

PGCOMP - Programa de Pós-Graduação em Ciência da Computação
Universidade Federal da Bahia (UFBA)
Av. Adhemar de Barros, s/n - Ondina
Salvador, BA, Brasil, 40170-110

<http://pgcomp.dcc.ufba.br>
pgcomp@ufba.br

Um processo de software explicitamente especificado aprimora a qualidade do produto gerado, pois a especificação direciona o caminho e passos a serem seguidos para que o software desenvolvido alcance a qualidade almejada pelo projeto. O processo de software evolui juntamente às necessidades da organização e dos profissionais que o utilizam, e precisa ser continuamente monitorado e avaliado para manter sua qualidade. Para avaliar um processo de software utiliza-se, comumente de simulações ou dados obtidos após uma execução. Entretanto, em ambos os casos é difícil prever problemas no processo de software em um determinado projeto de desenvolvimento antes de uma ou mais execuções. Diante disso, Process Smells foram propostos para possibilitar que problemas no design de processos de software pudessem ser identificados antes destes serem executados. A presença de um Process Smell poderá ter um impacto negativo na qualidade do processo afetando assim a qualidade do produto de software. Process Smells foram especificados utilizando-se a notação Software Process Specification Metamodel (SPEM). Apesar do SPEM ser uma linguagem para o domínio da modelagem de processo Business Process Model and Notation (BPMN) tem sido utilizada largamente para processos de forma geral, com alta popularidade tanto no mercado e academia. Da mesma forma que o SPEM, embora o BPMN seja usado para melhorar o entendimento de um processo, a especificação de um processo pode ser feita de forma inadequada, ferindo os fatores de qualidade indicados. Neste contexto, esta pesquisa teve como objetivo especificar um catálogo de Process Smells para apoiar a identificação de anomalias em processos de software especificado com BPMN a partir de um catálogo já existente, proposto em (SANTOS; MACIEL; SANT'ANNA, 2018). Para isto, foi estabelecido uma metodologia para especificação e avaliação da proposta a partir de um estudo de entrevista com profissionais da Engenharia de Software. Inicialmente, os Process Smells foram especificados em BPMN, originando assim um novo catálogo de Process Smells, o Process Smells 2.0. A especificação foi avaliada através de um estudo de entrevista, que indicou que o novo catálogo proposto de Process Smells foi aceito pelos engenheiros de software. Espera-se que esta especificação do catálogo possa apoiar a identificação de Process Smells em processo de software modelados usando BPMN, de forma a indicar os pontos onde o processo pode ser melhorado, antes mesmo da sua primeira execução, evitando problemas que afetam negativamente os atributos de qualidade do processo. Adicionalmente, espera-se que este novo catálogo possa ampliar a possibilidade de uso dos Process Smells.

Palavras-chave: Modelo de Processo de Software, Bad Smells, Atributos de Qualidade, Process Smells, BPMN.

Process Smell 2.0: Um catálogo de Bad Smells para processo de software utilizando BPMN.

Renata de Sousa Santos

Dissertação de Mestrado

Universidade Federal da Bahia

Programa de Pós-Graduação em
Ciência da Computação

Maio | 2023



Universidade Federal da Bahia
Instituto de Computação

Programa de Pós-Graduação em Ciência da Computação

**PROCESS SMELL 2.0 : UM CATÁLOGO DE
BAD SMELLS PARA PROCESSO DE
SOFTWARE UTILIZANDO A NOTAÇÃO
BPMN**

Renata de Sousa Santos

DISSERTAÇÃO DE MESTRADO

Salvador
04 de Maio de 2023

RENATA DE SOUSA SANTOS

**PROCESS SMELL 2.0 : UM CATÁLOGO DE BAD SMELLS PARA
PROCESSO DE SOFTWARE UTILIZANDO A NOTAÇÃO BPMN**

Esta Dissertação de Mestrado foi apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia, como requisito parcial para obtenção do grau de Mestre em Ciência da Computação.

Orientadora: Rita Suzana Pitangueira Maciel

Salvador
04 de Maio de 2023

Ficha catalográfica elaborada pela Biblioteca Universitária de
Ciências e Tecnologias Prof. Omar Catunda, SIBI – UFBA.

S237 Santos, Renata de Sousa
Process Smell 2.0: Um Catálogo de Bad Smells para Processo
de Software Utilizando a Notação BPMN. / Renata de Sousa
Santos. – Salvador, 2023.
101 f.

Orientadora: Prof^ª. Dr^ª. Rita Suzana Pitangueira Maciel

Dissertação (Mestrado) – Universidade Federal da Bahia,
Instituto de Computação, 2023.

1. Computação. 2. Software. 3. Engenharia de Software. I.
Maciel, Rita Suzana Pitangueira. II. Universidade Federal da
Bahia. III. Título.

CDU 004.4

“Process Smell 2.0: Um catálogo de Bad Smells para processo de software utilizando BPMN.”

Renata de Sousa Santos

Dissertação apresentada ao Colegiado do Programa de Pós-Graduação em Ciência da Computação na Universidade Federal da Bahia, como requisito parcial para obtenção do Título de Mestre em Ciência da Computação.

Banca Examinadora



Prof^a. Dr^a. Rita Suzana Pitangueira Maciel (Orientadora
PGCOMP)



Documento assinado digitalmente
ANA PATRICIA FONTES MAGALHAES MASC
Data: 26/07/2023 11:58:27-0300
Verifique em <https://validar.iti.gov.br>

Prof^a. Dr^a. Ana Patrícia Fontes Magalhães Mascarenhas (UNEB)



Documento assinado digitalmente
CLAUDIO NOGUEIRA SANT ANNA
Data: 04/09/2023 21:15:54-0300
Verifique em <https://validar.iti.gov.br>

Prof. Dr. Cláudio Nogueira Sant'Anna (UFBA)

Agradeço a Deus pela graça de uma boa saúde mental

AGRADECIMENTOS

O caminho para a educação na maioria das vezes costuma ser árduo. Imensuravelmente ganhamos, mas também perdemos. As noites mal dormidas, preocupações com prazos, conciliar o trabalho e o estudo, as olheiras, a saudade de casa à quilômetros de distância e até mesmo a saúde mental são sem dúvidas um dos vários sentimentos que vivi durante este meu percurso. Em contrapartida passei a valorizar o tempo e agradecer pelas pessoas preciosas que encontrei durante esse tempo.

Em todos esses anos, me senti numa montanha russa de emoções, às vezes a vontade era realmente deixar tudo pela metade do caminho, mas foi aí que encontrei as pessoas preciosas ao qual mencionei anteriormente e que sou completamente grata, pois me fizeram entender que no final a recompensa chega, em forma de orgulho, de dever cumprido e por ter conseguido carregar toda a bagagem.

Começo agradecendo primeiramente a DEUS, pois foram nas nossas conversas mais íntimas que eu pude perceber que eu estava no caminho certo, não era o caminho fácil, mas era o caminho que eu precisava viver. Agradeço aos meus pais Renato e Edinalva pelo empenho e dedicação, a minha irmã Salete pelo incentivo diário. Vocês sempre serão a minha base e eu espero conseguir retribuir todo o esforço de vocês. João Victor, Deus com certeza tinha um plano pra gente e eu nunca me cansarei de te agradecer por todo apoio dado durante esta jornada. Obrigada por ter sido paciente e por não ter me deixado desistir por diversas vezes.

Durante todo esse processo e principalmente por ter vindo para Salvador sem nenhum familiar, gostaria de agradecer a Rosa Julina, você com certeza é uma das pessoas de coração mais puro que eu pude conhecer. Obrigada pela confiança e por ter sido um anjo na minha vida, lembrarei de você sempre com carinho.

Meus mais sinceros agradecimentos as minhas amigas e companheiras dessa jornada, Kécia Santos, Verusca Rocha, Mayka Lima e Larissa Barbosa. Obrigada meninas, por essa caminhada que fizemos juntas, serei sempre grata a cada uma de vocês.

Ed, Edison, um irmão que o mestrado me deu, acredito que não há palavras suficientes para que eu possa externar a minha gratidão a você. Obrigada por tudo, pela sua paciência e dedicação em repassar o seu conhecimento, pelas madrugadas em que você se dedicava em me ajudar e principalmente por ter me ajudado a chegar até aqui, esse trabalho é tão seu quanto meu. Obrigada mesmo!

Por último e não menos importante, agradeço a minha orientadora Rita Suzana por todo esse tempo de dedicação e paciência, muita paciência. Obrigada por ter me ouvido e ter me dado palavras de conforto em um dos momentos em que cheguei acreditar que não seria capaz. Agradeço também por todo suporte e direcionamento nesses anos todos, você com certeza tem uma grande parcela pra que eu pudesse concluir mais uma etapa da minha vida acadêmica.

A todos que contribuíram direto ou indiretamente com a minha formação, meu muito obrigada!

A humildade exprime uma das raras certezas de que estou certo: a de que ninguém é superior a ninguém.

—PAULO FREIRE

RESUMO

Um processo de software explicitamente especificado aprimora a qualidade do produto gerado, pois a especificação direciona o caminho e passos a serem seguidos para que o software desenvolvido alcance a qualidade almejada pelo projeto. O processo de software evolui juntamente às necessidades da organização e dos profissionais que o utilizam, e precisa ser continuamente monitorado e avaliado para manter sua qualidade. Para avaliar um processo de software utiliza-se, comumente de simulações ou dados obtidos após uma execução. Entretanto, em ambos os casos é difícil prever problemas no processo de software em um determinado projeto de desenvolvimento antes de uma ou mais execuções. Diante disso, *Process Smells* foram propostos para possibilitar que problemas no design de processos de software pudessem ser identificados antes destes serem executados. A presença de um *Process Smell* poderá ter um impacto negativo na qualidade do processo afetando assim a qualidade do produto de software. Inicialmente *Process Smells* foram especificados utilizando-se a notação *Software Process Specification Metamodel* (SPEM). Apesar do SPEM ser uma linguagem para o domínio da modelagem de processos, *Business Process Model and Notation* (BPMN) tem sido utilizada largamente para processos de forma geral, com alta popularidade tanto no mercado e academia, neste sentido a especificação de um novo catálogo pode permitir que process smells sejam compreendidos de forma independente de domínio de linguagem. Ainda no contexto de melhoria de processo, da mesma forma que o SPEM, embora o BPMN seja usado para melhorar o entendimento de um processo, a especificação de um processo pode ser feita de forma inadequada, ferindo os fatores necessários para a qualidade de um processo de software. Neste contexto, esta pesquisa teve como objetivo especificar um catálogo de *Process Smells* para apoiar a identificação de anomalias em processos de software especificado com BPMN a partir da proposta de (SANTOS; MACIEL; SANT'ANNA, 2018). Inicialmente, os *Process Smells* foram especificados em BPMN, originando assim um novo catálogo com 8 *Process Smells*, o *Process Smells 2.0*. A especificação foi avaliada por trinta e dois profissionais através de um estudo de entrevista, que indicou que o novo catálogo proposto de *Process Smells* foi aceito por estes profissionais. Através dos resultados obtidos neste estudo foi possível verificar que a proposta de identificar smells independente do domínio de linguagem faz sentido, bem como a segunda etapa do estudo se mostrou mais convergente em relação ao SPEM. Espera-se que a especificação do catálogo possa apoiar a identificação de *Process Smells* em processo de software modelados usando BPMN, de forma a indicar os pontos onde o processo pode ser melhorado, antes mesmo da sua primeira execução, evitando problemas que afetam negativamente os atributos de qualidade do processo. Adicionalmente, espera-se que o novo catálogo possa ampliar a possibilidade de uso dos *Process Smells*.

Palavras-chave: Modelo de Processo de Software, *Bad Smells*, Atributos de Qualidade, *Process Smells*, BPMN.

ABSTRACT

An explicitly specified software process improves the quality of the generated product. The specification directs the path and steps to be followed so that the developed software reaches the quality desired by the project. The software process evolves along with the needs of the organization and the professionals who use it. It needs to be continuously monitored and evaluated to maintain its quality. To evaluate a software process, simulations or data obtained after execution are commonly used. However, in both cases, it is difficult to predict problems in the software process in a given development project before one or more executions. Therefore, *Process Smells* were proposed to enable problems in the design of software processes to be identified before they were executed. The presence of a *Process Smell* can negatively impact the quality of the Process, thus affecting the quality of the software product. Initially, *Process Smells* were specified using the *Software Process Specification Metamodel* (SPEM) notation. Although SPEM is a language for the domain of process modeling, *Business Process Model and Notation* (BPMN) has been widely used for processes in general, with high popularity both in the market and academia. In this sense, the specification of a new catalog can allow Process smells to be understood independently of the language domain. Still, in the context of process improvement, just like SPEM, although BPMN is used to improve the understanding of a process, the specification of a process can be done inappropriately, injuring the necessary factors for the quality of a process of software. In this context, this research aimed to specify a *Process Smells* catalog to support the identification of anomalies in software processes specified with BPMN based on the (SANTOS; MACIEL; SANT'ANNA, 2018) proposal. Initially, *Process Smells* were specified in BPMN, thus originating a new catalog with 8 *Process Smells*, *Process Smells* 2.0. Thirty-two professionals evaluated the specification through an interview study, which indicated that these professionals accepted the proposed new *Process Smells* catalog. The results obtained in this study made it possible to verify that the proposal to identify smells regardless of the language domain makes sense. The second stage of the study proved to be more concurrent with SPEM. The catalog specification is expected to support the identification of *Process Smells* in software processes modeled using BPMN to indicate the points where the Process can be improved, even before its first execution, avoiding problems that negatively affect process quality attributes. Additionally, the new catalog is expected to expand the possibility of using *Process Smells*.

Keywords: *Software Process Model, Bad Smells, Quality Attributes, Process Smells, BPMN.*

SUMÁRIO

Capítulo 1—Introdução	1
1.1 Objetivos	3
1.2 Metodologia	3
1.3 Estrutura do Trabalho	5
Capítulo 2—Revisão Bibliográfica	7
2.1 Processo de Software e Modelos de Processos de Software	7
2.2 BPMN	8
2.2.1 <i>Flow Objects</i>	8
2.2.2 <i>Connecting Objects</i>	10
2.2.3 <i>Swimlanes</i>	11
2.2.4 <i>Artifacts</i>	12
2.3 <i>Bad Smells</i>	13
Capítulo 3—Trabalhos Relacionados	15
Capítulo 4—Process Smells	19
Capítulo 5—Catálogo de Process Smells 2.0 - Primeira Versão	23
5.1 Especificação dos Process Smells	34
5.1.1 Definição de similaridades entre os elementos estruturais do SPEM e BPMN	35
5.2 Avaliação da Primeira versão do Catálogo de <i>Process Smells 2.0</i>	36
5.3 Design do Estudo de Entrevista	37
5.4 Estrutura do Questionário	38
5.4.1 Piloto do Estudo de Entrevista	38
5.4.2 Execução do Estudo de Entrevista	39
5.5 Análise do Estudo de Entrevista	39
5.5.1 Caracterização do Perfil dos Participantes	40
5.5.2 Análise dos Dados Quantitativos e Qualitativos	41
5.6 Comparação dos resultados entre o BPMN e SPEM	42
5.6.1 Message Chains	44
5.6.2 Brain Activity	44
5.6.3 Data Activity	44

5.6.4	Divergent Change	45
5.6.5	Large Activity	45
5.6.6	Feature Envy	45
5.6.7	Shotgun Surgery	46
5.6.8	Work Product Clump	46
5.6.9	Brain Task	46
5.6.10	Long Input List	47
5.7	Ameaças à Validade	47
5.8	Discussão	47
Capítulo 6—Catálogo de Process Smells 2.0 - Versão Final		49
6.1	Refinamento do Catálogo	49
6.2	Catálogo de Process Smells 2.0	51
6.3	Resultado do Estudo de Entrevista 2	60
6.3.1	Perfil dos Participantes	60
6.3.2	Análise Quantitativa e Qualitativa dos Resultados	60
6.3.3	Análise qualitativa de cada process smell	62
6.3.3.1	LONG INPUT LIST	62
6.3.3.2	SHOTGUN SURGERY	63
6.3.3.3	LARGE PARTITION	64
6.3.3.4	BRAIN TASK	65
6.3.3.5	FEATURE ENVY	66
6.3.3.6	BRAIN PARTITION	66
6.3.3.7	MESSAGE CHAINS	67
6.4	Ameaças à Validade	68
6.5	Discussão	68
Capítulo 7—Considerações Finais		71
7.1	Trabalhos Futuros	72
Apêndice A—Estudo de Entrevista		77

LISTA DE FIGURAS

1.1	Metodologia - Sequência de passos realizados	4
2.1	Flow Objects	9
2.2	Connecting Objects	10
2.3	Swimlanes	11
2.4	Artifacts	12
2.5	Exemplo de um processo modelado em BPMN	13
3.1	Características dos trabalhos relacionados	17
4.1	<i>Representação em BPMN do Brain Partition</i>	20
4.2	<i>Representação em SPEM do Brain Activity</i>	21
5.1	<i>Message Chains</i>	24
5.2	<i>Brain Activity</i>	25
5.3	<i>Divergent Change</i>	26
5.4	<i>Large Activity</i>	27
5.5	<i>Feature Envy</i>	28
5.6	<i>Shotgun Sugery</i>	29
5.7	<i>Brain Task</i>	30
5.8	<i>Long Input List</i>	31
5.9	<i>Work Product Clumps</i>	32
5.10	<i>Data Activity</i>	33
5.11	Fase de elaboração do Catálogo de Process Smells 2.0	34
5.12	Etapas para a especificação do catálogo	34
5.13	Adaptação para a representação de uma atividade em BPMN	36
5.14	Passo a passo do estudo de entrevista	37
6.1	Etapas Metodológicas para o Refinamento do Catálogo	50
6.2	<i>Message Chains</i>	52
6.3	<i>Brain Partition</i>	53
6.4	<i>Divergent Change</i>	54
6.5	<i>Large Partition</i>	55
6.6	<i>Feature Envy</i>	56
6.7	<i>Shotgun Surgery</i>	57
6.8	<i>Brain Task</i>	58
6.9	<i>Long Input List</i>	59
6.10	Representação da Aceitação dos Process Smells	61

6.11	Representação dos Impactos negativos aos Sub-atributos de qualidade	61
6.12	Avaliação de aceitação dos impactos negativos quanto aos process smells	62
A.1	<i>Message Chains</i>	79
A.2	<i>Brain Activity</i>	80
A.3	<i>Data Activity</i>	82
A.4	<i>Divergent Change</i>	84
A.5	<i>Large Activity</i>	86
A.6	<i>Feature Envy</i>	88
A.7	<i>Shotgun Sugery</i>	90
A.8	<i>Work Product Clump</i>	92
A.9	<i>Brain Task</i>	94
A.10	<i>Long Input List</i>	96

LISTA DE TABELAS

5.1	Tempo de experiência e processos modelados.	40
5.2	Resultado de aceitação do <i>Message Chains</i>	41
5.3	<i>Process Smells</i> Resultado da Aceitação	42
5.4	Comparação de Resultados entre BPMN e SPEM	43
6.1	Aceitação geral entre os estudos	70

LISTA DE SIGLAS

BPMN	Business Process Model and Notation	1
OMG	Object Management Group	1
PML	Process Modeling Language	1
SPML	Software Process Modeling Language	1
SPEM	Software e Systems Process Engineering Meta-Model	2
BNF	Backus-Naur Form	7
XML	Extensible Markup Language	8
BPMI	Business Process Modeling Initiative	8
UML	Unified Modeling Language	16
RM	Ranking Médio	41

Capítulo

1

Este capítulo apresenta o cenário em que a presente investigação se insere, destacando os aspectos que motivaram a definição do tema. Os objetivos da pesquisa e a metodologia são também abordados ao longo deste capítulo e, ao final, discute-se sobre as contribuições esperadas.

INTRODUÇÃO

Um processo de software pode ser definido como um conjunto de atividades utilizadas por empresas de desenvolvimento de software para a construção de um produto de software. Estes processos incluem diferentes tipos de atividades desde as diretamente voltadas para a interação com o produto como especificação, codificação, validação ou atividades gerenciais e de acompanhamento da produção (SOMMERVILLE, 2011). O processo de software apresenta um conjunto de atividades parcialmente ordenado empreendidas para transformar os requisitos dos usuários em soluções de software. Para Genvigir (2003), um processo de software fornece suporte para a construção, mudança e melhoria do produto de software, assim como é modificado para comportar evolutivamente esta produção.

Em busca de obter especificações visuais de processo de software mais precisas, com elementos comuns e sintaxe definida, foram criadas as linguagens de modelagem de processo, (*Process Modeling Language (PML)*) e para processo de software, (*Software Process Modeling Language (SPML)*). Apesar de existirem SPML desenvolvidas com o propósito de modelar estes elementos de representação visual, o *Business Process Model and Notation (BPMN)* (OMG, 2010) foi criado originalmente para a modelagem de processos de negócios e, ainda que esse fosse o objetivo original desta notação, ela tem se mostrado eficiente para modelar até mesmo processos mais específicos, como os processos de software (CAMPOS; OLIVEIRA, 2013) e (DUMAS; PFAHL, 2016). Proposto pelo *Object Management Group (OMG)*, o BPMN fornece uma notação gráfica para especificar processos de negócios e pode ser usada diretamente pelas partes interessadas que projetam, gerenciam e realizam os processos de negócios, mas ao mesmo tempo é preciso o suficiente para permitir que os diagramas BPMN sejam traduzidos em componentes de processo de software. Em concordância com o que foi apresentado Campos e Lima (2012) reforça que o BPMN tem por objetivo fornecer assistência ao gerenciamento de processo de negócio, tanto para os usuários técnicos quanto para os usuários de negócio, fornecendo uma notação intuitiva, tornando os usuários capazes de representarem semânticas

de processos complexos, tal como um sistema ou software. No entanto, usar uma linguagem para modelar o processo de software por si só não garantirá a qualidade do processo de software.

Processos são diferentes para cada empresa de desenvolvimento de software, portanto, as especificações e a modelagem usarão a autonomia conforme necessário para determinar as atividades e tarefas do processo. Neste sentido, poderão ser encontradas brechas para modelagens difíceis de serem modificadas ou compreendidas, ferindo fatores de qualidade. Por exemplo, uma especificação com muitos fluxos de decisão pode tornar o processo complexo devido a quantidade de decisões a serem tomadas. Muitas atividades também tornam o processo longo e conseqüentemente mais difícil de ser entendido. Neste sentido, os dois cenários evidenciam o possível impacto negativo quanto a compreensibilidade. A adição de muitas dependências entre as atividades durante a especificação torna o processo difícil de ser modificado, impactando a modificabilidade do mesmo. Esses cenários podem ser percebidos como anomalias e para evitá-los os processos devem ser avaliados constantemente.

De acordo com Kroeger, Davidson e Cook (2014), a engenharia de processo possui quatro métodos de avaliação de qualidade mais difundidos, sendo: (i) avaliação do produto frente ao processo, (ii) avaliação por modelos de avaliação, (iii) avaliação através de especificações formais e a (iv) avaliação baseada na visão sistêmica da engenharia de software centrada no ser humano. A avaliação do código fonte é um outro cenário de avaliação de qualidade em engenharia de software, que consiste em grande parte detectar estruturas de código como métodos, classes, bibliotecas, dentre outras, que venham afetar a qualidade esperada. Fowler (1999), pensando na forma de caracterizar problemas de design de código estabeleceu algumas taxonomias denominadas como *bad smells* e que devem ser evitadas. Os *bad smells* podem afetar propriedades fundamentais da estrutura de código, bem como a redução da sua qualidade.

Para a realização deste trabalho, tomamos como base o estudo já feito por Santos, Maciel e Sant'Anna (2018), onde foram investigados se o conceito de *bad smells* poderia ser aplicado para processo de software, uma vez que para Osterweil (1987) o processo também pode ser considerado como software. As similaridades existentes entre o processo e o código fonte evidenciam-se nas funções desempenhadas pelos seus respectivos elementos. As atividades do processo e classes do código orientado a objeto são os elementos que devem cumprir um propósito, por exemplo, a atividade de planejar o cronograma de um projeto ou a classe que deve gerar um arquivo de impressão. Já as tarefas representam cada ação que deve ser feita para completar uma atividade e podem ser comparadas aos métodos para as classes (SILINGAS; MILEVICIENE, 2011).

Em Santos, Maciel e Sant'Anna (2018) foi especificado um Catálogo de *Process Smells* em *Software e Systems Process Engineering Meta-Model (SPEM)* com o objetivo de apoiar a identificação de anomalias em tempo de design do processo, buscando assim evitar verificações tardias ou em tempo de execução. Ainda em Santos, Maciel e Sant'Anna (2018), a adaptação de processo foi estabelecido com o termo *Process Smell*, que corresponde a identificação de anomalias no design no processo e que impactam negativamente seus atributos de qualidade. Em seguida, compreendeu-se que um *Process Smell* precisa ser composto por uma definição, impactos negativos, representação visual e estratégia de

detecção.

O estudo realizado por Ramos, Mello e Fonseca (2022), apresenta uma abordagem para identificar code smells para diferentes tipos de linguagens, como por exemplo, Java, Python e C++, o que reforça o conceito de smells ser comumente aplicado independente de linguagem. Outros estudos também evidenciam a identificação de smells em teste, bem como em code review, como é apresentado por Doğan e Tüzün (2022). Diante disso, Process Smells foi proposto utilizando a notação SPEM e mostrou-se viável na percepção dos avaliadores participantes do estudo em Santos, Maciel e Sant'Anna (2018). Contudo, o conceito de process smells ainda não foi utilizado fora do contexto do SPEM. Considerando a popularidade do BPMN e a sua possível eficiência em modelar processos de software, Campos (2013); especificar e avaliar um catálogo de process smells em BPMN tem o intuito de expandir a identificação de process smells independente da notação/linguagem utilizada.

Para este estudo foi definida a seguinte questão de pesquisa norteadora:

***Process Smells* podem ser identificados em modelos de processo de software independente da linguagem de domínio?**

1.1 OBJETIVOS

Este trabalho tem por objetivo geral especificar um catálogo de *Process Smells*, versão 2.0, para apoiar a identificação de anomalias em design de processo de software especificados em BPMN a partir do catálogo já existente especificado em SPEM. A dissertação possui como objetivos específicos:

1. Verificar a aplicabilidade dos *Process Smells* no contexto de processo de software especificados em BPMN;
2. Especificar os *Process Smells 2.0*;
3. Validar a aplicação do catálogo de *Process Smells 2.0* em modelos de processo de software;
4. Comparar os resultados da avaliação com os resultados obtidos por Santos, Maciel e Sant'Anna (2018).

1.2 METODOLOGIA

O presente trabalho tem por objetivo especificar uma nova versão do catálogo de *process smells* especificado em SPEM Santos, Maciel e Sant'Anna (2018) para apoiar a identificação de anomalias em designs de processo de software especificados em BPMN.

Um estudo exploratório é estabelecido quando conceitos iniciais ainda precisam ser identificados, assim como quando se quer descobrir novas possibilidades em um determinado campo de pesquisa (LAKATOS; MARCONI, 2002). Diante disso, pensando em explorar e conhecer a opinião de profissionais sobre a aplicabilidade e o quão aderente

é esta proposta, foi conduzido um estudo de entrevista que é um método de pesquisa flexível, pois seu maior foco é coletar as informações necessárias para a pesquisa do que seguir fielmente um questionário pré-definido, podendo assim alcançar profundidade na coleta de informações (JÚNIOR; JÚNIOR, 2012). Os passos para realização desta etapa do trabalho podem ser vistos conforme a Figura 1.1.

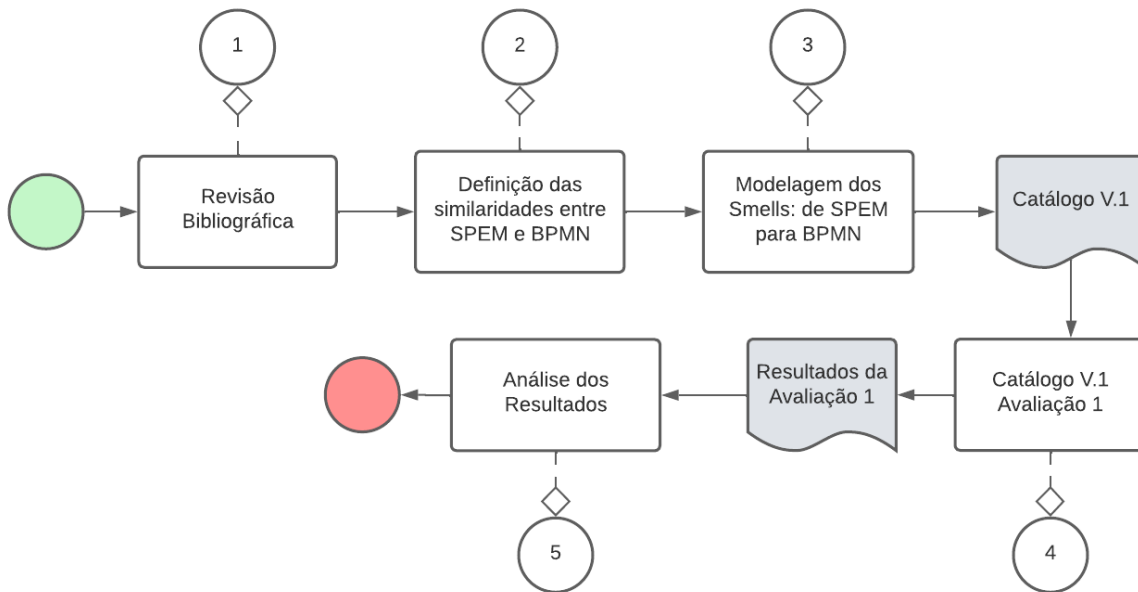


Figura 1.1 Metodologia - Sequência de passos realizados

A sequência de passos realizados teve como objetivo especificar e validar o catálogo de *process smells 2.0*. Os passos que constituem cada uma dessas atividades serão explicados a seguir:

1. Inicialmente foi realizada uma revisão bibliográfica na literatura a fim de melhor compreender os conceitos relacionados a processo de software, *bad smells* e BPMN.
2. Com o apoio das definições de cada elemento especificado pelo *Object Management Group* (OMG), iniciou-se neste passo a investigação entre as similaridades do SPEM e o BPMN 2.0. O intuito dessa etapa foi de verificar se os elementos do SPEM poderiam ser correspondentes a algum elemento no BPMN.
3. A partir dos *Process Smells* previamente selecionados no trabalho de Santos, Maciel e Sant'Anna (2018) e com o resultado obtido no passo dois, iniciou-se então a fase de modelagem dos smells para BPMN. Vale ressaltar que, os fatores de qualidade e os possíveis impactos aos atributos de qualidade (compreensibilidade e modificabilidade) foram mantidos, visto que, o objetivo do trabalho é de realizar a extensão do catálogo pensando na proposta de process smells independente de domínio, por isso as propriedades do trabalho realizado em SPEM se mantiveram.

Com a conclusão dos passos 2 e 3 obtemos a primeira versão do catálogo de *process smells 2.0* e em seguida através do estudo de entrevista com sete profissionais previamente selecionados foi realizada a primeira validação.

A proposta de validação deste catálogo consistiu em validar as características destes *process smells* compostos por suas definições e possíveis impactos que influenciariam na redução da qualidade do processo de software.

Por fim, no passo 5 foi realizado a análise dos resultados alcançados nesta pesquisa, onde os participantes consideraram relevante a proposta a cerca dos process smells, bem como concordaram com a sua definição os impactos causados ao atributos de qualidade.

1.3 ESTRUTURA DO TRABALHO

Os demais capítulos desta proposta de dissertação estão estruturados da seguinte forma: O Capítulo 2 apresenta os principais conceitos associados a este trabalho por meio do referencial teórico sobre Processo de Software, *Bad Smells* e Linguagem de Modelagem de Processo de Software. O Capítulo 3 discute os trabalhos existentes na literatura relacionados ao objeto de pesquisa. O Capítulo 4 apresenta o conceito do que são os process smells e suas representações. No Capítulo 5 é apresentado o Catálogo de *Process Smells* - primeira versão, desenvolvido na primeira fase desta pesquisa. O Capítulo 6 descreve a primeira avaliação da proposta e os resultados iniciais. O Capítulo 7 apresenta a versão final do catálogo de process smells e todas as etapas realizadas até a concepção final. Por fim, o Capítulo 8 apresenta as considerações finais relativas a este trabalho.

REVISÃO BIBLIOGRÁFICA

2.1 PROCESSO DE SOFTWARE E MODELOS DE PROCESSOS DE SOFTWARE

Um processo de software descreve os passos para o desenvolvimento de um produto de software. A elaboração de um processo busca generalizar a solução de um problema e cada realização individual do processo é considerada uma instância (OSTERWEIL, 1987). Neste sentido, as instâncias de um processo de software correspondem ao ciclo deste software, desde o momento em que o usuário final fornece as informações iniciais que orientam a construção, até a entrega final do software (OSTERWEIL, 1987). Segundo García-Borgoñon et al. (2014), a declaração dada por Osterweil que "Processos de software também devem ser considerados como software", foi um marco que apoiou a evolução de linguagens e modelos de processo de software, promovendo maior capacidade de compreensão e institucionalização deste tipo de processo.

A modelagem do processo deve tanto possibilitar a comunicação e entendimento do processo, bem como, apoiar o seu reuso, evolução e gerenciamento. Para isto, a modelagem do processo deve ser padronizada com propósito de fornecer capacidade de gerenciamento, revisão, suporte a ferramentas, contribuição para melhoria organizacional, base para medição e redução de esforços desnecessários (GENVIGIR, 2003). De acordo Genvigir (2003), a representação das atividades do mundo real de um processo de produção de software constitui um modelo de processo de software. Elementos de um modelo de processo devem ser bem definidos e bem relacionados e para cada tarefa devem ser estabelecidos os pré-requisitos, consequências e a sincronização com as demais tarefas.

Process Model Language (PML) é uma linguagem para descrever modelos de processo, independe de ferramenta e geralmente expressada pela gramática *Backus-Naur Form* (BNF). Assim uma PML possui declarações com palavras-chaves e

valores que pode descrever diagramas, atividades, entradas, saídas e ícones para representar esses elementos (GRUBEL, 1995). A PML surge da necessidade das indústrias em definirem seus processos de negócios, processos produtivos, industriais de manufatura e outros. A PML oferece suporte as colaborações empresariais, suporte a estruturas hierárquicas, interpretação flexível, robustez na definição do processo e suporte à integração, em destaque a outras linguagens já desenvolvidas com o mesmo propósito (ANDERL; RASSLER, 2008). *esta formatação está esquisita*

2.2 BPMN

De forma específica, pensando na modelagem do processo de negócio hoje destaca-se a BPMN, que embora não tenha sido desenvolvida para a modelagem de processo de software vem recebendo não só atenção do mercado, mas também da academia, em Wahl e Sindre (2006) percebe-se que praticamente todas as contribuições foram feitas em um nível analítico e conceitual.

Desenvolvida inicialmente pela *Business Process Modeling Initiative (BPMI)* a BPMN é um padrão para a modelagem de processos de negócio, padrão esse que é mantido nos dias atuais em conjunto pela BPMI e OMG. A BPMN tem o objetivo principal de prover uma notação compreensível pelos participantes do negócio, sendo os analistas de negócio que farão os rascunhos do processos, os desenvolvedores técnicos que são responsáveis por implementar a tecnologia, dentre outros envolvidos no gerenciamento do processo.

A BPMN 2.0 passou a ser mantida pela OMG e tem como linguagem abstrata MOF e um conjunto de padrões gráficos como linguagem concreta. Este conjunto de padrões gráficos que especificam os símbolos usados em diagramas e modelos de processos, permitindo modelar diferentes aspectos de fluxos de trabalho e fluxos de processos. A BPMN busca além de padronizar símbolos uniformizar a terminologia técnica de modelagem, possuindo a partir da versão 2.0, um formato padrão *Extensible Markup Language (XML)* que permite o intercâmbio do modelo do processo em diferentes ferramentas. A BPMN possui uma série de elementos gráficos e cada um deles representa uma determinada função. Esses elementos são organizados em quatro categorias e, os que serão usados neste trabalho serão apresentados nas subseções abaixo. Os demais elementos gráficos não serão apresentados pois não fizemos uso dos mesmos em nenhuma fase do trabalho. Neste cenário, consideraremos apenas os elementos a seguir.

2.2.1 Flow Objects

Os elementos responsáveis por definir o comportamento de um processo de negócio são os elementos de fluxo. Existem três elementos de fluxo principais que são: *Events* (Evento), *Activities* (Atividades) e *Gateways* (Passagens). Seus principais elementos podem ser vistos na Figura 2.1.




ELEMENTO	DESCRIÇÃO	NOTAÇÃO
<i>Event</i>	Um evento é algo que acontece durante a execução do processo. Ele afeta a execução do processo e, geralmente, possui uma causa (<i>trigger</i>) e um impacto (<i>result</i>). Eles podem ser de três tipos, baseado no momento em que afetam o processo: <i>Start</i> (Início), <i>Intermediate</i> (Intermediário) e <i>End</i> (Fim). Eventos do tipo início e intermediário também podem ser divididos quanto ao seu <i>trigger</i> (gatilho): <i>Message</i> , <i>Timer</i> , <i>Error</i> , <i>Cancel</i> , <i>Compensation</i> , <i>Rule</i> , <i>Link</i> , <i>Multiple</i> , <i>Terminate</i> .	
<i>Activity</i>	Atividade é um termo genérico para um trabalho desempenhado por uma organização. Uma atividade pode ser atômica ou não-atômica (composta). Os tipos de atividades que são parte de um modelo de processo são: <i>Process</i> (Processo), <i>Sub-Process</i> (Sub-Processo) e <i>Task</i> (Tarefa). <i>Task</i> e <i>Sub-Process</i> são representados por retângulos arredondados. <i>Process</i> não possui notação gráfica associada (é o modelo em si) ou é o conteúdo de um <i>Pool</i> (será visto mais adiante).	
<i>Gateway</i>	Um <i>Gateway</i> é usado para controlar a divergência ou convergência de fluxos de sequência. Dessa forma, ele irá determinar ramificação, bifurcação, ligação e junção de caminhos. Símbolos internos à notação gráfica irão indicar o tipo de controle.	

Figura 2.1 Flow Objects

2.2.2 Connecting Objects

Os objetos de fluxos são conectados pelos elementos de conexão. Há três tipos principais de conexão, que podem ser vistos na Figura 2.2.




ELEMENTO	DESCRIÇÃO	NOTAÇÃO
<i>Sequence Flow</i>	Um fluxo de sequência é usado para indicar a ordem em que atividades serão executadas em um processo. Os fluxos de sequência podem ser de quatro tipos: <i>Uncontrolled Flow</i> (Fluxo sem Controle), <i>Conditional Flow</i> (Fluxo condicional), <i>Default Flow</i> (Fluxo Padrão) e <i>Exception Flow</i> (Fluxo de Exceção).	
<i>Message Flow</i>	Um fluxo de mensagem é usado para mostrar a troca de mensagens entre dois participantes do processo. Em BPMN, dois <i>Pools</i> separados no diagrama representam dois participantes (uma entidade de negócio ou um papel do processo, por exemplo).	
<i>Association</i>	Usada para associar informação a objetos de fluxo. Objetos gráficos ou textuais, que não são de fluxos podem ser associados a objetos de fluxo.	

Figura 2.2 Connecting Objects

2.2.3 Swimlanes

São elementos que possibilitam o agrupamento de elementos primários, como atividades eventos e associações, em “raias”, o que pode indicar o conjunto de atividades de uma entidade do negócio, por exemplo. Esses elementos se apresentam na Figura 2.3 (BPMN, 2004).


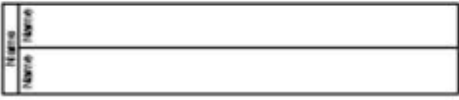
ELEMENTO	DESCRIÇÃO	NOTAÇÃO
<i>Pool</i>	Uma “piscina” representa um participante do processo. Ele representa um recipiente que separa um conjunto de atividades de outro <i>Pool</i> , geralmente no contexto de B2B (<i>Business To Business</i>).	
<i>Lane</i>	Uma “raia” é uma sub partição de um <i>Pool</i> . <i>Lanes</i> são usados para organizar e categorizar as atividades.	

Figura 2.3 Swimlanes

2.2.4 Artifacts

Os artefatos são usados para prover informações adicionais ao Processo. Existem três artefatos padrões, no entanto, ferramentas e modeladores podem adicionar quantos tipos de artefatos julgarem necessário. A Figura 2.4 descreve os principais tipos de artefatos usados na modelagem em BPMN (BPMN,2004).




ELEMENTO	DESCRIÇÃO	NOTAÇÃO
<i>Data Object</i>	Objetos de dados são considerados artefatos porque eles não têm efeito direto no fluxo de sequência ou fluxo de mensagens do processo, mas eles provêm informação acerca do que as atividades precisam para serem executadas e o que elas produzem.	
<i>Group</i> (uma caixa que engloba um grupo de objetos para propósitos de documentação)	Um grupo de atividades que não afeta o fluxo de sequência. O agrupamento pode ser realizado para fins de análise ou documentação. Grupos também podem ser usados para identificar atividades de uma transação distribuída que é mostrada entre <i>Pools</i> .	
<i>Text Annotation</i> (ligada a uma Associação)	Anotações de texto são mecanismos utilizados pelos modeladores para prover informação adicional aos leitores do diagrama de processo de negócio.	

Figura 2.4 Artifacts

A Figura 2.5 representa um processo especificado em BPMN, onde é possível notar o uso dos elementos de representação visual citados anteriormente, como: evento de início, atividade, subprocesso, *gateway*, evento de fim e as subpartições de uma *pool*.

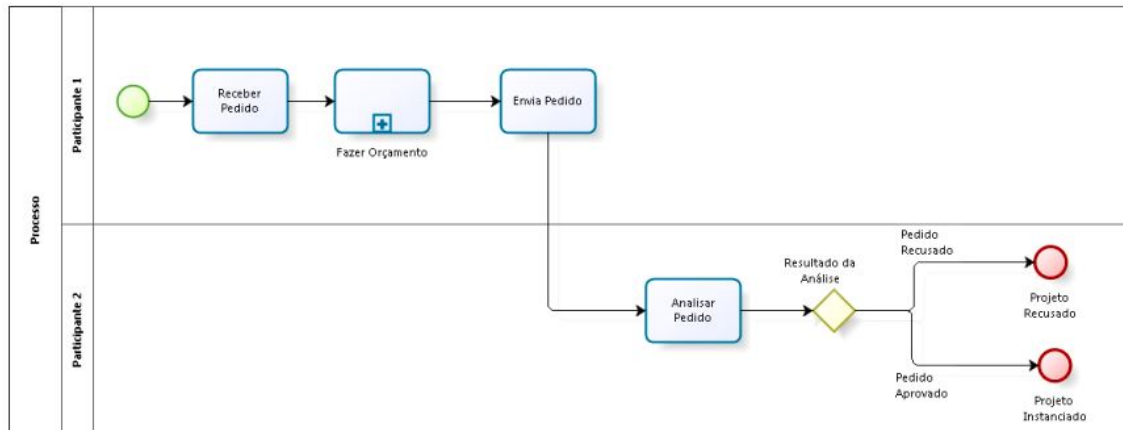


Figura 2.5 Exemplo de um processo modelado em BPMN

2.3 BAD SMELLS

Bad smells são propostas de design que podem reduzir a qualidade do código e indicar alguns problemas (FOWLER, 1999). Fowler (1999) definiu que *bad smells* são estruturas de código candidatas a refatoração. Ao todo, ele reuniu vinte e dois *bad smells* de código com suas características básicas, os contextos onde eles podem ser encontrados, as implicações negativas que o *bad smells* causa no design do software e a indicação das refatorações adequadas para reduzir o problema. As indicações para a identificação de *bad smells* são descritas a partir de algumas características que ferem alguns dos conceitos de programação. Essas indicações de identificação dos *bad smells* tiveram como validação a utilização da experiência profissional e verificações de acerto, com propósito de indicar para experiência futura uma melhor capacidade de decisão sobre o design do código.

Propostas de taxonomia surgiram após o estabelecimento dos *bad smells*, visando fornecer a compreensão sobre os impactos causados nos design de software, constituindo as seguintes categorias: *The Bloaters*, *The Object-Oriented Abusers*, *The Change Preventers*, *The Dispensables*, *The Couplers*, *Encapsulators* (MARTICORENA; LÓPEZ; CRESPO, 2006), (MANTYLA; VANHANEN; LASSENIUS, 2003). Lanza e Marinescu (2007), apresentam alguns *bad smells* e suas estratégias de detecção (LANZA; MARINESCU, 2007). Entre os exemplos, o *Brain Method* é um *bad smell* que define métodos longos, de difícil compreensão e reuso. Este *bad smell* se caracteriza por algumas heurísticas como o fato de que o *Brain Method* tende a ser um método longo e executar mais de uma funcionalidade, reduzindo assim a compreensibilidade e testabilidade do código, utilizar muitas ramificações

(se e senão) representando um design divergente ao orientado a objeto e fazer uso de muitas variáveis locais e globais.

Na literatura é abordado sobre a estratégia de detecção de *bad smells*, essas estratégias estabelecem mecanismos para a identificação de problemas no *design* de código orientado a objeto. Heurísticas são definidas conforme a estratégia e são quantificadas de forma a identificar desvios específicos no *design* do código. A partir do mecanismo de detecção são analisados se os elementos de design estão em conformidade com as regras pré estabelecidas e assim são apontados os problemas.

A refatoração de bad smells é um processo que consiste em identificar e corrigir problemas no código fonte por meio de técnicas de refatoração. Essas técnicas são aplicadas para melhorar a estrutura do código e torná-lo mais legível, manutenível e escalável. O conceito de refatoração de bad smells foi introduzido por Martin Fowler em seu livro "Refactoring: Improving the Design of Existing Code", publicado em 1999. Neste livro, Fowler descreve como a refatoração pode ajudar a melhorar a qualidade do código, eliminando problemas como código duplicado, funções muito longas, dependências desnecessárias e outras formas de bad smells. Ele apresenta diversas técnicas de refatoração que podem ser utilizadas para corrigir bad smells, tornando o código mais legível, manutenível e escalável. A refatoração de bad smells é uma prática essencial para garantir a qualidade do código e manter o software facilmente mantido e evoluído ao longo do tempo. (FOWLER, 1999)

Este capítulo tem como objetivo apresentar os trabalhos relacionados a essa pesquisa e suas principais características

TRABALHOS RELACIONADOS

O trabalho desenvolvido por Santos, Maciel e Sant'Anna (2018) abre caminho para a extensão de um novo catálogo, agora modelado em BPMN, pois a adaptação de *bad smells* para processo de software proposto pelos autores é um dos pontos de partida para o trabalho corrente. A pesquisa desenvolvida nesta dissertação propõe um catálogo para apoiar a identificação de anomalias em processo de software, os *process smells*, utilizando como referência os estudos realizados por Fowler (1999) e Marinescu (2004), bem como o trabalho desenvolvido por Santos, Maciel e Sant'Anna (2018).

Propostas de adaptações e detecção de *bad smells* para diferentes domínios tem sido realizadas conforme Macía, Sant'Anna e Staa (2008), Deursen et al. (2001), Greiler, Deursen e Storey (2013), Corradini et al. (2018), García et al. (2003) e Khlif et al. (2009), porém nenhuma delas tem como objetivo processos de software. Em comum, estas propostas têm estratégias de adaptação e detecção similares quanto a forma que estão realizadas. De forma similar, as estratégias adotadas dos demais trabalhos sobre detecção e adaptação de bad smells para diferentes domínios foram adotadas na pesquisa dessa dissertação, visto que a pesquisa busca adaptar smells de SPEM para BPMN, buscando verificar se os smells podem ser identificados independente desta linguagem de domínio.

As estratégias de adaptação e detecção das propostas citadas seguem alguns passos em comum. Inicialmente, estabelecida a forma com que ocorrerá a adaptação dos *bad smells* conforme o contexto aplicado. No geral essa adaptação se utiliza da especificação dos *bad smells* em comparação a especificação ao do domínio ao qual será adaptado. Em seguida se define as estratégias de detecção, métricas e valores limiares. Adicionalmente, faz-se necessário definir a validação desta adaptação dos smells para um novo contexto.

Macía et. al. (2008) propôs um estudo para a verificação dos *bad smells* *God Class* e *Data Class* em diagramas de classes definidos com *Unified Modeling Language (UML)*. O trabalho de Deursen et. al. (2001) propõe *bad smells* para o contexto de testes unitários definidos como *test code smells*.

Entre os trabalhos relacionados dois abordam *bad smells* em *Business Process Modeling Notation (BPMN)* (SILINGAS; MILEVICIENE, 2011) (CORRADINI et al., 2018). O primeiro adapta e o segundo detecta *bad smells* em BPMN. Em Silingas (2007) foi apresentado um conjunto de *bad smells* para notações de modelos de processo de negócio BPMN. No trabalho de Corradini et al. (2018), é proposto um *framework* para compreensibilidade de modelos BPMN.

Os trabalhos de Macía et. al, Deursen et. al. (2001) e Rompaey et. al. (2007) evidenciam que *bad smells* vem sendo atribuídos a diferentes contextos. O primeiro no contexto de diagramas de classes e o segundo em testes unitários. Diferente destes dois trabalhos citados, os trabalhos a seguir possuem maior aproximação com o tema da dissertação uma vez que, estes elaboraram estratégias de adaptação de *bad smells* para BPMN.

Em Silingas (2007), apesar da referência aos *bad smells* de Fowler, os *bad smells* apresentados para o BPMN são bem distintos, com exceção do *Large Process Diagrams* que apresenta a preocupação em relação ao tamanho do diagrama do processo e o impacto negativo a compreensibilidade do processo. Quanto aos demais *bad smells* para BPMN eles retrataram a nomeação inadequada dos elementos do modelo de processo de negócio, usos inadequados de elementos do BPMN como fluxos de decisão, eventos e *loops* além de outras falhas de *layout* do modelo de processo de negócio.

No *framework* apresentado por Corradini (2018) são incorporadas as referências dos *bad smells* de Silingas (2007), assim como referências de outros autores e ainda é apresentada uma ferramenta que realiza a detecção automática das diretrizes do *framework*. As estratégias de detecção se estabelecem pela verificação feita, por algoritmos, para cada diretriz do *framework*. Estes algoritmos fazem a leitura do arquivo convertido em *eXtensible Markup Language (XML)* a partir de arquivos com extensão BPMN e se utilizam dos dados do XML para realizar medições que alcancem os elementos que divergem das suas diretrizes. Contudo, o trabalho não apresenta a composição das estratégias de detecção de forma detalhada apenas referencia cada diretriz.

Os trabalhos relacionados anteriormente retratam o uso de métricas para detecção dos *bad smells*. Vanderfeesten et al. (2007) e Khlif et. al. (2009) utilizam como base para realizar a adaptação das métricas uma tabela comparativa entre os elementos de código de software e do BPMN. A adaptação busca poder medir a qualidade do *design* relacionada a cinco princípios. Ambos verificam acoplamento, coesão, complexidade, enquanto apenas o primeiro se estende verificando também modularidade e tamanho. Vanderfeesten et al. (2007) estabelece a comparação entre elementos de código e elementos do BPMN.

Diante aos trabalhos relacionados apresentados dois trabalhos abordam *bad smells* em BPMN, porém a pesquisa desta dissertação se diferencia pelos seguintes aspectos. O primeiro, Silingas (2007) adapta *bad smells* a partir de características que contrariam as boas práticas de modelagem com BPMN. Os *process smells* tem como base os *bad smells* de orientação a objeto e verificam os impactos negativos a atributos de qualidade definidos e evidenciados pelos usuários de processo conforme kroeger (2011). O segundo, Corradini(2018) detecta problemas no *design* de modelos BPMN seguindo referências de um conjunto de guias de qualidade de modelagem e métricas de verificação dos problemas. Este trabalho apesar de utilizar BPMN, assume algumas verificações de problemas como o uso não estruturado do BPMN o que difere do catálogo de *process smells 2.0* que só avalia a qualidade estrutural da modelagem do processo de software.

As contribuições dos trabalhos relacionados que colaboraram para a especificação do objeto proposto nesta pesquisa estão dispostos na tabela representada pela Figura 3.1 Nesta tabela são apresentadas as características e os estudos encontrados na literatura bem como a relação destes com a proposta deste trabalho.

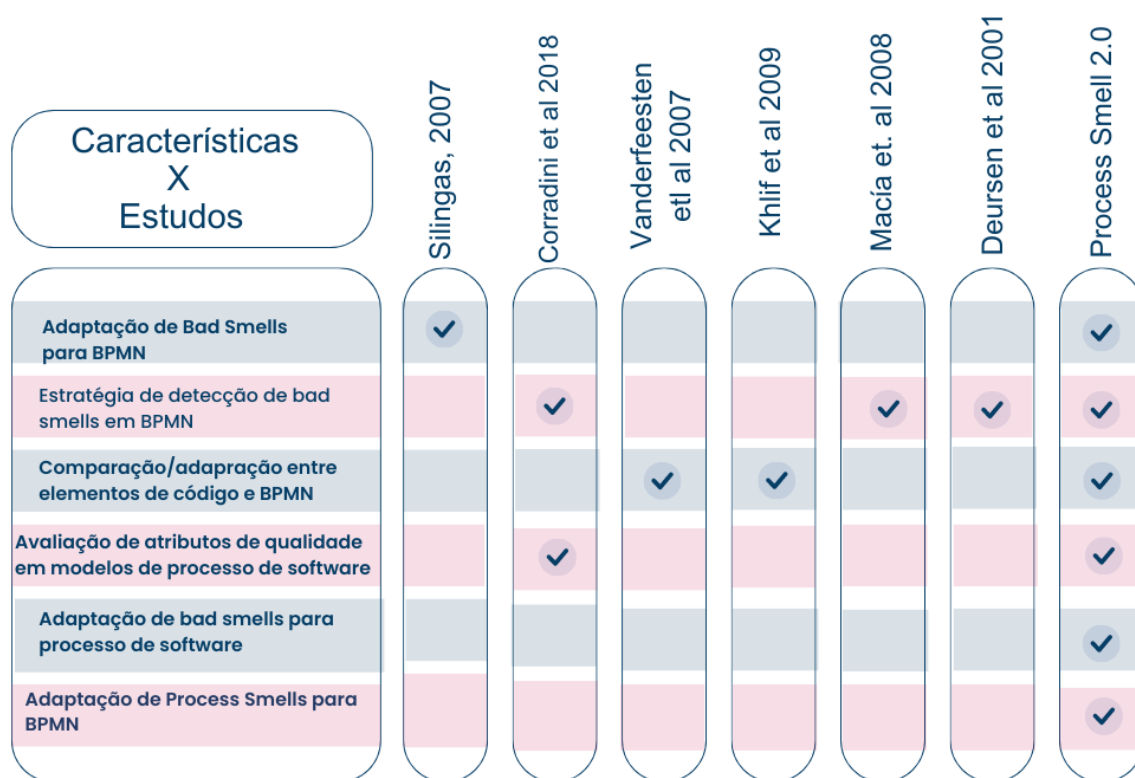


Figura 3.1 Características dos trabalhos relacionados

PROCESS SMELLS

Os *process smells* surgem como consequência das decisões efetuadas durante fases como concepção, modelagem e produção do modelo de processo. Esses *smells* afetam a disposição dos elementos do processo, levando a uma diminuição na qualidade dos fatores esperados para o processo. Neste capítulo, são fornecidas as definições e composições de cada *process smell*, tanto na notação BPMN quanto na notação SPEM. Cada um é composto pelo elemento impactado, uma descrição que o caracteriza e os possíveis efeitos resultantes. Adicionalmente, é apresentada uma ilustração que demonstra como um *smell* pode ser representado. Os exemplos a seguir apresentam o *Brain Partition*, representado em BPMN, e o *Brain Activity*, representado em SPEM.

Process Smell: *Brain Partition*

Elemento Impactado: Partição

Descrição: Brain Partition é um papel que centraliza a maior parte das ações relevantes de um Processo. Possui muitas tarefas e muitos fluxos de decisão para estruturar as tarefas.

Possíveis Impactos: Papéis centralizadores que concentram muitas tarefas e fluxos de decisão se tornam grandes e complexos e podem reduzir a compreensibilidade do usuário do processo.

Representação:

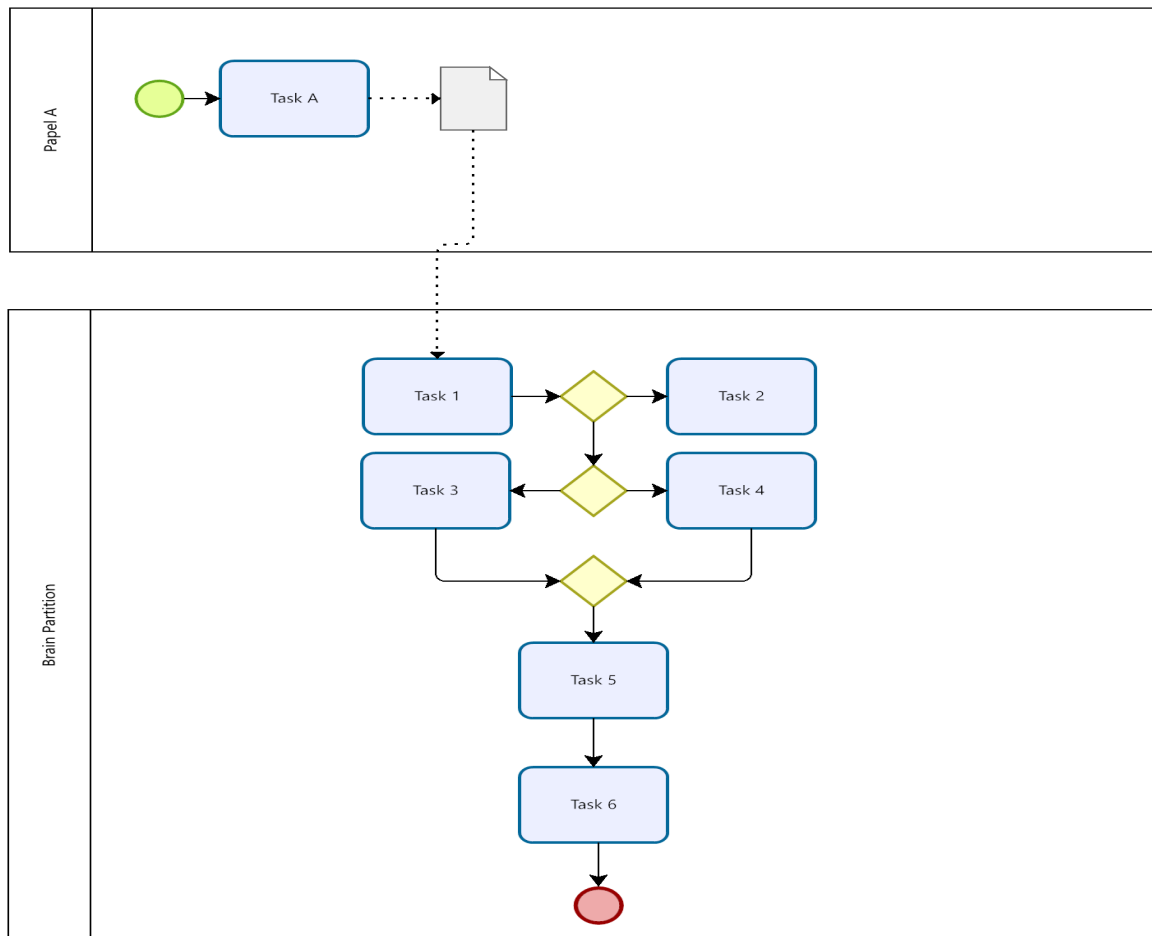


Figura 4.1 Representação em BPMN do Brain Partition

De acordo com Santos, Maciel e Sant'Anna (2018), a representação de uma *Brain Activity* no SPEM segue o seguinte modelo:

Process Smell: *Brain Activity*

Elemento Impactado: Atividade

Descrição: Uma Brain Activity é uma atividade que centraliza as ações de um processo. Essa atividade possui a maior parte das ações relevantes para o processo, possui muitas tarefas e muitos fluxos de decisão para estruturar as tarefas.

Possíveis Impactos: Atividades centralizadoras, que concentram muitas tarefas e fluxos de decisão se tornam atividades grandes e complexas, podem reduzir a compreensibilidade do usuário do processo.

Representação:

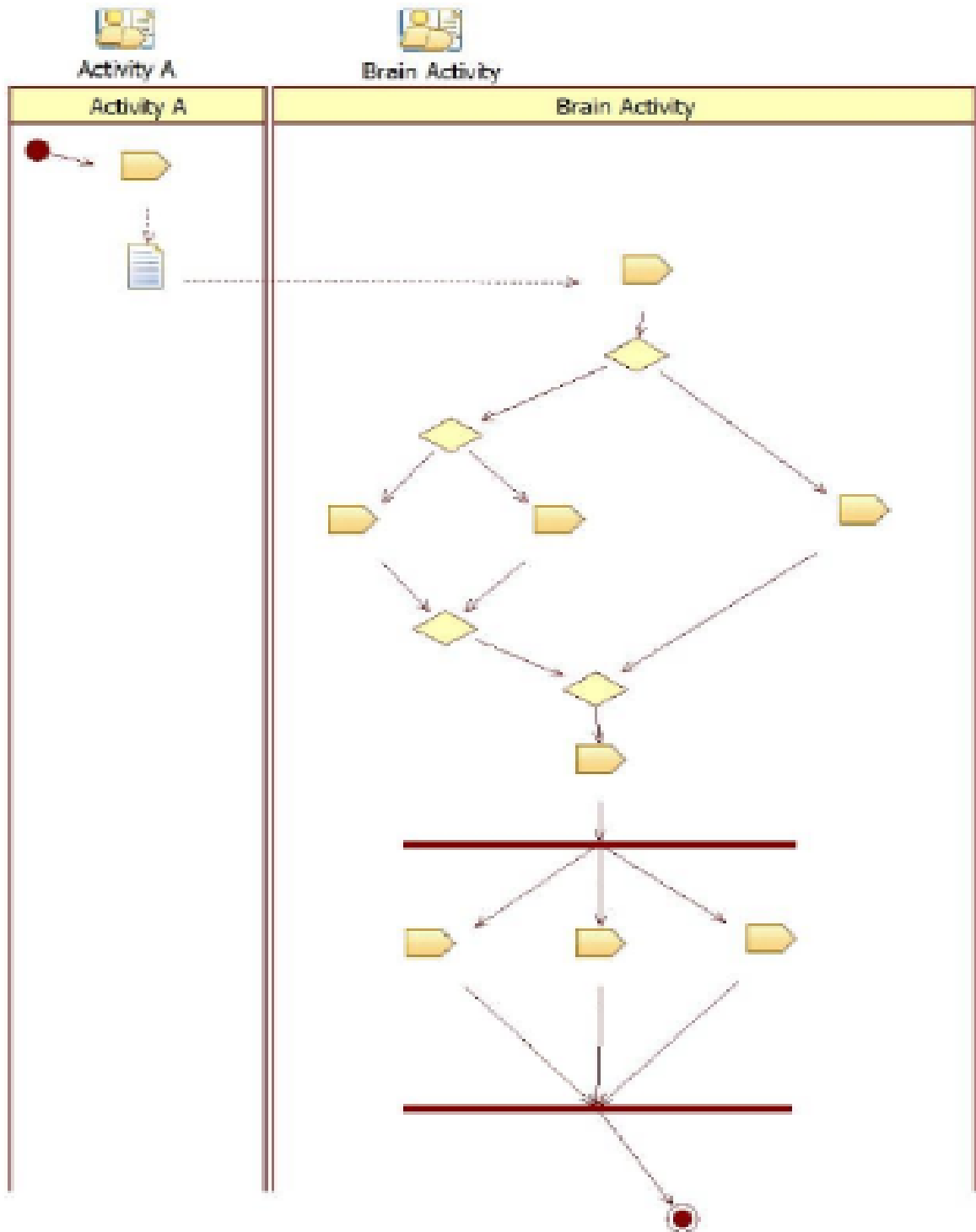


Figura 4.2 *Representação em SPEM do Brain Activity*

O catálogo apresentado por Santos, Maciel e Sant’Anna (2018) é composto por

dez process smells, sendo eles: Brain Activity, Brain Task, Message Chains, Large Activity, Shotgun Surgery, Divergent Change, Feature Envy, Long Input List, Data Activity e Work Product Clumps.

No capítulo 5, descrições, possíveis impactos e elemento impactado dos smells citados anteriormente serão apresentados uma vez que, para a primeira versão do catálogo em BPMN utilizamos os mesmos conceitos da proposta de (SANTOS; MACIEL; SANT'ANNA, 2018). No capítulo 6, se encontram todos os smells em BPMN contendo suas descrições, possíveis impactos e elemento impactado.

CATÁLOGO DE PROCESS SMELLS 2.0 - PRIMEIRA VERSÃO

A primeira versão deste catálogo conta com dez *process smells* caracterizados nesta pesquisa, contendo suas definições, elemento estrutural do BPMN impactado, representação visual e possíveis impactos que podem afetar negativamente os atributos de qualidade, no qual serão apresentados a seguir.

Process Smell: *Message Chains***Elemento impactado:** Tarefa

Descrição: O *Message Chains* define longas cadeias de trocas de produtos de trabalho entre tarefas que podem representar dependências disfarçadas. Uma tarefa precisa de um produto de trabalho de entrada que para ser gerado precisa esperar por uma cadeia de outras tarefas serem executadas até que este produto de trabalho de entrada esteja pronto.

Possíveis Impactos: Quanto mais longa for a cadeia de tarefas, maior será a dificuldade na legibilidade do processo, uma vez que aumenta a distância entre a tarefa solicitante e a tarefa que entregará os produtos de trabalho de entrada solicitados, afetando a compreensibilidade do processo.

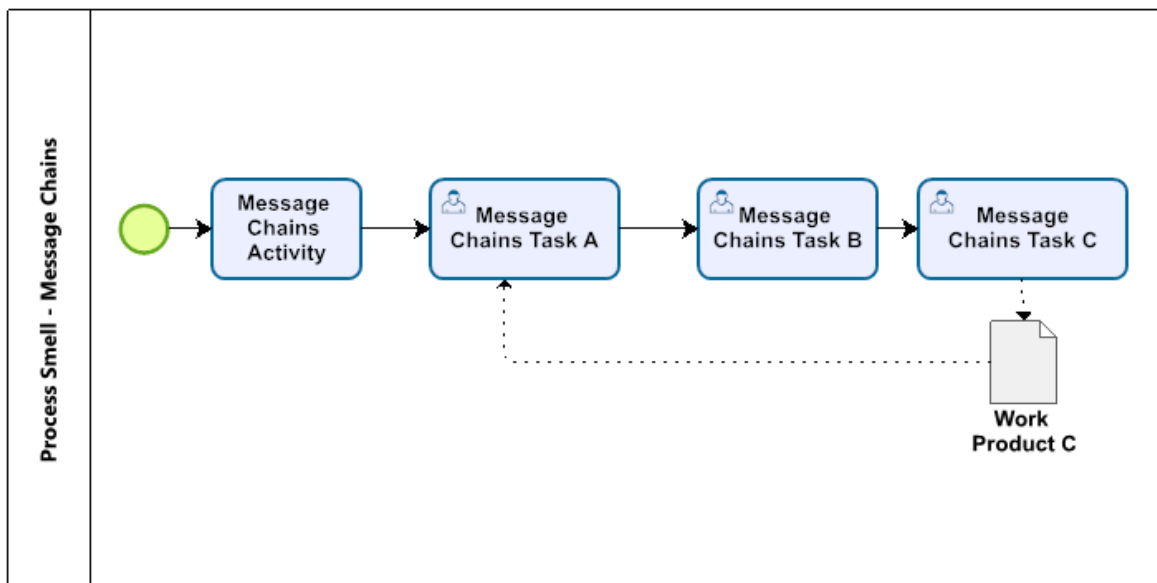
Representação:

Figura 5.1 *Message Chains*

Process Smell: *Brain Activity*

Elemento Impactado: Atividade

Descrição: Brain Activity é uma atividade que centraliza a maior parte das ações relevantes de um Processo. Possui muitas tarefas e muitos fluxos de decisão para estruturar as tarefas.

Possíveis Impactos: Atividades centralizadoras, que concentram muitas tarefas e fluxos de decisão tornam-se atividades grandes e complexas e podem reduzir a compreensão do usuário sobre o processo.

Representação:

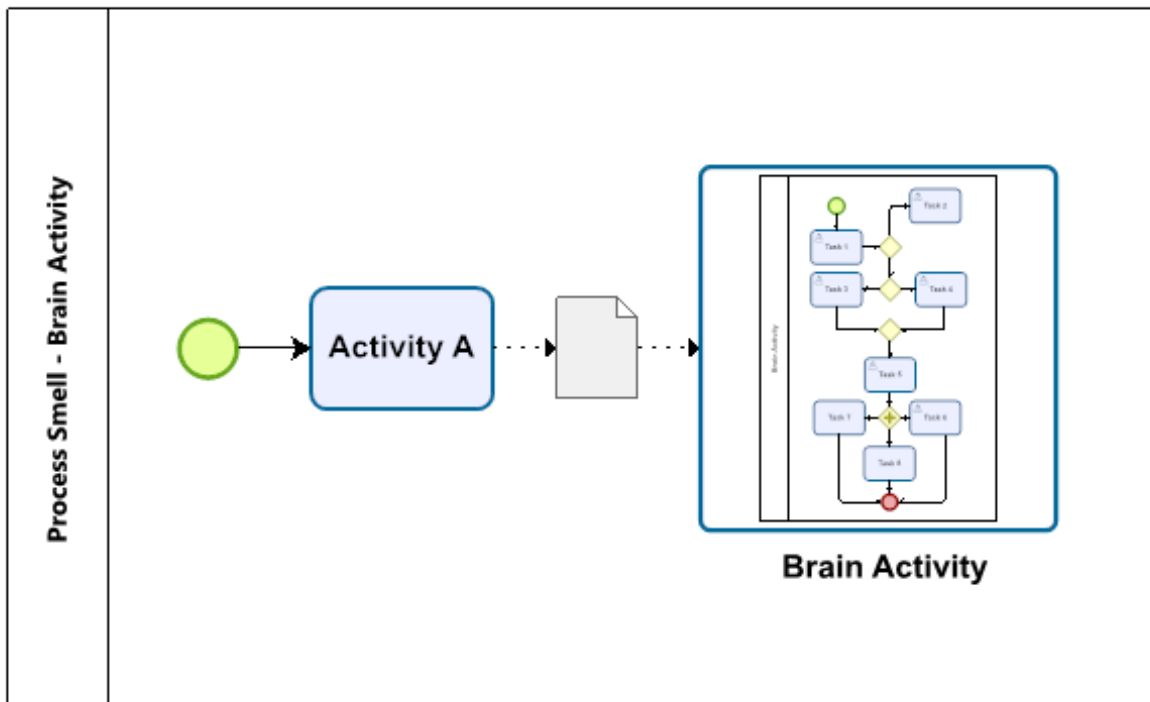


Figura 5.2 *Brain Activity*

Process Smell: *Divergent Change***Elemento Impactado:** Atividade

Descrição: Uma atividade que tem como propósito atender alguma "necessidade" do processo. Um exemplo de "necessidade" em um processo de software pode ser: construir o Termo de Abertura do projeto; validar um documento de especificação; entre outros. Assim, o Divergent Change ocorre quando uma atividade é configurada para atender a não somente uma "necessidade" do processo, tendo assim, diferentes conjuntos de tarefas para cada necessidade a qual atenderá. Em consequência, sempre que qualquer dessas "necessidades", que essa atividade atende sofrer mudanças, a parte das tarefas responsáveis por atendê-la também mudará, enquanto as outras tarefas que não fazem parte desse conjunto não sofrerão nenhuma mudança por este mesmo motivo.

Possíveis Impactos: Essa atividade se encarrega de diferentes propósitos, desta forma se torna pouco coesa ou seja, as tarefas desta atividade realizam ações de forma independente das outras tarefas, o que pode prejudicar a compressibilidade da mesma.

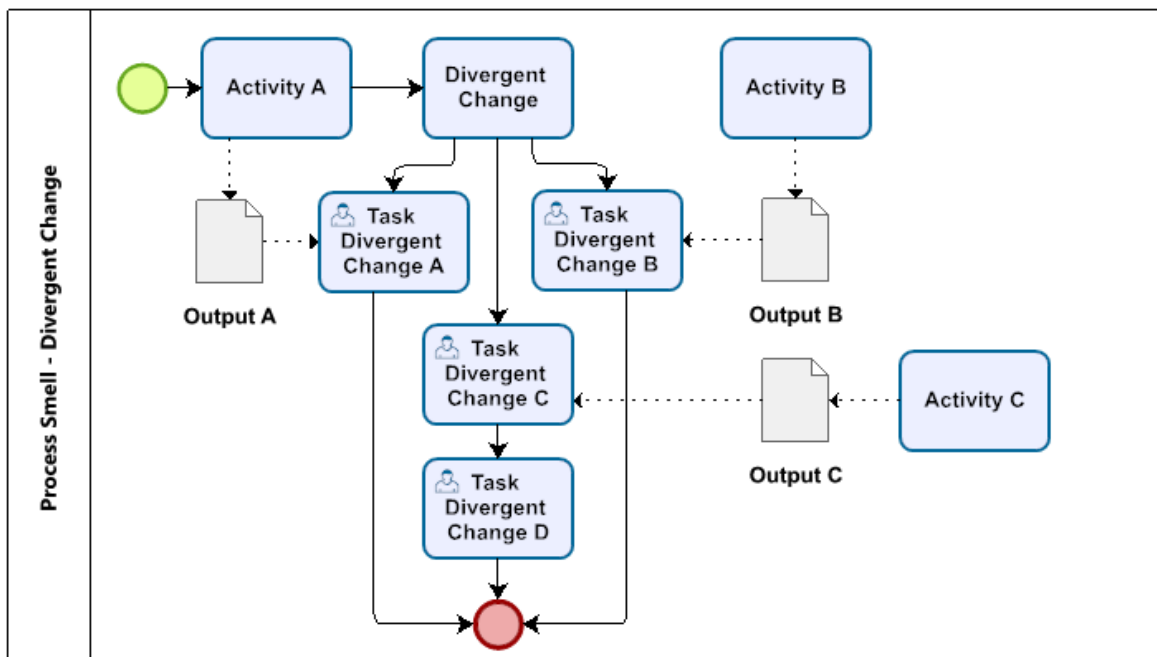
Representação:

Figura 5.3 *Divergent Change*

Process Smell: *Large Activity***Elemento Impactado:** Atividade

Descrição: Uma *Large Activity* se caracteriza quando uma atividade é grande, por conter muitas tarefas e possuir muitas responsabilidades. Ela centraliza as principais ações do Processo, deixando outras atividades com responsabilidades mais triviais e utilizando produtos de trabalho destas para concluir seus objetivos.

Possíveis Impactos: Apesar da sensação de simplificar o processo, a *Large Activity* possui muitas tarefas e muitos fluxos de decisão, assim aumenta a complexidade. A atribuição de muitas tarefas a uma única atividade gera o risco de confundir a responsabilidade da mesma, adicionando tarefas que cuidam de diferentes necessidades do processo. Este fato pode também impactar negativamente a coesão entre as tarefas. Essas duas características podem reduzir a compreensibilidade do processo.

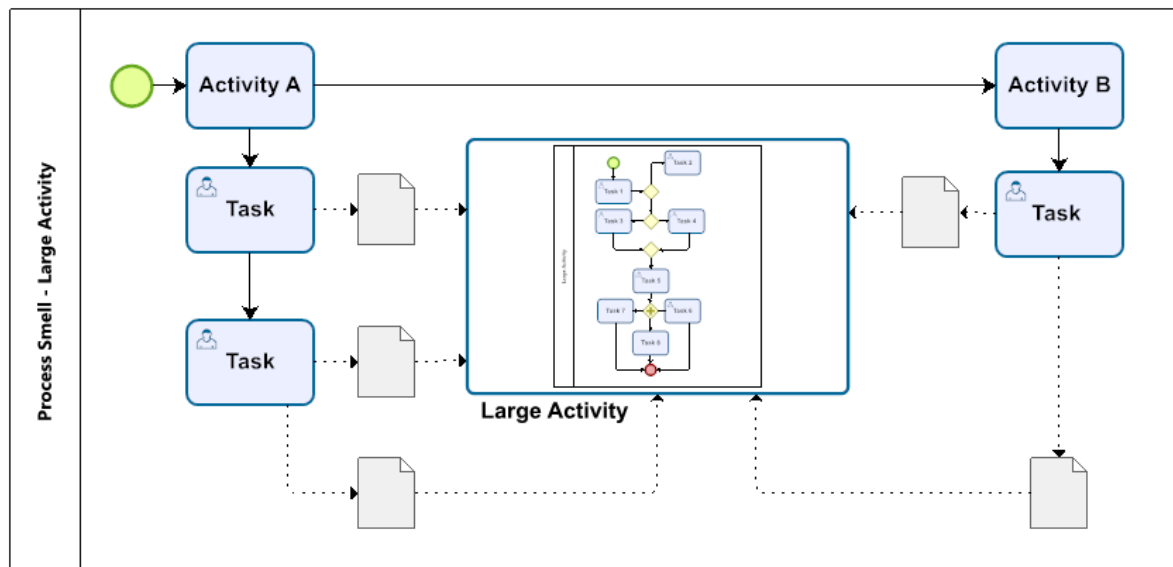
Representação:

Figura 5.4 *Large Activity*

Process Smell: Feature Envy

Elemento Impactado: Tarefa

Descrição: O Feature Envy é representado por uma tarefa que faz uso extensivo de produtos de trabalho de saída de outra atividade. Por esse motivo, talvez essa tarefa deve pertencer a outra atividade.

Possíveis Impactos: O Feature Envy é reconhecido pela distância que os produtos de trabalho de entrada têm em relação à tarefa que as irá utilizá-los, que normalmente estão em outra atividade e não na atividade a qual a tarefa pertence. Assim, quanto maior for a medida dessa distância, menor será a compreensibilidade do executante do processo, principalmente para os casos, onde o executante terá que entender a tarefa fornecedora.

Representação:

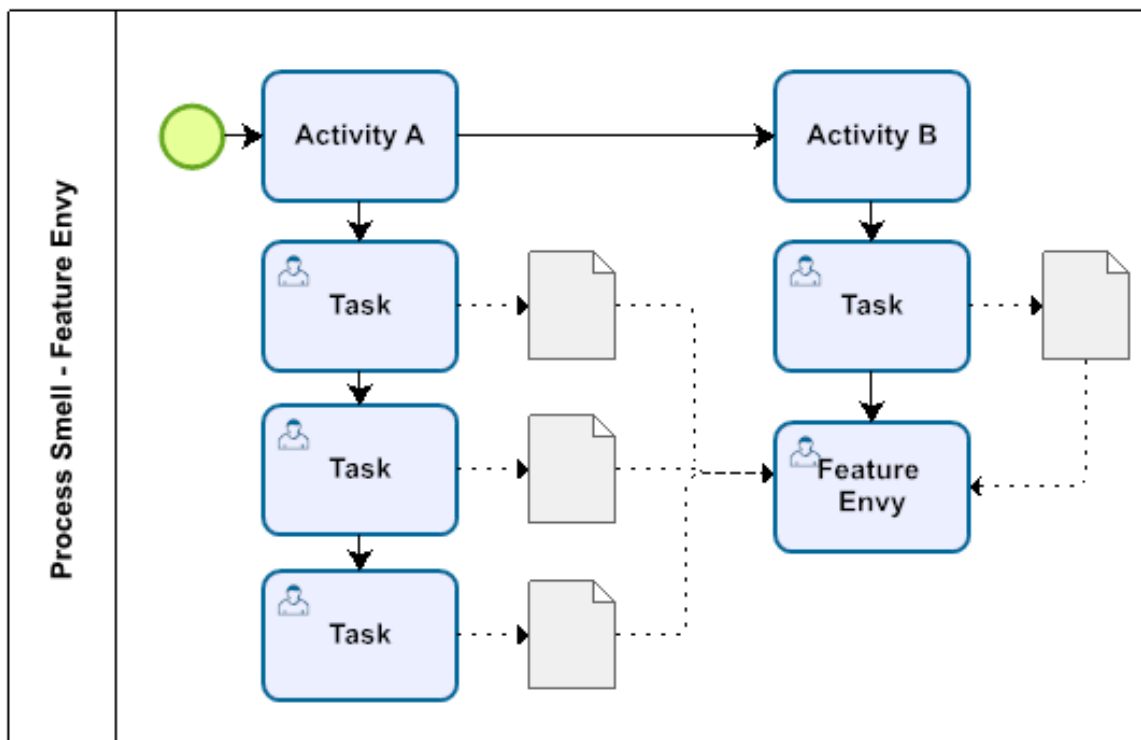


Figura 5.5 *Feature Envy*

Process Smell: *Shotgun Surgery*

Elemento Impactado: Partição

Descrição: O *Shotgun Surgery* se caracteriza por um papel que quando sofre uma mudança (em suas tarefas ou passos) desencadeará a necessidade de muitas pequenas mudanças em vários outros papéis.

Possíveis Impactos: Papéis com *Shotgun Surgery* ocasionam de acoplamento disperso no processo, ou seja, tarefas dependentes que estão espalhadas pelo processo. Assim, mudanças estruturais precisarão de maior esforço na rastreabilidade dos pontos que precisam ser ajustados, de forma a reestruturar o processo de forma consistente. Este fato reduz a capacidade de modificação.

Representação:

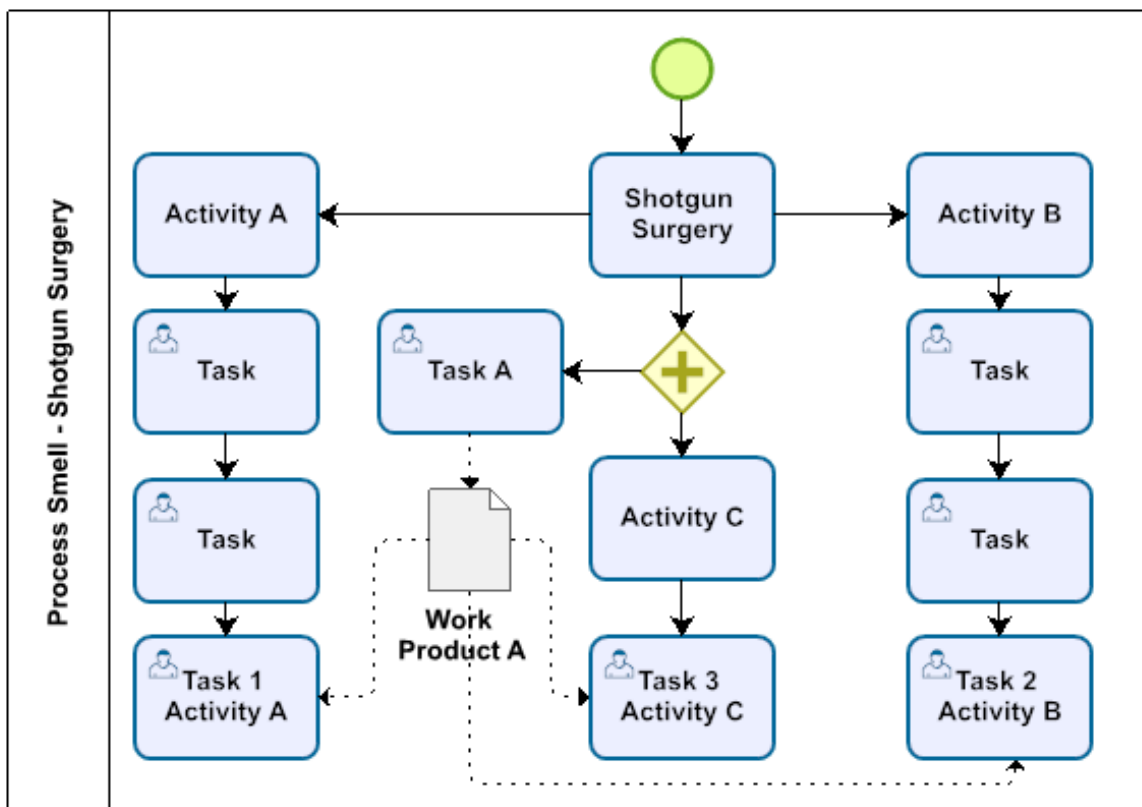
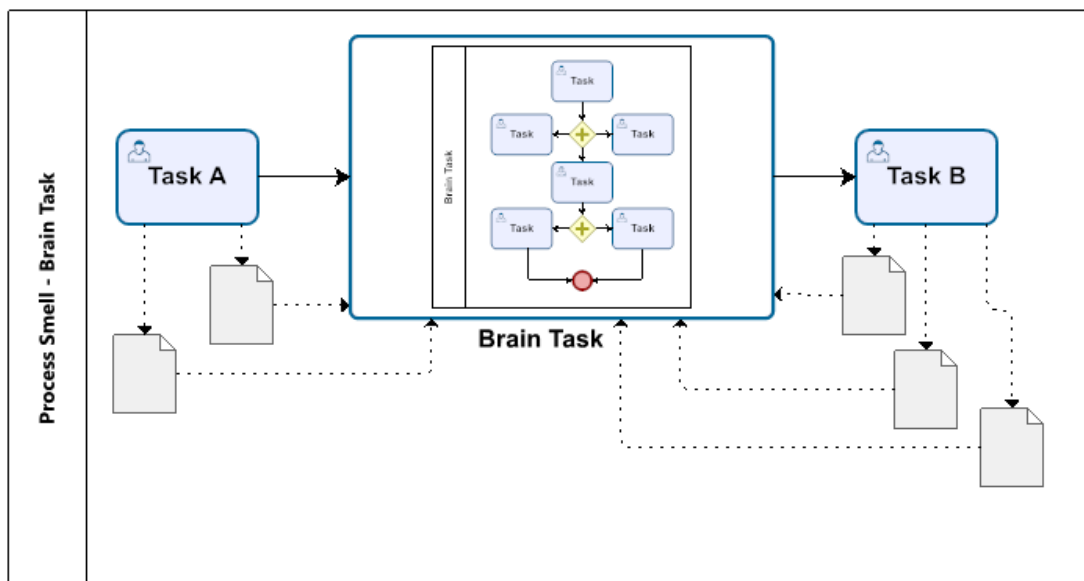


Figura 5.6 *Shotgun Surgery*

Process Smell: *Brain Task***Elemento Impactado:** Tarefa**Descrição:** Uma *Brain Task* é uma tarefa longa que centraliza as ações de um papel, possui muitos passos e fluxos de decisão para estruturar os passos.**Possíveis Impactos:** Apesar da sensação de simplificar o modelo usando poucas tarefas, uma *Brain Task* tem muitos passos e muitos fluxos de decisão, o que aumenta a complexidade e reduz a compreensibilidade da tarefa que sofre esse *process smell*.**Representação:****Figura 5.7** *Brain Task*

Process Smell: *Long Input List*

Elemento Impactado: Tarefa

Descrição: O *Long Input List* é configurado quando uma tarefa precisa de muitos produtos de trabalho de entrada.

Possíveis Impactos: A tarefa que possui uma longa lista de produtos de trabalho de entrada, pode estar sendo impactada por alguns problemas como: acúmulo de responsabilidades ou ser uma tarefa longa por possuir muitos passos e fluxos. E por sua vez pode prejudicar a compreensibilidade do usuário do processo.

Representação:

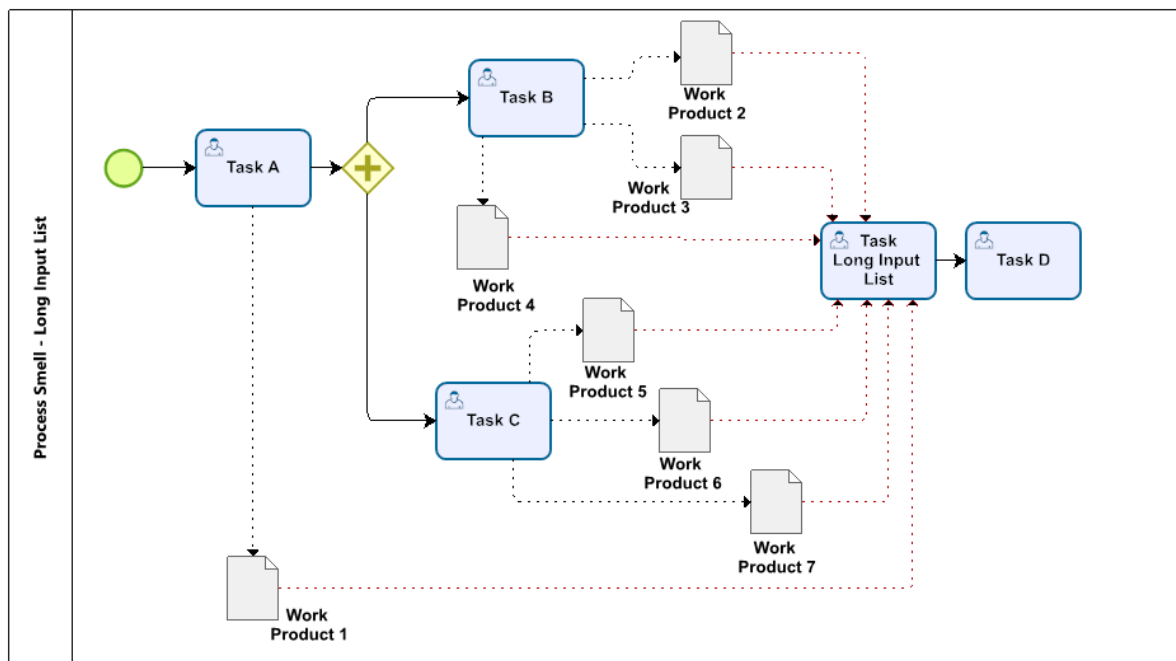


Figura 5.8 *Long Input List*

Process Smell: *Work Product Clumps*

Elemento Impactado: Tarefa

Descrição: *Work Product Clumps* acontecem quando um conjunto de produtos de trabalho de entrada ou saída de certas atividades/tarefas são constantemente vistos juntos. Isso pode indicar que eles devem pertencer à mesma atividade/tarefa.

Possíveis Impactos: *Work Product Clumps* promovem a dispersão de um conjunto de produtos de trabalho, todos necessários para concluir determinadas tarefas do processo. Essa dispersão reduz a modificabilidade do processo, pois será necessário um esforço constante para rastrear esses produtos de trabalho dispersos sempre que houver alguma alteração nesse conjunto.

Representação:

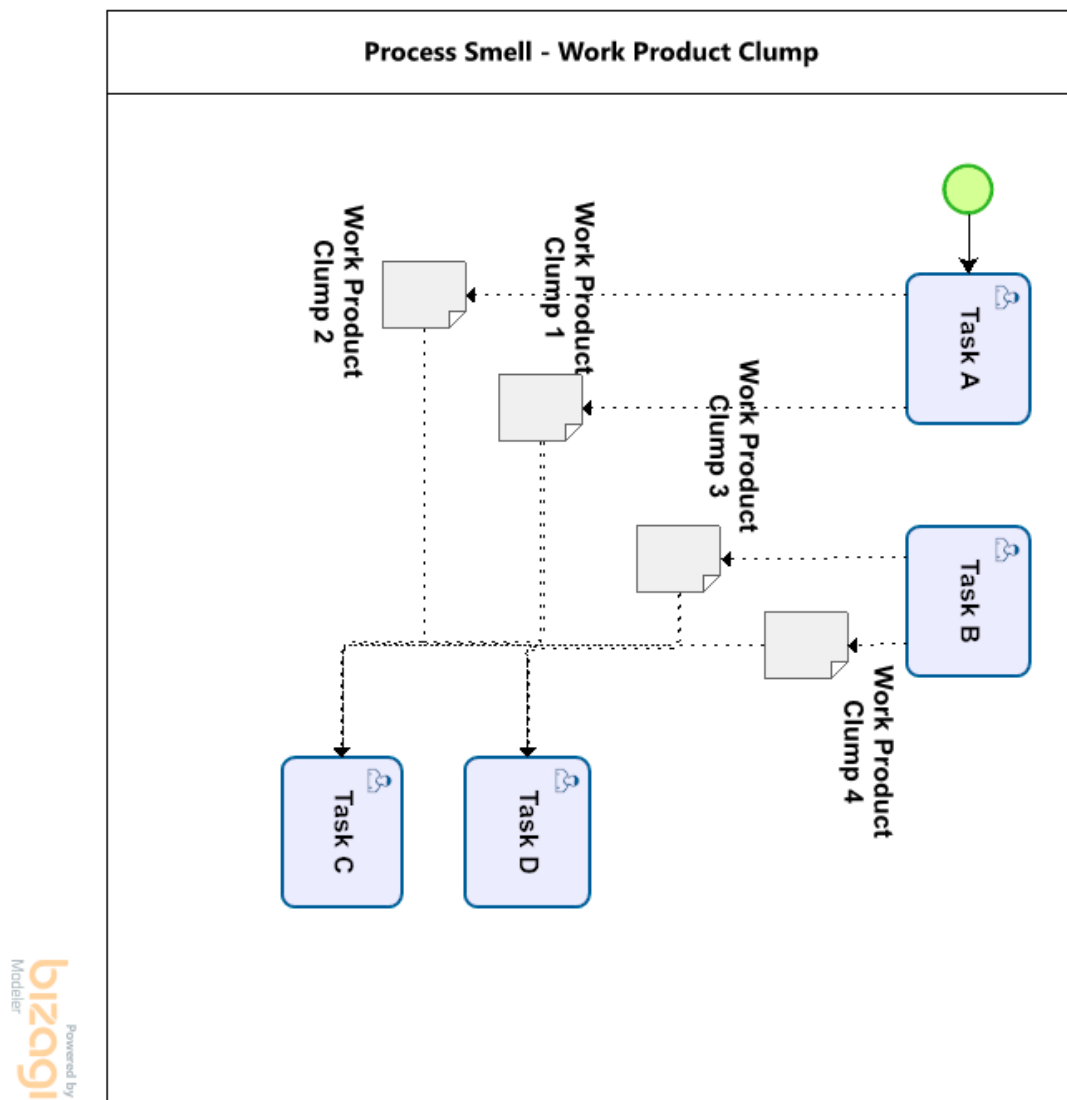
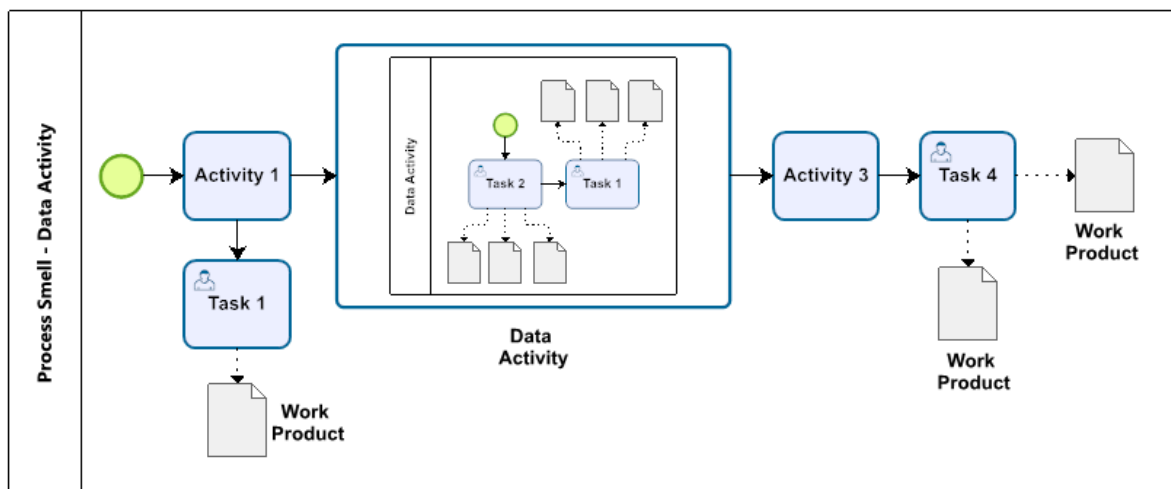


Figura 5.9 *Work Product Clumps*

Process Smell: *Data Activity***Elemento Impactado:** Atividade**Descrição:** Uma atividade se comporta como uma *Data Activity* quando não possui muitas tarefas, mas fornece muitos produtos de trabalho de saída (Saídas).**Possíveis Impactos:** Um agrupamento de muitos produtos de trabalho de saída na mesma atividade pode levar a um acoplamento frouxo (uma relação de dependência entre muitas atividades) ou acoplamento intensivo (uma relação de dependência entre poucas atividades), dificultando a modificação do processo uma vez que existem muitas dependências associados a esta atividade.**Representação:****Figura 5.10** *Data Activity*

5.1 ESPECIFICAÇÃO DOS PROCESS SMELLS

As etapas apresentadas a seguir constituem a primeira fase desta pesquisa que corresponde ao processo de especificação do catálogo de *process smells 2.0*. Inicialmente o catálogo foi especificado e depois avaliado através de um estudo de entrevista, vide Figura 4.1. A especificação do catálogo possui quatro etapas conforme apresentadas pela Figura 4.2:

- Definição de similaridades entre os elementos do SPEM e os elementos do BPMN.
- Modelagem dos *process smells* em SPEM para BPMN.
- Avaliação do catálogo de *process smells 2.0* em BPMN, onde foram apresentados a definição, elemento estrutural do BPMN impactado, representação visual do *process smell* e possíveis impactos que poderiam afetar negativamente os atributos de qualidade.

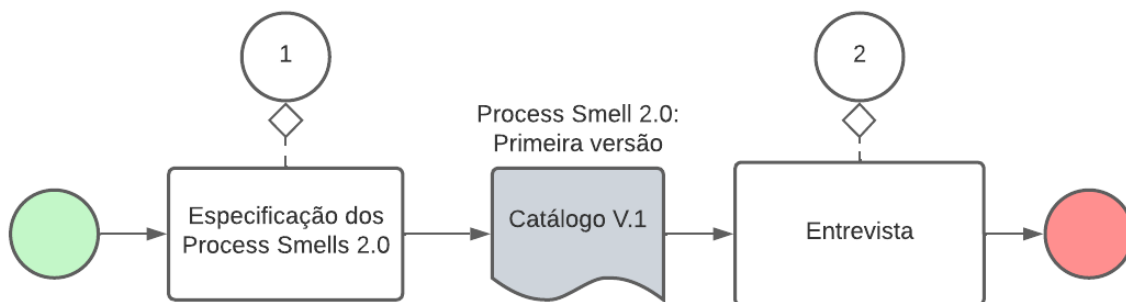


Figura 5.11 Fase de elaboração do Catálogo de Process Smells 2.0

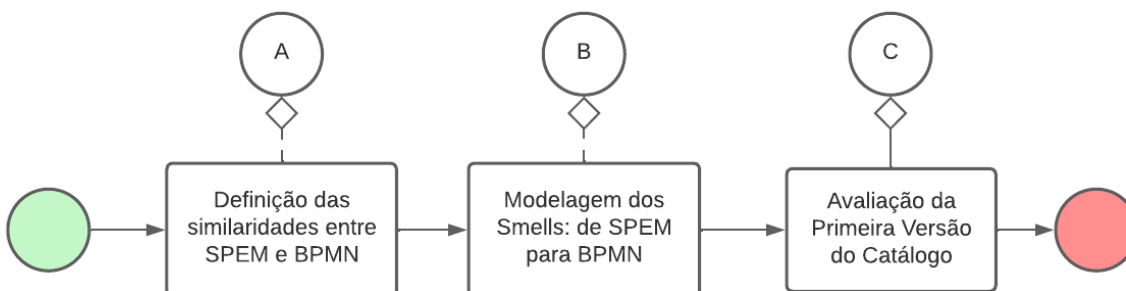


Figura 5.12 Etapas para a especificação do catálogo

5.1.1 Definição de similaridades entre os elementos estruturais do SPEM e BPMN

A primeira etapa da fase de elaboração do catálogo de *process smells 2.0* consistiu em definir (mapear) os elementos que poderiam fazer correspondência do SPEM aos elementos do BPMN. Uma vez que, no trabalho de Santos, Maciel e Sant’Anna (2018) essa análise foi realizada a partir do código OO para SPEM, no trabalho atual buscamos apenas verificar se somente a conversão dos elementos do SPEM para BPMN faria sentido.

Com o apoio do OMG (2010), foram analisadas as definições de cada elemento estrutural para cada uma das notações e se o BPMN apresentava algum elemento que corresponderia a um determinado elemento do SPEM, como por exemplo: no SPEM uma tarefa representa um método quando pensamos em programa de software, mas pensando em BPMN uma task também poderia representar um método se pensado em OO. Mas voltando ao ponto da nossa análise, uma tarefa/task em SPEM poderia fazer correspondência a uma outra tarefa no BPMN, assim essa mesma análise foi feita para os demais elementos. Elementos como atividade, foram complexos de serem abstraídos para o contexto do BPMN, uma vez que, o BPMN detém apenas de um elemento e esse já representaria a tarefa. Então, adaptamos esse elemento para que este fizesse correspondência a uma atividade do SPEM. A Figura 4.3 representa esta descrição:

Buscando representar uma Tarefa, um elemento com ícone de usuário foi adicionado, conforme indica a Figura 4.3, nomeamos o elemento que antes era uma Tarefa para Activity, e chamamos o elemento com o ícone de usuário de Task. Os demais elementos não precisaram desse tipo de ”adequação”.

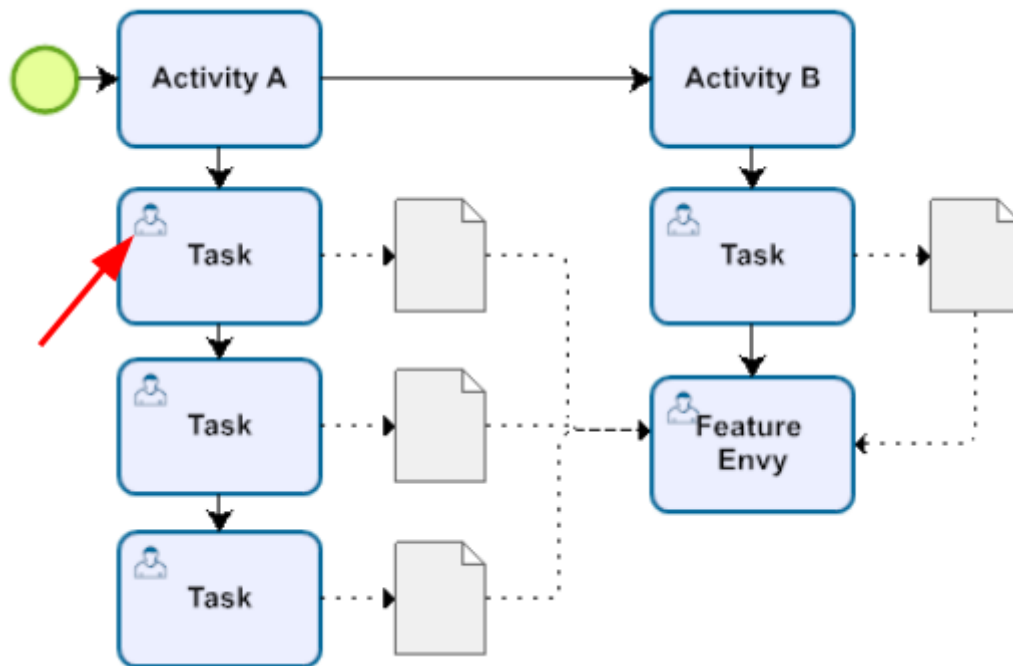


Figura 5.13 Adaptação para a representação de uma atividade em BPMN

Com base nas adaptações realizadas anteriormente, criou-se então a primeira versão do catálogo de process smells 2.0, que será apresentada na próxima seção. Todas as demais propriedades do estudo realizado em SPEM foram mantidas, como por exemplo os atributos de qualidade impactados, pois o intuito do trabalho atual é de estender a versão em SPEM de modo que seja possível verificar a identificação de process smells independente de domínio, bem como possibilitar uma análise comparativa em relação aos resultados dos dois trabalhos, uma vez que os impactos poderiam ser menos ou mais percebidos em uma notação do que em outra.

5.2 AVALIAÇÃO DA PRIMEIRA VERSÃO DO CATÁLOGO DE PROCESS SMELLS 2.0

A avaliação da concepção da proposta do catálogo de *process smells 2.0* em relação aos objetivos da pesquisa foram providos através de um estudo de entrevista e serão abordados neste capítulo. A organização deste capítulo segue conforme a Figura 5.1, onde a seção 5.1 aborda o *design* do estudo, 5.2 explica os pontos obtidos através da aplicação do piloto, 5.3 explica sobre a aplicação do estudo de entrevista, 5.4 apresenta a análise dos resultados obtidos e por fim as seções 5.5 e 5.6 tratam respectivamente das ameaças à validade e conclusões. O objetivo do Estudo de Entrevista foi validar a especificação dos *process smells 2.0* propostos, juntamente com os profissionais participantes, onde as definições, representações e possíveis impactos dos *process smells* puderam ser avaliados. A partir das respostas obtidas

foi possível verificar se os *process smells 2.0* propostos eram percebidos como uma anomalia do processo, bem como compreender o quanto alguns *process smells* se mostraram mais significativos que outros.



Figura 5.14 Passo a passo do estudo de entrevista

5.3 DESIGN DO ESTUDO DE ENTREVISTA

O projeto do estudo de entrevista seguiu o protocolo baseado nos métodos propostos por Seaman (1999). Para a autora em questão, usar métodos qualitativos forçam o pesquisador a mergulhar na complexidade do problema, assim, os resultados são mais ricos e informativos. Neste estudo a entrevista semi-estruturada foi escolhida como a principal técnica para a coleta de dados, por se tratar de um instrumento flexível. Utilizou-se de métodos quantitativos e qualitativos, uma vez que, a análise qualitativa é usada também para responder os "porquês" de questões abordadas pela pesquisa quantitativa.

A seguir serão apresentados a Estrutura do Questionário, Piloto do Estudo de Entrevista, Execução do Estudo de Entrevista, Análise do Estudo de Entrevista, Caracterização do Perfil dos Participantes e a Análise dos Dados Quantitativos e Qualitativos.

A estrutura do estudo de entrevista foi definida a partir das seguintes características:

Unidade de observação: Ponto de vista dos participantes em relação ao contexto abordado.

Unidade de análise: Catálogo de *Process Smells 2.0* proposto para processo de software modelados em BPMN.

Objetivo: Através da pesquisa exploratória validar o conjunto de *Process Smells 2.0*, para processos especificados em BPMN, validando suas definições, possíveis impactos quanto aos fatores de qualidade do processo compreensibilidade, modificabilidade ou a de ambos.

O estudo de entrevista buscou responder às seguinte questões de pesquisa:

- QP1. Os participantes concordam com a definição dos *process smells*?
- QP2. Os participantes concordam com os impactos negativos à qualidade?
- QP3. Os participantes concordam que faz sentido refatorar processos?

Participantes: Por conveniência e disponibilidade de participação foram selecionados sete profissionais para realização deste estudo.

Coleta de dados: Os dados foram coletados a partir da realização de entrevistas individuais e aplicação do questionário no intuito de verificar a concordância com a definição de cada *smell* e seus impactos negativos para os subatributos de qualidade do processo.

Análise e interpretação dos dados: Os dados foram analisados e interpretados a partir da taxa de aceitação juntamente com as percepções dos entrevistados por meio das justificativas de cada questão. Desse modo, as respostas obtidas foram analisadas do ponto de vista qualitativo e quantitativo. O aspecto quantitativo deu-se em relação as taxas de aceitação para cada *process smells* e o aspecto qualitativo está relacionado as justificativas de aceitação ou rejeição dos *process smells*.

5.4 ESTRUTURA DO QUESTIONÁRIO

A estrutura do questionário foi constituída por três seções, onde a primeira seção trouxe questões que compreendem o perfil e a experiência profissional do analista de processo de software. A segunda seção apresentou questões do tipo múltipla escolha utilizando a *Likert Scale 5 point* para a validação de concordância ou não com a definição dos *process smells* e o impacto negativo ao subatributo de qualidade relacionado. Para cada pergunta a justificativa em caso de discordância foi obrigatória e por fim a terceira seção contém questões de fechamento do estudo de entrevista, buscando entender a percepção do participante em relação a aplicação do questionário, bem como a clareza e a relevância da proposta. As questões sobre os *process smells* seguem o formato abaixo, conforme exemplo do *Message Chains*. O questionário completo se encontra disponível no Apêndice A.

- (a) Quanto a definição do process smell.
 - a. Uma cadeia longa de tarefas pode ser considerada um *process smell*? [Concordo plenamente, Concordo parcialmente, Neutro, Discordo plenamente, Discordo parcialmente].
 - b. Justifique caso sua resposta seja negativa. (Opcional para respostas positivas).
- (b) Verificação dos impactos negativos para a compreensibilidade.
 - a. Esse *process smell* afeta negativamente o atributo de qualidade compreensibilidade? [Concordo plenamente, Concordo parcialmente, Neutro, Discordo plenamente, Discordo parcialmente].
 - b. Justifique caso sua resposta seja negativa. (Opcional para respostas positivas).

5.4.1 Piloto do Estudo de Entrevista

Diante de todas as definições abordadas nas seções anteriores quanto a preparação do estudo de entrevista foi estruturada a aplicação do piloto com dois profissionais.

O intuito do piloto é verificar a estrutura proposta e obter *feedback* de possíveis ajustes a serem feitos caso necessário, antes da aplicação definitiva do estudo. Neste sentido, com a aplicação do piloto foram obtidos os seguintes pontos.

- (a) O tempo médio para a execução do estudo de entrevista era de 60 minutos;
- (b) Possibilidade de obter dados qualitativos a partir das justificativas e considerações dos entrevistados;
- (c) Referente a análise qualitativa das justificativas foi realizada a gravação da entrevista (autorizada pelo entrevistado) para que capturasse de forma completa o posicionamento dos participantes e realizar o detalhamento das opiniões;
- (d) A reformulação das questões também foi um ponto identificado, uma vez que as perguntas se iniciavam da seguinte forma: “Você concorda que...”, soando de forma indutiva ao participante.

5.4.2 Execução do Estudo de Entrevista

Antes da aplicação do questionário foi enviado ao participante um convite de participação no estudo de entrevista bem como um termo de consentimento (Ver Apêndice A) para formalizar o estudo, visto que seria necessário realizar a gravação da entrevista para coleta de dados, bem como para posteriormente realizar a transcrição. Após os trâmites de formalização, o entrevistado disponibilizava o horário para participar da entrevista, preferencialmente via Skype. Cada seção foi apresentada com uma introdução, perguntas de abertura, questões sobre os *process smells* e questões de fechamento do estudo. Na introdução foram apresentados o objetivo da pesquisa, os elementos do BPMN que compõem a representação dos *Process Smells* e as definições dos subatributos de qualidade considerados. Em seguida os *smells* eram apresentados junto as perguntas relativas a eles. Para cada *process smell* apresentado foram feitas perguntas baseadas em um questionário composto por alternativas de múltipla escolha (*Likert Scale 5 point*) pois a possibilidade de capturar nuances na percepção do estímulo que não são vislumbrados em um julgamento binário é bem maior. As questões tratavam sobre a definição do *process smell* e os possíveis impactos aos atributos (compreensibilidade e/ou modificabilidade). Para encerrar o estudo, as questões de fechamento objetivaram coletar as percepções dos participantes sobre a proposta do catálogo de *process smells 2.0* e sobre o estudo de entrevista executado.

5.5 ANÁLISE DO ESTUDO DE ENTREVISTA

Esta seção apresentará os resultados do estudo de entrevista para a validação do catálogo de *process smells 2.0*, onde os seguintes tópicos serão abordados: caracterização do perfil dos participantes, dados quantitativos e qualitativos resultantes das justificativas, comparação inicial dos resultados entre BPMN e SPEM e as questões

de fechamento do questionário. Ameaças à validade também serão discutidos nesta seção.

5.5.1 Caracterização do Perfil dos Participantes

A análise do perfil dos participantes foi baseada nos seus conhecimentos sobre processo de software quanto ao tempo de experiência e a quantidade de processos modelados.

Com base nos resultados obtidos em relação ao tempo de experiência e ao número de processos modelados, a maior parte dos profissionais possuem experiência acima de um ano com especificação de processo e já modelaram acima de dois processos de software. Ao todo foram entrevistadas sete pessoas. A tabela 5.1 apresenta individualmente o tempo de experiência e a quantidade de processos modelados por cada participante.

Tabela 5.1 Tempo de experiência e processos modelados.

Participantes (P)	Tempo de Experiência	Processos Modelados
P1	Mais de 4 anos	Mais de 15 processos
P2	Entre 1 a 2 anos	Entre 2 a 5 processos
P3	Entre 2 a 3 anos	Entre 6 a 15 processos
P4	Menos que 6 meses	Entre 2 a 5 processos
P5	Entre 1 a 2 anos	Entre 6 a 15 processos
P6	Entre 6 meses a 1 ano	Entre 2 a 5 processos
P7	Menos que 6 meses	Entre 2 a 5 processos

5.5.2 Análise dos Dados Quantitativos e Qualitativos

Os resultados obtidos através do questionário aplicado durante a entrevista semi-estruturada foram primeiramente analisados utilizando a abordagem quantitativa para estabelecer o Ranking Médio (RM), para mensurar o grau de concordância ou discordância na escala *Likert* de 5 pontos. O método de análise da escala *Likert* apresentado por Malhotra (2001) e Oliveira (2005) foi utilizado para a realização do cálculo do RM. A Tabela 5.2 exemplifica este cálculo.

Tabela 5.2 Resultado de aceitação do *Message Chains*

MESSAGE CHAINS	FREQUÊNCIA DE PARTICIPANTES					
1 - Uma cadeia longa de tarefas pode ser considerada um process smell?	1	2	3	4	5	AR
	6			2	1	2,1
1.1 - Este process smell afeta negativamente o atributo de qualidade compreensibilidade?						AR
		1	2	3	4	5
		3	4		2	2,1

$$\text{Ranking Médio} = (6 \times 1) + (2 \times 4) + (1 \times 5) = 19$$

$$\text{RM} = 19 / (6 + 2 + 1) = 2,1$$

Através da obtenção do RM pôde se verificar a concordância ou discordância das questões avaliadas, onde os valores menores que 3 são considerados como concordantes e, maiores que 3, como discordantes, considerando uma escala de 5 pontos. O valor central da escala, neste caso o 3, foi considerado “indiferente” ou “sem opinião”, sendo o “ponto neutro”, equivalente aos casos em que os respondentes foram imparciais.

Tabela 5.3 *Process Smells* Resultado da Aceitação

Process Smell	RM-D	Sub-atributo	RM-S
Feature Envy	1,4	Compreensibilidade	2,0
Brain Task	1,1	Compreensibilidade	1,5
Long Input List	1,0	Compreensibilidade	1,0
Divergent Change	1,6	Compreensibilidade	1,5
Message Chains	2,1	Compreensibilidade	2,1
Shotgun Surgery	1,3	Modificabilidade	1,0
Brain Activity	1,8	Compreensibilidade	2,3
Large Activity	1,0	Compreensibilidade	1,4
Work Product Clumps	2,5	Modificabilidade	2,4
Data Activity	3,5	Modificabilidade	3,7

Fonte: Resultado da pesquisa. **Elaboração própria, 2022.**

Estratificado: Ranking Médio (RM); Definição (D); Subatributo (S)

O RM definiu as taxas de aceitação de cada *process smell*, como pode ser visto na Tabela 5.3, onde são apresentados os valores referentes ao *process smell* avaliado, RM da definição (RM-D), subatributo que foi avaliado naquele *process smell* e RM do subatributo (RM-S).

Conforme apresentado na Tabela 5.3, dentre os dez *process smells* apresentados, nove foram bem aceitos, com exceção do *Data Activity* que obteve 3,6 na escala, ficando na colocação de não aceito na opinião dos participantes da entrevista. O *Feature Envy* foi um dos *process smells* mais bem aceito, obtendo uma avaliação de 1,1 de média geral.

Os dados qualitativos detalham as argumentações dos participantes quanto aos *process smells*, e foram organizados de modo a discutir os possíveis motivos que levaram a aceitação ou rejeição de determinados *process smells*. A ordem de disposição dos *process smells* é crescente, conforme a média geral de aceitação.

5.6 COMPARAÇÃO DOS RESULTADOS ENTRE O BPMN E SPEM

Os resultados obtidos neste estudo serão comparados nesta seção aos resultados da primeira fase do trabalho de Santos, Maciel e Sant'Anna (2018), onde também foram modelados dez *process smells* em SPEM.

Para avaliar os *process smells* Santos, Maciel e Sant'Anna (2018) usou como base as respostas dicotômicas para a elaboração da taxa de aceitação, onde a taxa de aceitação é igual a soma das respostas positivas de cada *process smell* e dividida pelo total de participantes. A taxa foi aplicada para avaliar os elementos: definição, compreensibilidade e modificabilidade. A Tabela 5.4 apresenta os resultados dos trabalhos para a aceitação da definição e dos subatributos de cada *process smell*.

Apesar de ter usado uma métrica de avaliação diferente ainda assim é possível analisar os resultados obtidos por cada notação.

Tabela 5.4 Comparação de Resultados entre BPMN e SPEM

Process Smell	Definição		Sub-atributos			S
	BPMN	X SPEM	BPMN	X SPEM		
Feature Envy	1,4	0,83	2,0	0,83		C
Brain Task	1,1	0,83	1,5	0,93		C
Long Input List	1,0	0,83	1,0	0,83		C
Divergent Change	1,6	0,92	1,5	0,92		C
Message Chains	2,1	0,67	2,1	0,58		C
Shotgun Surgery	1,3	1,00	1,0	1,00		M
Brain Activity	1,8	0,58	2,3	0,58		C
Large Activity	1,0	0,92	1,4	0,96		C
Work Product Clumps	2,5	0,83	2,4	0,83		M
Data Activity	3,5	0,83	3,7	0,83		M

Source: Elaboração própria, com base nos resultados obtidos na pesquisa e no trabalho de Santos, Maciel e Sant'Anna (2018).

Estratificado: Sub-atributos (S); Compreensibilidade (C); Modificabilidade (M).

As observações em relação a esta comparação serão abordadas a seguir, tratando individualmente cada *process smell*.

5.6.1 Message Chains

Conforme pode ser visto na Tabela 5.4, enquanto na especificação em BPMN o *Message Chains* foi bem aceito pelos participantes, em SPEM esse *process smell* não foi tão aceito em relação a aceitação dos demais *process smells*, obtendo 63% de aceitação, sendo 67% quanto a sua definição e 58% em relação ao impacto negativo no atributo de qualidade compreensibilidade. Em relação aos aspectos qualitativos quanto a rejeição desse *process smell* no estudo em SPEM mais da metade das opiniões compreenderam que não há um *process smell* neste caso, enquanto as demais opiniões compreendem que o problema do *Message Chains* para o processo corresponde do loop entre as tarefas, o que implica em aumentar a complexidade, e não na troca de produtos de trabalho entre um grande conjunto de tarefas referenciando o acoplamento entre as mesmas (SANTOS; MACIEL; SANT'ANNA, 2018). Já em BPMN embora o *Message Chains* não tenha sido rejeitado, os participantes opinaram justificando suas rejeições quanto ao atributo impactado, discordando que, embora o processo tenha cadeias longas, isso não necessariamente impactaria negativamente. Dentre os 10 *process smells* modelados em BPMN o *Message Chains* ficou em quinto lugar quanto a sua aceitação, enquanto em SPEM foi o *process smell* com menor aceitação no catálogo.

5.6.2 Brain Activity

O *Brain Activity* também não foi um *process smell* bem aceito quanto a sua definição e ao seu impacto negativo no atributo de qualidade compreensibilidade, contendo uma taxa de aceitação total de 58%. Em BPMN o modelo obteve uma taxa de aceitação total no Ranking Médio de 1,9, sendo 2,1 para a sua definição e 1,8 para o impacto negativo ao atributo de qualidade compreensibilidade. Quanto a análise qualitativa em relação as justificativas de rejeição os participantes mencionaram a complexidade para a realização de modificações no processo e enfatizaram que a quantidade de fluxos de decisões para estruturar as tarefas poderiam interferir na qualidade do processo. Em SPEM os participantes justificaram a rejeição desse *process smell* citando que: "a atividade por ter um grau de abstração maior ela pode ter muitas sub tarefas sem que isso seja um problema de modelagem" (SANTOS; MACIEL; SANT'ANNA, 2018). De acordo com o estudo de Santos, Maciel e Sant'Anna (2018), para os participantes, acumular tarefas e fluxos é uma característica inerente a uma atividade e tão pouco viria a reduzir a compreensibilidade da atividade.

5.6.3 Data Activity

Enquanto em BPMN esse *process smell* foi rejeitado pelos participantes, obtendo uma taxa acima de três na escala do RM totalizando uma média geral de 3,6, em SPEM o *Data Activity* foi bem aceito obtendo a taxa total de aceitação de 83% tanto para a sua definição quanto para o impacto ao atributo de qualidade modificabilidade. Embora tenha sido aceito no SPEM e rejeitado no BPMN, curiosamente

as justificativas para a rejeição em ambas as notações foram bem parecidas. Os participantes explicitaram que, apesar de não ser considerada uma atividade principal, a entrega de muitos produtos de trabalho não necessariamente quer dizer acoplamento entre as atividades do processo e que nem sempre os produtos de trabalho poderão ser reutilizados por uma outra atividade. Em geral, nos dois estudos a visão dos participantes quanto a rejeição é que o não acoplamento não gera um impacto significativo na modificabilidade do processo.

5.6.4 Divergent Change

Totalizando 1,6 no RM de aceitação, esse *process smell* obteve 1,7 de aceitação quanto a sua definição e 1,5 para o impacto negativo ao atributo de qualidade compreensibilidade no estudo voltado para o BPMN. Em SPEM o *Divergent Change* ficou entre os três *process smells* mais bem aceitos do catálogo, obtendo uma taxa de aceite de 92% tanto para a sua definição quanto para ao atributo de qualidade compreensibilidade. Em relação aos aspectos de rejeição em ambos os trabalhos as justificativas são de que, se a atividade atender as necessidades que foram designadas a ela dentro do processo, a baixa coesão entre as tarefas não seriam um problema e nem impactaria negativamente o atributo de qualidade compreensibilidade.

5.6.5 Large Activity

O *Large Activity* obteve uma taxa de aceitação geral de 2,0, em caso de uma análise na escala essa média representaria a concordância parcial dos participantes. Já em SPEM o *process smell* foi bem aceito, totalizando uma média de aceitação de 95%, onde 92% em relação a sua definição, 92% para o impacto negativo em relação ao atributo compreensibilidade e 100% de aceitação para a percepção quanto aos impactos negativos ao atributo de modificabilidade. No SPEM, de acordo com Santos, Maciel e Sant'Anna (2018), a rejeição quanto a definição foi relacionada ao fato da descrição deste *process smell* não sugerir a quantidade de papéis envolvidos na atividade. Já a rejeição quanto ao impacto na compreensibilidade propôs que uma atividade por representar uma unidade lógica de trabalho do processo mesmo com grande tamanho e a alta complexidade não impactaria negativamente a compreensibilidade.

5.6.6 Feature Envy

No presente estudo, o *Feature Envy* foi o *process smell* mais bem aceito no catálogo, ficando com uma média total de 1,1, onde 1,2 se deram para a sua aceitação quanto a definição e 1,1 na avaliação de impacto ao atributo de qualidade compreensibilidade. Em SPEM esse mesmo *process smell* obteve taxa de aceitação total de 83% recebendo essa mesma porcentagem para a sua definição e ao atributo de qualidade compreensibilidade. No SPEM foi sinalizado pelos participantes que em alguns

casos as dependências deste *process smell* são necessárias na especificação do processo. No BPMN não foram sinalizadas justificativas para casos de rejeição quanto ao *Feature Envy* por parte dos participantes.

5.6.7 Shotgun Surgery

Esse foi o *process smell* com 100% de aceitação no catálogo especificado em SPEM, não havendo discordâncias das características por parte dos participantes. No trabalho atual a média alcançada pelo *Shotgun Surgery* foi de 1,7, onde 1,7 da aprovação foi para a sua definição e 1,8 para o atributo de qualidade modificabilidade. Quanto as percepções para esse *process smell* os participantes mencionaram o retrabalho no caso de modificação neste processo.

5.6.8 Work Product Clump

Em BPMN o *Work Product Clump* obteve uma média geral de 2,4 na escala, aproximando-se mais no ponto neutro. O RM de aceitação quanto a sua definição foi de 2,5 e 2,4 para o impacto no atributo de qualidade modificabilidade. Embora este *process smell* tenha ficado próximo do ponto neutro isso não gerou informações suficientes que pudessem descaracterizá-lo. Em SPEM este *process smell* alcançou a taxa de 83% de aceitação, valor este que foi igualmente atribuído para a sua definição e para seu impacto negativo ao atributo de qualidade modificabilidade. Em relação a rejeição desse *process smell* em SPEM os participantes apenas não acreditam que um conjunto de produtos de trabalho constantemente vistos juntos venham ser um problema ou que impacte negativamente o atributo de qualidade modificabilidade. No SPEM, assim como o *Long Input List*, este valor possui uma taxa de aceitação expressiva.

5.6.9 Brain Task

Na especificação do catálogo em BPMN o *Brain Task* ocupou o segundo lugar na escala de aceitação, obteve 1,7 de aceitação quanto a sua definição na escala RM e 1,4 de aceitação quanto ao impacto para o atributo de qualidade compreensibilidade, totalizando uma aceitação média de 1,5. Em relação aos aspectos qualitativos deste estudo os participantes argumentaram que apesar da complexidade quanto a compreensão, em algum cenário é necessário o uso de uma tarefa centralizadora, e que na percepção deles para a realização de uma mudança o foco seria em uma única tarefa. Em SPEM o *Brain Task* obteve um total geral de aceitação de 88%, 83% de aceitação para a sua definição e 92% para o impacto negativo ao atributo de qualidade compreensibilidade. Conforme citado por Santos, Maciel e Sant'Anna (2018), o *Brain Task* recebeu rejeições dos participantes que argumentaram que uma tarefa pode ser grande e complexa uma vez que busca atender uma parte realizável do processo. Assim, não concordaram que o *Brain Task* poderia causar qualquer

impacto negativo ao processo. Na segunda situação especificamente se tratando da propriedade complexidade foi indicado que seria necessário informar a quantidade de papéis envolvidos na tarefa para melhor percepção da complexidade da tarefa *Brain Task*. Contudo, o *Brain Task* obteve uma taxa de aceitação considerável.

5.6.10 Long Input List

O *Long Input List* alcançou a média geral de aceitação de 1,6, onde 1,5 desse valor foi atribuído para a sua definição e 1,7 para o impacto negativo ao atributo de qualidade compreensibilidade. Em aspectos qualitativos foram analisadas as justificativas dos participantes onde foram argumentadas que a complexidade em relação a compreensibilidade fica eminente até mesmo na representação do processo em BPMN. Também foi citada possibilidades de problemas a níveis de usuário em relação a este process smell, no que se diz respeito a quantidade de dados de entrada. No estudo de Santos, Maciel e Sant'Anna (2018), o *Long Input List* obteve uma taxa bastante expressiva, somando 83% de aceitação, valor igualmente atribuído para a sua definição e para o impacto negativo ao atributo de qualidade compreensibilidade. As respostas de rejeições obtidas não forneceram informações suficientes para descaracterizar este *process smell*.

5.7 AMEAÇAS À VALIDADE

Como ameaças a validade deste estudo pode-se destacar a falta de conhecimento total e o conhecimento superficial de dois participantes quanto ao BPMN. Apesar disso, a representação de cada *process smell* e a apresentação dos artefatos facilitou o entendimento na hora da entrevista. Os termos *bad smells* e *process smells* apesar de terem sido apresentados no início da entrevista, também foram questionados durante a execução do estudo o que não veio a inviabilizá-lo, uma vez que, sempre que o participante sentia a necessidade de lembrar o conceito isso era repassado e exemplificado da forma mais clara possível. Em relação a proposição da existência de *process smells* em processo de software, as respostas obtidas ficaram entre relevante, muito relevante e muitíssimo relevante. Sobre o tempo decorrido para a realização da entrevista dois dos participantes alegaram o tempo exaustivo em relação a entrevista e a extensão do catálogo, onde o tempo médio foi de 60 minutos. Ao final, os participantes avaliaram positivamente a clareza do conteúdo, onde as respostas ficaram entre suficiente, adequado e muito adequado para o entendimento e avaliação dos *process smells 2.0* propostos.

5.8 DISCUSSÃO

A partir dos resultados alcançados nesta pesquisa, percebeu-se que os profissionais consideram importante algumas preocupações voltadas para a especificação do processo, evidenciam-se esse fato a aceitação em relação a definição dos *process smells*

2.0 e seus impactos sobre os seus atributos. As discordâncias pontuadas neste estudo apontam para um possível refinamento no catálogo, bem como reforçam algumas formas de especificação de processo que são evitadas pelos profissionais que participaram. Conclui-se também que apesar do alto índice de aceitação alguns *process smells* são mais significativos que outros.

De acordo com os resultados obtidos e até mesmo a partir das percepções durante a entrevista ficou evidente que o método de pesquisa se mostrou convergente aos resultados obtidos. Sobre os resultados comparativos pode-se destacar que a aceitação de um determinado *process smell* em um estudo não implica diretamente para que esse mesmo *process smell* venha ser aceito em outro, como é o caso do *Data Activity* que foi expressivamente aceito no SPEM, mas em BPMN esse *process smell* não foi aceito. A percepção quanto aos resultados comparativos levam em consideração a quantidade de participantes em cada estudo, o que pode ser um fator influenciável nos resultados.

CATÁLOGO DE PROCESS SMELLS 2.0 - VERSÃO FINAL

6.1 REFINAMENTO DO CATÁLOGO

Na primeira etapa deste estudo, dez Process Smells foram caracterizados contendo suas definições, elemento estrutural impactado do BPMN, representação visual e possíveis impactos que poderiam afetar negativamente os atributos de qualidade. A avaliação inicial foi feita por sete profissionais, que concordaram com a definição e os possíveis impactos dos Process Smells.

Baseado no objetivo de complementar o primeiro estudo, neste segundo estudo surgiu a necessidade de refinar o catálogo quanto a nomenclatura e a estrutura dos elementos do BPMN. A figura abaixo, representa as atividades executadas na segunda etapa deste estudo.

A seguir serão discutidas o que foi realizado em cada atividade.

- **1 - Estudo Complementar 1:** Após a primeira etapa, houve a necessidade de realizar um estudo complementar, onde denominamos “Estudo complementar 1”. O objetivo principal desta atividade era buscar por mais profissionais capacitados e que pudessem participar do estudo. Apesar do resultado da primeira etapa ter sido satisfatório com a participação de sete profissionais, um dos objetivos do estudo era ampliar o número de avaliações, visando obter o máximo de percepções possíveis a respeito deste trabalho.
- **2 - Remodelagem dos *Smells*:** No início do Estudo Complementar 1, levamos em consideração a opinião de alguns profissionais sobre a possibilidade de remodelar os *smells*, pois na visão deles alguns modelos não faziam sentido quando pensados em BPMN. Então, com base nas sugestões obtidas para a realização de uma melhor modelagem, os *smells* foram remodelados. Um

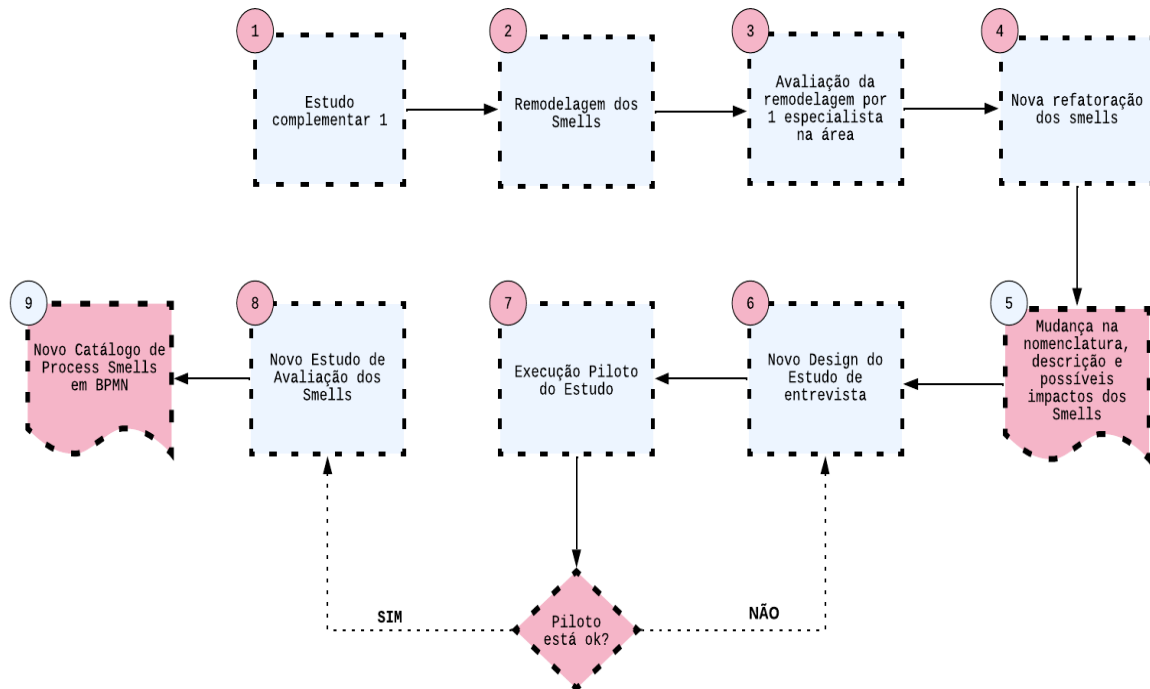


Figura 6.1 Etapas Metodológicas para o Refinamento do Catálogo

exemplo dessa alteração é que inicialmente definimos uma atividade dentro do BPMN para fazer correspondência a uma atividade no SPEM, no entanto com base no conhecimento profissional de alguns participantes, o BPMN não tem um elemento atividade, então foi pontuado que nos cenários onde essa correspondência foi realizada o mais ideal é que um "papal" representaria melhor uma atividade e os possíveis impactos seriam voltados a esse papal, na qual chamamos de *partition* e que será comentado a seguir.

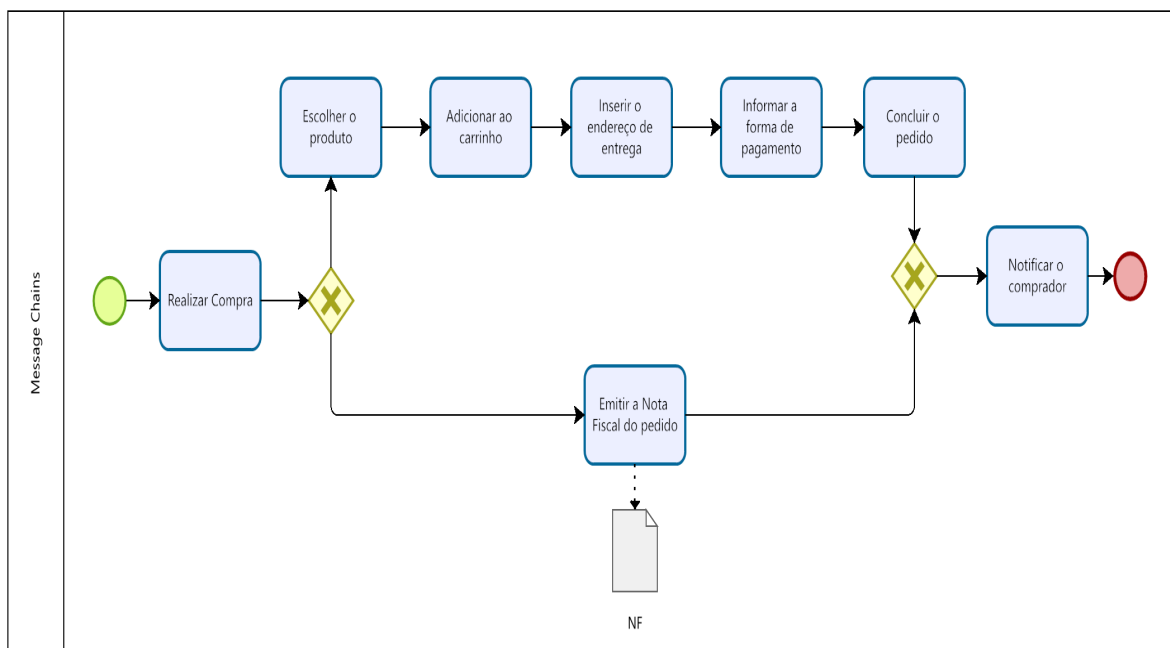
- **3 - Avaliação da Remodelagem:** Com os *smells* remodelados, uma nova avaliação foi realizada por um especialista na área e nesta etapa, dois dos dez *smells* propostos não faziam tanto sentido quanto a sua modelagem em BPMN e foram removidos do catálogo com a finalidade de serem investigados à parte, foram eles *Data Activity* e *Work Product Clumps*.
- **4 - Nova Refatoração dos *Smells*:** A partir dos pontos levantados pela avaliação da nova modelagem realizada anteriormente, uma nova refatoração dos *smells* se fez necessária, não nos *smells* propriamente dito, mas as nomenclaturas e elementos impactados precisaram ser repensados. Assim, onde tínhamos "*Brain Activity*", passamos a ter "*Brain Partition*", onde o elemento impactado era "*Activity*" passou a ser "*Partition*". As descrições dos *smells* que sofreram essas alterações também foram adaptadas.
- **5 - Novas nomenclaturas para os *Smells*:** Esta atividade pode ser consi-

derada como um artefato gerado a partir da execução da atividade anterior.

- **6 - Ajuste no Design do Estudo de Entrevista:** Diante de todas as mudanças geradas nas atividades anteriores, houve a necessidade de ajustes textuais no “Design do Estudo de Entrevista”, para que as alterações feitas fossem contempladas no próximo estudo.
- **7 - Estudo Piloto:** A atividade 7 foi a execução do Piloto, esta etapa teve participação de 5 pessoas e foi importante nessa fase, pois nos permitiu a realização de ajustes, tanto na modelagem quanto nas descrições dos *smells*.
- **8 - Nova Avaliação dos *Smells*:** Por fim, após o refinamento do catálogo, prosseguimos com a avaliação dos *process smells*. Nesta fase 32 participantes puderam avaliar 8 *smells* quanto a sua definição e possíveis impactos ao atributo de qualidade, dando origem a versão final do “Catálogo de Process Smells 2.0”.

6.2 CATÁLOGO DE PROCESS SMELLS 2.0

Nesta etapa foram obtidos 8 *process smells* caracterizados nesta pesquisa, contendo suas definições, elemento estrutural do BPMN impactado, representação visual e possíveis impactos que podem afetar negativamente os atributos de qualidade. Nesta seção se encontra o Catálogo completo dos process smells após todas as etapas de refinamento.

Process Smell: *Message Chains***Elemento impactado:** Tarefa**Descrição:** O *Message Chains* define longas cadeias de trocas de produtos de trabalho entre tarefas que podem representar dependências disfarçadas. Uma tarefa precisa de um produto de trabalho de entrada que para ser gerado precisa esperar por uma cadeia de outras tarefas serem executadas até que este produto de trabalho de entrada esteja pronto.**Possíveis Impactos:** Quanto mais longa for a cadeia de tarefas, maior será a dificuldade na legibilidade do processo, uma vez que aumenta a distância entre a tarefa solicitante e a tarefa que entregará os produtos de trabalho de entrada solicitados, afetando a compreensibilidade do processo.**Representação:****Figura 6.2** *Message Chains*

Process Smell: *Brain Partition*

Elemento Impactado: Partição

Descrição: Brain Partition é um papel que centraliza a maior parte das ações relevantes de um Processo. Possui muitas tarefas e muitos fluxos de decisão para estruturar as tarefas.

Possíveis Impactos: Papéis centralizadores que concentram muitas tarefas e fluxos de decisão se tornam grandes e complexos e podem reduzir a compreensibilidade do usuário do processo.

Representação:

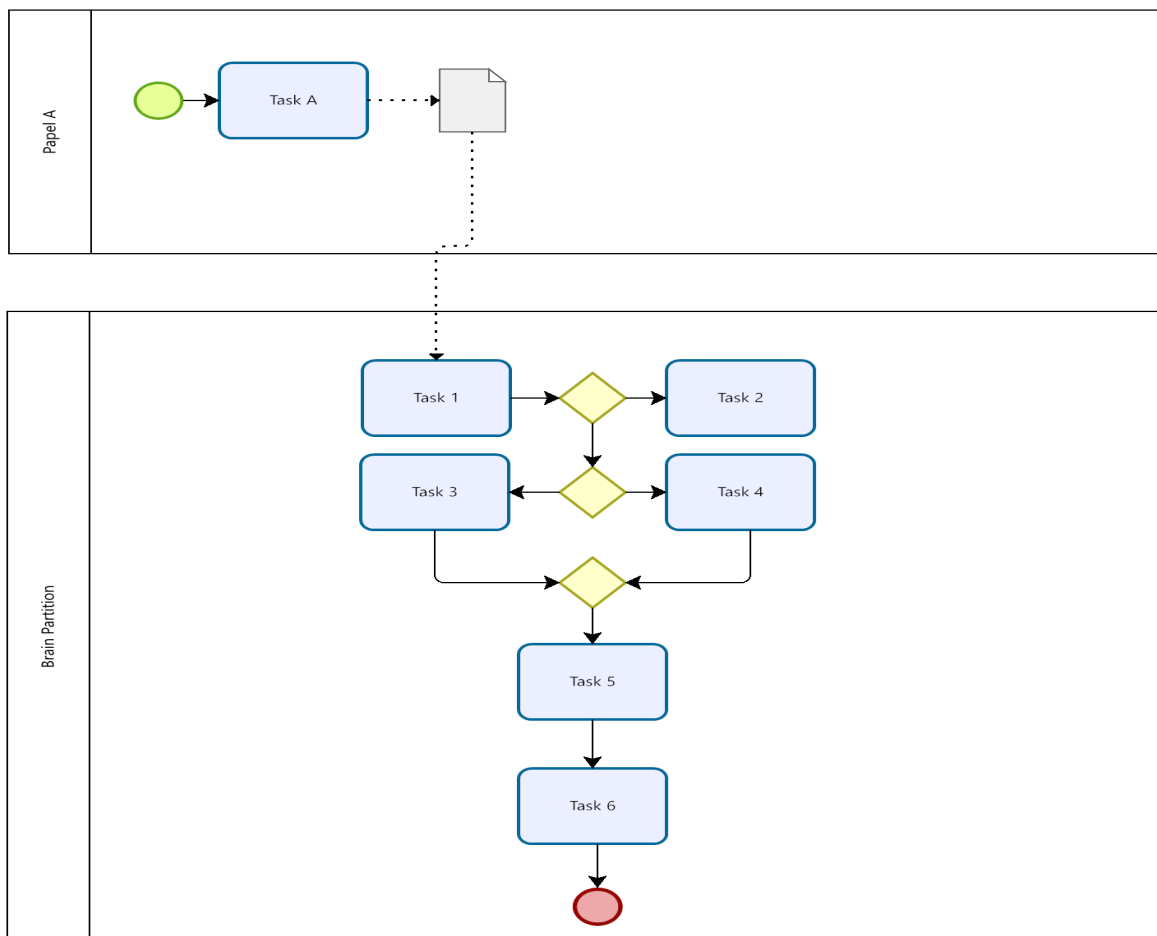
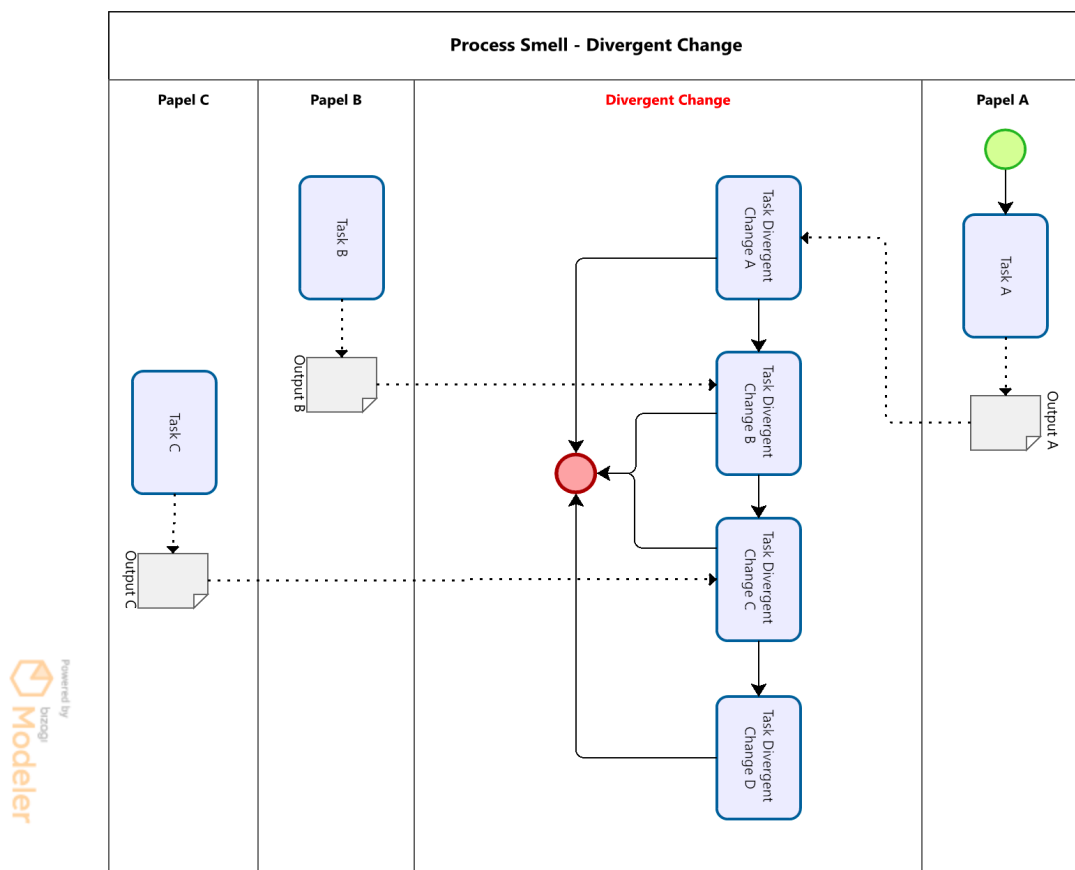


Figura 6.3 *Brain Partition*

Process Smell: *Divergent Change***Elemento Impactado:** Partição

Descrição: Um papel que tem como propósito atender alguma "necessidade" do processo. Um exemplo de "necessidade" em um processo de software pode ser: construir o Termo de Abertura do projeto; validar um documento de especificação; entre outros. Assim, o Divergent Change ocorre quando uma atividade é configurada para atender a não somente uma "necessidade" do processo, tendo assim, diferentes conjuntos de tarefas para cada necessidade a qual atenderá. Em consequência, sempre que qualquer dessas "necessidades", que esse papel atende sofrer mudanças, a parte das tarefas responsáveis por atendê-la também mudará, enquanto as outras tarefas que não fazem parte desse conjunto não sofrerão nenhuma mudança por este mesmo motivo.

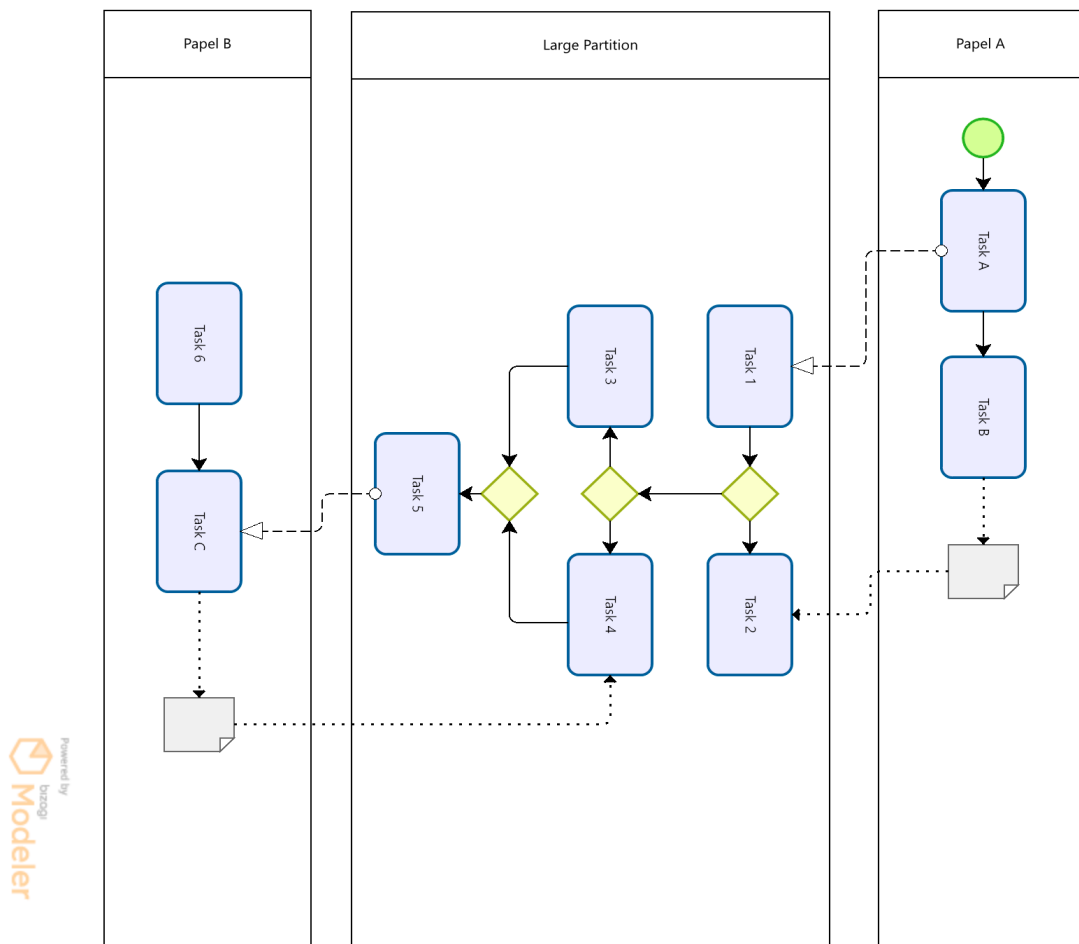
Possíveis Impactos: Esse papel se encarrega de diferentes propósitos, desta forma se torna pouco coeso, ou seja, as tarefas deste papel realizam ações de forma independente das outras tarefas, o que pode prejudicar a compressibilidade do mesmo.

Representação:**Figura 6.4** *Divergent Change*

Process Smell: *Large Partition***Elemento Impactado:** Partição

Descrição: Uma *Large Partition* se caracteriza quando um papel é grande, por conter muitas tarefas e possuir muitas responsabilidades. Ela centraliza as principais ações do Processo, deixando outros papéis com responsabilidades mais triviais e utilizando produtos de trabalho destes papéis para concluir seus objetivos.

Possíveis Impactos: Apesar da sensação de simplificar o processo, a *Large Partition* possui muitas tarefas e muitos fluxos de decisão, assim aumenta a complexidade. A atribuição de muitas tarefas a um único papel gera o risco de confundir a responsabilidade do mesmo, adicionando tarefas que cuidam de diferentes necessidades do processo. Este fato pode também impactar negativamente a coesão entre as tarefas. Essas duas características podem reduzir a compreensibilidade do processo.

Representação:**Figura 6.5** *Large Partition*

Process Smell: Feature Envy

Elemento Impactado: Tarefa

Descrição: O Feature Envy é representado por uma tarefa que faz uso extensivo de produtos de trabalho de saída de outra atividade. Por esse motivo, talvez essa tarefa deve pertencer a outra atividade.

Possíveis Impactos: O Feature Envy é reconhecido pela distância que os produtos de trabalho de entrada têm em relação à tarefa que as irá utilizá-los, que normalmente estão em outra atividade e não na atividade a qual a tarefa pertence. Assim, quanto maior for a medida dessa distância, menor será a compreensibilidade do executante do processo, principalmente para os casos, onde o executante terá que entender a tarefa fornecedora.

Representação:

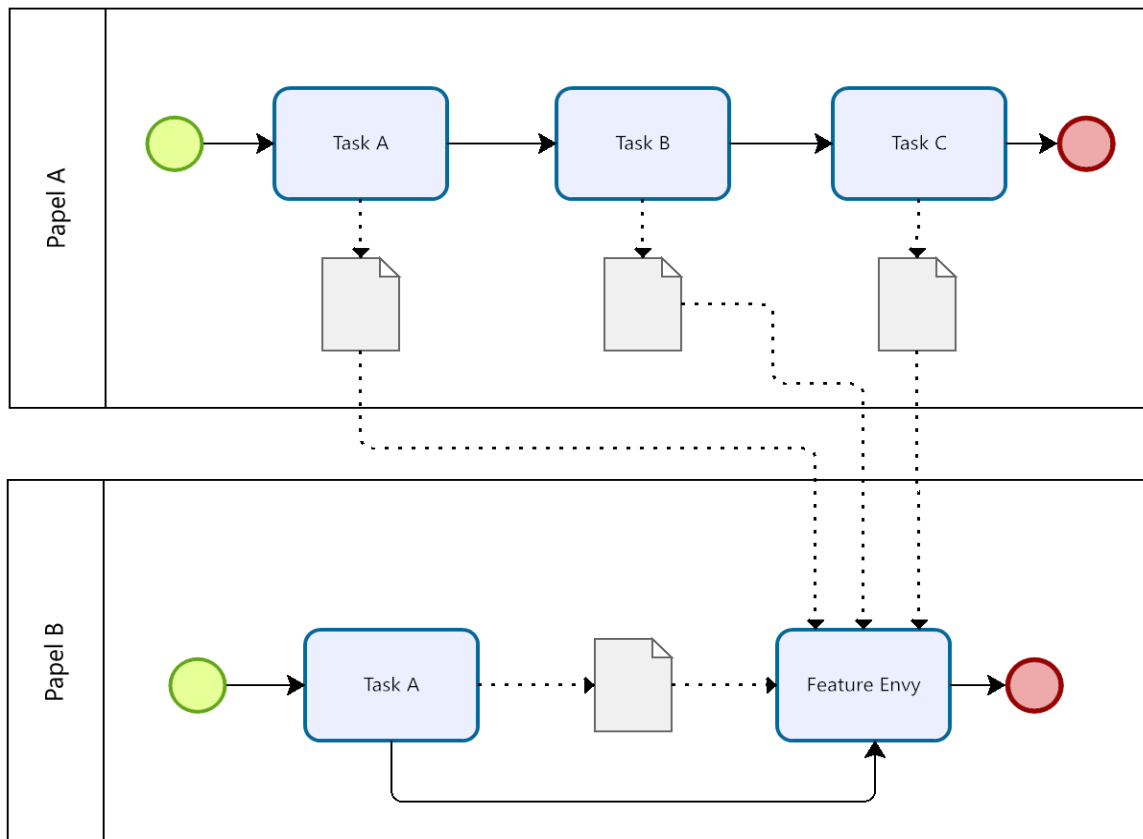


Figura 6.6 *Feature Envy*

Process Smell: *Shotgun Surgery*

Elemento Impactado: Partição

Descrição: O *Shotgun Surgery* se caracteriza por um papel que quando sofre uma mudança (em suas tarefas ou passos) desencadeará a necessidade de muitas pequenas mudanças em vários outros papéis.

Possíveis Impactos: Papéis com *Shotgun Surgery* ocasionam de acoplamento disperso no processo, ou seja, tarefas dependentes que estão espalhadas pelo processo. Assim, mudanças estruturais precisarão de maior esforço na rastreabilidade dos pontos que precisam ser ajustados, de forma a reestruturar o processo de forma consistente. Este fato reduz a capacidade de modificação.

Representação:

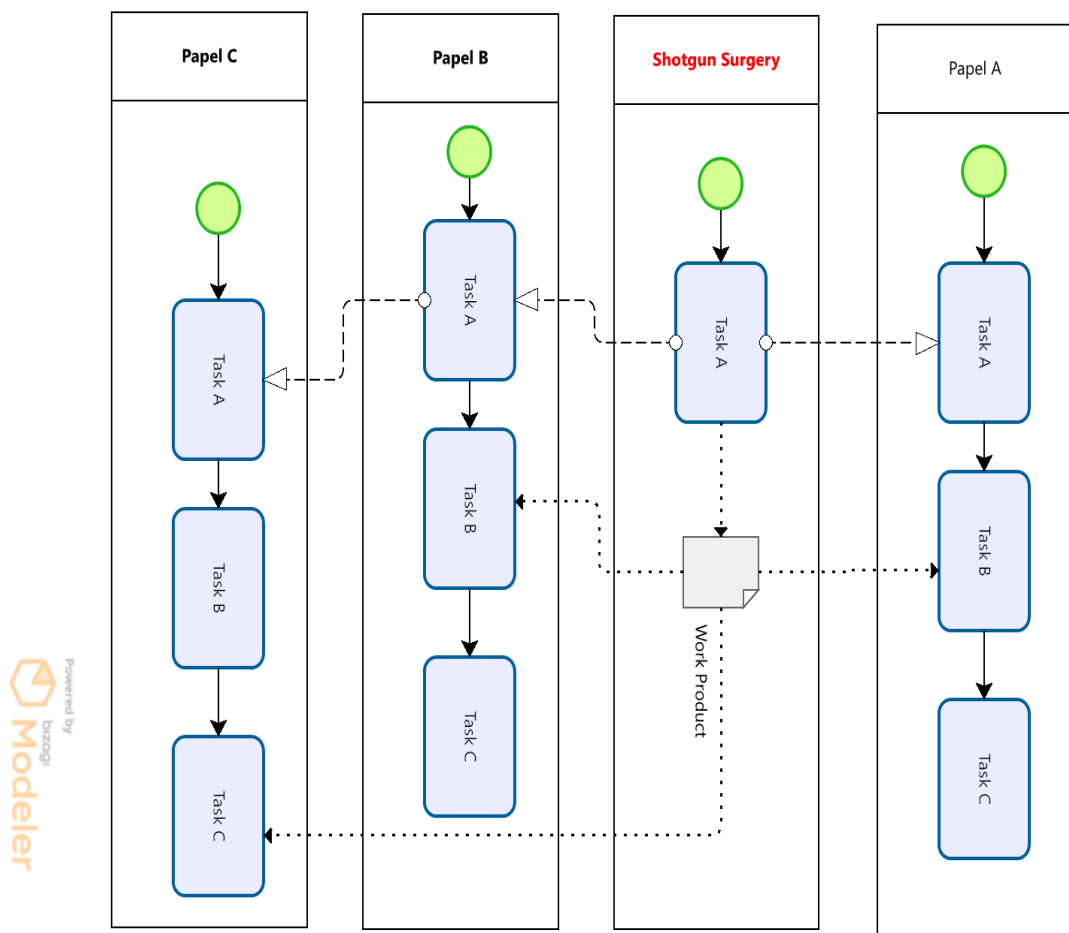
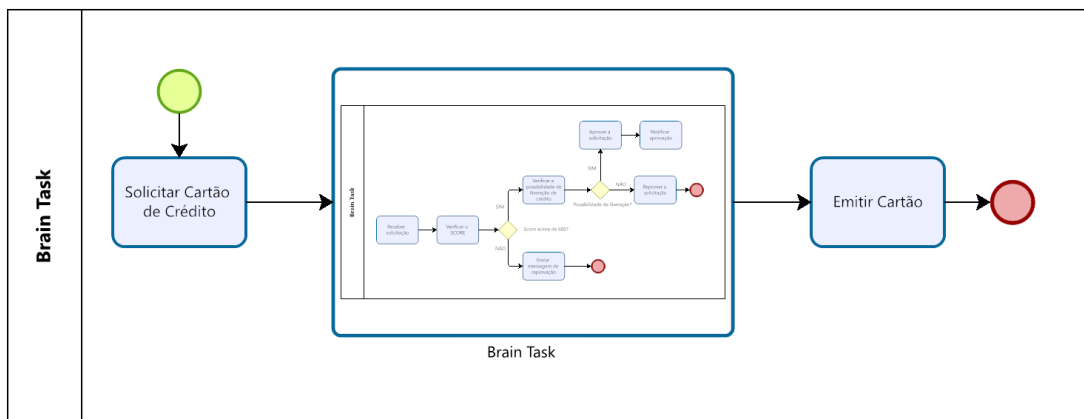


Figura 6.7 *Shotgun Surgery*

Process Smell: *Brain Task***Elemento Impactado:** Tarefa**Descrição:** Uma *Brain Task* é uma tarefa longa que centraliza as ações de um papel, possui muitos passos e fluxos de decisão para estruturar os passos.**Possíveis Impactos:** Apesar da sensação de simplificar o modelo usando poucas tarefas, uma *Brain Task* tem muitos passos e muitos fluxos de decisão, o que aumenta a complexidade e reduz a compreensibilidade da tarefa que sofre esse *process smell*.**Representação:****Figura 6.8** *Brain Task*

Process Smell: *Long Input List*

Elemento Impactado: Tarefa

Descrição: O *Long Input List* é configurado quando uma tarefa precisa de muitos produtos de trabalho de entrada.

Possíveis Impactos: A tarefa que possui uma longa lista de produtos de trabalho de entrada, pode estar sendo impactada por alguns problemas como: acúmulo de responsabilidades ou ser uma tarefa longa por possuir muitos passos e fluxos. E por sua vez pode prejudicar a compreensibilidade do usuário do processo.

Representação:

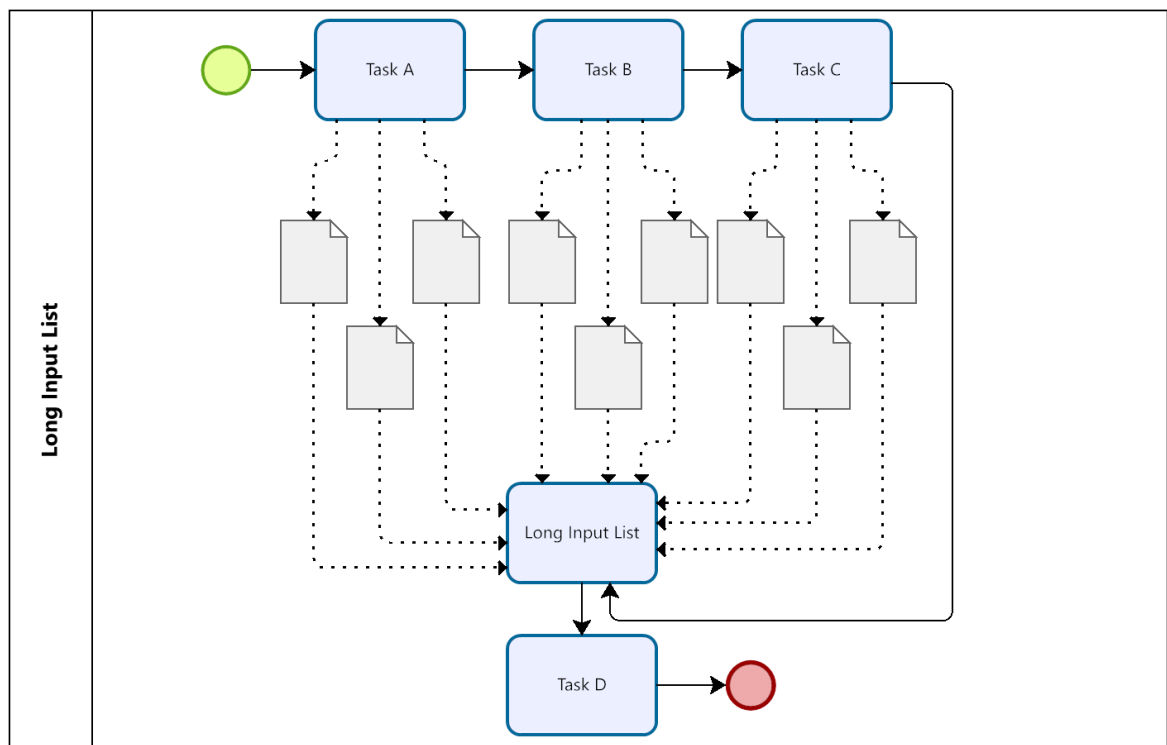


Figura 6.9 *Long Input List*

6.3 RESULTADO DO ESTUDO DE ENTREVISTA 2

Nesta seção serão apresentados os resultados do segundo estudo de entrevista, executado com o objetivo de complementar o primeiro estudo com um número maior de participantes para uma análise mais consistente. Este estudo visou verificar a aplicabilidade dos *process smells* em processos de software por meio da validação dos impactos apontados pelos participantes a respeito de cada *process smells*.

6.3.1 Perfil dos Participantes

Trinta e dois profissionais participaram do estudo de entrevista, entre eles, analistas de testes, product management, desenvolvedores e analistas de processo. Em relação ao tempo de experiência, todos os participantes tinham acima de um ano atuando com processos nas suas respectivas empresas. Sobre o conhecimento dos termos, *bad smells*, *test smells* ou similares a maioria indicou ter conhecimento, embora uma explicação analógica tenha sido dada a respeito dos termos. A seleção dos participantes deu-se pela disponibilidade de participação no estudo.

6.3.2 Análise Quantitativa e Qualitativa dos Resultados

Nesta seção serão discutidos os resultados obtidos durante este estudo. Para os aspectos qualitativos foi definido um método para analisar a aceitação dos participantes com base naquilo que esperávamos obter como resposta. Resultados quantitativos foram representados por meio de um gráfico geral, onde foi possível observar a aceitação de cada *smell* quanto a sua definição, bem como os possíveis impactos aos atributos de qualidade. Um outro ponto a destacar é que nessa segunda fase do estudo os resultados quantitativos serão avaliados conforme o percentual, pois para fins de comparação usar medidas iguais é mais coerente.

A Figura 6.10 representa a aceitação dos smells, onde os participantes puderam avaliá-los numa escala Likert de 5 pontos. O smell mais bem aceito dentre os 8 apresentados no catálogo foi o Long Input List, com 100% de aceitação entre os participantes, ao contrário do Brain Partition, que apesar de ter sido aceito, teve um índice considerável de concordâncias parciais e 6% de discordância. Os Smells, Shotgun Surgery, Divergent Change e Message Chains também foram aceitos, mas ambos receberam índice de rejeição, conforme pode ser visto na Figura 6.10 abaixo.

Já a Figura 6.11 representa a concordância dos participantes no que se diz respeito aos impactos negativos aos atributos de qualidade. Long Input List e Shotgun Surgery foram os Smells mais bem aceitos quando pensamos em impactos a estes atributos. O Message Chains não foi um Smell com um índice tão alto de concordância total, ele aparece no gráfico na última posição obtendo 44% de aceitação total e 38% de aceitação parcial. Na análise qualitativa, trechos justificarão os motivos aos quais os participantes não concordaram totalmente e também as discordâncias para com esse smell. Os demais smells, apesar de terem recebido algum índice

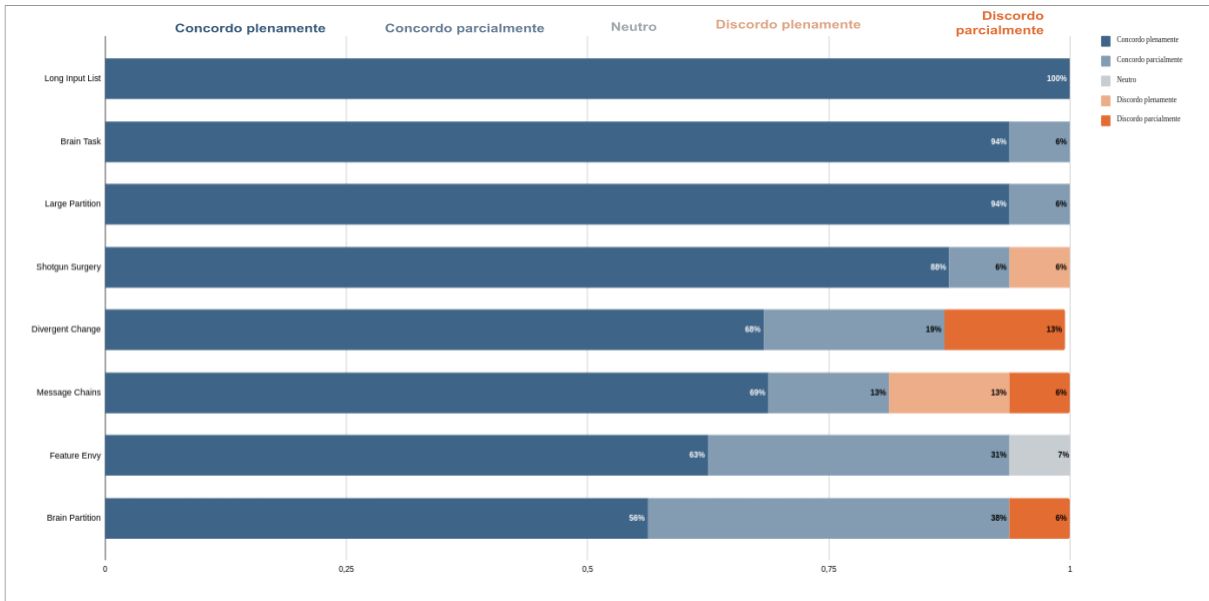


Figura 6.10 Representação da Aceitação dos Process Smells

de discordância, não foram totalmente descartados pelos participantes quanto aos possíveis impactos causados nos atributos de qualidade.

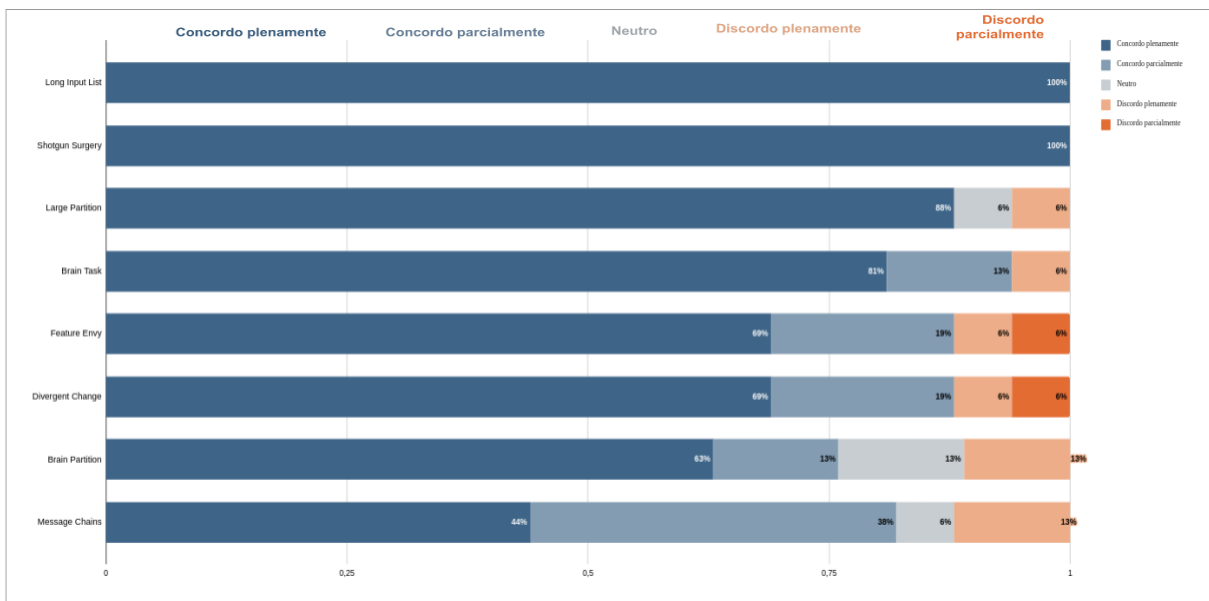


Figura 6.11 Representação dos Impactos negativos aos Sub-atributos de qualidade

6.3.3 Análise qualitativa de cada process smell

Os process smells tiveram suas características e impactos avaliados pelos participantes, onde os aspectos qualitativos foram extraídos a partir do posicionamento do participante ao justificar a aceitação ou não quanto a definição e impactos negativos ao atributo de qualidade por ele observado. Para os casos de aceitação é possível que, através das respostas as características possam ser confirmadas ou que propostas de características diferentes também sejam adicionadas. As discordâncias também foram justificadas pelos participantes e destacadas para os smells que as receberam.

A seguir os resultados das análises dos *process smells* serão apresentados.

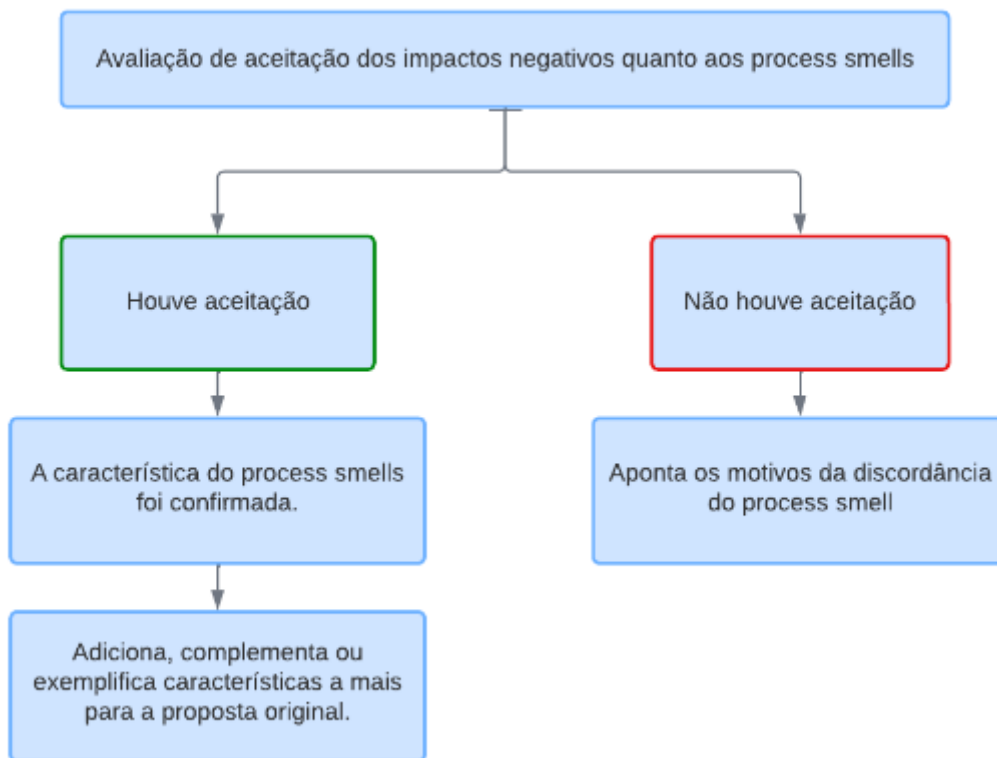


Figura 6.12 Avaliação de aceitação dos impactos negativos quanto aos process smells

6.3.3.1 LONG INPUT LIST

ASPECTOS QUALITATIVOS:

- (a) **Respostas onde, “A característica do process smell foi confirmada.”**
 Conforme mencionado anteriormente, o Long Input List foi um dos smells mais bem aceitos na avaliação e os trechos a seguir apresentam algumas das

opiniões em relação às características deste smell com as seguintes frases: “No caso deste smell, a compreensibilidade do processo fica comprometida, pois a tarefa possui vários produtos de entrada.”, “É um smell de difícil compreensão, porque a quantidade de produtos de entrada o torna complexo para o meu entendimento.”, “É um modelo que não me traz clareza do que está sendo feito ou deveria ser feito, acredito que a quantidade de produto de trabalho o prejudica neste sentido”.

A concordância dos participantes com a redução da compreensibilidade do processo causada pelo Long Input List se associa ao fato de existirem inúmeras informações a serem analisadas ao executar a tarefa. Os participantes evidenciaram a quantidade de produto de trabalho de entrada das tarefas, que é a principal característica de impacto negativo na compreensão do Long Input List.

(b) **Respostas que, “Adiciona, complementa ou exemplifica características a mais para a proposta original.”**

Ao justificar a sua resposta em relação ao Long Input List um dos participantes também levou em consideração a possibilidade de modificação do processo, bem como em algum momento o processo “precisar” de uma tarefa que concentre muitas responsabilidades.

6.3.3.2 SHOTGUN SURGERY

ASPECTOS QUALITATIVOS:

(a) **Respostas onde, “A característica do process smell foi confirmada.”**

A concordância dos participantes quanto à característica do Shotgun Surgery foi destacada pelos trechos a seguir: “Ao mudar um papel de forma que isso provoque mudanças em muitos outros papéis, o entendimento do processo como um todo pode ser dificultoso, o que pode aumentar a complexidade e reduzir a flexibilidade de uma possível adaptação do processo”. Um outro trecho destaca concordância, “Esse tipo de problema faz com que o processo tenha dificuldade em se adequar a mudanças nas demandas e nas necessidades dos clientes, atrasando o desenvolvimento e aumentando os custos. O processo neste caso, poderia ser simplificado, deixando as responsabilidades separadas e tentar reduzir o acoplamento entre os papéis”. Apesar de dois participantes terem concordado parcialmente, suas justificativas ao final levaram ao entendimento de que eles mais concordavam do que discordavam das características deste smell.

(b) **Respostas que, “Adiciona, complementa ou exemplifica características a mais para a proposta original.”**

Alguns participantes evidenciaram outros aspectos que podem complementar a nossa proposta em relação aos impactos que podem ser sofridos no processo,

como pode ser visto no seguinte trecho: "Esse exemplo de process smell pode ser um indício de que o processo pode ter problemas de acoplamento excessivo, o que pode levar a problemas de manutenção, escalabilidade e qualidade, além de torná-lo complexo".

(c) **Respostas que, "Apontam o motivo da discordância do process smell."**

Dois participantes discordaram das características do Shotgun Surgery, levando em consideração os pontos que foram destacados nos trechos a seguir: "Acredito que muitos processos precisam ou precisarão em algum momento sofrer algum tipo de modificação, dessa forma não há como isentar que outras partes do processo não sofra modificações também, isto é, pensando em melhorias", um outro exemplo de discordância foi pontuado pelo seguinte participante ao dizer que "No quesito desenvolvimento de software, eu discordo porque módulos sempre que precisam são modificados, um exemplo, sistemas legados passam por mudanças que, querendo ou não levam pequenas mudanças a outros módulos e isso de alguma forma é necessário para o cliente ou negócio".

6.3.3.3 LARGE PARTITION

ASPECTOS QUALITATIVOS:

(a) **Respostas onde, "A característica do process smell foi confirmada."**

Conforme mencionado nos aspectos quantitativos, grande parte dos participantes concordaram com as características deste process smell. Com base nas respostas obtidas durante as entrevistas, os trechos a seguir evidenciam as opiniões de alguns participantes. Um dos entrevistados concordou ao justificar sua resposta dizendo: "Analisando a imagem e a sua definição, imagino que os membros de um time podem ter dificuldade para colaborar dentro do processo, já que não fica claro quem é o responsável pelas tarefas e os produtos de entrada". Um outro participante justificou com a seguinte resposta: "Quando se tem um papel assumindo várias responsabilidades isso tende a ser problemático, pois pode levar a uma complexidade desnecessária e dificultar a compreensão do processo". um outro entrevistado complementa da seguinte forma, "A presença deste smell torna o processo menos eficiente e dificulta a evolução e manutenção de um sistema por exemplo".

(b) **Respostas que, "Adiciona, complementa ou exemplifica características a mais para a proposta original."**

O trecho a seguir destaca uma possibilidade de melhoria na eficiência do processo bem como ao atributo de compreensibilidade. "Acredito que neste caso podemos buscar formas de dividir os papéis maiores e complexos em papéis menores e mais especializados, com responsabilidades claras e bem definidas, melhorando assim a compreensibilidade e eficiência do processo".

(c) **Respostas que, “Apontam o motivo da discordância do process smell.”**

Em relação ao atributo de qualidade, duas pessoas discordaram quanto ao impacto a compreensão, para estes participantes ” Alguns papéis em determinado momento vão precisar assumir mais responsabilidades que outros e levando em consideração todo o alinhamento durante o projeto, as partes envolvidas estarão cientes do que cada um estará fazendo dentro do processo, neste caso acredito que a compreensão não seria um problema, desde que tudo esteja alinhado e documentado”.

6.3.3.4 BRAIN TASK**ASPECTOS QUALITATIVOS:**(a) **Respostas onde, “A característica do process smell foi confirmada.”**

Em relação ao Brain Task, os trechos a seguir apresentam as justificativas de concordância com a característica deste smell. Um dos entrevistados concordou com a definição proposta ao dizer: “Concordo que uma tarefa com muitos passos e muitos fluxos pode ser um process smell em um processo de software. Essa complexidade pode tornar a tarefa difícil de ser executada, gerenciada e compreendida, o que pode afetar negativamente a eficiência, qualidade e colaboração da equipe.” um outro participante concordou respondendo: “Sim, uma tarefa com muitos passos e fluxos pode ser um process smell, já que essa complexidade pode tornar a tarefa difícil de ser gerenciada e executada com eficiência, aumentando a probabilidade de erros e atrasos, além de dificultar a compreensão do processo.”

(b) **Respostas que, “Adiciona, complementa ou exemplifica características a mais para a proposta original.”**

Participantes levaram em consideração alguns outros impactos que poderiam ser sofridos no processo além do atributo de compreensibilidade. “...Essa complexidade pode levar a problemas de gerenciamento, execução, risco de erros e atrasos, além de prejudicar a colaboração da equipe.”

(c) **Respostas que, “Apontam o motivo da discordância do process smell.”**

Dois dos participantes discordaram com o impacto negativo a compreensibilidade do processo, isso porque na opinião deles o processo pode ser compreendido de forma clara e objetiva. Os trechos seguintes evidenciam tais posicionamentos: ”Concordo com a definição e possíveis impactos que um smell desse tipo pode causar dentro do processo, mas neste caso em específico acredito que a compreensibilidade não foi impactada. Talvez uma outra representação mude a minha opinião, mas pra este exemplo discordo.” , “Não acho que a compreensão seja impactada, levando em consideração a imagem, vejo que as

tarefas neste caso são bem definidas para realização das suas funções no processo”. Para estes dois casos, o modelo que representa o Brain Task pode ter sido um fato de influencia para a discordância ao atributo de qualidade compreensibilidade, tendo em vista que as duas justificativas apontam o modelo como compreensível.

6.3.3.5 FEATURE ENVY

ASPECTOS QUALITATIVOS:

(a) **Respostas onde, “A característica do process smell foi confirmada.”**

Em relação a concordância os participantes evidenciaram características deste smell ao justificarem conforme os trechos: “A distância entre os produtos de trabalho de entrada e a tarefa que irá utilizá-los pode ser um fator crítico para a compreensão e a execução da tarefa” , “...é importante minimizar essa distância para melhorar a compreensibilidade e a execução da tarefa” e “Quando os produtos de trabalho de entrada estão muito distantes da tarefa que irá utilizá-los, pode haver um problema que pode dificultar a compreensão e a execução da tarefa”. Os trechos acima fazem referência ao Feature Envy com base nos seus possíveis impactos propostos no catálogo. Em resumo, este process smell faz uso dos dados de outras tarefas em outro papel, assim os produtos de entrada vem de um outro papel o que proporciona a distância do destino e origem. Leva-se em consideração que quanto mais papeis dentro do processo, mais complexo se torna a compreensão do mesmo.

(b) **Respostas que, “Apontam o motivo da discordância do process smell.”**

Levando em consideração a definição do Feature Envy, alguns participantes discordaram. Nos trechos a seguir foram evidenciados alguns dos pontos que levaram a discordância dos participantes para com a definição deste smell. “Não necessariamente, tarefas que fazem uso extensivo de produto de trabalho de saída de outro papel no processo é comum em muitos fluxos de trabalho, então acredito que para considerá-lo como um process smell é preciso que sejam apresentadas outras características que indicassem um problema, como retrabalho, baixa qualidade do produto final, etc. , ”Discordo, porque é possível que o uso extensivo de produtos de trabalho de saída de outro papel seja um process smell em alguns casos, mas não em outros. É necessário analisar cada caso individualmente e considerar todos os fatores relevantes”.

6.3.3.6 BRAIN PARTITION

ASPECTOS QUALITATIVOS:

(a) **Respostas onde, “A característica do process smell foi confirmada.”**

Os trechos a seguir atestam a confirmação de aceitação por parte dos participantes em relação ao Brain Partition. “A compreensão do usuário do processo pode ser impactada devido a concentração de muitas tarefas e fluxos de decisão, tornando o processo grande e complexo.” “Processos onde os papéis possuem acúmulo de muitas responsabilidades tendem aumentar o tempo necessário para concluir as tarefas além de diminuir a qualidade do trabalho realizado e a sua compreensão.” Em ambos os comentários os destaques indicados foram quanto ao acúmulo de responsabilidades concentradas em um único papel, bem como o impacto gerado a compreensão do processo. Estas são duas das características principais do Brain Partition descritas no catálogo.

(b) **Respostas que, “Apontam o motivo da discordância do process smell.”**

Em relação as respostas onde os participantes discordaram da definição ou impactos causados ao atributo de qualidade, a seguinte justificativa consegue de modo geral resumir as demais colocações dos participantes que optaram por não concordar que o Brain Partion é um process smell. “A presença de muitas tarefas e fluxos de decisão dentro de um papel não deve ser considerada automaticamente como um process smell. A complexidade pode ser justificada pelo contexto do processo, desde que o processo seja eficiente e eficaz. Porém, se a complexidade levar a problemas no processo, como atrasos ou baixa qualidade do produto final, é necessário uma investigação mais aprofundada para determinar se há um process smell e como solucioná-lo”. Em resumo, para os participantes que discordaram a complexidade de um papel no processo só é um problema se for desnecessária ou causar problemas no desempenho do processo e em caso positivo é proposto uma investigação aprofundada e uma possível solução.

6.3.3.7 MESSAGE CHAINS**ASPECTOS QUALITATIVOS:**(a) **Respostas onde, “A característica do process smell foi confirmada.”**

O Message Chains foi aceito por 26 participantes no que se diz respeito a sua definição, bem como ao seu atributo de qualidade compreensibilidade. Os comentários a seguir atestam a opinião dos participantes: “Essas cadeias longas de dependências podem aumentar consideravelmente o tempo de espera para a conclusão da tarefa e afetar a eficiência do processo como um todo.” , “Muitas vezes, essas cadeias de dependências são criadas sem que se perceba, e isso pode causar um grande impacto na produtividade do time e no prazo de entrega do projeto.” , “...a definição de Message Chains é muito válida, pois dependências podem gerar gargalos significativos no processo de software.”. De modo geral os participantes entendem que identificar e mitigar as dependências

disfarçadas entre as tarefas é importante para melhorar a eficiência do processo e garantir que o produto final tenha qualidade.

(b) **Respostas que, “Apontam o motivo da discordância do process smell.”**

Sobre a definição e impacto ao atributo de qualidade compreensibilidade houveram discordâncias, bem como participantes que optaram por ficarem neutros. Justificando a discordância os participantes alegaram: “...vão existir processos onde não terá como fugir de dependências, isso pode ser analisado no contexto de desenvolvimento, onde várias dependências são criadas no início e que só serão de fato usadas por uma tarefa solicitante no final.”, “Quando se tem muitas tarefas mesmo que dependentes umas das outras, se dentro do processo o papel de cada uma delas estiver bem definido, acredito que a compreensão não será impactada.”, “Discordo, pois não vejo este smell como uma grande ameaça dentro do processo, pensando no cenário real muitas vezes essas dependências são criadas entre as tarefas mas por necessidade.” As justificativas de discordância quanto a definição e impacto ao atributo de qualidade apontam que para alguns profissionais, mesmo não sendo uma boa prática e que traga problemas futuros o *Message Chains* é “comum” em alguns aspectos do processo de software.

6.4 AMEAÇAS À VALIDADE

Como ameaças a validade deste estudo pode-se destacar que os participantes do estudo atual são diferentes do primeiro, neste sentido as mudanças de nomenclatura e da estrutura do modelo não são pontos de conhecimento para esses novos participantes. Os termos *bad smells* e *process smells* apesar de terem sido apresentados no início da entrevista, também foram questionados durante a execução do estudo o que não veio a inviabilizá-lo, uma vez que, sempre que o participante sentia a necessidade de lembrar o conceito isso era repassado e exemplificado da forma mais clara possível. Em relação a proposição da existência de *process smells* em processo de software, as respostas obtidas ficaram entre relevante, muito relevante e muitíssimo relevante. Apesar deste estudo ter sido realizado com dois smells a menos, o tempo decorrido para a realização da entrevista foi pontuado como exaustivo em relação a entrevista e a extensão do catálogo, onde o tempo médio foi de 45 minutos. Ao final, os participantes avaliaram positivamente a clareza do conteúdo, onde as respostas ficaram entre suficiente, adequado e muito adequado para o entendimento e avaliação dos *process smells 2.0* propostos. Todos os participantes concordaram com a proposta de refatoração de processos e a sua viabilidade.

6.5 DISCUSSÃO

Nesta seção serão discutidos os resultados obtidos neste estudo complementar, levando em consideração o primeiro estudo realizado com um número menor de pro-

fissionais, bem como isso reflete em comparação ao trabalho de Santos, Maciel e Sant'Anna (2018) e os resultados obtido por eles. A intenção é que esta comparação seja feita de modo a avaliar os pontos positivos de um estudo e outro, visando destacar as suas particularidades.

Ao concluir o estudo, um dos pontos que podemos destacar é o quão significativo foi complementá-lo, uma vez que, no primeiro momento o catálogo foi avaliado por apenas sete profissionais e apesar dos resultados obtidos terem sido satisfatórios, tínhamos como objetivo obter mais percepções e ampliar a avaliação dos process smells.

A tabela 6.1 representa uma visão geral dos resultados obtidos nos três estudos, sendo que na primeira fase de avaliação 10 process smells foram propostos, tanto no trabalho atual, quanto no trabalho de Santos, Maciel e Sant'Anna (2018). A fase dois deste estudo consistiu em avaliar somente 8 process smells, visto que, ao repensar a modelagem deles os elementos estruturais do BPMN não faziam sentido para readequá-los, neste caso os smells *Work Product Clumps* e *Data Activity* não foram avaliados. Apesar desses smells não terem sido modelados e avaliados nesta segunda fase, em relação ao primeiro estudo o *Data Activity* foi o smell de menor aceitação entre os participantes, obtendo apenas 43% de aceitação, enquanto no estudo da modelagem em SPEM esse mesmo smell obteve uma taxa de aceitação de 83%. Essa diferença pode indicar que no modelo SPEM o *Data Activity* pode ser mais perceptível ou melhor representável.

Large Activity é um Smell que pode ser destacado pelos seus resultados nos estudos, pois a sua aceitação ao ser modelado em SPEM foi de 92%, enquanto no primeiro estudo em BPMN obteve apenas 71%. Ao remodelar este smell e assumir que seu impacto estava relacionado ao papel e não a uma atividade dentro do processo, no segundo estudo o *Large Partition* que antes possuía o nome de "*Large Activity*", obteve 100% de aceitação entre os participantes. Em comparação ao resultado obtido por Santos, Maciel e Sant'Anna (2018), percebe-se que no SPEM o impacto a atividade faz mais sentido no processo e por isso pode ter sido melhor aceito. Enquanto que em BPMN ao remodelar e apontar o papel como o maior impactado dentro do processo isso pode ter feito mais sentido para os participantes ao avaliarem este smell.

Em contrapartida ao analisar os resultados das avaliações do "*Brain Activity*", agora nomeado de "*Brain Partition*" é possível notar que este smell não foi tão aceito no SPEM, se comparado aos demais Smells deste mesmo modelo. O *Brain Activity* obteve apenas 58% de aceitação, onde os participantes não concordaram com os possíveis impactos ou que isso poderia ser um problema dentro do processo. Já no segundo estudo o *Brain Partition* teve uma taxa de aceitação de 94% o que é 8% a mais que o primeiro estudo e pode ser considerado um smell melhor representado no BPMN ou que os impactos no processo deste modelo podem ser mais evidentes.

A análise dos resultados em relação ao *Message Chains* é um tanto curiosa, uma vez que este smell obteve a menor taxa de aceitação na segunda fase deste estudo e

é o segundo smell menos aceito no trabalho em SPEM, ficando com 82% e 67% de aceitação respectivamente. Em comparação ao primeiro estudo, esta segunda fase se mostrou convergente ao percentual de aceitação do estudo realizado por Santos, Maciel e Sant'Anna (2018).

Os demais smells não apresentaram muitas divergências de aceitação entre um estudo e outro. Apesar de alguns deles receberem uma avaliação menor e parecerem menos significativos que outros, nenhum participante discordou com a proposta do estudo. O SPEM possui as suas particularidades desde a sua forma estrutural e como um impacto e outro pode ser percebido ou não dentro de um processo. O mesmo acontece para o BPMN, alguns impactos podem ser melhores percebidos através deste modelo, assim como alguns modelos são melhores representados pelo SPEM. De modo geral as duas notações conseguiram representar os catálogos de process smells em ambos os estudos e obtiveram aceitação e incentivo dos participantes em relação a proposta apresentada.

Tabela 6.1 Aceitação geral entre os estudos

Visão Geral de Aceitação: Resultados (%)			
Process Smells	BPMN (Fase 1)	BPMN (Fase 2)	SPEM
Brain Activity	86%	-	58%
Brain Partion	-	94%	-
Brain Task	86%	100%	83%
Data Activity	43%	-	83%
Divergent Change	86%	87%	92%
Feature Envy	100%	94%	83%
Large Activity	71%	-	92%
Large Partition	-	100%	-
Long Input List	86%	100%	83%
Message Chains	86%	82%	67%
Shotgun Surgery	86%	94%	100%
Work Product Clumps	71%	-	83%

CONSIDERAÇÕES FINAIS

A partir dessa pesquisa conclui-se que os *bad smells* podem ser aplicáveis a processo de software e i identificá-los de forma independente da linguagem de especificação. A partir disso também é possível aplicá-los ao processo de negócio. Os resultados obtidos neste estudo mostraram que alguns *process smells* são mais significativos que outros, evidenciando o cenário comum no contexto da engenharia de software. Este fato ocorre em situações como a escolha do método, a prioridade dos requisitos, a escolha dos padrões do projeto, entre outros, a depender do contexto da discussão.

No contexto de refatoração de processos e a sua viabilidade, os participantes demonstraram total concordância para tal, o que aponta uma possível melhoria nos processos e com isso a possibilidade da diminuição dos impactos causados aos elementos estruturais dos modelos. Alguns dos participantes pontuaram que a refatoração em processos já é uma prática comum e que deve ser disseminada.

A elaboração de um novo catálogo não tem o intuito de descartar a versão proposta por Santos, Maciel e Sant'Anna (2018), mas sim instigar investigações a respeito da possibilidade de identificar process smells independente da linguagem/notação, o que já é aplicado em outros contextos de smells, como por exemplo bad smells, que são identificados de forma independente da linguagem no qual foram desenvolvidos. Além disso, com a elaboração do Catálogo de Process Smells 2.0 a proposta pode ser avaliada com mais profissionais o que permitiu a captação de novas nuances e percepções a respeito do estudo.

Por fim, os resultados obtidos durante este estudo abrem caminhos para ampliar o suporte aos profissionais que especificam processos, bem como permite a elaboração de novos process smells seja, tendo como base os modelos atuais e as heurísticas que os identificam. Este catálogo também poderá ser usado como referência para que as más práticas durante a especificação passem a ser evitadas e que os profissionais passem a compreender e modificar com o facilidade os processos.

7.1 TRABALHOS FUTUROS

Como trabalhos futuros percebidos durante a realização desta pesquisa podem ser destacados: a validação do catálogo de modo a identificar o grau de apoio que os especialistas teriam em encontrar problemas estruturais em processos; verificar se outros elementos do BPMN podem ser avaliados, como por exemplo, quais elementos melhor representaria os smells ausentes nessa versão final do catálogo, sendo eles *Data Activity* e *Work Product Clumps*. Investigar sobre a proposta de identificação de process smells independente de linguagem, bem como o suporte que o catálogo forneceria na realização da refatoração desses tipos de process smells dentro do processo.

Os possíveis trabalhos futuros mencionados aqui não inviabilizam a concepção de novos trabalhos futuros a partir deste estudo, no entanto, foram estes os pontos mais percebidos durante a execução da pesquisa.

REFERÊNCIAS BIBLIOGRÁFICAS

- ANDERL, R.; RASSLER, J. Pml, an object oriented process modeling language. In: *Computer-Aided Innovation (CAI)*. [S.l.]: Springer, 2008. p. 145–156.
- CAMPOS, A. Modelagem de processos: Notações. <https://www.tiespecialistas.com.br/modelagem-de-processos-notacoes/comments>, 2013.
- CAMPOS, A. L. N.; OLIVEIRA, T. Software processes with bpmn: An empirical analysis. In: HEIDRICH, J. et al. (Ed.). *Product-Focused Software Process Improvement*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013. p. 338–341. ISBN 978-3-642-39259-7.
- CAMPOS, R.; LIMA, S. Mapeamento de processos: importância para as organizações. *Rio de Janeiro: Universidade Federal Rural do Rio de Janeiro*, 2012.
- CORRADINI, F. et al. A guidelines framework for understandable bpmn models. *Data & Knowledge Engineering*, Elsevier, v. 113, p. 129–154, 2018.
- DEURSEN, A. V. et al. Refactoring test code. In: CITESEER. *Proceedings of the 2nd international conference on extreme programming and flexible processes in software engineering (XP2001)*. [S.l.], 2001. p. 92–95.
- DOĞAN, E.; TÜZÜN, E. Towards a taxonomy of code review smells. *Information and Software Technology*, Elsevier, v. 142, p. 106737, 2022.
- DUMAS, M.; PFAHL, D. Modeling software processes using bpmn: When and when not? In: _____. *Managing Software Process Evolution: Traditional, Agile and Beyond – How to Handle Process Change*. Cham: Springer International Publishing, 2016. p. 165–183. ISBN 978-3-319-31545-4. Disponível em: https://doi.org/10.1007/978-3-319-31545-4_9.
- FOWLER, M. Refactoring: improving the design of existing code. 1999. *Google Scholar Google Scholar Digital Library Digital Library*, 1999.
- GARCÍA-BORGOÑÓN, L. et al. Software process modeling languages: A systematic literature review. *Information and Software Technology*, Elsevier, v. 56, n. 2, p. 103–116, 2014.
- GARCÍA, F. et al. Integrated measurement for the evaluation and improvement of software processes. In: SPRINGER. *European Workshop on Software Process Technology*. [S.l.], 2003. p. 94–111.
- GENVIGIR, E. Filho, fl; sant’anna, n. *Modelagem de processos de software através do spem-software process engineering metamodel-conceitos e aplicação. III WOR-CAP, São José dos Campos, SP*, 2003.

- GREILER, M.; DEURSEN, A. V.; STOREY, M.-A. Automated detection of test fixture strategies and smells. In: IEEE. *2013 IEEE Sixth International Conference on Software Testing, Verification and Validation*. [S.l.], 2013. p. 322–331.
- GRUBEL, D. Process modeling language specification. 1995.
- JÚNIOR, Á. F. de B.; JÚNIOR, N. F. A utilização da técnica da entrevista em trabalhos científicos. *Revista Evidência*, v. 7, n. 7, 2012.
- KHLIF, W. et al. Quality metrics for business process modeling. In: *Proceedings of the 9th WSEAS international conference on Applied computer science*. [S.l.: s.n.], 2009. v. 9, n. 1, p. 195–200.
- KROEGER, T. A.; DAVIDSON, N. J.; COOK, S. C. Understanding the characteristics of quality for software engineering processes: A grounded theory investigation. *Information and Software Technology*, Elsevier, v. 56, n. 2, p. 252–271, 2014.
- LAKATOS, E. M.; MARCONI, M. d. A. Técnicas de pesquisa. 5ª edição. São Paulo, Atlas, 2002.
- MACÍA, I.; SANT'ANNA, C.; STAA, A. v. Detectando problemas de design em diagramas de classes: Um estudo experimental. In: *V Experimental Software Engineering Latin American Workshop (ESELAW'2008)*. [S.l.: s.n.], 2008.
- MALHOTRA, N. K. *Pesquisa de Marketing-: Uma Orientação Aplicada*. [S.l.]: Bookman Editora, 2001.
- MANTYLA, M.; VANHANEN, J.; LASSENIUS, C. A taxonomy and an initial empirical study of bad smells in code. In: IEEE. *International Conference on Software Maintenance, 2003. ICSM 2003. Proceedings*. [S.l.], 2003. p. 381–384.
- MARINESCU, R. Detection strategies: Metrics-based rules for detecting design flaws. In: IEEE. *20th IEEE International Conference on Software Maintenance, 2004. Proceedings*. [S.l.], 2004. p. 350–359.
- MARTICORENA, R.; LÓPEZ, C.; CRESPO, Y. Extending a taxonomy of bad code smells with metrics. In: CITESEER. *Proceedings of 7th International Workshop on Object-Oriented Reengineering (WOOR)*. [S.l.], 2006. p. 6.
- OLIVEIRA, L. H. d. Exemplo de cálculo de ranking médio para likert. *Notas de Aula. Metodologia Científica e Técnicas de Pesquisa em Administração. Mestrado em Adm. e Desenvolvimento Organizacional. PPGA CNEC/FACECA: Varginha*, 2005.
- OMG. Business process model and notation. <https://www.omg.org/spec/BPMN/2.0/>, 2010.
- OSTERWEIL, L. *Software processes are software too*. In *ICSE'87: Proc. of the 9th international conference on Software Engineering. Los Alamitos*. [S.l.]: CA, USA. IEEE Computer Society Press, 1987.
- RAMOS, M.; MELLO, R. de; FONSECA, B. *On Transfer Learning in Code Smells Detection*. [S.l.], 2022.

SANTOS, E. J.; MACIEL, R. S. P.; SANT'ANNA, C. A catalogue of bad smells for software process. In: *Proceedings of the 17th Brazilian Symposium on Software Quality*. [S.l.: s.n.], 2018. p. 1–10.

SEAMAN, C. Qualitative methods in empirical studies of software engineering. *IEEE Transactions on Software Engineering*, v. 25, n. 4, p. 557–572, 1999.

SILINGAS, D.; MILEVICIENE, E. Refactoring bpmn models: From ‘bad smells’ to best practices and. *BPMN 2.0 Handbook Second Edition: Methods, Concepts, Case Studies and Standards in Business Process Management Notation*, Future Strategies Inc., p. 125, 2011.

SOMMERVILLE, I. Software engineering 9th edition. *ISBN-10*, v. 137035152, p. 18, 2011.

WAHL, T.; SINDRE, G. An analytical evaluation of bpmn using a semiotic quality framework. In: *Advanced Topics in Database Research, Volume 5*. [S.l.]: IGI Global, 2006. p. 94–105.

Apêndice

A

ESTUDO DE ENTREVISTA

Nesta seção é apresentado o catálogo completo dos *process smells*, onde são apresentados os elementos impactados, a descrição do *process smell*, os possíveis impactos e a representação de cada *process smell*.

***Process Smells* - Entrevista 1** **Percepções Profissionais**

Introdução

Nesta seção serão apresentadas algumas definições e orientações para apoiar sua participação neste estudo de entrevista. Este trabalho tem como perspectiva avaliar situações sobre processo de desenvolvimento de software e o modelo que o representa, que correspondam ao processo utilizado no dia-a-dia organizacional. Não tem como propósito avaliar as estruturas de "modelos de processos gerais, ou paradigmas de processos" como waterfall (modelo cascata), modelo espiral, modelos iterativos e incrementais, modelos ágeis e entre outros.

Definição de *Process Smell*

Process smell são resultados das decisões tomadas durante as fases como: concepção, modelagem e a produção do modelo do processo que incidem resultados sobre elementos do processo em particular que venham a reduzir a qualidade de fatores esperados para um processo. Os *process smells* podem incidir na redução da qualidade de diferentes fatores inseridos no processo, propriamente dito, ou que são associados ao processo. Entretanto para este trabalho serão verificados somente os impactos aos seguintes subatributos de qualidade:

A - Compreensibilidade avalia a facilidade com que o executante do processo consegue compreender o processo;

B - Capacidade de modificabilidade reflete a facilidade em que o processo pode ser modificado. Este atributo de qualidade envolve a avaliação de esforço e impactos necessários para realizar uma mudança no processo.

Instruções: As perguntas a seguir devem ser avaliadas da seguinte forma:

- Ler o enunciado que explica determinado *process smell* proposto;
- Opinar, se realmente, cada *process smell* proposto, pode ser caracterizado como um problema sob o ponto de vista de processo de software.

Enunciado para a questão 1:

Process Smell: *Message Chains*

Elemento impactado: Tarefa

Descrição: O *Message Chains* define longas cadeias de trocas de produtos de trabalho entre tarefas que podem representar dependências disfarçadas. Uma tarefa precisa de um produto de trabalho de entrada que para ser gerado precisa esperar por uma cadeia de outras tarefas serem executadas até que este produto de trabalho de entrada esteja pronto.

Possíveis Impactos: Quanto mais longa for a cadeia de tarefas, maior será a dificuldade na legibilidade do processo, uma vez que aumenta a distância entre a tarefa solicitante e a tarefa que entregará os produtos de trabalho de entrada

solicitados, afetando a compreensibilidade do processo.

Representação:

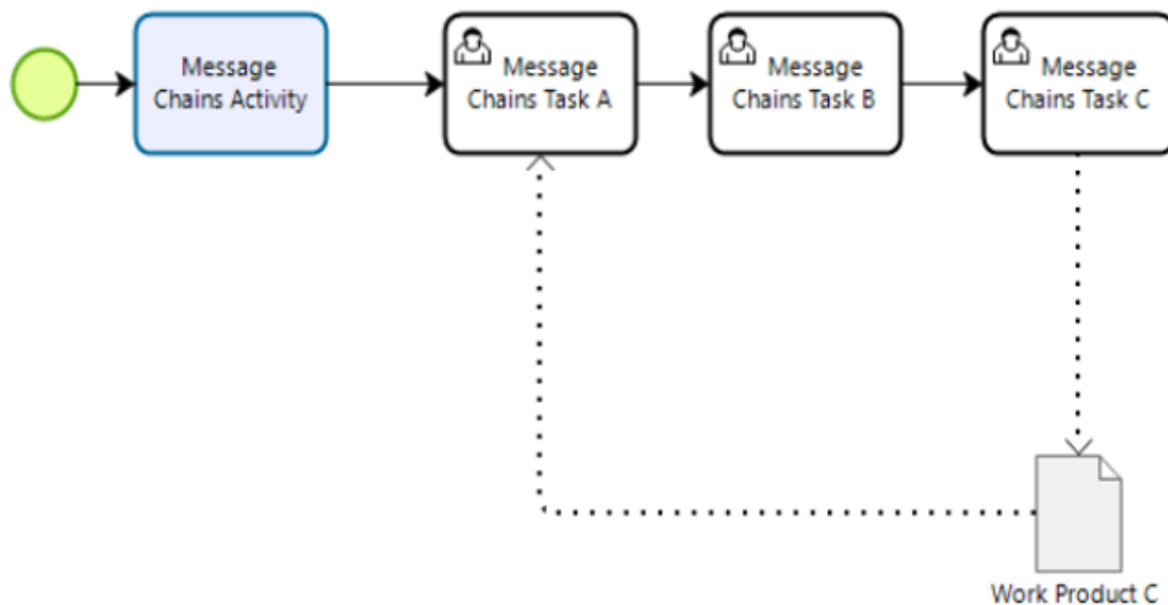


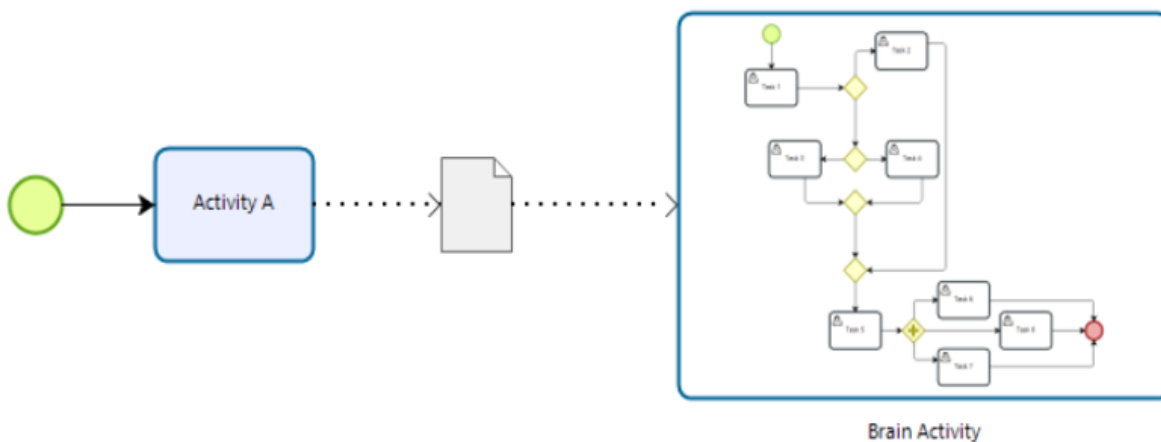
Figura A.1 *Message Chains*

Questão 1 - Uma cadeia longa de tarefas pode ser considerada um process smells?

- A - Concordo Plenamente
 - B - Concordo Parcialmente
 - C - Neutro
 - D - Discordo Plenamente
 - E - Discordo Parcialmente
- Justifique sua escolha

Esse process smell afeta negativamente o atributo de qualidade compreensibilidade?

- A - Concordo Plenamente
 - B - Concordo Parcialmente
 - C - Neutro
 - D - Discordo Plenamente
 - E - Discordo Parcialmente
- Justifique sua escolha

Process Smell: *Brain Activity***Elemento Impactado:** Atividade**Descrição:** Uma Brain Activity é uma atividade que centraliza as ações de um Processo. Essa atividade possui a maior parte das ações relevantes para o processo. Possui muitas tarefas e muitos fluxos de decisão para estruturar as tarefas.**Possíveis Impactos:** Atividades centralizadoras que concentram muitas tarefas e fluxos de decisão se tornam atividades grandes e complexas e podem reduzir a compreensibilidade do usuário do processo.**Representação:****Figura A.2** *Brain Activity*

Questão 2 - Uma atividade que possui muitas tarefas e possui muitos fluxos de decisão pode ser considerada um process smells?

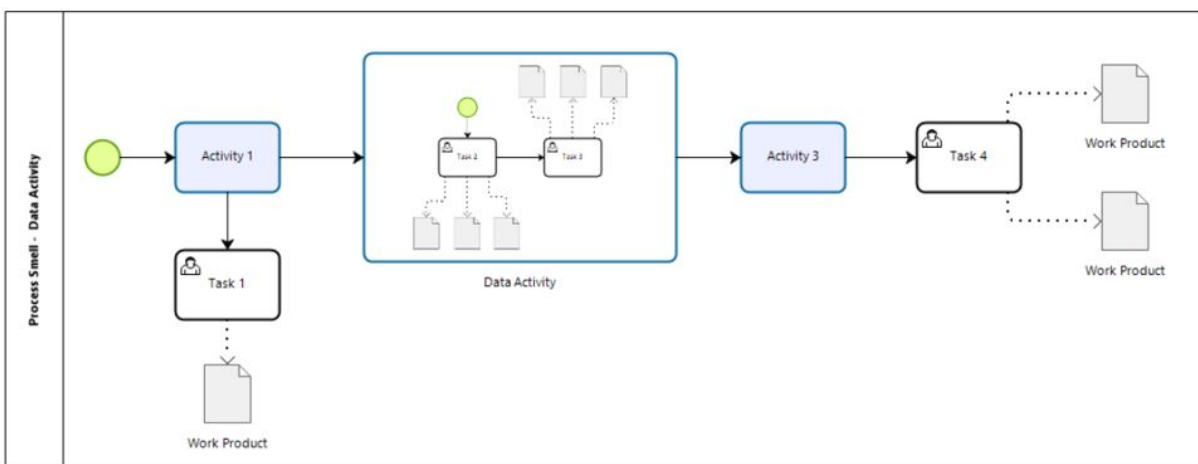
- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente
- E - Discordo Parcialmente

Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade compreensibilidade?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente
- E - Discordo Parcialmente

Justifique sua escolha

Process Smell: Data Activity**Elemento Impactado:** Atividade**Descrição:** Uma atividade se comporta como uma Data Activity quando não possui muitas tarefas, mas fornecem bastante produtos de trabalho de saída (Outputs).**Possíveis Impactos:** Um agrupamento de muitos produtos de trabalho de Saída em uma mesma atividade pode originar tanto um acoplamento disperso (um relacionamento dependente entre muitas atividades) tanto um acoplamento intensivo (um relacionamento dependente entre poucas atividades), dificultando a capacidade de modificação do processo, uma vez que existem muitas dependências associadas a essa atividade.**Representação:****Figura A.3** *Data Activity*

Questão 3 - Uma atividade que não exerce uma função principal no processo e entrega bastante produtos de trabalho de saída pode ser considerada um process smells?

- A - Concordo Plenamente
 - B - Concordo Parcialmente
 - C - Neutro
 - D - Discordo Plenamente
 - E - Discordo Parcialmente
- Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade capacidade de modificação?

- A - Concordo Plenamente
- B - Concordo Parcialmente

C - Neutro

D - Discordo Plenamente

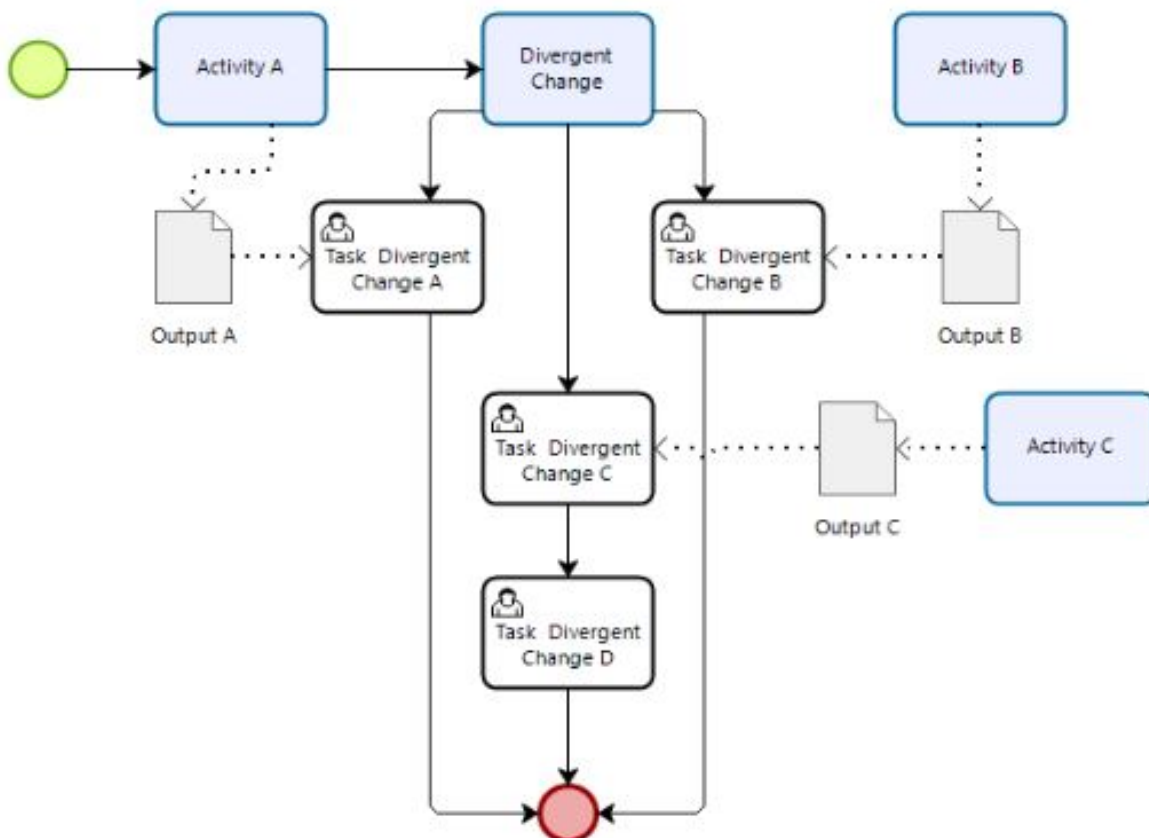
E - Discordo Parcialmente

Justifique sua escolha

Process Smell: Divergent Change**Elemento Impactado:** Atividade

Descrição: Uma atividade tem como propósito atender alguma "necessidade" do processo. Um exemplo de "necessidade" em um processo de software pode ser: construir o Termo de Abertura do projeto; validar um documento de especificação; entre outras. Assim, o Divergent Change ocorre quando uma atividade é configurada para atender a não somente uma "necessidade" do processo, tendo assim, diferentes conjuntos de tarefas para cada necessidade a qual atenderá. Em consequência, sempre que qualquer dessas "necessidades", que essa atividade atende sofrer mudanças, a parte das tarefas responsáveis por atendê-la também mudará, enquanto as outras tarefas que não fazem parte desse conjunto não sofrerão nenhuma mudança por este mesmo motivo.

Possíveis Impactos: Essa atividade se encarrega de diferentes propósitos, desta forma se torna uma atividade pouco coesa, ou seja, as tarefas desta atividade realizam ações de forma independente das outras tarefas, o que pode prejudicar a compressibilidade dessa atividade.

Representação:**Figura A.4** Divergent Change

Questão 4 - Uma atividade que é definida para atender a não somente uma "neces-

side” do processo pode ser considerada um process smells?

- A - Concordo Plenamente
 - B - Concordo Parcialmente
 - C - Neutro
 - D - Discordo Plenamente
 - E - Discordo Parcialmente
- Justifique sua escolha

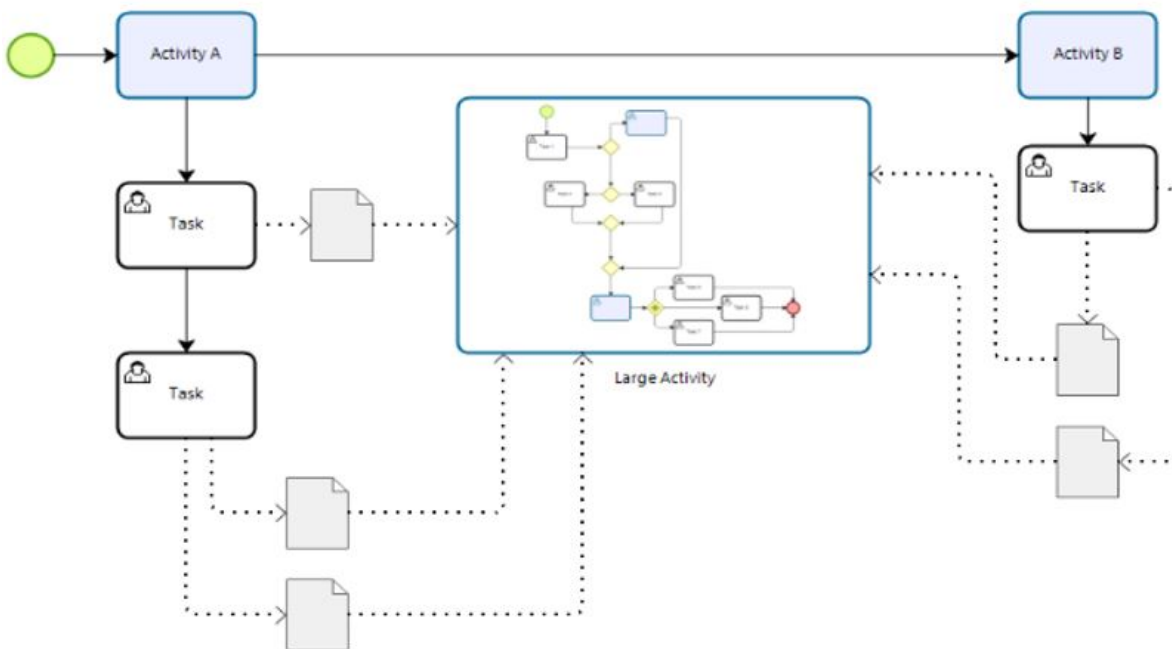
Esse process smells afeta negativamente o atributo de qualidade compreensibilidade?

- A - Concordo Plenamente
 - B - Concordo Parcialmente
 - C - Neutro
 - D - Discordo Plenamente
 - E - Discordo Parcialmente
- Justifique sua escolha

Process Smell: *Large Activity***Elemento Impactado:** Atividade

Descrição: Uma *Large Activity* se caracteriza quando uma atividade é grande, por conter muitas tarefas, e possui muitas responsabilidades. Ela centraliza as principais ações do Processo, deixando outras atividades com responsabilidades mais triviais e utilizando produtos de trabalho destas atividades para concluir seus objetivos.

Possíveis Impactos: Apesar da sensação de simplificar o modelo usando poucas atividades, a *Large Activity* possui muitas tarefas e muitos fluxos de decisão, assim aumenta a complexidade. A inserção de muitas tarefas em uma mesma atividade gera o risco de confundir a responsabilidade da atividade, adicionando tarefas que cuidam de diferentes necessidades do processo. Este fato pode também impactar negativamente a coesão entre as tarefas. Essas duas características podem reduzir a compreensibilidade da atividade.

Representação:**Figura A.5** *Large Activity*

Questão 5 - Uma atividade com muitas tarefas, que possui diferentes responsabilidades e faz uso de muitos produtos de entrada pode ser considerada um process smells?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente
- E - Discordo Parcialmente

Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade compreensibilidade?

A - Concordo Plenamente

B - Concordo Parcialmente

C - Neutro

D - Discordo Plenamente

E - Discordo Parcialmente

Justifique sua escolha

Process Smell: Feature Envy

Elemento Impactado: Tarefa

Descrição: O Feature Envy é representado por uma tarefa que faz uso extensivo de produtos de trabalho de saída de outra atividade. Por esse motivo, talvez essa tarefa deve pertencer a outra atividade.

Possíveis Impactos: O Feature Envy é reconhecido pela distância que os produtos de trabalho de entrada têm em relação à tarefa que as irá utilizá-los, que normalmente estão em outra atividade e não na atividade a qual a tarefa pertence. Assim, quanto maior for a medida dessa distância, menor será a compreensibilidade do executante do processo, principalmente para os casos, onde o executante terá que entender a tarefa fornecedora.

Representação:

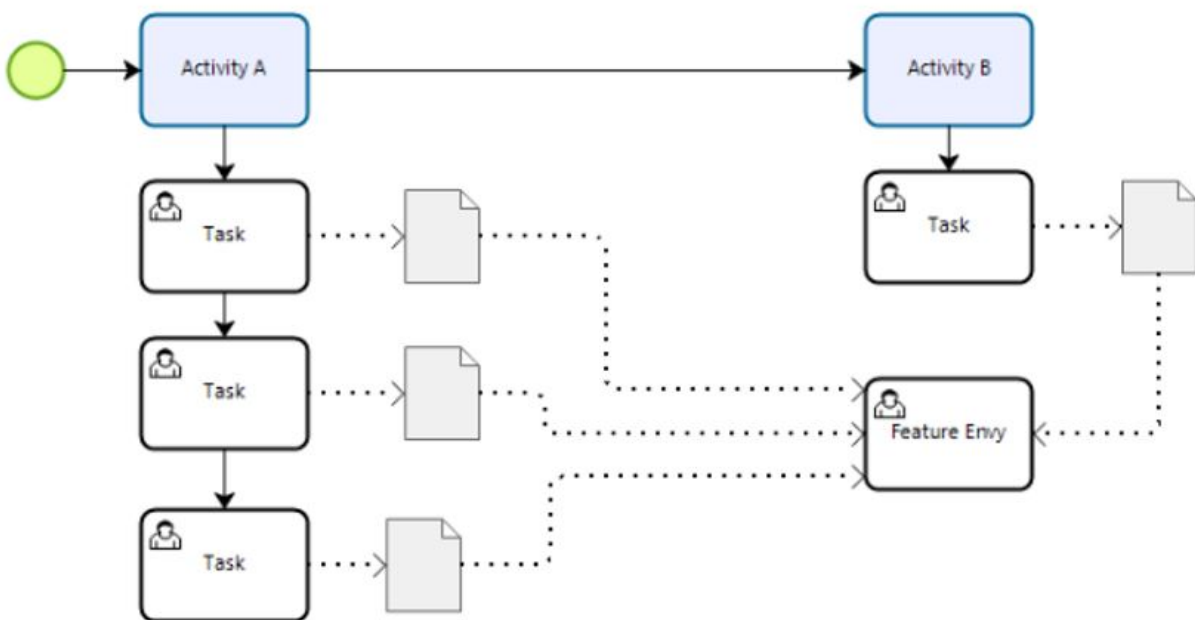


Figura A.6 *Feature Envy*

Questão 6 - Uma tarefa que faz uso extensivo de produtos de trabalho de saída de outra atividade pode ser considerada um process smells?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente
- E - Discordo Parcialmente

Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade compreensibilidade?

- A - Concordo Plenamente
 - B - Concordo Parcialmente
 - C - Neutro
 - D - Discordo Plenamente
 - E - Discordo Parcialmente
- Justifique sua escolha

Process Smell: *Shotgun Surgery*

Elemento Impactado: Atividade

Descrição: O *Shotgun Surgery* se caracteriza por uma atividade que quando sofre uma mudança (em suas tarefas ou passos) desencadeará a necessidade de muitas pequenas mudanças em várias outras atividades.

Possíveis Impactos: Atividades com *Shotgun Surgery* ocasionam de acoplamento disperso no processo, ou seja, atividades dependentes que estão espalhadas pelo processo. Assim, mudanças estruturais precisarão de maior esforço na rastreabilidade dos pontos que precisam ser ajustados, de forma a reestruturar o processo de forma consistente. Este fato reduz a capacidade de modificação.

Representação:

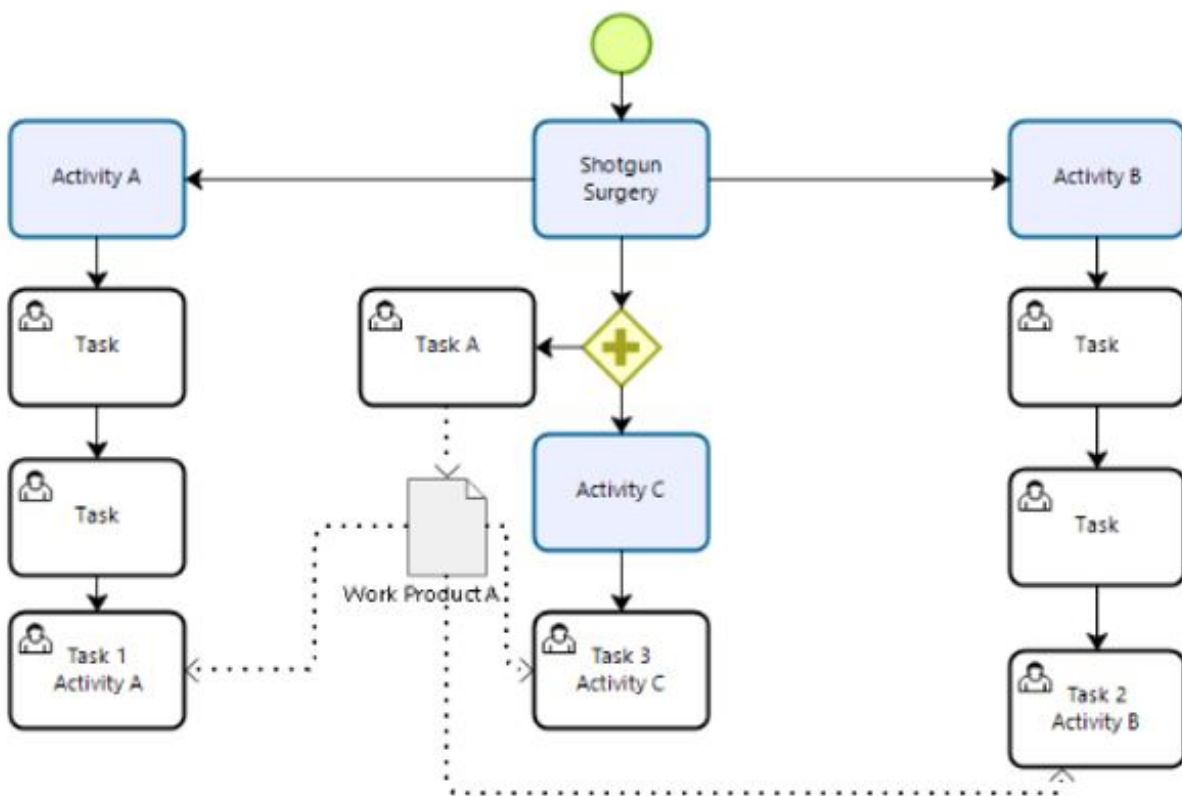


Figura A.7 *Shotgun Surgery*

Questão 7- Uma atividade que ao passar por mudanças podem desencadear mudanças em muitas outras atividades pode ser considerada um process smells?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente
- E - Discordo Parcialmente

Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade capacidade de modificação?

A - Concordo Plenamente

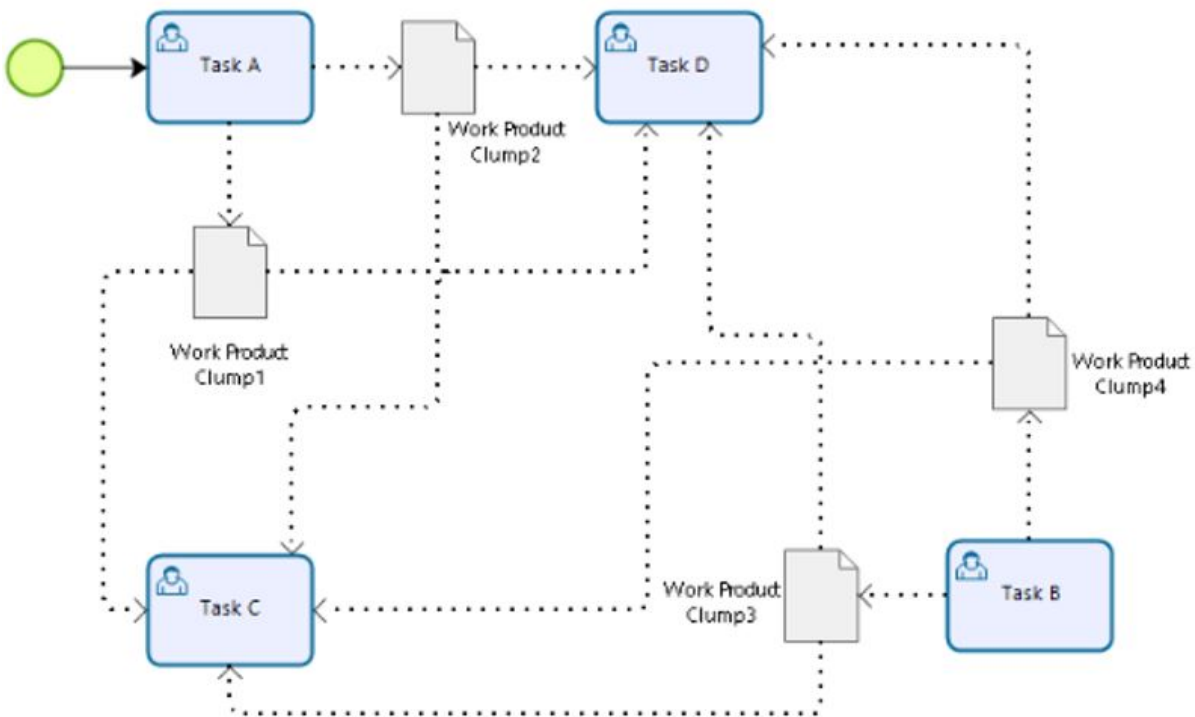
B - Concordo Parcialmente

C - Neutro

D - Discordo Plenamente

E - Discordo Parcialmente

Justifique sua escolha

Process Smells: *Work Product Clump***Elemento impactado:** Tarefa**Descrição:** O *Work Product Clump* acontece quando um conjunto de produtos de trabalho de entrada ou de saída de determinadas atividades/tarefas, são constantemente vistos juntos. Isso pode indicar que eles deveriam pertencer a uma mesma atividade/tarefa.**Possíveis Impactos:** O *Work Product Clump* promove a dispersão de um conjunto de produtos de trabalho, todos necessários para a conclusão de certas tarefas do processo. Essa dispersão reduz a capacidade de modificabilidade do processo, uma vez que será necessário um constante esforço de rastreabilidade desses produtos de trabalhos dispersos sempre que houver alguma mudança nesse conjunto.**Representação:****Figura A.8** *Work Product Clump*

Questão 8 - Em um cenário onde produtos de trabalho que deveriam estar agrupados estão dispersos pode ser considerado um process smells?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente
- E - Discordo Parcialmente

Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade capacidade de modificação?

A - Concordo Plenamente

B - Concordo Parcialmente

C - Neutro

D - Discordo Plenamente

E - Discordo Parcialmente

Justifique sua escolha

Process Smell: *Brain Task*

Elemento Impactado: Tarefa

Descrição: Uma *Brain Task* é uma tarefa longa que centraliza as ações de uma atividade, possui muitos passos e fluxos de decisão para estruturar os passos.

Possíveis Impactos: Apesar da sensação de simplificar o modelo usando poucas tarefas, uma *Brain Task* tem muitos passos e muitos fluxos de decisão, o que aumenta a complexidade e reduz a compreensibilidade da tarefa que sofre esse *process smell*.

Representação:

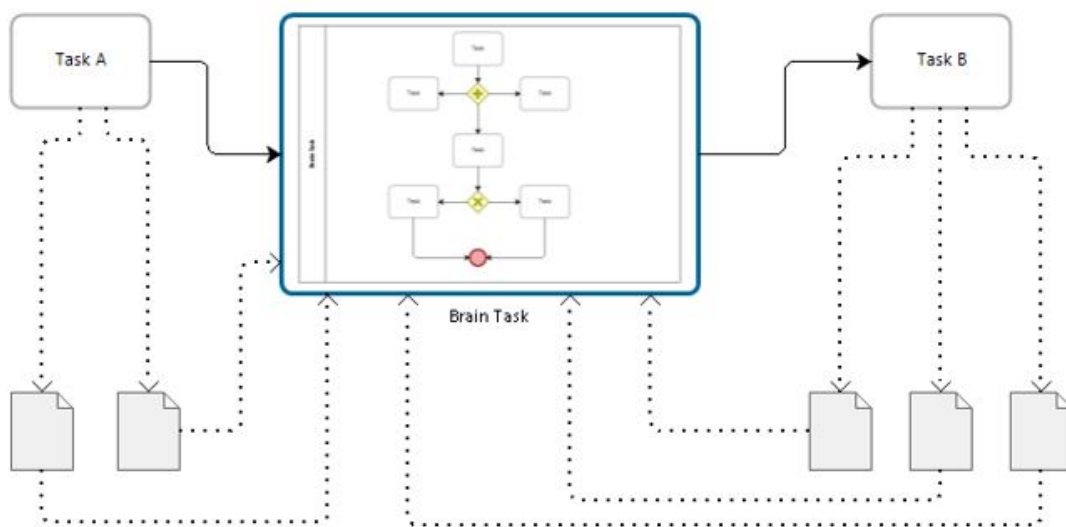


Figura A.9 *Brain Task*

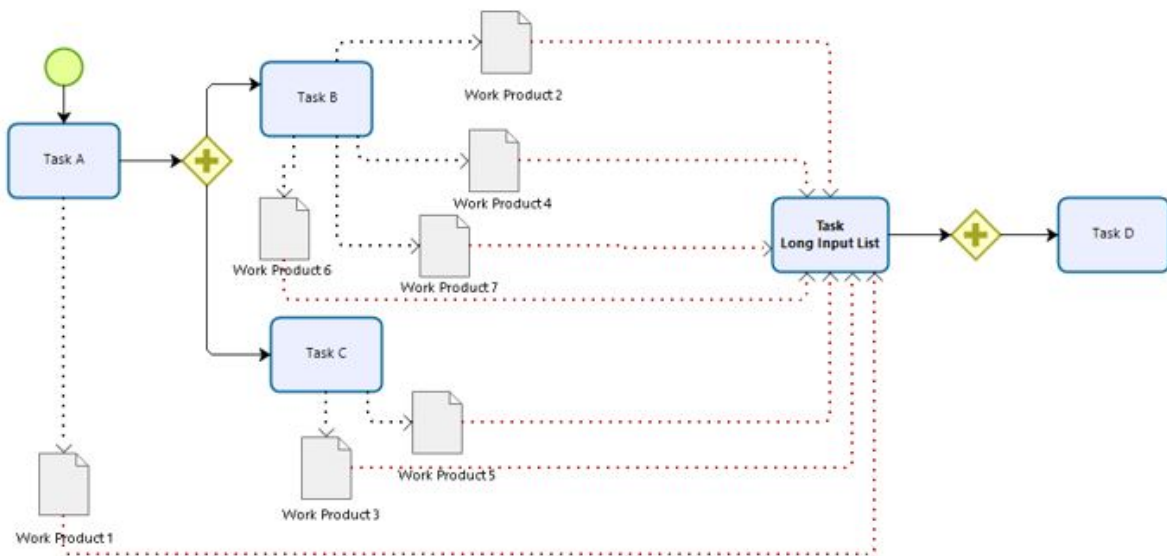
Questão 9 - Uma tarefa que possui muitos passos e muitos fluxos pode ser considerada um process smells?

- A - Concordo Plenamente
 - B - Concordo Parcialmente
 - C - Neutro
 - D - Discordo Plenamente
 - E - Discordo Parcialmente
- Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade compreensibilidade?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro

D - Discordo Plenamente
E - Discordo Parcialmente
Justifique sua escolha

Process Smell: Long Input List**Elemento Impactado:** Tarefa**Descrição:** O *Long Input List* é configurado quando uma tarefa precisa de muitos produtos de trabalho de entrada.**Possíveis Impactos:** A tarefa que possui uma longa lista de produtos de trabalho de entrada, pode estar sendo impactada por alguns problemas como: acúmulo de responsabilidades ou ser uma tarefa longa por possuir muitos passos e fluxos. E por sua vez pode prejudicar a compreensibilidade do usuário do processo.**Representação:****Figura A.10** *Long Input List*

Questão 10 - Uma tarefa que possui uma longa lista de produtos de trabalho de entrada pode ser considerada um process smells?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente
- E - Discordo Parcialmente

Justifique sua escolha

Esse process smells afeta negativamente o atributo de qualidade compreensibilidade?

- A - Concordo Plenamente
- B - Concordo Parcialmente
- C - Neutro
- D - Discordo Plenamente

E - Discordo Parcialmente
Justifique sua escolha

Process Smells

Prezado(a), este questionário destina-se a coletar dados para uma pesquisa na área de Ciência da Computação, conduzida na Universidade Federal da Bahia (UFBA), com objetivo de receber opiniões de concordância ou não, sobre process smells, uma proposta de situações em geral, que acreditamos que prejudicam a qualidade de processos de software. A sua participação é de fundamental importância.

*Obrigatório

Termo de Consentimento Livre e Esclarecido

- 1) O/A senhor(a) está sendo convidado(a) a participar voluntariamente da pesquisa sobre Process Smells.
- 2) Sua participação não é obrigatória.
- 3) A qualquer momento você pode desistir de participar e retirar seu consentimento.
- 4) Sua recusa não trará nenhum prejuízo em sua relação com a pesquisadora ou com a instituição.
- 5) Este formulário tem por objetivo receber opiniões de concordância ou não, sobre process smells.
- 6) Sua participação neste formulário consistirá em responder questões objetivas e subjetivas.
- 7) Sua identificação é opcional, ou seja, você não precisa informar nome ou e-mail caso assim deseje.
- 8) A aplicação do formulário está sendo realizada por Renata de Sousa Santos, estudante da pós graduação em Ciência da Computação na Universidade Federal da Bahia, sob a supervisão da Prof^a. Dr^a. Rita Suzana Pitangueira Maciel.
- 9) Os benefícios relacionados à sua participação estão apenas em contribuir com a pesquisa científica. Será permitido acesso aos resultados desta pesquisa por meio da dissertação ou publicações científicas realizadas a partir desse estudo.
- 10) As informações pessoais obtidas através desta pesquisa serão confidenciais e não serão distribuídas ou divulgadas pela pesquisadora.
- 11) Os dados coletados neste formulário não serão divulgados de forma a possibilitar sua identificação.
- 12) Ao continuar respondendo este questionário, o/a senhor(a) concorda com as informações aqui descritas, porém a qualquer momento o/a senhor(a) poderá interromper a pesquisa sem ônus algum.
- 13) Este questionário utiliza o pacote de aplicativo Google Docs, portanto a coleta e o uso de informações do Google estão sujeitos à Política de privacidade do Google (<https://www.google.co.uk/policies/privacy/>).
- 14) Abaixo seguem os dados de contato dos responsáveis por esta pesquisa, com os quais você pode tirar suas dúvidas sobre sua participação.
Pesquisadores responsáveis:
Renata de Sousa Santos - renatassantos@ufba.br
Rita Suzana Pitangueira Maciel - ritasuzana@dcc.ufba.br (supervisora, link lattes <http://lattes.cnpq.br/0290950513716961>).
Universidade Federal da Bahia (UFBA)
Instituto de Matemática e Estatística
Departamento de Ciência da Computação
Av. Adhemar de Barros, s/n, Ondina, 40170-110, Salvador – BA.

Salvador - Ba, Setembro de 2020.

DECLARO QUE ENTENDI OS OBJETIVOS, RISCOS E BENEFÍCIOS DE MINHA PARTICIPAÇÃO NA PESQUISA. * *

Concordo

Caro(a) Participante, saudações!

Sou aluna do Mestrado, orientada pela Profª. Rita Suzana Pitangueira Maciel na UFBA e estou trabalhando na ESPECIFICAÇÃO DE UM CATÁLOGO DE *BAD SMELLS* PARA PROCESSO DE SOFTWARE UTILIZANDO NOTAÇÃO BPMN.

Gostaria de contar com a sua colaboração em uma avaliação que irá em muito contribuir com o meu projeto de pesquisa.

Objetivo do estudo: Realizar entrevistas com especialistas ou pessoas que já tiveram alguma experiência em processo de software a fim de avaliar a percepção do entrevistado em relação a modelagem dos *smells* propostos.

A entrevista será dividida em fases, seguindo o seguinte planejamento:

1. Inicialmente o entrevistado deverá responder ao questionário sobre a caracterização de perfil e ao conceito de processo de software. Essa seção é composta por 10 questões, sendo 9 múltipla escolha e 1 de resposta curta.

2. Na segunda etapa será apresentada a definição de *process smells* bem como a descrição, representação e possíveis impactos causados dos respectivos *smells* propostos. Em seguida, o entrevistado responderá perguntas com base na análise da modelagem de cada *smell*.

3. A terceira e última etapa tem como objetivo obter a opinião dos respondentes em relação ao trabalho apresentado.

As entrevistas serão feitas por Skype e agendadas de acordo com a disponibilidade do entrevistado, preferencialmente nos finais de tarde.

Desde já, agradecemos sua disponibilidade em participar da nossa entrevista. Pedimos que nos envie o seu endereço Skype junto com a confirmação de disponibilidade para participação no estudo.