

# Usando Modelos de Processos de Negócio para Especificação de Requisitos de Software

Fernando Aparecido Nogueira<sup>1</sup>, Hilda Carvalho de Oliveira<sup>2</sup>.

## Resumo

Este trabalho apresenta um meio de aproximar a Engenharia de Requisitos dos processos de negócio de uma organização, visando melhor aderência dos requisitos ao domínio do negócio. Modelos de processos de negócio organizam e documentam vários elementos, ações e decisões que contribuem para o levantamento e elaboração de modelos de requisitos do software. Assim, este trabalho apresenta uma sistemática para extrair automaticamente requisitos funcionais, não-funcionais e

restrições do software a partir de um modelo de processos de negócio em BPMN (*Business Process Modeling Notation*). Para isso, considera a conversão do modelo para XPDL (*XML Process Definition Language*) e um conjunto de heurísticas definido. O resultado é um documento que apoia a especificação de requisitos do software, contendo uma parte textual e diagramas UML (*Unified Modeling Language*). O trabalho exemplifica a aplicação da sistemática usando modelos reais de processos de negócio.

**Palavras chave:** Engenharia de requisitos; modelagem de processos de negócio; BPMN; XPDL; levantamento de requisitos; especificação de requisitos; heurísticas; UML.

<sup>1</sup> Mestrando em Ciência da Computação na Universidade Estadual Paulista na área de Engenharia de Software, atua profissionalmente como Analista de Sistemas Pleno (Desenvolvimento) na Empresa Brasileira de Correios e Telégrafos. Possui Especialização em Tecnologia da Informação pela Universidade Estadual Paulista - Bauru (2007) e graduação em Informática - Ênfase em Gestão de Negócios pela Faculdade de Tecnologia - Botucatu (2005). Tem experiência na área de Ciência da Computação, com ênfase em Desenvolvimento de Sistemas.

<sup>2</sup> Doutora em Engenharia Elétrica (Sistemas Digitais) pela Escola Politécnica da USP, com pesquisas em integração de informações em sistemas abertos. Mestre em Ciência da Computação pela Unicamp, na linha de Bancos de Dados Não-convencionais. Bacharel em Matemática pela Universidade Estadual Paulista (Unesp), com trabalhos de pesquisa em sistemas especialistas. É professora da Unesp desde 1989 na área de Ciência da Computação, atuando junto à graduação e pós-graduação. Coordena o Grupo de Pesquisa "Engenharia de Software e Tecnologias da Informação e Comunicação (LesTIC)". Atualmente, suas áreas de interesse são direcionadas à convergência digital, e-learning, t-Learning, inclusão digital, gestão de projetos de software e desenvolvimento de software orientado a domínio. Além disso, desenvolve projetos de extensão universitária voltados à inclusão digital e social de adolescentes e adultos, bem como participa de projetos de apoio administrativo.

## 1. INTRODUÇÃO

No levantamento de requisitos de um software, é importante que os analistas de requisitos capturem informações sobre o domínio de negócio da organização. Isso envolve uma série de atividades, tais como: entrevistas, análises de documentos, discussões sobre sistemas legados, entre outras. A comunicação entre as equipes técnicas e de negócio deve ser eficiente, de modo que o produto de software reflita as reais necessidades das partes interessadas. Entretanto, essas equipes usam diferentes vocabulários, diferentes linguagens e modelos técnicos -o que dificulta o levantamento e a modelagem de requisitos do software (BOUSETTA et al., 2013) (VIEIRA et al., 2012). Isso pode levar a diversos problemas, como, por exemplo, dificuldades na definição do escopo da solução de software e falhas de comunicação entre clientes, usuários e equipes de desenvolvimento, assim como requisitos definidos de modo incompleto e/ou mesmo inexistentes. A modelagem de processos de negócio tem sido cada vez mais utilizada por organizações de várias áreas, visando a eficiência e efetividade de seus processos, bem como análises para otimização dos recursos. Isso tem ajudado a enfatizar a adoção da abordagem de gerenciamento de processos de negócio, conhecida como BPM (*Business Process Management*), que envolve análise, definição, execução, monitoramento e administração dos processos.

Processos de negócio bem definidos proporcionam o entendimento do que a organização faz nas suas áreas de negócio e como o processo é executado, passando por diferentes setores da organização (SGANDERLA, 2014). De modo geral, os setores da organização (departamentos, divisões, diretorias, etc.) organizam as pessoas e seus trabalhos, mas um produto entregue ao cliente final é resultado da interação entre diferentes setores, com clientes internos à organização. Esse fluxo transversal de interações é documentado pelos modelos de processos de negócio, que também propiciam um

cenário claro para a introdução de melhorias na execução dos processos, considerando softwares e demais recursos utilizados.

Segundo Sganderla (2014), os requisitos de software podem ser determinados de forma mais assertiva se forem considerados os modelos de processos de negócio da organização. Quanto mais a documentação de requisitos do software espelhar os processos da organização, maior será o nível de conformidade desses requisitos com as necessidades do cliente (VIEIRA et al., 2012). Consequentemente, isso terá reflexos positivos nas próximas etapas de desenvolvimento do software, cujo produto quando entregue ao cliente final irá atender de forma mais efetiva as necessidades dos usuários e das partes interessadas da organização.

Nesse contexto, este trabalho propõe a aproximação entre o conhecimento do negócio e as fases de levantamento, análise e especificação da Engenharia de Requisitos. Para isso, é proposta uma sistemática para extração automática de requisitos funcionais, não funcionais e restrições do software a partir de modelos de processos de negócio na notação BPMN v2.0 (*Business Process Model and Notation*) (OMG, 2011). Essa notação é definida pelo consórcio OMG (*Object Management Group*), sendo composta por elementos gráficos que representam os elementos dos processos de negócio, suas conexões e fluxo de informações entre eles. Observa-se que os modelos de processos de negócio não são restritos à notação gráfica.

De forma a enriquecer a documentação dos processos de negócio, a especificação BPMN prevê o registro de informações textuais no próprio modelo. Ainda nesse sentido, as ferramentas de modelagem BPMN permitem a criação de atributos estendidos em cada elemento do modelo de processos, de acordo com a necessidade da equipe de modelagem.

Convém destacar que para a aplicação da sistemática, é importante que os modelos de negócio

considerados sejam bem documentados e aderentes à realidade da organização. Assim, este trabalho apresenta algumas recomendações que podem propiciar ajustes nos modelos de negócio para torná-los mais completos, beneficiando também a organização.

A sistemática proposta requer que os modelos de processos de negócio em BPMN v2.0 sejam convertidos para XPDL v2.2 (*XML Process Definition Language*), definida pelo consórcio WfMC (*Workflow Management Coalition*) (WFMC, 2012). Essa conversão é feita geralmente pelos próprios sistemas de modelagem BPMN. XPDL estabelece um formato estruturado baseado em XML (*eXtensible Markup Language*) para representação das definições dos processos. A partir da especificação de um metamodelo único, diferentes ferramentas de modelagem podem interagir e gerar documentos no padrão XPDL a partir de modelos BPMN, sem perda de informações nesse processo de transformação (VAN DER AALST, 2003).

Para a extração de informações dos modelos de processos de negócio em XPDL, foi definido um conjunto de heurísticas que, aplicadas aos modelos, ajudam a identificar os elementos que apresentam conteúdos de interesse.

Os conteúdos extraídos são passados para uma estrutura que compõe um documento de Especificação de Requisitos de software, que contempla texto e diagramas em UML (Unified Modeling Language). Esse documento constitui um ponto de partida para os analistas de requisitos, de modo a facilitar a comunicação com a organização e melhorar o tempo gasto na especificação dos requisitos do software.

Frente ao exposto, a seção 2 apresenta alguns trabalhos que também usam modelos de processos de negócio para apoiar o levantamento de requisitos de software, embora de maneira diferenciada deste trabalho. A seção 3 apresenta conceitos relacionados à modelagem de processos de negócio em

BPMN e XPDL, que são relevantes à proposta. Já a sistemática para extração de requisitos de software a partir de modelos de processos de negócio é apresentada na seção 4. A seção 5 apresenta a aplicação da sistemática proposta, utilizando modelos de processo de negócio reais. As considerações finais e perspectivas para trabalhos futuros são apresentadas na seção 6.

## 2. TRABALHOS RELACIONADOS

Na literatura podem ser encontrados alguns trabalhos que contribuem para o levantamento de requisitos de software a partir de modelos de processos de negócio. Esses trabalhos diferem quanto à notação de modelagem, complexidade e detalhamento dos processos de negócio, assim como ao modo de tratamento dos requisitos de software.

Nesta seção, são destacados cinco trabalhos, com abordagens diferentes: (1) Xavier et al. (2010), Bousetta et al. (2013) e Vieira et al. (2012), que também usam modelos de processos de negócio em BPMN; (2) Dias et al. (2006), que utiliza diagramas de atividades UML como insumos; (3) Oliveira et al. (2013), que usa modelos que foram construídos usando a abordagem de modelagem organizacional EKD (*Enterprise Knowledge Development*) para identificação dos requisitos.

Xavier et al. (2010) destacam que modelos em BPMN não contemplam a representação de requisitos de qualidade (não funcionais) de forma explícita. Assim, os autores estendem a notação BPMN para atender à modelagem de requisitos não funcionais. Contudo, isso requer o uso de duas ferramentas distintas para geração dos modelos: um sistema de modelagem em BPMN e o arcabouço de requisitos não funcionais definido pelos autores.

Bousetta et al. (2013) propõem um modelo baseado no domínio de regras de negócio, que auxilia a definição da estrutura do software, por meio da geração de diagramas de classes, de casos de uso e de sequência usando UML. Os autores utilizam um

conjunto de heurísticas para reduzir as ambiguidades encontradas nos diagramas de processos de negócio. Diagramas BPMN de alto nível e de baixo nível (mais detalhados) são utilizados de maneiras diferentes. Casos de uso são identificados a partir dos modelos de alto nível, enquanto diagramas de atividades e de classes são gerados a partir dos modelos de baixo nível. Todo o processo é semi-automatizado, pois é necessário ajustar manualmente os diagramas gerados.

Vieira et al. (2012) utilizam heurísticas para identificação de requisitos funcionais e não funcionais a partir de modelos de processos de negócio em BPMN. Segundo os autores, esses modelos constituem uma pré-condição para se compreender o contexto para o qual o software será desenvolvido. Contudo, Oliveira et al. (2013) comentam que essa necessidade de modelos de processos de negócio em BPMN traz uma grande limitação ao trabalho de Vieira et al. (2012). Assim, Oliveira et al. (2013) adaptaram a técnica de Vieira et al. (2012) para usar, como pré-condição para a identificação de requisitos, modelos de processos de negócio gerados sob o método de modelagem organizacional EKD.

Dias et al. (2006) exploram a relação entre as atividades de processos de negócio e casos de uso. Uma atividade de processo de negócio consiste de um conjunto de ações detalhadas para o cumprimento do objetivo do processo. Por outro lado, um caso de uso é um conjunto de passos a serem executados com a finalidade de atingir um objetivo computacional. Assim, os autores usam heurísticas para a definição de casos de uso, estabelecendo uma relação entre diagramas BPMN e a especificação de requisitos com UML.

É importante observar que os trabalhos de Bousetta et al. (2013), Vieira et al. (2012) e Oliveira et al. (2013) utilizaram heurísticas para a extração dos requisitos de software. Essas heurísticas contribuíram para que os procedimentos executados fossem registrados, validados e repetidos a cada etapa da execução das técnicas apresentadas.

Considerando os trabalhos mencionados nesta seção, vale destacar que apenas os procedimentos descritos por Dias et al. (2006) são executados de forma automática, por meio do uso de uma ferramenta de software específica. Nos demais trabalhos (OLIVEIRA et al., 2013) (VIEIRA et al., 2012) (XAVIER et al., 2010), os procedimentos são realizados manualmente ou de forma semiautomática (BOUSETTA et al., 2013), o que impacta no tempo utilizado e nos resultados obtidos.

### 3. MODELOS DE PROCESSOS DE NEGÓCIO

Para início da modelagem de processos de negócio de uma organização, devem ser identificados os elementos envolvidos na execução dos processos. Isso requer conhecimento dos processos, seus atores, clientes internos e externos, recursos utilizados e restrições associadas. Há diferentes notações e técnicas para se elaborar modelos de processos de negócio, como, por exemplo: BPMN, fluxogramas, diagramas de fluxos de dados, diagramas UML, método EKD, entre outros. Os modelos também podem ter diferentes níveis de detalhes, podendo expressar desde uma visão contextual abstrata até uma visão bem detalhada, com informações adicionais associadas a cada elemento. Os modelos de processos de negócio podem ser categorizados de acordo com o momento e necessidade da organização (SGANDERLA, 2014):

- **“as-is”**: representa a situação atual dos processos da organização e serve de documentação para conhecimento do processo atual”;
- **“to-be”**: representa uma evolução do modelo “as-is”, com a reavaliação das questões de negócio pelos profissionais de modelagem;
- **“to-do”**: tem como base o processo “to-be”, busca prover um olhar do processo a partir da visão da área de sistemas, identificando maneiras de agregar valor ao processo por meio de recursos tecnológicos;

Neste trabalho, os modelos de processos de negócio utilizados apresentam as visões “*to-be*” e “*to-do*”. Isso visa garantir que os modelos estejam próximos da representação de atividades que serão automatizadas por meio do desenvolvimento de uma solução de software.

Jung et al. (2004) identificam duas abordagens de modelagem de processos de negócio: uma usando notação gráfica, como BPMN, e outra usando linguagens de execução, como BPML (*Business Process Modeling Language*), BPEL (*Business Process Execution Language*) e XPDL. Os autores deixam evidente que as duas maneiras trazem benefícios à modelagem. As linguagens de execução possibilitam a compreensão das informações que fazem o processo ser executado. Geralmente, essas linguagens são estruturadas de acordo com o padrão XML, tais como a linguagem XPDL. As subseções 3.1. e 3.2 apresentam uma visão geral sobre BPMN e XPDL, respectivamente.

### 3.1 NOTAÇÃO BPMN

Segundo Correia e Abreu (2015), BPMN é atualmente a notação de modelagem de processos de negócio mais usada entre profissionais de BPM, devido à sua flexibilidade e abrangência. BPMN é relativamente recente e pode ser utilizada por profissionais com diferentes níveis de conhecimento técnico. Embora não tenha recursos para representar o mecanismo interno da execução de um processo, essa notação ele permite construir modelos que ajudam o projeto de um processo de negócio.

A especificação BPMN provê uma notação gráfica para representar processos de negócio por meio de um diagrama denominado BPD (*Business Process Diagram*). Esse diagrama é concebido a partir de um conjunto de elementos gráficos que compõem diagramas simples de serem desenvolvidos e compreendidos. Para este trabalho, os modelos construídos devem estar na versão 2.0 de BPMN, publicada em 2011 (OMG, 2011). A especificação completa

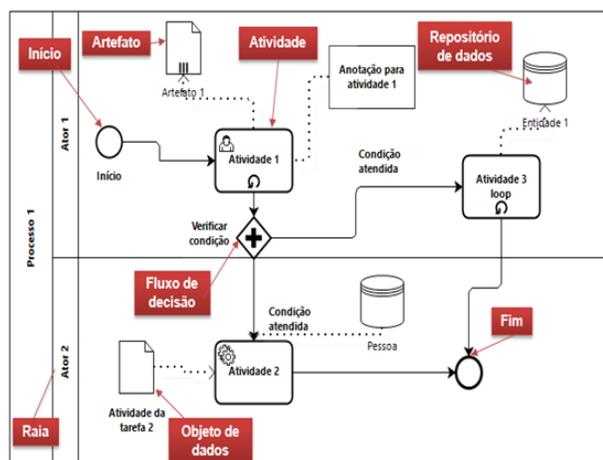
de BPMN v2.0 define atributos que são agrupados em quatro categorias básicas de elementos: objetos de fluxo (*Flow Objects*), objetos de conexão (*Connecting Objects*), vias ou raias (*Swimlanes*) e artefatos (*Artifacts*). A Figura 1 exemplifica um diagrama de processo de negócio simples, com alguns elementos da notação BPMN v2.0 identificados por meio de notas, em cor mais escura.

Observa-se que os elementos básicos usados na construção de modelos BPMN são os objetos de fluxo, tais como: eventos, atividades e fluxos de decisão (*gateways*). Objetos de conexão são usados para interconectar objetos de fluxo usando diferentes tipos de setas. As vias (*swimlanes*) são usadas para agrupar as atividades em categorias separadas, de acordo com suas capacidades funcionais ou responsabilidades. Já os artefatos adicionam informações aos elementos do processo, tais como dados processados ou comentários (XAVIER et al., 2010).

### 3.2 LINGUAGEM XPDL

A principal característica do formato XPDL para representação dos processos de negócio é possibilitar que um processo de negócio definido com a estrutura XPDL possa ser interpretado por diferentes ferramentas de software (WFMC, 2012). A especificação da linguagem XPDL prevê que as ferramentas de modelagem proporcionem suporte a duas operações: (1) importar uma definição de processo a partir de código XPDL; (2) exportar para XPDL uma definição de processo, segundo a representação interna da ferramenta para os processos de negócio. Para isso, é utilizado um documento no padrão XSD (*XML Schema Definition*) (WFMC, 2012).

**Figura 1** — Exemplo de processo de negócio representado em BPMN.



A especificação da linguagem XPD L estabelece um conjunto básico de entidades e atributos para representação de vias, raias, processos, participantes, fluxos de mensagem e diferentes tipos de elementos que representam o fluxo de controle em modelos de processos de negócio. Além disso, a especificação define a serialização dos modelos de processos em formato XML, incluindo informações conceituais e gráficas (para representação dos diagramas de processos de negócio) (WESKE, 2012). Neste trabalho, a linguagem XPD L apoia a automatização da sistemática proposta porque fornece um padrão estruturado de informações sobre os elementos dos processos de negócio.

A **Figura 2** apresenta um exemplo de representação de um processo de negócio no formato XPD L. É possível identificar o processo de *workflow* que representa a definição de processo, identificado como “Processo”. Em seguida, são representadas as atividades identificadas como “A” e “B”, respectivamente. Essas atividades são conectadas por um elemento do tipo transição (*Transition*), identificado como “AB”.

**Figura 2** — Exemplo de processo de negócio representado no formato XPD L.

```
<WorkflowProcess Id="Process A">
  <Activities>
    <Activity Id="A">
      ...
    </Activity>
    <Activity Id="B">
      ...
    </Activity>
  </Activities>
  <Transitions>
    <Transition Id="AB" From="A" To="B"/>
  </Transitions>
</WorkflowProcess>
```

#### 4. SISTEMÁTICA PARA EXTRAÇÃO DE REQUISITOS DE SOFTWARE

Para a aplicação da sistemática proposta, é necessário que se tenha um modelo de processos de negócio com diagramas em BPMN v2.0, com alguns elementos básicos que possibilitem a extração de informações. Os requisitos deverão ser extraídos de cada diagrama que faz parte do modelo. O diagrama deve apresentar pelo menos uma raia e uma piscina. Além disso, deve conter apenas um evento de início, pelo menos um evento de fim e, no mínimo, uma atividade.

É comum encontrar modelos de processos de negócio com diagramas BPMN que não apresentam um nível de detalhe adequado para apoiar o trabalho das equipes de desenvolvimento. Nesse sentido, os elementos dos modelos de processo podem ser documentados de forma gráfica, por meio da inclusão de objetos e símbolos específicos, ou textualmente junto ao respectivo elemento gráfico. Assim, é interessante que o modelo considerado seja submetido a uma análise para melhoria de seu conteúdo semântico. Isso pode ser feito por meio de heurísticas apresentadas na *subseção 4.1*.

Uma vez ajustado, o modelo deve ser convertido para a linguagem XPD L v2.2. Isso pode ser feito

por meio das próprias ferramentas de modelagem BPMN v2.0, que exportam os diagramas para códigos estruturados em XPD. Essa conversão é necessária para se estabelecer um formato estruturado em XML, de modo que se automatize o processo de identificação dos elementos dos processos de negócio, a partir dos quais serão extraídas informações para o documento de Especificação de Requisitos do software.

Após essa etapa, inicia-se o processo de extração de requisitos do software com a aplicação de um conjunto de heurísticas específicas, conforme apresentado na *subseção 4.2*. Essas heurísticas definem os procedimentos para identificação dos requisitos funcionais, requisitos não-funcionais e regras de negócio a partir dos modelos de processos de negócio.

A representação desses requisitos é feita de forma textual nas respectivas seções do documento de Especificação de Requisitos, apresentado na *subseção 4.4*. A etapa final, apresentada na *subseção 4.3*, é a construção dos diagramas de caso de uso e diagramas de classes UML, de acordo com os casos de uso e classes de domínio identificados depois da aplicação das heurísticas de extração de requisitos.

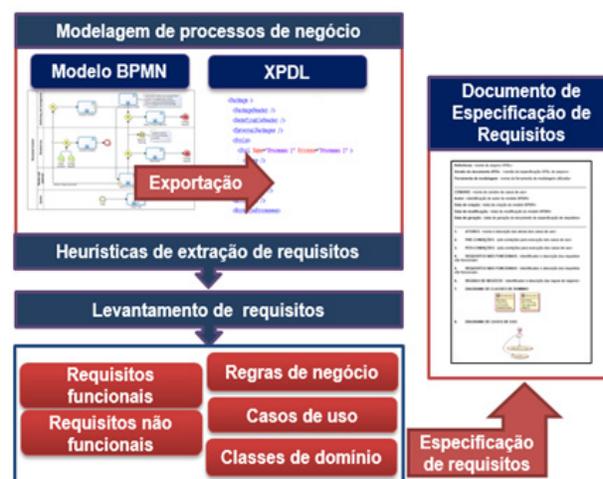
O resultado da extração dos requisitos é um documento de Especificação de Requisitos de software, cuja estrutura é apresentada na *subseção 4.4*. Esse documento servirá de apoio para a comunicação entre as equipes técnicas de análise de requisitos e as equipes da organização que passou o modelo de processos de negócio. A **Figura 3** apresenta uma visão geral da sistemática proposta.

#### 4.1 AJUSTES NOS MODELOS BPMN

Para a extração de requisitos e restrições de software, é importante que os modelos de processos de negócio sejam semanticamente ricos, ou seja, apresentem conteúdo que represente adequadamente o conhecimento do domínio do negócio. Isso visa garantir que as equipes de modelagem e de

desenvolvimento tenham o mesmo entendimento sobre os processos de negócio da organização. Para isso, é essencial o uso correto dos diversos recursos disponibilizados pela notação BPMN para criação de modelos de processos, de forma que os modelos reflitam os processos de negócio.

**Figura 3** — Visão geral da sistemática proposta.



Assim, esta etapa da sistemática recomenda que o modelo de processo de negócio considerado seja avaliado e ajustado, se necessário, com a introdução de informações adicionais. Essas informações são inseridas em pontos chave do modelo, de acordo com a análise propiciada pela aplicação de oito heurísticas, apresentadas na **Tabela 1**.

Os ajustes não envolvem alteração nos fluxos de decisão ou a remoção de algum elemento original. No entanto, pode ser necessário acrescentar novos elementos, que representem as informações manipuladas pelas atividades durante o fluxo do processo. Esses elementos podem ser objetos de dados e repositórios de dados, de acordo com a situação. A informação sobre o tipo de atividade (“Usuário”, “Serviço” ou “Script”) também deve ser verificada e corrigida, caso necessário. Essa informação será utilizada para a extração de requisitos funcionais, conforme apresentado na subseção seguinte.

Convém observar que esta etapa do processo pode ser repetida ao final da sistemática, avaliando-se o documento de Especificação de Requisitos gerado. Caso o documento contemple informações insuficientes ou inadequadas, recomenda-se submeter novamente o modelo de processos de negócio à aplicação das heurísticas da **Tabela 1** para que novos resultados sejam obtidos. Isso pode ser feito sempre que necessário, visando obter modelos mais ricos semanticamente e mais aderentes ao domínio do negócio.

**Tabela 1** — Heurísticas de negócio e de requisitos usadas para identificação dos requisitos de software.

Heur. Neg.	Heur. Req.	Descrição da heurística de negócio
<b>HN1</b>	<b>HR1, HR2</b>	Adequar a descrição das condições de início e fim de do processo aos respectivos eventos, de acordo com o contexto do processo de negócio.
<b>HN2</b>	<b>HR3</b>	Atribuir o executor correto em cada uma das atividades, de acordo com o contexto e a raia em que estão posicionadas.
<b>HN3</b>	<b>HR3</b>	Atribuir o tipo correto em cada uma das atividades do modelo de processos de negócio.
<b>HN4</b>	<b>HR4</b>	Adequar a descrição de cada um dos fluxos de decisão às regras de negócio, de acordo com o contexto do processo.
<b>HN5</b>	<b>HR5</b>	Identificar os requisitos não funcionais informados nas descrições textuais e atributos estendidos de cada uma das atividades, transpondo-as para os atributos estendidos "RNF".

<b>HN6</b>	<b>HR6</b>	Identificar as regras de negócio nas descrições textuais e atributos estendidos de cada uma das atividades, transpondo-as para os atributos estendidos "REGRA".
<b>HN7</b>	<b>HR7</b>	Identificar as entidades manipuladas por cada uma das atividades, inserindo repositórios de dados que representem essas entidades; Identificar os atributos, de acordo com o contexto do processo, e transpô-los para os atributos estendidos "ATRIBUTOS".
<b>HN8</b>	<b>HR7</b>	Identificar os artefatos gerados e consumidos por cada uma das atividades, inserindo objetos de dados que representem esses artefatos; Identificar os atributos, de acordo com o contexto do processo, e transpô-los para os atributos estendidos "ATRIBUTOS".

#### 4.2 LEVANTAMENTO DE REQUISITOS DE SOFTWARE

Após a avaliação do modelo de processo de negócio segundo a *subseção 4.1*, o modelo deve ser convertido para código XPDL v2.2. Para a estabilidade da sistemática, foi estabelecida uma relação entre os elementos BPMN v2.0 e XPDL v2.2. Isso foi possível porque as especificações de BPMN e XPDL definem a função de cada elemento na representação de processos de negócio (MORA et al., 2007) (VAN der AALST, 2003). Entretanto, o relacionamento entre elas não é definido de forma explícita pelos consórcios que as definem. Assim, foi utilizada a relação entre BPMN e XPDL propostas por Van der Aalst (2003), bem como por Mora et al. (2007). Por outro lado, para relacionar os elementos da

linguagem XPDL com os tipos de requisitos de software a serem extraídos, foi necessário estabelecer uma correspondência entre esses elementos, visando a automação da extração de requisitos de software e outras informações. A **Tabela 2** apresenta os elementos BPMN utilizados neste trabalho, bem como os respectivos elementos XPDL e os elementos de software a serem extraídos.

**Tabela 2** — Relação entre os elementos BPMN e XPDL, com os respectivos elementos de software.

Elemento BPMN	Elemento XPDL	Elemento de software
Diagrama	Processo de workflow (WorkflowProcess)	Cenário
Executor	Ator (Performer)	Ator
Atividade	Atividade (Activity)	Funcionalidade/caso de uso
Fluxo de decisão	Atividade do tipo Rota (Route)	Regra de negócio
Objeto de dados	Objeto de dados (DataObject)	Classe e atributos
Repositório de dados	Referência de repositório de dados (DataStoreReference)	Classe e atributos

Para o processo de extração dos requisitos e demais informações do software, foram definidas sete Heurísticas Requisitos, representadas por “HR” e um número sequencial, conforme apresentado na **Tabela 3**. Cada uma dessas heurísticas leva em consideração um tipo de elemento do processo de negócio e apresenta os passos que devem ser seguidos para identificação dos requisitos de software.

A heurística HR1 identifica o evento que inicia o processo de negócio, visando obter a pré-condição para a execução dos casos de uso. A documentação desse evento serve como base para extração da pré-condição, registrada na seção “Pré-condição” do documento de Especificação de Requisitos de software. De maneira semelhante, a heurística HR2

é responsável pela extração das pós-condições, a partir da documentação dos eventos de fim de processo. Cabe ressaltar que uma das restrições da sistemática apresentada é que os modelos tenham um único evento de início e pelo menos um de fim.

A heurística HR3 é responsável pela identificação dos requisitos funcionais, bem como dos casos de uso que serão representados graficamente no documento de Especificação de Requisitos. Para estabelecer uma relação entre o papel das atividades no

processo de negócio e os casos de uso em um software, a heurística HR3 identifica as atividades dos processos de negócio que são do tipo “Usuário”, “Serviço” ou “Script”. Atividades do tipo “Usuário” representam a execução de uma tarefa por um ser humano com auxílio de uma ferramenta computacional. As atividades do tipo “Serviço” representam ações realizadas de forma puramente computacional. Já as atividades do tipo “Script” são executadas por meio de mecanismos criados em determinada linguagem compreendida pelo processo de negócio. Assim, a partir da documentação de cada uma das atividades selecionadas, os re-

quisitos funcionais são identificados e registrados no documento de Especificação de Requisitos, na seção “Requisitos funcionais”.

**Tabela 3** — Heurísticas utilizadas para extração dos requisitos de software.

Heurística	Descrição
<b>HR1: pré-condições</b>	Identificar o elemento de início do processo; Transformar o elemento identificado em pré-condição.
<b>HR2: pós-condições</b>	Identificar os elementos de fim do processo; Verificar a existência de informação sobre o evento de fim; Transformar o elemento identificado em pós-condição.
<b>HR3: casos de uso</b>	Identificar as atividades do tipo "Usuário", "Serviço" ou "Script"; Transformar as atividades em casos de uso; Verificar a existência de recursos que executam a atividade; Transformar os recursos em atores; Relacionar os atores identificados aos respectivos casos de uso.
<b>HR4: requisitos não-funcionais</b>	Identificar as atividades do tipo "Usuário", "Serviço" ou "Script"; Transformar os atributos estendidos "RNF" encontrados em requisitos não funcionais.
<b>HR5: regras de negócio (fluxos de decisão)</b>	Identificar os fluxos de decisão; Transformar as condições documentadas nos fluxos em regras de negócio.

<b>HR6: regras de negócio (atividades)</b>	Identificar as atividades do tipo "Usuário", "Serviço" ou "Script"; Transformar os atributos estendidos "REGRA" encontrados em regras de negócio.
<b>HR7: classes de domínio</b>	Identificar os objetos de dados e repositórios de dados; Transformar os objetos de dados e repositórios de dados em classes de domínio; Transformar os atributos estendidos "ATRIBUTOS" encontrados em atributos das classes de domínio; Relacionar os atributos às respectivas classes de domínio identificadas pela heurística.

Considerando, neste trabalho, que requisitos não funcionais são relacionados a fatores de qualidade na execução de um software, foi definido que esses requisitos seriam informados junto aos casos de uso. Para isso, as atividades candidatas a serem transformadas em casos de uso devem ser ajustadas, de modo que se registrem os requisitos não funcionais, possibilitando a aplicação da heurística HR4. Foram então definidos atributos estendidos, identificados como "RNF", que possibilitam o registro dos requisitos não funcionais nas atividades do processo de negócio. Para permitir a inserção de múltiplos requisitos não funcionais para uma mesma atividade, foi utilizado o caractere ponto e vírgula, ";", como separador entre cada requisito não funcional.

Na modelagem de processos de negócio com BPMN, os fluxos de decisão representam a possibilidade de o fluxo de atividades tomar um ou mais caminhos distintos. Neste trabalho, foi estabelecido que essas restrições associadas a tomadas de decisão, documentadas nos respectivos fluxos de decisão, representam as próprias regras de negócio.

Essas restrições são descritas em linguagem natural e documentam a forma com que o conhecimento do negócio da organização é usado para tomada de decisões. Nesse sentido, a heurística HR5 utiliza a documentação dos fluxos de decisão para extrair as regras de negócios do software.

Por outro lado, há regras de negócio que não são implementadas em fluxos de decisão. Um exemplo disso é o caso em que uma determinada tarefa deve ser executada em certo tempo por certo usuário. Assim, essa regra deveria ter sido registrada na própria atividade que apresenta uma restrição. Para isso, foi criada a heurística HR6, que prevê a inclusão dos atributos estendidos identificados como “REGRA” nas atividades do processo de negócio para as quais a organização estabelece restrições.

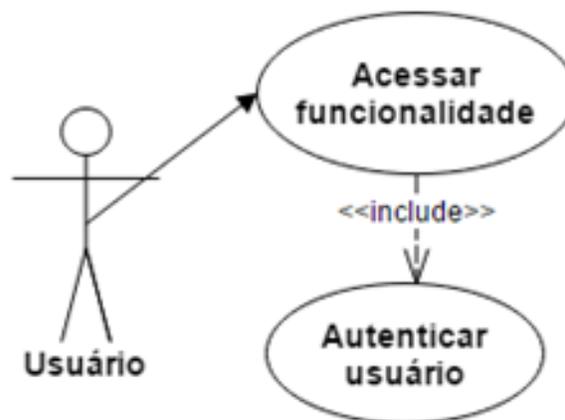
### 4.3 GERAÇÃO DE DIAGRAMAS UML

A obtenção dos atores dos casos de uso é feita pela identificação dos executores de cada uma das atividades identificadas. Essa identificação é prevista na notação BPMN e na linguagem XPDL através da informação do executor (*performer*) da atividade. Os atores identificados são registrados na seção “Atores” do documento de Especificação de Requisitos.

Apesar dos casos de uso já terem sido definidos, é necessário que sejam identificadas as relações de associação entre eles, com destaque para a associação de inclusão. Conceitualmente, uma associação de inclusão expressa a relação em que um caso de uso invoca a execução de outro caso de uso para continuidade do fluxo de informações de um software. Dessa forma, foi estabelecida uma relação de inclusão de um primeiro para um segundo caso de uso, representada por meio de uma linha tracejada e do termo “<<include>>” no diagrama de casos de uso. A **Figura 4** apresenta um exemplo de associação de inclusão, em que o caso de uso “Acessar funcionalidade” invoca a execução do caso de uso “Autenticar usuário”.

Do ponto de vista de sistemas de informação, os elementos identificados como repositórios de dados e objetos de dados da notação BPMN representam objetos de armazenamento de informações, que são manipuladas durante a execução do processo de negócio. Assim, é possível identificar as classes de domínio do software a partir da documentação da estrutura desses elementos e das associações com as atividades dos processos de negócio. Nesse contexto, a aplicação da heurística HR7 é responsável por atribuir uma classe de domínio a cada um dos objetos de dados e repositórios de dados. Contudo, como a representação dos atributos que compõem esses objetos não é prevista de maneira explícita pela notação BPMN, foram incluídos os atributos estendidos “ATRIBUTOS” para o registro dos atributos dos objetos de dados e repositórios de dados.

**Figura 4** — Exemplo de associação de inclusão entre casos de uso.



Para representar o domínio de cada atributo, é necessário registrar os tipos de dados dos atributos identificados. Para reduzir problemas decorrentes da utilização de linguagem natural no processo de identificação dos atributos dos objetos de dados e repositórios de dados, foi definida uma sentença formal para representação do nome do atributo e

seu respectivo tipo de dado. Complementarmente, também foi definido um dicionário de termos para representação dos tipos de dados, conforme apresentado na **Tabela 4**. A criação desse dicionário teve dois motivos: facilitar o registro dos tipos de dados durante a modelagem do processo de negócio e restringir a utilização de linguagem natural.

Para a definição dos tipos de dados apresentados graficamente nos diagramas de classe de domínio gerados, devem ser utilizados tipos de dados primitivos normalmente utilizados em linguagens de programação orientadas a objetos. A **Tabela 4** apresenta os tipos de dados utilizados neste trabalho, bem como seu significado e respectivo tipo de dados no diagrama de classes de domínio.

**Tabela 4** — Domínio dos tipos de dados usados para construção dos diagramas de classes UML.

Símbolo	Descrição	Tipo de dado no diagrama de classes
C	Caractere	<i>char</i>
T	Texto	<i>string</i>
D	Decimal	<i>float</i>
I	Inteiro	<i>int</i>
B	Booleano	<i>boolean</i>

#### 4.4 DOCUMENTO DE ESPECIFICAÇÃO DE REQUISITOS

Para este trabalho foi desenvolvida uma ferramenta de software que recebe como entrada o arquivo XPDL v2.2 que representa o modelo de processo de negócio, originalmente modelado em BPMN v2.0. Essa ferramenta, denominada SPRD (*Software Requirements from Process Definitions*), gera uma versão do documento de Especificação de Requisitos, que pode ser utilizada por analistas de sistemas durante o desenvolvimento de softwares.

No processo de desenvolvimento de software, é importante que a documentação de requisitos seja clara e concisa, de forma que seja compreensível

por equipes técnicas, clientes, usuários e outras partes interessadas. Assim, para a geração de um documento de especificação requisitos de software foi necessário definir uma estrutura padrão para o documento. A definição dessa estrutura foi baseada na norma ISO/IEC/IEEE/29148:2011 (ISO, 2011). Entretanto, as atividades realizadas até o momento ainda não cobrem todas as seções do documento de boas práticas. Isso acontece porque o documento apresentado neste trabalho é gerado por um processo automatizado, cuja fonte de informação é restrita a um modelo de processos de negócio BPMN. Contudo, o processo de extração de requisitos desenvolvido possibilitou a representação das seguintes informações:

- **Referências:** documentos referenciados pelo documento de requisitos, identificados por um título, data e autor;
- **Funções do produto:** principais funções do software, em formato textual ou gráfico, de forma que seja compreensível por qualquer pessoa que estiver lendo o documento pela primeira vez;
- **Restrições:** itens que de alguma maneira limitam a execução do software, tais como: normas regulatórias, requisitos de tempo de execução e considerações sobre segurança;
- **Premissas e dependências:** necessidades que caso não sejam atendidas demandam alteração nos requisitos;
- **Requisitos específicos:** apresenta todos os requisitos em um nível de detalhe suficiente para o trabalho das equipes de projeto e teste. Como exemplos, temos as interfaces externas, funções de validação, entidades e requisitos de desempenho, entre outros requisitos.

É importante mencionar que os modelos de processos de negócios podem conter informações que não são previstas no documento de Especificação

de Requisitos. Considerar essas informações requer um sistema mais complexo de gestão de requisitos, que não faz parte do escopo deste trabalho. Assim, foi feita a opção de estruturar um documento que contemplasse informações indicadas para o levantamento de requisitos segundo Pressman (2011), Sommerville (2011) e Yayici (2013). Desse modo, o documento de Especificação de Requisitos considerado neste trabalho apresenta uma lista de requisitos, categorizados por tipo, mantendo o conteúdo recomendado e selecionado do documento de boas práticas (ISO, 2011). Esse documento foi então estruturado da seguinte forma:

- **Controle de versionamento:** apresenta informações sobre o modelo de processos de negócio: nome da ferramenta de modelagem, autor, data de criação e data da modificação do modelo;
- **Referências:** apresenta os documentos utilizados como base para identificação dos requisitos de software. No caso deste trabalho, foi usado somente o documento XPDL obtido a partir do modelo de processo de negócio em BPMN, que contém a definição de processo do modelo de processo de negócio. Para isso, o documento de Especificação de Requisitos apresenta o nome do arquivo XPDL utilizado;
- **Características do software:** descreve de forma textual os atores, requisitos funcionais e requisitos não funcionais identificados. Essa seção do documento também apresenta a pré-condição e as pós-condições para execução dos casos de uso. Além disso, apresenta as regras de negócio, identificadas a partir da aplicação da técnica apresentada neste trabalho;
- **Modelos de requisitos:** apresenta o diagrama de caso de uso na notação UML, gerado a partir da identificação dos atores e casos

de uso. Além disso, apresenta o diagrama de classes de domínio do software, construído a partir da identificação das classes e seus respectivos atributos.

Frente ao exposto, a **Figura 5** ilustra a estrutura do documento de Especificação de Requisitos utilizado neste trabalho. Cada documento de Especificação de Requisitos é gerado a partir da aplicação da técnica sobre um modelo de processos de negócio em BPMN, que será apresentada na *subseção 4.2*.

**Figura 5** — Estrutura do documento de Especificação de Requisitos proposto.

Referência: <nome do arquivo XPDL>	
Versão do documento XPDL: <versão da especificação XPDL do arquivo>	
Ferramenta de modelagem: <nome da ferramenta de modelagem utilizada>	
CENÁRIO: <nome do cenário de casos de uso>	
Autor: <identificação do autor do modelo BPMN>	
Data de criação: <data de criação do modelo BPMN>	
Data de modificação: <data de modificação do modelo BPMN>	
Data de geração: <data de geração do documento de especificação de requisitos>	
1.	ATORES: <nome e descrição dos atores dos casos de uso>
2.	PRÉ-CONDIÇÕES: <pré-condições para execução dos casos de uso>
3.	PÓS-CONDIÇÕES: <pós-condições para execução dos casos de uso>
4.	REQUISITOS FUNCIONAIS: <identificador e descrição dos requisitos funcionais>
5.	REQUISITOS NÃO FUNCIONAIS: <identificador e descrição dos requisitos não funcionais>
6.	REGRAS DE NEGÓCIO: <identificador e descrição das regras de negócio>
7.	DIAGRAMA DE CLASSES DE DOMÍNIO: 
8.	DIAGRAMA DE CASOS DE USO: 

A partir desse modelo, são identificados os elementos relevantes para extração de requisitos de software, de acordo com as heurísticas que serão apresentadas na *subseção 4.2*. Em relação à gestão dos requisitos de software, buscou-se estabelecer uma relação entre os modelos de processos de

negócio e os requisitos de software extraídos. De forma textual, a rastreabilidade de características, fontes e dependências é registrada no próprio documento de Especificação de Requisitos.

Ao final do processo, o documento de Especificação de Requisitos gerado é analisado junto às partes interessadas, de forma a identificar necessidades de melhoria. Isso garante que os modelos apresentarão maior conformidade ao domínio de negócio da organização. Além disso, o documento de requisitos gerado a partir dos modelos de processos de negócio melhorados tende a apresentar maior conformidade em relação aos processos de negócio da organização.

## 5. AVALIAÇÃO DA SISTEMÁTICA PROPOSTA

Nesta seção, a sistemática apresenta neste trabalho é aplicada sobre três modelos de processos de negócio selecionados a partir de um repositório público (BIZAGI, 2015). Os modelos selecionados foram construídos na notação BPMN 2.0 e representam diversas áreas de negócio, com diferentes níveis de complexidade. Adicionalmente, a base de testes conta com modelos de processos de negócio em BPMN de processos de negócio resultantes da modelagem de processos de uma biblioteca universitária (PUCCI, 2016). Essa base de testes possibilita a aplicação da técnica em diferentes situações possíveis, com diferentes áreas de negócio, inclusive com ambientes comprovadamente reais - características esses importantes e avaliadas nos experimentos realizados.

De forma a sistematizar a execução dos experimentos realizados neste trabalho, cada um dos modelos foi submetido à aplicação das heurísticas de negócio HN1, HN2 e HN3. A aplicação dessas heurísticas contribui para o entendimento inicial do processo de negócio por meio da identificação dos eventos que ocorrem nas fases inicial e final do processo. Ainda, auxilia a compreensão do fluxo do processo por meio da análise das atividades e seus respectivos executores.

A revisão da documentação dos demais elementos dos processos de negócio possibilitou a aplicação das heurísticas HN4, HN5 e HN6. Isso permitiu a identificação dos requisitos funcionais, requisitos não funcionais e regras de negócio. Cabe ressaltar que a aplicação das heurísticas HN4, HN5 e HN6 envolveu uma análise criteriosa da documentação das atividades do processo, bem como do contexto do negócio de cada um dos processos.

### 5.1 RESULTADOS OBTIDOS

Em síntese, é importante ressaltar que, em termos gráficos, pouca coisa muda após a realização dos ajustes sobre os modelos de processos. O que pode mudar é a definição dos tipos de tarefas, com um símbolo sendo inserido na tarefa original. Para este trabalho, há interesse especial na representação nas atividades dos tipos “Usuário”, “Serviço” e “Script”. Essas atividades são selecionadas pela heurística HR3 para identificação dos requisitos funcionais e casos de uso. Por outro lado, em alguns casos, foi necessário acrescentar objetos e repositórios de dados, para representar formulários e informações persistidas durante a execução do processo. A **Figura 6** mostra parte do diagrama de casos de uso gerado.

**Figura 6** — Exemplo de diagrama de caso de uso gerado a partir do processo de negócio de contas a pagar.



O primeiro experimento usou um modelo de processos de negócio do setor de contas a pagar de uma organização. O processo consiste basicamente na validação da documentação enviada pelos fornecedores e posterior liberação de pagamento aos respectivos fornecedores. A **Figura 7** apresenta a versão ajustada desse modelo de negócios, após a aplicação das heurísticas de negócio apresentadas na *subseção 4.1*. As partes destacadas com um círculo mostram os elementos gráficos que foram ajustados ou adicionados ao modelo após os ajustes.

Com a aplicação da heurística de negócio HN6, foi possível verificar que a documentação dos fluxos de decisão deveria ser ajustada. Assim, a partir da análise do contexto do negócio e das atividades relacionadas a esses fluxos, foram registradas as regras de negócios do processo de contas a pagar.

Ainda, identificou-se a necessidade de representar os documentos manipulados durante a execução do processo, a saber: “Invoice” e “Purchase Order”. Mediante documentações das atividades do processo, foi possível listar os atributos de cada um desses documentos. Então, foram inseridos elementos repositórios de dados de acordo com as heurísticas de requisitos de identificação de classes. Da mesma forma, verificou-se a existência de “formulários de recebimento” e “formulários de aprovação da documentação”. As informações registradas nas atividades que manipulam esses formulários levaram à criação de objetos de dados que representam esses formulários, bem como seus atributos. A **Figura 8** apresenta parte do diagrama de classes de domínio gerado no documento de Especificação de Requisitos.

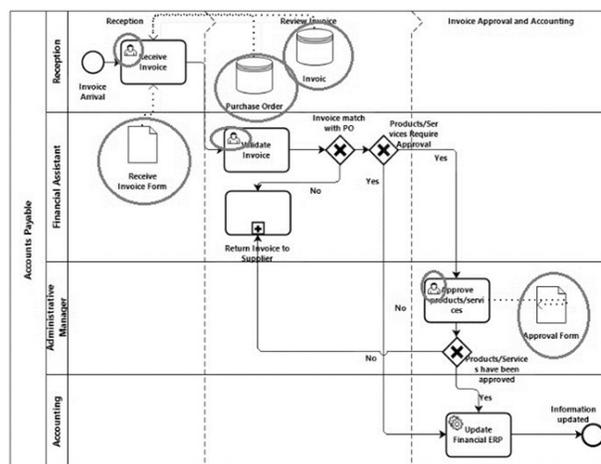
Com a aplicação da heurística de negócio HN6, foi possível verificar que a documentação dos fluxos de decisão deveria ser ajustada. Assim, a partir da análise do contexto do negócio e das atividades relacionadas a esses fluxos, foram registradas as regras de negócios do processo de contas a pagar. Essas regras foram registradas nos fluxos de

decisão e em atividades cuja execução é restrita pelas regras identificadas.

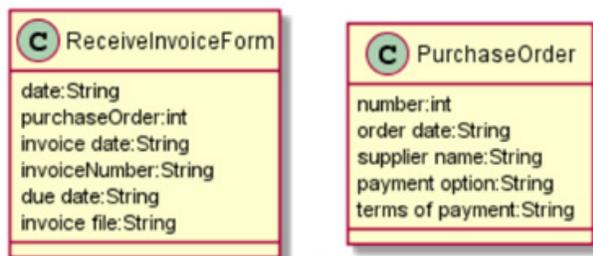
A aplicação da heurística de negócio HN4 envolveu a abstração dos requisitos não funcionais de qualidade do software para que fossem registrados na segunda versão dos modelos. Nesse sentido, foram registrados na segunda versão do modelo atributos de desempenho e usabilidade do software, de acordo com o contexto do processo de negócio de contas a pagar.

Outros dois experimentos foram realizados, em modelos de processos de negócio de uma organização de Tecnologia de Informação e de uma biblioteca universitária. De maneira geral, esses experimentos seguiram a mesma lógica de aplicação das heurísticas de negócio e de requisitos sobre as duas versões dos modelos. Ao final dos experimentos apresentados, os requisitos obtidos foram sumarizados por tipo, demonstrando de forma quantitativa os resultados obtidos em cada versão do modelo, conforme demonstrado na **Tabela 5**.

**Figura 7** — Modelo BPMN ajustado para o processo de negócio de contas a pagar.



**Figura 8** — Parte do diagrama de classes gerado a partir do processo de negócio de contas a pagar.



Os resultados mostraram, de forma geral, um aumento considerável na quantidade de requisitos identificados após a aplicação dos ajustes sobre os modelos. Considerando a identificação dos requisitos funcionais, verificou-se que a indicação do tipo de atividade tem um papel importante na melhoria dos processos, possibilitando a identificação das atividades que serão transformadas em casos de uso.

**Tabela 5** — Elementos de requisitos nas versões original e ajustada do modelo de negócio.

Versão dos modelos	Ator	Req, Não Func.	Reg. de Neg.	Reg. de Neg.	Classe de Domínio
Original	11	0	5	11	0
Ajustado	12	9	19	18	13

Cabe ressaltar que a aplicação das heurísticas de extração de requisitos não garante que os requisitos identificados tenham a semântica desejada para extração de requisitos. Essas heurísticas definem que a versão inicial do modelo deve ser ajustada, mas não há uma definição de como fazer esses ajustes. Assim, a aplicação das heurísticas depende, em grande parte, do conhecimento do negócio e da experiência do analista em identificar situações de melhoria nos processos de negócio.

## 6. CONCLUSÕES

O desenvolvimento de metodologias e técnicas que possibilitem a transformação automática de

modelos de negócios em modelo de requisitos de software contribuirá para padronizar, agilizar e melhorar as atividades de levantamento e especificação de requisitos e outros processos da Engenharia de Software. Nessa direção, é possível afirmar que este trabalho vem apoiar a concretização dessa transformação de modelos.

A utilização de um documento padrão estruturado de acordo com boas práticas de documentação de requisitos e a geração de diagramas UML de forma padronizada possibilitam enriquecer semanticamente a Especificação de Requisitos de software. Isso contribui para o melhor entendimento por parte das equipes de negócio e equipes de desenvolvimento. A geração de diagramas de classes de domínio possibilita visualizar de maneira padronizada os elementos manipulados durante a execução dos processos de negócio. Isso contribui para identificação de classes de domínio, que podem ser utilizadas por outras etapas do desenvolvimento do software, como projeto e testes.

Ao mesmo tempo, os ajustes realizados nos modelos de processos de negócio em BPMN contribuem para o enriquecimento da documentação do negócio da organização. Assim, a aplicação da sistemática proposta irá apresentar melhores resultados, com documentos de requisitos para a solução de software mais completos e aderentes às necessidades da organização.

Em trabalhos futuros, o conjunto de heurísticas proposto neste trabalho deve ser revisado, de forma a identificar a necessidade de ajustes que proporcionem melhores resultados. Para isso, novos experimentos devem ser realizados em modelos de processos de negócio voltados a diferentes áreas de negócio e com diferentes complexidades. Os resultados desses experimentos servirão como base para os ajustes necessários

## REFERÊNCIAS

BIZAGI. **BPMN process models & process apps**. 2015. Disponível em: <<http://www.bizagi.com/en/community/process-xchange>>. Acesso em: 10 ago. 2015.

BOUSETTA, B.; EL BEGGAR, O.; GADI, T. A methodology for CIM modelling and its transformation to PIM. **Journal of Information Engineering and Applications**, New York, v. 3, n. 2, p. 1-21, 2013.

CORREIA, A.; ABREU, F. B. Enhancing the correctness of BPMN models. In: VARAJÃO, J. E.; CRUZ-CUNHA, M. M.; MARTINHO, R. **Improving organizational effectiveness with Enterprise Information Systems**. Hershey: IGI Global, 2015. v.1, p. 241-261.

DIAS, F.; MORGADO, G.; OSCAR, P., SILVEIRA, D.; ALENCAR A.; LIMA, P.; SCHMITZ, E. Uma abordagem para a transformação automática do modelo de negócio em modelo de requisitos. In: WORKSHOP EM ENGENHARIA DE REQUISITOS, 6., 2006, Rio de Janeiro. **Anais...** Rio de Janeiro, 2006. p. 51-60.

ISO - INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC/IEEE/29148:2011. Systems and software engineering - Life cycle processes - Requirements Engineering. Technical report, **ISO/IEC/IEEE**, 2011.

JUNG, M.; KIM, H. S.; JO, M. H.; TAK, K. H.; CHA, H. S.; SON, J. H. Mapping from BPMN-formed business processes to XPD L business processes. In: INTERNATIONAL CONFERENCE ON ELECTRONIC BUSINESS, 4., 2004, Beijing. **Proceedings...** Beijing, 2004. p. 422-427.

MORA, B.; RUIZ, F.; GARCÍA, F.; PIATTINI, M. Experiencia en transformación de modelos de procesos de negocios desde BPMN a XPD L. In: CONGRESO IBEROAMERICANO EN SOFTWARE ENGINEERING, 10., 2007, Isla de Marguerita. **Proceedings...** Isla de Marguerita, 2007. p. 165-178.

OLIVEIRA, M. A. B.; CONTE, T.; MARCZAK, S.; SANTOS, D. V.; VIEIRA, S. R. C. Elicitação de requisitos a partir de modelos de processos de negócio e modelos organizacionais: uma pesquisa para definição de técnicas baseadas em heurísticas. In: REQUIREMENTS ENGINEERING @ BRAZIL, 2013, Rio de Janeiro. **Anais...** Rio de Janeiro, 2013. p. 86-91.

OMG - OBJECT MANAGEMENT GROUP. **Business Process Model and Notation Version 2.0**. Needham: Object Management Group, 2011. Disponível em: <<http://www.bpmn.org/>>. Acesso em: 02 maio 2014.

PRESSMAN, R. S. **Engenharia de Software: uma abordagem profissional**. 7 ed. Porto Alegre: McGraw Hill, 2011.

PUCCI, M. A. F. S. **Gerenciamento de processos de negócio voltado à Engenharia de Software, com ênfase em modelagem de processos**. Trabalho de Conclusão de Curso (Graduação em Ciência da Computação). Universidade Estadual Paulista Júlio de Mesquita Filho, Rio Claro, 2016.

SGANDERLA, K. **O que BPM tem a ver com requisitos de software? Tudo!** 2014. Disponível em: <<http://blog.iprocess.com.br/2014/07/o-que-bpm-tem-a-ver-com-requisitos-de-software-tudo/>>. Acesso em: 01 jun. de 2015.

SOMMERVILLE, I. **Engenharia de Software**. 9. ed. São Paulo: Pearson Prentice Hall, 2011.

VAN DER AALST, W. M. P. **Patterns and XPDL: a critical evaluation of the XML Process Definition Language**. QUT report FIT-TR-2003-06. Disponível em: <<http://bpmcenter.org/reports>>. Acesso em: 10 ago. 2015.

VIEIRA, S. R. C.; CONTE, T.; NASCIMENTO, R.; VIANA, D. Evaluating a technique for requirements extraction from Business Process Diagrams through empirical studies. In: CONFERENCIA LATINOAMERICANA EN INFORMATICA, 38., 2012, Medellin. **Anais...** Medellin, 2012. p. 245-254.

WESKE, M. **Business process management: concepts, languages, architectures**, 2. ed. Heidelberg: Springer, 2012.

WFMC - Workflow Management Coalition. **Process Definition Interface: XML Process Definition Language**. WfMC-TC-1025. Version 2.2. 2012. Disponível em: <<http://www.xpdl.org>>. Acesso em: 01 dez. 2015.

XAVIER, L.; ALENCAR, F.; CASTRO, J.; PIMENTEL, J. Integração de requisitos não-funcionais a processos de negócio: integrando BPMN e NFR. In: WORKSHOP ENGENHARIA DE REQUISITOS, 10., 2010, Cuenca. **Anais...** Cuenca, 2010. p. 29-50

YAYICI, E. **Business analyst's mentor book: with best practice business analysis techniques and software requirements management tips**. Istanbul: Emrah Yayici, 2013.