



UNIVERSIDADE FEDERAL DO ESTADO DO RIO DE JANEIRO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
ESCOLA DE INFORMÁTICA APLICADA

Implementação de Algoritmos para Melhoria de Processos de Negócio

Renata Emanuelle Vasconcellos Anhon

Orientadores:

Flávia Maria Santoro e Leonardo Guerreiro Azevedo (até 15 de agosto de 2018)
Rodrigo Pereira dos Santos (de 15 de agosto de 2018 até 06 de dezembro de 2018)

RIO DE JANEIRO, RJ – BRASIL

DEZEMBRO DE 2018

Catálogo informatizada pelo(a) autor(a)

A596i Anhon, Renata Emanuelle Vasconcellos
Implementação de Algoritmos para Melhoria de
Processos de Negócio / Renata Emanuelle Vasconcellos
Anhon. -- Rio de Janeiro, 2018.
113 f.

Orientador: Rodrigo Pereira dos Santos.
Trabalho de Conclusão de Curso (Graduação) -
Universidade Federal do Estado do Rio de Janeiro,
Graduação em Sistemas de Informação, 2018.

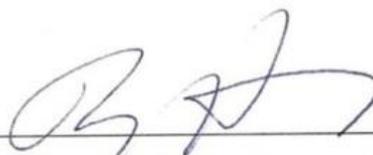
1. Algoritmos. 2. Processos de negócio. 3.
Redesenho. 4. Heurísticas. I. Santos, Rodrigo
Pereira dos, orient. II. Título.

Implementação de Algoritmos para Melhoria de Processos de Negócio

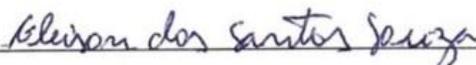
Renata Emanuelle Vasconcellos Anhon

Projeto de Graduação apresentado à Escola de
Informática Aplicada da Universidade Federal do
Estado do Rio de Janeiro (UNIRIO) para obtenção
do título de Bacharel em Sistemas de Informação.

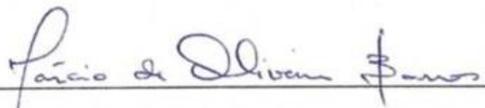
Aprovado por:



Rodrigo Pereira dos Santos (UNIRIO)



Gleison dos Santos Souza (UNIRIO)



Márcio de Oliveira Barros (UNIRIO)

RIO DE JANEIRO, RJ – BRASIL.

DEZEMBRO DE 2018

*“It's supposed to be hard.
If it wasn't hard, everyone would do it.”*

Jimmy Dugan, A League of Their Own

AGRADECIMENTOS

Em primeiro lugar agradeço à minha mãe, Elizabeth, por estar ao meu lado em todos os momentos, me apoiando e me dando o suporte necessário para que eu possa seguir em frente.

Ao meu pai, João, pelo incentivo e apoio sempre que precisei.

Agradeço ao Cayo por ser um companheiro compreensível e por ter ficado ao meu lado nos meus momentos mais difíceis.

Agradeço à minha irmã Sunny e à minha sobrinha Gabriela, por todo amor e carinho que me deram ao longo de toda minha vida.

Aos amigos que a UNIRIO me trouxe, em especial Paula, Bruna, Rodrigo e Igor pela parceria ao longo desses quatro anos, mesmo com tantos altos e baixos, sem nunca desistir da amizade.

Aos amigos do estágio, em especial Mariana e Treicy, que sempre tiveram ouvidos prontos pra me ouvir quando eu precisei.

Agradeço também aos professores Flávia e Leonardo por me permitirem participar desse projeto e por todo suporte que me foi dado.

Ao professor Rodrigo por ter aceitado seguir como meu orientador nesta fase final do projeto e por ter me ajudado em diversos momentos.

A todos que passaram pela minha vida e que de alguma forma contribuíram para que eu fosse quem sou hoje.

RESUMO

O gerenciamento de processos de negócio é fundamental para que uma organização alcance seus objetivos de forma eficiente. Diversos estudos apontam padrões para reestruturação de processos com o objetivo de alcançar uma melhoria dos resultados. Contudo, a análise para aplicação dos padrões não é simples. Uma sistematização auxilia na identificação dos processos nos quais podem ser aplicados padrões de redesenho. O presente trabalho tem como objetivo desenvolver uma aplicação que implementa algoritmos propostos através de heurísticas, que utilizam padrões de redesenho para identificação de oportunidades de melhoria em processos de negócio. Ao final, foi realizada uma avaliação que mostra que a aplicação desenvolvida apresenta os resultados esperados para melhoria em processos de negócio.

Palavras-chave: algoritmos, melhoria em processos de negócio, padrões de redesenho, implementação de heurísticas.

ABSTRACT

Business process management is critical for an organization to achieve its goals efficiently. Several studies indicate patterns for restructuring work systems in order to achieve better results. However, the analysis of standards is not simple. Systematization helps to identify the processes in which redesigning patterns can be applied. This work aims to develop an application that implements the algorithms proposed through heuristics, which use redesign patterns to identify opportunities for improvement in business processes. At the end, an evaluation was performed and results corroborate the expected results regarding business processes improvement.

Keywords: algorithms, improvement in business processes, redesign patterns, implementation of heuristics.

Sumário

| | |
|------------------------------------------------------------------------------|----|
| 1. Introdução..... | 12 |
| 1.1. Motivação | 12 |
| 1.2. Objetivos..... | 13 |
| 1.3. Organização do texto | 14 |
| 2. Conceitos Básicos..... | 15 |
| 2.1. Conceitos básicos BPM..... | 15 |
| 2.1.1. Redes de Petri (Petri Nets) | 16 |
| 2.2. Melhoria de processos | 18 |
| 2.3. Padrões de redesenho..... | 19 |
| 2.4. Trabalhos Relacionados..... | 21 |
| 2.5. Algoritmos para implementação do padrão de redesenho..... | 23 |
| 2.5.1. Composição de Atividades | 24 |
| 2.5.2. Aumento de Poder | 27 |
| 2.5.3. Ressequenciamento | 29 |
| 2.6. Considerações finais | 32 |
| 3. Implementação dos algoritmos para aplicação dos padrões de redesenho..... | 33 |
| 3.1. Descrição geral da solução | 33 |
| 3.2. Especificação da ferramenta..... | 33 |
| 3.2.1. Requisitos funcionais..... | 33 |
| 3.2.2. Diagrama de classes..... | 34 |
| 3.2.3. Diagrama de atividades | 35 |
| 3.3 Tecnologias utilizadas | 38 |
| 3.4 Funcionalidades | 39 |
| 3.4.1. Acessar o sistema | 39 |
| 3.4.2. Selecionar arquivo XML | 40 |
| 3.4.3. Visualizar resultados das heurísticas | 41 |

| | |
|------------------------------------------------------------------|----|
| 3.4.4. Visualizar detalhes das transições | 44 |
| 3.5. Considerações finais | 45 |
| 4. Avaliação | 46 |
| 4.1. Metodologia..... | 46 |
| 4.2. Modelos utilizados nos testes | 46 |
| 4.2.1 Modelo 1 | 46 |
| 4.2.2 Modelo 2..... | 48 |
| 4.2.3 Modelo 3..... | 50 |
| 4.3. Resultados Obtidos | 50 |
| 4.3.1. Algoritmo 1: Composição de atividades | 51 |
| 4.3.2. Algoritmo 2: Composição de atividades pelo analista | 54 |
| 4.3.3. Algoritmo 3: Aumento de poder..... | 58 |
| 4.3.4. Algoritmo 4: Transições independentes | 61 |
| 4.3.5. Algoritmo 5: Transições similares..... | 63 |
| 4.4 Considerações finais | 66 |
| 5. Conclusão | 67 |
| 5.1. Contribuições..... | 67 |
| 5.2. Limitações do projeto | 67 |
| 5.3. Trabalhos futuros | 68 |
| Referências Bibliográficas..... | 69 |
| Anexo 1 - Classes do Projeto..... | 72 |
| Anexo 2 - XML dos modelos | 89 |

Índice de Tabela

| | |
|----------------------------------------------------------------------------------------------|----|
| Tabela 1: Detalhes do modelo 1 (parte 1) | 47 |
| Tabela 2: Detalhes do modelo 2 | 49 |
| Tabela 3: Detalhes do modelo 3 | 50 |
| Tabela 4: Resultados obtidos manualmente, aplicando o algoritmo 1 ao modelo 1 | 51 |
| Tabela 5: Resultados obtidos manualmente, aplicando o algoritmo 1 ao modelo 2 | 52 |
| Tabela 6: Resultados obtidos manualmente, aplicando o algoritmo 1 ao modelo 3 | 53 |
| Tabela 7: Resultados obtidos manualmente, aplicando o algoritmo 2 ao modelo 1 | 55 |
| Tabela 8: Resultados obtidos manualmente, aplicando o algoritmo 2 ao modelo 2 | 56 |
| Tabela 9: Resultados obtidos manualmente, aplicando o algoritmo 2 ao modelo 3 (parte 2)..... | 58 |
| Tabela 10: Resultados obtidos manualmente, aplicando o algoritmo 3 ao modelo 1 | 59 |
| Tabela 11: Resultados obtidos manualmente, aplicando o algoritmo 3 ao modelo 2 | 59 |
| Tabela 12: Resultados obtidos manualmente, aplicando o algoritmo 3 ao modelo 3 | 60 |
| Tabela 13: Resultados obtidos manualmente, aplicando o algoritmo 4 ao modelo 1 | 61 |
| Tabela 14: Resultados obtidos manualmente, aplicando o algoritmo 4 ao modelo 2 | 62 |
| Tabela 15: Resultados obtidos manualmente, aplicando o algoritmo 4 ao modelo 3 | 62 |
| Tabela 16: Resultados obtidos manualmente, aplicando o algoritmo 5 ao modelo 2 | 64 |
| Tabela 17: Resultados obtidos manualmente, aplicando o algoritmo 5 ao modelo 3 | 65 |

Índice de Figuras

| | |
|------------------------------------------------------------------------------------------------------------------------------------------------|----|
| Figura 1: Principais elementos das redes de Petri (RAEDTS et al., 2007) | 17 |
| Figura 2: Modelo em alto nível (SOUZA, 2013) | 18 |
| Figura 3: Algoritmo 1 – Algoritmo para encontrar atividades finas e largas com <i>boxplot</i> (SOUZA, 2017) | 25 |
| Figura 4: Algoritmo 2 – Algoritmo para encontrar as atividades finas e largas usando a definição do analista (SOUZA, 2017) | 26 |
| Figura 5: Algoritmo 3 – Algoritmo para encontrar transições que podem ter aumento de poder (SOUZA, 2017) | 28 |
| Figura 6: Algoritmo 4 – Algoritmo para encontrar transições independentes (SOUZA, 2017)..... | 30 |
| Figura 7: Algoritmo 5 – Algoritmo para ordenar as duplas de atividades de acordo com o percentual de palavras coincidentes (SOUZA, 2017) | 31 |
| Figura 8: Diagrama de classes | 34 |
| Figura 9: Diagrama de atividades..... | 35 |
| Figura 10: Exemplo com todas as <i>tags</i> que o sistema reconhece..... | 36 |
| Figura 11: Elementos que podem se repetir | 37 |
| Figura 12: Tela inicial..... | 39 |
| Figura 13: Buscar arquivo XML no computador | 40 |
| Figura 14: Mensagens de Alerta..... | 40 |
| Figura 15: Tela de resultados..... | 41 |
| Figura 16: Tela de resultados Composição de Atividades – <i>boxplot</i> padrão..... | 42 |
| Figura 17: Tela de resultados Composição de Atividades - Definido pelo analista..... | 42 |
| Figura 18: Tela de resultados Aumento de Poder..... | 43 |
| Figura 19: Tela de resultados Ressequenciamento - Transições Independentes e Similares | 44 |
| Figura 20: Tela detalhes das atividades (transições) | 44 |
| Figura 21: Modelo 1 | 47 |
| Figura 22: Modelo 2 | 48 |
| Figura 23: Modelo 3 | 50 |
| Figura 24: Resultados obtidos com a ferramenta, aplicando o algoritmo 1 ao modelo 1 | 52 |

| | |
|-------------------------------------------------------------------------------------|----|
| Figura 25: Resultados obtidos com a ferramenta, aplicando o algoritmo 1 ao modelo 2 | 53 |
| Figura 26: Resultados obtidos com a ferramenta, aplicando o algoritmo 1 ao modelo 3 | 54 |
| Figura 27: Resultados obtidos com a ferramenta, aplicando o algoritmo 2 ao modelo 1 | 55 |
| Figura 28: Resultados obtidos com a ferramenta, aplicando o algoritmo 2 ao modelo 2 | 57 |
| Figura 29: Resultados obtidos com a ferramenta, aplicando o algoritmo 2 ao modelo 3 | 58 |
| Figura 30: Resultados obtidos com a ferramenta, aplicando o algoritmo 3 ao modelo 1 | 59 |
| Figura 31: Resultados obtidos com a ferramenta, aplicando o algoritmo 3 ao modelo 2 | 60 |
| Figura 32: Resultados obtidos com a ferramenta, aplicando o algoritmo 3 ao modelo 3 | 60 |
| Figura 33: Resultados obtidos com a ferramenta, aplicando o algoritmo 4 ao modelo 1 | 61 |
| Figura 34: Resultados obtidos com a ferramenta, aplicando o algoritmo 4 ao modelo 2 | 62 |
| Figura 35: Resultados obtidos com a ferramenta, aplicando o algoritmo 4 ao modelo 3 | 63 |
| Figura 36: Resultados obtidos com a ferramenta, aplicando o algoritmo 5 ao modelo 2 | 64 |
| Figura 37: Resultados obtidos com a ferramenta, aplicando o algoritmo 5 ao modelo 3 | 65 |

1. Introdução

1.1. Motivação

De acordo com Dumas (2013), o conceito de Gestão de Processos de Negócios pode ser definido como o conjunto de instrumentos e técnicas que visam à gestão sistemática e integrada de uma organização, envolvendo os atores de cada processo de negócio, com o objetivo de alcançar maior qualidade, produtividade e lucratividade. Segundo Weske (2012) e Thom et al. (2007), trata-se de uma abordagem de gerenciamento que sistematiza e facilita os mais complexos processos organizacionais, tanto dentro como fora das empresas. Para isso, são utilizadas as melhores práticas de gestão e controle das metodologias, mapeamento de processos, modelagem e otimização das atividades que envolvem recursos humanos, documentos e outras fontes de informação para melhoria e transformação contínua dos processos, atingindo assim os resultados esperados.

Os processos de negócios apresentam características próprias inerentes a cada organização. Para Souza (2013), é necessário dar atenção à modelagem desses processos, explorando os motivos e objetivos que motivam os diversos elementos de uma organização. A Modelagem de Processos de Negócios é uma atividade que analisa os componentes dos processos que fazem parte da cadeia funcional do negócio, apresentando fontes de informação relevantes para um processo.

Um processo é, portanto, uma ordenação específica das atividades de trabalho no tempo e no espaço, com uma estrutura de ação que consiste em um começo, um fim, e entradas e saídas claramente identificadas. Esse elemento estrutural dos processos é a chave para a obtenção das vantagens da engenharia de processos (Davenport, 1993).

Os processos têm características intrínsecas, tais como custo, prazos, qualidade de produção, satisfação do cliente etc. Quando uma estratégia pré-definida é atingida, ocorre uma melhoria no processo de negócios, que pode se refletir na redução do custo ou no aumento da satisfação do cliente (Gonçalves, 2000). Por sua vez, a modelagem de

processos tem por principal objetivo representá-los de maneira clara e formal em diferentes níveis de abstração (Serrano, 1997). A disponibilidade de modelos elaborados desta forma permite uma análise crítica das atividades existentes para definir melhorias e racionalização dos processos.

O gerenciamento de processos de negócio é fundamental para que uma organização alcance seus objetivos de forma eficiente. Diversos estudos apontam padrões para reestruturação de processos com o objetivo de alcançar uma melhoria dos resultados. Contudo, a análise para aplicação dos padrões não é simples. Uma sistematização auxilia na identificação dos processos nos quais podem ser aplicados padrões de redesenho.

1.2. Objetivos

Este trabalho tem como objetivo a implementação dos algoritmos propostos por Souza et al. (2017) para melhoria de processos de negócios, em uma ferramenta para automatizar o gerenciamento de processos na busca por pontos de melhoria, diminuindo o esforço gasto pelo analista.

Souza (2013) apresentou heurísticas baseadas em padrões de redesenho de modo a identificar oportunidades de melhoria em modelos de processos de negócios. A partir dessas heurísticas, Souza et al. (2017) desenvolveram algoritmos para facilitar a automatização da sua aplicação. Contudo, os algoritmos foram desenvolvidos em pseudocódigos – algoritmos descritos de forma genérica e em linguagem simples –, que não dispõem de uma linguagem de programação específica, impossibilitando a entrada e saída de dados de forma automatizada.

Sendo assim, a proposta deste trabalho consiste em implementar uma ferramenta automatizada com base nos algoritmos propostos por Souza et al. (2017), que poderá apontar de maneira mais rápida e automática os possíveis pontos de melhoria. A meta é garantir ao analista a verificação e exploração desses resultados, decidindo a melhor maneira de aplicá-los em cada processo.

1.3. Organização do texto

O presente trabalho está estruturado em capítulos e, além desta introdução, será desenvolvido da seguinte forma:

Capítulo 2: Conceitos básicos acerca de Gerenciamento de Processo de Negócio, Petri Net, Melhoria de Processo, Padrões de Redesenho e Algoritmos Relacionados.

Capítulo 3: Implementação dos algoritmos propostos através de uma ferramenta que aplica os algoritmos, fornecendo como saída possíveis pontos de melhoria que auxiliam na análise do processo e na tomada de decisões.

Capítulo 4: Avaliação dos resultados pela abordagem manual e pela abordagem automatizada pela ferramenta, comparando os resultados.

Capítulo 5: Conclusão, contemplando as considerações finais, assinalando as contribuições da pesquisa e sugerindo possibilidades de aprofundamento posterior.

2. Conceitos Básicos

Este capítulo apresenta os conceitos básicos necessários para o entendimento do trabalho.

A Seção 2.1 trata sobre Gerenciamento de Processos de Negócio (*Business Process Management* - BPM) e apresenta o modelo que será utilizado como base para os algoritmos Redes de Petri (Petri Nets).

A Seção 2.2 introduz o conceito de Melhoria de Processos de Negócio.

A Seção 2.3 apresenta os padrões de redesenho propostos por REIJERS e MANSAR (2005).

A Seção 2.4 apresenta outros trabalhos relacionados ao tema.

Por fim, a Seção 2.5 traz os algoritmos que serão implementados no Capítulo 3.

2.1. Conceitos básicos BPM

“Gerenciamento de Processos de Negócio (Business Process Management - BPM) consiste na ciência de supervisionar como o trabalho é realizado em uma organização para garantir resultados consistentes e aproveitar as oportunidades de melhoria” (DUMAS, 2013, p.1).

Business Process Management (BPM) – traduzido por Gestão de Processos de Negócios – pode ser definido como uma combinação de instrumentos e técnicas visando à gestão sistemática e integrada de uma organização, em conjunto com os atores de cada processo de negócio, em busca de maior qualidade, produtividade e lucratividade. Trata-se de uma abordagem de gerenciamento que sistematiza e facilita os mais complexos processos organizacionais, tanto dentro como fora das empresas. Para isso, é necessário lançar mão das melhores práticas de gestão e controle das metodologias, ferramentas e técnicas de melhoria de análise, mapeamento de processos, modelagem, publicação e otimização das atividades que envolvem recursos humanos, documentos e outras fontes

de informação para melhoria e transformação contínua dos processos, atingindo assim os resultados esperados.

Processo de negócio é um conjunto de atividades coordenadas com o objetivo de atingir um determinado objetivo. Assim, através da execução de processos de negócio as organizações realizam seus propósitos (WESKE, 2012; THOM et al., 2007). Já a modelagem de processos de negócio apresenta as metodologias e práticas empregadas para representar visualmente todas as características de um processo de negócio. Modelo é um mecanismo utilizado pelas empresas que serve de documento ou simulação para compartilhar, implementar e avaliar suas operações para uma melhora contínua. Desse modo, a modelagem dos processos permite responder às questões relacionadas aos negócios (Onde? Quando? Quem? Por quê? O quê? Como?) a partir de um determinado contexto, descrevendo em detalhes um dado processo de negócio com informações sobre fluxo de atividades, papéis executores, regras de negócio, portadores de informação, entradas e saídas, e várias outras relacionadas ao processo.

Segundo Souza (2013), um modelo de processo descreve detalhadamente um processo de negócio de uma organização. Com ele, é possível documentar, simular, implementar, analisar, otimizar e controlar as operações. Este recurso contém informações como fluxo de atividades, papéis executores, entrada e saídas, entre outras. Para atingir seus objetivos, as instituições utilizam os modelos para gerir, controlar e continuamente melhorar suas operações.

Para elaboração destes modelos de processos, são utilizadas diversas notações, como as Redes de Petri, isto é, um modelo de representação precisa e formal dos processos de negócio (DESEL, 2005). Este trabalho utiliza representação em Redes de Petri (*Petri Nets*), por ser um modelo formal matemático simples, que será apresentado na seção a seguir.

2.1.1. Redes de Petri (Petri Nets)

A Rede de Petri (Petri Net) é um grafo direcionado com dois tipos diferentes de nós: posições, representadas por círculos (ou elipses) e transições por retângulos (ou barras). As redes de Petri são bipartidas, ou seja, nenhum arco conecta duas posições ou

duas transições, como mostra a Figura 1. Além disso, posições, transições e arcos podem conter diversas anotações, como atributos. Essa representação é utilizada em aplicações computacionais, pois com ela é possível aplicar técnicas de simulação e análise, além de possuir algumas propriedades matemáticas, como alcançabilidade. É uma representação mais simples e mais abrangente, se comparada a outras, que permite analisar um modelo de um ponto de vista matemático. Ademais, os algoritmos que serão implementados foram baseados neste modelo.

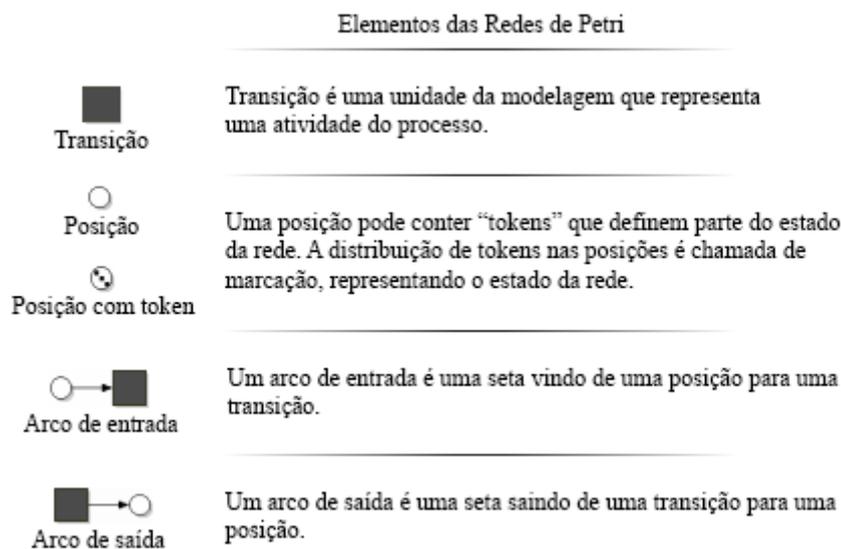


Figura 1: Principais elementos das redes de Petri (RAEDTS et al., 2007)

Souza et al. (2017) mencionam que dependendo do uso pretendido do modelo de processo, é preciso que este contenha informações específicas. Esta informação pode ser apresentada em um modelo de alto nível, como mostrado na Figura 2. Neste exemplo, para cada transição, são descritos o nome da transição, a descrição, os parâmetros de entrada e saída, o papel responsável pela execução e o tempo de processamento. Este tempo pode ser obtido das instâncias do processo e deve estar disponível no modelo, por exemplo, armazenado como um atributo. O tempo de processamento é uma informação dinâmica: quando as novas instâncias de processo forem executadas, o tempo deve ser atualizado.

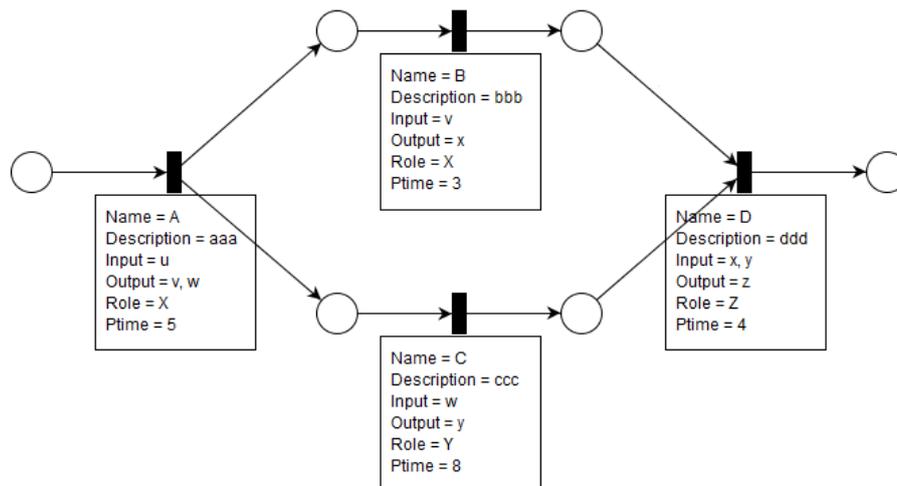


Figura 2: Modelo em alto nível (SOUZA, 2013)

2.2. Melhoria de processos

A Melhoria de Processos de Negócios aborda a necessidade contínua de revisão de operações com o objetivo de solucionar problemas, racionalizar e aumentar a produtividade, dentre outros agentes, buscando a otimização do processo. De acordo com Forster (2006), melhoria de processos é uma abordagem sistemática que ajuda as organizações a melhorarem seus processos e assim atingir resultados melhores. Segundo a pesquisa de Gartner, as empresas consideram como maior prioridade a melhoria de seus processos (GARTNER, 2009).

Apesar de existirem muitas abordagens e métodos, muitas delas não utilizam de maneira concreta o redesenho do processo. No final, não fica claro como as alternativas de processo, os chamados processos “to-be”, foram derivados dos processos correntes ou “as-is” (NETJES, 2010). Desse modo, Reijers e Mansar (2005) propõem uma abordagem para melhoria de processos baseada em padrões (boas práticas) de redesenho de processos que mostram de forma clara como criar as alternativas de processo, ou seja, como ir de um modelo as-is para o to-be. Reijers e Mansar (2005) afirmam que, durante anos, essas boas práticas foram desenvolvidas para melhoria de processos, expressas como padrões de redesenho de processos, que descrevem a melhor maneira de tratar um problema particular, podendo ser aplicados a situações similares.

Este trabalho se concentra na utilização destes padrões de redesenho (*redesign patterns*) como uma abordagem focada na melhoria de processos de negócio. Foram utilizados como base os padrões apresentados por Reijers e Mansar (2005) detalhados na próxima seção.

2.3. Padrões de redesenho

Reijers e Mansar (2005) propõem uma abordagem com base em padrões (boas práticas) de redesenho para melhoria de processos e demonstram de maneira simples a criação de alternativas de processo, ou seja, como ir do *as-is* para o *to-be*. Os autores afirmam ainda que estas boas práticas foram utilizadas para redesenhar processos, descrevendo uma solução que atenda não só a uma situação específica como também a problemas similares, apresentando suas características, o impacto que cada um causa no processo e algumas situações comuns que favorecem sua aplicação.

Mansar et al. (2009), Netjes et al. (2009), Agrahari et al. (2010) e Netjes et al. (2010) apresentaram mecanismos de seleção de padrões específicos para aplicação em um processo de negócio. Não há indicações da parte onde cada padrão pode ser aplicado, sendo responsabilidade do analista do processo a identificação dos pontos no processo para aplicação dos padrões.

As métricas aplicadas nestes trabalhos podem ajudar a apontar os padrões mais corretos a serem utilizados. Esses padrões são baseados em perspectivas de tempo, custo, qualidade e flexibilidade.

A perspectiva de tempo diz respeito ao tempo gasto para se levar um caso (instância do processo) do início ao fim. Existem muitas formas de abordar esta perspectiva, como, por exemplo, tentar diminuir o tempo médio gasto no processo ou o tempo máximo, ou até mesmo a variação do tempo entre casos.

A perspectiva de custo está relacionada à questão financeira. O foco de melhoria comumente é reduzir custos operacionais, principalmente em recursos humanos. A automação é vista como uma alternativa. Contudo, outros custos estão envolvidos, como o desenvolvimento e manutenção de aplicações de suporte ao processo.

A perspectiva de qualidade geralmente está associada à qualidade interna e externa. A qualidade interna está ligada a condições e controle do trabalho. A qualidade externa diz respeito à satisfação do cliente em relação ao produto e ao processo. Perspectivas que buscam melhorar custo, em geral, pioram a qualidade (SOUZA et al., 2017).

A perspectiva de flexibilidade pode ser definida como a capacidade de reagir a mudanças. Como exemplos, o gerenciamento de mudanças na estrutura do processo e alocação de recursos e mudanças na estrutura e responsabilidade dos processos para atingir necessidades do mercado e dos parceiros de negócio.

Reijers e Mansar (2005) propuseram padrões de redesenho relacionados às perspectivas. Os analistas podem usar a lista de padrões de redesenho para avaliar e melhorar um processo de negócio existente. A seguir, são apresentados três padrões de redesenho para ilustrar a proposta.

- O **padrão de composição de atividades** propõe combinar pequenas tarefas (ou seja, tarefas com poucos procedimentos) em uma única tarefa, ou dividir uma tarefa grande (ou seja, tarefa com muitos procedimentos) em tarefas menores;
- O **padrão de redesenho de aumento de poder** propõe dar autoridade aos tomadores de decisão e a reduzir a burocracia gerencial;
- O **padrão de redesenho de ressequenciamento** propõe mover tarefas para locais mais apropriados no modelo de processo de negócio. Por exemplo, a utilização do padrão pode sugerir a movimentação de uma tarefa para estar próxima a tarefas semelhantes ou adiá-la para evitar execuções desnecessárias, já que a reordenação não resulta em conflitos.

Esses padrões são detalhados mais adiante na apresentação dos algoritmos.

2.4. Trabalhos Relacionados

Esta seção apresenta trabalhos relacionados ao projeto, que fornecem abordagens de melhoria de processos de negócio usando os padrões de redesenho propostos por Reijers e Mansar (2005).

Netjes et al. (2009) propõem o redesenho de modelos de processos de negócio por meio das seguintes etapas: i) Computar medidas de processo; ii) Avaliar condições para encontrar as melhores práticas aplicáveis; iii) Criar modelos alternativos baseados nas melhores práticas selecionadas; e iv) Avaliar o desempenho das alternativas criadas. Assim, o resultado final é um novo modelo de processo de negócio. As medidas de processo calculadas na etapa (i) representam as características globais do processo. Na etapa (ii), as medidas de processo são combinadas na declaração de condição para cada padrão de redesenho. Quando uma condição é verdadeira, sugere a aplicação do padrão de redesenho relacionado. Na etapa (iii), os padrões selecionados de redesenho são usados para criar modelos alternativos. Finalmente, na etapa (iv), os desempenhos das alternativas criadas são avaliados e a melhor alternativa é selecionada para ser implementada como o novo processo de negócios. Este trabalho identifica quais padrões de redesenho são favoráveis para serem aplicados em um processo de negócio. No entanto, ele não indica partes específicas do processo em que um padrão de redesenho poderia ser aplicado.

Netjes et al. (2010) propõem uma ferramenta chamada PrICE toolkit (*Process Improvement by Creating and Evaluating* - Melhoria de Processos por Criação e Avaliação) para auxiliar o analista na criação de alternativas de processo. A ferramenta é baseada na arquitetura ProM, um *framework* extensível que suporta uma grande variedade de técnicas de mineração de processos em forma de plug-ins, que são: (i) O *plug-in* de importação lê um tipo de modelo específico em um formato de entrada específico; (ii) O *plug-in* de exportação escolhe um objeto do *object pool* (conjunto de objetos de modelos de processo inicializados) e os exporta em um formato de saída específico; (iii) O *plug-in* de conversão transforma um modelo de tipo específico (p. ex.: EPC) em outro tipo de modelo (p. ex.: Rede de Petri); (iv) O *plug-in* de filtro de *log* transforma um *log* de eventos em outro *log* de eventos; (v) O *plug-in* de mineração

aplica técnicas de mineração de processo em um *log* de eventos com o objetivo de, por exemplo, identificar discrepâncias entre o modelo existente e o que está contido nos logs; e (vi) O *plug-in* de análise usa um modelo do *object pool* como entrada e, usando parâmetros configurados pelo usuário, retorna um conjunto de objetos e o mostra ao usuário.

Com apoio da PRICE Toolkit, o analista seleciona as operações de redesenho (baseado nos padrões de redesenho) e, em seguida, seleciona uma parte do modelo para redesenho clicando nas atividades no modelo de processo. Cores são utilizadas para auxiliar o usuário e exibir que atividades podem ser adicionadas à seleção atual de uma parte do processo. Desta forma, é garantido que a entrada (parte do processo) para a criação de uma alternativa de processo está correta. Por fim, uma alternativa do processo é criada e o usuário pode simular e avaliar a alternativa criada (SOUZA, 2010).

Mansar et al. (2009) propõem um método, apoiado por uma ferramenta, para ajudar os analistas a selecionar os padrões de redesenho mais apropriados para melhorar um processo. A abordagem visa reduzir o tempo gasto pelos analistas, discutindo a utilidade dos padrões de redesenho e fornecendo-lhes uma avaliação clara da importância e impacto dos padrões de redesenho. A avaliação dos padrões de redesenho segue os critérios: i) popularidade do padrão de redesenho; ii) componente do redesenho considerando o cliente, informação, produto, visão de operação, visão de comportamento, organização e tecnologia; iii) impacto do padrão de redesenho nos processos considerando tempo, custo, qualidade e flexibilidade; iv) redesenhar metas considerando redução de tempo, redução de custos, melhoria de qualidade, melhoria de produtividade, aumento de receita, melhoria de atendimento ao cliente, melhoria de capacidades de TI e melhoria de competitividade; e v) redesenho de riscos.

Com base nesses cinco critérios, foram propostos indicadores para mensurar cada critério. Assim, para cada padrão de redesenho, cada critério recebe um valor indicador. Finalmente, os padrões de redesenho são ordenados de acordo com os resultados dos indicadores. A lista classificada de padrões de redesenho indica os padrões de redesenho mais apropriados a serem aplicados para o redesenho de

processos. No entanto, nenhuma parte específica do processo em que um padrão de redesenho seria aplicado é indicada.

Tsoury et al. (2016) afirmam que mudanças de dados podem afetar grande parte de um processo de negócios. Os autores propõem um conjunto de definições (primitivas) que ofereça suporte à análise dos efeitos dos elementos de dados em outros elementos do processo de negócios (atividades, restrições de roteamento e outros elementos de dados). Os autores afirmam que a pesquisa sobre padrões de mudança de processos, como Reijers e Mansar (2005), não aborda especificamente aspectos de dados. Embora não diretamente relacionado, o trabalho de Tsoury et al. (2016) poderia apoiar a proposta deste trabalho na formalização de aspectos relacionados aos dados de entrada/saída das atividades.

A análise e comparação dos trabalhos relacionados com a proposta apresentada neste trabalho foram feitos numa perspectiva qualitativa, uma vez que, embora todas as propostas tratem dos mesmos problemas, elas diferem em termos de execução e objetivos. A principal observação que pode ser feita diz respeito ao fato de que nenhuma das propostas na literatura aborda como decidir em quais pontos de um modelo de processo os padrões de redesenho devem ser aplicados. Assim, nossa principal contribuição é complementar a pesquisa sobre melhoria de processos com base em padrões que fornecem suporte para a tomada de decisão do analista.

2.5. Algoritmos para implementação do padrão de redesenho

O trabalho de Souza et al. (2013) propõe um conjunto de heurísticas que podem servir de base para o reconhecimento de chances para um processo de negócio aperfeiçoado, identificando como poderiam ser aplicados os padrões de redesenho de processos abordados por Reijers e Mansar (2005). Ao invés de apresentar ganho de tempo com a sistematização da identificação de melhoria, o trabalho objetiva provar que a sistematização chega a resultados confiáveis.

As heurísticas servem para automatizar a identificação de partes do processo de negócios para aplicar padrões de redesenho. Para cada heurística, existe uma lógica por trás de sua definição e de sua formalização.

Com base nessas heurísticas, Souza et al. (2017) propõem os algoritmos para implementação de padrões de redesenho, apresentados nas seções a seguir.

2.5.1. Composição de Atividades

O padrão de composição de atividades visa unificar pequenas tarefas em grandes tarefas ou dividir grandes tarefas em pequenas tarefas viáveis. Então, a questão é a classificação de tarefas em “pequenas” (finas) e “grandes” (grossas).

Conforme argumentado por Buzacott (1996) e Seidmann e Sundararajan (1997), quanto maior o tempo de processamento, maior será a tarefa. Por outro lado, quanto menor o tempo de processamento, menor a tarefa.

Utilizando o tempo de processamento como parâmetro para definir o tamanho das atividades, Souza et al. (2017) propuseram dois algoritmos: (i) levando em consideração a aplicação de *boxplot* (MCGILL et al., 1978) para identificar as atividades finas (baixo tempo de processamento) e grossas (alto tempo de processamento); e (ii) considerando a experiência do analista para definir a partir de qual distância do tempo médio de processamento uma atividade é identificada como fina ou grossa.

2.5.1.1. Proposta 1

Esta proposta é baseada na ferramenta gráfica utilizada na estatística descritiva chamada *boxplot*, visando identificar pequenas e grandes tarefas considerando o tempo de processamento destas tarefas. O *boxplot* apresenta diferenças entre as populações. É uma técnica matemática onde o espaçamento entre as regiões interquartílicas indica a dispersão e a assimetria nos dados e, assim, identifica valores muito discrepantes dos demais. Um quartil é qualquer um dos três valores que dividem o conjunto de dados ordenados em quatro partes iguais, correspondendo a 25% da amostra ou da população (MCGILL et al., 1978).

A limitação desta proposta ocorre quando a variação dos tempos de processamento é pequena. Nesse caso, o aplicativo heurístico poderia resultar em

nenhuma tarefa identificada como grande ou pequena, já que todas as tarefas estariam localizadas entre quartis, mesmo se todas as tarefas fossem grandes ou todas as tarefas fossem pequenas. A Figura 3 apresenta este algoritmo.

```

1  Algorithm 1: getSmallandLargeTasks(PT: Array)
2
3  // PT is an array of transitions and corresponding processing time (pt),
4  // i.e., each element pt of the array has a transition and a processingTime attribute
5
6  Declare P: Array, Po: Array
7  // P is computed as all processing time of the transitions in PT
8  // Po is computed as the ordered list of processing time of P
9
10 Declare ST, LT:Array
11 //ST and LT are the arrays of small and large transitions returned by the algorithm
12
13 Declare tp, q1, q2, q3, iqr:Number
14 //tp (transition element), q1 (lower interquartil),
15 //q2 (median), q3 (upper interquartil), iqr (interquartil range)
16
17 Begin
18     // getOrderTransitionsByItsProcessingTime returns PT elements
19     // ordered by their processing time
20     for each pt in PT:
21         P.add(pt.processingTime)
22     Po = getOrderArray(P)
23     // get the processing time corresponding to the lower quartil
24     q1 = Po[(length(Po)+1)*1/4]
25     // getMedian calculate the median element in the ordered array
26     q2 = getMedian(Po)
27     //get the processing time corresponding to the upper quartile
28     q3 = Po[(length(Po)+1)*3/4]
29     //calculate interquartile range
30     iqr = q3 - q1
31     //build small and large transition's arrays
32     for each pt in PT do
33         //build array of small transitions
34         If (pt.processingTime < (q1 - 1.5*iqr))
35             ST.add(pt)
36         //build large transition's array as formula 8
37         else
38             If (pt.processingTime > (q3 + 1.5*iqr))
39                 LT.add(pt)
40     //return small and large transition's arrays
41     return { ST, LT }
42 End

```

Figura 3: Algoritmo 1 – Algoritmo para encontrar atividades finas e largas com *boxplot* (SOUZA, 2017)

2.5.1.2. Proposta 2

Essa proposta baseia-se na experiência do analista para determinar a distância da mediana dos tempos de processamento a ser usada como definição para classificar

tarefas como pequenas ou grandes. A mediana foi escolhida como parâmetro, uma vez que é uma tendência central. Ela é usada para tentar evitar valores discrepantes, o que poderia afetar se usássemos a média.

Uma limitação desta heurística pode ocorrer se a variação dos tempos de processamento for pequena. Neste caso, a aplicação da heurística poderia resultar em nenhuma tarefa identificada como grande ou pequena, já que todas as tarefas estariam localizadas entre a mediana e a distância d definida pelo analista – mais formalmente, todos os valores entre $(1 - d) * m$ e $(1 + d) * m$, mesmo que todas as tarefas sejam grandes ou todas as tarefas sejam pequenas. A Figura 4 apresenta este algoritmo.

```

1  Algorithm 2: getSmallandLargeTasks(PT: Array)
2
3  // PT is an array of transitions and corresponding processing time (pt),
4  // i.e., each element pt of the array has a transition and a processingTime attribute
5
6  Declare P: Array, Po: Array
7  // P is computed as all processing time of the transitions in PT
8  // Po is computed as the ordered list of processing time of P
9
10 Declare ST, LT:Array
11 //ST and LT are the arrays of small and large transitions returned by the algorithm
12
13 Declare tp, m:Number
14 //tp (transition element), m (median), d (distance percentage parameter)
15
16 Begin
17   // getOrderTransitionsByItsProcessingTime returns PT elements
18   // ordered by their processing time
19   for each pt in PT:
20     P.add(pt.processingTime)
21   Po = getOrderArray(P)
22   // getMedian calculate the median element in the ordered array
23   m = getMedian(Po)
24   //get from user the distance percentage parameter
25   Read d
26   //build small and large transition's arrays
27   for each pt in PT do
28     //build array of small transitions
29     If (pt.processingTime < ((1-d)*m))
30       ST.add(pt)
31     //build large transition's array as formula 8
32     else
33       If (pt.processingTime > ((1+d)*m))
34         LT.add(pt)
35   //return small and large transition's arrays
36   return { ST, LT }
37 End

```

Figura 4: Algoritmo 2 – Algoritmo para encontrar as atividades finas e largas usando a definição do analista (SOUZA, 2017)

2.5.2. Aumento de Poder

O padrão de aumento de poder visa dar autoridade de tomada de decisão aos trabalhadores para reduzir as etapas de gerenciamento. Nos processos de negócio tradicionais, tempo considerável pode ser gasto na autorização do trabalho que deve ser feito por outros.

Quando os trabalhadores têm mais autoridade para tomar decisões, o resultado pode ser mais eficiência devido ao menor tempo de processamento. No entanto, a qualidade da decisão pode ser menor e os erros podem passar despercebidos. Além disso, se as decisões levarem ao retrabalho, o custo de execução pode ser maior.

A proposta para aplicar esse padrão é verificar as transferências em duas tarefas consecutivas e uma tomada de decisão na segunda tarefa. Quando um trabalhador não tem poder suficiente para decidir e outra pessoa executa essa tarefa, há uma transferência. Após uma transferência, é necessário verificar se uma tomada de decisão é realizada.

Segundo Van Der Aalst (2003), existem três tipos de decisões: escolha exclusiva, escolha múltipla e escolha adiada. A escolha exclusiva é um ponto no fluxo de trabalho onde, com base em dados de decisão ou controle de fluxo de trabalho, uma das várias ramificações é escolhida. A escolha múltipla é um ponto no fluxo de trabalho onde, com base em dados de decisão ou controle de fluxo de trabalho, as ramificações são escolhidas. Representa uma decisão com mais de uma saída possível. Uma escolha adiada é um ponto no fluxo de trabalho em que uma ramificação é escolhida. No entanto, é diferente da escolha exclusiva, uma vez que a escolha não é feita explicitamente, mas é baseada na instância, onde um ramo é ativado e o outro é cancelado.

A heurística de aumento de poder deste trabalho se concentra em decisões explícitas, ou seja, decisões executadas por alguém porque outro trabalhador não tem autoridade suficiente para fazê-lo. Então, a escolha adiada não é considerada.

A premissa da heurística é baseada nos trabalhos de Thom et al. (2007) e Van Der Aalst (2003). Segundo esses autores, uma aprovação ou decisão sempre leva a um

XOR-split ou OR-split. Se uma aprovação ou decisão é executada sem a modelagem dessas divisões, é um erro de modelagem e o caso é ignorado. Sem a divisão, o significado da modelagem é que, independente do resultado da aprovação e decisão, ela sempre levará ao mesmo caminho no modelo, o que é contraditório ao conceito de aprovação que pode resultar em “aprovado” ou “rejeitado”. O conceito de decisão que pode resultar em uma ou mais ações a serem tomadas.

Formalizando, dadas duas transições sequenciais T_j e T_i , desempenhadas por diferentes funções x e y , respectivamente, se após a segunda transição T_i existir um XOR-split ou OR-split, sugere-se aplicar o padrão de redesenho “Empoderamento” na transição T_i . Em outras palavras, T_i pode ser executado por x . A Figura 5 apresenta esse algoritmo.

```

1  Algorithm 3: getEmpowerTransitions( placeList:LinkedList, transitionList:LinkedList)
2  // placeList is a linked list where each element is a place with two set attributes
3  // previousTransitions and nextTransitions
4
5  // transitionList is a linked list where each element is a transition with
6  // two set attributes previousPlaces and nextPlaces
7
8  Declare DS:Array
9  // DS is the array of places with only one previous transition
10 // and at least two following transitions
11
12 Declare DT:Array
13 // DT is the array of transitions with only one previous place
14 // and the following place is in the DS array
15
16 Declare D:Array
17 // D is the array of transitions to indicate empower redesign pattern application
18
19 Begin
20
21   for each place in placeList do
22     if length(place.previousTransitions)==1 and length(place.nextTransitions)>=2 then
23       DS.add(place)
24   for each transition in transitionList do
25     if length(transition.previousPlaces)==1 then
26       for each place in transition.nextPlaces do
27         if place in DS then
28           DT.add(transition)
29   for each transition1 in transitionList do
30     for transition2 in DT do
31       if intersection(transition1.nextPlaces, transition2.previousPlaces) != Empty
32         and transition1.role != transition2.role then
33         D.add(transition2)
34
35   return D
36
37 End

```

Figura 5: Algoritmo 3 – Algoritmo para encontrar transições que podem ter aumento de poder (SOUZA, 2017)

2.5.3. Ressequenciamento

O padrão de ressequenciamento visa mover as tarefas para lugares mais apropriados no processo de negócio (REIJERS e MANSAR, 2005). Às vezes, é melhor adiar a execução de uma tarefa se a saída não for necessária pelas tarefas imediatamente a seguir. Isso economiza custos. Uma tarefa também pode ser movida para a proximidade de tarefas semelhantes, reduzindo os tempos de configuração.

A heurística para automatizar a identificação de partes do processo para aplicar o padrão de redesenho de "Ressequenciamento" deve identificar tarefas que não estão nos locais mais apropriados. Duas propostas foram elaboradas: (i) Analisar a necessidade de execução de tarefas em um determinado momento; e (ii) Verificar a similaridade entre as tarefas.

2.5.3.1. Execução de tarefas em um determinado momento

De acordo com Reijers e Mansar (2005), em alguns casos é melhor adiar a execução de uma tarefa se ela não for necessária para as tarefas seguintes. Isso indica que a execução da tarefa nesse momento é desnecessária e economiza custos.

Segundo Netjes (2010), para decidir se duas tarefas sequenciais podem ser executadas em paralelo, é necessário verificar se a segunda utiliza como entrada a saída da tarefa anterior. Como não há dependência entre as duas tarefas, segundo Netjes (2010), elas podem ser executadas em paralelo. Assim, dadas duas tarefas sequenciais, deve-se indicar a primeira tarefa a ser adiada se a segunda tarefa não usar como entrada nada produzido pela primeira tarefa.

É importante destacar que é possível existirem casos em que as tarefas não possuem dependências de entrada/saída, porém devem ser executadas devido à ordem temporal. Para esses casos, o analista deve analisar para decidir se a tarefa pode ser adiada. A Figura 6 apresenta esse algoritmo.

```

1 Algorithm 4: getNoDependenceTransitions(transitionList:LinkedList)
2
3 // transitionList is a linked list where each element is a transition with
4 // four set attributes previousPlaces, nextPlaces, inputs and outputs
5
6 Declare ST:Array
7 // ST is the array of transitions with no outputs used as input
8 // in the following transition
9
10 Begin
11
12     for each transition1 in transitionList do
13         for each transition2 in transitionList do
14             if intersection(transition1.nextPlaces, transition2.previousPlaces) != Empty
15             and intersection(transition1.outputs, transition2.inputs) = Empty then
16                 ST.add(transition1)
17
18     return ST
19
20 End

```

Figura 6: Algoritmo 4 – Algoritmo para encontrar transições independentes (SOUZA, 2017)

2.5.3.2. Similaridade entre as tarefas

Segundo Reijers e Mansar (2005), uma tarefa pode ser movida para a proximidade de tarefas similares, a fim de reduzir os tempos de setup. Weidlich et al. (2010) propõem o uso do conceito de “documento virtual” para identificar a similaridade entre as tarefas. Um documento virtual de um nó consiste em todas as palavras de informações textuais relacionadas a esse nó. Dados dois documentos virtuais, a similaridade é calculada com base na coincidência dos termos apresentados nos dois documentos virtuais. Weidlich et al. (2010) define um documento virtual de uma tarefa como os termos do rótulo da tarefa e outras informações relacionadas, como rótulos de função, entradas e saídas e a descrição textual da tarefa. Assim, a proposta é ordenar todos os pares de tarefas de forma decrescente de acordo com a porcentagem de palavras coincidentes, considerando as palavras envolvidas no contexto da tarefa.

É importante ressaltar que as técnicas para verificar a similaridade entre as tarefas utilizadas na literatura (EHRIG et al., 2007; VAN DONGEN et al., 2008; DIJKMAN et al., 2009; WEIDLICH et al., 2010; YAN et al., 2010) identificam tarefas similares em processos distintos, ou mesmo em organizações distintas, em um cenário onde as organizações são fundidas e os processos de negócios devem ser avaliados para identificar processos correspondentes. Este trabalho se concentra na identificação de

similaridade entre tarefas de um único modelo. Desta forma, as diferenças entre nomes e termos usados para descrever algo semelhante tendem a ser menos impactantes em nosso cenário, uma vez que os termos e conceitos estão no escopo de um único processo. Para auxiliar neste tópico, Leopold et al. (2013) propuseram uma técnica automática para identificar violações em padrões de nomeação de tarefas, o que ajudaria a reduzir a ocorrência de diferenças entre nomeações e termos usados para descrever algo semelhante. Além disso, as características pessoais da escrita tendem a ser menores, já que estamos analisando um único processo provavelmente modelado por uma única pessoa ou equipe. O analista deve ainda confirmar se os pares de tarefas retornados podem ser colocados próximos ou não (devido a dependências temporais ou de entrada/saída).

Palavras de parada (artigos, pronomes, preposições, conjunções e interjeições) não são consideradas na construção da lista de palavras envolvidas, e as palavras devem ser normalizadas para excluir inflexões de gerúndios e verbos (WEIDLICH et al., 2010). A Figura 7 apresenta esse algoritmo.

```

1  Algorithm 5: getSimilarTransitionsRank(transitionList:List)
2
3  // transitionList is a list where each element is a transition with an attribute setOfWords.
4  // Each element of setOfWords is a word with two attributes theWord and quantity,
5  // quantity is the number of times the word appears in the transition
6
7  Declare TP:Array
8  // TP is the list of similar transitions and the percentage of similarity
9
10 Begin
11   for i=1 to len(transitionList) do
12     transition1 = transitionList[i]
13     for j=i+1 to len(transitionList) do
14       transition2 = transitionList[j]
15       intersection_quantity = 0
16       union_quantity = 0
17       for each word1 in transition1.setOfWords do
18         union_quantity += word1.quantity
19         for each word2 in transition2.setOfWords do
20           union_quantity += word2.quantity
21           if word1.theWord == word2.theWord then
22             intersection_quantity += word1.quantity + word2.quantity
23         percentage = intersection_quantity / union_quantity
24         TP.add(transition1, transition2, percentage)
25     // Return TP in descrescent order by percentage value
26     return orderDescByPercentage(TP)
27 End

```

Figura 7: Algoritmo 5 – Algoritmo para ordenar as duplas de atividades de acordo com o percentual de palavras coincidentes (SOUZA, 2017)

2.6. Considerações finais

Nesse capítulo foram introduzidos os conceitos básicos de gerenciamento de processos de negócios, assim como os algoritmos desenvolvidos por Souza et al. (2017), que por sua vez foram baseados nas heurísticas que Souza (2013) propôs, utilizando padrões de redesenho para melhoria de processos. No capítulo a seguir, é apresentada a implementação desses algoritmos.

3. Implementação dos algoritmos para aplicação dos padrões de redesenho

3.1. Descrição geral da solução

A solução proposta é uma ferramenta que, ao receber um arquivo XML contendo um processo de negócio, aplica os algoritmos elaborados, fornecendo como saída possíveis pontos de melhoria. Estes resultados, obtidos de forma automatizada, auxiliam na análise do processo como um todo, permitindo ao analista tomar decisões relacionadas à melhoria dos processos que sejam adequadas à sua realidade. Deste modo, o analista não precisa se preocupar com a aplicação dos padrões e heurísticas, podendo focar todo seu esforço na análise em si.

3.2. Especificação da ferramenta

3.2.1. Requisitos funcionais

RF01: O sistema deve ser capaz de ler um processo de negócio em um arquivo XML

RF02: O sistema deve exibir resultados das heurísticas implementadas

RF03: O sistema deve permitir alteração do percentual (distância da mediana) na heurística Composição de Atividades - Proposta 2

RF04: O sistema deve exibir detalhes das transições

RF05: O sistema deve permitir a troca do arquivo a ser lido

RF06: O sistema deve exibir sua descrição

3.2.2. Diagrama de classes

O diagrama de classe na Figura 8 representa as classes implementadas no projeto. A classe *TaskComposition* representa a heurística Composição de Atividades; *Empower* representa a heurística Aumento de Poder; e *Resequencing* representa a heurística Ressequenciamento. A classe *BusinessProcess* é composta pelos elementos que formam um Processo de Negócio, sendo as transições representadas na classe *Transition*, as posições na classe *Place* e os arcos na classe *Arc*. Vale ressaltar que, nesse projeto, a classe *Arc* não é utilizada, mas foi modelada e implementada para possível utilização em projetos futuros.

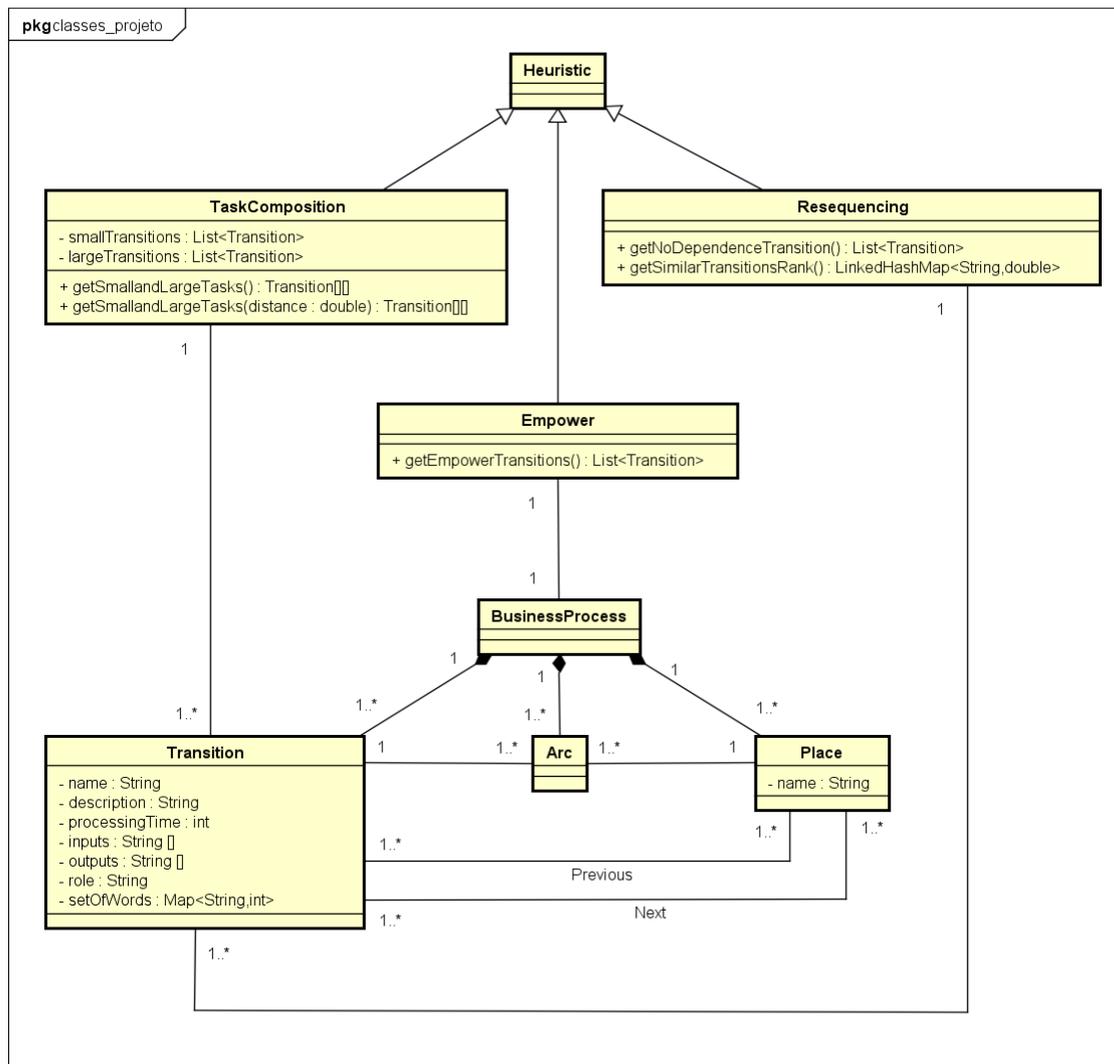


Figura 8: Diagrama de classes

As classes *TaskComposition* e *Resequencing* utilizam um conjunto de *Transitions* para calcular os resultados das heurísticas, enquanto a classe *Empower* utiliza um *BusinessProcess* para gerar os resultados. A classe *Transition* possui dois conjuntos de *Places*, onde um representa as posições anteriores de uma transição, e o outro, as posições seguintes. Analogamente, a classe *Place* possui dois conjuntos de *Transitions*: as transições anteriores e as transições posteriores. A classe *Arc* possui uma *Transition* e um *Place* como origem e destino. O Anexo 1 apresenta a implementação dessas classes.

3.2.3. Diagrama de atividades

A Figura 9 representa as atividades da ferramenta. As funcionalidades serão especificadas mais detalhadamente na seção seguinte.

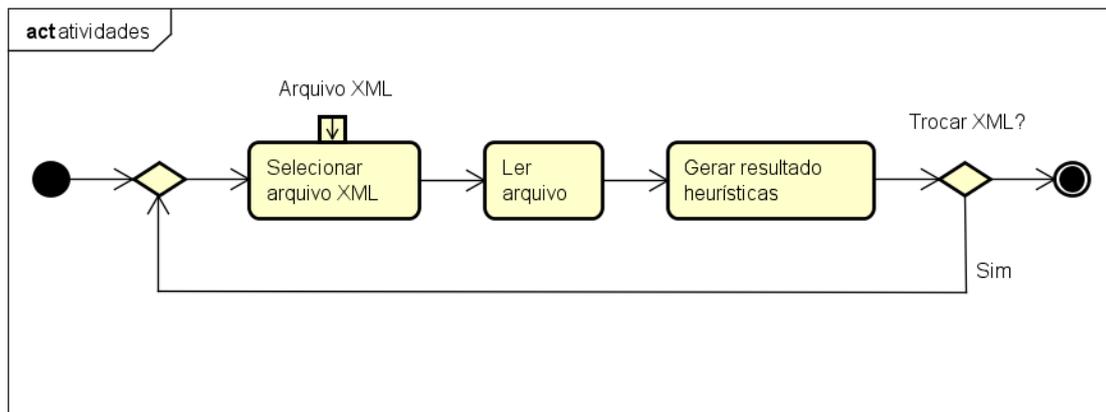


Figura 9: Diagrama de atividades

A entrada de dados é feita por meio de um arquivo no formato XML. Este arquivo deve estar corretamente formatado; caso contrário, a ferramenta não processará os dados. A Figura 10 representa um exemplo de um processo que apresenta apenas uma transição, uma posição e um arco. No entanto, um processo de negócio completo é representado pela repetição de diversos elementos, como ilustrado pela Figura 11. É importante observar que os elementos devem seguir a ordem exibida nas Figuras 10 e 11.

```

1  <businessProcess>
2    <transitions>
3      <id>tl</id>
4      <name>Atender o cliente</name>
5      <description>Atender cliente para preenchimento
6        da intenção de compra de automóvel.</description>
7      <processingTime>150</processingTime>
8      <inputs>Intenção de compra</inputs>
9      <outputs>Intenção de compra preenchida</outputs>
10     <outputs>Dados financeiros cliente</outputs>
11     <previousPlaces>p1</previousPlaces>
12     <nextPlaces>p2</nextPlaces>
13     <role>Vendedor</role>
14     <setOfWords>
15       <entry>
16         <key>atender</key>
17         <value>2</value>
18       </entry>
19     </setOfWords>
20   </transitions>
21   <places>
22     <id>p1</id>
23     <name></name>
24     <previousTransitions></previousTransitions>
25     <nextTransitions>tl</nextTransitions>
26   </places>
27   <arcs>
28     <source>p1</source>
29     <target>tl</target>
30   </arcs>
31 </businessProcess>

```

Figura 10: Exemplo com todas as *tags* que o sistema reconhece.

```

1  <businessProcess>
2    <transitions>
3      <inputs></inputs>
4      <inputs></inputs>
5      ...
6      <outputs></outputs>
7      <outputs></outputs>
8      ...
9      <previousPlaces></previousPlaces>
10     <previousPlaces></previousPlaces>
11     ...
12     <nextPlaces></nextPlaces>
13     <nextPlaces></nextPlaces>
14     ...
15     <setOfWords>
16       <entry>
17       </entry>
18       <entry>
19       </entry>
20       ...
21     </setOfWords>
22   </transitions>
23   <transitions>
24   </transitions>
25   ...
26   <places>
27     <previousTransitions></previousTransitions>
28     <previousTransitions></previousTransitions>
29     ...
30     <nextTransitions></nextTransitions>
31     <nextTransitions></nextTransitions>
32     ...
33   </places>
34   <places>
35   </places>
36   ...
37   <arcs>
38   </arcs>
39   <arcs>
40   </arcs>
41   ...
42 </businessProcess>

```

Figura 11: Elementos que podem se repetir

3.3 Tecnologias utilizadas

As tecnologias usadas na implementação da ferramenta são descritas a seguir.

- XML

Extensible Markup Language (XML) é uma linguagem de marcação que serve para guardar dados de uma forma estruturada. Tem formato simples de texto e apresenta bastante flexibilidade. Um XML é um arquivo de texto puro, portanto independente de plataforma.

- Java 8

Java é uma linguagem de programação orientada a objetos que utiliza o conceito de máquina virtual, que interpreta o código compilado. Dessa forma, a aplicação é independente do sistema operacional. Além disso, possui diversos *frameworks* e bibliotecas que facilitam o desenvolvimento.

- Netbeans 8.2

NetBeans IDE é o ambiente de desenvolvimento oficial para Java 8, gratuito e de código aberto para desenvolvedores de software em diversas linguagens.

- Swing

Swing é um *widget toolkit GUI* (Interface de Usuário Gráfica) para uso com o Java. Com ele é possível desenvolver interface gráfica arrastando e posicionando os componentes em um *canvas* (tela).

- JAXB

Java Architecture for XML Binding (Arquitetura Java para vinculação XML) é uma estrutura de software que permite aos desenvolvedores Java mapearem classes Java para representações XML. O JAXB fornece dois recursos principais: a capacidade de empacotar objetos Java em XML e o inverso, ou seja, desempacotar XML de volta em objetos Java.

- Git (Bitbucket)

Git é um sistema de controle de versões distribuído, usado principalmente no desenvolvimento de software, para registrar o histórico de edições de qualquer tipo de

arquivo. Por sua vez, *Bitbucket* é um serviço de hospedagem de repositórios de controle de versão baseado na web, para projetos de código-fonte e desenvolvimento que usam sistemas de controle de revisão *Git*. O *Bitbucket* oferece contas gratuitas com um número ilimitado de repositórios privados.

- **Astah Community**

Astah Community é uma ferramenta de modelagem UML gratuita, que possui interface amigável e permite a criação de diversos diagramas, como o de Classes, Atividades, Estados, entre outros.

3.4 Funcionalidades

A seguir, são exibidas as telas e as funcionalidades de cada uma delas. Cabe ressaltar que a ferramenta apenas exhibe os resultados em tela, não há exportação de dados em nenhum outro formato e não é possível salvar os resultados obtidos.

3.4.1. Acessar o sistema

A Figura 12 mostra a tela inicial da ferramenta. Essa tela possui um botão para que seja selecionado o arquivo XML correspondente ao processo em que se deseja aplicar os algoritmos.

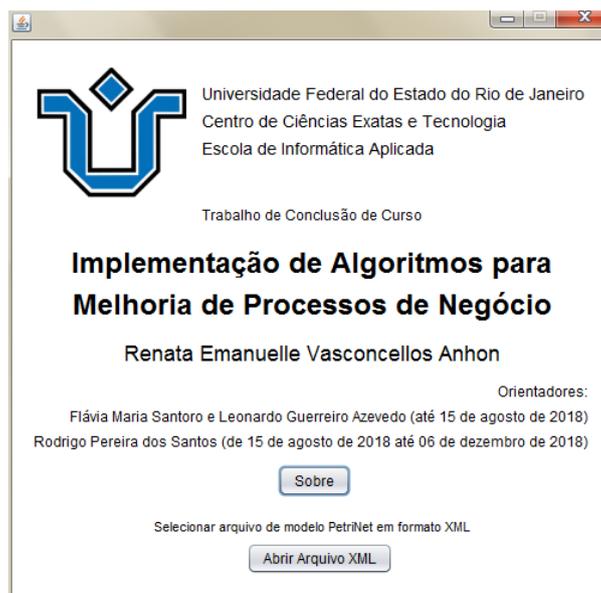


Figura 12: Tela inicial

3.4.2. Selecionar arquivo XML

Ao clicar no botão para selecionar o arquivo, a ferramenta exibe uma caixa para pesquisa do arquivo desejado no sistema, conforme ilustrado na Figura 13.

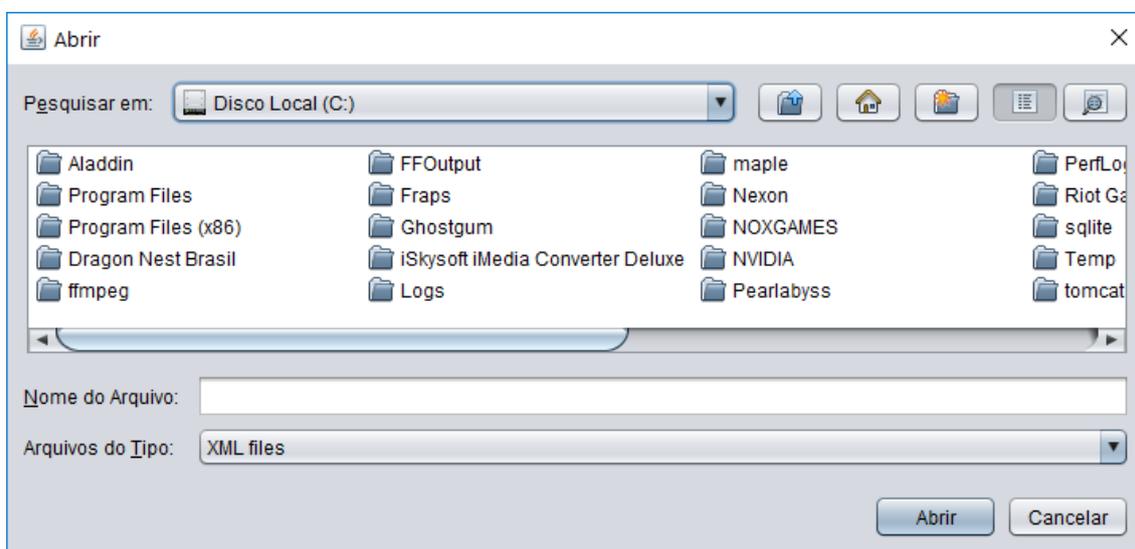


Figura 13: Buscar arquivo XML no computador

Caso não seja selecionado nenhum arquivo, seja selecionado um arquivo com extensão diferente de “.xml” ou o arquivo XML não esteja formatado corretamente, a ferramenta exibe uma mensagem de alerta conforme Figura 14.

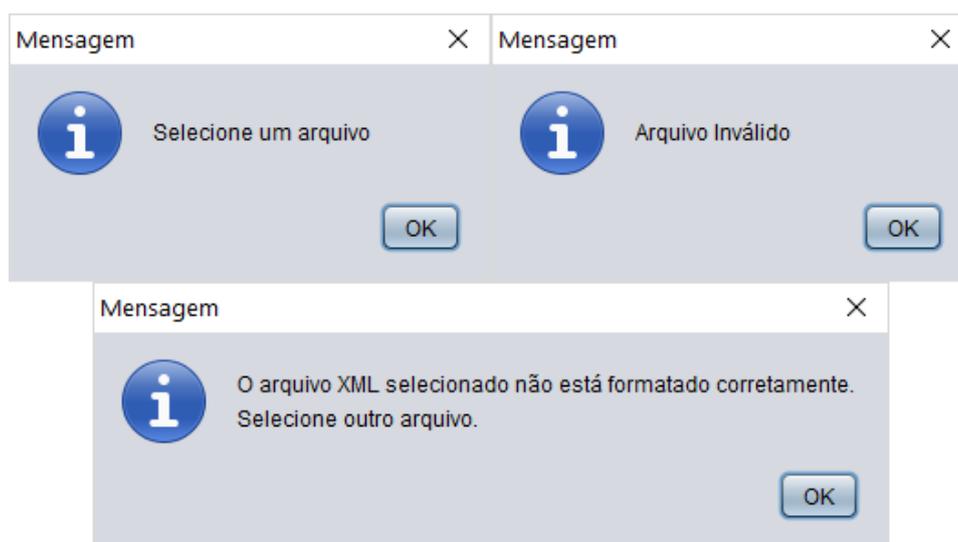


Figura 14: Mensagens de Alerta

3.4.3. Visualizar resultados das heurísticas

Após a seleção de um arquivo XML válido, a ferramenta faz a leitura dos dados e exibe os resultados dos algoritmos. Conforme mostra a Figura 15, na tela relacionada, é possível selecionar a heurística desejada, clicando nas abas na parte superior direita. É possível também retornar à tela inicial, selecionando o botão “Trocar Processo”. Desse modo, é possível alterar o arquivo sem que seja necessário fechar e reabrir a ferramenta.

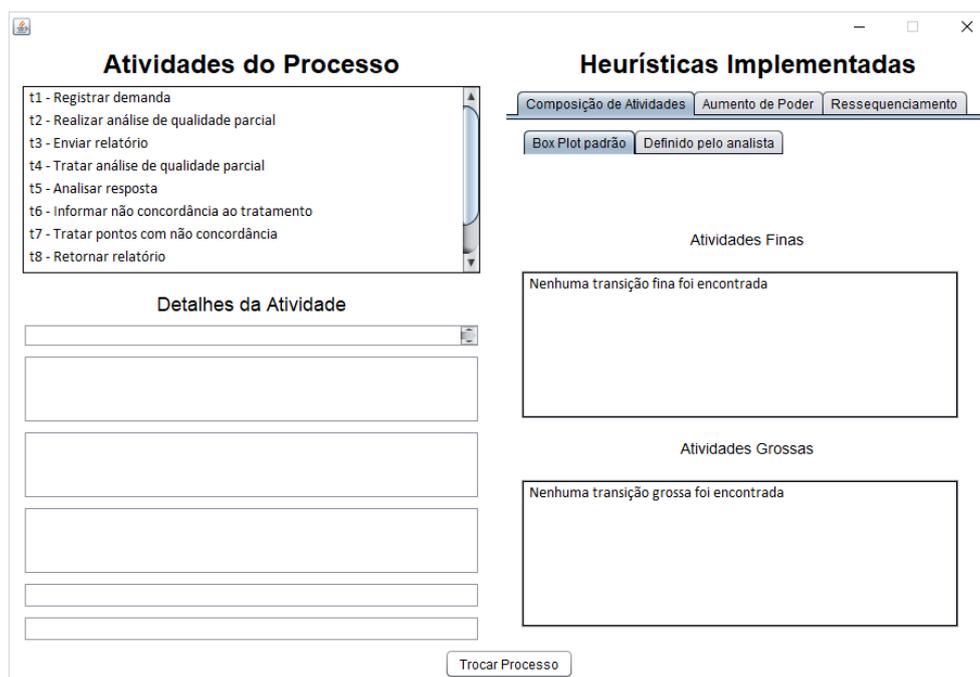


Figura 15: Tela de resultados

3.4.3.1. Composição de Atividades

Ao selecionar a aba “Composição de Atividades”, é possível visualizar os resultados das heurísticas de *boxplot* padrão ou aquela definida pelo analista. A Figura 16 mostra os resultados encontrados pelo algoritmo de *boxplot* padrão. A caixa superior exibe as transições finas e a caixa inferior, as transições grossas. Caso nenhuma transição seja identificada como fina ou grossa, será exibida na caixa correspondente a mensagem “Nenhuma transição fina [grossa] encontrada”.

Ao selecionar a aba “Definido pelo analista”, os resultados exibidos estarão de acordo com a porcentagem inserida. Por padrão, a porcentagem inicial é de 60%; para alterar, basta inserir o valor desejado no campo de texto e clicar no botão “Alterar percentual”, conforme exibido na Figura 17. Assim como no anterior, caso nenhuma transição seja identificada como fina ou grossa, será exibida na caixa correspondente a mensagem “Nenhuma transição fina [grossa] encontrada”.

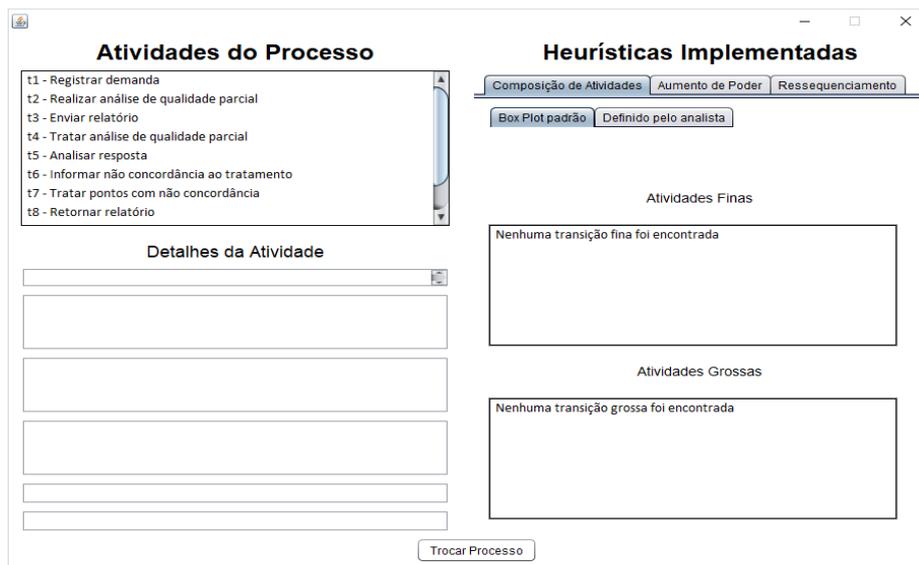


Figura 16: Tela de resultados Composição de Atividades – boxplot padrão

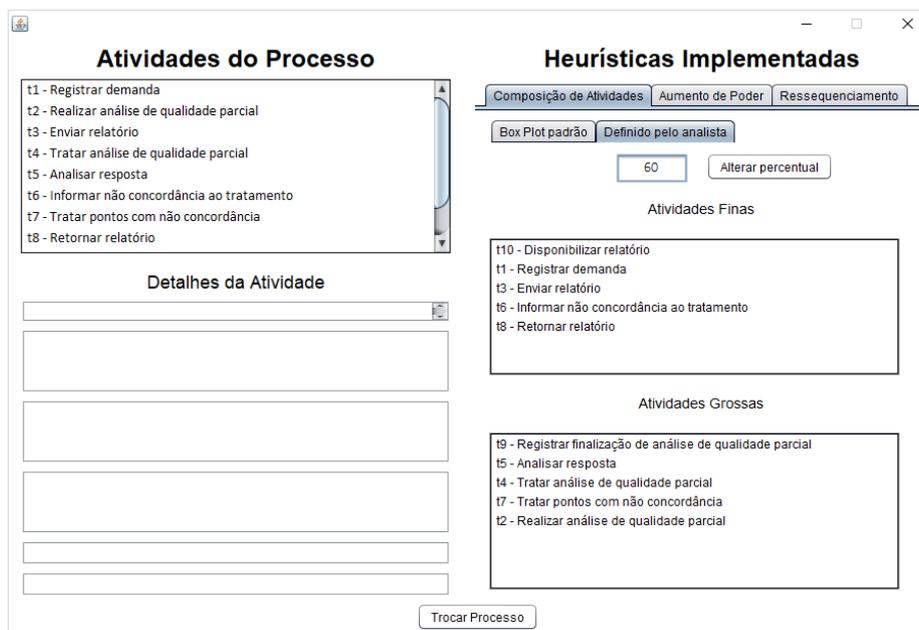


Figura 17: Tela de resultados Composição de Atividades - Definido pelo analista

3.4.3.2. Aumento de poder

A aba “Aumento de poder” exibe os resultados da heurística de aumento de poder, ilustrada na Figura 18. Caso nenhuma transição seja identificada, a caixa exibirá a mensagem “Nenhuma transição para aumento de poder foi encontrada”.

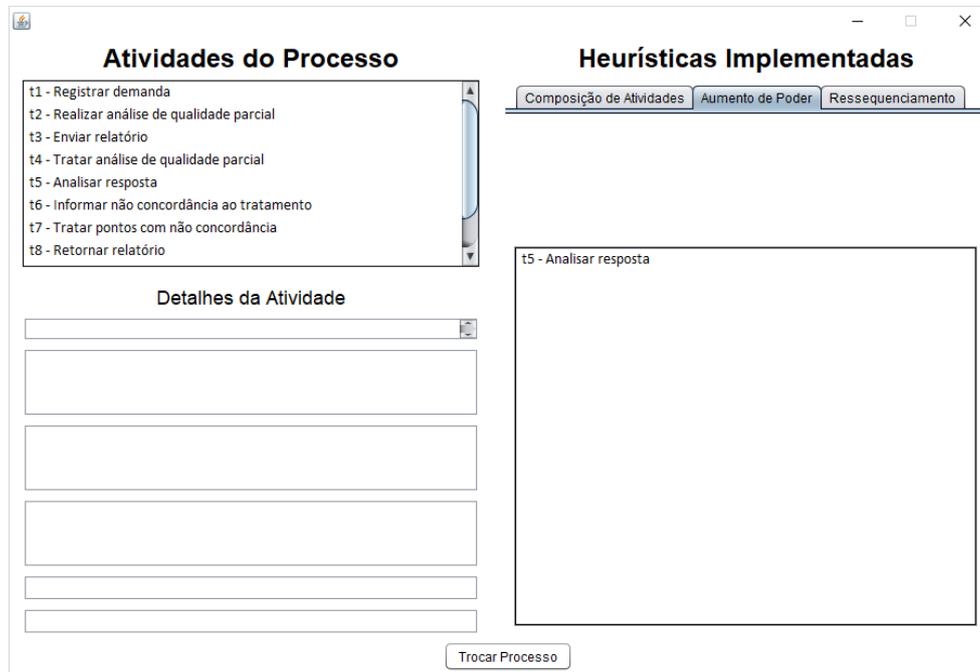


Figura 18: Tela de resultados Aumento de Poder

3.4.3.3. Ressequenciamento

Para os resultados das heurísticas de ressequenciamento, basta o usuário selecionar a aba “Ressequenciamento”. Conforme a Figura 19, na caixa superior, estão os resultados para a heurística de transições independentes. Caso nenhuma transição seja identificada, a caixa exibirá a mensagem “Nenhuma transição independente foi encontrada”. Na caixa inferior, estão os resultados da heurística de transições similares, onde os pares estão ordenados por percentual de similaridade. Para essa heurística, mesmo que não haja conjunto de palavras-chave para as transições, o resultado é exibido; porém, os percentuais terão resultado zero.

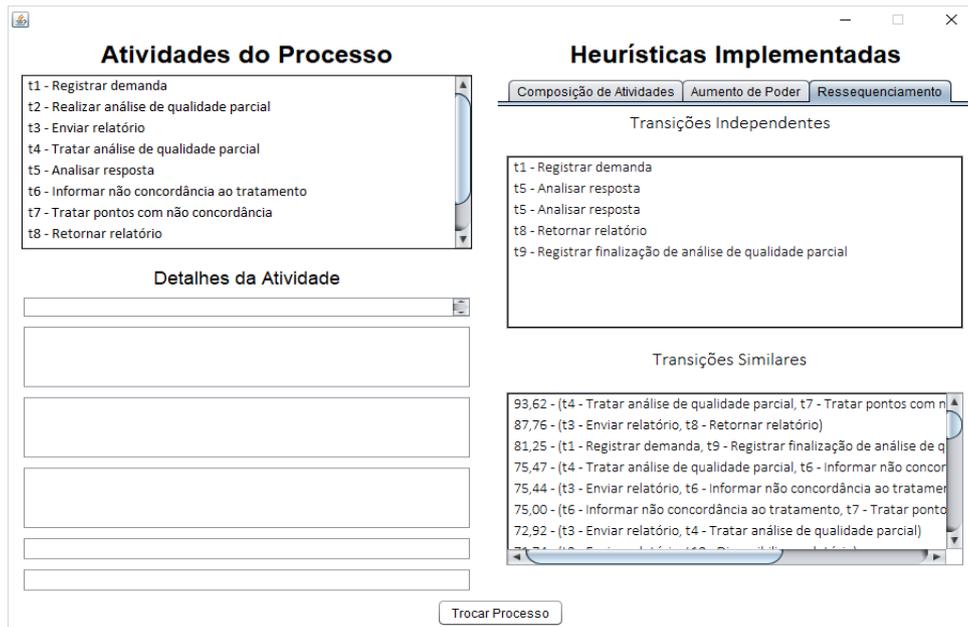


Figura 19: Tela de resultados Ressequenciamento - Transições Independentes e Similares

3.4.4. Visualizar detalhes das transições

Para visualizar os detalhes de alguma transição, basta selecionar a transição desejada na lista à esquerda, e detalhes serão exibidos abaixo, como mostra a Figura 20.

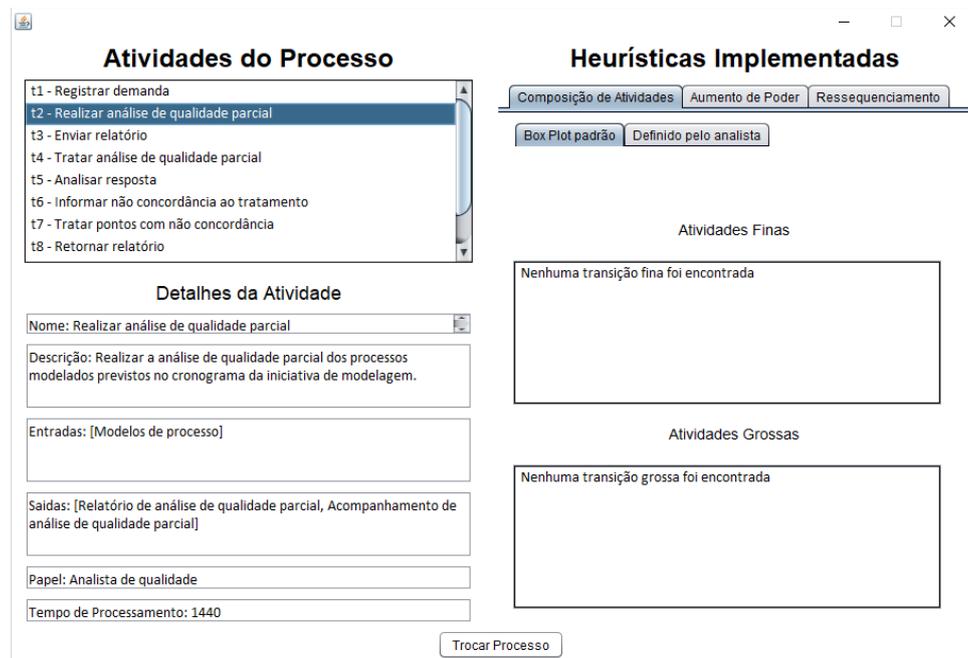


Figura 20: Tela detalhes das atividades (transições)

3.5. Considerações finais

Neste capítulo foi feita a descrição da ferramenta de solução, que apresenta como saída possíveis pontos de melhoria, que auxiliam na análise do processo como um todo. A seguir, são apresentadas as avaliações da ferramenta.

4. Avaliação

4.1. Metodologia

Com o objetivo de avaliar a implementação feita na ferramenta, foi conduzida uma avaliação. Para isso, foram utilizados três modelos de processos de negócio fictícios, produzidos apenas para fins de teste.

Na primeira etapa, foi utilizada a abordagem manual de execução dos algoritmos descritos na Seção 2.5. O foco foi verificar as saídas produzidas na aplicação dos algoritmos.

Na segunda etapa, a execução dos algoritmos foi feita utilizando a ferramenta proposta no Capítulo 3. Assim como na avaliação anterior, o foco é verificar as saídas produzidas na aplicação dos algoritmos.

Ao final, os resultados obtidos foram comparados com o objetivo de se verificar a acurácia nas saídas obtidas pela ferramenta.

4.2. Modelos utilizados nos testes

Nesta seção, são apresentados os detalhes dos modelos de processos de negócio utilizados para testes. Os arquivos XML correspondentes a cada modelo encontram-se no Anexo 2.

4.2.1 Modelo 1

O modelo 1, ilustrado na Figura 21, foi baseado em um modelo presente em Dumas (2013). Os dados adicionais são fictícios e os atributos mais relevantes estão detalhados na Tabela 1.

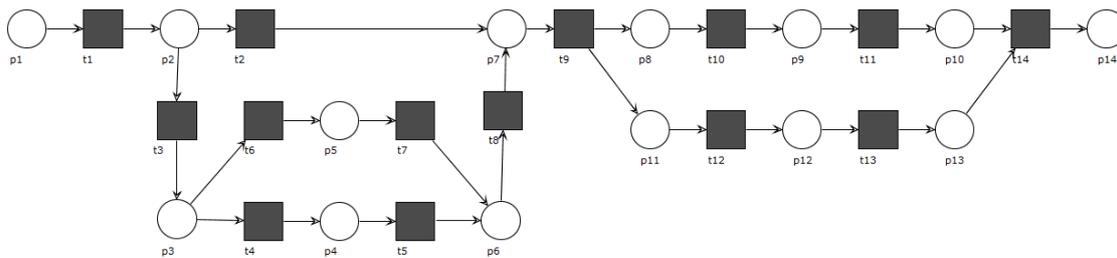


Figura 21: Modelo 1

| ID | Nome | Entradas | Saídas | Papel Responsável | Tempo |
|-----|---------------------------------------------|--------------------------------------|----------------------------|--------------------------|-------|
| t1 | check stock availability | purchase order; warehouse db | | W&D / ERPSsystem | 1 |
| t2 | retrieve product from warehouse | products warehouse | product | Warehouse & Distribution | 20 |
| t3 | check raw materials availability | purchase order; suppliers catalog | | W&D / ERPSsystem | 1 |
| t4 | request raw materials from supplier 1 | | | Warehouse & Distribution | 5 |
| t5 | obtain raw materials from supplier 1 | | raw materials | Warehouse & Distribution | 12 |
| t6 | request raw materials from supplier 2 | | | Warehouse & Distribution | 5 |
| t7 | obtain raw materials provided by supplier 2 | | raw materials | Warehouse & Distribution | 15 |
| t8 | manufacture product | raw materials | product | Warehouse & Distribution | 50 |
| t9 | confirm order | purchase order | purchase order (confirmed) | Sales | 3 |
| t10 | emit invoice | | invoice | Sales | 8 |

Tabela 1: Detalhes do modelo 1 (parte 1)

| ID | Nome | Entradas | Saídas | Papel Responsável | Tempo |
|-----|----------------------|-------------------------------------|-----------------------|--------------------------|-------|
| t11 | receive payment | purchase order (confirmed); payment | purchase order (paid) | Sales | 480 |
| t12 | get shipping address | | shipment address | Warehouse & Distribution | 10 |
| t13 | ship product | product; shipment address | | Warehouse & Distribution | 960 |
| t14 | arquivo order | purchase order (paid) | orders db | Sales | 3 |

Tabela 1: Detalhes do modelo 1 (parte 2)

4.2.2 Modelo 2

O modelo 2, ilustrado na Figura 22, foi baseado em um modelo presente em Souza (2010). Os dados adicionais são fictícios e os atributos mais relevantes estão detalhados na Tabela 2.

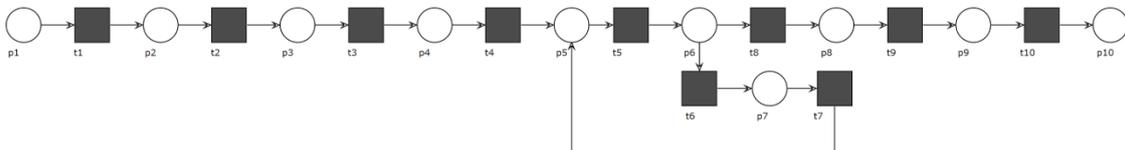


Figura 22: Modelo 2

| ID | Nome | Entradas | Saídas | Papel Responsável | Tempo |
|-----|-------------------------------------------------------|------------------------------------------------|----------------------------------------------------------------------------------------------|-----------------------|-------|
| t1 | Registrar demanda | | Acompanhamento de análise de qualidade parcial | Analista de qualidade | 10 |
| t2 | Realizar análise de qualidade parcial | Modelos de processo | Relatório de análise de qualidade parcial; Acompanhamento de análise de qualidade parcial | Analista de qualidade | 1440 |
| t3 | Enviar relatório | Relatório de análise de qualidade parcial | Relatório de análise de qualidade parcial; Acompanhamento de análise de qualidade parcial | Analista de qualidade | 15 |
| t4 | Tratar análise de qualidade parcial | Relatório de análise de qualidade parcial | Relatório de análise de qualidade parcial | Líder de projeto | 960 |
| t5 | Analisar resposta | Relatório de análise de qualidade parcial | Acompanhamento de análise de qualidade parcial | Analista de qualidade | 480 |
| t6 | Informar não concordância ao tratamento | | Relatório de análise de qualidade parcial; Acompanhamento de análise de qualidade parcial | Analista de qualidade | 15 |
| t7 | Tratar pontos com não concordância | Relatório de análise de qualidade parcial | Relatório de análise de qualidade parcial | Líder de projeto | 960 |
| t8 | Retornar relatório | Relatório de análise de qualidade parcial | Relatório de análise de qualidade parcial | Analista de qualidade | 15 |
| t9 | Registrar finalização de análise de qualidade parcial | Acompanhamento de análise de qualidade parcial | Acompanhamento de análise de qualidade parcial | Analista de qualidade | 240 |
| t10 | Disponibilizar relatório | Relatório de análise de qualidade parcial | Relatório de análise de qualidade parcial | Analista de qualidade | 5 |

Tabela 2: Detalhes do modelo 2

4.2.3 Modelo 3

O modelo 3, ilustrado na Figura 23, foi baseado em um modelo presente em Souza (2010). Os dados adicionais são fictícios e os atributos mais relevantes estão detalhados na Tabela 3.

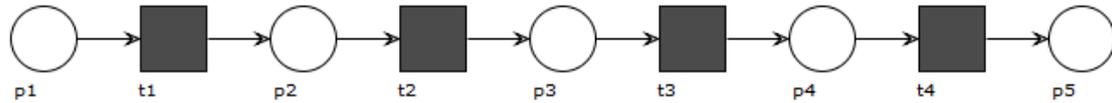


Figura 23: Modelo 3

| ID | Nome | Entradas | Saídas | Papel Responsável | Tempo |
|----|------------------------|--------------------------------------------|----------------------------------------------------------|-------------------|-------|
| t1 | Atender o cliente | Intenção de compra | Intenção de compra preenchida; dados financeiros cliente | Vendedor | 150 |
| t2 | Gerar pedido de compra | Intenção de compra preenchida | Pedido de compra | Vendedor | 40 |
| t3 | Receber pagamento | Dados financeiros cliente | Confirmação de pagamento | Financeiro | 30 |
| t4 | Entregar automóvel | Pedido de compra; confirmação de pagamento | Pedido de compra finalizado | Concessionária | 15 |

Tabela 3: Detalhes do modelo 3

4.3. Resultados Obtidos

Os resultados obtidos com a execução manual dos algoritmos e por meio da ferramenta foram idênticos, porém as figuras apresentadas mostram resultados parciais da ferramenta devido à sua extensão, servindo apenas para efeito de comparação. A seguir, os resultados detalhados.

4.3.1. Algoritmo 1: Composição de atividades

Considerando o *boxplot*, os cálculos para a execução manual foram feitos seguindo as seguintes etapas:

1. Ordenação do conjunto de transições, de acordo com o tempo de processamento, de forma crescente (conjunto Po);
2. Cálculo do quartil inferior ($Q1$), do quartil superior ($Q3$), a partir dos tempos de processamento e do intervalo interquartil (IQR);
3. Separar as atividades finas, cujo tempo de processamento for menor que $Q1 - 1,5 \cdot IQR$ (conjunto ST);
4. Separar as atividades finas, cujo tempo de processamento for maior que $Q3 + 1,5 \cdot IQR$ (conjunto LT).

A Tabela 4 apresenta os resultados obtidos destes cálculos do algoritmo 1 no modelo 1. A Figura 24 mostra os resultados obtidos com a ferramenta.

| | | |
|---------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------|------------|
| $Po = \{ (t1, 1), (t3, 1), (t9, 3), (t14, 3), (t4, 5), (t6, 5), (t10, 8), (t12, 10), (t5, 12), (t7, 15), (t2, 20), (t8, 50), (t11, 480), (t13, 960) \}$ | | |
| $Q1 = Po[3] = 3$ | $Q3 = Po[11] = 20$ | $IQR = 17$ |
| $ST = \{ \}$ <i>[não foram identificadas atividades finas]</i> | | |
| $LT = \{ t8, t11, t13 \}$ | | |

Tabela 4: Resultados obtidos manualmente, aplicando o algoritmo 1 ao modelo 1

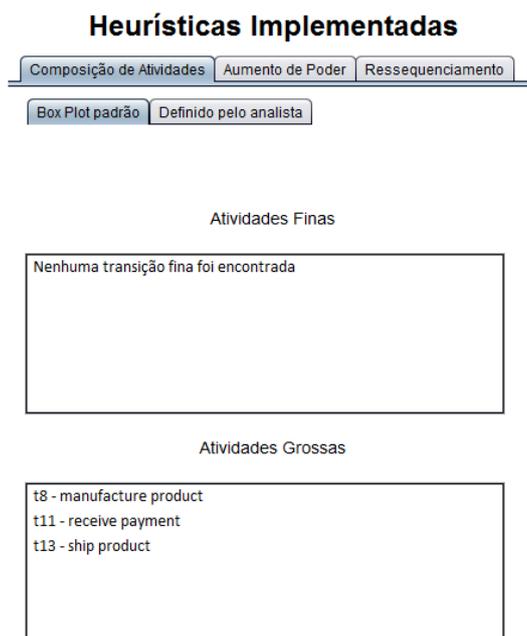


Figura 24: Resultados obtidos com a ferramenta, aplicando o algoritmo 1 ao modelo 1

A Tabela 5 apresenta os resultados obtidos pelos cálculos do algoritmo 1 no modelo 2. A Figura 25 mostra os resultados obtidos com a ferramenta.

| | | |
|-----------------------------------------------------------------------------------------------------------------------------------|--------------------|-------------|
| $Po = \{ (t_{10}, 5), (t_1, 10), (t_3, 15), (t_6, 15), (t_8, 15), (t_9, 240), (t_5, 480), (t_4, 960), (t_7, 960), (t_2, 1440) \}$ | | |
| $Q1 = Po[2] = 10$ | $Q3 = Po[8] = 960$ | $IQR = 950$ |
| $ST = \{ \}$ <i>[não foram identificadas atividades finas]</i> | | |
| $LT = \{ \}$ <i>[não foram identificadas atividades grossas]</i> | | |

Tabela 5: Resultados obtidos manualmente, aplicando o algoritmo 1 ao modelo 2

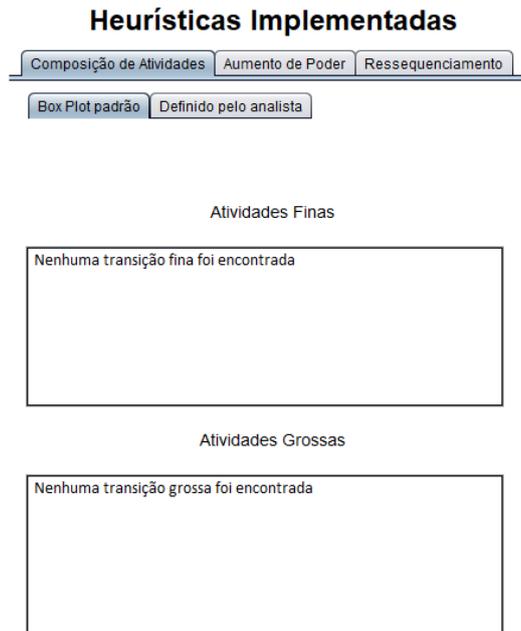


Figura 25: Resultados obtidos com a ferramenta, aplicando o algoritmo 1 ao modelo 2

A Tabela 6 apresenta os resultados obtidos destes cálculos do algoritmo 1 no modelo 3. A Figura 26 mostra os resultados obtidos com a ferramenta.

| | | |
|---------------------------------------------------------------------------------------------------|-------------------|------------|
| $Po = \{ 15, 30, 40, 150 \}$ | | |
| $Q1 = Po[1] = 15$ | $Q3 = Po[3] = 40$ | $IQR = 25$ |
| $ST = \{ \}$ <i>[não foram identificadas atividades finas]</i> | | |
| $LT = \{ t1 \}$ | | |

Tabela 6: Resultados obtidos manualmente, aplicando o algoritmo 1 ao modelo 3

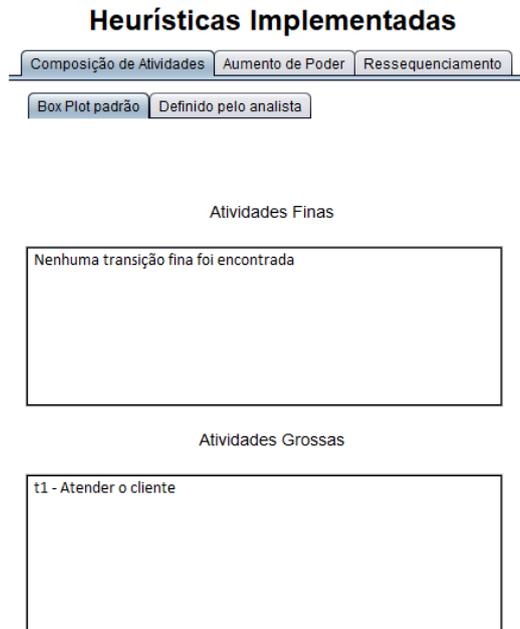


Figura 26: Resultados obtidos com a ferramenta, aplicando o algoritmo 1 ao modelo 3

4.3.2. Algoritmo 2: Composição de atividades pelo analista

Utilizando o conceito de definição do analista, os cálculos para a primeira avaliação foram feitos seguindo as seguintes etapas:

1. Considerando o conjunto Po obtido no teste anterior, calcular a mediana m dos tempos de processamento;
2. Definir o percentual de distância d da mediana (para esse passo, foram considerados os valores de d como 0%, 10%, 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%, 100%);
3. Separar as atividades finas, cujo tempo de processamento for menor que $(1 - d) * m$ (conjunto ST)
4. Separar as atividades finas, cujo tempo de processamento for maior que $(1 - d) * m$ (conjunto LT).

A Tabela 7 apresenta os resultados obtidos destes cálculos do algoritmo 2 com percentuais de 0 a 100 no modelo 1. A Figura 27 mostra os resultados obtidos com a ferramenta utilizando o percentual de 50.

$P_0 = \{ (t1, 1), (t3, 1), (t9, 3), (t14, 3), (t4, 5), (t6, 5), (t10, 8), (t11, 10), (t5, 12), (t7, 15), (t2, 20), (t8, 50), (t13, 480), (t12, 960) \}$

$m = 9$

| d | ST | LT |
|-----|----------------------------------|-----------------------------------|
| 0 | { t1, t3, t4, t6, t9, t10, t14 } | { t2, t5, t7, t8, t11, t12, t13 } |
| 10 | { t1, t3, t4, t6, t9, t10, t14 } | { t2, t5, t7, t8, t11, t12, t13 } |
| 20 | { t1, t3, t4, t6, t9, t14 } | { t2, t5, t7, t8, t12, t13 } |
| 30 | { t1, t3, t4, t6, t9, t14 } | { t2, t5, t7, t8, t12, t13 } |
| 40 | { t1, t3, t4, t6, t9, t14 } | { t2, t7, t8, t12, t13 } |
| 50 | { t1, t3, t9, t14 } | { t2, t7, t8, t12, t13 } |
| 60 | { t1, t3, t9, t14 } | { t2, t7, t8, t12, t13 } |
| 70 | { t1, t3 } | { t2, t8, t12, t13 } |
| 80 | { t1, t3 } | { t2, t8, t12, t13 } |
| 90 | { } | { t2, t8, t12, t13 } |
| 100 | { } | { t2, t8, t12, t13 } |

Tabela 7: Resultados obtidos manualmente, aplicando o algoritmo 2 ao modelo 1

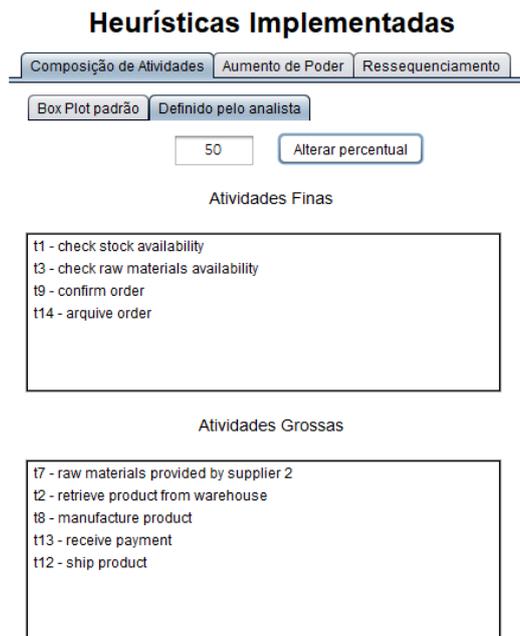


Figura 27: Resultados obtidos com a ferramenta, aplicando o algoritmo 2 ao modelo 1

A Tabela 8 apresenta os resultados obtidos destes cálculos do algoritmo 2 com percentuais de 0 a 100 no modelo 2. A Figura 28 mostra os resultados obtidos com a ferramenta utilizando o percentual de 50.

$P_0 = \{ (t_{10}, 5), (t_1, 10), (t_3, 15), (t_6, 15), (t_8, 15), (t_9, 240), (t_5, 480), (t_4, 960), (t_7, 960), (t_2, 1440) \}$

$m = 9$

| d | ST | LT |
|-----|-------------------------|------------------------|
| 0 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 10 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 20 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 30 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 40 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 50 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 60 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 70 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 80 | { t1, t3, t6, t8, t10 } | { t2, t4, t5, t7, t9 } |
| 90 | { t1, t10 } | { t2, t4, t5, t7 } |
| 100 | { } | { t2, t4, t5, t7 } |

Tabela 8: Resultados obtidos manualmente, aplicando o algoritmo 2 ao modelo 2

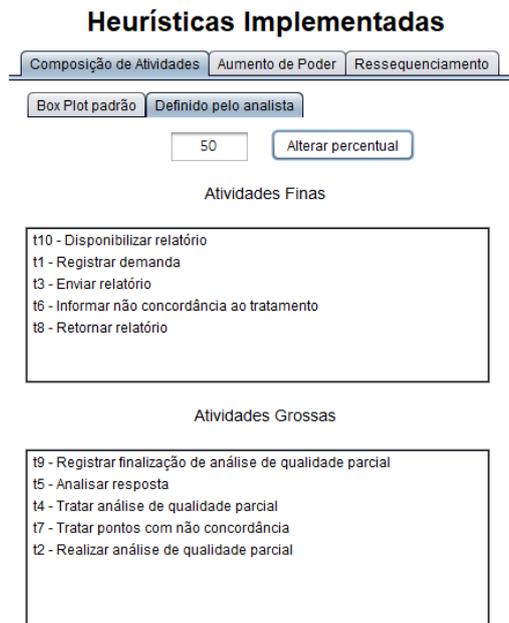


Figura 28: Resultados obtidos com a ferramenta, aplicando o algoritmo 2 ao modelo 2

A Tabela 9 apresenta os resultados obtidos destes cálculos do algoritmo 2 com percentuais de 0 a 100 no modelo 3. A Figura 29 mostra os resultados obtidos com a ferramenta utilizando o percentual de 10.

$$Po = \{ (t4, 15) ,(t3, 30), (t2, 40), (t1, 150) \}$$

$$m = 35$$

| d | ST | LT |
|----|------------|------------|
| 0 | { t4, t3 } | { t2, t1 } |
| 10 | { t4, t3 } | { t2, t1 } |
| 20 | { t4 } | { t1 } |
| 30 | { t4 } | { t1 } |
| 40 | { t4 } | { t1 } |
| 50 | { t4 } | { t1 } |
| 60 | { } | { t1 } |
| 70 | { } | { t1 } |

Tabela 9: Resultados obtidos manualmente, aplicando o algoritmo 2 ao modelo 3 (parte 1)

| d | ST | LT |
|-----|-----|--------|
| 80 | { } | { t1 } |
| 90 | { } | { t1 } |
| 100 | { } | { t1 } |

Tabela 9: Resultados obtidos manualmente, aplicando o algoritmo 2 ao modelo 3 (parte 2)

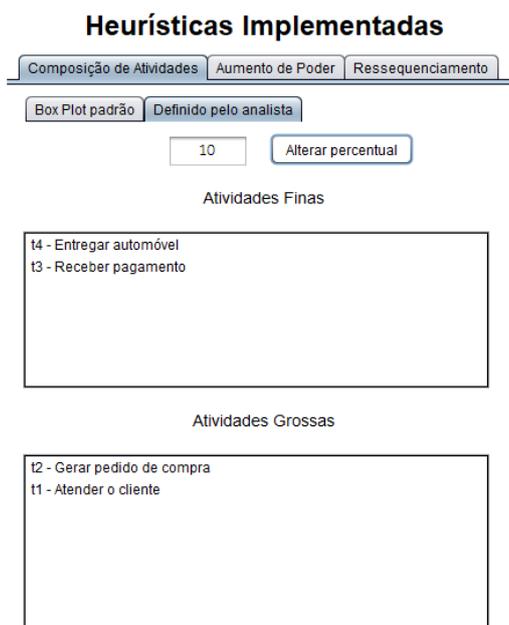


Figura 29: Resultados obtidos com a ferramenta, aplicando o algoritmo 2 ao modelo 3

4.3.3. Algoritmo 3: Aumento de poder

Para a proposta de aumento de poder, os cálculos para a primeira avaliação foram feitos seguindo as etapas abaixo:

1. Separar as posições que possuem apenas uma transição anterior e pelo menos duas transições seguintes (conjunto DS);
2. Separar as transições que possuem apenas uma posição anterior e a posição seguinte esteja em DS (conjunto DT);
3. Para cada transição dt_i em DT, identificar se as transições que possuem a posição seguinte igual a posição anterior de ds_i são executadas por papéis

diferentes de dt_i . Em caso positivo, dt_i é indicado para aumento de poder (conjunto D).

A Tabela 10 apresenta os resultados obtidos destes cálculos do algoritmo 3 no modelo 1. A Figura 30 mostra os resultados obtidos com a ferramenta.

| |
|---------------------|
| $DS = \{ p2, p3 \}$ |
| $DT = \{ t1, t3 \}$ |
| $D = \{ \}$ |

Tabela 10: Resultados obtidos manualmente, aplicando o algoritmo 3 ao modelo 1

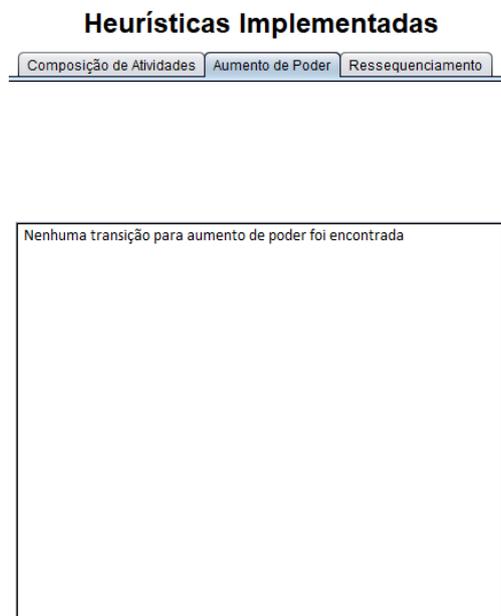


Figura 30: Resultados obtidos com a ferramenta, aplicando o algoritmo 3 ao modelo 1

A Tabela 11 apresenta os resultados obtidos destes cálculos do algoritmo 3 no modelo 2. A Figura 31 mostra os resultados obtidos com a ferramenta.

| |
|-----------------|
| $DS = \{ p6 \}$ |
| $DT = \{ t5 \}$ |
| $D = \{ t5 \}$ |

Tabela 11: Resultados obtidos manualmente, aplicando o algoritmo 3 ao modelo 2

Heurísticas Implementadas

Composição de Atividades Aumento de Poder Ressequenciamento

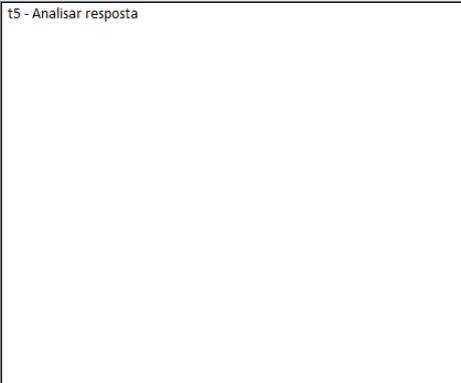


Figura 31: Resultados obtidos com a ferramenta, aplicando o algoritmo 3 ao modelo 2

A Tabela 12 apresenta os resultados obtidos destes cálculos do algoritmo 3 no modelo 3. A Figura 32 mostra os resultados obtidos com a ferramenta.

| |
|--------------|
| $DS = \{ \}$ |
| $DT = \{ \}$ |
| $D = \{ \}$ |

Tabela 12: Resultados obtidos manualmente, aplicando o algoritmo 3 ao modelo 3

Heurísticas Implementadas

Composição de Atividades Aumento de Poder Ressequenciamento

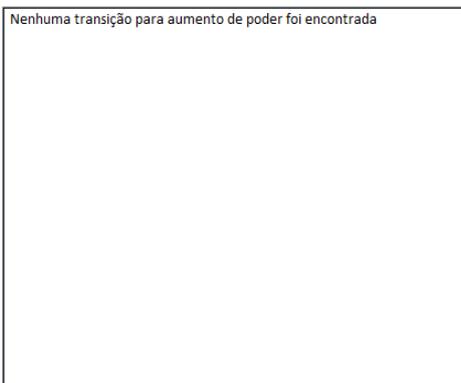


Figura 32: Resultados obtidos com a ferramenta, aplicando o algoritmo 3 ao modelo 3

4.3.4. Algoritmo 4: Transições independentes

Para encontrar as transições independentes pela primeira avaliação, foram feitas as seguintes etapas:

1. Identificar os pares de transições sequenciais, isto é, em que a interseção do conjunto de posições seguintes de t_i com o conjunto de posições anteriores de t_j seja diferente de vazio (conjunto S);
2. Para cada par (t_i, t_j) identificado em S, se t_j não possuir nenhuma entrada que seja saída de t_i , então t_i pode ser postergada (conjunto ST).

A Tabela 13 apresenta os resultados obtidos destes cálculos do algoritmo 4 no modelo 1. A Figura 33 mostra os resultados obtidos com a ferramenta.

| |
|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| $S = \{ (t1, t2); (t1, t3); (t2, t9); (t3, t4); (t3, t6); (t4, t5); (t5, t8); (t6, t7); (t7, t8); (t8, t9); (t9, t10); (t9, t12); (t10, t11); (t11, t14); (t12, t13); (t13, t14) \}$ |
| $ST = \{ t1, t1, t2, t3, t3, t4, t6, t8, t9, t9, t10, t13 \}$ |

Tabela 13: Resultados obtidos manualmente, aplicando o algoritmo 4 ao modelo 1

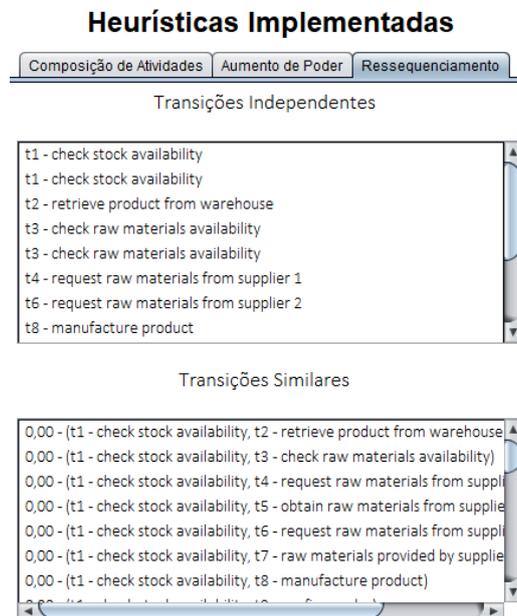


Figura 33: Resultados obtidos com a ferramenta, aplicando o algoritmo 4 ao modelo 1

A Tabela 14 apresenta os resultados obtidos destes cálculos do algoritmo 4 no modelo 2. A Figura 34 mostra os resultados obtidos com a ferramenta.

| |
|-----------------------------------------------------------------------------------------------------------------|
| $S = \{ (t1, t2); (t2, t3); (t3, t4); (t4, t5); (t5, t6); (t5, t8); (t6, t7); (t7, t5); (t8, t9); (t9, t10) \}$ |
| $ST = \{ t1, t5, t5, t8, t9 \}$ |

Tabela 14: Resultados obtidos manualmente, aplicando o algoritmo 4 ao modelo 2

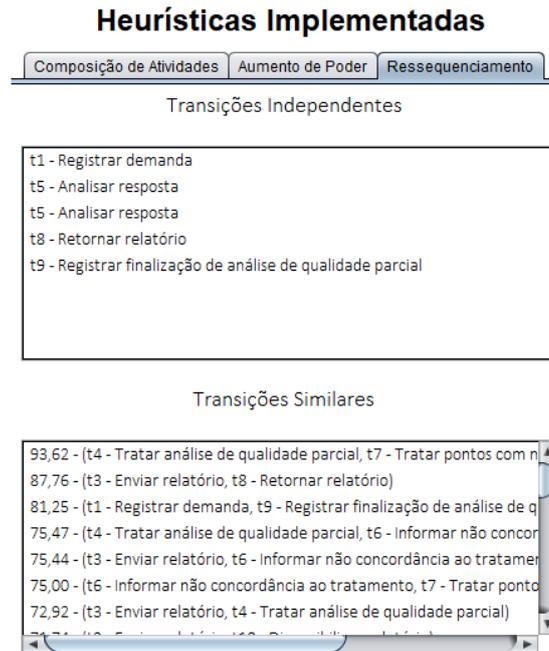


Figura 34: Resultados obtidos com a ferramenta, aplicando o algoritmo 4 ao modelo 2

A Tabela 15 apresenta os resultados obtidos destes cálculos do algoritmo 4 no modelo 3. A Figura 35 mostra os resultados obtidos com a ferramenta.

| |
|------------------------------------------|
| $S = \{ (t1, t2); (t2, t3); (t3, t4) \}$ |
| $ST = \{ t2 \}$ |

Tabela 15: Resultados obtidos manualmente, aplicando o algoritmo 4 ao modelo 3

Heurísticas Implementadas

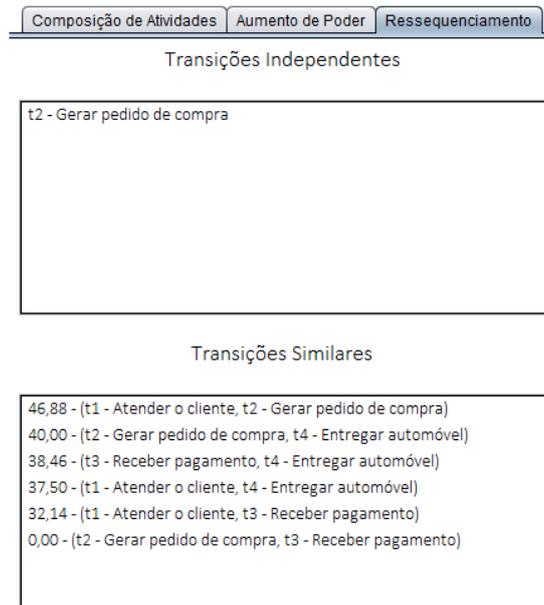


Figura 35: Resultados obtidos com a ferramenta, aplicando o algoritmo 4 ao modelo 3

4.3.5. Algoritmo 5: Transições similares

Para encontrar as transições similares pela primeira avaliação, foram feitas as seguintes etapas:

1. Identificar todas as possíveis combinações de duplas de atividades envolvidas no processo;
2. Para cada dupla, calcular a quantidade de palavras que aparecem na lista de ambas as atividades e calcular o total de palavras das duas atividades;
3. O percentual de similaridade se dará pela divisão da quantidade de palavras de ambas as atividades pela quantidade total de palavras das duas atividades;
4. Ordenar as duplas de atividades de forma decrescente a partir do percentual.

Esse algoritmo não foi aplicado no modelo 1, pois as transições não foram suficientemente detalhadas e por isso não foi construído um conjunto de palavras para esse modelo.

A Tabela 16 apresenta os resultados obtidos destes cálculos do algoritmo 5 no modelo 2 com todos os pares de transições. A Figura 36 mostra os sete primeiros resultados obtidos com a ferramenta.

| Par | Percentual | Par | Percentual | Par | Percentual |
|-----------|------------|-----------|------------|-----------|------------|
| (t4, t7) | 93,62 | (t4, t10) | 66,67 | (t1, t2) | 56,76 |
| (t3, t8) | 87,76 | (t3, t9) | 65,96 | (t5, t7) | 56,00 |
| (t1, t9) | 81,25 | (t2, t8) | 65,31 | (t7, t10) | 55,56 |
| (t4, t6) | 75,47 | (t2, t10) | 65,22 | (t2, t5) | 54,90 |
| (t3, t6) | 75,44 | (t3, t7) | 62,75 | (t6, t9) | 53,85 |
| (t6, t7) | 75,00 | (t5, t8) | 62,50 | (t1, t8) | 52,94 |
| (t3, t4) | 72,92 | (t1, t3) | 62,16 | (t6, t10) | 52,94 |
| (t3, t10) | 71,74 | (t8, t9) | 61,36 | (t1, t10) | 51,61 |
| (t4, t8) | 71,11 | (t9, t10) | 60,98 | (t2, t6) | 50,88 |
| (t6, t8) | 70,37 | (t5, t6) | 60,71 | (t2, t7) | 49,02 |
| (t2, t9) | 70,21 | (t7, t8) | 60,42 | (t1, t4) | 48,48 |
| (t8, t10) | 69,77 | (t5, t9) | 58,70 | (t7, t9) | 47,83 |
| (t3, t5) | 68,63 | (t2, t4) | 58,33 | (t1, t6) | 47,62 |
| (t4, t5) | 68,09 | (t4, t9) | 58,14 | (t1, t5) | 47,22 |
| (t2, t3) | 67,31 | (t5, t10) | 57,78 | (t1, t7) | 36,11 |

Tabela 16: Resultados obtidos manualmente, aplicando o algoritmo 5 ao modelo 2

Heurísticas Implementadas

Composição de Atividades Aumento de Poder Ressequenciamento

Transições Independentes

t1 - Registrar demanda
 t5 - Analisar resposta
 t5 - Analisar resposta
 t8 - Retornar relatório
 t9 - Registrar finalização de análise de qualidade parcial

Transições Similares

93,62 - (t4 - Tratar análise de qualidade parcial, t7 - Tratar pontos com n
 87,76 - (t3 - Enviar relatório, t8 - Retornar relatório)
 81,25 - (t1 - Registrar demanda, t9 - Registrar finalização de análise de q
 75,47 - (t4 - Tratar análise de qualidade parcial, t6 - Informar não concor
 75,44 - (t3 - Enviar relatório, t6 - Informar não concordância ao tratamen
 75,00 - (t6 - Informar não concordância ao tratamento, t7 - Tratar ponto
 72,92 - (t3 - Enviar relatório, t4 - Tratar análise de qualidade parcial)

Figura 36: Resultados obtidos com a ferramenta, aplicando o algoritmo 5 ao modelo 2

A Tabela 17 apresenta os resultados obtidos destes cálculos do algoritmo 5 no modelo 3 com todos os pares de transições. A Figura 37 mostra todos os resultados obtidos com a ferramenta.

| Par | Percentual |
|----------|------------|
| (t1, t2) | 46,88 |
| (t2, t4) | 40,00 |
| (t3, t4) | 38,46 |
| (t1, t4) | 37,50 |
| (t1, t3) | 32,14 |
| (t2, t3) | 0,00 |

Tabela 17: Resultados obtidos manualmente, aplicando o algoritmo 5 ao modelo 3

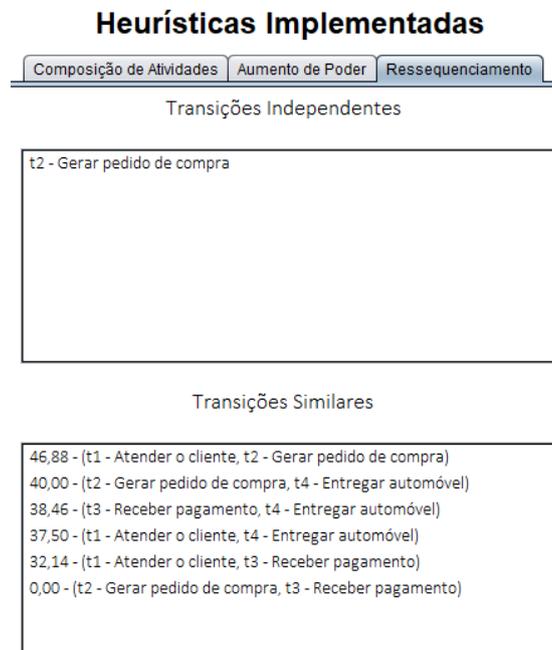


Figura 37: Resultados obtidos com a ferramenta, aplicando o algoritmo 5 ao modelo 3

4.4 Considerações finais

A partir da avaliação acima reportada, percebeu-se que os resultados obtidos manualmente foram similares aos resultados obtidos com a ferramenta, o que pode indicar que sua implementação está adequada. Por outro lado, o fato de haver poucos processos de negócio para serem testados com a ferramenta, para uma verificação de sua corretude, pode ser considerado uma limitação do experimento. Ou seja, necessitam-se de execução de mais testes para garantir a sua eficácia.

5. Conclusão

Este trabalho teve como objetivo a criação de uma ferramenta para implementar os algoritmos desenvolvidos por Souza et al. (2017) para identificação de possíveis melhorias em processos de negócio. A principal motivação é a automatização de modo a facilitar o gerenciamento de processos. Ao final, foi produzida uma ferramenta que, utilizando um arquivo XML com o processo de negócio como entrada, produz como saída possíveis pontos de melhoria, seguindo determinadas perspectivas.

5.1. Contribuições

A contribuição deste trabalho consiste na automatização da aplicação das heurísticas propostas por Souza et al. (2017), de modo a tornar mais rápida a verificação dos possíveis pontos de melhoria dos processos de negócio. Essa melhoria se traduz na diminuição do esforço do analista no trabalho de análise, principalmente no que diz respeito às heurísticas para identificar atividades finas e grossas e na identificação de transições similares, pois estas demandam um esforço de tempo muito grande no cálculo manual e com a ferramenta é possível se obter estes resultados com apenas um clique, em processos previamente modelados.

5.2. Limitações do projeto

Apesar dos resultados alcançados, não podemos deixar de mencionar as limitações do projeto. A primeira delas está relacionada com a falta de dados mais numerosos e de pessoas para efetuar os testes, de modo a garantir a integridade da implementação. A segunda limitação está relacionada aos possíveis erros na modelagem dos processos de negócio que podem gerar resultados imprecisos. Outra limitação verificada está relacionada ao padrão de composição de atividades na proposta que utiliza a visualização do *boxplot* (algoritmo 1), que em processos pequenos não gera resultados satisfatórios.

5.3. Trabalhos futuros

Este trabalho teve como foco a implementação dos algoritmos propostos por Souza et al. (2017). A ferramenta produzida utiliza arquivos em formato XML, que devem seguir um modelo específico. Porém, a criação destes arquivos não foi prevista neste projeto. Assim, a implementação de uma ferramenta que auxilie na criação e conversão do XML é uma sugestão de próximo passo a seguir.

Seria interessante apresentar a visualização do processo de negócio aplicando os resultados obtidos. Outra funcionalidade importante é a exportação dos dados de saída. Além disso, como forma de facilitar e incentivar o uso, é necessária a criação de uma interface mais amigável, que permita que as mudanças sugeridas pela ferramenta possam ser facilmente visualizadas.

Referências Bibliográficas

- Agrahari A., Mithal S., Bhat J. M. (2010) Pattern-based process optimizer. In: U.S. Patent No. 20,100,114,632. Washington, DC: U.S. Patent and Trademark Office
- Buzacott, J.A. Commonalities in reengineered business process: models and issues. *Management Science*, 1996; pp. 42(5): 768-82.
- Davenport, T. H. *Process Innovation*, Harvard Business School Press, Boston, 1993.
- Desel, J. Process modeling using petri nets. *Process-Aware Information Systems: Bridging People and Software through Process Technology*, 2005, pp. 147-177.
- Dumas, M., La Rosa, M., Mendling, J., Reijers, H.. *Fundamentals of Business Process Management*. Springer-Verlag Berlin Heidelberg, 2018.
- Dijkman R., Dumas M., García-Bañuelos L. (2009) Graph Matching Algorithms for Business Process Model Similarity Search. In: Dayal U., Eder J., Koehler J., Reijers H.A. (eds) *Business Process Management. BPM 2009. Lecture Notes in Computer Science*, vol 5701. Springer, Berlin, Heidelberg. pp. 48-63.
- Ehrig, M., Koschmider, A., Oberweis, A.: Measuring similarity between semantic business process models. In *APCCM. CRPIT 67, Austral. Comp. Soc.*, 2007, pp. 71-80.
- Forster, F. The Idea behind Business Process Improvement: Toward a Business Process Improvement Pattern Framework. 2006. Disponível em: <http://www.bptrends.com/publicationfiles/04-06-ART-PatternFramework-Forster.pdf> Acessado em Setembro, 2018.
- Gartner. Meeting the challenge: The 2009 CIO Agenda, 2009. Disponível em: <https://www.gartner.com/doc/862717> Acessado em Setembro 2018.
- Gonçalves, J. E. L., As Empresas são Grandes Coleções de Processos, *RAE - Revista de Administração de Empresas*, Jan/Mar, V.40, n.1, 2000.
- Leopold, H., Eid-Sabbagh, R., H., Mendling, J., Azevedo, L. G., Baião, F. A. Detection of naming convention violations in process models for different languages, *Decis. Support Syst.*, Jul. 2013.
- Mansar, L. S., Reijers, H. A. Best Practices in Business Process Redesign: Use and Impact. *Business Process Management Journal*, 13(2), 2007, pp. 193-213.
- Mansar, L.S., Reijers, H. A., OUNNAR, F. Development of a decision-making strategy to improve the efficiency of BPR. *Expert Systems with Applications*, 36, 2009, pp. 3248 – 3262.
- McGill, R., Tukey, J. W., Larsen, W. A. Variations of box plots. In *The American Statistician*, 1978, 32(1), pp. 12-16.

- Netjes, M. Process improvement : the creation and evaluation of process alternatives. Eindhoven: Technische Universiteit Eindhoven, 2010. DOI: 10.6100/IR685378.
- Netjes, M., Reijers, H. A., Van Der Aalst, W. M. P. The Price Tool Kit: Tool Support for Process Improvement. In M. La Rosa, editor, Business Process Management Demonstration Track: BPM 2010, volume 615, 2010. Disponível em: <<http://ceurws.org/Vol-615>> Acessado em Agosto 2018.
- Netjes, M., Mansar, S. L., Reijers, H. A., Van Der Aalst, W. M. P. Performing Business Process Redesign with Best Practices: An Evolutionary Approach. In: ICEIS, 2009, pp. 199-211. Springer Press, Heidelberg.
- Raedts, Ivo & Petkovic, Marija & S. Usenko, Yaroslav & Van der Werf, Jan Martijn & Groote, Jan & J. Somers, Lou. Transformation of BPMN Models for Behaviour Analysis, 2007, pp. 126-137.
- Reijers, H. Process Design and Redesign. In: Process-Aware Information Systems, pp. 207-234. John Wiley & Sons, Inc., Hoboken, New Jersey, USA, 2005.
- Reijers, H.A., Mansar, S.L. Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics. pp. 283–306. Omega 33, 2005.
- Seidmann, A., Sundararajan, A. The effects of task and information asymmetry on business process redesign. International Journal of Production Economics 1997; 50(2/3): pp. 117-28.
- Serrano, M. A. B., Análise de Negócio Aplicada à Modelagem de Meta Ambientes Automatizados, Tese de Doutorado, PUC-Rio, 1997.
- Souza, A., Azevedo, L. G., Santoro, F. M. Um Método para Seleção de Padrões para Melhoria de Processos de Negócio. In: VIII Simpósio Brasileiro de Sistemas de Informação, v. 1, pp. 279-290. São Paulo, Brasil. 2012.
- Souza, A., Azevedo, L. G., Santoro, F. M.. Melhoria de processos de negócio: sistematizando a seleção de padrões de redesenho. RESI : Revista Eletrônica de Sistemas de Informação, 2012, v. 11, pp. 1-20.
- Souza, A. Melhoria de processos de negócio: Sistematizando a identificação de oportunidades de melhoria. Dissertação (Mestrado em Sistemas de Informação) - Programa de Pós-Graduação em Informática (UNIRIO). Orientadores: Leonardo Guerreiro Azevedo e Flávia Maria Santoro. 2013.
- Souza, A., Azevedo, L. G., Santoro, F. M.. Automating the Identification of Opportunities for Business Process Improvement Patterns Application. International Journal of Business Process Integration and Management (IJBPIIM), Vol. 8, No. 4, 2017
- Thom, L.H, Iochpe, C., Chiao, C. Padrões de Workflow para Reuso em Modelagem de Processo de Negócio. In: Conferência Latino Americana em Linguagens de Programação, SugarloafPlop, Porto de Galinhas (Brasil): v.6 2007.

- Tsoury, A., Soffer, P., Reinhartz-Berger, I. Towards Impact Analysis of Data in Business Processes. In: Enterprise, Business-Process and Information Systems Modeling, Volume 248 of the series Lecture Notes in Business Information Processing, 2016, pp 125-140.
- Van Der Aalst, W. M. P., Ter Hofstede, A. H., Keipuszewski, B., Barros, A. P. Workflows patterns. Distributed and parallel databases 2003; 14(1), pp. 5-51.
- Weidlich, M., Dijkman, R., Mendling, J.: The iCoP framework: Identification of correspondences between process models. In: Pernici, B. (ed.) CAiSE 2010. LNCS, vol. 6051, pp. 483–498. Springer, Heidelberg.
- Weske, M., Business Process Management – Concepts, Languages, Architectures, Verlag; Berlin; Heidelberg: Springer, 2012, 403 p.
- Van Dongen, B.F., Dijkman, R.M., Mendling, J.: Measuring similarity between business process models. In: CAiSE. LNCS 5074, 2008, pp. 450-464.
- Yan, Z., Dijkman, R., Grefen, P. Fast business process similarity search with feature-based similarity estimation, in: On the Move to Meaningful Internet Systems: OTM 2010, Vol. 6426 of LNCS, Springer, pp. 60-77.

Anexo 1 - Classes do Projeto

O código completo encontra-se em: https://bitbucket.org/renataemanuelle/projeto_tcc

```
package model.heuristics;

import java.util.ArrayList;
import java.util.Collections;
import model.Transition;

/**
 *
 * @author renata.anhon
 */
public class TaskComposition {

    private final ArrayList<Transition> transitions;

    private ArrayList<Transition> smallTransitions;
    private ArrayList<Transition> largeTransitions;

    private double q1, q2, q3, iqr;

    public TaskComposition(ArrayList<Transition> transitions) {
        this.transitions = transitions;
    }

    /**
     * Sorts transitions by processing time
     * @param transitionsOrderedbyPT
     */
    private void orderTransitionsbyPT(ArrayList<Transition>
transitionsOrderedbyPT) {
        Collections.sort (transitionsOrderedbyPT, (Object o1, Object
o2) -> {
            Transition t1 = (Transition) o1;
            Transition t2 = (Transition) o2;
            return t1.getProcessingTime() < t2.getProcessingTime() ? -1
: (t1.getProcessingTime() > t2.getProcessingTime() ? +1 : 0);
        });
    }

    /**
     * Removes transitions with processing time zero or less than zero

```

```

    * @param transitionsOrderedbyPT array with transitions to be
removed
    */
    private void removeTransitionsWithPTZeroOrLess
(ArrayList<Transition> transitionsOrderedbyPT) {
        int indexAux = -1;
        for (int i = 0; i < transitionsOrderedbyPT.size(); i++) {
            if (transitionsOrderedbyPT.get(i).getProcessingTime()>0){
                indexAux = i-1;
                i = transitionsOrderedbyPT.size();
            }
        }
        while(indexAux > -1) {
            transitionsOrderedbyPT.remove(indexAux);
            indexAux--;
        }
    }

    /**
    * Based on the application of the box plot and outliers' theory
    * to identify small and large tasks considering the tasks
processing time.
    *
    * @return An array[][] where
    *         array[0] corresponds small transitions
    *         array[1] corresponds large transitions
    */
    public Transition[][] getSmallandLargeTasks() {
        ArrayList<Transition> transitionsOrderedbyPT = new
ArrayList<>(transitions);
        orderTransitionsbyPT(transitionsOrderedbyPT);
        removeTransitionsWithPTZeroOrLess(transitionsOrderedbyPT);

        int index;
        //lower quartil
        index = (int)((transitionsOrderedbyPT.size()+1)/4);
        if(index > 0) index--;
        q1 = transitionsOrderedbyPT.get(index).getProcessingTime();

        //median
        if (transitionsOrderedbyPT.size() % 2 == 0) {
            q2 =
(transitionsOrderedbyPT.get((transitionsOrderedbyPT.size()/2)-
1).getProcessingTime() +
transitionsOrderedbyPT.get(transitionsOrderedbyPT.size()/2).getProcess
ingTime()) / 2;
        } else {

```

```

        q2 =
transitionsOrderedbyPT.get (transitionsOrderedbyPT.size()/2).getProcess
ingTime();
    }

    //upper quartil
    index = (int) (((transitionsOrderedbyPT.size()+1)*3)/4);
    if(index > 0) index--;
    q3 = transitionsOrderedbyPT.get (index).getProcessingTime();

    //interquartil range
    iqr = q3 - q1;

    smallTransitions = new ArrayList<>();
    largeTransitions = new ArrayList<>();

    transitionsOrderedbyPT.stream().forEach((t) -> {
        if (t.getProcessingTime() < (q1 - 1.5*iqr)) {
            smallTransitions.add(t);
        } else if (t.getProcessingTime() > (q3 + 1.5*iqr)) {
            largeTransitions.add(t);
        }
    });

    Transition[][] t = {smallTransitions.toArray(new
Transition[smallTransitions.size()]),
                        largeTransitions.toArray(new
Transition[largeTransitions.size()])};
    return t;

}

/**
 * Based on the analyst experience to determine the distance from
the median
 * of processing times to be used as definition to classify tasks
as small or large.
 *
 * @param distance distance percentage from the median
 * @return An array[][] where
 *         array[0] corresponds small transitions
 *         array[1] corresponds large transitions
 */
public Transition[][] getSmallandLargeTasks(double distance) {
    ArrayList<Transition> transitionsOrderedbyPT = new
ArrayList<>(transitions);
    orderTransitionsbyPT(transitionsOrderedbyPT);
    removeTransitionsWithPTZeroOrLess(transitionsOrderedbyPT);

```

```

        //median
        if (transitionsOrderedbyPT.size() % 2 == 0) {
            q2 =
(transitionsOrderedbyPT.get((transitionsOrderedbyPT.size()/2)-
1).getProcessingTime() +
transitionsOrderedbyPT.get(transitionsOrderedbyPT.size()/2).getProcess
ingTime()) / 2;
        } else {
            q2 =
transitionsOrderedbyPT.get(transitionsOrderedbyPT.size()/2).getProcess
ingTime();
        }

        smallTransitions = new ArrayList<>();
        largeTransitions = new ArrayList<>();

        transitionsOrderedbyPT.stream().forEach((t) -> {
            if (t.getProcessingTime() < ((1-distance)*q2)) {
                smallTransitions.add(t);
            } else if (t.getProcessingTime() > (1+distance)*q2) {
                largeTransitions.add(t);
            }
        });

        Transition[][] t = {smallTransitions.toArray(new
Transition[smallTransitions.size()]),
                            largeTransitions.toArray(new
Transition[largeTransitions.size()])};
        return t;
    }
}

```

```

package model.heuristics;

import java.util.ArrayList;
import model.Place;
import model.BusinessProcess;
import model.Transition;

/**
 *
 * @author Renata Anhon
 */
public class Empower {

```

```

private final BusinessProcess process;

public Empower(BusinessProcess process) {
    this.process = process;
}

public ArrayList<Transition> getEmpowerTransitions(){
    //placelist is a linked list where each element is a place
    with two set attributes
    //previousTransitions and nextTransitions

    //transitionList is a linked list where each element is a
    transition with two set
    //attributes previousPlaces and nextPlaces

    //array of places with only 1 previous transition and at least
    2 following transition
    ArrayList<String> ds = new ArrayList<>();

    //array of transitions with only 1 previous place and the
    following place is in ds
    ArrayList<Transition> dt = new ArrayList<>();

    //array of transitions to indicate empower redesign pattern
    application
    ArrayList<Transition> d = new ArrayList<>();

    for (Place p : process.getPlaces()) {
        if(p.getPreviousTransitions().length==1 &&
        p.getNextTransitions().length >=2) {
            ds.add(p.getId());
        }
    }

    for (Transition t : process.getTransitions()) {
        if(t.getPreviousPlaces().length==1) {
            for (String p : t.getNextPlaces()){
                if (ds.contains(p)) {
                    dt.add(t);
                }
            }
        }
    }

    for (Transition t1: process.getTransitions()) {
        for (Transition t2: dt) {
            for (String p1 : t1.getNextPlaces()) {

```



```

        for (String p11 : transition1.getNextPlaces()) {
            for (String p12 : transition2.getPreviousPlaces())
            {
                if (p11.equalsIgnoreCase(p12)) {
                    intersection_places = true;
                }
            }
        }
        if (intersection_places) {
            for (String str1 : transition1.getOutputs()) {
                for (String str2 : transition2.getInputs()) {
                    if (str1.equalsIgnoreCase(str2)) {
                        intersection_inputOutput = true;
                    }
                }
            }
        }
        if (intersection_places && !intersection_inputOutput)
        {
            st.add(transition1);
        }
        intersection_places = false;
        intersection_inputOutput = false;
    }
}

return st;
}

```

```

public LinkedHashMap<String, Double> getSimilarTransitionsRank(){

```

```

    //TP: Array, is the list of similar transitions and the
    percentage of similarity

```

```

    LinkedHashMap<String, Double> TP = new LinkedHashMap<>();
    LinkedHashMap<String, Double> tpOrdenado = new
    LinkedHashMap<>();

```

```

    int intersection_quantity, union_quantity;
    Transition transition1, transition2;

```

```

    for(int i = 0; i < transitions.size(); i++) {
        transition1 = transitions.get(i);
        for (int j = i+1; j < transitions.size(); j++) {
            transition2 = transitions.get(j);
            intersection_quantity = 0;
            union_quantity = 0;
            for (String word1 :
transition1.getSetOfWords().keySet()) {

```

```

        union_quantity +=
transition1.getSetOfWords().get(word1);
        for (String word2 :
transition2.getSetOfWords().keySet()) {
            if(word1.equalsIgnoreCase(word2)) {
                intersection_quantity +=
transition1.getSetOfWords().get(word1) +
transition2.getSetOfWords().get(word2);
            }
        }
        for (String word2 :
transition2.getSetOfWords().keySet()) {
            union_quantity +=
transition2.getSetOfWords().get(word2);
        }
        if (union_quantity == 0) {
            TP.put("(" + transition1.getId() + " -
"+transition1.getName() + ", " + transition2.getId() + " -
"+transition2.getName() + ") ", 0.0);
        } else {
            TP.put("(" + transition1.getId() + " -
"+transition1.getName() + ", " + transition2.getId() + " -
"+transition2.getName() + ") ", (double) (intersection_quantity) /
(double) (union_quantity)*100);
        }
    }
}

while (!TP.isEmpty()) {
    String str = null;
    double valor = -1;
    for (String st : TP.keySet()) {
        if(TP.get(st) > valor) {
            str = st;
            valor = TP.get(st);
        }
    }
    tpOrdenado.put(str, valor);
    TP.remove(str);
}

return tpOrdenado;
}
}

```

```

package model;

import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;
import javax.xml.bind.annotation.XmlElement;
import javax.xml.bind.annotation.XmlRootElement;
import javax.xml.bind.annotation.XmlType;

/**
 *
 * @author Renata Anhon
 */

@XmlRootElement
@XmlType(propOrder = {"transitions", "places", "arcs"})
public class BusinessProcess {

    private List<Transition> transitions;
    private List<Place> places;
    private List<Arc> arcs;

    //Constructors
    public BusinessProcess() {
        transitions = new ArrayList<>();
        places = new ArrayList<>();
        arcs = new ArrayList<>();
    }

    public BusinessProcess(ArrayList<Transition> transitions,
ArrayList<Place> places, ArrayList<Arc> arcs) {
        this.transitions = transitions;
        this.places = places;
        this.arcs = arcs;
    }

    //Getters and Setters
    @XmlElement
    public List<Transition> getTransitions() {
        return transitions;
    }

    public void setTransitions(ArrayList<Transition> transitions) {
        this.transitions = transitions;
    }

    @XmlElement
    public List<Place> getPlaces() {
        return places;
    }
}

```

```

    }

    public void setPlaces(ArrayList<Place> places) {
        this.places = places;
    }

    @XmlElement
    public List<Arc> getArcs() {
        return arcs;
    }

    public void setArcs(ArrayList<Arc> arcs) {
        this.arcs = arcs;
    }

    public void addTransition(Transition transition) {
        transitions.add(transition);
    }

    public void addPlace(Place place) {
        places.add(place);
    }

    public void addArc(Arc arc) {
        arcs.add(arc);
    }

    @Override
    public String toString() {
        return "BusinessProcess:"
            + "\n-----\n"
            + "transitions:\n" + transitions
            + "\n-----\n"
            + "places:\n" + places
            + "\n-----\n"
            + "arcs:\n" + arcs + "\n";
    }
}

```

```

package model;

```

```

import java.util.Arrays;
import java.util.LinkedHashMap;
import java.util.Map;
import javax.xml.bind.annotation.XmlElement;
import javax.xml.bind.annotation.XmlRootElement;

```

```

import javax.xml.bind.annotation.XmlType;

/**
 *
 * @author Renata Anhon
 */

@XmlRootElement
@XmlType(propOrder = {"id", "name", "description", "processingTime",
"inputs", "outputs", "previousPlaces", "nextPlaces", "role",
"setOfWords"})
public class Transition {
    private String id;
    private String name;

    private String description;

    private int processingTime;

    private String[] inputs = new String[0];
    private String[] outputs = new String[0];

    private String[] previousPlaces = new String[0];
    private String[] nextPlaces = new String[0];

    private String role;

    private Map <String, Integer> setOfWords = new LinkedHashMap<>();

    //Constructors
    public Transition() {
    }

    public Transition(String id, String name, String description, int
processingTime, String[] input, String[] output, String[]
previousPlaces, String[] nextPlaces, String role, Map<String, Integer>
setOfWords) {
        this.id = id;
        this.name = name;
        this.description = description;
        this.processingTime = processingTime;
        this.inputs = input;
        this.outputs = output;
        this.previousPlaces = previousPlaces;
        this.nextPlaces = nextPlaces;
        this.role = role;
        this.setOfWords = setOfWords;
    }
}

```

```

//Getters and Setters
@XmlElement
public String getId() {
    return id;
}

public void setId(String id) {
    this.id = id;
}

@XmlElement
public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}

@XmlElement
public String getDescription() {
    return description;
}

public void setDescription(String description) {
    this.description = description;
}

@XmlElement
public int getProcessingTime() {
    return processingTime;
}

public void setProcessingTime(int processingTime) {
    this.processingTime = processingTime;
}

@XmlElement
public String[] getInputs() {
    return inputs;
}

public void setInputs(String[] inputs) {
    this.inputs = inputs;
}

@XmlElement
public String[] getOutputs() {
    return outputs;
}

```

```

}

public void setOutputs(String[] outputs) {
    this.outputs = outputs;
}

@XmlElement
public String[] getPreviousPlaces() {
    return previousPlaces;
}

public void setPreviousPlaces(String[] previousPlaces) {
    this.previousPlaces = previousPlaces;
}

@XmlElement
public String[] getNextPlaces() {
    return nextPlaces;
}

public void setNextPlaces(String[] nextPlaces) {
    this.nextPlaces = nextPlaces;
}

@XmlElement
public String getRole() {
    return role;
}

public void setRole(String role) {
    this.role = role;
}

@XmlElement
public Map <String, Integer> getSetOfWords() {
    return setOfWords;
}

public void setSetOfWords(Map <String, Integer> setOfWords) {
    this.setOfWords = setOfWords;
}

public void addNextPlace(String placeId) {
    String[] aux = new String[nextPlaces.length + 1];
    System.arraycopy(nextPlaces, 0, aux, 0, nextPlaces.length);
    aux[aux.length - 1] = placeId;
    nextPlaces = aux;
}

```

```

    public void addPreviousPlace(String placeId) {
        String[] aux = new String[previousPlaces.length + 1];
        System.arraycopy(previousPlaces, 0, aux, 0,
previousPlaces.length);
        aux[aux.length - 1] = placeId;
        previousPlaces = aux;
    }

    public void addInput(String input) {
        String[] aux = new String[inputs.length + 1];
        System.arraycopy(inputs, 0, aux, 0, inputs.length);
        aux[aux.length - 1] = input;
        inputs = aux;
    }

    public void addOutput(String output) {
        String[] aux = new String[outputs.length + 1];
        System.arraycopy(outputs, 0, aux, 0, outputs.length);
        aux[aux.length - 1] = output;
        outputs = aux;
    }

    public void addWord(String word, int quantity) {
        setOfWords.put(word, quantity);
    }

    @Override
    public String toString() {
        return "\nTransition id: " + id + "\n"
            + "name: " + name + "\n"
            + "description: " + description + "\n"
            + "processingTime: " + processingTime + "\n"
            + "inputs: " + Arrays.toString(inputs) + "\n"
            + "outputs: " + Arrays.toString(outputs) + "\n"
            + "previousPlaces: " + Arrays.toString(previousPlaces)
+ "\n"
            + "nextPlaces: " + Arrays.toString(nextPlaces) + "\n"
            + "role: " + role + "\n"
            + "setOfWords: " + setOfWords;
    }
}
}

```

```

package model;

```

```

import java.util.Arrays;
import javax.xml.bind.annotation.XmlElement;

```

```

import javax.xml.bind.annotation.XmlRootElement;
import javax.xml.bind.annotation.XmlType;

/**
 *
 * @author renata.anhon
 */

@XmlRootElement
@XmlType(propOrder = {"id", "name", "previousTransitions",
"nextTransitions"})
public class Place {

    private String id;
    private String name;
    private String[] previousTransitions = new String[0];
    private String[] nextTransitions = new String[0];

    //Constructor
    public Place() {
    }

    public Place(String id, String name, String[] previousTransitions,
String[] nextTransitions) {
        this.id = id;
        this.name = name;
        this.previousTransitions = previousTransitions;
        this.nextTransitions = nextTransitions;
    }

    public Place(String id, String name) {
        this.id = id;
        this.name = name;
    }

    //Getters and Setters
    @XmlElement
    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    @XmlElement
    public String getName() {
        return name;
    }
}

```

```

public void setName(String name) {
    this.name = name;
}

@XmlElement
public String[] getPreviousTransitions() {
    return previousTransitions;
}

public void setPreviousTransitions(String[] previousTransitions) {
    this.previousTransitions = previousTransitions;
}

@XmlElement
public String[] getNextTransitions() {
    return nextTransitions;
}

public void setNextTransitions(String[] nextTransitions) {
    this.nextTransitions = nextTransitions;
}

public void addNextTransition(String transitionId) {
    String[] aux = new String[nextTransitions.length + 1];
    System.arraycopy(nextTransitions, 0, aux, 0,
nextTransitions.length);
    aux[aux.length - 1] = transitionId;
    nextTransitions = aux;
}

public void addPreviousTransition(String transitionId) {
    String[] aux = new String[previousTransitions.length + 1];
    System.arraycopy(previousTransitions, 0, aux, 0,
previousTransitions.length);
    aux[aux.length - 1] = transitionId;
    previousTransitions = aux;
}

@Override
public String toString() {
    return "\nPlace id: " + id + "\n"
        + "name: " + name + "\n"
        + "previousTransitions: " +
Arrays.toString(previousTransitions) + "\n"
        + "nextTransitions: " +
Arrays.toString(nextTransitions);
}

```

```
}
```

```
package model;

import javax.xml.bind.annotation.XmlElement;
import javax.xml.bind.annotation.XmlRootElement;
import javax.xml.bind.annotation.XmlType;

/**
 *
 * @author Renata
 */

@XmlRootElement
@XmlType(propOrder = {"source", "target"})
public class Arc {

    private String source;
    private String target;

    public Arc() {
    }

    public Arc(String source, String target) {
        this.source = source;
        this.target = target;
    }

    @XmlElement
    public String getSource() {
        return source;
    }

    public void setSource(String source) {
        this.source = source;
    }

    @XmlElement
    public String getTarget() {
        return target;
    }

    public void setTarget(String target) {
        this.target = target;
    }
}
```

}

Anexo 2 - XML dos modelos

Modelo 1:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<businessProcess>
  <transitions>
    <id>t1</id>
    <name>check stock availability</name>
    <processingTime>1</processingTime>
    <inputs>purchase order</inputs>
    <inputs>warehouse db</inputs>
    <previousPlaces>p1</previousPlaces>
    <nextPlaces>p2</nextPlaces>
    <role>WD / ERPSysyem</role>
    <setOfWords/>
  </transitions>
  <transitions>
    <id>t2</id>
    <name>retrieve product from warehouse</name>
    <processingTime>20</processingTime>
    <inputs>products warehouse</inputs>
    <outputs>product</outputs>
    <previousPlaces>p2</previousPlaces>
    <nextPlaces>p7</nextPlaces>
    <role>Warehouse Distribution</role>
    <setOfWords/>
  </transitions>
  <transitions>
    <id>t3</id>
    <name>check raw materials availability</name>
    <processingTime>1</processingTime>
    <inputs>purchase order</inputs>
    <inputs>suppliers catalog</inputs>
    <previousPlaces>p2</previousPlaces>
    <nextPlaces>p3</nextPlaces>
    <role>WD / ERPSysyem</role>
    <setOfWords/>
  </transitions>
  <transitions>
    <id>t4</id>
    <name>request raw materials from supplier 1</name>
```

```

    <processingTime>5</processingTime>
    <previousPlaces>p3</previousPlaces>
    <nextPlaces>p4</nextPlaces>
    <role>Warehouse Distribution</role>
    <setOfWords/>
</transitions>
<transitions>
  <id>t5</id>
  <name>obtain raw materials from supplier 1</name>
  <processingTime>12</processingTime>
  <outputs>raw materials</outputs>
  <previousPlaces>p4</previousPlaces>
  <nextPlaces>p6</nextPlaces>
  <role>Warehouse Distribution</role>
  <setOfWords/>
</transitions>
<transitions>
  <id>t6</id>
  <name>request raw materials from supplier 2</name>
  <processingTime>5</processingTime>
  <previousPlaces>p3</previousPlaces>
  <nextPlaces>p5</nextPlaces>
  <role>Warehouse Distribution</role>
  <setOfWords/>
</transitions>
<transitions>
  <id>t7</id>
  <name>raw materials provided by supplier 2</name>
  <processingTime>15</processingTime>
  <outputs>raw materials</outputs>
  <previousPlaces>p5</previousPlaces>
  <nextPlaces>p6</nextPlaces>
  <role>Warehouse Distribution</role>
  <setOfWords/>
</transitions>
<transitions>
  <id>t8</id>
  <name>manufacture product</name>
  <processingTime>50</processingTime>
  <inputs>raw materials</inputs>
  <outputs>product</outputs>
  <previousPlaces>p6</previousPlaces>
  <nextPlaces>p7</nextPlaces>
  <role>Warehouse Distribution</role>
</transitions>
<transitions>
  <id>t9</id>
  <name>confirm order</name>
  <processingTime>3</processingTime>

```

```

    <inputs>purchase order</inputs>
    <outputs>purchase order (confirmed)</outputs>
    <previousPlaces>p7</previousPlaces>
    <nextPlaces>p8</nextPlaces>
    <nextPlaces>p11</nextPlaces>
    <role>Sales</role>
</transitions>
<transitions>
  <id>t10</id>
  <name>emit invoice</name>
  <processingTime>8</processingTime>
  <outputs>invoice</outputs>
  <previousPlaces>p8</previousPlaces>
  <nextPlaces>p9</nextPlaces>
  <role>Sales</role>
</transitions>
<transitions>
  <id>t11</id>
  <name>receive payment</name>
  <processingTime>480</processingTime>
  <inputs>purchase order (confirmed)</inputs>
  <inputs>payment</inputs>
  <outputs>purchase order (paid)</outputs>
  <previousPlaces>p9</previousPlaces>
  <nextPlaces>p10</nextPlaces>
  <role>Sales</role>
  <setOfWords/>
</transitions>
<transitions>
  <id>t12</id>
  <name>get shipping adress</name>
  <processingTime>10</processingTime>
  <outputs>shipment address</outputs>
  <previousPlaces>p11</previousPlaces>
  <nextPlaces>p12</nextPlaces>
  <role>Warehouse Distribution</role>
  <setOfWords/>
</transitions>
<transitions>
  <id>t13</id>
  <name>ship product</name>
  <processingTime>960</processingTime>
  <inputs>product</inputs>
  <inputs>shipment address</inputs>
  <previousPlaces>p12</previousPlaces>
  <nextPlaces>p13</nextPlaces>
  <role>Warehouse Distribution</role>
  <setOfWords/>
</transitions>

```

```

<transitions>
  <id>t14</id>
  <name>arquive order</name>
  <processingTime>3</processingTime>
  <inputs>purchase order (paid)</inputs>
  <outputs>orders db</outputs>
  <previousPlaces>p10</previousPlaces>
  <previousPlaces>p13</previousPlaces>
  <nextPlaces>p14</nextPlaces>
  <role>Sales</role>
  <setOfWords/>
</transitions>
<places>
  <id>p1</id>
  <name></name>
  <previousTransitions></previousTransitions>
  <nextTransitions>t1</nextTransitions>
</places>
<places>
  <id>p2</id>
  <name></name>
  <previousTransitions>t1</previousTransitions>
  <nextTransitions>t2</nextTransitions>
  <nextTransitions>t3</nextTransitions>
</places>
<places>
  <id>p3</id>
  <name></name>
  <previousTransitions>t3</previousTransitions>
  <nextTransitions>t4</nextTransitions>
  <nextTransitions>t6</nextTransitions>
</places>
<places>
  <id>p4</id>
  <name></name>
  <previousTransitions>t4</previousTransitions>
  <nextTransitions>t5</nextTransitions>
</places>
<places>
  <id>p5</id>
  <name></name>
  <previousTransitions>t6</previousTransitions>
  <nextTransitions>t7</nextTransitions>
</places>
<places>
  <id>p6</id>
  <name></name>
  <previousTransitions>t5</previousTransitions>
  <previousTransitions>t7</previousTransitions>

```

```

        <nextTransitions>t8</nextTransitions>
</places>
<places>
  <id>p7</id>
  <name></name>
  <previousTransitions>t2</previousTransitions>
  <previousTransitions>t8</previousTransitions>
  <nextTransitions>t9</nextTransitions>
</places>
<places>
  <id>p8</id>
  <name></name>
  <previousTransitions>t9</previousTransitions>
  <nextTransitions>t10</nextTransitions>
</places>
<places>
  <id>p9</id>
  <name></name>
  <previousTransitions>t10</previousTransitions>
  <nextTransitions>t11</nextTransitions>
</places>
<places>
  <id>p10</id>
  <name></name>
  <previousTransitions>t11</previousTransitions>
  <nextTransitions>t14</nextTransitions>
</places>
<places>
  <id>p11</id>
  <name></name>
  <previousTransitions>t9</previousTransitions>
  <nextTransitions>t12</nextTransitions>
</places>
<places>
  <id>p12</id>
  <name></name>
  <previousTransitions>t12</previousTransitions>
  <nextTransitions>t13</nextTransitions>
</places>
<places>
  <id>p13</id>
  <name></name>
  <previousTransitions>t13</previousTransitions>
  <nextTransitions>t14</nextTransitions>
</places>
<places>
  <id>p14</id>
  <name></name>
  <previousTransitions>t14</previousTransitions>

```

```

        <nextTransitions></nextTransitions>
    </places>
</businessProcess>

```

Modelo 2:

```

<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<businessProcess>
    <transitions>
        <id>t1</id>
        <name>Registrar demanda</name>
        <description>Registrar demanda no Jira.</description>
        <processingTime>10</processingTime>
        <outputs>Acompanhamento de análise de qualidade
parcial</outputs>
        <previousPlaces>p1</previousPlaces>
        <nextPlaces>p2</nextPlaces>
        <role>Analista de qualidade</role>
        <setOfWords>
            <entry>
                <key>registrar</key>
                <value>2</value>
            </entry>
            <entry>
                <key>demanda</key>
                <value>2</value>
            </entry>
            <entry>
                <key>qualidade</key>
                <value>2</value>
            </entry>
            <entry>
                <key>jira</key>
                <value>1</value>
            </entry>
            <entry>
                <key>acompanhamento</key>
                <value>1</value>
            </entry>
            <entry>
                <key>analise</key>
                <value>1</value>
            </entry>
            <entry>
                <key>parcial</key>
                <value>1</value>
            </entry>
        </setOfWords>
    </transitions>
</businessProcess>

```

```

        <entry>
            <key>analista</key>
            <value>1</value>
        </entry>
    </setOfWords>
</transitions>
<transitions>
    <id>t2</id>
    <name>Realizar análise de qualidade parcial</name>
    <description>Realizar a análise de qualidade parcial dos
processos modelados previstos no cronograma da iniciativa de
modelagem.</description>
    <processingTime>1440</processingTime>
    <inputs>Modelos de processo</inputs>
    <outputs>Relatório de análise de qualidade parcial</outputs>
    <outputs>Acompanhamento de análise de qualidade
parcial</outputs>
    <previousPlaces>p2</previousPlaces>
    <nextPlaces>p3</nextPlaces>
    <role>Analista de qualidade</role>
    <setOfWords>
        <entry>
            <key>realizar</key>
            <value>2</value>
        </entry>
        <entry>
            <key>analise</key>
            <value>4</value>
        </entry>
        <entry>
            <key>qualidade</key>
            <value>5</value>
        </entry>
        <entry>
            <key>parcial</key>
            <value>4</value>
        </entry>
        <entry>
            <key>processos</key>
            <value>1</value>
        </entry>
        <entry>
            <key>modelados</key>
            <value>1</value>
        </entry>
        <entry>
            <key>previstos</key>
            <value>1</value>
        </entry>
    </setOfWords>
</transitions>
</transitions>

```

```

    <entry>
      <key>cronograma</key>
      <value>1</value>
    </entry>
    <entry>
      <key>iniciativa</key>
      <value>1</value>
    </entry>
    <entry>
      <key>modelagem</key>
      <value>1</value>
    </entry>
    <entry>
      <key>modelos</key>
      <value>1</value>
    </entry>
    <entry>
      <key>processo</key>
      <value>1</value>
    </entry>
    <entry>
      <key>relatorio</key>
      <value>1</value>
    </entry>
    <entry>
      <key>acompanhamento</key>
      <value>1</value>
    </entry>
    <entry>
      <key>analista</key>
      <value>1</value>
    </entry>
  </setOfWords>
</transitions>
<transitions>
  <id>t3</id>
  <name>Enviar relatório</name>
  <description>Enviar, via e-mail ou JIRA, o relatório da
análise de qualidade parcial para o líder de projeto. </description>
  <processingTime>15</processingTime>
  <inputs>Relatório de análise de qualidade parcial</inputs>
  <outputs>Relatório de análise de qualidade parcial</outputs>
  <outputs>Acompanhamento de análise de qualidade
parcial</outputs>
  <previousPlaces>p3</previousPlaces>
  <nextPlaces>p4</nextPlaces>
  <role>Analista de qualidade</role>
  <setOfWords>
    <entry>

```

```
        <key>enviar</key>
        <value>2</value>
</entry>
<entry>
    <key>relatorio</key>
    <value>4</value>
</entry>
<entry>
    <key>via</key>
    <value>1</value>
</entry>
<entry>
    <key>email</key>
    <value>1</value>
</entry>
<entry>
    <key>jira</key>
    <value>1</value>
</entry>
<entry>
    <key>analise</key>
    <value>4</value>
</entry>
<entry>
    <key>qualidade</key>
    <value>5</value>
</entry>
<entry>
    <key>parcial</key>
    <value>4</value>
</entry>
<entry>
    <key>lider</key>
    <value>1</value>
</entry>
<entry>
    <key>projeto</key>
    <value>1</value>
</entry>
<entry>
    <key>acompanhamento</key>
    <value>1</value>
</entry>
<entry>
    <key>analista</key>
    <value>1</value>
</entry>
</setOfWords>
</transitions>
```

```

<transitions>
  <id>t4</id>
  <name>Tratar análise de qualidade parcial</name>
  <description>Tratar os pontos de atenção e de não concordância
da análise de qualidade parcial. </description>
  <processingTime>960</processingTime>
  <inputs>Relatório de análise de qualidade parcial</inputs>
  <outputs>Relatório de análise de qualidade parcial</outputs>
  <previousPlaces>p4</previousPlaces>
  <nextPlaces>p5</nextPlaces>
  <role>Líder de projeto</role>
  <setOfWords>
    <entry>
      <key>tratar</key>
      <value>2</value>
    </entry>
    <entry>
      <key>analise</key>
      <value>4</value>
    </entry>
    <entry>
      <key>qualidade</key>
      <value>4</value>
    </entry>
    <entry>
      <key>parcial</key>
      <value>4</value>
    </entry>
    <entry>
      <key>pontos</key>
      <value>1</value>
    </entry>
    <entry>
      <key>atencao</key>
      <value>1</value>
    </entry>
    <entry>
      <key>nao</key>
      <value>1</value>
    </entry>
    <entry>
      <key>concordancia</key>
      <value>1</value>
    </entry>
    <entry>
      <key>relatorio</key>
      <value>2</value>
    </entry>
  </setOfWords>
</transitions>

```

```

        <key>lider</key>
        <value>1</value>
    </entry>
    <entry>
        <key>projeto</key>
        <value>1</value>
    </entry>
</setOfWords>
</transitions>
<transitions>
    <id>t5</id>
    <name>Analisar resposta</name>
    <description>Analisar as respostas do líder de projeto aos
pontos de atenção da análise de qualidade, verificando se os mesmos
foram alterados no ARIS. </description>
    <processingTime>480</processingTime>
    <inputs>Relatório de análise de qualidade parcial</inputs>
    <outputs>Acompanhamento de análise de qualidade
parcial</outputs>
    <previousPlaces>p5</previousPlaces>
    <nextPlaces>p6</nextPlaces>
    <role>Analista de qualidade</role>
    <setOfWords>
        <entry>
            <key>analisar</key>
            <value>2</value>
        </entry>
        <entry>
            <key>resposta</key>
            <value>1</value>
        </entry>
        <entry>
            <key>respostas</key>
            <value>1</value>
        </entry>
        <entry>
            <key>lider</key>
            <value>1</value>
        </entry>
        <entry>
            <key>projeto</key>
            <value>1</value>
        </entry>
        <entry>
            <key>pontos</key>
            <value>1</value>
        </entry>
        <entry>
            <key>atencao</key>

```

```

        <value>1</value>
    </entry>
    <entry>
        <key>analise</key>
        <value>3</value>
    </entry>
    <entry>
        <key>qualidade</key>
        <value>4</value>
    </entry>
    <entry>
        <key>verificando</key>
        <value>1</value>
    </entry>
    <entry>
        <key>mesmos</key>
        <value>1</value>
    </entry>
    <entry>
        <key>foram</key>
        <value>1</value>
    </entry>
    <entry>
        <key>alterados</key>
        <value>1</value>
    </entry>
    <entry>
        <key>aris</key>
        <value>1</value>
    </entry>
    <entry>
        <key>relatorio</key>
        <value>1</value>
    </entry>
    <entry>
        <key>parcial</key>
        <value>2</value>
    </entry>
    <entry>
        <key>acompanhamento</key>
        <value>1</value>
    </entry>
    <entry>
        <key>analista</key>
        <value>1</value>
    </entry>
</setOfWords>
</transitions>
<transitions>

```

```

<id>t6</id>
<name>Informar não concordância ao tratamento</name>
<description>Informar via e-mail ou JIRA, ao líder de projeto
e ao líder de AMPN local a não concordância ao tratamento dos pontos
de atenção da análise de qualidade parcial.</description>
<processingTime>15</processingTime>
<outputs>Relatório de análise de qualidade parcial</outputs>
<outputs>Acompanhamento de análise de qualidade
parcial</outputs>
<previousPlaces>p6</previousPlaces>
<nextPlaces>p7</nextPlaces>
<role>Analista de qualidade</role>
<setOfWords>
  <entry>
    <key>informar</key>
    <value>2</value>
  </entry>
  <entry>
    <key>nao</key>
    <value>2</value>
  </entry>
  <entry>
    <key>concordancia</key>
    <value>2</value>
  </entry>
  <entry>
    <key>tratamento</key>
    <value>2</value>
  </entry>
  <entry>
    <key>via</key>
    <value>1</value>
  </entry>
  <entry>
    <key>email</key>
    <value>1</value>
  </entry>
  <entry>
    <key>jira</key>
    <value>1</value>
  </entry>
  <entry>
    <key>lider</key>
    <value>2</value>
  </entry>
  <entry>
    <key>projeto</key>
    <value>1</value>
  </entry>
</setOfWords>

```

```

    <entry>
      <key>ampn</key>
      <value>1</value>
    </entry>
    <entry>
      <key>local</key>
      <value>1</value>
    </entry>
    <entry>
      <key>pontos</key>
      <value>1</value>
    </entry>
    <entry>
      <key>atencao</key>
      <value>1</value>
    </entry>
    <entry>
      <key>analise</key>
      <value>3</value>
    </entry>
    <entry>
      <key>qualidade</key>
      <value>4</value>
    </entry>
    <entry>
      <key>parcial</key>
      <value>3</value>
    </entry>
    <entry>
      <key>relatorio</key>
      <value>1</value>
    </entry>
    <entry>
      <key>acompanhamento</key>
      <value>1</value>
    </entry>
    <entry>
      <key>analista</key>
      <value>1</value>
    </entry>
  </setOfWords>
</transitions>
<transitions>
  <id>t7</id>
  <name>Tratar pontos com não concordância</name>
  <description>Tratar os pontos de atenção da análise de
qualidade parcial em que há não concordância da AMPN
Sede.</description>
  <processingTime>960</processingTime>

```

```
<inputs>Relatório de análise de qualidade parcial</inputs>
<outputs>Relatório de análise de qualidade parcial</outputs>
<previousPlaces>p7</previousPlaces>
<nextPlaces>p5</nextPlaces>
<role>Líder de projeto</role>
<setOfWords>
  <entry>
    <key>tratar</key>
    <value>2</value>
  </entry>
  <entry>
    <key>pontos</key>
    <value>2</value>
  </entry>
  <entry>
    <key>nao</key>
    <value>2</value>
  </entry>
  <entry>
    <key>concordancia</key>
    <value>2</value>
  </entry>
  <entry>
    <key>atencao</key>
    <value>1</value>
  </entry>
  <entry>
    <key>analise</key>
    <value>3</value>
  </entry>
  <entry>
    <key>qualidade</key>
    <value>3</value>
  </entry>
  <entry>
    <key>parcial</key>
    <value>3</value>
  </entry>
  <entry>
    <key>ha</key>
    <value>1</value>
  </entry>
  <entry>
    <key>ampn</key>
    <value>1</value>
  </entry>
  <entry>
    <key>sede</key>
    <value>1</value>
  </entry>
</setOfWords>
```

```

    </entry>
    <entry>
      <key>relatorio</key>
      <value>2</value>
    </entry>
    <entry>
      <key>lider</key>
      <value>1</value>
    </entry>
    <entry>
      <key>projeto</key>
      <value>1</value>
    </entry>
  </setOfWords>
</transitions>
<transitions>
  <id>t8</id>
  <name>Retornar relatório</name>
  <description>Retornar via e-mail ou JIRA, o relatório de
análise de qualidade parcial de processos ao líder de projeto.
</description>
  <processingTime>15</processingTime>
  <inputs>Relatório de análise de qualidade parcial</inputs>
  <outputs>Relatório de análise de qualidade parcial</outputs>
  <previousPlaces>p6</previousPlaces>
  <nextPlaces>p8</nextPlaces>
  <role>Analista de qualidade</role>
  <setOfWords>
    <entry>
      <key>retornar</key>
      <value>2</value>
    </entry>
    <entry>
      <key>relatorio</key>
      <value>4</value>
    </entry>
    <entry>
      <key>via</key>
      <value>1</value>
    </entry>
    <entry>
      <key>email</key>
      <value>1</value>
    </entry>
    <entry>
      <key>jira</key>
      <value>1</value>
    </entry>
  </setOfWords>
</transitions>
<entry>

```

```

        <key>analise</key>
        <value>3</value>
    </entry>
    <entry>
        <key>qualidade</key>
        <value>4</value>
    </entry>
    <entry>
        <key>parcial</key>
        <value>3</value>
    </entry>
    <entry>
        <key>processos</key>
        <value>1</value>
    </entry>
    <entry>
        <key>lider</key>
        <value>1</value>
    </entry>
    <entry>
        <key>projeto</key>
        <value>1</value>
    </entry>
    <entry>
        <key>analista</key>
        <value>1</value>
    </entry>
</setOfWords>
</transitions>
<transitions>
    <id>t9</id>
    <name>Registrar finalização de análise de qualidade
parcial</name>
    <description>Registrar a finalização da análise de qualidade
parcial de processos. </description>
    <processingTime>240</processingTime>
    <inputs>Acompanhamento de análise de qualidade
parcial</inputs>
    <outputs>Acompanhamento de análise de qualidade
parcial</outputs>
    <previousPlaces>p8</previousPlaces>
    <nextPlaces>p9</nextPlaces>
    <role>Analista de qualidade</role>
    <setOfWords>
        <entry>
            <key>registrar</key>
            <value>2</value>
        </entry>
    </setOfWords>
</transitions>

```

```

        <key>finalizacao</key>
        <value>2</value>
    </entry>
    <entry>
        <key>analise</key>
        <value>4</value>
    </entry>
    <entry>
        <key>qualidade</key>
        <value>5</value>
    </entry>
    <entry>
        <key>parcial</key>
        <value>4</value>
    </entry>
    <entry>
        <key>processos</key>
        <value>1</value>
    </entry>
    <entry>
        <key>acompanhamento</key>
        <value>2</value>
    </entry>
    <entry>
        <key>analista</key>
        <value>1</value>
    </entry>
    </setOfWords>
</transitions>
<transitions>
    <id>t10</id>
    <name>Disponibilizar relatório</name>
    <description>Disponibilizar o relatório de análise de
qualidade parcial no ambiente colaborativo de
iniciativas.</description>
    <processingTime>5</processingTime>
    <inputs>Relatório de análise de qualidade parcial</inputs>
    <outputs>Relatório de análise de qualidade parcial</outputs>
    <previousPlaces>p9</previousPlaces>
    <nextPlaces>p10</nextPlaces>
    <role>Analista de qualidade</role>
    <setOfWords>
        <entry>
            <key>disponibilizar</key>
            <value>2</value>
        </entry>
        <entry>
            <key>relatorio</key>
            <value>4</value>

```

```

    </entry>
    <entry>
      <key>analise</key>
      <value>3</value>
    </entry>
    <entry>
      <key>qualidade</key>
      <value>4</value>
    </entry>
    <entry>
      <key>parcial</key>
      <value>3</value>
    </entry>
    <entry>
      <key>ambiente</key>
      <value>1</value>
    </entry>
    <entry>
      <key>colaborativo</key>
      <value>1</value>
    </entry>
    <entry>
      <key>iniciativas</key>
      <value>1</value>
    </entry>
    <entry>
      <key>analista</key>
      <value>1</value>
    </entry>
  </setOfWords>
</transitions>
<places>
  <id>p1</id>
  <name></name>
  <previousTransitions></previousTransitions>
  <nextTransitions>t1</nextTransitions>
</places>
<places>
  <id>p2</id>
  <name></name>
  <previousTransitions>t1</previousTransitions>
  <nextTransitions>t2</nextTransitions>
</places>
<places>
  <id>p3</id>
  <name></name>
  <previousTransitions>t2</previousTransitions>
  <nextTransitions>t3</nextTransitions>
</places>

```

```

<places>
  <id>p4</id>
  <name></name>
  <previousTransitions>t3</previousTransitions>
  <nextTransitions>t4</nextTransitions>
</places>
<places>
  <id>p5</id>
  <name></name>
  <previousTransitions>t4</previousTransitions>
  <previousTransitions>t7</previousTransitions>
  <nextTransitions>t5</nextTransitions>
</places>
<places>
  <id>p6</id>
  <name></name>
  <previousTransitions>t5</previousTransitions>
  <nextTransitions>t6</nextTransitions>
  <nextTransitions>t8</nextTransitions>
</places>
<places>
  <id>p7</id>
  <name></name>
  <previousTransitions>t6</previousTransitions>
  <nextTransitions>t7</nextTransitions>
</places>
<places>
  <id>p8</id>
  <name></name>
  <previousTransitions>t8</previousTransitions>
  <nextTransitions>t9</nextTransitions>
</places>
<places>
  <id>p9</id>
  <name></name>
  <previousTransitions>t9</previousTransitions>
  <nextTransitions>t10</nextTransitions>
</places>
<places>
  <id>p10</id>
  <name></name>
  <previousTransitions>t10</previousTransitions>
  <nextTransitions></nextTransitions>
</places>
</businessProcess>

```

Modelo 3:

```
<?xml version="1.0" encoding="UTF-8"?>
<businessProcess>
  <transitions>
    <id>t1</id>
    <name>Atender o cliente</name>
    <description>Atender cliente para preenchimento da intenção de
compra de automóvel.</description>
    <processingTime>150</processingTime>
    <inputs>Intenção de compra</inputs>
    <outputs>Intenção de compra preenchida</outputs>
    <outputs>Dados financeiros cliente</outputs>
    <previousPlaces>p1</previousPlaces>
    <nextPlaces>p2</nextPlaces>
    <role>Vendedor</role>
    <setOfWords>
      <entry>
        <key>atender</key>
        <value>2</value>
      </entry>
      <entry>
        <key>cliente</key>
        <value>3</value>
      </entry>
      <entry>
        <key>preenchimento</key>
        <value>1</value>
      </entry>
      <entry>
        <key>intenção</key>
        <value>3</value>
      </entry>
      <entry>
        <key>compra</key>
        <value>3</value>
      </entry>
      <entry>
        <key>automóvel</key>
        <value>1</value>
      </entry>
      <entry>
        <key>preenchida</key>
        <value>1</value>
      </entry>
      <entry>
        <key>dados</key>

```

```

        <value>1</value>
    </entry>
    <entry>
        <key>financeiros</key>
        <value>1</value>
    </entry>
    <entry>
        <key>vendedor</key>
        <value>1</value>
    </entry>
</setOfWords>
</transitions>
<transitions>
    <id>t2</id>
    <name>Gerar pedido de compra</name>
    <description>Gerar pedido de compra com base no formulário de
intenção de compra.</description>
    <processingTime>40</processingTime>
    <inputs>Intenção de compra preenchida</inputs>
    <outputs>Pedido de compra</outputs>
    <previousPlaces>p2</previousPlaces>
    <nextPlaces>p3</nextPlaces>
    <role>Vendedor</role>
    <setOfWords>
        <entry>
            <key>gerar</key>
            <value>2</value>
        </entry>
        <entry>
            <key>pedido</key>
            <value>3</value>
        </entry>
        <entry>
            <key>compra</key>
            <value>5</value>
        </entry>
        <entry>
            <key>base</key>
            <value>1</value>
        </entry>
        <entry>
            <key>formulário</key>
            <value>1</value>
        </entry>
        <entry>
            <key>intenção</key>
            <value>2</value>
        </entry>
    </setOfWords>
</transitions>
</transitions>

```

```

        <key>vendedor</key>
        <value>1</value>
    </entry>
</setOfWords>
</transitions>
<transitions>
    <id>t3</id>
    <name>Receber pagamento</name>
    <description>Receber o pagamento do cliente.</description>
    <processingTime>30</processingTime>
    <inputs>Dados financeiros cliente</inputs>
    <outputs>Confirmação de pagamento</outputs>
    <previousPlaces>p3</previousPlaces>
    <nextPlaces>p4</nextPlaces>
    <role>Financeiro</role>
    <setOfWords>
        <entry>
            <key>receber</key>
            <value>2</value>
        </entry>
        <entry>
            <key>pagamento</key>
            <value>3</value>
        </entry>
        <entry>
            <key>cliente</key>
            <value>2</value>
        </entry>
        <entry>
            <key>dados</key>
            <value>1</value>
        </entry>
        <entry>
            <key>financeiros</key>
            <value>1</value>
        </entry>
        <entry>
            <key>confirmação</key>
            <value>1</value>
        </entry>
        <entry>
            <key>financeiro</key>
            <value>1</value>
        </entry>
    </setOfWords>
</transitions>
<transitions>
    <id>t4</id>
    <name>Entregar automóvel</name>

```

```

<description>Entregar o automóvel ao cliente, após recebimento
de confirmação de pagamento.</description>
<processingTime>15</processingTime>
<inputs>Pedido de compra</inputs>
<inputs>Confirmação de pagamento</inputs>
<outputs>Pedido de compra finalizado</outputs>
<previousPlaces>p4</previousPlaces>
<nextPlaces>p5</nextPlaces>
<role>Concessionária</role>
<setOfWords>
  <entry>
    <key>entregar</key>
    <value>2</value>
  </entry>
  <entry>
    <key>automóvel</key>
    <value>2</value>
  </entry>
  <entry>
    <key>cliente</key>
    <value>1</value>
  </entry>
  <entry>
    <key>recebimento</key>
    <value>1</value>
  </entry>
  <entry>
    <key>confirmação</key>
    <value>2</value>
  </entry>
  <entry>
    <key>pagamento</key>
    <value>1</value>
  </entry>
  <entry>
    <key>pedido</key>
    <value>2</value>
  </entry>
  <entry>
    <key>compra</key>
    <value>2</value>
  </entry>
  <entry>
    <key>finalizado</key>
    <value>1</value>
  </entry>
  <entry>
    <key>concessionária</key>
    <value>1</value>
  </entry>

```

```

        </entry>
    </setOfWords>
</transitions>
<places>
    <id>p1</id>
    <name></name>
    <previousTransitions></previousTransitions>
    <nextTransitions>t1</nextTransitions>
</places>
<places>
    <id>p2</id>
    <name></name>
    <previousTransitions>t1</previousTransitions>
    <nextTransitions>t2</nextTransitions>
</places>
<places>
    <id>p3</id>
    <name></name>
    <previousTransitions>t2</previousTransitions>
    <nextTransitions>t3</nextTransitions>
</places>
<places>
    <id>p4</id>
    <name></name>
    <previousTransitions>t3</previousTransitions>
    <nextTransitions>t4</nextTransitions>
</places>
<places>
    <id>p5</id>
    <name></name>
    <previousTransitions>t4</previousTransitions>
    <nextTransitions></nextTransitions>
</places>
</businessProcess>

```
