabela 5.3: tabela auxiliar que contém informações relacionadas aos Casos de Uso obtidos. ....	40
Tabela 5.4: tabela auxiliar que contém informações relacionadas aos passos dos Casos de Uso. ....	41
Tabela 5.5: Comparação entre as propostas de derivação.....	42
Tabela 6.1: Resultados das questões conforme as métricas. ....	47

# LISTA DE ABREVIATURAS E SIGLAS

BPMN	Business Process Model and Notation
LES	Laboratório de Engenharia de Software
JGOOSE	Java Goal Into Object Oriented Standard Extension
UML	Unified Modeling Language
E4J	Editor 4 JGOOSE
BPM	Business Process Management
BPML	Business Process Modeling Language
XML	Extensible Markup Language
BPEL	Business Process Execution Language
BPMI	Business Process Management Initiative
OMG	Object Management Group
BPD	Business Process Diagram
DQ	Data Quality
QVT	Query/View/Transformation
BP2UC	Business Process To Use Cases
GQM	Goal/Question/Metric

# SUMÁRIO

<b>Lista de Figuras .....</b>	<b>vi</b>
<b>Lista de Tabelas.....</b>	<b>vii</b>
<b>Lista de Abreviaturas e Siglas .....</b>	<b>viii</b>
<b>Sumário .....</b>	<b>ix</b>
<b>Resumo .....</b>	<b>xi</b>
<b>Introdução .....</b>	<b>1</b>
1.1 Contexto .....	1
1.2 Motivação .....	2
1.3 Proposta .....	3
1.4 Contribuições esperadas .....	3
1.5 Estrutura do trabalho .....	3
<b>Modelagem Organizacional e BPMN .....</b>	<b>4</b>
2.1 Modelagem Organizacional.....	4
2.2 BPMN.....	6
2.2.1 Histórico .....	6
2.2.2 Tipos de diagramas .....	7
2.2.3 Elementos .....	7
2.3 Considerações Finais do Capítulo .....	12
<b>Casos de Uso UML .....</b>	<b>13</b>
3.1 Casos de Uso .....	13
3.2 Diagrama de Casos de Uso.....	15
3.3 Considerações Finais do Capítulo .....	17
<b>Integração de Processos de Negócio com Casos de Uso UML – Trabalhos Relacionados</b>	<b>18</b>
4.1 BPiDQ* .....	18
4.2 Transformações QVT .....	22
4.3 Considerações Finais do Capítulo .....	27
<b>Diretrizes para a derivação de Casos de Uso a partir de modelos BPMN .....</b>	<b>28</b>

5.1 BP2UC.....	28
5.2 Exemplo de Uso .....	37
5.3 Diferenciando o BP2UC em relação às técnicas BPiDQ* e QVT .....	42
5.4 Considerações Finais do Capítulo .....	42
<b>Estudo de Viabilidade.....</b>	<b>43</b>
6.1 Metodologia.....	43
6.2 Estudo de Viabilidade .....	45
6.3 Considerações Finais do Capítulo .....	48
<b>Conclusões .....</b>	<b>49</b>
7.1 Resultados.....	49
7.2 Conclusões.....	49
7.3 Limitações .....	50
7.4 Trabalhos Futuros .....	50
<b>Apêndice A .....</b>	<b>51</b>
<b>Apêndice B .....</b>	<b>54</b>
<b>Referências Bibliográficas .....</b>	<b>62</b>

# RESUMO

Uma das etapas mais críticas do desenvolvimento de software é a Engenharia de Requisitos. Nesta etapa, os modelos organizacionais buscam garantir um bom entendimento do domínio da aplicação para auxiliar a correta elicitação, refinamento e validação dos requisitos. Dentre as técnicas de modelagem organizacional que estão se popularizando e ganhando o mercado está o BPMN. Por outro lado, modelos organizacionais como BPMN e outros como i\*, devem ser utilizados para gerar especificações de requisitos mais completas e consistentes. Neste sentido, cabe destacar que o grupo de pesquisa “Laboratório de Engenharia de Software” (LES) tem desenvolvido a ferramenta denominada JGOOSE, a qual permite gerar Casos de Uso a partir de modelos organizacionais i\*. Contudo, o framework i\* precisa ser complementado pela técnica BPMN para gerar casos de uso mais completos e consistentes. Desta forma, este trabalho apresenta a criação de diretrizes para derivar Casos de Uso a partir de modelos baseados em BPMN, visando fornecer a Engenheiros de Software e analistas uma abordagem bem definida que auxilie neste processo. As diretrizes propostas são aplicadas a um estudo de viabilidade. Em trabalhos futuros essas diretrizes serão suportadas pela ferramenta JGOOSE e os Casos de Uso gerados a partir de BPMN serão confrontados a partir do framework i\*.

**Palavras-chave:** Modelagem Organizacional, BPMN, Casos de Uso.

# Capítulo 1

## Introdução

Este primeiro capítulo tem como objetivo a apresentação geral do trabalho. É realizada a contextualização e delimitação da pesquisa no escopo da Engenharia de Software, bem como são destacados os principais objetivos da pesquisa. Na seção 1.1, é apresentado o contexto do problema a ser abordado. Na seção 1.2, são apresentadas as motivações para a realização do trabalho. Na seção 1.3 é apresentada uma visão geral da proposta. Em seguida, na seção 1.4, descreve-se as contribuições esperadas do trabalho. Por fim, na seção 1.5, é apresentada a estrutura geral desta monografia.

### 1.1 Contexto

A área de Engenharia de Requisitos, subárea da Engenharia de Software, é responsável por diversas atividades que abrangem os processos de análise, elicitação, especificação, avaliação, ajuste, documentação e evolução dos requisitos de um sistema computacional [1]. É uma das áreas mais críticas para o sucesso e qualidade de um projeto de software [2].

Visando diminuir os problemas relacionados às fases iniciais do projeto, pesquisas recentes mostram que a comunidade tem buscado estabelecer e utilizar padrões de técnicas, métodos e ferramentas para tratar especificamente da fase inicial de desenvolvimento de software [3]. Neste contexto, surge como principal etapa, a Modelagem Organizacional, que fornece recursos que permitem modelar as intenções, relacionamentos e motivações entre os membros de uma organização [4].

Dentre as várias técnicas que podem ser utilizadas na modelagem organizacional destaca-se o *Business Process Model and Notation* (BPMN). A notação está conquistando cada vez mais espaço no mercado. Esta técnica propõe a modelagem dos processos através de um conjunto de procedimentos e atividades, executados seguindo uma ordem pré-definida, que

coletivamente compõe um objetivo de negócio ou meta política, dentro de um contexto de uma estrutura organizacional que define papéis funcionais ou relacionamentos [5]. Esta modelagem, no contexto da Engenharia de Requisitos, auxilia no entendimento dos processos da organização e permite estabelecer uma compreensão comum a cerca desses processos de negócios antes de definitivamente iniciar a elicitação e análise dos requisitos do sistema computacional pretendido.

Contudo, esses modelos organizacionais devem ser integrados de uma forma sistemática à outras etapas do processo de Engenharia de Software. Observando este desafio, a comunidade da área tem procurado meios para auxiliar neste processo de integração. No nosso trabalho investigamos como o BPMN pode ser utilizado para gerar requisitos do sistema a ser desenvolvido.

## 1.2 Motivação

Alguns trabalhos presentes na literatura buscam integrar Modelos Organizacionais com Casos de Uso UML [6] para auxiliar no processo de desenvolvimento de software. Entre eles destaca-se o trabalho realizado por Santander [7], que propõe diretrizes para a conversão de Modelos Organizacionais baseados no framework *i\** [8] (lê-se *i-star*) em Casos de Uso UML. Mais tarde essas diretrizes foram implementadas no *Java Goal Into Object Oriented Standard Extension* (JGOOSE) [9] para a derivação automática dos Casos de Uso UML a partir de diagramas do *i\**. Atualmente o JGOOSE está integrado ao E4J [10], um editor de modelos *i\** desenvolvido no contexto do JGOOSE.

No entretanto, existem poucos trabalhos na literatura que integram o BPMN a outros artefatos e/ou fases do processo de desenvolvimento. Observando esta lacuna, motivamos o fato de encontrar processos que apoiem engenheiros de software nesta integração. Como já comentado, o uso de BPMN vêm crescendo principalmente na área industrial e engenheiros de software utilizam estes modelos como base para evoluir em direção a requisitos funcionais de sistema computacionais.

Contudo, este processo em sua maioria é *ad hoc* ou ainda modelos BPMN gerados acabam não sendo utilizados. Isto gera um desperdício de tempo e esforço. Assim, a proposta de integração de BPMN a requisitos funcionais apoiado por diretrizes e ferramentas computacionais pode reduzir estes problemas. Além disso, a ferramenta JGOOSE citada, poderá também contemplar a derivação de requisitos via Casos de Uso UML a partir de

BPMN. Isto permitirá confrontar as propostas de derivação dos Casos de Uso UML a partir do framework i\* com a proposta de derivação dos Casos de Uso UML a partir de BPMN.

## 1.3 Proposta

Neste trabalho, apresenta-se a criação de diretrizes, chamadas de BP2UC (*Business Process To Use Cases*) para a derivação de Casos de Uso UML a partir de modelos BPMN. O BP2UC busca auxiliar engenheiros de software e analistas no processo de derivação de Casos de Uso a partir de modelos organizacionais criados com BPMN.

## 1.4 Contribuições esperadas

Espera-se que o BP2UC possibilite um meio para derivar Casos de Uso a partir de modelos BPMN e facilitem o árduo trabalho de Engenheiros de Requisitos na elicitação e análise de requisitos, pois ele oferece uma forma sistemática de se obter os Casos de Uso, não sendo necessária uma interpretação pessoal do analista do processo de negócio.

## 1.5 Estrutura do trabalho

Na sequência, o trabalho encontra-se organizado da seguinte maneira:

- **Capítulo 2:** são apresentados os fundamentos teóricos da modelagem organizacional. Será mostrada a notação BPMN.
- **Capítulo 3:** são apresentados os fundamentos teóricos necessários para a compreensão dos diagramas e descrição de Casos de Uso UML.
- **Capítulo 4:** são apresentadas algumas propostas de derivação de Casos de Uso UML a partir de extensões de modelos BPMN.
- **Capítulo 5:** será detalhado o BP2UC e um exemplo da utilização do mesmo para gerar um diagrama de Casos de Uso UML a partir de um modelo BPMN.
- **Capítulo 6:** é apresentado um estudo de viabilidade.
- **Capítulo 7:** são feitas as conclusões finais.

## Capítulo 2

# Modelagem Organizacional e BPMN

Neste capítulo são apresentados os conceitos básicos necessários para o entendimento da Modelagem Organizacional, com foco na descrição da técnica BPMN que está ganhando muito espaço nas empresas para a modelagem de seus processos de negócio. Inicialmente, na seção 2.1, são apresentados os conceitos necessários para o entendimento básico da Modelagem Organizacional. Em seguida, na seção 2.2, é apresentada a notação BPMN. Por fim, na seção 2.3, são realizadas as considerações finais do capítulo.

## 2.1 Modelagem Organizacional

Não há dúvidas da importância da informação para as organizações, sendo, se não o mais importante, ao menos um dos recursos cuja gestão e aproveitamento se relacionam diretamente ao sucesso desejado. Como apresentado por [11], a principal diferença entre uma empresa de sucesso e uma empresa com sérios problemas, na maioria dos casos, pode estar relacionada ao gerenciamento dos processos de negócios.

O gerenciamento de processos de negócio (BPM, em inglês *Business Process Management*) é uma abordagem disciplinada para identificar, desenhar, executar, documentar, medir, monitorar, controlar e melhorar processos de negócio automatizados ou não a fim de alcançar os resultados desejados de forma consistente e alinhados com as metas estratégicas de uma organização [12]. Por outro lado, também é importante compreender quais são e como ocorrem às relações entre os membros da organização para poder compreender melhor seus processos e características.

Buscando auxiliar os atores da organização a atingirem seus objetivos específicos e as metas gerais da empresa, um sistema de informação visa automatizar tarefas ou atividades de um processo de negócio. Desta maneira, o estudo do ambiente organizacional no qual o

software será inserido tem sido reconhecido como parte fundamental da Engenharia de Requisitos [13]. Grande parte das técnicas tradicionalmente utilizadas no desenvolvimento de software focam em aspectos relacionados às funcionalidades do sistema, a descrição de atividades e entidades, as entradas que serão processadas e as saídas que deverão ser produzidas.

Contudo, tais técnicas não consideram outros aspectos, como os objetivos da organização, as regras de negócio, as restrições e aspectos não funcionais ligados à organização, que também são importantes no contexto do desenvolvimento do sistema de informação. Devido a isso, é necessária uma abordagem mais rica, capaz de facilitar os esforços da Engenharia de Requisitos em obter uma melhor compreensão entre os relacionamentos da organização e os vários atores do sistema.

Neste contexto surge a Modelagem Organizacional, sendo uma abordagem que facilita a compreensão do ambiente organizacional e por isso é considerada uma atividade valiosa para a Engenharia de Requisitos [13]. Os objetivos da Modelagem Organizacional, segundo [14] são:

- a) fornecer um objeto, que seja uma representação compartilhável e reusável da cadeia de fornecimento de informação e conhecimento;
- b) suportar tarefas da cadeia de fornecimento, pela habilitação de respostas a questionamentos, que não estão explicitamente representados no modelo;
- c) definir os objetos de maneira precisa, de forma que sejam consistentemente aplicados, por meio dos domínios e interpretados pelos usuários;
- d) suportar visualização do modelo, de forma intuitiva, simples e consistente.

Dentre as técnicas de modelagem organizacional destaca-se o BPMN, que está se popularizando tanto no mercado industrial quanto na área acadêmica por ser uma notação simples e extremamente útil. Neste contexto cabe destacar o estudo comparativo entre técnicas de modelagem organizacional realizado em [15].

Neste trabalho, chegou-se a conclusão que a técnica BPMN possui uma notação gráfica simples, sendo considerada uma técnica de fácil aprendizado e entendimento por sustentar a noção de fluxogramas, mesmo sem possuir uma descrição textual dos seus elementos. No entanto, a técnica BPMN não permite a inserção de um sistema computacional na modelagem da organização, resultando em uma maior dificuldade na identificação das funcionalidades

que vão ser alocadas ao novo software a ser inserido. Contudo, apesar desta dificuldade ainda é possível gerar modelos funcionais a partir de seus modelos de negócio.

## 2.2 BPMN

O BPMN é uma notação gráfica que retrata as etapas de um processo de negócio. A notação foi projetada especificamente para coordenar a sequência de processos e as mensagens que fluem entre os diferentes participantes do processo em um conjunto de atividades relacionadas [16]. O principal objetivo do BPMN é fornecer uma notação que é facilmente compreensível por todos os usuários de negócios, desde os analistas de negócios que criam os rascunhos iniciais dos processos, para os desenvolvedores técnicos responsáveis pela implementação da tecnologia que irá executar os processos e, finalmente, para as pessoas de negócios que irão gerenciar e monitorar os processos [17].

Desta forma, o BPMN cria uma ponte padronizada para o espaço entre o desenho de processos de negócios e implementação de processos [17].

### 2.2.1 Histórico

Em 2001, durante o desenvolvimento da BPML (*Business Process Modeling Language*) [18], uma linguagem de execução de processos XML (*Extensible Markup Language*) [19] que mais tarde seria substituída pela BPEL (*Business Process Execution Language*) [20], a BPML.org (*Business Process Management Initiative*) [21] percebeu a necessidade de uma representação gráfica para os processos de negócio. Tal representação gráfica deveria ser orientada as necessidades de usuários de negócio e não representar diretamente uma linguagem precisa de execução para desenvolvimento.

Então, em agosto de 2001, formou-se o *Notation Working Group*. Composto de 35 companhias, organizações e indivíduos que possuíam grandes perspectivas de modelagem, desenvolveu-se a primeira versão do BPMN em conjunto com a BPML.org. Entretanto, ela foi disponibilizada para o público apenas em maio de 2004.

Visando fornecer padrões de liderança na área de processos de negócio [21], em junho de 2005 o setor de BPM (*Business Process Management*) da BPML.org foi fundida com a OMG (*Object Management Group*) [22]. Como resultado, em fevereiro de 2006 o BPMN foi adotado como um padrão pela OMG [23], sendo que no início de 2008 foi disponibilizada a versão 1.1 do BPMN e, em janeiro de 2009, foi liberada a versão 1.2 da notação. A última

versão (2.0) foi lançada em janeiro de 2011 e apresentou uma série de melhorias em relação às versões anteriores.

### 2.2.2 Tipos de diagramas

O BPMN oferece suporte a três tipos de diagramas em sua atual versão [17]:

- **Diagrama de Processos de Negócio** (ou BPD, do inglês *Business Process Diagram*): Modela o conjunto de elementos de um fluxo. Existem três tipos básicos de BPD:
  - **Processo de Negócio Privado Executável:** É interno a uma organização específica e é modelado para ser executado efetivamente segundo a semântica da notação BPMN [17];
  - **Processo de Negócio Privado Não Executável:** É interno a uma organização e é modelado com o propósito de documentar o comportamento do processo em um nível de detalhe fornecido pelo modelador [17];
  - **Processo de Negócio Público:** Representa a interação entre um Processo de Negócio Privado e algum outro Processo ou Participante [17].
- **Diagrama de Coreografia:** Define a sequência de interações entre os Participantes de um processo [17].
- **Diagrama de Colaboração:** Mostra apenas os Participantes de um processo e suas interações. O Diagrama de Conversação, por ser uma versão simplificada e manter todas as características, é utilizado como uma descrição informal do Diagrama de Colaboração [17].

No contexto do nosso trabalho, torna-se mais interessante utilizar os BPD como técnica para Modelagem Organizacional, pois através dele é possível modelar todo o fluxo de trabalho da organização. Desta maneira, é possível saber “quem” faz “o que” dentro da organização e com base nisso extrair os Casos de Uso UML.

### 2.2.3 Elementos

O BPMN apresenta uma grande quantidade de elementos gráficos, sendo estes divididos em quatro categorias de acordo com as funções que cada um pode desempenhar nos

diagramas. Além disso, a notação possui dois conjuntos de elementos gráficos: o primeiro conjunto é simplificado e atende grande parte dos processos tipicamente encontrados; e o segundo conjunto é avançado, contendo a lista de elementos simplificados acrescentados de outros recursos que permitem a criação de modelos mais complexos além de suportar o mapeamento para linguagens de execução. A seguir serão apresentados os elementos simplificados disponíveis para a modelagem de Diagramas de Processos de Negócio [17].

1. **Flow Objects:** são os principais elementos gráficos que definem o comportamento do processo de negócio. Eles são subdivididos em três categorias: Events, Activities e Gateways;

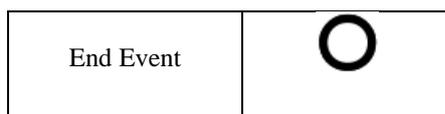
a) **Events:** de acordo com WHITE e MIERS [23], um **Event** indica algo que “acontece” durante a execução do processo, afetando desta forma o fluxo do mesmo. Um Event é representado por um círculo, podendo ou não conter um marcador em seu centro que indica sua função. Events são divididos em três categorias, diferenciados graficamente por suas bordas:

- **Start Events:** indicam onde um processo começa e são representados por um círculo com uma borda simples;
- **Intermediate Events:** indicam que algum evento ocorreu entre o começo e o final do processo e são representados por um círculo duplo;
- **End Events:** indicam quando um processo acaba e são representados por um círculo de borda simples, porém mais espessa que os eventos de início.

A Tabela 2.1 apresenta os Events do conjunto simplificado de elementos disponíveis na atual versão 2.0 do BPMN.

Tabela 2.1: Events do conjunto simplificado disponíveis na versão 2.0 do BPMN.

Elemento	Símbolo gráfico
Start Event	
Intermediate Event	

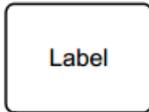


b) **Activities:** Segundo [17], uma **Activity** é um trabalho que acontece em um Processo de Negócio, podendo ou não ser atômico. Quando uma Activity é atômica ela é chamada de **Task**, caso contrário é conhecido como **Sub Process**.

- **Tasks:** São representadas por um retângulo de borda fina. Uma Task é um trabalho que não pode ser quebrado em um nível menor de detalhamento.
- **Sub Process:** São representados por um retângulo de borda fina com um marcador em forma de “+” na parte central inferior quando estão na forma contraída. Em um Sub Process, por ele ser uma Activity não atômica, seus detalhes internos são modelados usando outras Activities, Gateways, Events e Sequence Flows.

A Tabela 2.2 apresenta as Activities disponíveis na atual especificação 2.0 do BPMN.

Tabela 2.2: Activities disponíveis na versão 2.0 do BPMN.

Elemento	Símbolo gráfico
Task	
Sub Process	

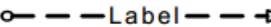
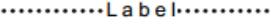
c) **Gateways:** Segundo [17], os Gateways são usados para controlar como o Sequence Flow converge e diverge dentro de um processo. Eles são representados por um losango. Um Gateway pode conter múltiplos fluxos de entradas e saídas. A Tabela 2.3 apresenta o Gateway disponível para o conjunto simplificado de elementos na especificação 2.0 do BPMN.

Tabela 2.3: Gateway disponível na atual especificação 2.0 do BPMN.

Elemento	Símbolo gráfico
Gateway	

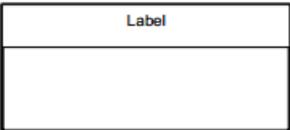
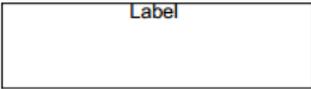
2. **Connecting Objects:** Um Connection Object tem como função conectar dois Flow Objects, criando desta forma a estrutura básica de um processo de negócio. Na atual especificação do BPMN existem três tipos diferentes de Connection Objects: o Sequence Flow define a ordem em que as atividades são executadas em um processo de negócio e é representado por uma seta sólida; o Message Flow define a troca de mensagem entre dois participantes do processo e é representado por uma seta tracejada; e a Association associa dados, anotações ou outros artefatos com um Flow Object e é representado por uma seta ou reta pontilhada. A Tabela 2.4 apresenta os tipos de Connection Objects disponíveis para o conjunto simplificado de elementos.

Tabela 2.4: Connecting Objects disponíveis na atual especificação do BPMN.

Elemento	Símbolo gráfico
Sequence Flow	
Message Flow	
Association	

3. **Swinlanes:** Uma swinlane agrupa os elementos de modelagem de forma a organizar Activities separadas visualmente por categorias. Na atual especificação do BPMN existem dois elementos para realizar esta função: a Pool representa um participante do processo; e a Lane é uma subdivisão dentro de uma Pool, usado para organizar e categorizar as Activities. A Tabela 2.5 apresenta as swinlanes suportadas pelo BPMN.

Tabela 2.5: Swimlanes disponíveis na versão 2.0 do BPMN.

Elemento	Símbolo gráfico
Horizontal Pool	
Vertical Pool	
Horizontal Lane	
Vertical Lane	

4. **Artifacts:** Artifacts fornecem informações adicionais sobre o processo. São subdivididos em duas categorias: Groups reúnem diferentes Activities sem afetar o fluxo do diagrama; e as Annotations auxiliam na compreensão do diagrama. A Tabela 2.6 apresenta os Artifacts disponíveis.

Tabela 2.6: Artifacts disponíveis na versão 2.0 do BPMN.

Elemento	Símbolo gráfico
Group	
Annotation	

5. **Data Object:** São informações necessárias ou produzidas em uma Activity. Alguns autores tratam os Data Objects como parte dos Artifacts, contudo eles são descritos como outra categoria na especificação do BPMN [17]. A Tabela

2.7 apresenta o conjunto simplificado de Data Object disponíveis na versão 2.0 do BPMN.

Tabela 2. 7: Data Object disponível na versão 2.0 do BPMN.

<b>Elemento</b>	<b>Símbolo gráfico</b>
Data Object	

## **2.3 Considerações Finais do Capítulo**

Neste capítulo foram apresentados os conceitos gerais da notação BPMN. O estudo da notação BPMN faz-se necessário para a correta elaboração das diretrizes propostas neste trabalho. Cabe destacar a utilização dos diagramas do tipo BPD com apenas o conjunto de elementos simplificados para a modelagem organizacional.

# Capítulo 3

## Casos de Uso UML

Segundo [24], a UML é definida como uma linguagem gráfica para visualização, especificação, construção e documentação de artefatos de sistemas complexos de software, proporcionando desta maneira, uma forma padrão de preparação de planos de arquitetura de projetos de sistema.

Dentre os diversos diagramas e descrições definidos pela UML destacam-se os Casos de Uso. Um Caso de Uso “*especifica o comportamento de um sistema ou parte de um sistema e é uma descrição de um conjunto de sequências de ações, incluindo duas variantes realizadas pelo sistema para produzir um resultado observável do valor de um ator*” [24].

Neste sentido, este capítulo apresenta os conceitos necessários para a compreensão de Casos de Uso UML bem como dos elementos que compõe o diagrama de Casos de Uso. Compreender esta técnica é importante no nosso trabalho, pois os modelos BPMN serão utilizados para gerar Casos de Uso que irão descrever funcionalidades desejadas do processo de negócio que deseja-se automatizar.

Na seção 3.1 serão apresentados os conceitos fundamentais de Casos de Uso UML, bem como o *template* textual utilizado para sua descrição. A seção 3.2 apresenta alguns conceitos e os elementos disponíveis para a construção de diagramas de Casos de Uso. Por fim, na seção 3.3, são feitas as considerações finais do capítulo.

### 3.1 Casos de Uso

Casos de Uso podem ser utilizados para captar o comportamento desejado do sistema, sem a necessidade de descrever como esse comportamento é efetivamente implementado. Eles descrevem um conjunto de sequencias onde cada uma delas representa a interação de itens externos ao sistema com o próprio sistema. De acordo com [24], um Caso de Uso especifica o

comportamento de um sistema ou parte de um sistema, sendo uma descrição de um conjunto de sequências de ações, incluindo as possíveis variações nestas ações, que produzem um resultado observável de valor para um ator. Para [25], um Caso de Uso é uma unidade coerente de funcionalidade expressa como uma transação entre os atores e o sistema. Segundo [26], um Caso de Uso captura um contato dos *stakeholders* do sistema e seu comportamento. Desta maneira, os Casos de Uso fornecem uma maneira comum de compreensão dos requisitos do sistema tanto para os desenvolvedores, usuários finais quanto especialistas do domínio [24].

Um Caso de Uso é expresso na forma textual para dizer como o sistema se comporta, representando ações que devem ser realizadas entre o usuário e o sistema para concluir o objetivo associado a ele [24]. Neste trabalho adota-se o *template* para especificação textual de Casos de Uso proposto por Cockburn [26]. Este *template* define explicitamente os objetivos do Caso de Uso bem como os níveis associados a estes objetivos. Cabe ressaltar que este *template* foi adotado pela ferramenta JGOOSE para a especificação dos Casos de Uso e também será adotado neste trabalho. A seguir é apresentado o *template*:

**Use Case:** <nome> << o nome é um objetivo descrito com uma frase curta contendo um verbo na voz ativa >>

---

#### **CHARACTERISTIC INFORMATION**

**Goal in Context:** <uma sentença mais longa do objetivo do caso de uso se for necessário>

**Scope:** <Qual sistema está sendo considerado (por exemplo, organização ou sistema computacional)>

**Preconditions:** <o que é necessário que já esteja satisfeito para realizar o caso de uso>

**Success End Condition:** <o que ocorre/muda após a obtenção do objetivo do caso de uso>

**Failed End Condition:** <o que ocorre/muda se o objetivo é abandonado>

**Primary Actor:** <o nome do papel para o ator primário, ou descrição>

---

#### **MAIN SUCCESS SCENARIO**

<coloque aqui os passos do cenário necessários para a obtenção do objetivo >

<passo #> <descrição da ação >

---

#### **EXTENSIONS**

<coloque aqui as extensões, uma por vez, cada uma referenciando o passo associado no cenário principal >

<passo alterado> <condição> : <ação ou sub.caso de uso >

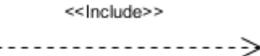
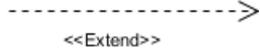
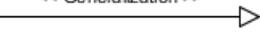
<passo alterado > <condição> : <ação ou sub.caso de uso >

Cabe destacar que, como os Casos de Uso representam aspectos funcionais e comportamentais do sistema, eles podem ser utilizados em outras etapas do processo de desenvolvimento de software.

## 3.2 Diagrama de Casos de Uso

Pode-se reunir vários Casos de Uso em um diagrama buscando descrever o que o sistema faz [24]. Para tal, existe um conjunto pré-definido de elementos que podem ser utilizados no diagrama. Tais elementos são apresentados na Tabela 3.1.

Tabela 3.1: Elementos disponível no diagrama de Casos de Uso [26].

Nome	Elemento
Ator (ou <i>Actor</i> )	
Caso de Uso (ou <i>Use Case</i> )	
Associação (ou <i>Association</i> )	
Inclusão (ou <i>Include</i> )	
Extensão (ou <i>Extend</i> )	
Generalização (ou <i>Generalization</i> )	

Um Ator pode ser definido como “*uma representação de um conjunto coerente de papéis que os usuários de Casos de Uso desempenham quando interagem com esses Casos de Uso*” [24]. Em [25] um Ator é definido como uma idealização de uma pessoa externa, processo ou algo interagindo com o sistema, subsistema ou classe, caracterizando desta forma as interações que os usuários externos podem ter com o sistema.

A Associação é um relacionamento estrutural que indica que objetos de um tipo estão conectados com objetos de outro tipo [24]. Uma ligação do tipo Associativa representa a comunicação entre Atores e Casos de Uso [25].

O relacionamento de Inclusão indica que um Caso de Uso base incorpora explicitamente o comportamento de outro Caso de Uso em um determinado local especificado no Caso de Uso base [24]. Este tipo de relacionamento é utilizado para evitar descrever o mesmo fluxo várias vezes, sendo que desta forma é possível criar um Caso de Uso específico com um comportamento padrão e todos os demais Casos de Uso relacionados podem fazer uso do mesmo, quando necessário.

O relacionamento de Generalização pode ocorrer entre Atores ou Casos de Uso. Em ambos os casos ela indica que o elemento filho herda o comportamento e o significado do elemento pai, acrescentando ou até mesmo sobrescrevendo o comportamento do pai [24]. O elemento filho pode ainda substituir o elemento pai em qualquer local onde o pai apareça.

Por fim, o relacionamento de Extensão indica que um Caso de Uso base incorpora implicitamente o comportamento de outro Caso de Uso em um determinado local especificado indiretamente no Caso de Uso estendido [24]. Este tipo de relacionamento é utilizado para indicar um comportamento opcional do sistema em relação a um Caso de Uso. A Figura 3.1 exemplifica um Diagrama de Casos de Uso.

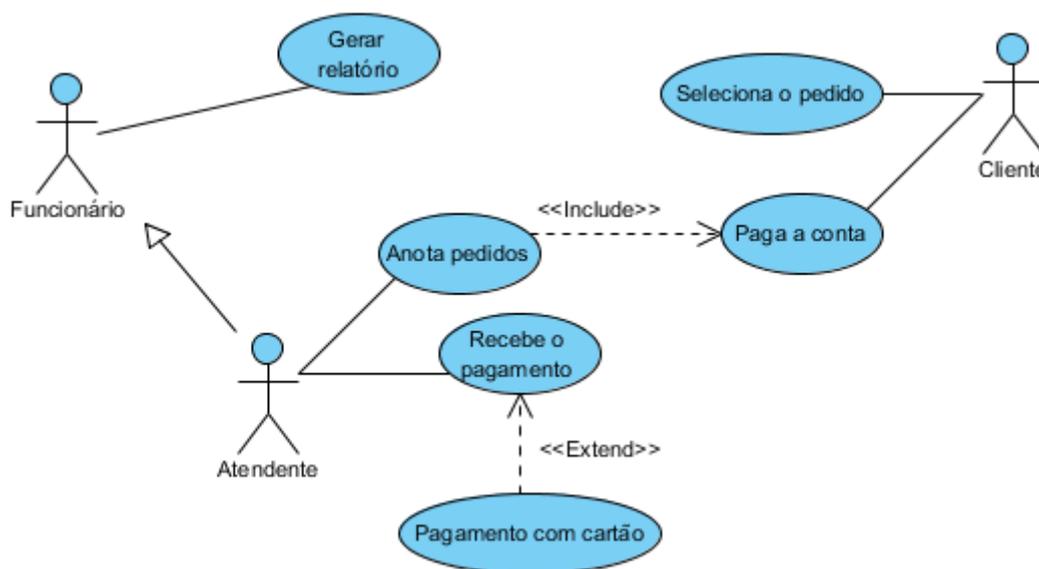


Figura 3.1: Exemplo de um Diagrama de Casos de Uso.

Neste diagrama, o Ator Atendente é uma generalização do Ator Funcionário. O Ator Funcionário está associado ao Caso de Uso “Gerar Relatório”. O Ator Cliente está associado aos Casos de Uso “Seleciona o pedido” e “Paga a conta”. O Ator Atendente está associado aos Casos de Uso “Anota Pedidos”, “Recebe o pagamento” e “Gerar Relatório” (pois o Atendente herda do Funcionário este comportamento). O Caso de Uso “Anota pedidos” inclui o Caso de Uso “Paga a conta”. O Caso de Uso “Pagamento com cartão” é uma extensão do Caso de Uso “Recebe o pagamento”, isto é, ele é uma ação opcional.

### **3.3 Considerações Finais do Capítulo**

Neste capítulo foram apresentados os conceitos gerais sobre Casos de Uso UML. O estudo desta técnica é de grande importância, pois seus conceitos são necessários para um melhor entendimento das diretrizes que serão apresentadas no Capítulo 5.

## Capítulo 4

# Integração de Processos de Negócio com Casos de Uso UML – Trabalhos Relacionados

Considerando os capítulos anteriores, neste capítulo serão abordadas algumas propostas presentes na literatura que integram Processos de Negócio descritos em extensões do BPMN com os Casos de Uso UML. O estudo de tais propostas é importante, pois servem de base para a elaboração das diretrizes que serão propostas neste trabalho no próximo capítulo.

Na seção 4.1 apresenta-se uma proposta de derivação de Casos de Uso focados na qualidade dos dados a partir de uma extensão do BPMN. Na seção 4.2 apresenta-se uma proposta de derivação de Casos de Uso a partir de processos de negócio seguros descritos com uma extensão do BPMN. Por fim, na seção 4.3 são realizadas as considerações finais do capítulo.

### 4.1 BPIDQ\*

Visando a especificação de requisitos inicial e com foco na **qualidade dos dados**, no trabalho realizado por [27] apresenta-se um método, chamado BPIDQ\*, no qual especialistas de negócios podem modelar processos de negócio cientes da qualidade dos dados visando buscar artefatos úteis para o restante do processo de desenvolvimento de software. Para tal, utiliza-se uma extensão da técnica BPMN para modelar os processos de negócio da organização com foco na qualidade dos dados a fim de se obter Casos de Uso UML que auxiliem nas demais etapas do processo de desenvolvimento do software. Segundo [28], qualidade dos dados (ou DQ, em inglês *Data Quality*) é uma característica essencial que determina a confiabilidade dos dados para a tomada de decisões. Desta maneira, busca-se garantir a qualidade dos dados que são armazenados no banco de dados da organização a fim

de proporcionar uma melhor gestão de seus processos de negócio e auxiliar na tomada de decisões.

Neste contexto, o BPIDQ\* foi desenvolvido para suportar a especificação de requisitos focados na qualidade dos dados a partir de processos de negócio e obter requisitos de software centrados na qualidade dos dados expressos na forma de Casos de Uso UML. O método consiste em quatro etapas:

1. **BPIDQ–S1 Modelagem de Processos de Negócio consciente da qualidade dos dados:** Nesta etapa busca-se a captura inicial dos requisitos de qualidade de dados representados em um modelo BPMN. Durante a modelagem, incorpora-se marcadores em alguns elementos do modelo (os marcadores são chamados de DQ-Flags) para estimar a qualidade dos dados. Como entrada deste processo tem-se o padrão BPMN e a extensão que permite incluir requisitos de qualidade de dados. Ao final deste processo obtém-se uma descrição do Processo de Negócio onde foram incluídas DQ-Flags que representam o interesse dos especialistas de negócio em uma melhor definição dos requisitos de qualidade dos dados que eles consideram ser importantes para o bom desempenho do Processo de Negócio.
2. **BPIDQ–S2 Especificação de requisitos de qualidade dos dados:** Nesta etapa busca-se especificar detalhadamente os requisitos de qualidade dos dados definidos no processo de negócio. Como entrada desta fase tem-se o modelo de Processo de Negócio com requisitos de qualidade dos dados expressos através das DQ-Flags. Ao final deste processo deve-se obter um modelo do Processo de Negócio com DQ-Flags e, para cada DQ-Flag, uma especificação detalhada que contém o elemento do Processo de Negócio associado a ela, a importância do requisito de qualidade de dados no Processo de Negócio (alta, média ou baixa), a probabilidade de execução da atividade associada a especificação do requisito de qualidade de dado, as dimensões de qualidade dos dados associadas, a sobrecarga do Processo de Negócio devido a incorporação de novas atividades associadas as dimensões de qualidade dos dados, o nome do elemento de dados envolvido no requisito de qualidade dos dados, sua descrição, meio de suporte e origem.
3. **BPIDQ–S3 Análise e melhora dos Processos de Negócio relacionados com a qualidade dos dados:** Nesta etapa analisa-se e busca-se melhorar o modelo de Pro-

cesso de Negócio levando em consideração os requisitos de qualidade dos dados especificados. Como entrada desta fase tem-se a descrição do Processo de Negócio com especificações de qualidade dos dados (DQ-Flags), o detalhamento das especificações de qualidade dos dados do Processo de Negócio e um repositório com as atividades que se relacionam com as dimensões de qualidade de dados. Ao final do processo deve-se obter uma descrição do Processo de negócio na qual foram incluídas novas atividades que consideram os requisitos de qualidade dos dados.

4. **BPiDQ-S4 Geração dos Diagramas de Caso de Uso:** Nesta etapa gera-se os Casos de Uso relacionados com a qualidade dos dados e também os Casos de Uso gerais do modelo BPMN. Como entrada desta fase tem-se a descrição do Processo de Negócio com as atividades de qualidade de dados agregadas e o repositório padrão de Casos de Uso relacionados a qualidade dos dados. Ao final do processo deve-se obter um conjunto de Casos de Uso UML que podem ser utilizados no processo de desenvolvimento de software.

Ao final desta técnica, obtém-se um diagrama de Casos de Uso com as principais funcionalidades do sistema, bem como funcionalidades específicas para a garantia da qualidade dos dados. Para exemplificar esta técnica, considere o modelo apresentado na Figura 4.1. Neste modelo, a letra 'D' representa os elementos que necessitam de atenção quanto a qualidade dos dados.

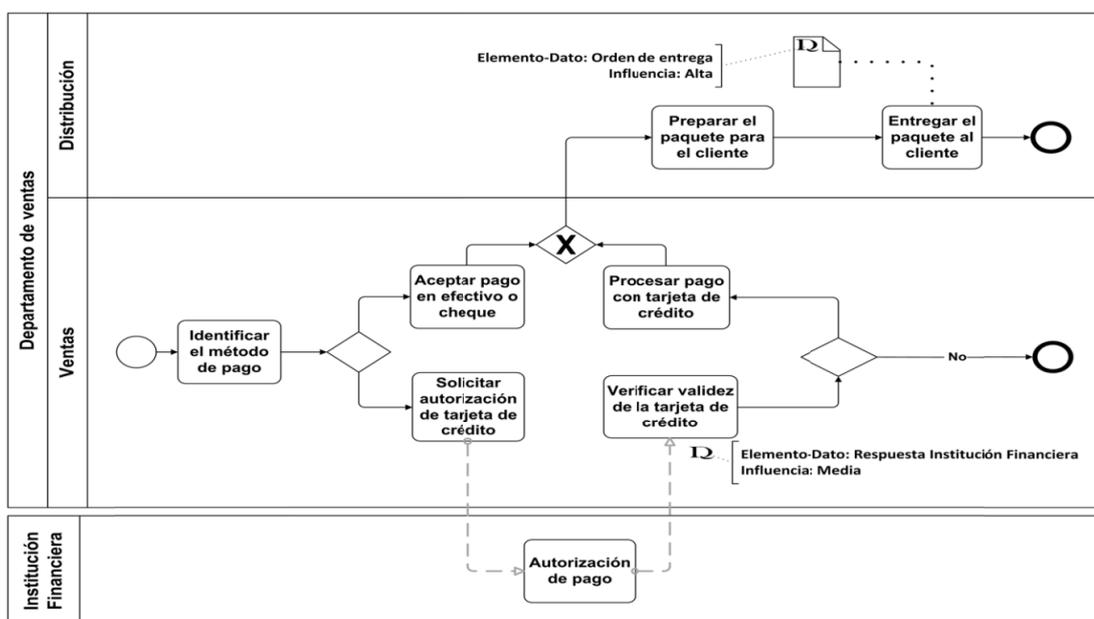


Figura 4.1: Exemplo de um processo de negócio de pagamento e entrega de produtos (retirado de [27]).

Neste processo de negócio o participante “Ventas” (que faz parte do “Departamento de Ventas”) identifica o método de pagamento e aceita, caso o método de pagamento for com dinheiro ou cheque, ou solicita autorização do cartão de crédito caso contrário. Em seguida, a “Institución Financiera” autoriza o pagamento e, após “Ventas” verificar a validade do cartão de crédito, ela processa o pagamento com o cartão de crédito. Por fim, a “Distribución” prepara o pacote para o cliente e entrega o pacote ao cliente.

Utilizando a técnica BPiDQ\*, adiciona-se marcadores nas Tasks “Verificar validez de la tarjeta de crédito” e “Entregar el paquete al cliente” que irão expressar a necessidade de qualidade dos dados (BPiDQ–S1). Em seguida, os analistas especificam requisitos de qualidade dos dados que serão associados a estas duas Tasks (BPiDQ–S2). Na terceira etapa, melhora-se o processo de negócio original para que ele passe a conter os requisitos de qualidade dos dados (BPiDQ–S3). A Figura 4.2 expressa o processo de negócio resultante desta etapa.

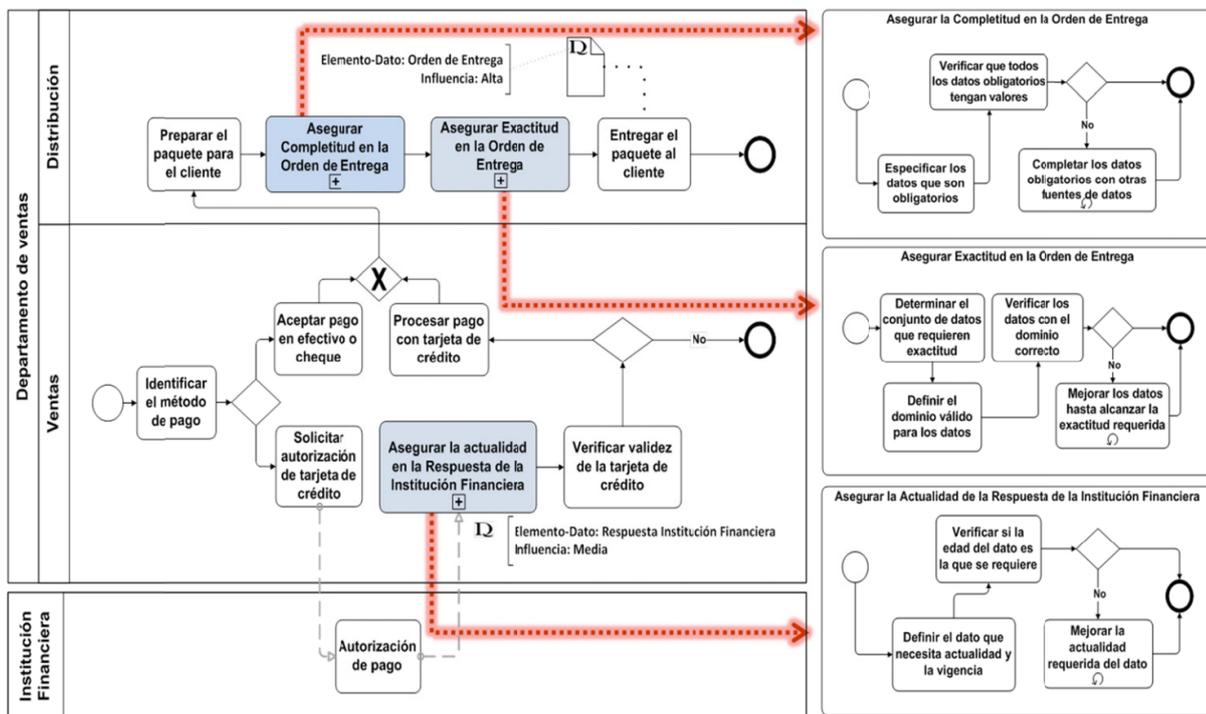


Figura 4.2: Processo de negócio melhorado (retirado de [27]).

Os Sub Process em azul são as Activities adicionadas para garantir a qualidade dos dados. As setas vermelhas direcionam para a parte interna das Activities que foram adicionadas. Por fim, na última etapa (BPiDQ–S3) gera-se o diagrama de Casos de Uso UML do processo de

negócio melhorado considerando os requisitos que qualidade dos dados. A Figura 4.3 mostra o diagrama de Casos de Uso UML gerados para este exemplo.

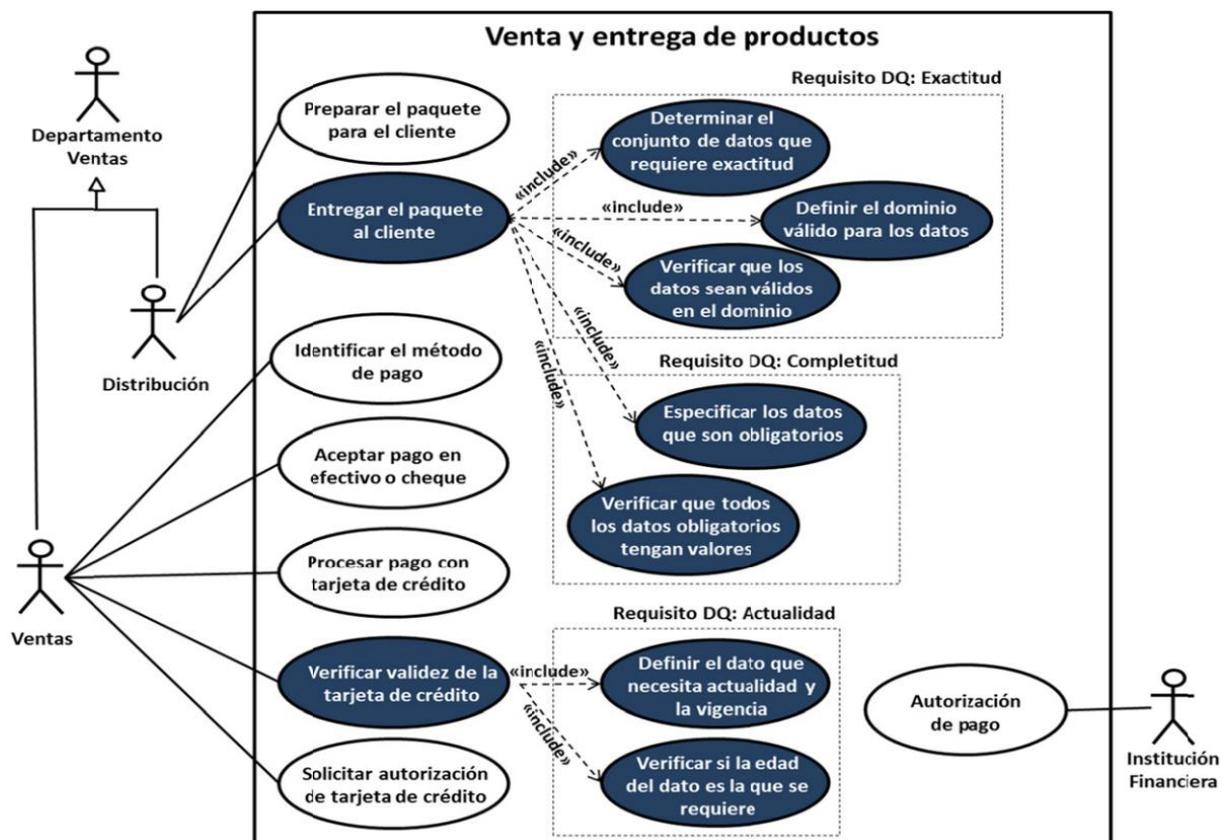


Figura 4.3: Diagrama de Casos de Uso resultantes da aplicação do método (retirado de [27]).

Os Casos de Uso em cor branca representam os Casos de Uso gerais do sistema. Os Casos de Uso em cor azul representam os Casos de Uso associados aos requisitos de qualidade dos dados que foram levantados.

## 4.2 Transformações QVT

Buscando obter Casos de Uso a partir de Processos de Negócio descritos com BPMN e que consideram requisitos de segurança, o trabalho realizado por [28], apresenta um conjunto de regras com QVT (Query/View/Transformation) [29] e uma lista de verificação que permitem realizar esta derivação. O autor considera os seguintes requisitos de segurança: detecção de dano de ataque (*Attack Harm Detection*); controle de acesso (*Acess Controll*); registro de auditoria (*Audit Register*); integridade (*Integrity*); privacidade (*Privacy*); e não repúdio (*Non Repudiation*). O autor utiliza uma versão estendida do BPMN que adiciona marcadores para

os requisitos de segurança.

O processo de derivação consiste em duas etapas:

1. **Aplicação das regras QVT:** As regras QVT tem por objetivo transformar o Diagrama de Processo de Negócio e os elementos adicionais relacionados à Segurança do Processo de Negócio em dois Diagramas de Casos de Uso.
  - a. **Transformação BPD para Diagrama de Casos de Uso:** Segundo as regras definidas em QVT, o autor propõe que todas as *Pools*, *Lanes* e Grupos do BPD serão mapeados para um ator no Diagrama de Casos de Uso, e todas as Atividades do BPD serão mapeadas para Casos de Uso no Diagrama de Casos de Uso e associadas ao ator correspondente a *Pool*, *Lane* ou Grupo a qual a Atividade pertence no BPD.
  - b. **Transformação de Processos de Negócio Seguros para Diagrama de Casos de Uso:** Segundo as regras definidas em QVT, o autor propõe que cada marcador de requisito de segurança do BPD será mapeado para um assunto. Este assunto depois será detalhado e agregará Casos de Uso específicos para atender ao requisito de segurança.
2. **Aplicação das regras de refinamento e listas de verificação:** As regras de refinamento têm por objetivo complementar as regras QVT. Para cada assunto obtido, deve-se agregar Casos de Uso específicos definidos pelo autor no diagrama de Casos de Uso. Se o assunto pertencer a uma *Pool* ou *Lane* o Caso de Uso vai ser associado ao ator correspondente, caso contrário será associado a um ator genérico chamado “Security Staff”.

Ao final deste processo será gerado dois Diagramas de Casos de Uso: o primeiro irá conter um conjunto de Casos de Uso gerais do sistema; e o segundo irá conter um conjunto de Casos de Uso específicos para a segurança do processo de negócio. A Figura 4.4 apresenta um processo de negócio que trata da admissão de pacientes em uma clínica médica. Neste modelo, os elementos que apresentam um “cadeado” são os que devem ser considerados requisitos de segurança.

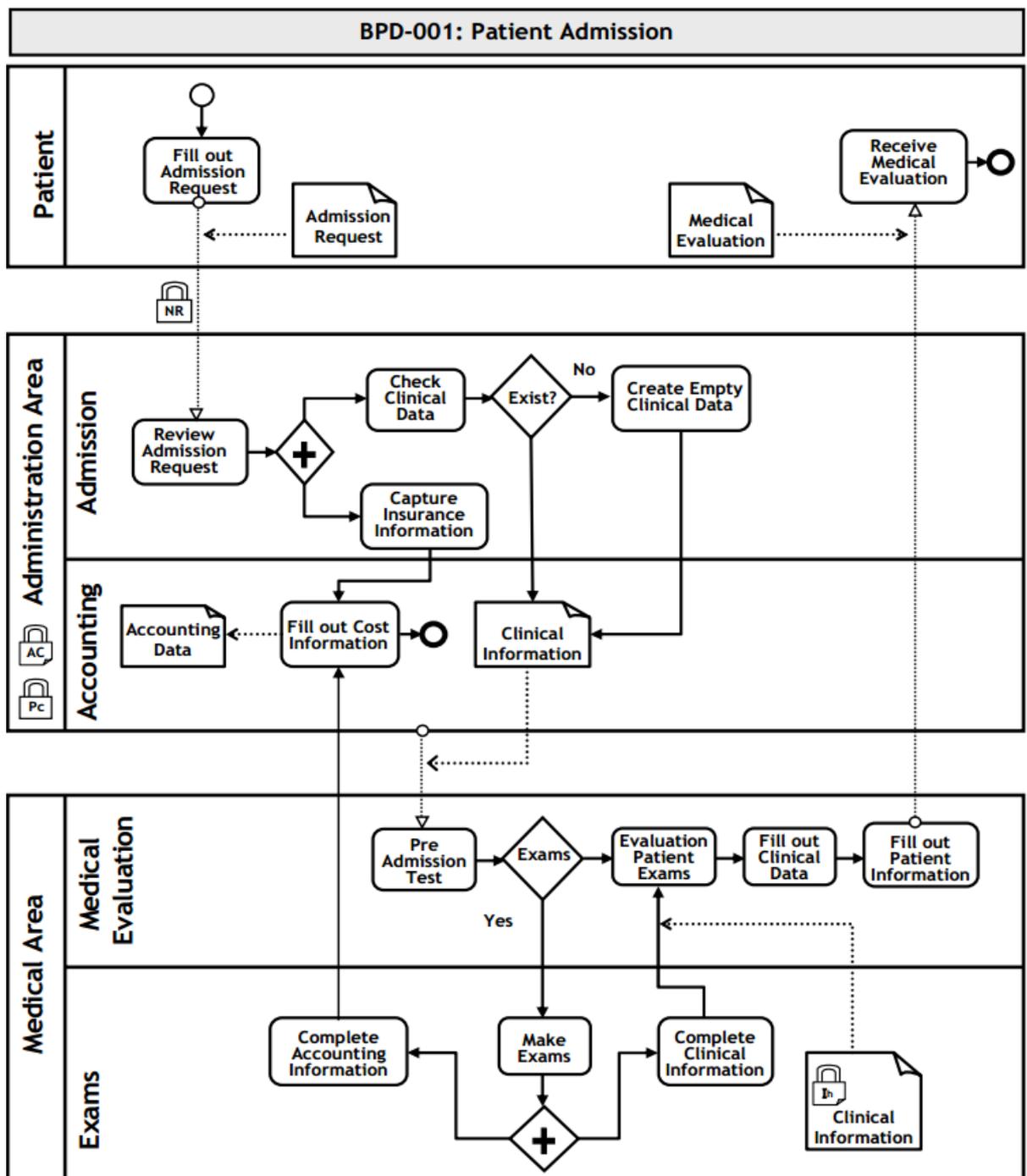


Figura 4.4: Processo de negócio de admissão médica em uma clínica (retirado de [28]).

Neste exemplo, a Pool “Administration Area”, o Data Object “Clinical Information” e a Message Flow entre as Task “Fill out Admission Request” e “Review Admission Request” necessitam de requisitos para garantir a segurança do processo de negócio em questão. A partir deste processo de negócio, aplicando as regras QVT propostas por Rodriguez [28],

obtemos os diagramas de Casos de Uso mostrados na Figura 4.5 e 4.6.

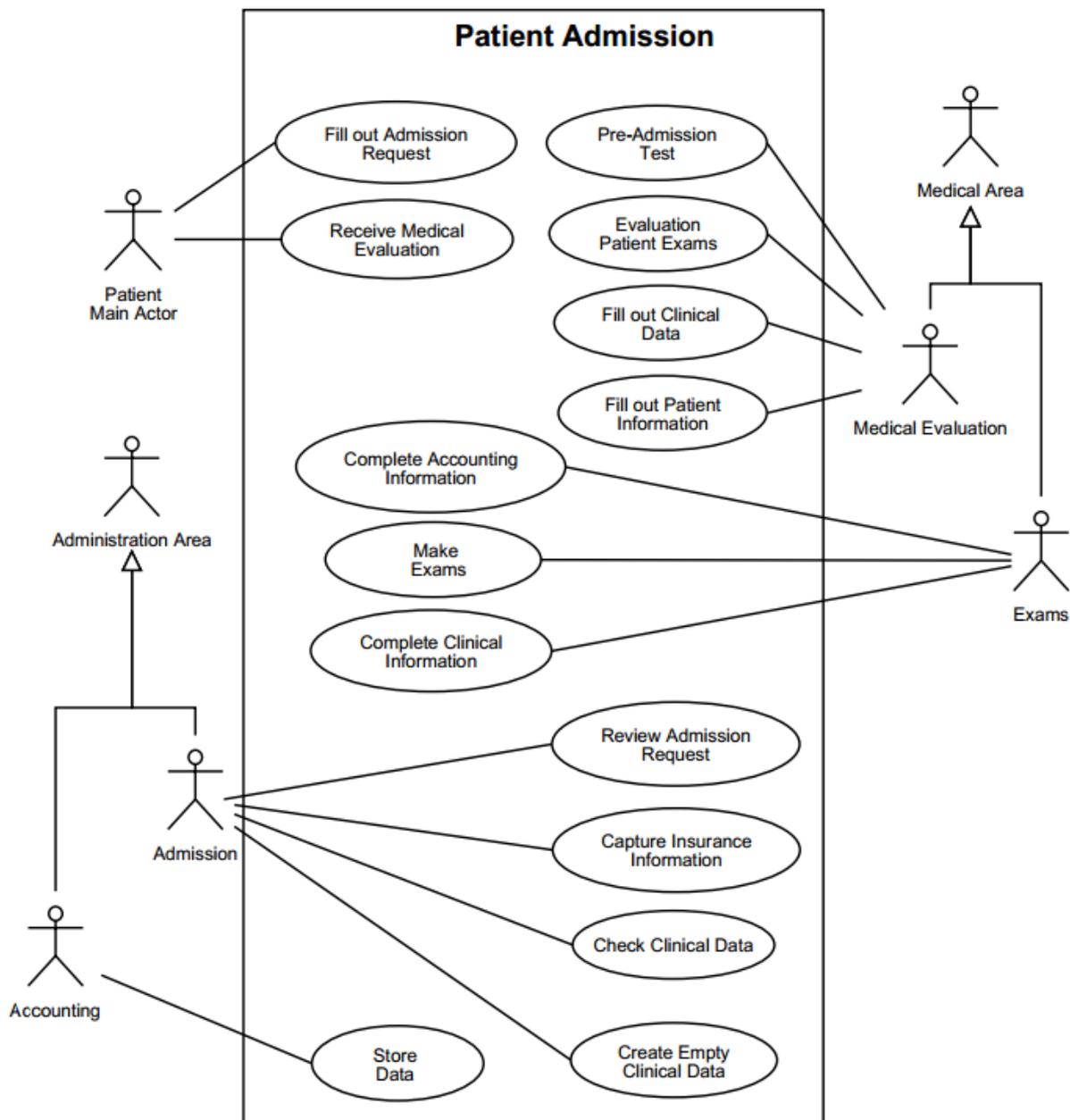


Figura 4.5: Casos de Uso gerais do processo de negócio de admissão médica (retirado de [28]).

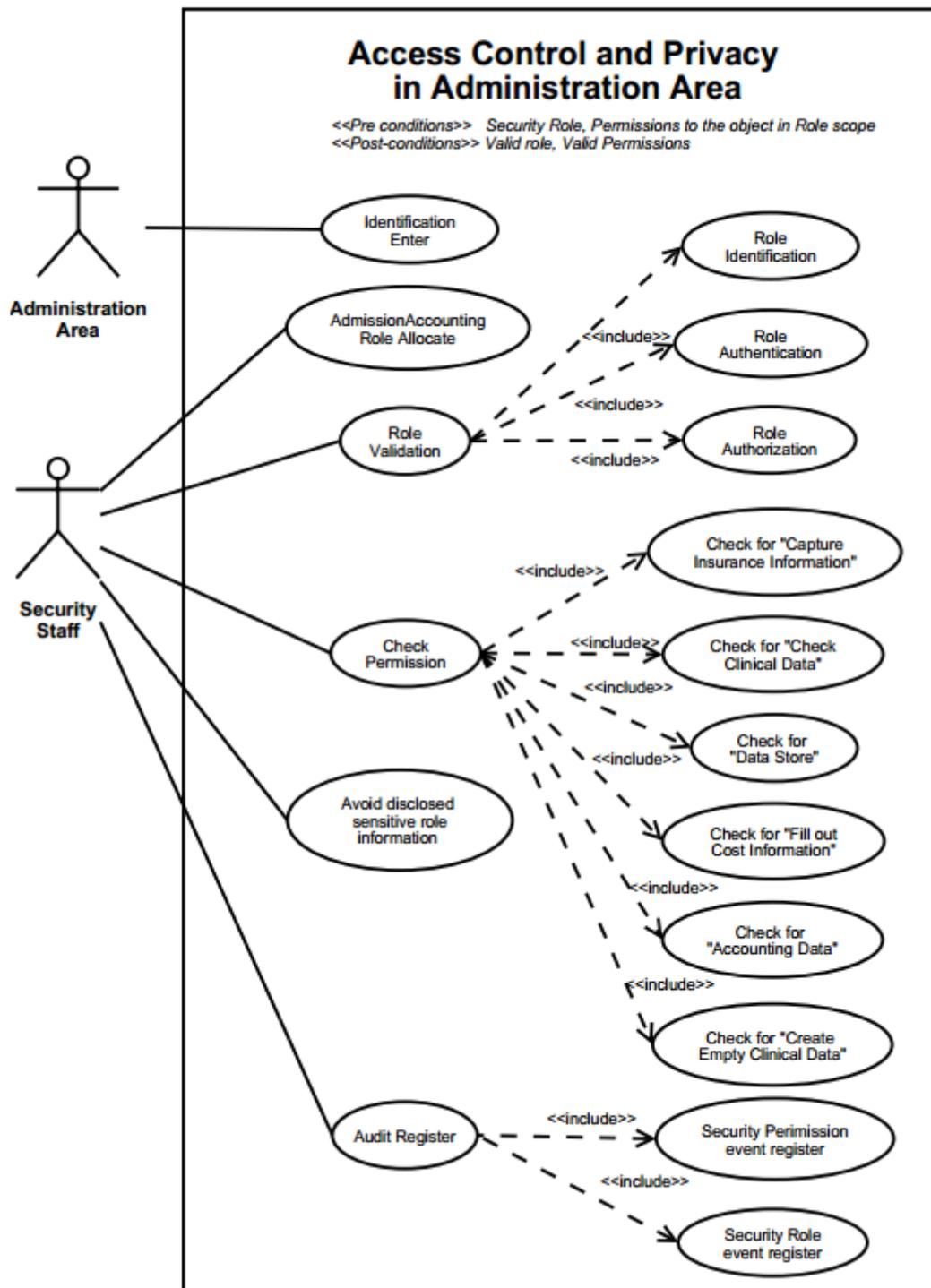


Figura 4.6: Casos de Uso específicos para tratar da segurança do processo de negócio de admissão médica (retirado de [28]).

Os Casos de Uso apresentados na Figura 4.5 são os Casos de Uso gerais do processo de negócio enquanto os Casos de Uso mostrados na Figura 4.6 tratam de questões de segurança do processo de negócio.

## 4.3 Considerações Finais do Capítulo

Cabe destacar que tanto a técnica BPiDQ\* quanto o QVT utilizam uma extensão do BPMN para derivar Casos de Uso focados em aspectos específicos, como qualidade dos dados ou requisitos de segurança de processos de negócio. O estudo destas técnicas é importante, pois certos aspectos serão utilizados nas diretrizes propostas neste trabalho, como o mapeamento das *Pools* e *Lanes* do BPD para atores no Diagrama de Casos de Uso e o mapeamento das Atividades do BPD para Casos de Uso no Diagrama de Casos de Uso. Contudo, diferente dos trabalhos apresentados neste capítulo, em nosso trabalho não focaremos em aspectos específicos como qualidade dos dados ou segurança dos processos de negócio, nossa preocupação será apenas gerar Casos de Uso gerais, mas com mais detalhes, da aplicação a partir da notação “pura” do BPMN.

## Capítulo 5

# Diretrizes para a derivação de Casos de Uso a partir de modelos BPMN

Em capítulos anteriores foram apresentados os fundamentos teóricos, a motivação de pesquisas na área de modelagem organizacional e a importância da integração entre a Modelagem Organizacional e Casos de Uso UML. No capítulo 4 foram apresentadas algumas propostas para derivação de Casos de Uso a partir de Processos de Negócio baseadas em extensões da notação BPMN. Neste capítulo será apresentada a proposta das diretrizes para a derivação de Casos de Uso a partir de modelos descritos apenas com o BPMN.

Diferente das outras propostas, neste trabalho buscamos realizar a derivação de Casos de Uso gerais de um processo de negócio utilizando apenas a notação BPMN “pura”. Desta maneira, não serão adicionados novos elementos a notação original, permitindo que os modelos sejam criados em qualquer ferramenta com suporte a notação. Devido a isso, empresas e organizações que já possuem processos de negócio modelados em BPMN não necessitam refazer seus modelos para poder derivar Casos de Uso.

Inicialmente, na seção 5.1, serão apresentadas as diretrizes propostas. Em seguida, na seção 5.2, será realizado um exemplo de uso aplicando as diretrizes. Na seção 5.3 será realizada uma breve comparação entre o BP2UC e as duas propostas descritas no Capítulo 4. Por fim, na seção 5.4, serão realizadas as considerações finais do capítulo.

### 5.1 BP2UC

O BP2UC utiliza os diagramas simplificados de Processos de Negócio (BPD) da notação BPMN (ver seção 2.2.3 do Capítulo 2) para modelar os processos de negócio da organização para a qual deseja-se obter Casos de Uso UML para um software pretendido. Desta forma,

utiliza-se o modelo BPD do BPMN para a etapa de Modelagem Organizacional e os Casos de Uso UML para a etapa de Requisitos, Análise e Design.

*Pools* e *Lanes* do modelo BPD serão mapeadas diretamente para atores em Casos de Uso. O processo de busca por Casos de Uso consiste em analisar todas as *Activities* e suas *Message Flows*. Para auxiliar na compreensão das diretrizes, considere o processo de negócio apresentado na Figura 5.1 relacionado ao processo de pedido, preparação e entrega de pizzas. Considere também que a Figura 5.2 é a parte interna do Sub Process “Finalizar pedido” existente na Figura 5.1.

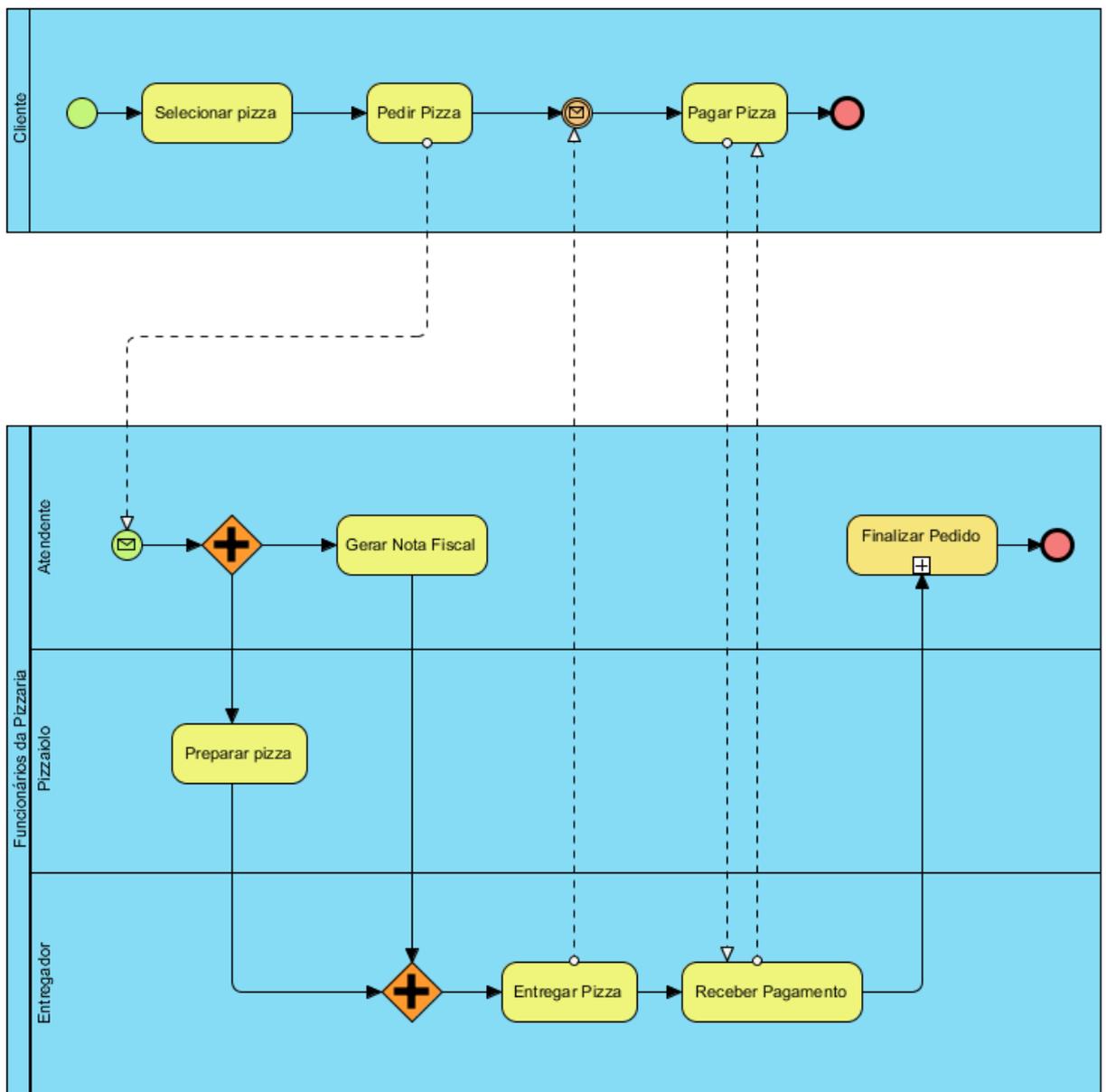


Figura 5.1: Processo de negócio relacionado a entrega de pizza.

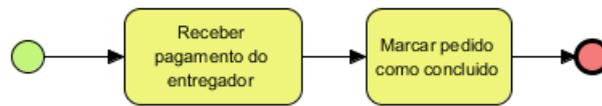


Figura 5.2: Parte inteira do Sub Process “Finalizar pedido”.

Este processo de negócio começa no Start Event da Pool “Cliente”, onde ele seleciona uma pizza (Task “Selecionar pizza”), depois faz o pedido da mesma (Task “Pedir pizza”), que dispara uma Message para a Lane “Atendente”, e aguarda pelo recebimento da pizza. A Message disparada pelo “Cliente” ativa o Start Event do “Atendente”, onde o fluxo de execução se divide, ficando a cargo do “Atendente” gerar a nota fiscal do pedido (Task “Gerar nota fiscal”) enquanto a Lane “Pizzaiolo” fica responsável por preparar a pizza (Task “Preparar pizza”). Após essas duas Task terem sido concluídas, ambos os fluxos de execução se unem para que a Lane “Entregador” realize a entrega da pizza (Task “Entregar pizza”). Esta Task dispara uma Message que ativa o Intermediate Event do “Cliente”. Após receber a pizza (Intermediate Event “Pizza recebida”), o cliente realiza o pagamento (Task “Pagar pizza”). Ao pagar pela pizza, a Task dispara uma mensagem para o “Entregador” com o dinheiro e ele responde recebendo o pagamento (Task “Receber pagamento”) e enviando a nota fiscal ao “Cliente”. Após receber a nota fiscal a participação do “Cliente” termina enquanto o fluxo de execução passa do “Entregador” para o “Atendente” que finaliza o pedido (Sub Process “Finalizar pedido”).

A seguir são listadas as diretrizes para derivação de Casos de Uso UML a partir de Diagramas de Processos de Negócio baseados em BPMN. A Figura 5.3 apresenta um diagrama de atividade que o processo de derivação dos Casos de Uso segundo o BP2UC.

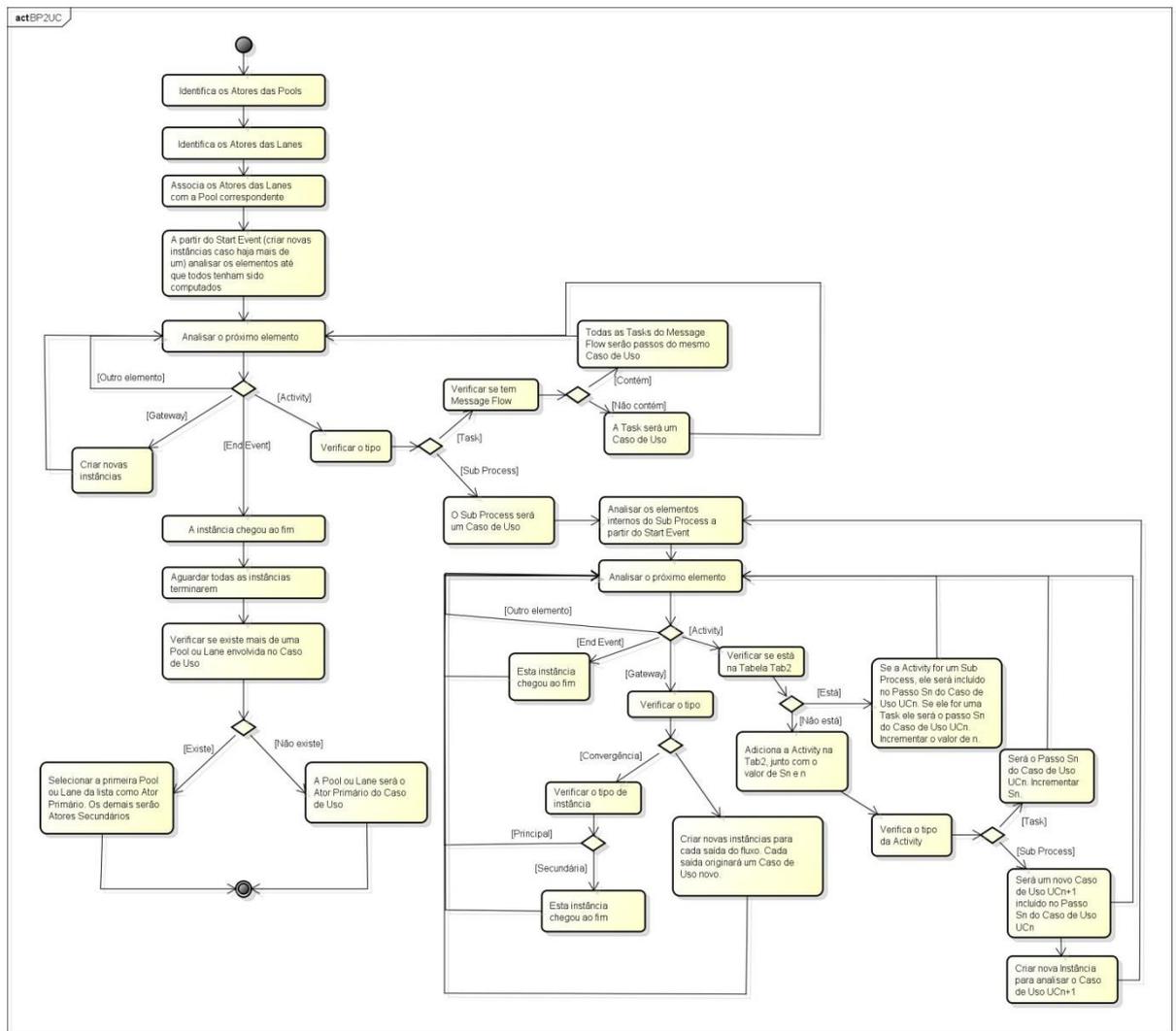


Figura 5.3: Diagrama de atividades do BP2UC.

### 1º Passo: Descobrimo Atores

**Diretriz 1:** Cada Pool no BPMN deve ser considerado um ator nos Casos de Uso. O nome do ator será o mesmo da Pool. Uma Pool representa um participante em um Processo de Negócio. Por exemplo, na Figura 5.1 a Pool “Cliente” originará um Ator chamado **Cliente**.

**Diretriz 2:** Cada Lane no BPMN deve ser considerado um ator nos Casos de Uso. O nome de cada ator será o mesmo da Lane. O Ator originado da Lane será uma generalização do Ator originado do Pool em que está inserido. A Lane no geral é utilizada para organizar e categorizar as Atividades. Por exemplo, na Figura 5.1 a Pool “Funcionários da Pizzaria” originará um Ator chamado **Funcionários da Pizzaria** e as Lanes “Atendente”, “Pizzaiolo” e “Entregador” originarão três atores (**Atendente**, **Pizzaiolo** e **Entregador**) que serão uma

generalização do Ator Funcionários da Pizzaria.

### **2º Passo: Descobrendo Casos de Uso.**

**Diretriz 3:** O processo de descoberta de casos de uso começa observando o primeiro Start Event descrito no BPMN bem os como demais elementos que seguem este Event. Se houverem mais Start Events, criar várias instâncias ( $I_1, I_2, \dots, I_n$ ) e repetir o processo para cada Start Event existente. A análise destes Start Events deve seguir a ordem em que os mesmos aparecem de cima para baixo no modelo BPMN. Observando a Figura 5.1, o processo de análise começaria no Start Event da Pool “Cliente” e outra instância de análise deveria ser criada para analisar o Start Event da Pool “Atendente”.

**Diretriz 4:** Analisar os elementos do modelo BPMN em busca de Casos de Uso segundo as diretrizes abaixo. Cada Caso de Uso identificado deve ser colocado em uma tabela  $Tab_1$  para auxiliar o próximo passo (3º) do processo. Um *template* para esta tabela é apresentado na Tabela 5.1.

**Diretriz 4.1:** Se o elemento que está sendo analisado for uma Task ( $T_1$ ), ela será associada a um único Caso de Uso. Deve-se escolher uma das ações, respeitando a ordem cima-baixo, apresentadas a seguir:

**Diretriz 4.1.1:** Se houver um Message Flow entre esta Task ( $T_1$ ) e outras Tasks ( $T_2, T_3, \dots, T_n$ ), todas serão consideradas passos do mesmo Caso de Uso. Inicialmente o nome do Caso de Uso será o nome da Task ( $T_1$ ). Em seguida deve-se escolher uma das ações, respeitando a ordem cima-baixo, apresentadas a seguir:

**Diretriz 4.1.1.1:** Se houver um Message Flow chegando e saindo entre a Task ( $T_1$ ) e uma Task ( $T_2$ ), a Task ( $T_1$ ) será um passo do Caso de Uso. Deve-se criar uma instância de análise para cada saída do Message Flow da Task ( $T_1$ ) e repetir a Diretriz 4.1.1 em cada instância. A instância de análise da Task ( $T_2$ ) será considerada parte do Cenário Principal do Caso de Uso enquanto as outras instâncias serão consideradas Extensões para aquele ponto. Ao final, deve-se adicionar o Caso de Uso descoberto, as Pool/Lanes envolvidas e a Diretriz 4.1.1.1 na  $Tab_1$ .

**Diretriz 4.1.1.2:** Se houver um Message Flow chegando na Task ( $T_1$ ), ela será avaliada em outra etapa do processo. A análise então deve continuar no próximo elemento da Task ( $T_1$ ).

**Diretriz 4.1.1.3:** Se houver um Message Flow saindo da Task ( $T_1$ ), então a Task ( $T_1$ ) será um passo do Caso de Uso. Deve-se criar uma instância de análise para cada saída de Message Flow da Task ( $T_1$ ) e repetir a Diretriz 4.1.1 em cada instância. A primeira instância criada será considerada parte do Cenário Principal do Caso de Uso e as outras instâncias serão consideradas Extensões para aquele ponto do Caso de Uso. O processo terminará quando todas as instâncias chegarem a Tasks que apenas possuem fluxo de mensagem chegando na mesma. Ao final do processo o analista pode escolher um novo nome para o Caso de Uso descoberto e deve-se adicionar o Caso de Uso descoberto, as Pools/Lanes envolvidas e a Diretriz 4.1.1.3 na Tab<sub>1</sub>.

**Diretriz 4.1.2:** Se o próximo elemento for um Intermediate Event, o Caso de Uso foi descoberto. O nome do Caso de Uso deve ser o mesmo nome da Task ( $T_1$ ). Deve-se adicionar o Caso de Uso descoberto, a Pool/Lane envolvida e a Diretriz 4.1.2 na Tab<sub>1</sub>. Por fim, deve-se repetir a Diretriz 4 para o próximo elemento do fluxo. Por exemplo, na Figura 5.1 o próximo elemento da Task “Pedir Pizza” na Pool “Cliente” é um Intermediate Event, por isso a Task originará um Caso de Uso chamado **Pedir Pizza**.

**Diretriz 4.1.3:** Se o próximo elemento for um End Event, o Caso de Uso foi descoberto. O nome do Caso de Uso deve ser o mesmo nome da Task ( $T_1$ ). Deve-se adicionar o Caso de Uso descoberto, a Pool/Lane envolvida e a Diretriz 4.1.3 na Tab<sub>1</sub>. O processo chegou ao fim para esta instância, devendo continuar as restantes, caso existam. Por exemplo, na Figura 5.1 o próximo elemento da Task “Pagar Pizza” na Pool “Cliente” é um End Event e caso não existisse o Message Flow, a Task originaria um Caso de Uso chamado **Pagar Pizza** (como existe o Message Flow entre ela e a Task “Receber Pagamento” na Lane “Entregador”, a diretriz aplicada é a 4.1.1.1).

**Diretriz 4.1.4:** Se o próximo elemento for um Gateway, o Caso de Uso foi descoberto. O nome do Caso de Uso deve ser o mesmo nome da Task ( $T_1$ ). Deve-se adicionar o Caso de Uso descoberto, a Pool/Lane envolvida e a Diretriz 4.1.4 na Tab<sub>1</sub>. Por fim, deve-se repetir a Diretriz 4 para o próximo elemento do fluxo. Por exemplo, na Figura 5.1 o próximo elemento da Task “Preparar Pizza” na Lane “Pizzaiolo” é um Gateway, portanto a Task originará um Caso de Uso chamado **Preparar Pizza**.

**Diretriz 4.1.5:** Se o próximo elemento for uma Task ( $T_2$ ), o Caso de Uso foi descoberto. O nome do Caso de Uso deve ser o mesmo nome da Task ( $T_1$ ). Deve-se

adicionar o Caso de Uso descoberto, a Pool/Lane envolvida e a Diretriz 4.1.5 na Tab<sub>1</sub>. Por fim, deve-se então repetir a Diretriz 4.1 para a próxima Task (T<sub>2</sub>). Por exemplo, na Figura 5.1 o próximo elemento da Task “Entregar Pizza” na Lane “Entregador” é a Task “Receber Pagamento”, portanto a Task “Entregar Pizza” originará um Caso de Uso chamado **Entregar Pizza**.

**Diretriz 4.1.6:** Se o próximo elemento for um Sub Process, o Caso de Uso foi descoberto. O nome do Caso de Uso deve ser o mesmo nome da Task (T<sub>1</sub>). Deve-se adicionar o Caso de Uso descoberto, a Pool/Lane envolvida e a Diretriz 4.1.6 na Tab<sub>1</sub>. Por fim, deve-se então executar a Diretriz 4.2 para o Sub Processo. Por exemplo, na Figura 5.1 o próximo elemento da Task “Receber Pagamento” na Lane “Entregador” é o Sub Process “Finalizar Pedido”, portanto a Task originará um Caso de Uso chamado **Receber Pagamento**.

**Diretriz 4.2:** Se o elemento que está sendo analisado for um Sub Process (SP<sub>1</sub>), ele será um Caso de Uso. Deve-se criar uma variável  $n$ , do tipo inteira e com valor inicial 1, que representará o nome temporário do Caso de Uso (UC <sub>$n$</sub> ). Caso a variável  $n$  já exista, basta incrementar seu valor (exceto nos casos que ele é passado por outra instância, isto é, quando um Sub Process SP <sub>$x$</sub>  contém outro Sub Process SP <sub>$y$</sub> , neste caso SP <sub>$x$</sub>  irá passar o valor do seu  $n + 1$  para o SP <sub>$y$</sub> ). Em seguida, deve-se criar uma nova instância de análise e executar os passos abaixo. Ao final desta instância, um nome para o Caso de Uso (UC <sub>$n$</sub> ) será o nome do Sub Process (SP<sub>1</sub>), devendo adicionar o Caso de Uso descoberto, a Pool/Lane envolvida e a Diretriz 4.2 na Tab<sub>1</sub>, e o processo de análise deve continuar a partir do próximo elemento do fluxo de sequência. Por exemplo, na Figura 5.1 o Sub Process “Finalizar Pedido” será um Caso de Uso temporário chamado UC<sub>1</sub> (neste ponto  $n$  vale 1). Se ele contivesse outro Sub Process SP<sub>2</sub>, o valor de  $n$  dentro de SP<sub>2</sub> seria 2. Por fim, o Caso de Uso UC<sub>1</sub> será renomeado para **Finalizar Pedido**.

**Diretriz 4.2.1:** Deve-se criar uma variável S <sub>$n$</sub> , do tipo inteira e com valor inicial 1, que representará os passos do Caso de Uso (UC <sub>$n$</sub> ). Deve-se criar uma tabela auxiliar Tab<sub>2</sub>, um *template* é mostrado na Tabela 5.2, que fornecerá suporte ao processo. Em seguida, deve-se analisar os elementos internos do Sub Process (SP<sub>1</sub>) segundo os passos abaixo:

**Diretriz 4.2.1.1:** Se o elemento que está sendo analisado for um Start Event, a análise deve continuar a partir do próximo elemento do fluxo de sequência. Por exemplo, esta diretriz é aplicada no Start Event da Figura 5.2.

**Diretriz 4.2.1.2:** Se o elemento que está sendo analisado for um Intermediate Event, a análise deve continuar a partir do próximo elemento do fluxo de sequência.

**Diretriz 4.2.1.3:** Se o elemento que está sendo analisado for um End Event, esta instância de análise chegou ao fim. Por exemplo, esta diretriz é aplicada no End Event da Figura 5.2.

**Diretriz 4.2.1.4:** Se o elemento que está sendo analisado for uma Task (T), verificar as seguintes condições:

**Diretriz 4.2.1.4.1:** Se a Task (T) não está na tabela auxiliar Tab<sub>2</sub>, então ela será o passo S<sub>n</sub> do Caso de Uso (UC<sub>n</sub>). Deve-se colocar a Task (T), e as variáveis S<sub>n</sub> e UC<sub>n</sub> na tabela auxiliar Tab<sub>2</sub>. Em seguida, incrementar o valor de S<sub>n</sub> em 1. Por fim, a análise deve continuar a partir do próximo elemento do fluxo de sequência. Por exemplo, na Figura 5.2, o valor de *n* e S<sub>1</sub> será 1. Inicialmente a Task “Receber pagamento do entregador” não está na Tabela auxiliar Tab<sub>2</sub>, portanto a Task será o passo S<sub>1</sub> do Caso de Uso UC<sub>1</sub>. Em seguida, o valor de S<sub>1</sub> é incrementado (agora passa a ser 2) e a análise continua no próximo elemento do fluxo (Task “Marcar pedido como concluído” da Figura 5.2).

**Diretriz 4.2.1.4.2:** Se a Task (T) está na tabela auxiliar Tab<sub>2</sub>, deve-se resgatar da tabela os valores de S<sub>x</sub> e UC<sub>x</sub> associados a Task (T). O Caso de Uso UC<sub>n</sub> irá incluir o Caso de Uso (UC<sub>x</sub>). Desta forma, o passo S<sub>n</sub> do Caso de Uso (UC<sub>n</sub>) será um redirecionamento para o passo S<sub>x</sub> do Caso de Uso (UC<sub>x</sub>). Por fim, deve-se incrementar o valor de S<sub>n</sub> e a análise deve continuar a partir do próximo elemento do fluxo de sequência da Task (T).

**Diretriz 4.2.1.5:** Se o elemento que está sendo analisado for um Sub Process (SP<sub>2</sub>), verificar as seguintes condições:

**Diretriz 4.2.1.5.1:** Se o Sub Process (SP<sub>2</sub>) não está na tabela auxiliar Tab<sub>2</sub>, então o Caso de Uso UC<sub>n</sub> irá incluir um novo Caso de Uso (UC<sub>n+1</sub>) em seu passo S<sub>n</sub>. Deve-se adicionar o Sub Process (SP<sub>2</sub>) e as variáveis S<sub>n</sub> e UC<sub>n</sub> na tabela auxiliar Tab<sub>2</sub>. Em seguida, deve-se incrementar o valor de S<sub>n</sub>. Por fim, deverá criar uma nova instância, repetindo a Diretriz 4.2, para analisar os elementos internos do Sub Process (SP<sub>2</sub>) que será o Caso de Uso (UC<sub>n+1</sub>). Ao final desta instância, o processo deve continuar a partir do próximo elemento do fluxo de sequência do Sub Process (SP<sub>2</sub>).

**Diretriz 4.2.1.5.2:** Se o Sub Process (SP<sub>2</sub>) está na tabela auxiliar Tab<sub>2</sub>, então deve-se resgatar da tabela os valores de S<sub>x</sub> e UC<sub>x</sub> associados ao Sub Process (SP<sub>2</sub>). O Caso de Uso UC<sub>n</sub> irá incluir o Caso de Uso (UC<sub>x</sub>). Desta forma, o passo S<sub>n</sub> do Caso de Uso (UC<sub>n</sub>) será um redirecionamento para o passo S<sub>x</sub> do Caso de Uso (UC<sub>x</sub>). Por fim, deve-se incrementar o valor de S<sub>n</sub> e a análise deve continuar a partir do próximo elemento do fluxo de sequência do Sub Process (SP<sub>2</sub>).

**Diretriz 4.2.1.6:** Se o elemento que está sendo analisado for um Gateway (G), verificar as seguintes condições:

**Diretriz 4.2.1.6.1:** Se o Gateway (G) for um gateway de divergência (divide o fluxo), então cada saída do fluxo será um novo Caso de Uso (UC<sub>n+1</sub>, UC<sub>n+2</sub>, ... , UC<sub>n+m</sub>). A saída de aceitação (caso não exista uma saída de aceitação deve-se escolher uma das saídas como a saída principal) será o Caso de Uso (UC<sub>n+1</sub>), e o Caso de Uso (UC<sub>n</sub>) irá incluí-lo no passo S<sub>n</sub>. Os demais Casos de Uso (UC<sub>n+2</sub>, ... , UC<sub>n+m</sub>) serão extensões do passo S<sub>n</sub>. Em seguida, deve-se criar uma instância de análise para cada Caso de Uso (UC<sub>n+1</sub>, UC<sub>n+2</sub>, ... , UC<sub>n+m</sub>) da saída e repetir a Diretriz 4.2 em cada uma delas. Ao final de cada instância o analista deve escolher um novo nome para os Casos de Uso (UC<sub>n+1</sub>, UC<sub>n+2</sub>, ... , UC<sub>n+m</sub>). Por fim, deve-se incrementar o valor de S<sub>n</sub> e continuar a análise a partir do próximo elemento em cada saída do fluxo (criar m instâncias e repetir a Diretriz 4 em cada uma delas).

**Diretriz 4.2.1.6.2:** Se o Gateway (G) for um gateway de convergência (une o fluxo), e esta instância for a principal a análise deve continuar no próximo elemento do fluxo, caso contrário a instância chegou ao fim.

**Diretriz 4.3:** Se o elemento que está sendo analisado for um Start Event, repetir a Diretriz 4 para o próximo elemento do fluxo de sequência. Por exemplo, na Figura 5.1 quando o Start Event na Pool “Cliente” é analisado esta diretriz é aplicada.

**Diretriz 4.4:** Se o elemento que está sendo analisado for um Intermediate Event, repetir a Diretriz 4 para o próximo elemento do fluxo de sequência. Por exemplo, na Figura 5.1 quando o Intermediate Event na Pool “Cliente” é analisado esta diretriz é aplicada.

**Diretriz 4.5:** Se o elemento que está sendo analisado for um End Event, o processo terminou para esta instância. Por exemplo, na Figura 5.1 quando o End Event na Pool

“Cliente” é analisado esta diretriz é aplicada.

**Diretriz 4.6:** Se o elemento que está sendo analisado for um Gateway, criar várias instâncias e repetir a Diretriz 4 para o próximo elemento de todas as ramificações do fluxo de sequência do Gateway. Por exemplo, na Figura 5.1, quando o Gateway na Lane “Atendente” é analisado, duas instâncias devem ser criadas: a instância I<sub>1</sub> tratará do caminho que vai em direção a Task “Gerar nota fiscal” e a instância I<sub>2</sub> tratará do caminho que vai em direção a Task “Preparar Pizza”.

**Diretriz 4.7:** Uma Activity só pode estar associada a um único Caso de Uso. Por exemplo, na Figura 5.1, a Task “Receber pagamento” faz parte do Caso de Uso **Pagar Pizza**, desta forma a Task não poderá ser ou fazer parte de outro Caso de Uso.

### **3º Passo: Associando Casos de Uso aos Atores.**

**Diretriz 5:** Para cada Caso de Uso obtido no Passo 2, avaliar os participantes (Pools e Lanes do BPMN) envolvidos segundo as diretrizes abaixo:

**Diretriz 5.1:** Se o Caso de Uso foi obtido na Diretriz 4.1.1, ele será associado ao Ator que representa a Pool ou Lane que contém a Task associada ao primeiro passo do Caso de Uso. Por exemplo, na Figura 5.1 o Caso de Uso **Pagar Pizza** será associado ao ator Cliente.

**Diretriz 5.2:** Se o Caso de Uso foi obtido em qualquer outra Diretriz, ele será associado ao Ator que representa a Pool ou Lane na qual a Task ou Sub Process está inserida. Por exemplo, na Figura 5.1 o Caso de Uso **Gerar nota fiscal** será associado ao ator Atendente.

Tabela 5.1: template para a tabela de auxílio no 3º passo das diretrizes.

Nome do Caso de Uso	Pool/Lanes envolvidas	Diretriz Aplicada
Caso de Uso 1	Pool 1; Pool 2; Lane 2	D 4.1.1
Caso de Uso 2	Lane 2	D 4.2.1.4.2

Tabela 5. 2: template para a tabela auxiliar da Diretriz 4.2.1.

Nome da Activity (Task ou Sub Process)	Passo (S <sub>n</sub> ) do Caso de Uso	Caso de Uso (UC <sub>n</sub> ) associado
Task 1	1	UC <sub>1</sub>
Task 2	2	UC <sub>1</sub>
Sub Process 1	1	UC <sub>2</sub>

## **5.2 Exemplo de Uso**

O processo de derivação começa identificando os atores dos Casos de Uso (Passo 1). Desta maneira, pela Diretriz 1, a *Pool* “Cliente” originará um ator **Cliente**. Em seguida, através da Diretriz 2, as *Lanes* “Atendente”, “Pizzaiolo” e “Entregador” serão mapeados para os atores **Atendente**, **Pizzaiolo** e **Entregador**, respectivamente. Por fim, ainda na Diretriz 2, a *Pool* “Funcionários da Pizzaria” originará um ator, chamado de **Funcionários da Pizzaria**, e os atores Atendente, Pizzaiolo e Entregador serão generalizados a partir dele.

Em seguida, começa-se a descobrir os Casos de Uso (Passo 2). Segundo a Diretriz 3, existem duas instâncias iniciais de execução deste passo. A primeira instância começa no Start Event da *Pool* “Cliente” e será chamada de  $I_1$ . A segunda instância começa no Evento Inicial da *Lane* “Atendente” e será chamada de  $I_2$ .

Resolvendo a instância  $I_1$  primeiro temos que o elemento que está sendo analisado atualmente é o Start Evento da *Pool* “Cliente”, portanto a análise deve continuar no próximo elemento do fluxo (Diretriz 4.3).

O próximo elemento do fluxo é a Task “Selecionar Pizza”. O próximo elemento do fluxo após a Task “Selecionar pizza” é outra Task, portanto a Task “Selecionar pizza” será mapeada como um Caso de Uso chamado **Selecionar pizza** (Diretriz 4.1.5), devemos adicionar as informações do Caso de Uso na tabela de auxílio do 3º Passo e o processo de análise continua na Task “Pedir pizza”.

O próximo elemento da Task “Pedir pizza” é um Intermediate Event, portanto a Task “Pedir pizza” será mapeada como um Caso de Uso chamado **Pedir pizza** (Diretriz 4.1.2), adicionamos as informações do Caso de Uso na tabela e o processo de análise continua no Intermediate Event.

Pela Diretriz 4.4 a análise deve continuar no próximo elemento do Intermediate Event, portanto a Task “Pagar pizza” deverá ser analisada. Na análise da Task “Pagar Pizza”, duas diretrizes podem ser aplicadas: a Diretriz 4.1.1 ou a Diretriz 4.1.3. Deve-se respeitar a ordem delas, portanto será aplicada a Diretriz 4.1.1. A Task “Pagar pizza” possui um fluxo de mensagem com a Task “Receber pagamento” da *Lane* “Entregador”, portanto “Receber Pagamento” será o próximo passo do Caso de Uso **Pagar Pizza** (Diretriz 4.1.1.1). Ainda pela Diretriz 4.1.1.1, devemos criar uma instância  $I_3$  para repetir a Diretriz 4.1.1 na Task “Receber pagamento”.

Como a Task “Receber pagamento” não possui mais nenhum fluxo de mensagem, a instância  $I_3$  chegou ao fim, o Caso de Uso foi totalmente descoberto e devemos adicionar as

informações dele na tabela auxiliar. Voltando a análise da instância  $I_1$ , o processo deverá continuar no próximo elemento da Task “Pagar pizza” que, neste caso, é o End Event, por isso encerra-se esta instância de execução (Diretriz 4.1.3).

Ao concluir a instância  $I_1$ , agora podemos resolver a instância  $I_2$ . Da mesma maneira que na instância  $I_1$ , o processo nesta instância inicia-se pelo Start Event na *Pool* “Atendente”, portanto a análise deve continuar para o próximo elemento do fluxo (Diretriz 4.3). O próximo elemento do fluxo é um Gateway e, de acordo com a Diretriz 4.6, devemos continuar a análise em todas as saídas do Gateway, que serão nomeadas de  $G_1$  (*Lane* “Atendente”) e  $G_2$  (*Lane* “Pizzaiolo”).

Vamos começar executando a análise na instância  $G_1$ , sendo que devemos avaliar a Task “Gerar nota fiscal”. O próximo elemento do fluxo após a Task “Gerar nota fiscal” é um Gateway, devido a isso a Task será mapeada como um Caso de Uso chamado **Gerar nota fiscal** (Diretriz 4.1.4), adicionamos as informações do Caso de Uso na tabela e o processo de análise continua no Gateway. Como o Gateway só possui uma saída, não se faz necessário criar outras instâncias de análise.

Segundo a Diretriz 4.6 o processo de análise deve continuar no próximo elemento após o Gateway, portanto devemos analisar a Task “Entregar pizza”. O próximo elemento da Task “Entregar pizza” é outra Task, portanto, segundo a Diretriz 4.1.5 a Task “Entregar pizza” será mapeada como um Caso de Uso chamado **Entregar pizza**, adicionamos as informações do Caso de Uso na tabela e o processo de análise deve continuar na Task “Receber pagamento”. A Tarefa “Receber pagamento” já foi associada a um Caso de Uso (**Pagar Pizza**), portanto ela não deve ser associada a outro Caso de Uso (Diretriz 4.7), eliminando a necessidade de análise da mesma.

O processo deve então continuar no próximo elemento do fluxo que é o Sub Process “Finalizar pedido”. Segundo a Diretriz 4.2, o Sub Process “Finalizar Pedido” será um Caso de Uso. Devemos criar uma variável  $n$  com valor 1 para representar o nome temporário do Caso de Uso  $UC_1$  (que será associado ao Sub Process “Finalizar Pedido” no final do processo).

Em seguida, criamos uma nova instância de análise  $I_3$  para analisar os elementos internos do Sub Process. Pela Diretriz 4.2.1 devemos criar uma variável  $S_1$  com valor inicial 1 que representará os passos de  $UC_1$  e outra Tabela auxiliar para dar suporte a análise dos elementos internos do Sub Process. O processo de análise começa no Start Event e, pela Diretriz 4.2.1.1 devemos continuar a análise no próximo elemento do fluxo.

O próximo elemento do fluxo é a Task “Receber pagamento do entregador”. Pela Diretriz 4.2.1.4.1, a Task não está na nossa Tabela auxiliar, portanto ela será o passo  $S_1$  (atualmente  $S_1$  vale 1) do nosso  $UC_1$ . Colocamos a Task,  $S_1$  e  $UC_1$  na Tabela auxiliar, incrementamos o valor de  $S_1$  (agora seu valor passa a ser 2) e continuamos a análise no próximo elemento do fluxo.

O próximo elemento do fluxo é a Task “Marcar pedido como concluído”, portanto, como ela não está em nossa tabela auxiliar, pela Diretriz 4.2.1.4.1, ela será o passo  $S_1$  (que atualmente vale 2) do nosso  $UC_1$ . Colocamos a Task,  $S_1$  e  $UC_1$  na Tabela auxiliar, incrementamos o valor de  $S_1$  (agora seu valor passa a ser 3) e continuamos a análise no próximo elemento do fluxo.

O próximo elemento do fluxo é um End Event e, pela Diretriz 4.2.1.3 a nossa instância  $I_3$  chegou ao fim. Com isso, o nosso Caso de Uso  $UC_1$  será renomeado para **Finalizar Pedido** pela Diretriz 4.2, devemos agora adicionar as informações do Caso de Uso na tabela auxiliar (a tabela de auxílio do 3º passo e não na tabela de auxílio da Diretriz 4.2). Em seguida, devemos continuar a execução da instância  $G_1$ . O próximo elemento após o Sub Process é um End Event, portanto, pela Diretriz 4.5 esta instância chegou ao fim.

Continuando a análise da instância  $G_2$ , temos que o próximo elemento após o Gateway é a Task “Preparar Pizza”. Pela Diretriz 4.1.4 ela será o Caso de Uso **Preparar Pizza**, devemos adicionar na tabela auxiliar as informações do Caso de Uso e o processo de análise deve continuar no próximo elemento do fluxo. Como todos os próximos elementos do fluxo já foram analisados (o Gateway, a Task “Entregar Pizza”, a Task “Receber Pagamento”, o Sub Process “Finalizar Pedido” e o End Event) esta instância chegou ao fim. Como a instância  $G_1$  e  $G_2$  terminaram, segundo Diretriz 4.6 a instância  $I_2$  também chegou ao fim, pois foi à instância  $I_2$  que criou as instâncias  $G_1$  e  $G_2$ . Como todas as instâncias chegaram ao final o processo de análise está concluído. A Tabela 5.3 mostra os Casos de Uso descobertos de acordo com o *template* proposto. A Tabela 5.4 mostra a tabela auxiliar da Diretriz 4.2.1.

Tabela 5.3: tabela auxiliar que contém informações relacionadas aos Casos de Uso obtidos.

Nome do Caso de Uso	Pool/Lanes envolvidas	Diretriz Aplicada
Selecionar Pizza	Cliente	D 4.1.5
Pedir Pizza	Cliente	D 4.1.2
Pagar Pizza	Cliente, Entregador	D 4.1.1
Gerar Nota Fiscal	Atendente	D 4.1.4
Entregar Pizza	Entregador	D 4.1.5
Finalizar Pedido	Atendente	D 4.2

Preparar Pizza	Pizzaiolo	D 4.1.4
----------------	-----------	---------

Tabela 5.4: tabela auxiliar que contém informações relacionadas aos passos dos Casos de Uso.

Nome da Activity (Task ou Sub Process)	Passo ( $S_n$ ) do Caso de Uso	Caso de Uso ( $UC_n$ ) associado
Receber pagamento do entregador	1	UC1
Marcar pedido como concluído	2	UC1

Por fim, devemos associar os Casos de Uso obtidos no Passo 2 aos atores obtidos no Passo 1. Para isso, vamos utilizar a Diretriz 5 e a tabela auxiliar gerada (Tabela 5.3). O Caso de Uso **Selecionar pizza** será associado ao ator Cliente. O Caso de Uso **Pedir pizza** será associado ao ator Cliente. O Caso de Uso **Pagar Pizza** terá como ator primário o Cliente e como ator secundário o Entregador. O Caso de Uso **Gerar nota fiscal** será associado ao ator Atendente. O Caso de Uso **Finalizar pedido** será associado ao ator Atendente. O Caso de Uso **Entregar pizza** será associado ao ator Entregador. O Caso de Uso **Finalizar pedido** será associado ao ator Atendente. O Caso de Uso **Preparar pizza** será associado ao ator Pizzaiolo.

A Figura 5.3 ilustra o diagrama de Casos de Uso gerado através da aplicação das diretrizes para o exemplo demonstrado.

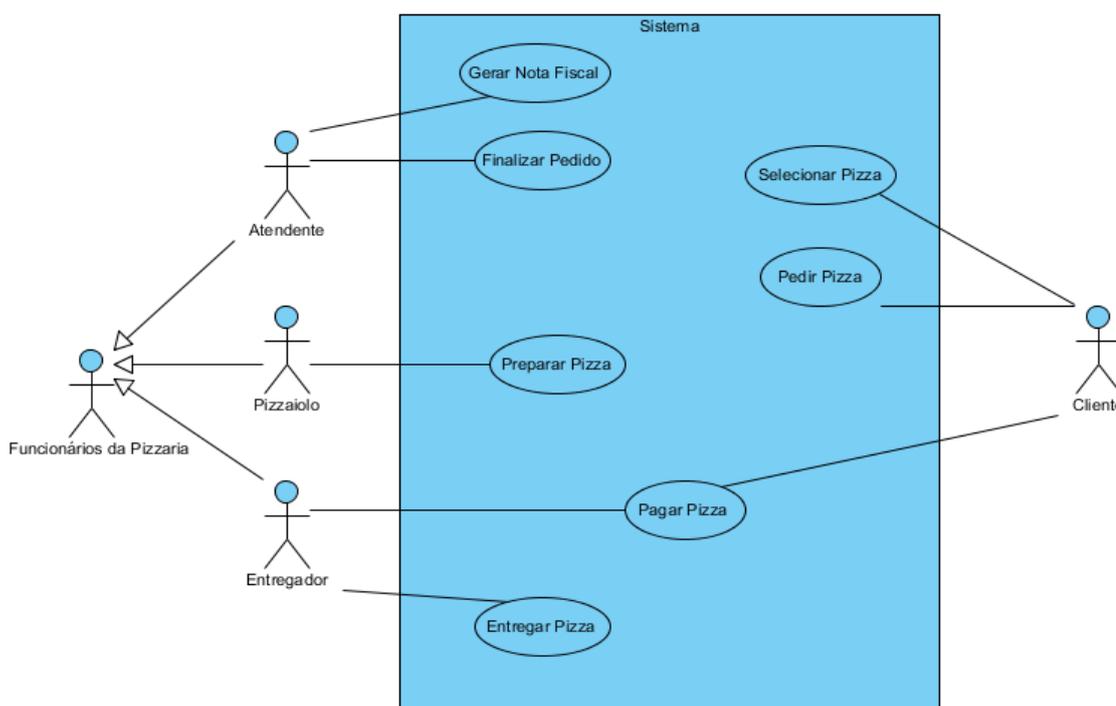


Figura 5.4: Diagrama de Casos de Uso gerado a partir da aplicação das diretrizes.

## 5.3 Diferenciando o BP2UC em relação às técnicas BPiDQ\* e QVT

Algumas características destas técnicas foram reaproveitadas em nossa proposta, contudo muitos aspectos são diferentes. A Tabela 5.5 apresenta uma breve comparação destas diferenças.

Tabela 5.5: Comparação entre as propostas de derivação.

<b>Crítérios</b>	<b>BPiDQ*</b>	<b>QVT</b>	<b>BP2UC</b>
Adiciona elementos na notação BPMN?	SIM	SIM	NÃO
Gera Casos de Uso gerais?	SIM	SIM	SIM
Gera Casos de Uso específicos?	SIM	SIM	NÃO
Gera os cenários dos Casos de Uso?	NÃO	NÃO	SIM
Identifica atores?	SIM	SIM	SIM
Identifica generalização entre os atores?	SIM	SIM	SIM
Identifica inclusão entre Casos de Uso?	SIM	SIM	SIM
Identifica extensão entre Casos de Uso?	NÃO	NÃO	SIM

Cabe destacar que apenas a BP2UC não estende a notação BPMN para gerar Casos de Uso UML e é a única capaz de gerar a descrição textual dos Casos de Uso. Entretanto, apenas o BPiDQ\* e o QVT conseguem capturar aspectos específicos (qualidade de dados e segurança dos processos de negócio) de software.

## 5.4 Considerações Finais do Capítulo

Este capítulo apresentou as diretrizes para derivação de Casos de Uso a partir de modelos baseados na notação BPMN. Tais diretrizes auxiliam engenheiros de software e analistas de sistema a obterem Casos de Uso a partir de modelos baseados em BPMN. Posteriormente, essas diretrizes serão mapeadas para o JGOOSE, possibilitando a derivação dos Casos de Uso de forma automatizada.

# Capítulo 6

## Estudo de Viabilidade

Neste trabalho adotaremos a metodologia empírica proposta por Shull, Carver e Travassos [30] para introduzir processos de software. Contudo, devido a limitação de tempo apenas o Estudo de Viabilidade foi realizado. Inicialmente, na seção 6.1, será apresentada a metodologia. Em seguida, na seção 6.2, será apresentado o Estudo de Viabilidade. Por fim, na seção 6.3, serão realizadas as considerações finais do capítulo.

### 6.1 Metodologia

A metodologia proposta por Shull, Carver e Travassos [30] busca avaliar de forma incremental um novo processo. Para isso, ela apresenta uma série de questões que devem ser abordadas, bem como os tipos de estudos que melhor respondem a essas perguntas. A Figura 6.1 apresenta a visão geral da metodologia.

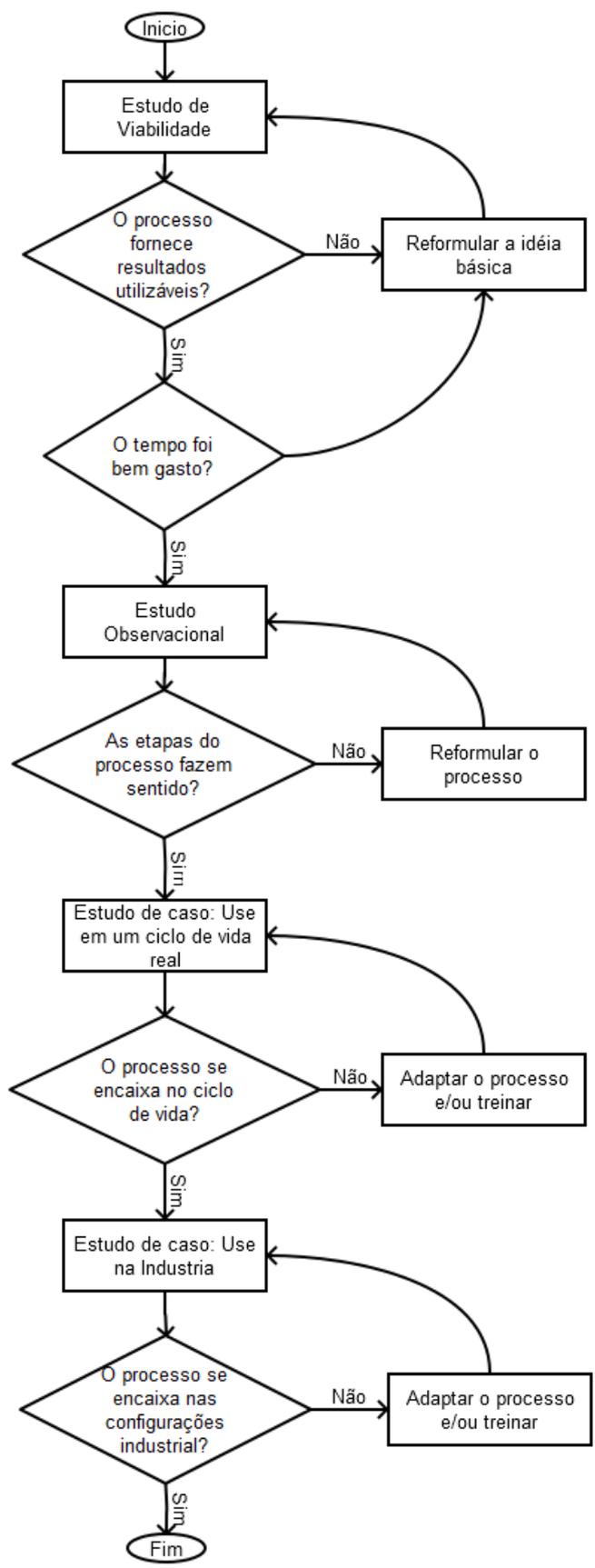


Figura 6.1: Visão geral da metodologia empírica.

No estudo de viabilidade espera-se responder duas perguntas: o processo fornece resultados utilizáveis? E o tempo foi bem gasto? A primeira pergunta busca avaliar se o processo cumpriu a meta global para o qual ele foi criado. Neste ponto, avalia-se se vale a pena gastar os recursos necessários para continuar através da metodologia. A segunda pergunta busca avaliar se os resultados obtidos poderiam ter sido alcançados de forma mais eficiente em termos de custo.

Em seguida, no estudo observacional busca-se responder a pergunta: as etapas do processo fazem sentido? Essa pergunta avalia se cada uma das etapas do novo processo é eficaz e a ordem na qual elas são executadas faz sentido. Nesta fase as etapas do processo podem sofrer modificações.

A próxima etapa consiste em realizar um estudo de caso para verificar se o processo se encaixa em um ciclo de desenvolvimento real. O objetivo do uso de um estudo de caso neste momento no desenvolvimento da nova tecnologia é fazer algum ajuste fino ou adaptação da tecnologia.

Por fim, após adaptar o novo processo ao ciclo de desenvolvimento real, precisa-se utilizar o novo processo no ambiente industrial para investigar se o processo tem quaisquer interações negativas imprevistas com a configuração industrial. Neste ponto outro estudo de caso é realizado.

## 6.2 Estudo de Viabilidade

Para responder a meta global do BP2UC vamos utilizar o Goal/Question/Metric (GQM). O GQM é uma abordagem orientada a metas e utilizada em Engenharia de Software para a medição de produtos e processos de software.

- **Objetivo global:** gerar Casos de Uso UML a partir de modelos de processos de negócio descritos com BPMN.
- **Objetivo do estudo:**

**Analisar** os elementos gráficos e a descrição textual dos Casos de Uso

**Com a finalidade** verificá-los

**Com respeito à** sintaxe dos elementos gráficos segundo os padrões da UML e a coerência e utilidade dos Casos de Uso.

**Do ponto de vista dos** analistas e engenheiros de requisitos

No contexto da geração de Casos de Uso.

- **Questões (Q)/Métricas (M):**

**Q01:** Os Casos de Uso gerados pelo BP2UC estão semanticamente corretos?

**M01:** Casos de Uso são funcionalidades do sistema.

**Q02:** Os atores gerados pelo BP2UC estão semanticamente corretos?

**M02:** Os atores são agentes externos que interagem com o sistema.

**Q03:** Os artefatos gerados são úteis?

**M03:** O diagrama e a descrição textual de Casos de Uso podem ser utilizados em outras etapas do processo de desenvolvimento.

Para responder essas perguntas utilizamos o BP2UC em um processo de negócio que foi cedido pela setor de Produto da empresa de desenvolvimento de software Wealth Systems que representa o ciclo de vida de um novo projeto dentro do setor. O processo de negócio está descrito no Apêndice B. O diagrama resultante da aplicação das diretrizes no processo de negócio é apresentado na Figura 6.2. As descrições dos Casos de Uso obtidos segundo o *template* de Cockburn são apresentadas no Apêndice C.



Figura 6.2: Diagrama de Casos de Uso originado a partir da aplicação das diretrizes no processo de negócio.

Considerando o diagrama apresentado na Figura 6.2 e as descrições dos Casos de Uso apresentadas no Apêndice C, a Tabela 6.1 apresenta os resultados das questões.

Tabela 6.1: Resultados das questões conforme as métricas.

MÉTRICA	Diagrama de Casos de Uso	Descrição dos Casos de Uso
M01	Sim	Sim
M02	Sim	Sim
M03	Sim	Sim

## **6.3 Considerações Finais do Capítulo**

Neste capítulo apresentou-se o Estudo de Viabilidade que compõe a primeira etapa da metodologia para inserir novos processos de software. Verificou-se neste estudo que o BP2UC gera Casos de Uso semanticamente corretos e os artefatos gerados são úteis do ponto de vista dos analistas e engenheiros de requisitos. Contudo, este estudo ainda pode ser complementado através da elaboração de um Quase-Experimento com alunos de graduação e pós-graduação.

# Capítulo 7

## Conclusões

Neste trabalho de conclusão de curso, elaborou-se as diretrizes que permitem a derivação de Casos de Uso UML a partir de processos de negócio modelados com BPMN. Na seção 7.1 são apresentados os resultados desta pesquisa. Em seguida, na seção 7.2 são realizadas as conclusões. Na seção 7.3 são apresentadas as limitações do trabalho. Por fim, na seção 7.4 são propostos algumas sugestões de trabalhos futuros com base nesta pesquisa.

### 7.1 Resultados

A elaboração das diretrizes que permitem a derivação de modelos funcionais a partir de modelos de processos de negócio modelados com BPMN auxilia na elicitação de requisito de organizações que já possuem seus processos de negócios modelados com tal notação. Desta forma, é possível reutilizar artefatos das organizações no processo de Engenharia de Requisitos, mais especificamente na Modelagem Organizacional, visando um ganho de tempo e reaproveitamento de esforço.

### 7.2 Conclusões

Com base nos resultados obtidos, podemos afirmar que as diretrizes auxiliam Engenheiros de Software na elicitação dos requisitos, pois oferecem um meio sistemático de obter Casos de Uso UML a partir de modelos de processos de negócio modelados com BPMN. Desta maneira, um processo que antes ocorria de forma *ad hoc*, agora pode ser realizado de forma precisa e com menos esforço.

## **7.3 Limitações**

Cabe destacar que as diretrizes trabalham apenas sob o conjunto simplificado de elementos do BPMN. Alguns elementos também não são tratados pelas diretrizes (como Datas, Groups e Associations), por questões de tempo.

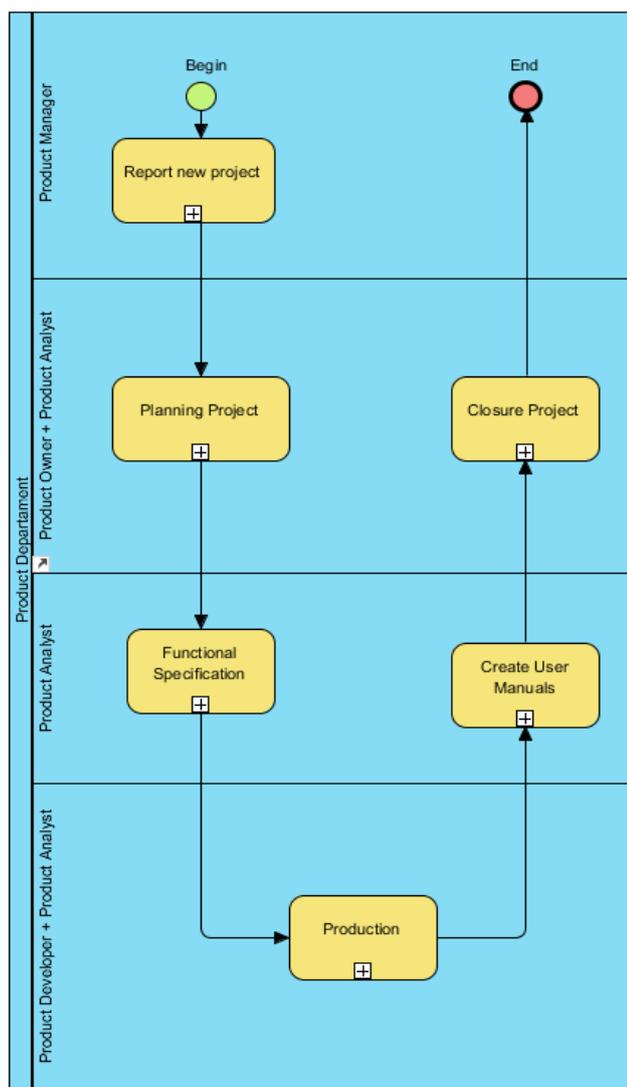
## **7.4 Trabalhos Futuros**

Com base neste trabalho, podemos destacar alguns trabalhos futuros:

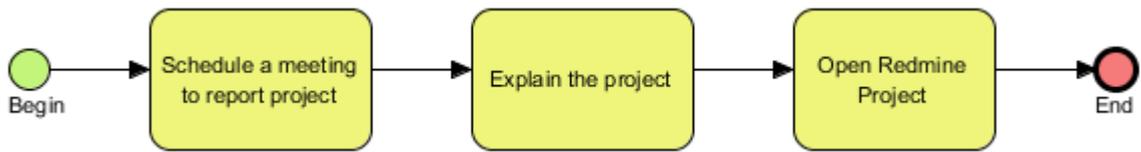
- Implementação das diretrizes no JGOOSE para que o processo de derivação dos Casos de Uso aconteça de forma automatizada.
- Criação de um editor BPMN integrado ao E4J e JGOOSE para dar suporte a modelagem dos processos de negócio no contexto do JGOOSE para a derivação dos Casos de Uso.
- Estudo comparativo entre os Casos de Uso gerados via modelos baseados em BPMN com os Casos de Uso gerados via modelos baseados no framework i\*.

# Apêndice A

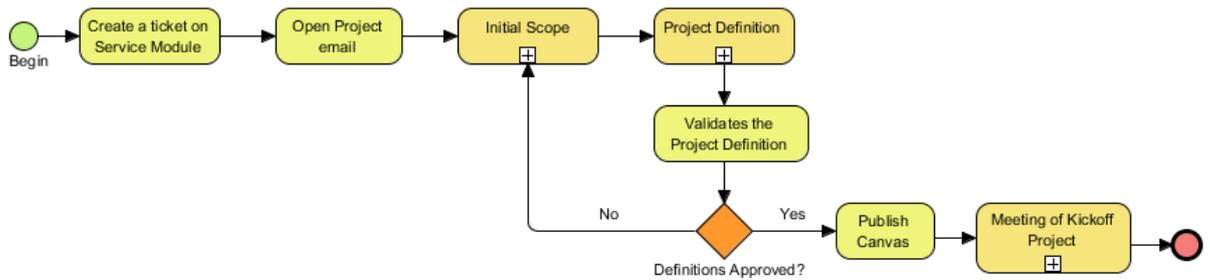
## Processos de Negócio disponibilizados pela Wealth Systems



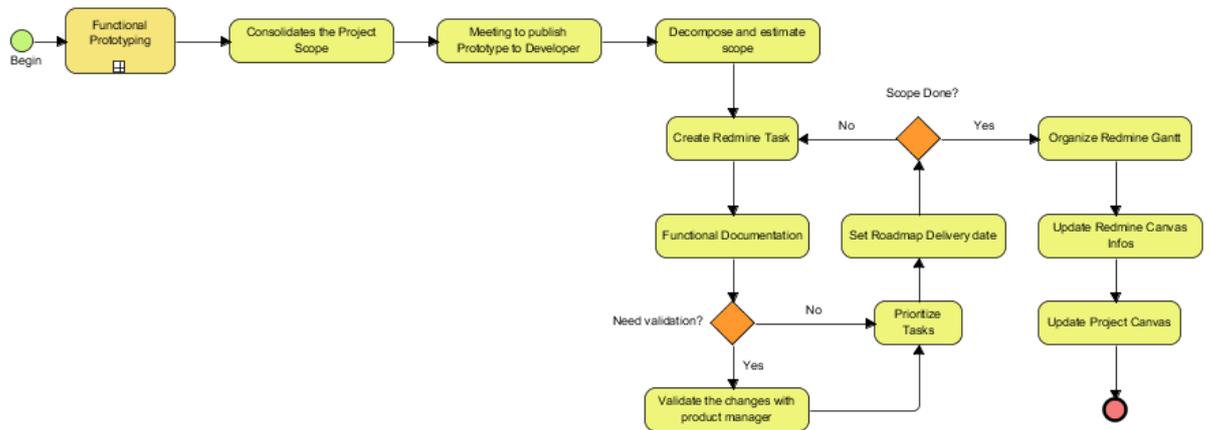
A.1: Processo de negócio disponibilizado para o Estudo de Caso.



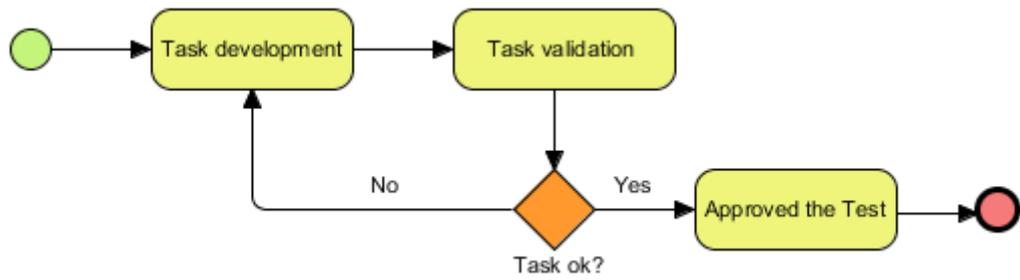
A.2: Parte interna do Sub Process “Report new project” da Figura A.1.



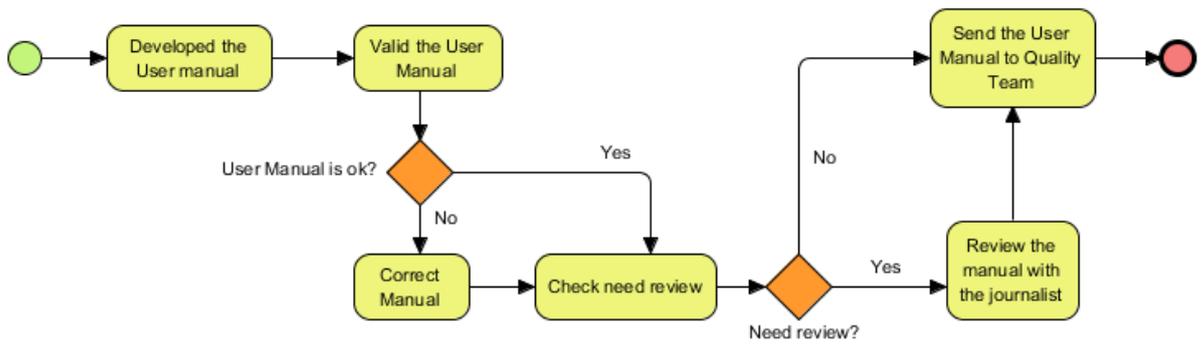
A.3: Parte interna do Sub Process “Planning Project” da Figura A.1.



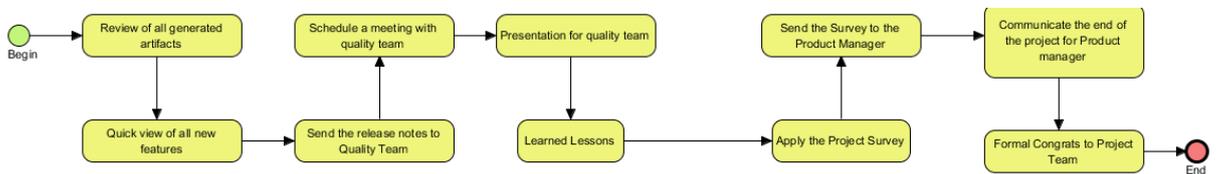
A.4: Parte interna do Sub Process “Functional Specification” da Figura A.1.



A.5: Parte interna do Sub Process “Production” da Figura A.1.



A.6: Parte interna do Sub Process “Create User Manuals” da Figura A.1.



A.7: Parte interna do Sub Process “Closure Project” da Figura A.1.

# Apêndice B

## Descrição dos Casos de Uso

*Use Case: Report new project*

---

### *CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Manager*

---

### *MAIN SUCESS SCENARIO*

*<passo 1> Schedule a meeting to report project*

*<passo 2> Explain the project*

*<passo 3> Open Redmine Project*

*Use Case: Planning project*

---

### *CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Owner + Product Analyst*

---

### *MAIN SUCESS SCENARIO*

*<passo 1> Create a ticket on Service Modelo*

*<passo 2> Open project email*

*<passo 3> Executa o caso de uso **Initial Scope***

*<passo 4> Executa o caso de uso **Project Definition***

*<passo 5> Validates the Project definition*

*<passo 6> Executa o caso de uso **UC2***

---

**EXTENSIONS**

<passo 6> <Deffinitions not approved> : Executa o caso de uso **UC3**

*Use Case: Initial Scope*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Owner + Product Analyst

*Use Case: Project Definition*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Owner + Product Analyst

*Use Case: Meeting of Kickoff Project*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Owner + Product Analyst

*Use Case: UC2*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Owner + Product Analyst

---

**MAIN SUCESS SCENARIO**

<passo 1> Publish Canvas

<passo 2> Executa o caso de uso **Meeting of Kickoff Project**

*Use Case: UC3*

---

*CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Owner + Product Analyst*

---

***MAIN SUCESS SCENARIO***

*<passo 1> Vai para o <passo 3> do caso de uso **Planning project***

*Use Case: Closure Project*

---

*CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Owner + Product Analyst*

---

***MAIN SUCESS SCENARIO***

*<passo 1> Review of all generated artifacts*

*<passo 2> Quick view of all new features*

*<passo 3> Send the release notes to Quality Team*

*<passo 4> Schedule a meeting with Quality Team*

*<passo 5> Presentation for Quality Team*

*<passo 6> Learned Lessons*

*<passo 7> Apply the Project Survey*

*<passo 8> Send the Survey to the Product Manager*

*<passo 9> Communicate the end of the project for Product Manager*

*<passo 10> Formal Congrats to Project Team*

*Use Case: Functional Specification*

---

*CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Analyst*

---

***MAIN SUCESS SCENARIO***

<passo 1> Executa o caso de uso **Functional Prototyping**

<passo 2> Consolidates the Project Scope

<passo 3> Meeting to publish Prototype to Developer

<passo 4> Decompose and estimate scope

<passo 5> Create Redmine Task

<passo 6> Functional Documentation

<passo 7> Executa o caso de uso **UC4**

---

#### **EXTENSIONS**

<passo 7> <Not need validation> : Executa o caso de uso **UC5**

**Use Case: UC4**

---

#### **CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Analyst

---

#### **MAIN SUCESS SCENARIO**

<passo 1> Validate the changes with product manager

<passo 2> Priorize Tasks

<passo 3> Set roadmap Delivery date

<passo 4> Executa o caso de uso **UC6**

---

#### **EXTENSIONS**

<passo 4> <Scope not done> : Executa o caso de uso **UC7**

**Use Case: UC5**

---

#### **CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Analyst

---

**MAIN SUCESS SCENARIO**

*<passo 1> Vai para o <passo 2> do caso de uso UC4*

*Use Case: UC6*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** *Product Analyst*

---

**MAIN SUCESS SCENARIO**

*<passo 1> Organize Redmine Grantt*

*<passo 2> Update Redmine Canvas infos*

*<passo 3> Update Project Canvas*

*Use Case: UC7*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** *Product Analyst*

---

**MAIN SUCESS SCENARIO**

*<passo 1> Vai para o <passo 5> do caso de uso **Functional Specification***

*Use Case: Functional Prototyping*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** *Product Analyst*

*Use Case: Create User Manuals*

---

*CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Analyst*

---

**MAIN SUCESS SCENARIO**

*<passo 1> Developed the User Manual*

*<passo 2> Valid the user manual*

*<passo 3> Executa o caso de uso UC10*

---

**EXTENSIONS**

*<passo 3> <Manual not ok>: Executa o caso de uso UC11*

*Use Case: UC11*

---

*CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Analyst*

---

**MAIN SUCESS SCENARIO**

*<passo 1> Correct Manual*

*<passo 2> Vai para o <passo 1> do caso de uso UC10*

*Use Case: UC10*

---

*CHARACTERISTIC INFORMATION*

**Primary Actor:** *Product Analyst*

---

**MAIN SUCESS SCENARIO**

*<passo 1> Check need review*

*<passo 2> Executa o caso de uso UC12*

---

**EXTENSIONS**

<passo 2> <Not need review>: Executa o caso de uso **UC13**

*Use Case: UC12*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Analyst

---

**MAIN SUCESS SCENARIO**

<passo 1> Review the manual with the journalist

<passo 2> Send the user Manual to Quality Team

*Use Case: UC13*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Analyst

---

**MAIN SUCESS SCENARIO**

<passo 1> Vai para o <passo 2> do caso de uso **UC12**

*Use Case: Production*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** Product Developer + Product Analyst

---

**MAIN SUCESS SCENARIO**

<passo 1> Task development

<passo 2> Task validation

<passo 3> Executa o caso de uso **UC8**

---

**EXTENSIONS**

<passo 3> <Task not ok> : Executa o caso de uso **UC9**

*Use Case: UC8*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** *Product Developer + Product Analyst*

---

**MAIN SUCESS SCENARIO**

<passo 1> *Approved the Test*

*Use Case: UC9*

---

**CHARACTERISTIC INFORMATION**

**Primary Actor:** *Product Developer + Product Analyst*

---

**MAIN SUCESS SCENARIO**

<passo 1> *Vai para o <passo 1> do caso de uso **Production***

## Referências Bibliográficas

- [1] THAYER, M. D. R. H. Software requirements engineering. *IEEE. IEEE Computer Society Press*, 1993.
- [2] KOTONYA, G.; SOMMERVILLE, I. *Requirements engineering: processes and techniques*. [S.l.]: J. Wiley, 1998.
- [3] SCHNEIDER, F.; BERENBACH, B. A literature survey on international standards for systems requirements engineering. *Procedia Computer Science*, Elsevier, v. 16, p. 796–805, 2013.
- [4] MASON, G. L. A conceptual basis for organizational modelling. *Systems Research and Behavioral Science*, Wiley Online Library, v. 14, n. 5, p. 331–345, 1997.
- [5] CHINOSIA, M.; TROMBETTA, A. BPMN: An introduction to the standard. *Computer Standards & Interfaces*, v. 34, p. 124-134, 2012.
- [6] BOOCH, G. RUMBAUGH, J. JACOBSON, I. *UML: Guia do Usuário*, 2º Edição, Editora Campus, 2005.
- [7] SANTANDER, V. F. A. *Integrando Modelagem Organizacional com Modelagem Funcional*. Tese (Tese de Doutorado) — Universidade Federal de Pernambuco, Recife - PE, Dezembro 2002.
- [8] YU, E. S.-K. Modelling strategic relationships for process reengineering. Tese (Doutorado) — University of Toronto, 1995.
- [9] VICENTE, A. A. *JGOOSE: Uma ferramenta de Engenharia de Requisitos para Integração da Modelagem Organizacional i\* com a Modelagem Funcional de Casos de Uso UML*. Trabalho de Conclusão de Curso – Universidade Estadual do Oeste do Paraná, Cascavel – PR, 2006.

- [10] MERLIN, L. P. *E4J: Editor i\* para JGOOSE*. Trabalho de Conclusão de Curso – Universidade Estadual do Oeste do Paraná, Cascavel – PR, 2013.
- [11] TROVA, E. C. V. *A importância da modelagem dos processos de negócios para o desenvolvimento de sistema de informação: uma aplicação em gestão e controle acadêmico*. Dissertação (Dissertação de Mestrado) – Universidade Metodista de Piracicaba, Santa Bárbara D'Oeste - SP, 2004.
- [12] ABPMP. Guia para o Gerenciamento de Processos de Negócio Corpo Comum de Conhecimento. Versão 2.0, 2009.
- [13] ANTON, A. Goal Based Requirements Analysis. In: SECOND INTERNATIONAL CONFERENCE ON REQUIREMENTS ENGINEERING – ICRE'96. Colorado Springs, Colorado, USA: [s.n.], p. 136–144, 1996.
- [14] ALENCAR, F. M. R. Mapeando a Modelagem Organizacional em Especificações Precisas. Tese (Tese de Doutorado) – Centro de Informática – Universidade Federal de Pernambuco, Recife - PE, 1999.
- [15] GRANDO, F. L. Avaliando técnicas de modelagem organizacional no contexto de desenvolvimento de sistemas computacionais. Trabalho de Conclusão de Curso - Universidade Estadual do Oeste do Paraná, Cascavel – PR, 2010.
- [16] Business Process Modeling Notation (BPMN) Information. Disponível em: <<http://www.omg.org/bpmn/Documents/FAQ.htm>>. Consultado na Internet em: 20/07/2014.
- [17] Business Process Model and Notation (BPMN). Disponível em: <<http://www.omg.org/spec/BPMN/2.0>>. Consultado na Internet em: 14/10/2014.
- [18] BPML. Disponível em: <<http://www.omg.org/bpmn/Documents/BPML-2003.pdf>>. Consultado na Internet em: 14/10/2014.
- [19] XML. Disponível em: <<http://www.w3.org/TR/2008/REC-xml-20081126/>>. Consultado na Internet em: 14/10/2014.
- [20] BPEL. Disponível em <[http://www.omg.org/bpmn/Documents/Mapping\\_BPMN\\_to\\_BPEL\\_Example.pdf](http://www.omg.org/bpmn/Documents/Mapping_BPMN_to_BPEL_Example.pdf)>. Consultado na Internet em: 14/10/2014.

- [21] Business Process Management Initiative. Disponível em: <<http://www.bpmi.org>>. Consultado na Internet em: 14/10/2014.
- [22] OMG. Disponível em: <<http://www.omg.org/>>. Consultado na Internet em: 14/10/2014.
- [23] WHITE, S. A.; MIERS, D. BPMN Modeling and Reference Guide: Understanding and Using BPMN. Future Strategies Inc. Florida, 2008.
- [24] BOOCH, G.; RUMBAUGH, J. JACOBSON, I. UML- Guia de Usuário. Vol. 2. Rio de Janeiro: Campus, 1999.
- [25] RUMBAUGH, J.; JACOBSON, I; BOOCH, G. The Unified Modeling Language Reference Manual. Massachusetts: Addison-Wesley Longman Publishing Co., 1999.
- [26] COCKBURN, A. Writing effective Use Cases. 1ª ed. Boston: Addison-Wesley Longman Publishing Co., 2000.
- [27] RODRÍGUEZ, A.; CARO, A. Obteniendo Casos de Uso centrados en la Calidad de los Datos desde Procesos de Negocio descritos con BPMN. Revista Ibérica Sistemas e Tecnologias de Informação, Nº 10, p. 65-80, 2012.
- [28] RODRÍGUEZ, A.;MEDINA, E. F; PIATTINI, M. Using QVT to obtain Use Cases from Secure Business Processes modeled with BPMN. 8th Workshop on Business Process, 2007.
- [29] QVT. Disponível em: <<http://www.omg.org/spec/QVT/1.1/PDF/>>. Consultado na Internet em: 14/10/2014.
- [30] SHULL, F.; CARVER, J.; TRAVASSOS, G. H. An Empirical Methodology for Introducing Software Processes. In *Proceedings of the 8th European software engineering conference held jointly with 9th ACM SIGSOFT international symposium on Foundations of software engineering (ESEC/FSE-9)*. ACM, New York, NY, USA, 288-296.
- [31] SOLINGEN, R.; BERGHOUT, E. The goal/question/metric method: a practical guide for quality improvement of software development. UK: *McGraw-Hill*, 1999.