



PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
UNIVERSIDADE FEDERAL DO AMAZONAS
INSTITUTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA



Diretivas de Comunicabilidade para Artefatos de Software

Adriana Lopes Damian

Manaus

2020

Adriana Lopes Damian

Diretivas de Comunicabilidade para Artefatos de Software

Tese submetida ao corpo docente do Programa de Pós-Graduação em Informática da Universidade Federal do Amazonas.

Área de concentração: Engenharia de Software.

Orientadora: Prof^a. Dra. Tayana Uchôa Conte

Colaboradora: Prof^a. Dra. Clarisse Sieckenius de Souza

Manaus

2020

Ficha Catalográfica

Ficha catalográfica elaborada automaticamente de acordo com os dados fornecidos pelo(a) autor(a).

D158d Damian, Adriana Lopes
Diretivas de Comunicabilidade para Artefatos de Software /
Adriana Lopes Damian . 2020
III f.: il. color; 31 cm.

Orientadora: Tayana Uchôa Conte
Tese (Doutorado em Informática) - Universidade Federal do
Amazonas.

1. comunicação através de artefatos de software. 2. comunicação
através de modelos de software. 3. Engenharia Semiótica. 4.
Engenharia de Software. I. Conte, Tayana Uchôa. II. Universidade
Federal do Amazonas III. Título



PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
INSTITUTO DE COMPUTAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA



UFAM

FOLHA DE APROVAÇÃO

"DIRETIVAS DE COMUNICABILIDADE PARA ARTEFATOS DE SOFTWARE"

ADRIANA LOPES DAMIAN

Tese de Doutorado defendida e aprovada pela banca examinadora constituída pelos Professores:

Prof. Tayana Uchôa Conte - PRESIDENTE

Prof. Bruno Freitas Gadelha - MEMBRO INTERNO

Prof. Luciana Aparecida Martinez Zaina - MEMBRO EXTERNO

Prof. Cleidson Ronald Botelho de Souza - MEMBRO EXTERNO

Prof. Simone Diniz Junqueira Barbosa - MEMBRO EXTERNO

Manaus, 08 de Setembro de 2020

*“Tu és o meu Deus; graças te darei! Ó meu Deus, eu te exaltarei!
Deem graças ao Senhor, porque Ele é bom; o seu amor dura para sempre”(Salmos 118:28,29)*

Agradecimentos

Durante essa pesquisa, aprendi que uma das razões para a comunicação humana falhar está relacionada à falta de informações necessárias. Talvez eu falhe aqui ao expressar meus agradecimentos, pois não há palavras que possam exprimir o quanto eu tenho a agradecer aos que contribuíram, diretamente e indiretamente, para a conclusão desta tese.

Agradeço primeiramente ao meu Deus, por me dar sabedoria, inspiração, ânimo, saúde e pessoas tão especiais que me encorajaram a enfrentar os desafios desta jornada. Sempre te louvarei!

Aos meus pais, Enildo e Ana Lopes, que sempre me incentivaram a ir além, com todo o amor possível. Todas as vezes que eu me senti incapaz ou precisei desabafar, eles estavam ali com uma palavra sábia e com suas orações para eu pudesse aquietar minha alma. Muito dos passos que ando são meus porque vocês me apoiaram e me ensinaram que nada é impossível quando pedimos a direção de Deus. Aos meus avós, Francisco (*in memoriam*) e Isabel, pelo amor incondicional, eu bem sei dos esforços deles desde a minha infância, minha base de tudo. Às minhas queridas irmãs, Thayane, Maria Isabel e Eloana Paula, que se alegram com as minhas vitórias. Ao meu sobrinho Thiago, que alegra o coração de sua 'super tia'. Ao meu esposo, Daniel Damian, por estar ao meu lado sempre que precisei, por compreender meus momentos de ausência e por acreditar em mim. Agradeço à minha sogra Lícia Mara, pela sua amizade e carinho de sempre. Aos meus tios, Marly Lopes e Valquimar Muniz, que me receberam em sua casa, quando eu tinha apenas 14 anos, para eu estudar em Manaus. Agradeço aos meus primos Ana Carolina Lopes e Marcus Vinícius Lopes, que sempre me trataram como irmã. Com certeza, esse foi o passo inicial para que eu pudesse chegar até aqui.

À minha orientadora, mãe acadêmica, Tayana Conte por toda sua dedicação neste tempo. Eu lembro muito bem quando escrevi para ela, pela primeira vez, pedindo uma orientação de TCC. Ela prontamente me respondeu e me recebeu com várias ideias. Foi um momento inspirador e um grande passo para mim em relação à pesquisa. Essa orientação resultou no meu primeiro artigo e eu tive a oportunidade de estar em um congresso nacional e me sentir segura, apesar do nervosismo por apresentar um artigo pela primeira vez. O principal ponto aqui é a diferença que ela faz na vida dos seus alunos como orientadora. Se eu pudesse representar ela através de uma estrela, eu escolheria o sol. Ela é cheia de luz e proporciona energia para aqueles que se apaixonam por pesquisa científica. Eu nunca esquecerei do companheirismo dela nos meus momentos alegres e críticos, sempre pronta para estender a mão para alguém que não poderia retribuir da mesma maneira. Por isso digo que és um presente de Deus para a minha vida! Admiro você porque és apaixonada por tudo que faz e não mede esforços para aqueles que são orientados por você alcancem o seu melhor. Quero continuar seguindo seus passos, és uma pesquisadora brilhante. Meu crescimento, com certeza, vem do compartilhamento do seu conhecimento e experiência. Espero que continuemos pesquisando juntas e que nossa amizade só fortaleça.

À minha colaborada querida, coorientadora deste trabalho, Clarisse Sieckenius de Souza, que sempre compartilhou seu conhecimento com carinho e paciência. Tenho orgulho de ter recebido

orientações valiosas para esta pesquisa e sou grata pela sua atenção em todos os momentos desta jornada. Eu também agradeço aos pesquisadores do grupo SERG por terem me recebido tão bem na PUC-Rio no início do doutorado, quando tive a oportunidade de realizar a missão de estudos nesta instituição. Foi um momento muito importante, pois recebi excelentes contribuições.

Aos professores que também contribuíram para a formação desta pesquisadora, Simone Diniz Junqueira Barbosa e Alessandro Garcia que, além de me receberem bem na PUC-Rio, me deram a oportunidade de ver diferentes perspectivas da pesquisa científica. Ao professor Bruno Gadelha, por compartilhar sua experiência com Design Science Research e pelos conselhos preciosos durante os USES Day. Ao Edson Oliveira, que sempre forneceu excelentes contribuições ao longo desta pesquisa. Todas as vezes que colaboramos, eu aprendi muito e sou grata por isso.

Aos amigos do grupo de pesquisa USES, Sérgio Cavalcante e Ana Carolina Oran, que intermediaram a condução de estudos na indústria e me ajudaram de todas as maneiras possíveis. Também gostaria de agradecer especialmente a Luana Lobão, Carla Oran e Edna Dias, que se dispuseram a utilizar a proposta desta pesquisa em seus projetos de software.

Aos demais amigos do grupo de pesquisa USES, meus queridos companheiros, Anna Beatriz Marques, Natasha Valentim, Ursula Campos, Elizamary Nascimento, Márcia Lima, Walter Nakamura, Felipe Manzoni, Luis Rivero, Jacilane Rabelo, Leonardo Marques, Randerson Silva, Williamson Silva, Flávio Coelho e Patrícia Matsubara. Contar com a amizade e cooperação de vocês foi um prazer! Aos amigos do grupo de pesquisa OPUS, em especial ao Leonardo da Silva Sousa e Roberto Oliveira por momentos incríveis em nossa colaboração.

Aos professores Cleidson de Souza e Luciana Zaina por aceitarem participar da minha banca de qualificação de doutorado. Tive a oportunidade de ouvir valiosas contribuições e evoluir esta pesquisa. Agradeço mais uma vez ao Cleidson de Souza e Luciana Zaina e aos professores Bruno Gadelha e Simone Diniz Junqueira Barbosa que aceitaram participar da defesa de doutorado. Foi uma honra ter cada um de vocês neste dia tão importante para mim.

À CAPES, pela bolsa concedida no durante o doutorado. Ao projeto PROCAD, pelo apoio concedido durante a missão de estudos no início do doutorado e apoio às conferências nacionais e internacionais que apresentei resultados desta pesquisa. Participar de conferências e ver tantos trabalhos excelentes me fizeram amar ainda mais a pesquisa científica.

Agradeço ao professor Eduardo Feitosa, coordenador da pós-graduação do Instituto de Computação. Também agradeço aos funcionários da secretaria por prestarem apoio quando precisei de informações relacionadas ao doutorado, em especial ao Frank Azevedo.

Aos meus queridos pastores Kennedy e Silvana Oliveira, por suas orações e conselhos. Aos meus amigos Jéssica dos Anjos, Aline e Ederson Martins, Jackeline Nascimento, Atila e Isaque Tomaz. Louvar a Deus com vocês é muito especial! Creio que nenhum resultado é o mesmo quando pedimos direção de Deus para nossos trabalhos.

Resumo

Diretivas de Comunicabilidade para Artefatos de Software

Orientadora: Tayana Uchôa Conte

Colaboradora: Clarisse Sieckenius de Souza

Os artefatos de software elaborados nas etapas iniciais do processo de desenvolvimento auxiliam no entendimento e no projeto da estrutura do sistema, além da forma como os usuários podem interagir com tal sistema. Por descreverem as soluções propostas para o desenvolvimento de software, estes artefatos são importantes para a comunicação de equipes. Falhas de comunicação podem ocorrer porque os produtores tendem a se concentrar apenas no conteúdo dos artefatos, sem refletir como estes serão interpretados pelos demais membros da equipe de desenvolvimento, os consumidores. Neste contexto, esta tese teve por objetivo responder a seguinte questão de pesquisa: Como apoiar a comunicação efetiva de produtores e consumidores de artefatos desenvolvidos nas etapas iniciais do desenvolvimento de software? Para responder essa questão, esta pesquisa foi baseada no ciclo de Design Science Research (DSR). O ciclo de DSR inicia com a investigação de um problema e então são especificados artefatos como soluções, que são avaliados no contexto do problema. Os resultados dessa avaliação podem iniciar uma nova volta no ciclo de DSR. O problema abordado nesta tese trata-se de um *wicked problem*, no qual os pesquisadores buscam soluções que aproximem uma determinada lacuna. Portanto, a solução desenvolvida nesta tese está relacionada à tentativa de mitigar falhas de comunicação a partir de artefatos de software, promovendo assim a comunicação efetiva de equipes de desenvolvimento através destes artefatos. Para a investigação do problema, foram realizados diferentes estudos experimentais e um mapeamento sistemático da literatura, cujos resultados forneceram *insights* para a proposta de um artefato como solução. Com isso, as Diretivas de Comunicabilidade (DCs) foram propostas, fundamentadas em teorias que investigam diferentes perspectivas de comunicação, com o objetivo de apoiar a reflexão dos produtores sobre sua comunicação através dos artefatos de software. Foram realizados diferentes estudos experimentais para avaliar o uso das DCs para a redução dos riscos de falhas de comunicação em artefatos de software e suas contribuições para a comunicação de equipes a partir destes artefatos. Os resultados indicaram que a reflexão dos produtores sobre sua comunicação através do artefato, com base nas DCs, contribuem para uma comunicação eficaz. Portanto, as DCs podem ser aplicadas no desenvolvimento de software para apoiar a comunicação via artefato de software.

Palavras-chave: comunicação através de artefatos de software, comunicação através de modelos de software.

Abstract

Directives of Communicability for Software Artifacts

Advisor: Tayana Uchôa Conte

Colaborator: Clarisse Sieckenius de Souza

Software artifacts developed in the early stages of the software development process help practitioners to understand the structure and architecture design of the system, as well as how users can interact with the system. As these artifacts describe the proposed solutions for software development, they are important for team communication. Communication failures can occur because producers tend to focus only on the content of artifacts, although they should also reflect on how model consumers will interpret them. In this context, this thesis aimed to answer the following research question: How to support effective communication between producers and consumers of artifacts developed in the early stages of software development? To answer this question, this research was based on the Design Science Research (DSR) cycle. The DSR cycle begins with the investigation of a problem and then artifacts are specified as solutions, which are evaluated in the context of the problem. The results of this evaluation could start a new round in the DSR cycle. The problem addressed in this thesis is a wicked problem, in which researchers seek solutions for the reduction of a specific gap. Therefore, the solution developed in this thesis aims to mitigate communication failures from software artifacts, promoting effective software development team communication through these artifacts. To investigate the problem, we carried out different empirical studies and a systematic mapping of the literature. The results provided insights for the proposal of an artifact as a solution. Thereby, we proposed the Directives of Communicability (DCs), based on theories that investigate different communication perspectives, to support producers' reflections on their communication through software artifacts. We carried out different empirical studies for evaluating the DCs to reduce the risks of communication failures in artifacts and to support team communication through these artifacts. The results indicated that the producers' reflection based on DCs promotes effective communication through the artifact. Therefore, DCs can be applied in software development to support effective communication through software artifact.

Keywords: communication through software artifacts, communication through software models.

Lista de Figuras

Figura 1.1: Cenário dos Artefatos de Software como um dos Apoios à Comunicação de Equipes.	19
Figura 1.2: Metodologia Utilizada na Pesquisa (Peppers et al. (2007)).	25
Figura 1.3: Execução do Ciclo de DSR – Primeira e Segunda Iteração.	27
Figura 2.1: Formulário de Metacomunicação (adaptado de Souza et al., 2016).	35
Figura 2.2: Análise de Mockups com o Formulário de Metacomunicação - adaptado de De Mello Brandão et al. (2016).	35
Figura 2.3: Estrutura de Metacomunicação (adaptado de Souza et al., 2016).	36
Figura 2.4: Análise de Mockups com o Formulário de Metacomunicação - adaptado de De Souza et al. (2016).	36
Figura 3.1: Espaço de Comunicação entre Produtores e Consumidores de Artefatos.	40
Figura 3.2: Condução do Mapeamento Sistemático.	46
Figura 3.3: String de Busca Utilizada no Mapeamento Sistemático.	48
Figura 3.4: Procedimento para a Seleção dos Artigos.	51
Figura 3.5: Distribuição das Publicações por Ano.	52
Figura 3.6: Artefatos Relatados Na Literatura que Apoiam a Comunicação no Desenvolvimento de Software.	53
Figura 3.7: Notações Mencionadas na Produção de Artefatos que Apoiam a Comunicação. ...	54
Figura 3.8: Artefatos investigados em Pesquisas Relacionadas à Comunicação no Desenvolvimento de Software.	55
Figura 3.9: Estado da Arte sobre Propostas que Apoiam a Comunicação a partir dos Artefatos.	57
Figura 3.10: Análise das Propostas para a Produção dos Artefatos com Objetivo de Melhorar a Comunicação.	58
Figura 3.11: Pesquisas que contribuem para a Comunicação Via Artefatos.	61
Figura 3.12: Contexto dos Estudos apresentados nas Publicações encontradas.	63
Figura 3.13: Oportunidades de Pesquisa para a Melhoria da Comunicação via Artefato.	64
Figura 4.1: Condução de Estudos Experimentais sobre a Comunicação via Artefato.	66
Figura 4.2: Atividades Realizadas para a Construção dos Artefatos na Sprint 2.	70
Figura 4.3: Atividades Realizadas para a Construção dos Artefatos na Sprint 3.	72
Figura 4.4: Análise do Mockup Produzido por P1 e P2 com SFYI Message.	80
Figura 4.5: Mapeamento de Atividades dos Participantes na Construção de Mockups a partir de Casos de Uso – P1 e P2.	82
Figura 4.6: Número de Falhas de Comunicação Relacionadas aos Problemas de Comunicabilidade do UC.	85
Figura 4.7: Etapas Realizadas no Estudo.	89
Figura 4.8: Total de Violações das Máximas de Grice na Especificação de Caso de Uso de Cada Dupla Produtora.	90
Figura 4.9: Violações das Máximas de Grice na Especificação de Casos de Uso que não Comprometeram o Princípio Cooperativo.	92
Figura 4.10: Princípio Cooperativo não Prejudicado q Partir de Máximas Violadas.	93
Figura 4.11: Violações das Máximas de Grice na Especificação de Casos De Uso que não Estabeleceram o Princípio Cooperativo.	93
Figura 5.1: Proposta das Diretivas de Comunicabilidade e Três Estudos Iniciais.	99
Figura 5.2: Exemplo de uso das Diretivas de Comunicabilidade.	102
Figura 5.3: Total de Riscos de Falhas de Comunicação nos Diagramas de Classes.	106
Figura 5.4: Total de Riscos de Falhas de Comunicação nos Diagramas BPMN.	107

Figura 5.5: Total de Riscos de Falhas de Comunicação nos Diagramas IFML..... 108

Figura 5.6: Exemplo de uso das Propostas Apresentadas nesta Tese. 117

Figura 6.1: Estudos na Indústria com as Diretivas de Comunicabilidade. 128

Figura 6.2: Etapas para a Condução do Estudo 1. 133

Figura 6.3: Associação dos Códigos e Dados qualitativos coletados durante o Estudo. 134

Figura 6.4: Categoria que Representa a Comunicação da Equipe 1..... 135

Figura 6.5: Fatores Relacionados à Comunicação via Artefato da Equipe 1. 136

Figura 6.6: Impacto das Atividades da Equipe 1 nas Informações dos Artefatos. 137

Figura 6.7: Benefícios obtidos para a Equipe 1 com a Comunicação via Artefato. 139

Figura 6.8: Categoria que Representa a Comunicação da Equipe 2..... 141

Figura 6.9: Fatores Relacionados à Comunicação via Artefato da Equipe 2. 142

Figura 6.10: Impacto das Atividades da Equipe 2 nas Informações dos Artefatos. 143

Figura 6.11: Benefícios obtidos para a Equipe 2 com a Comunicação via Artefato. 144

Figura 6.12: Variações do Uso da DC1, DC2 e combinação da DC2 e DC3 nos Artefatos... 145

Figura 6.13: Variações do Uso da DC3 e DC4 nos Artefatos..... 146

Figura 6.14: Percepções sobre as DCs: Apoio na melhoria do conteúdo dos artefatos..... 147

Figura 6.15: Percepções sobre as DCs: Apoio na reflexão do produtor sobre a comunicação via artefato. 148

Figura 6.16: Etapas para a Condução do Estudo 2. 152

Figura 6.17: Categoria que Representa o Apoio dos Artefatos na Comunicação da Equipe 3. 153

Figura 6.18: Fatores Relacionados à Comunicação via Artefato da Equipe 3. 154

Figura 6.19: Impacto das Atividades da Equipe 3 nas Informações dos Artefatos. 154

Figura 6.20: Diagnóstico dos Protótipos da Equipe 4. 155

Figura 6.21: Categoria que Representa a Comunicação da Equipe 4 Apoiada por Artefatos. 156

Figura 6.22: Práticas da Equipe 4 que Impactam a Comunicação via Artefato. 156

Figura 6.23: Fatores Relacionados à Comunicação via Artefato da Equipe 4. 157

Figura 6.24: Diagnóstico dos Protótipos da Equipe 4. 158

Figura 6.25: Categoria que Representa a Comunicação da Equipe 5 Apoiada por Artefatos. 159

Figura 6.26: Fatores Relacionados à Comunicação via Artefato da Equipe 5. 160

Figura 6.27: Práticas da Equipe 5 que impactam a Comunicação via Artefato..... 161

Figura 6.28: Variações do Uso das DCs no Diagnóstico dos Artefatos..... 161

Figura 6.29: Variações do Uso das DCs no Diagnóstico dos Artefatos..... 162

Figura 6.30: Percepções dos Produtores ao Aplicarem as DCs no Diagnóstico dos Artefatos. 163

Figura 7.1: Grau de Aceitação dos Participantes com relação ao uso das DCs na produção de artefatos. 168

Figura 8.1: Questionário de Compreensão da e-VOL BPMN..... 175

Figura 8.2: Exemplos de Cenários de Modificação..... 176

Figura 8.3: Gráfico Boxplot para Completude e Corretude dos modelos BPMN..... 178

Figura 8.4: Parte da técnica ComD2 para diagramas de classes. 182

Figura 8.5: Número de Defeitos no Diagrama de Classes..... 188

Lista de Tabelas

Tabela 3.1: Termos utilizados para Definir a String de Busca.	47
Tabela 3.2: Critérios para Inclusão de Artigos.	49
Tabela 3.3: Critérios para Exclusão de Artigos.	49
Tabela 3.4: Formulário de Extração.	50
Tabela 3.5: Interpretação dos resultados do Kappa segundo Landis e Koch (1977).	51
Tabela 4.1: Máximas de Grice adaptadas para a Análise do UC.	79
Tabela 4.2: Problemas de Comunicabilidade no Caso de Uso.	79
Tabela 4.3: Análise das Informações nos Mockups em Comparação com o UC.	81
Tabela 4.4: Símbolos utilizados no Mapeamento de Atividades dos Participantes.	82
Tabela 4.5: Falhas de Comunicação entre Produtores e Consumidores a partir do UC.	84
Tabela 4.6: Descrição das Máximas de Grice Violadas por cada Dupla Produtora na Especificação de Casos de Uso.	91
Tabela 4.7: Tipos de Defeitos (adaptados de Granda et al., 2015).	94
Tabela 4.8: Descrição dos defeitos Cometidos por Cada Dupla Consumidora da Especificação de Casos De Uso.	95
Tabela 5.1: Versão Inicial das Diretivas de Comunicabilidade.	101
Tabela 5.2: Versão Inicial das Diretivas de Comunicabilidade.	105
Tabela 5.3: Riscos de Falhas de Comunicação em Diagramas de Classes.	106
Tabela 5.4: Riscos de Falhas de Comunicação nos Diagramas BPMN.	107
Tabela 5.5: Riscos de Falhas de Comunicação nos Diagramas IFML.	108
Tabela 5.6: Emprego das Diretivas de Comunicabilidade na Modelagem.	109
Tabela 5.7: Passos para Utilizar a Segunda Versão das Diretivas de Comunicabilidade.	113
Tabela 5.8: Grupos de Modelagem de Casos de Uso com as DCs.	119
Tabela 5.9: Nível de Experiência dos Participantes do Segundo Estudo Experimental.	120
Tabela 5.10: Riscos de Falhas de Comunicação nos Artefatos Desenvolvidos.	120
Tabela 5.11: Percepção dos Participantes sobre as DCs na Produção de Casos de Uso.	121
Tabela 5.12: Propagação de Problemas de Comunicabilidade do UC para mockups.	125
Tabela 8.1: Resultados por participante com a família ComD2.	184
Tabela 8.2: Riscos de Falhas de Comunicação em Diagramas de Classes.	189
Tabela 8.3: Emprego das Diretivas de Comunicabilidade na Modelagem.	190
Tabela 9.1: Análise dos Mockups com a Ferramenta SFYI Message – Parte 1 (Grupo Experimental).	227
Tabela 9.2: Análise dos Mockups com a Ferramenta SFYI Message – Parte 2 (Grupo Experimental).	228
Tabela 9.3: Análise dos Mockups com a Ferramenta SFYI Message – Parte 3 (Grupo Experimental).	229
Tabela 9.4: Análise dos Mockups com a Ferramenta SFYI Message – Parte 1 (Controle).	230
Tabela 9.5: Análise dos Mockups com a ferramenta SFYI Message – Parte 2 (Controle). ...	231
Tabela 9.6: Análise dos Mockups com a Ferramenta SFYI Message – Parte 3 (Controle).	232

Sumário

CAPÍTULO 1 – INTRODUÇÃO	17
1.1. Contexto	17
1.2. Motivação	18
1.3. Definição do Problema	20
1.4. Objetivos de Pesquisa	21
1.5. Perfil da Pesquisadora Principal	22
1.6. Metodologia Utilizada na Pesquisa	23
1.7. Principais Contribuições	28
1.8. Publicações Resultantes	29
1.9. Organização	31
CAPÍTULO 2 – FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS	33
2.1. Fundamentação Teórica	33
2.1.1 Engenharia Semiótica	33
2.1.2 Princípio de Cooperação de Grice	37
2.1.3 Comunicação via Artefato e Apropriação da Teoria da Engenharia Semiótica para tal Comunicação	39
2.2. Trabalhos Relacionados	41
2.3. Considerações sobre o Capítulo	44
CAPÍTULO 3 – ARTEFATOS QUE APOIAM A COMUNICAÇÃO NO DESENVOLVIMENTO DE SOFTWARE	45
3.1. Introdução	45
3.2. Mapeamento Sistemático sobre Artefatos que Apoiam a Comunicação no Desenvolvimento de Software	46
3.2.1 Planejamento do Mapeamento Sistemático	46
3.2.1.1 Estratégia de Busca dos Estudos Primários	47
3.2.1.2 Critérios de Seleção de Artigos	49
3.2.1.3 Procedimentos para a Seleção de Artigos	49
3.2.2 Execução do Mapeamento Sistemático	50
3.2.3 Resultados do Mapeamento Sistemático	51
3.2.4 Discussão dos Resultados do Mapeamento Sistemático	63
3.3. Considerações sobre o Capítulo	64
CAPÍTULO 4 - ESTUDOS PARA A COMPREENSÃO DO PROBLEMA	66
4.1 Introdução	66
4.2 Investigando a Comunicação via Artefato de uma Equipe de Projeto	67
4.2.1 Atividades de Etapa de Análise da HCDP na Sprint 1	69
4.2.2 Construção dos Artefatos de Software na Sprint 2	69
4.2.3 Construção dos Artefatos de Software na Sprint 3	71
4.2.4 Limitações do Estudo	73
4.2.5 Discussão dos Resultados	73
4.3 Investigando a Comunicação de Produtores de Casos de Uso para Desenvolvedores	74

4.3.1 Planejamento do Estudo	76
4.3.2 Execução do Estudo.....	78
4.3.3 Resultados do Estudo	78
4.3.3.1 <i>Análise do Casos de Uso</i>	78
4.3.3.2 <i>Análise dos Mockups</i>	79
4.3.3.3 <i>Análise da Comunicação Via Artefato</i>	81
4.3.4 Limitações do Estudo	85
4.3.5 Discussão dos Resultados.....	86
4.4 Investigando a Comunicação de Produtores e Consumidores de Casos de Uso com o Apoio do Ferramenta SigniFYIng Message	87
4.4.1 Planejamento do Estudo	88
4.4.2 Execução do Estudo.....	88
4.4.3 Resultados do Estudo	89
4.4.3.1 <i>Máximas de Grice Adaptadas para a Análise de Rupturas de Comunicação em Casos de Uso</i> 90	
4.4.3.2 <i>Princípio Cooperativo Estabelecido a partir da Especificação de Casos de Uso</i>	92
4.4.3.3 <i>Princípio Cooperativo não Estabelecido a partir da Especificação de Casos de Uso</i> 93	
4.4.3.4 <i>Consequências Observadas</i>	94
4.4.4 Limitações do Estudo	95
4.4.5 Discussão dos Resultados.....	96
4.5 Considerações sobre o Capítulo	97
CAPÍTULO 5 – DIRETIVAS DE COMUNICABILIDADE	98
5.1. Introdução	98
5.2. Proposta Inicial das Diretivas de Comunicabilidade	100
5.3. Primeiro Estudo Experimental: Análise da Proposta das DCs	102
5.3.1 Planejamento do Estudo	103
5.3.2 Execução do Estudo.....	103
5.3.3 Resultados do Estudo	104
5.3.3.1 <i>Análise do Diagrama de Classes</i>	105
5.3.3.2 <i>Análise do Diagrama BPMN</i>	107
5.3.3.3 <i>Análise do Diagrama IFML</i>	108
5.3.3.4 <i>Análise do Uso das Diretivas de Comunicabilidade</i>	109
5.3.4 Limitações do Estudo	110
5.3.5 Discussão dos Resultados.....	111
5.4. Melhorias na Versão das Diretivas de Comunicabilidade	112
5.5. Segundo Estudo Experimental: Análise da Percepção de Profissionais sobre o Uso das DCs	117
5.5.1 Planejamento do Estudo	118
5.5.2 Execução do Estudo.....	119
5.5.3 Resultados do Estudo	120
5.5.4 Limitações do Estudo	122
5.5.5 Discussão dos Resultados.....	122
5.6. Terceiro Estudo Experimental: Análise do Consumo de Artefatos Produzidos com as DCs	122

5.6.1 Resultados do Estudo	124
5.6.2 Limitações do Estudo	126
5.6.3 Discussão dos Resultados.....	126
5.7. Considerações sobre o Capítulo.....	127
CAPÍTULO 6 – ESTUDOS NA INDÚSTRIA COM AS DIRETIVAS DE COMUNICABILIDADE	128
6.1 Introdução	128
6.2 Estudo 1 – Uso das DCs durante o Desenvolvimento de Software.....	131
6.2.1 Planejamento e Execução do Estudo	132
6.2.2 Resultados da Equipe 1.....	134
6.2.2.1 <i>Resultado da Comunicação via Artefato da Equipe 1.....</i>	<i>135</i>
6.2.2.2 <i>Resultado dos Benefícios das DCs na Comunicação via Artefato da Equipe 1.....</i>	<i>137</i>
6.2.3 Resultados da Equipe 2.....	140
6.2.3.1 <i>Resultado da Comunicação via Artefato da Equipe 2.....</i>	<i>140</i>
6.2.3.2 <i>Resultado dos Benefícios das DCs na Comunicação via Artefato da Equipe 2.....</i>	<i>143</i>
6.2.4 Percepções sobre o Uso das DCs.....	145
6.2.5 Discussão dos Resultados.....	149
6.3 Estudo 2 – Apoio das DCs para o Diagnóstico de Artefatos	150
6.3.1 Planejamento e Execução do Estudo	151
6.3.2 Resultados da Equipe 3.....	152
6.3.2.1 <i>Resultado da Comunicação via Artefato da Equipe 3.....</i>	<i>152</i>
6.3.2.2 <i>Resultado do Diagnóstico dos Artefatos com o Apoio das DCs.....</i>	<i>155</i>
6.3.3 Resultados da Equipe 4.....	155
6.3.3.1 <i>Resultado da Comunicação via Artefato da Equipe 4.....</i>	<i>155</i>
6.3.3.2 <i>Resultado do Diagnóstico dos Artefatos com o Apoio das DCs.....</i>	<i>158</i>
6.3.4 Resultados da Equipe 5.....	158
6.3.4.1 <i>Resultado da Comunicação via Artefato da Equipe 5.....</i>	<i>158</i>
6.3.4.2 <i>Resultado do Diagnóstico dos Artefatos com o Apoio das DCs.....</i>	<i>161</i>
6.3.5 Percepções sobre o Uso das DCs no Diagnóstico de Artefatos	162
6.3.6 Discussão dos Resultados.....	164
6.4 Limitações dos Estudos	164
6.5 Considerações sobre o Capítulo.....	165
CAPÍTULO 7 –AVALIAÇÃO DA ACEITAÇÃO DAS DIRETIVAS DE COMUNICABILIDADE E ANÁLISE DOS FATORES QUE IMPACTAM A COMUNICAÇÃO	166
7.1 Introdução	166
7.2 Avaliação da Aceitação das Diretivas de Comunicabilidade.....	166
7.3 Avaliação dos Fatores que Impactam na Comunicação Via Artefato	169
7.4 Considerações sobre o Capítulo.....	171
CAPÍTULO 8 – CONTRIBUIÇÕES DAS DIRETIVAS DE COMUNICABILIDADE PARA ASPECTOS RELACIONADOS À COMUNICAÇÃO VIA ARTEFATO	172
8.1 Introdução	172
8.2 Contribuições das DCs para a Compreensão de Modelos BPMN.....	172
8.2.1 Estudo Experimental com a técnica e-VOL BPMN.....	176
8.2.2 Resultados do Estudo	177

8.2.3 Limitações do Estudo	179
8.3 Contribuições das DCs para o Desenvolvimento da Família de Técnicas ComD2	180
8.3.1 Estudo Experimental com as técnicas ComD2.....	182
8.3.2 Resultados do Estudo	183
8.3.3 Limitações do Estudo	186
8.4 Contribuições para o Ensino de Modelagem com foco na Comunicação	186
8.4.1 Resultados do Estudo	187
8.4.2 Limitações do Estudo	191
8.5 Considerações sobre o Capítulo	191
CAPÍTULO 9 – CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS	193
9.1 Considerações sobre o Trabalho	193
9.2 Perspectivas Futuras.....	195
REFERÊNCIAS	197
APÊNDICE A – ARTIGOS IDENTIFICADOS NO MAPEAMENTO SISTEMÁTICO	208
APÊNDICE B – MATERIAL DE APOIO UTILIZADO NOS ETUDOS	209
APÊNDICE C – DIRETIVAS DE COMUNICABILIDADE	221
APÊNDICE D – QUESTIONÁRIOS PARA AVALIAÇÃO DOS FATORES QUE IMPACTAM NA COMUNICAÇÃO VIA ARTEFATO	225
APÊNDICE E – DADOS ANALISADOS NOS ESTUDOS	227
APÊNDICE F – DIRETIVAS DE COMUNICABILIDADE ADAPTADAS PARA O DIAGNÓSTICO DE COMUNICABILIDADE DE ARTEFATOS	235

CAPÍTULO 1 – INTRODUÇÃO

Este capítulo apresenta a contextualização desta pesquisa de doutorado. Além disso, a motivação, a definição do problema, os objetivos de pesquisa, o perfil da pesquisadora principal, a metodologia seguida e as principais contribuições pretendidas são apresentadas.

1.1. Contexto

No desenvolvimento de software, os profissionais precisam entender o problema para o qual um software foi requisitado a fim de desenvolver soluções em diferentes perspectivas para o seu desenvolvimento. Estas soluções são elaboradas em diferentes artefatos para representar a estrutura e arquitetura do sistema (Chaudron *et al.*, 2018), como os diferentes diagramas da *Unified Modeling Language* (UML) (OMG, 2015), além dos artefatos que representam a forma como os usuários podem interagir com o sistema, como modelos de interação, mockups e protótipos (Garcia *et al.*, 2017). **Por despreverem as soluções propostas para o desenvolvimento de software, estes artefatos são utilizados na comunicação de equipes** (Petre, 2013).

A comunicação é considerada como um fator importante no desenvolvimento de software, pois as falhas de comunicação são uma das causas de baixa produtividade e falhas de software (Käfer, 2017). No que se refere à comunicação humana, esta segue a máxima geral de que as falas devem ser projetadas especificamente para os seus receptores e para a ocasião em que são emitidas (Barbosa e Silva, 2010). Nesta linha, Jakobson (1960) estrutura um espaço de comunicação em termos de contexto, emissor, receptor, mensagem, código e canal, da seguinte maneira: “*Um emissor transmite uma mensagem a um receptor através de um canal. A mensagem é expressa em um código e se refere a um contexto*”. Para que a comunicação seja bem-sucedida, o emissor deve escolher cuidadosamente uma expressão para o conteúdo que deseja comunicar, utilizando um código que o receptor seja capaz de interpretar.

Herbsleb and Mockus (2003) definem dois tipos de comunicação: a comunicação informal e comunicação formal. A comunicação informal está relacionada às conversações entre os diferentes membros da equipe em uma organização de software, a

qual também pode ser apoiada pelo uso de telefone ou videoconferência (Henttonen e Blomqvist 2005). A comunicação formal refere-se à comunicação formal da documentação, como a especificação de documentos (Pikkarainen *et al.*, 2008). Nesse sentido, esta pesquisa está relacionada com a comunicação formal, pois diferentes artefatos são especificados para apoiar a comunicação da equipe. Em relação à comunicação via artefato, **esta pesquisa a define como ‘a compreensão recíproca entre os emissores e receptores sobre o conteúdo da mensagem transmitida’**. Os emissores são chamados nesta pesquisa de produtores do artefato e os receptores são chamados de consumidores do artefato.

No que se refere à comunicação via artefato, essa perspectiva pode ser alinhada à teoria da Engenharia Semiótica (de Souza 2005, de Souza *et al.*, 2016), a qual investiga diferentes formas de comunicação em tempo de desenvolvimento e uso de software. Abdelzad *et al.* (2016) explorou as contribuições da Engenharia Semiótica para a Engenharia de Software, relatando desafios relacionados à comunicação entre produtores e consumidores de modelos UML, pois tal comunicação “não foi bem definida”. Portanto, é necessário entender como ocorre esse tipo de comunicação e a Engenharia Semiótica pode contribuir para isso. Com isso, pode ser possível desenvolver suporte para apoiar a comunicação eficaz entre produtores e consumidores de artefatos.

1.2.Motivação

De acordo com Diel *et al.* (2016), existem diferentes vias de comunicação no desenvolvimento de software, como discussões face a face em equipes co-localizadas e distribuídas. Além disso, existem diversos suportes para apoiar a comunicação, como listas de discussão, fóruns e outros (Käfer, 2017). Neste contexto, como já exposto, os artefatos desenvolvidos nas etapas iniciais do processo de desenvolvimento de software comunicam a solução tal desenvolvimento (Petre, 2013).

A Figura 1.1 representa um cenário de comunicação entre produtores e consumidores a partir de artefatos. No elemento A, da Figura 1.1, um produtor está elaborando diagrama de classes e mockups para um melhor entendimento da equipe sobre o que deve ser implementado. Quando o produtor destes dois artefatos considerar que a solução concebida nestes é a forma como o sistema deve ser desenvolvido, ele deve “emitir” estes artefatos para os demais membros da equipe de desenvolvimento, como

mostra o elemento B da Figura 1.1. Então os demais membros da equipe podem consumir as informações em tais artefatos, tornando-se um dos meios de comunicação na equipe de desenvolvimento, como ilustra o elemento C da Figura 1.1.

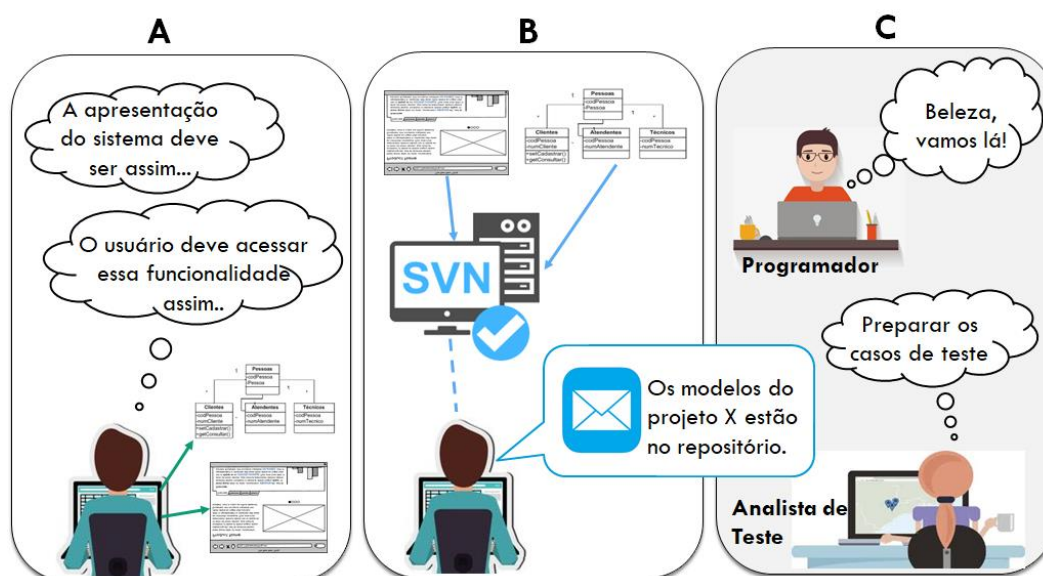


Figura 1.1: Cenário dos Artefatos de Software como um dos Apoios à Comunicação de Equipes.

Sobre a comunicação apoiada pelo uso de artefatos, alguns trabalhos destacam sua importância para equipes co-localizadas (Voigt, 2017) e distribuídas (Bordin e De Angeli, 2016). Este tipo de comunicação também tem sido discutida por equipes que adotam metodologias ágeis, fazendo com que os profissionais reflitam sobre a importância da documentação (Hess *et al.*, 2017) (Vivian *et al.*, 2018).

A comunicação humana com apoio dos artefatos está sujeita a interpretações incompatíveis, causando falhas de comunicação. Nesta tese, as falhas de comunicação a partir de artefatos são definidas como interpretações dos consumidores a partir de artefatos de forma incompatível da intenção dos produtores ao comunicarem uma solução para o desenvolvimento de software.

No contexto dos produtores de software não escolherem cuidadosamente uma expressão para o conteúdo que desejam comunicar, Lange e Chaudron (2006) apresentam uma pesquisa que investigou os efeitos de defeitos em diagramas da UML em relação a diferentes interpretações. Eles realizaram dois experimentos controlados com um grande grupo de estudantes e profissionais da indústria. As duas principais contribuições deste

trabalho são as investigações sobre detecção de defeitos e diferentes interpretações causadas por defeitos não detectados. Schoonewille *et al.* (2011) apresentam uma contribuição relacionada aos aspectos cognitivos na compreensão de documentação de projeto de software. Eles investigaram, em um estudo, a capacidade dos participantes em extrair informações a partir diagramas e textos (gramaticalmente e sintaticamente corretos). Com essa pesquisa, percebeu-se que a auto-avaliação pode ser problemática, pois foi observado que os desenvolvedores ficaram satisfeitos em ‘preencher’ informações ausentes da documentação, sem necessariamente ter o mesmo entendimento dos produtores da documentação. Isso pode causar interpretações incorretas em relação ao software. Os trabalhos de Lange e Chaudron (2006) e Schoonewille *et al.* (2011) evidenciam que produtores de software devem refletir sobre a forma como o conteúdo informacional é expresso para que os consumidores tenham o entendimento comum.

Falhas de comunicação podem introduzir informações incorretas no software, tornando-se defeitos de software (de Mello *et al.*, 2014) e/ou falhas de software (Käfer, 2017), gerando retrabalho para a equipe na correção destes. Existem trabalhos na literatura que investigam falhas de comunicação causadas por diferenças culturais, barreiras linguísticas (Alzoubi *et al.*, 2016), e outras perspectivas (Storey *et al.*, 2017). Porém, **pouco se sabe sobre falhas de comunicação a partir de artefatos.**

A comunicação eficaz entre produtores e consumidores de artefatos de software pode auxiliar na redução de problemas advindos de falhas de comunicação, como a redução de defeitos no software que são causados por tais falhas. Isso pode poupar recursos durante o desenvolvimento de software, contribuindo assim para redução de custos no processo de desenvolvimento. Por esta razão, **é importante apoiar a comunicação eficaz entre produtores e consumidores a partir de artefatos.**

1.3. Definição do Problema

O problema abordado nesta tese é um *wicked problem*, para qual os pesquisadores podem buscar soluções que ‘aproximem’ as extremidades de uma lacuna difícil de contornar. *Wicked problem* é o termo formulado por Rittel e Webber (1973) para definir problemas extremamente complexos, de escala e escopo indeterminados. Por ainda não existir um consenso quanto a sua tradução em português, adotou-se o termo na língua inglesa, que literalmente quer dizer “problema malvado (ou perverso)”. **Portanto,**

a solução desenvolvida nesta tese está relacionada à tentativa de mitigar falhas de comunicação entre produtores e consumidores. Assim, esta tese buscou responder a questão de pesquisa:

Como apoiar a comunicação efetiva de produtores e consumidores de artefatos desenvolvidos nas etapas iniciais do desenvolvimento de software?

Com o objetivo de responder essa questão, esta pesquisa foi encaminhada para o desenvolvimento de uma proposta que tem por objetivo apoiar a comunicação entre produtores e consumidores de artefatos desenvolvidos nas etapas iniciais do processo de desenvolvimento através da melhoria na comunicabilidade de artefatos. A comunicabilidade de artefatos se refere à capacidade que eles têm de efetuar o processo de comunicação entre pessoas, ou de serem usados como instrumentos para realizar parte significativa deste processo (Ferreira, 2015). Ressalta-se que esta tese não foca apenas na transmissão de dados de um artefato para outros artefatos de software, como os trabalhos relacionados às abordagens *model-driven* (dirigidas a modelo), mas na busca do entendimento mútuo entre produtores e consumidores de artefatos de software.

1.4. Objetivos de Pesquisa

O objetivo principal deste trabalho consiste em apoiar a comunicação entre produtores e consumidores de artefatos desenvolvidos nas etapas iniciais do processo de desenvolvimento. Os objetivos específicos desta pesquisa consistiram em:

- Fornecer evidências sobre o estado da arte da comunicação via artefato para compreender os trabalhos na literatura que apoiam este tipo de comunicação;
- Fornecer evidências experimentais acerca da comunicação apoiada por artefatos.
- Criar uma proposta e demonstrar que esta pode nortear os produtores sobre os fatores que podem impactar este tipo de comunicação e apoiar a melhoria da comunicabilidade de artefatos.

O propósito final é que a proposta possa ser empregada em contextos de desenvolvimento de software em que equipes utilizem artefatos como meio de

comunicação, sejam estas equipes co-localizadas, distribuídas, ou que adotem metodologias ágeis. No aspecto de melhoria da comunicabilidade, a proposta visa apoiar o entendimento mútuo entre produtores e consumidores. Por exemplo, a intenção de modelagem proposta por um analista de sistemas em um diagrama de casos de uso pode ser compreendida de maneira diferente pelos desenvolvedores do sistema. Por mais que os desenvolvedores conheçam a notação de modelagem, a forma como a modelagem é expressa pelo analista de sistemas pode afetar o entendimento mútuo entre estes profissionais, causando a introdução de informações incorretas no software.

1.5.Perfil da Pesquisadora Principal

Sendo este trabalho mais próximo da pesquisa qualitativa, a qual é fortemente baseada na análise e interpretação do pesquisador que a realiza, é importante apresentar o perfil da principal pesquisadora deste trabalho. Por isso, peço licença ao leitor por escrever na primeira pessoa do singular:

“Eu me chamo Adriana Lopes Damian e sou a pesquisadora principal deste trabalho. Formada como bacharel em Ciência da Computação pela UFAM, comecei a fazer parte do mercado de trabalho durante a graduação. Aprendendo a prática do desenvolvimento de software, sempre tive questionamentos em relação às informações que são expressas em modelos utilizados nos diferentes projetos em que atuei como Analista de Teste, os quais auxiliavam na comunicação das equipes. Quando ingressei no mestrado, eu tinha uma visão de que o uso de determinados modelos, como modelos de interação, combinados ao uso de técnicas de inspeção, poderiam ajudar produtores de modelos a refletir sobre as funcionalidades do software e a melhorar a comunicação das equipes a partir destes, sendo este o tema da minha dissertação. Porém, pesquisadores e colaboradores da minha dissertação, que tive a honra de encontrar, sempre me perguntavam o quanto um modelo de interação poderia ser compreendido mesmo que determinados defeitos estivessem presentes nestes. Esses questionamentos, de uma certa forma provocadores, trouxeram uma visão de que era necessário ir além neste caminho, uma vez que este é um problema que afeta profissionais de diferentes equipes de desenvolvimento de software. Portanto, minha experiência profissional e acadêmica pode influenciar minha visão e postura como pesquisadora, por isso decidi me apresentar

e contar um pouco da minha história. Isso poderá fornecer informações para que o leitor entenda, talvez, algum viés que afeta a condução e as conclusões desta pesquisa”.

1.6. Metodologia Utilizada na Pesquisa

A metodologia utilizada nesta pesquisa tem por base o ciclo de *Design Science Research* (DSR). DSR estabelece um processo sistemático no desenvolvimento de artefatos para a resolução de um problema a partir de conhecimentos e conjecturas sobre tal contexto (Wieringa, 2014). O ciclo de DSR inicia com a investigação de um problema. Então são especificados artefatos como soluções, que são avaliados para o contexto do problema. O resultado é então avaliado, podendo dar início em uma nova volta no ciclo de DSR. Os principais elementos que compõem o DSR foram utilizados nesta pesquisa, com base nos trabalhos de Wieringa (2014), Hevner e Chatterjee (2010) e Pimentel (2017). Tais elementos são apresentados e discutidos em relação ao seu uso nesta pesquisa a seguir:

Contexto e Problema - O DSR começa com a definição do problema e/ou oportunidades de melhoria em um determinado contexto. A partir disso, podem ser estabelecidas questões de pesquisa que auxiliam na identificação de artefatos como solução ou mitigação para tal problema.

Artefatos - Depois da compreensão sobre o problema, o pesquisador propõe um ou mais artefatos para a resolução ou mitigação do problema. Os artefatos geralmente produzidos em DSR são relacionados a conceitos, modelos e métodos. O artefato é avaliado, verificando se este produz os efeitos desejados.

Conjecturas Teóricas - O uso dos artefatos permite corroborar ou questionar a validade das conjecturas teóricas. Nesta pesquisa, as conjecturas teóricas referem-se ao uso dos artefatos para: **(i) apoio aos produtores na reflexão da comunicação a partir dos artefatos;** **(ii) melhoria da comunicabilidade de artefatos de software.** Essas conjecturas teóricas foram elaboradas para o problema investigado nesta tese a partir de teorias que investigam a comunicação (Jakobson, 1960; Grice, 1975) e a comunicação durante o desenvolvimento de software (de Souza *et al.*, 2016).

Em relação ao uso do artefato proposto para analisar *se o problema foi resolvido e se as conjecturas teóricas são válidas*, houve a condução de diferentes estudos

experimentais. Em tais estudos, foram investigados diferentes artefatos que são elaborados nas etapas iniciais do processo de desenvolvimento e que apoiam a comunicação via artefato, como os diferentes diagramas da UML e mockups. Isso ocorreu devido ao conhecimento dos participantes sobre o uso de diferentes notações utilizadas no desenvolvimento destes artefatos. Os estudos realizados são classificados em:

Estudos *in vitro* - estudos realizados em um ambiente controlado, com a participação de pessoas que atuam como representantes da população de interesse. (Travassos e Barros, 2003)

Estudos *in vivo* - estudos que envolvem pessoas em seu próprio ambiente de trabalho em condições realistas (Travassos e Barros, 2003).

O método *Design Science Research Methodology* (DSRM) proposto por Peffers *et al.* (2007) foi adotado neste trabalho para demonstrar as etapas realizadas durante a execução desta pesquisa, como mostra a Figura 1.2. Esta metodologia é dividida em seis etapas, sendo:

1. Identificação do problema - nessa etapa o pesquisador define o problema e justifica a importância da pesquisa, considerando sua relevância do problema que está sendo investigado.

2. Objetivos da solução - nesta etapa o pesquisador define os resultados esperados para o problema que está sendo investigado, ou seja, o que artefato pode promover.

3. Projeto e desenvolvimento - o pesquisador utiliza o conhecimento teórico existente, a fim de propor artefatos que suportem a solução do problema. São definidos os requisitos desejados dos artefatos.

4. Demonstração - uso do artefato para solucionar o problema em questão. Essa etapa pode ser desenvolvida por meio de experimentação.

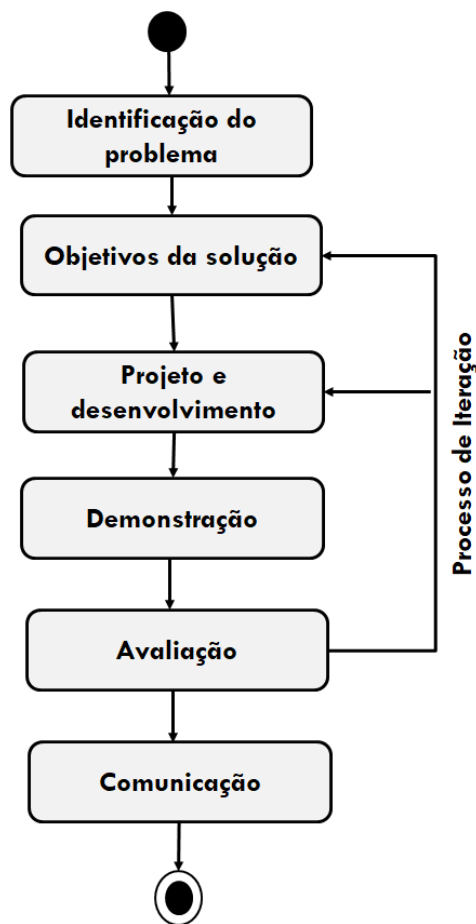


Figura 1.2: Metodologia Utilizada na Pesquisa (Peffers et al. (2007).

5. Avaliação - o pesquisador deve comparar os resultados obtidos com os objetivos da solução definidos na segunda etapa do método. Caso o resultado encontrado não seja o esperado, poderá retornar à etapa de projeto e desenvolvimento a fim de desenvolver um novo artefato.

6. Comunicação - o pesquisador apresenta o problema que foi estudado e sua importância, a qual pode ser realizada através de publicações acadêmicas.

Na etapa ‘identificação do problema’, um mapeamento sistemático da literatura¹ foi realizado para compreender o estado da arte dos artefatos como apoio à comunicação (apresentado no Capítulo 3). Além disso, foram conduzidos três estudos experimentais para a compreensão deste tipo de comunicação (apresentados no Capítulo 4).

¹ De acordo com Kitchenham *et al.* (2010), um mapeamento sistemático da literatura pode fornecer benefícios significativos no estabelecimento de linhas de base para pesquisas futuras.

Na etapa ‘objetivos da solução’, foram obtidos *insights* para a proposta de um artefato com o objetivo de promover a reflexão dos produtores de artefatos sobre o seu conteúdo para que ocorresse a compreensão recíproca entre eles e os demais membros de uma equipe de software. Tal *insights* foram obtidos a partir dos resultados do mapeamento sistemático e dos estudos experimentais realizados na etapa de ‘identificação do problema’.

Na etapa de ‘projeto e desenvolvimento’, as Diretivas de Comunicabilidade (DCs) foram propostas, tendo sua fundamentação principal na teoria da Engenharia Semiótica (de Souza 2005, de Souza *et al.*, 2016). As DCs foram propostas para melhorar a comunicabilidade de artefatos de software com o objetivo de apoiar a comunicação eficaz via artefato. Tais diretivas foram propostas para serem aplicadas em qualquer artefato de software proposto nas fases iniciais do processo de desenvolvimento, mas nesta tese são apresentados DCs elaboradas apenas para alguns destes artefatos de software. Com isso, profissionais podem ver exemplos do suporte das diretivas para a melhoria da intenção de comunicação por parte de quem produz artefatos.

Na etapa de ‘demonstração’, primeira iteração na metodologia, como mostra a Figura 1.3, foi realizado um primeiro estudo experimental com a proposta inicial das DCs com o objetivo de reduzir os riscos de falhas de comunicação de diferentes artefatos (Lopes *et al.*, 2019c). Os resultados deste estudo mostraram que estas promoveram a redução dos riscos de falhas de comunicação, melhorando a comunicabilidade de artefatos. Após esse estudo, oportunidades de melhorias foram identificadas. Com isso, a partir da etapa ‘avaliação’ retornou-se para a etapa ‘projeto e desenvolvimento’ com o objetivo de melhorar a proposta das DCs, considerada como uma segunda iteração na metodologia, como mostra a Figura 1.3.

Juntamente com estas melhorias realizadas nas DCs, foi realizada uma proposta de análise dos fatores que podem impactar neste tipo de comunicação. A nova versão das DCs foi avaliada na etapa de ‘demonstração’ em um segundo estudo experimental. Os resultados obtidos na avaliação desta segunda proposta mostraram que estas continuam promovendo apoio para a redução de riscos de falhas de comunicação. Assim, as DCs foram consideradas válidas para serem avaliadas em relação à produção e consumo de artefatos com o objetivo de avaliar a comunicação entre produtores e consumidores.

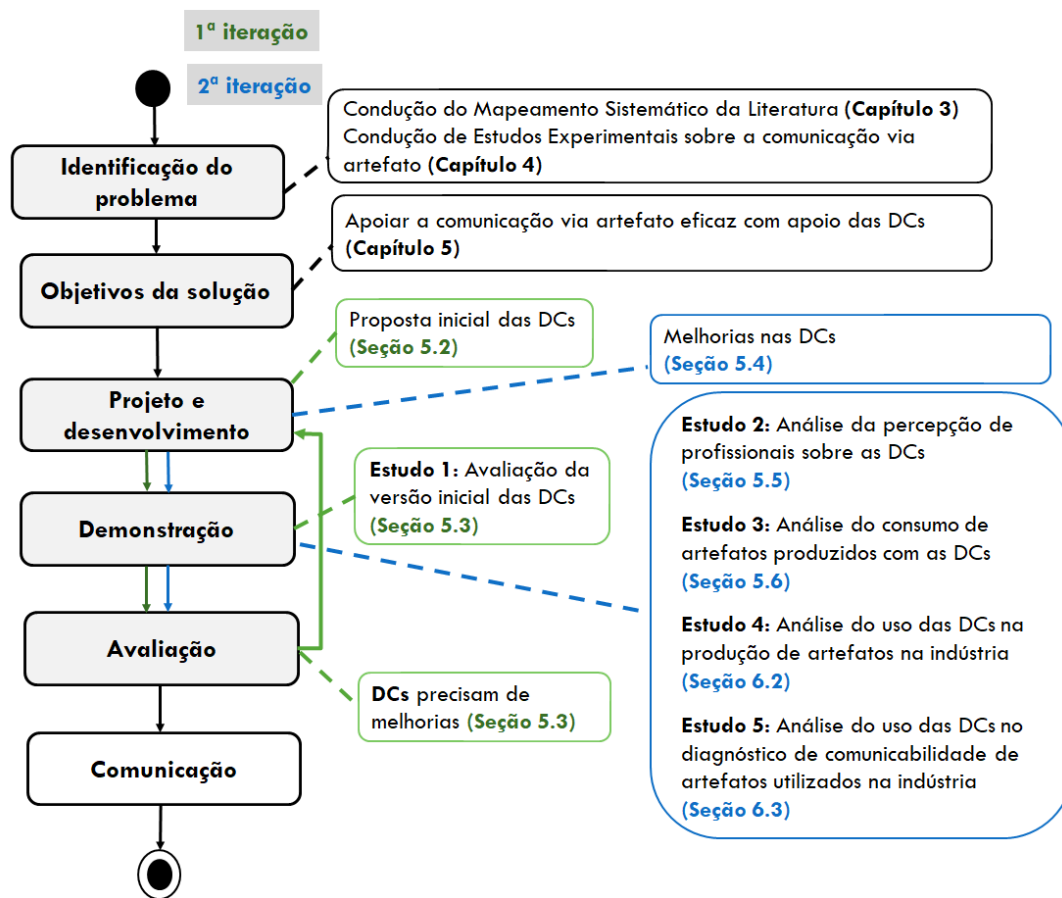


Figura 1.3: Execução do Ciclo de DSR – Primeira e Segunda Iteração.

Ainda na etapa de ‘demonstração’, foram realizados mais três estudos. Um terceiro estudo foi realizado com o objetivo de analisar a melhoria da comunicação entre produtores e consumidores com o apoio das DCs (Lopes *et al.*, 2019a). Foram produzidos artefatos por diferentes profissionais, os quais possuíam o mesmo nível de experiência em modelagem de software. Um profissional utilizou as DCs na produção dos artefatos e o outro não. Após isso, foi analisado se os consumidores destes artefatos compreendiam a intenção de seus produtores. Os resultados mostraram que os consumidores dos artefatos produzidos com as DCs promoveram uma menor quantidade de falhas de comunicação, comparados aos consumidores dos artefatos produzidos sem as DCs.

As DCs foram avaliadas na indústria de software em dois estudos, sendo estes o quarto e quinto estudos experimentais realizados com as diretivas. No quarto estudo, foi analisado o emprego das DCs nos artefatos produzidos nas etapas iniciais do desenvolvimento de software de duas equipes. Tais artefatos foram consumidos pelos

demais membros da equipe de desenvolvimento. Os resultados mostraram que as DCs apoiaram a reflexão dos produtores sobre sua comunicação com os demais membros da equipe, promovendo a comunicação efetiva via artefato. No quinto estudo, as DCs foram empregadas como diagnóstico de comunicabilidade dos artefatos de três equipes de desenvolvimento. Os resultados mostraram que é possível identificar problemas de comunicabilidade e isso pode ajudar as equipes como lições aprendidas para os próximos projetos. Em ambos os estudos foram analisados os fatores que podem afetar a comunicação via artefato.

1.7. Principais Contribuições

A principal contribuição desta pesquisa consiste na proposta das Diretivas de Comunicabilidade para apoiar a comunicação entre produtores e consumidores de artefatos de software. A fundamentação principal desta proposta se baseia na teoria da Engenharia Semiótica (de Souza 2005, de Souza *et al.*, 2016). Trata-se de uma ferramenta de apoio para a análise de comunicabilidade de artefatos e reflexão dos produtores de artefatos sobre sua comunicação com os demais membros de uma equipe de desenvolvimento. A proposta apresentada nesta tese também contribuiu para outros aspectos relacionados à comunicação via artefato, tais como (mais detalhes são apresentados no Capítulo 8):

- Técnica que apoia a manutenção de diagramas BPMN (*Business Process Model and Notation*) (Campos *et al.*, 2019), chamada e-VOL BPMN, a qual foca na compreensão e modificação dos modelos. As DCs foram utilizadas como base para desenvolver questões que auxiliam na compreensão dos modelos.
- O desenvolvimento de uma família de técnicas que focam na inspeção de defeitos que podem afetar a comunicação entre produtores e consumidores (Lopes *et al.*, 2018b). As DCs foram utilizadas como base para desenvolver itens de verificação para a identificação dos defeitos.
- As DCs foram adaptadas para apoiar o ensino de modelagem de diagramas UML com foco no apoio à comunicação (Lopes *et al.*, 2019b).

1.8. Publicações Resultantes

Esta pesquisa resultou em publicações em conferências e periódicos relacionadas aos resultados desta tese. As referências são listadas a seguir em ordem cronológica:

1. **LOPES, A.**; VALENTIM, N.; FERREIRA, B. M.; ZILSE, R.; CONTE, T. Utilizando Modelagem de Interação para Projetar uma Aplicação Móvel Multiusuário: Um Relato de Experiência. In: Anais do XV Simpósio Brasileiro de Qualidade de Software (SBQS 2016), 2016, 374-388.
2. **LOPES, A.**; CONTE, T.; DE SOUZA, C. S. Analyzing the Use Case Communicability. In: Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems – (IHC 2017), 2017, 1-10 pages.
3. **LOPES, A.**; CAMPOS, U.; CONTE, T. U.; SOUZA., C. S. ComD2: Family of Techniques for Inspecting Defects in Models that Affect Team Communication. In: Proceedings of the International Conference on Software Engineering and Knowledge Engineering (SEKE 2018), 2018, Article 155, 6 pages.
4. **LOPES, A.**; VALENTIM, N.; MORAES, B.; ZILSE, R.; CONTE, T. Applying User-Centered Techniques to Analyze and Design a Mobile Application. Journal Of Software Engineering Research And Development, v. 6 (5), 2018.
5. **LOPES, A.**; OLIVEIRA, E.; CONTE, T. U.; SOUZA., C. S. Directives of Communicability: Towards Better Communication Through Software Models. In: Proceedings of the 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE '19), 2019, 45-48.
6. **LOPES, A.**; OLIVEIRA, E.; CONTE, T. U.; SOUZA., C. S. Directives of Communicability: Teaching Students How to Improve Communication Through Software Modeling. In: Proceedings of 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSECompanion), 2019, 306-307.
7. **LOPES, A.**; CONTE, T. U.; SOUZA., C. S. Reducing the Risks of Communication Failures Through Software Models. In: Proceedings of XVIII Brazilian Symposium on Human Factors in Computing Systems (IHC 2019), 2019, 10 pages.
8. CAMPOS, U.; **Damian, A. L.**; OLIVEIRA, E.; GADELHA, B.; CONTE, T. e-VOL BPMN: A Technique to Support the Evolution and Learning of BPMN Diagrams. IET Software, v. 1, p. 1, 2019.

Além disso, outros artigos foram publicados em colaboração com outras pesquisas e são listados a seguir:

9. **LOPES, A.**; MARQUES, A. B. S.; VALENTIM, N.; CONTE, T. Práticas da Indústria Aplicadas em Sala de Aula Para Apoiar o Ensino de Modelagem de Interação. In: 8th Workshop on HCI Education (WEIHC) - 16th Brazilian Symposium on Human Factors in Computing Systems, 2017.
10. NASCIMENTO, I.; SILVA, W.; **LOPES, A.**; CABREJOS, L. J. E. R.; GADELHA, B.; OLIVEIRA, E.; CONTE, T. An Empirical Study to Evaluate the Feasibility of a UX and Usability Inspection Technique for Mobile Applications. In: Proceedings of the 28th International Conference on Software Engineering & Knowledge Engineering (SEKE 2016), 2016, 595-599.
11. VALENTIM, N. M. C.; OLIVEIRA, E.; VINCENZI, A. M. R.; **LOPES, A.**; CONTE, T.; MALDONADO, J. C. An Acceptance Empirical Assessment of Open Source Test Tools. In: Proceedings of the 19th International Conference on Enterprise Information Systems, 2017, 379-386.
12. SOUSA, L.; OLIVEIRA, E.; LUCENA, C.; OLIVEIRA, R.; GARCIA, A.; LEE, J.; CONTE, T.; OIZUMI, W.; DE MELLO, R.; **LOPES, A.**; VALENTIM, N. How Do Software Developers Identify Design Problems? In: Proceedings of the 31st Brazilian Symposium on Software Engineering (SBES'17), 2017. p. 54.
13. OLIVEIRA, R.; SOUSA, L.; DE MELLO, R.; VALENTIM, N. **LOPES, A.**; CONTE, T.; GARCIA, A.; OLIVEIRA, E.; LUCENA, C. Collaborative Identification of Code Smells: A Multi-Case Study. In: IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP), 2017, 33-42.
14. QUEIROZ, R.; MARQUES, A. B. S.; **LOPES, A.**; OLIVEIRA, E.; CONTE, T. Evaluating Usability of IFML Models: How Usability is Perceived and Propagated. In: Proceedings of the 7th Brazilian Symposium on Human Factors in Computing Systems, 2018.
15. CAMPOS, U.; **LOPES, A.**; BARBOSA, S. D. J.; CONTE, T. U. Empirical Studies Concerning the Maintenance of BPMN Diagrams: A Systematic Mapping Study. In: Proceedings of the 31st International Conference on Software Engineering & Knowledge Engineering (SEKE 2019), 2019, 325-330.
16. **Damian, A. L.**; MARQUES, A. B. S.; SILVA, W.; BARBOSA, S. D. J.; CONTE, T. U. Checklist-Based Techniques with Gamification and Traditional Approaches for Inspection of Interaction Models. IET Software, p. 12, 2020.

1.9. Organização

Este trabalho está organizado em mais oito capítulos. A organização do texto deste trabalho segue a seguinte estrutura:

Capítulo 2 – FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS: apresenta as teorias que apoiaram a investigação dos artefatos como apoio à comunicação no desenvolvimento e os trabalhos relacionados.

Capítulo 3 – ARTEFATOS QUE APOIAM A COMUNICAÇÃO NO DESENVOLVIMENTO DE SOFTWARE: apresenta os conceitos relacionados aos artefatos como apoio à comunicação no desenvolvimento de software, incluindo a condução de um mapeamento sistemático da literatura sobre tal tópico.

Capítulo 4 – ESTUDOS PARA A COMPREENSÃO DO PROBLEMA: apresenta diferentes estudos experimentais para uma melhor compreensão da comunicação via artefato.

Capítulo 5 – DIRETIVAS DE COMUNICABILIDADE: apresenta a proposta inicial das diretivas de comunicabilidade e o primeiro estudo experimental. Também apresenta as melhorias realizadas nas diretivas e o segundo experimental realizado. Além disso, é apresentado um terceiro estudo experimental para analisar o apoio das diretivas na comunicação via artefato.

Capítulo 6 – ANÁLISE DO USO DAS DIRETIVAS DE COMUNICABILIDADE NA INDÚSTRIA DE SOFTWARE: apresenta o quarto e quinto estudo realizado com as DCs. No quarto estudo, as DCs foram empregadas durante a produção dos artefatos, proporcionando a comunicação efetiva de produtores e consumidores. No quinto estudo, as DCs foram empregadas para analisar problemas de comunicabilidade em artefatos de software que podem ter causado falhas de comunicação em equipes em projetos de desenvolvimento concluídos.

Capítulo 7 – AVALIAÇÃO DAS DIRETIVAS DE COMUNICABILIDADE PARA A PRODUÇÃO DE ARTEFATOS: apresenta a análise da aceitação das diretivas como apoio à comunicação via artefato. Além disso, é apresentada uma análise dos fatores que devem ser considerados na comunicação via artefato.

Capítulo 8 – CONTRIBUIÇÕES DAS DIRETIVAS DE COMUNICABILIDADE PARA TEMAS RELACIONADOS À COMUNICAÇÃO

A PARTIR DE ARTEFATOS: este capítulo apresenta a contribuição das Diretivas de Comunicabilidade para outros aspectos relacionados à comunicação via artefato.

Capítulo 9 – CONSIDERAÇÕES FINAIS: este capítulo contém as considerações sobre as principais contribuições desta pesquisa, além de fornecer perspectivas futuras.

CAPÍTULO 2 – FUNDAMENTAÇÃO TEÓRICA E TRABALHOS RELACIONADOS

Este capítulo apresenta os conceitos que contribuíram para a investigação da comunicação entre membros de uma equipe de desenvolvimento de software a partir de artefatos. Os principais trabalhos relacionados também são apresentados neste capítulo.

2.1. Fundamentação Teórica

2.1.1 Engenharia Semiótica

Na produção de artefatos de software envolvem-se as intenções de seus produtores sobre como estes podem ser entendidos pelos membros da equipe em relação ao que estes sabem e conhecem sobre o software. Essa perspectiva pode ser alinhada à teoria da Engenharia Semiótica (de Souza 2005, de Souza *et al.*, 2016).

Os softwares na Engenharia Semiótica são considerados como artefatos de *metacomunicação*, ou seja, artefatos que comunicam uma mensagem do designer para os usuários sobre como eles, usuários, podem ou devem se comunicar, por sua vez, com o sistema para fazer o que desejam. O conteúdo da mensagem de *metacomunicação*, ou *metamensagem*, pode ser parafraseado no seguinte *template*:

“Este é o meu entendimento, como designer, de quem você, usuário, é, do que aprendi que você quer ou precisa fazer, de que maneiras prefere fazer, e por quê. Este, portanto, é o sistema que projetei para você, e esta é a forma como você pode ou deve utilizá-lo para alcançar uma gama de objetivos que se encaixam nesta visão”.

A Engenharia Semiótica possui métodos de avaliação propostos para apoiar a comunicação designer-usuário com o objetivo de compreender como a *metamensagem* está sendo recebida pelo usuário. O princípio de categorização das falhas de comunicação apresentado pela Engenharia Semiótica está relacionado à três categorias:

- **Completa** - quando a intenção da comunicação e seu efeito são inconsistentes.
- **Parciais** - quando parte do efeito pretendido da comunicação não é atingido.
- **Temporárias** - na intenção de um ato comunicativo entre usuário e sistema, usuário tem dificuldade momentânea de continuar conversando com o sistema.

A Engenharia Semiótica utiliza o modelo de espaço de comunicação proposto por Jakobson (1960), estruturado em termos de: contexto, emissor, receptor, mensagem, código e canal. “*Um emissor transmite uma mensagem a um receptor através de um canal. A mensagem é expressa em um código e se refere a um contexto*”. Sendo assim, ao projetar sua *metamensagem*, o designer de IHC precisa tomar decisões sobre cada um desses elementos.

Esta teoria ampliou sua perspectiva original de IHC para uma perspectiva da área de Computação Centrada em Humanos - do inglês *Human-Centered Computing* - (HCC) (de Souza *et al.*, 2016). HCC é um campo de pesquisa que integra teorias e pesquisas sobre humanos e domínios de aplicações (Sebe, 2010). A compreensão do comportamento humano ao integrar tecnologias em contextos sociais e culturais também é um dos objetivos de HCC (Jaimes *et al.*, 2007).

Na versão mais recente da teoria da Engenharia Semiótica é apresentada a suíte de ferramentas intituladas SigniFYI (*Signs For Your Interpretation*) (de Souza *et al.*, 2016). Tais ferramentas auxiliam a investigação de significados em um software durante o processo de desenvolvimento e na comunicação entre seus produtores e consumidores.

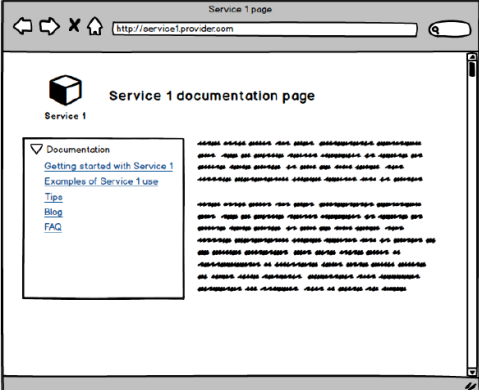
Dentre essas ferramentas, a ferramenta SigniFYI Message pode ser usada de duas maneiras, para capturar os significados em artefatos. Essa ferramenta pode ser aplicada por aqueles que pretendem investigar a comunicação de produtores de artefatos de software de uma maneira geral, como designers de sistemas com os usuários e designers de ferramentas de desenvolvimento de software com desenvolvedores.

O formulário de metacomunicação da ferramenta SigniFYI Message apresenta uma versão operacional do modelo de *metacomunicação*, cuja avaliação pode ser definida em porções da mensagem designer-usuário, como mostra a Figura 2.1. O formulário de metacomunicação detalha: (i) o entendimento do desenvolvedor sobre o usuário; (ii) a intenção e expectativas do designer-desenvolvedor em relação ao modo de uso do sistema; e (iii) o apoio oferecido ao usuário para ressignificar ou redefinir o sistema (ou partes dele) em contextos não previstos pelo designer-desenvolvedor.

I. O que o desenvolvedor sabe sobre o usuário:
o perfil do usuário
o objetivo do usuário
as necessidades do usuário
as preferências do usuário
o contexto do usuário
II. A intenção e expectativa do desenvolvedor sobre:
a descrição do sistema
a funcionalidade do sistema
o modo de usar o sistema
o design do sistema
III. Apoio e suporte do desenvolvedor para:
criar modos alternativos de uso do sistema

Figura 2.1: Formulário de Metacomunicação (adaptado de Souza et al., 2016).

De Mello Brandão *et al.* (2016) apresentam um relato sobre o uso da SigniFYI Message com o formulário de metacomunicação para análise da comunicação de designers, através de um mockup para uma Plataforma como Serviço (PaaS), para usuários que são desenvolvedores de software. A Figura 2.2 apresenta um exemplo do uso da SigniFYI Message para parte destas telas.



I. O que o desenvolvedor sabe sobre o usuário:
Você é um desenvolvedor que precisa combinar idiomas com serviços, APIs e APPs para criar aplicativos ...
II. A intenção e expectativa do desenvolvedor sobre:
Portanto, aqui você pode escolher QUALQUER IDIOMA e combiná-lo com QUALQUER SERVIÇO, API ou APP que você deseja...
III. Apoio e suporte do desenvolvedor para:
... para criar seu novo aplicativo para o qual configurarei automaticamente o ambiente de execução com base nas suas opções de linguagem e serviços de programação

Figura 2.2: Análise de Mockups com o Formulário de Metacomunicação - adaptado de De Mello Brandão *et al.* (2016).

A outra maneira de utilizar a SigniFYI Message é através da estrutura de metacomunicação, que organiza os processos comunicativos entre os envolvidos com o software, sejam humanos, ou sistemas interativos. A Figura 2.3 apresenta a estrutura de metacomunicação com os elementos estruturais relacionados. A estrutura de metacomunicação está fortemente relacionada com o formulário de metacomunicação.

Quem está falando (em nome de quem) ...
{designers, desenvolvedores, sistema, usuário}
... para quem?
{sistema, usuário}
... sobre o quê?
{domínio, tarefa, interação, interface, sistema, usuário, designers, desenvolvedores}
... por quê?
Refere-se à intenção do falante sobre a comunicação
... como?
Refere-se à forma de comunicação do falante
... quando e onde?
Refere-se ao contexto de comunicação do falante

Figura 2.3: Estrutura de Metacomunicação (adaptado de Souza et al., 2016).

De Souza et al. (2016) apresentam um relato sobre o uso da SigniFYI Message para estruturar a comunicação do designer, através de um mockup que representa um sistema mobile que gerencia o agendamento de defesas de mestrado e doutorado, para os usuários. Esse sistema proporciona o ajuste automático de fuso horário quando os membros de uma banca estão verificando sua disponibilidade para uma sessão de defesa, caso um professor esteja no Rio de Janeiro e outro professor em Londres por exemplo.

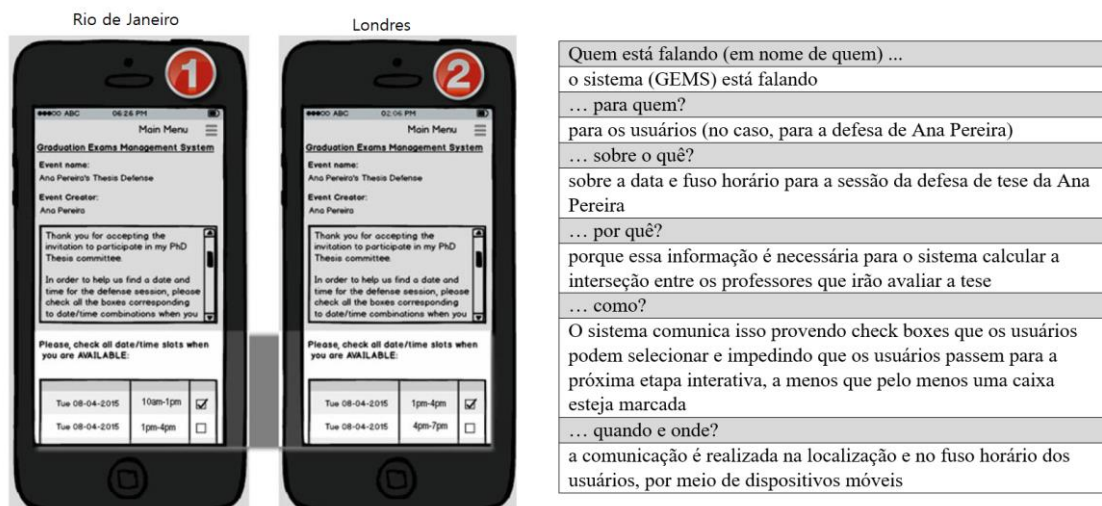


Figura 2.4: Análise de Mockups com o Formulário de Metacomunicação - adaptado de De Souza et al. (2016).

Com o formulário de metacomunicação é possível explicitar a mensagem completa do designer, pois suas intenções são detalhadas. Com a estrutura de metacomunicação pode-se focar mais diretamente na maneira como a mensagem é (re)construída, caso interessante para análises da interpretação dada à mensagem. Através do formulário de metacomunicação e da estrutura de metacomunicação é possível, então, rastrear os

significados inseridos por designers e desenvolvedores ao construir mensagem de metacomunicação para os usuários finais de distintos sistemas interativos.

O conteúdo em um artefato de software pode relacionar o que é “dito” através da descrição de informações explícitas (significado convencional das palavras utilizadas) e o “não dito” através de informações implícitas (intenção de comunicar uma ideia diferente ou para além daquela que é literalmente expressa). Para Grice (1975), a lógica de uma interação social, realizada através de comunicação verbal, depende da observação de um princípio de cooperação recíproca entre os interlocutores. Tal princípio é apresentado na próxima subseção.

2.1.2 Princípio de Cooperação de Grice

De Souza *et al.* (2016) afirmam que o Princípio de Cooperação de Grice (Grice, 1975) auxilia na expressão de características essenciais de uma comunicação efetiva e eficiente. Isto depende da observação de um princípio de cooperação recíproca (Grice, 1975), no qual os falantes reconhecem um objetivo comum que determina como a conversação deve ser conduzida de maneira eficaz. Em sua teoria da comunicação, este princípio cooperativo é estabelecido a partir de quatro máximas, sendo estas:

- **Qualidade** - Tente fazer com que a sua contribuição seja verdadeira. Não diga o que acredita ser falso e não diga algo de que você não tem evidência adequada. Como exemplo de não observância desta máxima, apresenta-se o seguinte diálogo: “(A) – *O que você acha de Brainstorming com a equipe?* (B) – *Acho antidemocrático demais*”. Este exemplo evidencia que a resposta claramente afirma uma inverdade, se o falante supostamente sabe que se trata de uma inverdade cabal, só pode ser interpretada como um ato de ironia. Portanto, a interpretação correta da intenção do falante requer recursos especiais dos ouvintes.
- **Quantidade** - Faça com que sua contribuição seja tão informativa quanto necessário, e não mais que o necessário. Como exemplo de não observância desta máxima, apresenta-se o seguinte diálogo: “(A) – *O que você pensa de personas?* (B) – *Uma persona é uma persona*”. Neste exemplo, (B) responde à pergunta como uma afirmação redundante para implicar que o próprio conceito de persona fala por si e dispensa comentários. A ausência de uma colocação explícita não apenas não

responde integralmente à pergunta feita, como abre caminho para interpretações até inconsistentes umas com as outras.

- **Relação** – Seja relevante, isto é não introduza questões que não vêm ao caso em discussão. Como exemplo de não observância desta máxima, apresenta-se o seguinte diálogo: “(A) – *Você quer acrescentar alguma coisa?* (B) – *Sim, aquele caso de uso da compra de produto*”. Considerando que a equipe está em determinado estágio de projeto e que este caso de uso deveria ser desenvolvido somente no próximo ciclo, se (B) pede para discuti-lo, a única forma de sua intervenção não ser uma não observância da máxima de relevância seria supor que tal caso de uso tem uma relação importante com o assunto que está sendo discutido. Se não tem, a consequência pode ser tempo perdido pela equipe buscando estabelecer uma relação que não existe.
- **Modo** - Seja claro, breve e organizado com sua contribuição. Evite a obscuridade de expressão, ambiguidade. Como exemplo de não observância desta máxima, apresenta-se o seguinte diálogo: “(A) *Este caso de uso é muito polêmico.* (B) *Como assim ‘polêmico’?* (A) *Ele passa por umas estruturas muito complexas para o usuário.* (B) *Quais estruturas?* (A) *Todas as vezes que o usuário vai acabar tendo problemas de usabilidade para interagir com o Sistema*”. A não observância desta máxima, através de contribuições vagas, incompletas ou inapropriadas para o contexto mostra claramente consequências potencialmente desastrosas para o entendimento mútuo na conversação.

A inobservância de uma ou mais destas máximas podem causar falhas na comunicação entre os interlocutores, que no contexto deste trabalho são os membros da equipe de desenvolvimento de software. Além disso, o conceito de implicatura é muito importante para tratar das máximas de Grice, o qual se refere às informações que podem ser inferidas a partir do que é efetivamente dito. O conceito é dividido em duas categorias:

- **Implicatura convencional** – os interlocutores podem inferir o conteúdo implicado pela(s) própria(s) sentença(s) proferida. Por exemplo, a expressão “*João possui uma certificação de projeto de mockups, logo essa parte estará concluída*” leva à inferência de que a certificação de João trará bons resultados e produtividade à equipe, que assim terá plenas condições de concluir a etapa relativa à mockups.

- **Implicatura conversacional** – ao contrário do caso anterior, neste os interlocutores ao interagir de forma cooperativa podem inferir significados implícitos que devem ser verdadeiros a julgar pelas características do discurso naquela situação. Por exemplo, durante uma reunião de várias horas de duração, um participante afirma logo de início que: “*O essencial para todos nós é a usabilidade*”. Nenhum dos demais participantes contesta a afirmação e a conversa evolui para discutir várias alternativas de design. Já no final da reunião, quando se vai decidir qual é a melhor, um dos participantes descarta uma delas dizendo: “*O essencial não foi comprovado*”. Todos os presentes na reunião aceitam o descarte, entendendo que a alternativa em questão não passou em testes ou avaliações de usabilidade.

2.1.3 Comunicação via Artefato e Apropriação da Teoria da Engenharia Semiótica para tal Comunicação

Os artefatos desenvolvidos nas etapas iniciais do processo de desenvolvimento apresentam as soluções propostas para o desenvolvimento de software, como os diferentes diagramas da UML (OMG, 2015). Por descreverem as soluções propostas para o desenvolvimento de software, artefatos são utilizados na comunicação de equipes (Petre, 2013). Em relação ao uso de artefatos de software, Chaudron *et al.* (2018) apresentam três etapas que caracterizam a perspectiva dos estágios de desenvolvimento em que estes são usados:

- **Ideação** - o principal objetivo desta etapa é criar um conceito do sistema a ser desenvolvido. Esta é a etapa mais criativa no projeto, pois requer a exploração, formação e combinação de ideias.
- **Externalização** - O principal objetivo desta etapa é criar uma representação externa do sistema a ser construído, em oposição à interna, que está na mente do projetista. Com isso, é possível promover o entendimento compartilhado em uma equipe.
- **Produção** - Nesta etapa, o sistema é desenvolvido e os artefatos de software apoiam esta implementação.

Sobre a comunicação entre produtores e consumidores de artefatos, de acordo com a Engenharia Semiótica (de Souza 2005), o espaço de comunicação de Jakobson (1960) pode ser adaptado para representar as condições de comunicação, como: o artefato é disponibilizado com o suporte de uma ferramenta (o canal) com informações do domínio

do problema (contexto) para apoiar a comunicação entre seus produtores (os emissores) e consumidores (os receptores). O produtor, em sua mensagem, deve considerar a forma como o conteúdo é expresso (o uso do código) em tais artefatos. Foi criada a Figura 2.5 para caracterizar este tipo de comunicação.

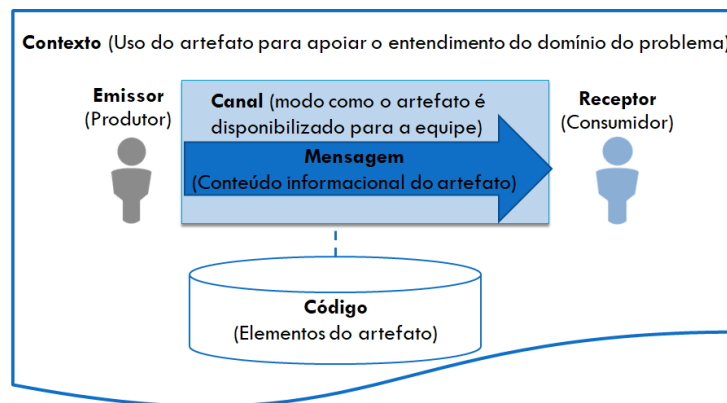


Figura 2.5: Espaço de Comunicação entre Produtores e Consumidores de Artefatos.

Nessa perspectiva, a compreensão recíproca entre os produtores de artefatos de software e consumidores pode ser definida como uma comunicação eficaz entre esses profissionais. Sobre este tipo de comunicação, é importante também compreender a definição de outros conceitos, tais como:

Comunicabilidade de artefatos - capacidade que eles têm de efetuar o processo de comunicação entre pessoas, ou de serem usados como instrumentos para realizar parte significativa deste processo.

Problemas de comunicabilidade nos artefatos de software - refere-se às características e expressões no artefato que podem estar associadas a uma incompatibilidade de significados associados a eles por seus produtores e consumidores.

Riscos de falhas de comunicação a partir de artefatos - a probabilidade de um problema de comunicabilidade causar falhas de comunicação entre produtores e consumidores de artefatos.

Falhas de comunicação a partir de artefatos - interpretações incompatíveis dos consumidores de artefatos na perspectiva dos produtores são consideradas como falhas de comunicação. Para caracterizar as falhas de comunicação, foi utilizada a seguinte classificação de falhas de comunicação, com base na classificação descrita da Engenharia Semiótica:

- **Completa** - quando o consumidor compreende a intenção de comunicação do

produtor de forma inconsistente;

- **Parcial** - quando o consumidor compreende parcialmente a intenção de comunicação do produtor;
- **Temporária** - ocorre uma interrupção na interpretação do consumidor sobre as informações no artefato de software, o qual é compreendido posteriormente.

2.2.Trabalhos Relacionados

De acordo com Barbosa e Silva (2010), a comunicação é guiada por uma intenção, os efeitos que o emissor quer provocar ao transmitir o conteúdo da sua mensagem ao receptor. Para que a comunicação seja eficiente, o emissor deve escolher cuidadosamente uma expressão para o conteúdo que deseja comunicar, utilizando um código que o receptor seja capaz de interpretar. Nesse sentido, foram identificados trabalhos que estão relacionados à compreensibilidade dos artefatos, que se refere à interpretação do receptor do que o emissor disse em seu ato comunicativo.

Cruz-Lemus *et al.* (2010) apresentam um modelo preditivo de compreensibilidade para diagramas de máquinas de estado UML, analisando sua complexidade estrutural. O objetivo dos autores com essa proposta foi reduzir o impacto da compreensão deste diagrama. Este modelo preditivo foi desenvolvido a partir de uma família de três experimentos, cujos resultados mostraram três dimensões de complexidade estrutural que afetam a compreensibilidade: (i) o tamanho e a complexidade do fluxo de controle do diagrama de estados em termos de recursos, como o número de estados, eventos, guardas e transições de estado; (ii) as ações que são realizadas ao entrar ou sair de um estado; (iii) a sequência de ações que é executada enquanto permanece dentro de um estado.

Nakamura *et al.* (2011) propuseram três métricas relativas à compreensibilidade de diagramas de classes da UML nos seguintes aspectos: (1) estrutura de classes, (2) estrutura de pacotes e (3) atributos e operações. Os autores afirmam que as métricas ajudam na estimativa do custo de tempo para a compreensão de um diagrama de classes. Tais métricas foram avaliadas analisando a correlação entre as medidas destas e avaliações subjetivas da compreensibilidade do diagrama de classes realizadas por especialistas. De acordo com os autores, o resultado da correlação entre a avaliação subjetiva e a compreensibilidade da estrutura de classes, por exemplo, foi forte,

mostrando que tal métrica é útil para estimar o custo de tempo gasto para compreender um diagrama de classes.

Phalp *et al.* (2007) apresentam fatores que são chamados de 7Cs da Comunicabilidade, tais como: Cobertura (o caso de uso deve conter tudo o que é necessário para responder ao problema); Convincente (o caso de uso deve seguir um caminho lógico e na ordem correta); Coerente (A descrição é mais fácil de ler e mais rápida de entender se houver coerência lógica por toda parte); Consistente em relação à estrutura e gramática (o caso de uso deve estar em um nível consistente de abstração e gramática); e Consideração de Alternativas (Deve haver uma seção separada para quaisquer caminhos alternativos / excepcionais o fluxo principal). Estes fatores podem ser utilizados para apoiar o desenvolvimento ou revisão de casos de uso com o objetivo de melhorar a qualidade deste.

Fantechi *et al.* (2003) analisaram diferentes atributos, como a expressividade, consistência e a completude de casos de uso, com o uso de técnicas linguísticas. A partir disso, os autores propuseram métricas que definem a qualidade de casos de uso em relação à compreensão de casos de uso. Técnicas linguísticas podem fornecer um suporte eficaz para a realização de avaliações de qualidade dos requisitos de linguagem natural, mas não são suficientes para abordar completamente os aspectos relacionados à correção e consistência dos requisitos. No entanto, a aplicação de técnicas de análise linguística pode enriquecer as informações dos casos de uso.

Ferreira (2015) em sua tese propôs a tripla *Tool-Notation-People* (TNP), um recurso de articulação de três fatores relacionados à modelagem de software: Ferramenta (*Tool*), Notação (*Notation*) e Pessoas (*People*) (Ferreira *et al.*, 2014). Com a tripla TNP pode-se caracterizar as questões relacionadas às perspectivas do contexto de desenvolvimento de software e da interação com ferramentas de modelagem. Por exemplo: O designer (T), representado pela ferramenta, sempre considera que o usuário é um profundo conhecedor da notação UML. T disponibiliza os elementos da UML (N) relacionados ao modelo desejado, considerando que o produtor do modelo (P) conheça bem a especificação de N e suas restrições. Neste ponto tem-se os fatores TNP combinados, isto é, as decisões de T em relação à N considerando o P. Neste trabalho, junto com a tripla TNP, também é apresentado um método que combina as perspectivas cognitiva e semiótica para avaliar as ferramentas de modelagem de software com o

objetivo de analisar a metacomunicação designer-usuário, sendo os usuários, neste caso, os desenvolvedores de software. Com isso, pode-se rastrear potenciais relações entre a experiência de interação humano-computador dos envolvidos no processo de desenvolvimento de software no momento de criar, compreender ou editar modelos. A principal contribuição desta pesquisa refere-se à caracterização de problemas e questões identificadas no processo de produção e consumo de modelos de software com apoio de ferramentas de modelagem, considerando os fatores da tripla TNP. De acordo com Ferreira (2015), ferramentas de modelagem têm impacto direto na comunicabilidade de modelos de software que são produzidos e consumidos através de ferramentas de modelagem, já que os produtores e consumidores de modelos interagem com tais ferramentas ao longo do processo de desenvolvimento do software.

Embora esses trabalhos lidem com questões relacionadas à comunicação com o apoio dos artefatos desenvolvidos nas etapas iniciais do desenvolvimento de software, é necessário explorar a comunicação recíproca entre produtores e seus consumidores por meio destes artefatos. Essa comunicação recíproca está relacionada ao entendimento comum dos consumidores em relação ao que os produtores expressaram no conteúdo informacional dos artefatos de software para o que estes desejam comunicar sobre o sistema. Os trabalhos de Cruz-Lemus *et al.* (2010), Nakamura *et al.* (2011), Phalp *et al.* (2007) e Fantechi *et al.* (2003) focam em propostas que promovam a compreensão do artefato, mas não buscam promover o entendimento na perspectiva do produtor para o consumidor. O trabalho de Ferreira (2015) explorou processo de produção e consumo de modelos de software com apoio de ferramentas de modelagem, destacando questões relacionadas à comunicabilidade de modelos. No entanto, a comunicabilidade de modelos não foi explorada em relação à intenção de comunicação de produtores para que ocorra o entendimento mútuo entre estes e os consumidores de artefatos de software.

As contribuições desta tese apresentam estudos que forneceram dados para a compreensão da comunicação recíproca entre produtores e consumidores de artefatos, ressaltando aspectos de comunicabilidade de artefatos, incluindo seu estado da arte, uma vez que esse tipo de comunicação não está bem definido na literatura (Abdelzad *et al.*, 2016). A partir disto, as Diretivas de Comunicabilidade foram propostas para apoiar a comunicação entre produtores e consumidores de artefatos desenvolvidos nas etapas

iniciais do processo de desenvolvimento. Isso pode ser obtido quando os produtores realizam melhorias no conteúdo destes artefatos propostos, contribuindo para a comunicabilidade de tais artefatos, e analisam os fatores que podem impactar sua comunicação através destes.

2.3.Considerações sobre o Capítulo

Este capítulo apresentou a teoria da Engenharia Semiótica e o Princípio de Cooperação de Grice (Grice, 1975), os quais fornecem base teórica para a investigação da comunicação entre produtores e consumidores de artefatos. Além disso, foram apresentados os principais trabalhos relacionados, uma vez que estes focam na compreensão dos artefatos, a qual está relacionada à maneira como o emissor expressa o conteúdo que deseja comunicar.

A partir da compreensão destes trabalhos relacionados, foi observada uma oportunidade de pesquisa em relação à análise da comunicação recíproca entre produtores e consumidores de artefatos. Com isso, um mapeamento sistemático da literatura e diferentes estudos experimentais foram realizados para uma melhor compreensão deste tipo de comunicação. O mapeamento sistemático da literatura é apresentado no Capítulo 3 e os estudos experimentais são apresentados no Capítulo 4.

CAPÍTULO 3 – ARTEFATOS QUE APOIAM A COMUNICAÇÃO NO DESENVOLVIMENTO DE SOFTWARE

Este capítulo apresenta um mapeamento sistemático da literatura sobre o estado da arte da comunicação apoiada por artefatos.

3.1. Introdução

Os artefatos elaborados durante a etapa de análise do processo de desenvolvimento proporcionam uma compreensão maior acerca do sistema a ser desenvolvido, enquanto os artefatos que são elaborados durante a etapa de projeto promovem uma visão geral da solução que deve ser concebida para o sistema (Gomaa, 2006; Génova *et al.*, 2009).

É importante investigar como ocorre a comunicação entre produtores e consumidores de artefatos em um projeto de software. Essa investigação pode ser iniciada a partir do entendimento das notações que vem sendo utilizadas para a produção destes artefatos, como diagramas e representações visuais da interface, além das pesquisas realizadas para a melhoria deste tipo de comunicação.

Neste contexto, Garcia *et al.* (2017) conduziram um mapeamento sistemático para identificar artefatos e seus papéis na comunicação entre as áreas de metodologia ágil e Design Centrado no usuário. Como resultados, por exemplo, artefatos como protótipos, personas e *sketches* são usados durante reuniões com as equipes para facilitar a comunicação. No entanto, é importante analisar o uso dos artefatos como apoio à comunicação em um contexto amplo de desenvolvimento de software, não focando apenas no uso de artefato do contexto de metodologia ágil, além de analisar as pesquisas realizadas para esse tipo de comunicação. Com isso, um mapeamento sistemático da literatura foi realizado, apresentado na próxima subseção.

3.2. Mapeamento Sistemático sobre Artefatos que Apoiam a Comunicação no Desenvolvimento de Software

Um mapeamento sistemático pode fornecer benefícios significativos no estabelecimento de linhas de base para pesquisas futuras (Kitchenham *et al.*, 2010), pois é possível identificar o estado da arte, oportunidades de pesquisa e lacunas existentes em um determinado tópico. Com o intuito de compreender o estado da arte sobre artefatos como apoio à comunicação de membros de equipes, foi conduzido um mapeamento sistemático da literatura² com base nas diretrizes propostas por Kitchenham e Charters (2007). Ressalta-se que este mapeamento sistemático foi realizado na etapa ‘identificação do problema’ da metodologia utilizada nesta pesquisa, como mostra a Figura 3.1.

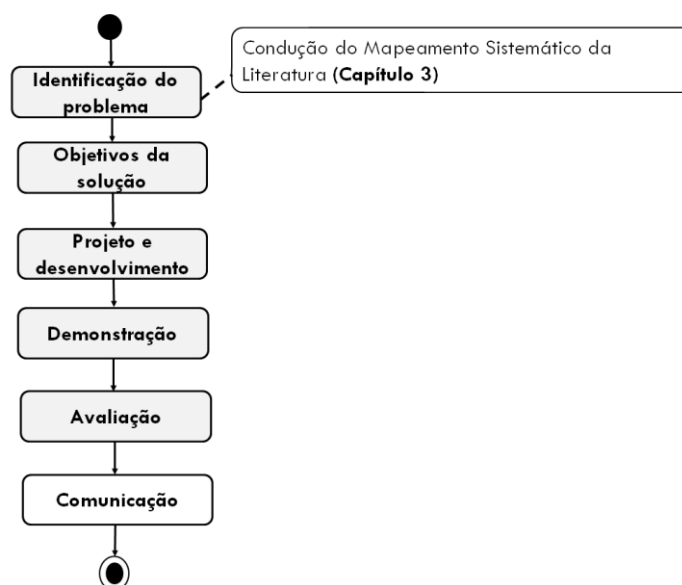


Figura 3.1: Condução do Mapeamento Sistemático.

3.2.1 Planejamento do Mapeamento Sistemático

Diante do exposto até aqui, foi definida seguinte questão de pesquisa para a compreensão da comunicação via artefato no mapeamento sistemático: “Qual o estado da arte sobre artefatos de software como apoio à comunicação no desenvolvimento de software?”. Foram definidas as seguintes subquestões para auxiliar a busca de evidências para a questão principal:

² Um mapeamento sistemático é um tipo de revisão sistemática que investiga um tópico de pesquisa específico (Kitchenham *et al.*, 2010).

SQ1. Quais são os artefatos utilizados para apoiar a comunicação de equipes no desenvolvimento de software? – Definida com o objetivo de identificar relatos de equipes de desenvolvimento de software utilizando artefatos como meio de comunicação.

SQ2. Quais são os artefatos investigados em pesquisas relacionadas à comunicação no desenvolvimento de software? - Definida com o objetivo de compreender as pesquisas realizadas com artefatos neste tipo de comunicação.

SQ3. Qual o estado da arte sobre propostas que apoiam o uso de artefatos como meio de comunicação? – Definida com o objetivo de compreender as contribuições realizadas para apoiar a comunicação eficaz por meio de artefatos.

SQ4. Quais as pesquisas realizadas no desenvolvimento de software contribuem para a comunicação a partir de artefatos? – Definida com o objetivo de compreender as pesquisas que podem contribuir com esse tipo de comunicação.

3.2.1.1 Estratégia de Busca dos Estudos Primários

Nesta subseção é descrita a estratégia definida para a busca dos estudos primários, incluindo as fontes de pesquisa, os termos de busca e etc.

Definição da *string* de busca: Foram determinados três conceitos que representam a comunicação a partir de artefatos de software, como mostra a Tabela 3.1. O primeiro conceito (C1 na Tabela 3.1) refere-se ao contexto em que os artefatos são utilizados em um contexto amplo de desenvolvimento de software. O segundo conceito (C2 na Tabela 3.1) está relacionado aos artefatos que representam soluções elaborados por profissionais em diferentes perspectivas para o desenvolvimento de sistemas, como a representação da estrutura e arquitetura do sistema, incluindo a forma como os usuários podem interagir com tais sistemas. O terceiro conceito (C3 na Tabela 3.1) está relacionado à comunicação. A partir desses conceitos foram criados termos de busca.

Conceitos	Termos de busca
C1. Desenvolvimento de Software	“software development” OR “software engineering”
C2. Artefatos criados nas etapas iniciais do processo de desenvolvimento	“software model” OR “software diagram” OR “software artifact” OR “documentation”
C3. Apoio na comunicação	“communication”

Tabela 3.1: Termos utilizados para Definir a String de Busca.

Inicialmente, a pesquisadora principal realizou a leitura completa de alguns artigos disponíveis no Google Scholar, de forma aleatória, que pudessem estar relacionados com este mapeamento sistemático. Após isso, alguns artigos tornaram-se “artigos de controle” para a avaliação dos termos, como os artigos descritos abaixo com as respectivas palavras-chaves. Tais termos poderiam estar no título, resumo e palavras-chave dos artigos.

- Improving Coordination and Communication in Distributed Software Development through Context-Based Software Artifacts Awareness: A Controlled Experiment (palavras-chave: *context awareness, software artifacts, distributed teams, experiment*).
- Practices and Perceptions of UML Use in Open Source Projects (palavras-chave: *architecture documentation, OSS projects, GitHub, motivation, communication, effectiveness of UML*).
- A Cognitive Perspective on Developer Comprehension of Software Design Documentation (palavras-chave: *documentation, design, experimentation, human factors*).

A estrutura utilizada, para a definição destes conceitos e termos relacionados, foi baseada no trabalho de Fernandez *et al.* (2011). Os termos foram combinados entre si, formando a string de busca, como mostra a Figura 3.2.

```

("software development" OR "software engineering" )
AND
("software model" OR "software diagram" OR "software
artifact" OR "documentation" )
AND
("communication")
```

Figura 3.2: String de Busca Utilizada no Mapeamento Sistemático.

Fontes de pesquisa: As bibliotecas digitais Scopus e Engineering Village foram adotadas pelos seguintes motivos: (a) por serem meta-bibliotecas que indexam publicações de editoras conhecidas, como ACM, IEEE e Elsevier; (b) por permitem a definição de filtros, como idiomas e tipos de documentos; (c) por permitirem a obtenção das publicações.

Idiomas dos artigos: O idioma escolhido foi o inglês, por ser adotado pela grande maioria das conferências e periódicos internacionais relacionados ao tema da pesquisa.

3.2.1.2 Critérios de Seleção de Artigos

Kitchenham e Charters (2007) sugerem a definição de critérios de seleção para os artigos que são retornados pela *string* de busca. Os critérios de seleção tem como objetivo identificar estudos que forneçam evidências para a questão de pesquisa. Os critérios de inclusão e exclusão deste mapeamento foram descritos na Tabela 3.2 e na Tabela 3.3. Sobre o “Exc4”, este foi definido devido aos artigos que são disponibilizados com título e resumo em inglês, mas que possuem o restante do texto em outro idioma.

Tipo	Identificador	Descrição
Critério de inclusão	[Inc1]	O artigo deve descrever artefatos que apoiam a comunicação no desenvolvimento de software.
	[Inc2]	O artigo deve descrever estudos (academia)/relatos de experiência (indústria) com os artefatos que apoiam a comunicação.
	[Inc3]	O artigo deve descrever práticas da indústria relacionadas à construção de artefatos que melhoraram a comunicação de equipes.
	[Inc4]	O artigo deve descrever práticas da indústria relacionadas ao consumo de artefatos que melhoraram a comunicação de equipes.
	[Inc5]	O artigo deve descrever uso de abordagens e ferramentas que apoiam a comunicação a partir de artefatos.
	[Inc6]	O artigo deve descrever uso de abordagens e ferramentas que apoiam a construção de artefatos que apoiam a comunicação.
	[Inc7]	O artigo deve descrever pesquisas (concluídas, em andamento ou futuras) sobre artefatos que apoiam a comunicação.

Tabela 3.2: Critérios para Inclusão de Artigos.

Tipo	Identificador	Descrição
Critério de exclusão	[Exc1]	Não atender nenhum dos critérios de inclusão.
	[Exc2]	Abordagens e ferramentas de geração automática de modelos que não são utilizados para apoiar a comunicação.
	[Exc3]	A versão completa do artigo não estar disponível entre as fontes selecionadas.
	[Exc4]	O idioma do artigo não ser inglês.
	[Exc5]	O artigo descrever artefatos de comunicação que não estão no contexto da pesquisa.

Tabela 3.3: Critérios para Exclusão de Artigos.

3.2.1.3 Procedimentos para a Seleção de Artigos

O processo de seleção de artigos foi realizado da seguinte maneira: **remoção dos artigos duplicados** consistiu em analisar os dados obtidos nas fontes de pesquisa, os quais foram armazenados na ferramenta Start (Fabbri *et al.*, 2012). A ferramenta indicou os artigos duplicados.

No **primeiro filtro**, os títulos e resumos dos artigos retornados na execução da *string* de busca foram analisados quanto aos critérios de inclusão e exclusão. A decisão de incluir ou excluir um artigo foi registrada para posterior verificação da orientadora deste trabalho. No **segundo filtro**, a leitura completa dos artigos incluídos no primeiro filtro foi realizada para julgar os artigos em relação aos critérios de inclusão e exclusão. Os artigos selecionados no segundo filtro foram submetidos ao **processo de extração de dados**. Os itens do formulário de extração deste mapeamento sistemático estão descritos na Tabela 3.4.

Item de extração	Descrição
Título, Conferência, Autores, Filiação, Ano de Publicação e Foco do artigo	Descrição das informações gerais do artigo.
SQ1.	Descrição para SQ1.
SQ2.	Descrição para SQ2.
SQ3.	Descrição para SQ3.
SQ4.	Descrição para SQ4.
Pesquisa realizada na indústria ou academia?	(a) Indústria; (b) Academia.
Qual a metodologia de desenvolvimento?	(a) Metodologia Ágil; (b) Processo Tradicional
Quem são os participantes da pesquisa?	Descrição dos participantes.

Tabela 3.4: Formulário de Extração.

3.2.2 Execução do Mapeamento Sistemático

Inicialmente, o mapeamento sistemático foi executado em Maio de 2019. Como resultados, foram retornados 730 artigos a partir das bibliotecas digitais selecionadas. Deste total, 372 artigos foram obtidos da Scopus e 358 foram obtidos da Engineering Village.

Com o intuito de analisar a confiabilidade do processo de seleção, foi selecionada uma amostra aleatória de 20 artigos para analisar o grau de concordância entre as pesquisadoras envolvidas. Para avaliar se a concordância é razoável, foi realizado o teste Kappa (Landis e Koch, 1977) para avaliar a significância de tal concordância. A medida de concordância tem como valor máximo 1, onde este valor representa total concordância. Os valores próximos de 0, indicam nenhuma concordância. A Tabela 3.5 mostra os valores para a interpretação dos resultados do teste Kappa, sugerido por Landis e Koch (1977).

Em relação à amostra selecionada, a pesquisadora principal e a orientadora analisaram os artigos individualmente, atribuindo um critério de seleção. Os resultados dos critérios de seleção utilizados para a análise dos 20 artigos foram comparados da seguinte

forma: para critérios de inclusão foi atribuído o valor 1 e para critérios de exclusão foi atribuído o valor 2. Desta forma, foi analisada a concordância das pesquisadoras na inclusão ou exclusão dos artigos, não exatamente no critério específico de inclusão ou exclusão atribuído. O nível de concordância obtido foi de 0.732, indicando concordância significativa entre as pesquisadoras. Após isso, pesquisadora principal continuou com a execução do primeiro filtro individualmente.

Valores de Kappa	Interpretação
0	Sem concordância (<i>no agreement</i>)
0 – 0,19	Concordância fraca (<i>poor agreement</i>)
0,20 – 0,39	Concordância razoável (<i>fair agreement</i>)
0,40 – 0,59	Concordância moderada (<i>moderate agreement</i>)
0,60 – 0,79	Concordância significativa (<i>substantial agreement</i>)
0,80 – 1	Concordância quase perfeita (<i>almost perfect agreement</i>)

Tabela 3.5: Interpretação dos resultados do Kappa segundo Landis e Koch (1977).

A Figura 3.3 apresenta um resumo quantitativo do resultado obtido com a remoção dos artigos duplicados (com 283 artigos duplicados), resultado do primeiro filtro (com 62 artigos selecionados) e do segundo filtro (com 21 artigos selecionados). Após o segundo filtro, iniciou-se o processo de extração de dados, os quais foram revisados pela orientadora. Os principais resultados obtidos neste mapeamento sistemático são apresentados na próxima subseção.

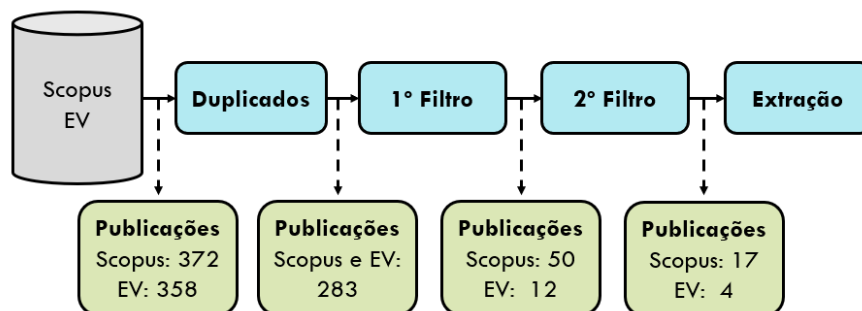


Figura 3.3: Procedimento para a Seleção dos Artigos.

3.2.3 Resultados do Mapeamento Sistemático

Os resultados do mapeamento sistemático resultaram em 21 artigos extraídos. A Figura 3.4 ilustra o período de publicação dos artigos. Em relação aos veículos de publicação, seis dos 21 artigos foram publicados em periódicos, enquanto 15 foram

publicados em conferências. Esses artigos são apresentados no Apêndice A, referenciados no texto a seguir com “P001, P002,” e etc.

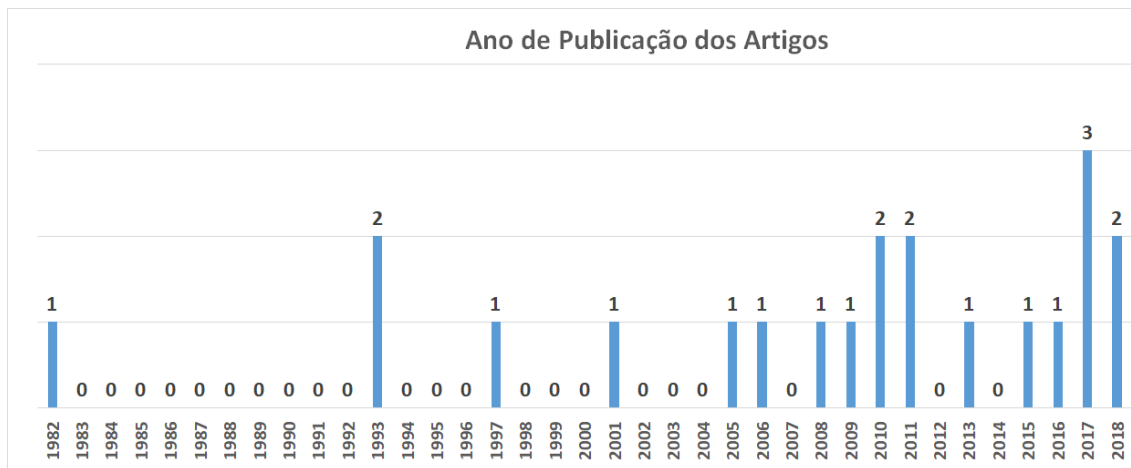


Figura 3.4: Distribuição das Publicações por Ano.

Em relação às subquestão de pesquisa “*SQL. Quais são os artefatos utilizados para apoiar a comunicação de equipes no desenvolvimento de software?*”, os resultados estão divididos em relação ao artefatos utilizados pelas equipes como apoio à comunicação e notações de modelagem utilizadas na produção de alguns destes artefatos.

A Figura 3.5 ilustra os resultados dos artefatos de software que apoiam a comunicação. Foram identificados protótipos, mockups, documentação e especificação de requisitos, conforme a descrição de cada publicação. Em relação aos protótipos, Kautz (1993) [P011] apresenta um estudo de caso com duas estratégias para o uso de *protótipos* em equipes de desenvolvimento. Essas estratégias foram: adoção de consultores de comunicação em um projeto e o uso de diários do projeto. Os consultores de comunicação são membros do projeto e estes são responsáveis pelos aspectos da comunicação relacionados a um determinado projeto. Os diários do projeto foram considerados um meio de “capturar o andamento de um determinado projeto”. A partir deles, a equipe refletiu sobre as atividades de projeto, sendo este considerado um documento complementar ao produto.

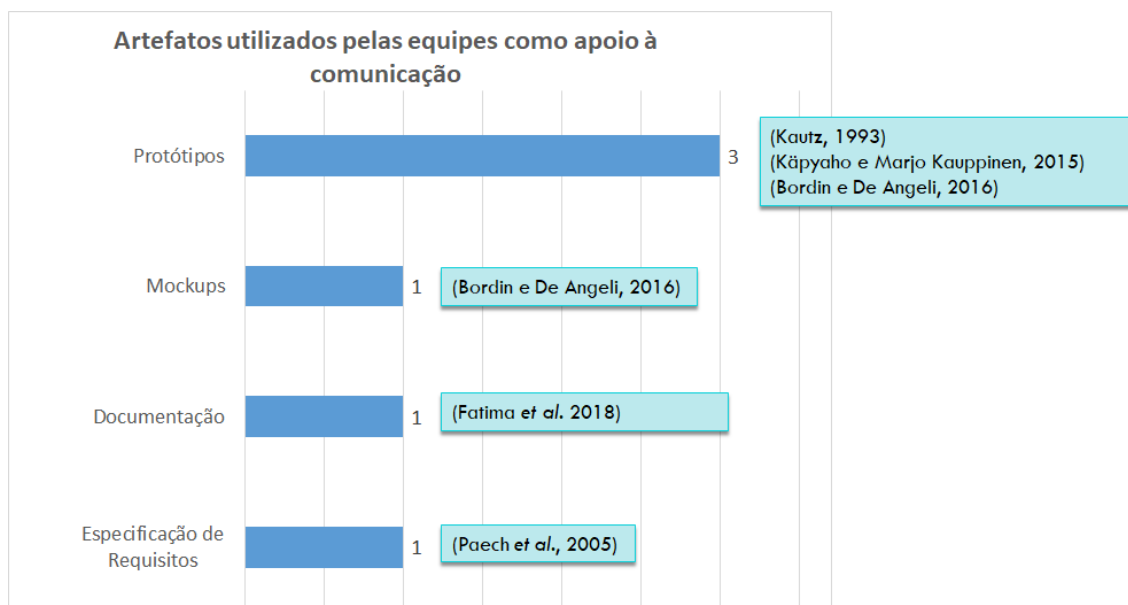


Figura 3.5: Artefatos Relacionados Na Literatura que Apoiam a Comunicação no Desenvolvimento de Software.

Em relação aos *requisitos*, Paech *et al.* (2005) [P014] apresentam um modelo de informação que torna explícitos os documentos utilizados e as responsabilidades dos stakeholders durante a engenharia de requisitos. O modelo de informação caracteriza um conjunto de documentos de requisitos que evoluem em projetos diferentes, incluindo as responsabilidades dos stakeholders. De acordo com os autores, o modelo de informação é um meio efetivo e prático para garantir que os stakeholders estejam mutuamente conscientes de suas responsabilidades e isso pode melhorar a comunicação.

Käpyaho e Marjo Kauppinen (2015) [P010] relatam um estudo de caso realizado, em larga escala, em um projeto ágil. O objetivo do estudo foi explorar como os *protótipos* podem ajudar a resolver alguns desafios relacionados à engenharia de requisitos, como a falta de documentação e problemas que afetam a qualidade da comunicação. No entanto, o artigo não relata como os protótipos são consumidos pela equipe de desenvolvimento e sim do seu apoio à comunicação.

Em relação aos *protótipos* e *mockups*, Bordin e De Angeli (2016) [P004] relatam que no caso de sistemas existentes, como melhorias desses, os designers preparam os protótipos de alta fidelidade, que dependem da interface existente. No caso de um novo software, os designers preparam mockups, representando fluxo de interação e layouts. Em seguida, os mockups são discutidos iterativamente com o cliente. Isso permite

verificar se os requisitos foram corretamente compreendidos e garantir que o cliente esteja ciente do status do projeto. Ainda de acordo com os autores, estes artefatos também podem ter uma função de equilibrar a relação da equipe com o cliente, protegendo a empresa contra mudanças insustentáveis nos requisitos.

Fatima *et al.* (2018) [P007] relatam que a falta de clareza na *documentação* leva a equipe a tomar decisões por conta própria. Com isso, neste artigo é apresentada uma metodologia que proporciona o desenvolvimento de jogos com um processo mais efetivo em relação à documentação. Nesta metodologia, os próprios desenvolvedores de jogos criam os protótipos e estes são compartilhados com a equipe, que está distribuída geograficamente.

A Figura 3.6 ilustra as notações utilizadas na produção de artefatos nas etapas iniciais do processo de desenvolvimento que apoiam a comunicação. Davies *et al.* (2006) [P006] apresentam um survey que buscou investigar a prática de modelagem conceitual na Austrália. Os autores observaram seis notações mais utilizadas em organizações de médio e grande porte, que são: *Data flow diagram*, *System flowcharts*, *Workflow modeling*, *UML* (Unified Modeling Language) e *Structured charts*. Além disso, os autores citam as ferramentas mais utilizadas e fatores que estão relacionados com a modelagem, como a comunicação a partir dos modelos, conhecimento da equipe sobre as notações de modelagem e deficiências de ferramentas.

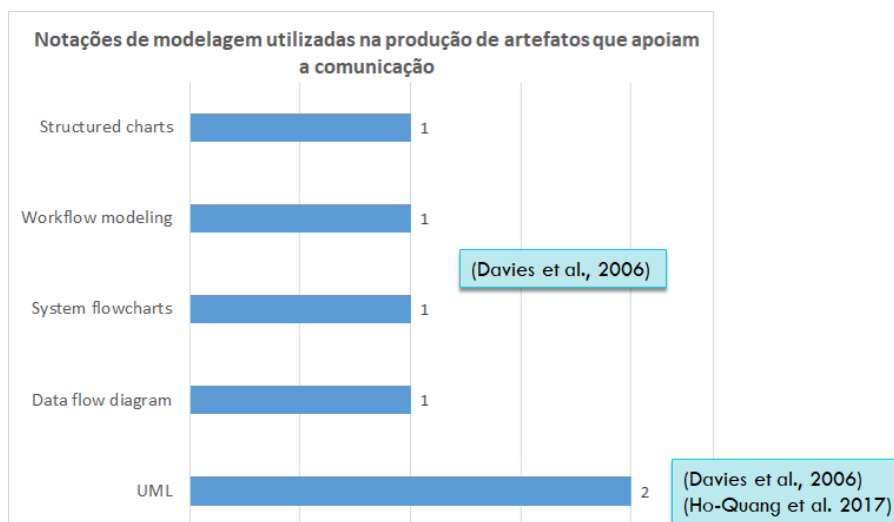


Figura 3.6: Notações Mencionadas na Produção de Artefatos que Apoiam a Comunicação.

Ho-Quang *et al.* (2017) [P009] apresentam um survey sobre o uso da UML em projetos open source para identificar pontos em comum e diferenças para o uso da UML na indústria. De acordo com os autores, este trabalho apresenta *insights* que podem ajudar empresas a decidirem se promovem o uso da UML em projetos open source. Além disso, os autores relatam que os modelos UML apoiam a comunicação em tais projetos.

A Figura 3.7 ilustra os artefatos investigados em pesquisas relacionadas à comunicação no desenvolvimento de software, respondendo a subquestão de pesquisa “SQ2. Quais são os artefatos investigados em pesquisas relacionadas à comunicação no desenvolvimento de software?”.

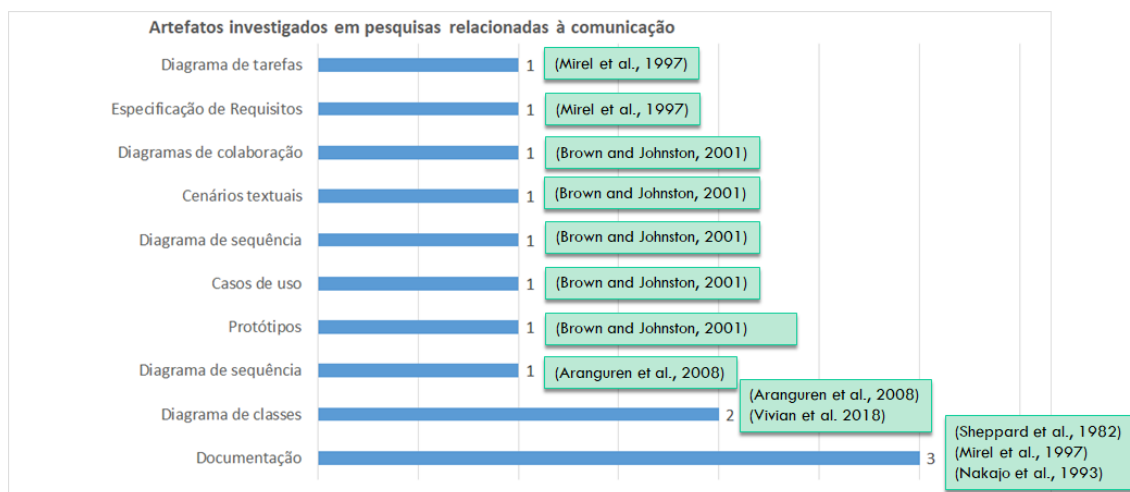


Figura 3.7: Artefatos investigados em Pesquisas Relacionadas à Comunicação no Desenvolvimento de Software.

Sheppard *et al.* (1982) [P017] relatam a condução de três experimentos conduzidos para examinar os efeitos do formato da *documentação* sobre o desempenho de programadores em diferentes tarefas relacionadas ao software. De acordo com os autores, essa formatação está relacionada: (i) uso de símbolos em que as informações são apresentadas; (ii) organização dessas informações e (iii) representações gráficas. Os resultados destes experimentos forneceram evidências de que o formato de documentação pode ter um significativo efeito no desempenho dos programadores em tarefas relacionadas ao software. Nakajo *et al.* (1993) [P013] apresentam questões que devem ser consideradas no desenvolvimento de um sistema de documentação de interface para evitar erros de comunicação nas atividades de design de software: (i) Quais informações devem ser

incluídas na documentação? (ii) Qual estrutura a documentação deve ter? e (iii) Qual a forma descritiva deve ser usada nesta documentação?

Mirel *et al.* (1997) [P012] apresentam um relato em uma disciplina do curso de Engenharia de Software sobre melhorias nos efeitos da comunicação para promover a usabilidade nos produtos. Os alunos escreveram a documentação do usuário com o objetivo de este ser usado como objeto de informação no projeto, empregando a *especificação de requisitos, diagrama de tarefas e documentação* de forma geral. Foram enfatizadas atividades como: (i) treino dos alunos com técnicas para definir uma visão do produto; (ii) ênfase à perspectiva do cliente e do usuário através do design da interface, incluindo a análise dos efeitos da interface no restante do código; e (iii) realização de testes iterativos de usabilidade, começando no início do ciclo do projeto.

Brown e Johnston (2001) [P005] relatam a proposta de uma ferramenta que vincula protótipos a outros artefatos criados durante o processo de desenvolvimento de software. No estudo de caso realizado com a ferramenta, os autores analisaram protótipos criados por especialistas de IHC, *protótipos* estáticos (sketches e digital photographs) e protótipos dinâmicos (derivados das especificações de artefatos criados por engenheiros de software). Em relação aos artefatos produzidos por engenheiros de software, estes foram: *casos de uso, diagrama de sequência, cenários textuais e diagramas de colaboração*.

Aranguren *et al.* (2008) [P002] apresentam a *Ontology Design Patterns* (ODP), que são soluções para problemas típicos de modelagem que os bioontologistas podem usar na construção de ontologias. Os autores apresentam o uso de ODP para o desenvolvimento diagramas da UML, como o *diagrama de classes* e *diagrama de sequência*.

Vivian *et al.* (2018) [P020] apresentam um estudo experimental para avaliar a viabilidade da abordagem DiSEN-CollaborAR, uma proposta para apoiar o uso de artefatos no desenvolvimento de software distribuído. Neste estudo, os autores analisaram a produção e modificação de *diagramas de classe* e código fonte. Os resultados indicam que DiSEN-CollaborAR pode melhorar a coordenação e a comunicação no desenvolvimento de software distribuído.

A Figura 3.8 ilustra o estado da arte sobre as propostas para o desenvolvimento de artefatos de software que apoiam a comunicação, respondendo a subquestão de pesquisa “SQ3. Qual o estado da arte sobre propostas que apoiam o uso de artefatos como meio de comunicação?”. Como resposta para esta pergunta, foram identificadas: (i) propostas para

a produção dos artefatos com objetivo de melhorar a comunicação; e (ii) propostas de atividades para a melhoria da comunicação por meio dos artefatos.

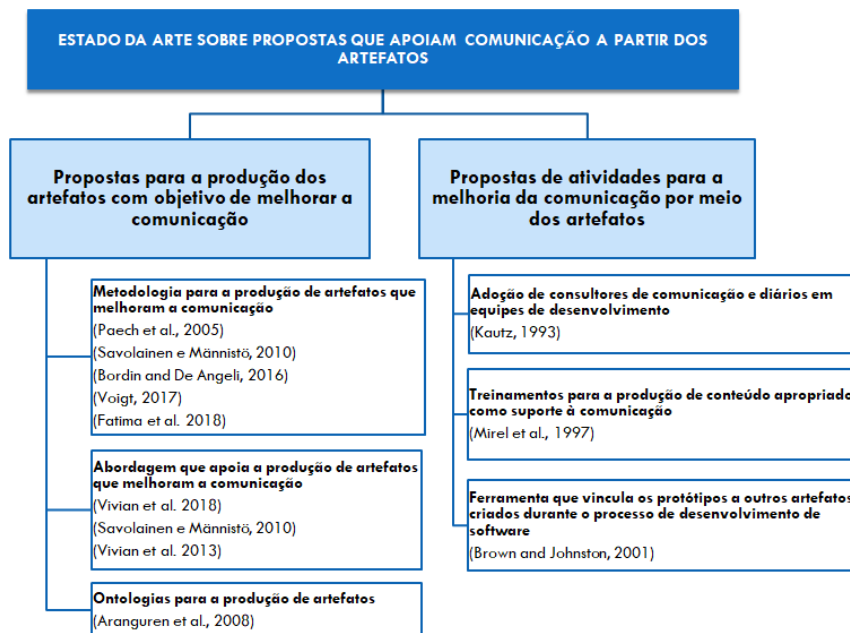


Figura 3.8: Estado da Arte sobre Propostas que Apoiam a Comunicação a partir dos Artefatos.

Em relação às propostas para a produção dos artefatos com o objetivo de melhorar a comunicação, nós analisamos cada uma das propostas com base nos elementos do espaço de comunicação (Jakobson, 1960), adaptados para representar as condições de comunicação via artefato, como: o artefato é disponibilizado com o suporte de uma ferramenta (o canal) com informações do domínio do problema (contexto) para apoiar a comunicação entre seus produtores (os emissores) e consumidores (os receptores). O produtor, em sua mensagem, deve considerar a forma como o conteúdo é expresso (o uso do código) em tais artefatos. Com isso, os trabalhos foram explorados com os seguintes focos: (1) uso de determinados artefatos para apoiar a comunicação; (2) envolvidos do processo comunicativo, isto é, os produtores e consumidores; (3) conteúdo dos; (4) canal de comunicação. Além disso, foi verificado se tais propostas foram analisadas em equipes de desenvolvimento e se foram expandidas, como mostra o resumo da Figura 3.9.

		Foco no uso de determinados artefatos para apoiar a comunicação	Foco nos envolvidos na comunicação a partir de artefatos	Foco no conteúdo dos artefatos	Foco no canal de comunicação	Avaliação em equipes de desenvolvimento?	Trabalho expandido/ explorado na indústria?
Metodologia para a produção de artefatos	(Paech et al., 2005)		X			Sim	Não
	(Savolainen e Männistö, 2010)		X			Não	Sim
	Voigt, 2017	X	X	X	X	Sim	Não
	Fatima et al. 2018	X				Sim	Não
Abordagem que apoia o uso de artefatos	(Aranguren et al., 2008)			X		Avaliação na academia (Não)	Sim
	Vivian et al. 2018		X	X		Avaliação na academia (Não)	Não
Ontologias para a produção de artefatos	Vivian et al. 2013		X	X		Não	Sim

Figura 3.9: Análise das Propostas para a Produção dos Artefatos com Objetivo de Melhorar a Comunicação.

Em relação aos envolvidos no processo comunicativo via artefato, Paech *et al.* (2005) [P014] apresentam um modelo de informação que auxilia na comunicação dos envolvidos, mostrando suas responsabilidades. Esse modelo apoia a reflexão dos envolvidos com os seguintes questionamentos: “Quais pontos de vista são capturados em quais documentos?”; “Quem cria quais requisitos e para qual público?”; “Quem aprova os documentos?”; “Quem revisa os documentos?”; “Quem é responsável pela consistência dos requisitos?”; “Quem é responsável por aprovar e propagar as mudanças de requisitos?”.

Em relação ao conteúdo dos artefatos, Aranguren *et al.* (2008) [P002] apresentam ODP. De acordo com os autores, ao criar ontologias, a modelagem é documentada com mais precisão com as ODPs. Desta maneira, o uso de ODPs deve melhorar a comunicação entre desenvolvedores de ontologias.

Savolainen e Männistö (2010) [P015] apresentam uma abordagem centrada em conflito para documentação de arquitetura (*conflict-centric approach to architecture documentation*). De acordo com os autores, uma visão arquitetônica de requisitos conflitantes por parte dos stakeholders ajuda os arquitetos a se concentrarem nas decisões importantes e isto reduz a quantidade de documentação arquitetural. Os autores buscam com isso melhorar a comunicação a partir da documentação arquitetônica.

No que se refere aos envolvidos na comunicação via artefato e conteúdo dos artefatos, Vivian *et al.* (2013) [P019] apresentam a abordagem DiSEN-CollaborAR, a qual fornece suporte para o conhecimento do contexto em que os artefatos estão inseridos para apoiar a colaboração de equipes. A abordagem fornece mecanismos para capturar informações contextuais em relação ao ambiente de trabalho e também repositórios compartilhados. A abordagem oferece suporte para gerar dependências entre artefatos de software e exibir informações contextuais. Com essa abordagem, os autores esperam minimizar os problemas de comunicação e coordenação e, conseqüentemente, aumentar a produtividade e a qualidade do produto de software desenvolvido.

O trabalho de Voigt (2017) [P021] foca nos quatro aspectos avaliados. O autor apresenta um método de documentação ágil, o qual deve ser utilizado por desenvolvedores. O método apoia a produção de *documentação* com o objetivo de fazer o produtor refletir sobre a importância dos documentos em projetos ágeis, destacando também a importância da documentação para a comunicação.

Em Vivian *et al.* (2018) [P020], um estudo experimental é apresentado para avaliar a viabilidade da abordagem DiSEN-CollaborAR no desenvolvimento de software distribuído. A abordagem DiSEN-CollaborAR foi comparada com uma abordagem ad-hoc. Os autores analisaram as abordagens em relação à produção e modificação dos artefatos de software. Os resultados indicam que DiSEN-CollaborAR pode melhorar a coordenação e a comunicação no desenvolvimento de software distribuído.

No que se refere ao uso de determinados artefatos para apoiar a comunicação, Fatima *et al.* (2018) [P007] apresentam uma metodologia para o desenvolvimento global de jogos (*global game development methodology*). De acordo com os autores, esta metodologia proporciona o desenvolvimento de jogos com um processo mais efetivo, a qual adota uma *documentação* e *prototipagem* na pré-produção. Com isso, os autores afirmam que é possível comunicar funcionalidades e a proposta para o design de interação de uma única vez para a equipe.

Destas propostas, apenas as propostas de Voigt (2017) [P021], Fatima *et al.* (2018) [P007] e Paech *et al.* (2005) [P014] foram avaliadas experimentalmente na indústria, mas tais propostas não foram expandidas. Sobre os trabalhos expandidos, o estudo apresentado em Savolainen e Männistö (2010) [P015] foi reaplicado por Khatwani *et al.* (2017). Neste,

os autores adaptaram a abordagem centrada em conflito para documentação de arquitetura para mesclar pontos de vista e demonstrar como essa nova proposta pode ser aplicada através de alguns exemplos. Em Seddig-Raufie *et al.* (2012), a principal contribuição deste trabalho é a classificação de três estratégias diferentes para a especialização de ODP (Aranguren *et al.*, 2008) [P002] para *Ontology Design Pattern Property Specialisation Strategies*. Os autores fizeram uma comparação com outras ontologias para problemas de design. Por fim, a abordagem proposta por Vivian *et al.* (2013) [P019] foi aplicada para obter informações contextuais a partir de arquivos de código fonte escritos em Java (Massago *et al.*, 2013).

Sobre as propostas de atividades para a melhoria da comunicação por meio dos artefatos, Kautz (1993) [P011] apresenta um estudo de caso com duas estratégias para melhorar a comunicação. O autor ressalta que um líder de projeto pode agir como um consultor de comunicação e o uso de diários são benéficos, pois capturam o entendimento dos membros de uma equipe para a compreensão do andamento do projeto. No entanto, escrever diários podem ser tarefas demoradas e devem ser descritas sem erros. A partir dessas propostas, não foram identificados trabalhos de extensão.

Mirel *et al.* (1997) [P012] apresentam um relato sobre a importância do fator da comunicação em uma disciplina do curso de Engenharia de Software. Com isso, pode-se induzir que a proposta dos autores é conscientizar futuros engenheiros de software sobre os artefatos como meio de comunicação, reduzindo problemas advindos de falhas de comunicação.

Brown e Johnston (2001) [P005] relatam a proposta de uma ferramenta chamada Panorama, a qual vincula os protótipos a outros artefatos criados durante o processo de desenvolvimento de software. Os artefatos foram criados por especialistas em IHC e engenheiros de software durante o desenvolvimento do software. A ferramenta Panorama facilita o compartilhamento de artefatos e visa aumentar a qualidade da comunicação entre engenheiros de software e especialistas em IHC durante as fases de desenvolvimento e manutenção.

A Figura 3.10 ilustra as pesquisas realizadas no desenvolvimento de software que contribuem para a comunicação a partir de artefatos, respondendo a subquestão de pesquisa “SQ4. *Quais as pesquisas realizadas no desenvolvimento de software contribuem para a comunicação a partir de artefatos?*”. Essas pesquisas foram classificadas de acordo com

suas contribuições para os diferentes elementos relacionados à comunicação, tais como: análise de conteúdo dos artefatos, suporte ao canal de comunicação a partir dos artefatos, análise do consumo dos artefatos e análise de falhas de comunicação a partir dos artefatos.

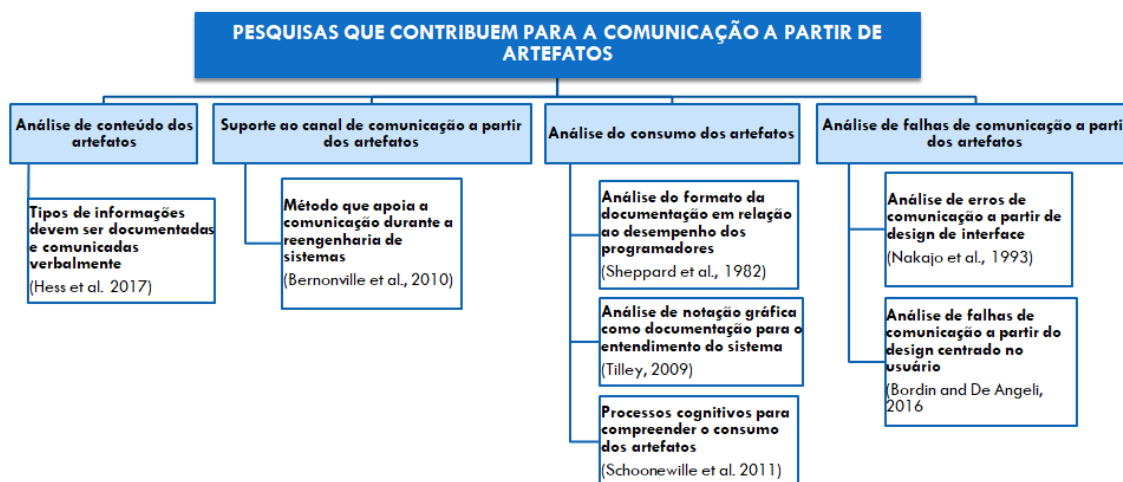


Figura 3.10: Pesquisas que contribuem para a Comunicação Via Artefatos.

Em relação à análise de conteúdo dos artefatos, Hess *et al.* (2017) [P008] apresentam um artigo que questiona os tipos de informações que devem ser documentadas e/ou comunicados verbalmente em projetos ágeis. Os autores realizaram um estudo inicial com membros de equipes ágeis e os resultados indicaram, por exemplo, que uma documentação menos detalhada pode causar inconsistências nos produtos de software. Portanto, é necessário ter artefatos que apoiem a comunicação, mesmo em projetos que são desenvolvidos com metodologia ágil.

Sobre o suporte ao canal de comunicação a partir de artefatos, Bernonville *et al.* (2010) [P003] apresentam um relato sobre a análise de métodos de Engenharia de Software e IHC como suporte à comunicação, facilitando a reengenharia de um software hospitalar que permite a solicitação ou dispensa de medicamentos para apoiar médicos e enfermeiros durante suas atividades. Os autores focam no uso do método ErgoPNets, que é caracterizado pelo uso de redes Petri (PN), o qual inclui explicações sobre problemas ergonômicos e fornece recomendações para a redução destes problemas.

Em relação ao consumo de artefatos, Sheppard *et al.* (1982) [P017] apresentam três experimentos conduzidos para examinar os efeitos do formato da documentação sobre o desempenho de programadores em diferentes tarefas relacionadas ao software. Os

resultados destes experimentos forneceram evidências de que o formato de documentação, como o uso de fluxogramas e de *Program Design Language*, pode ter um significativo efeito no desempenho dos programadores em tarefas relacionadas ao software.

Tilley (2009) [P018] apresentam uma pesquisa que sumariza 15 anos de pesquisa sobre uso de notação gráfica como documentação para o entendimento do sistema. De acordo com o autor, a notação gráfica pode ajudar a entender o sistema e a comunicação. No entanto, os “comunicadores” técnicos não costumam estar envolvidos neste processo. Ainda de acordo com o autor, o resultado é que os engenheiros, que têm o melhor das intenções, não possuem o *background* necessário para explorar os recursos da documentação gráfica para apoiar tarefas dos usuários finais. Portanto, o autor relata uma lição aprendida: “*precisamos saber conversar*”. Portanto, isso ressalta a importância de o produtor pensar no consumidor dos modelos produzidos.

Schoonewille *et al.* (2011) [P016] apresentam um estudo experimental que investigou a seguinte questão de pesquisa: “*Como os desenvolvedores de software compreendem representações da arquitetura de software?*”. No estudo foi avaliado um diagrama de componentes da UML e uma descrição textual deste diagrama sob a perspectiva cognitiva. Os participantes analisaram o diagrama e descrição textual. Em seguida, eles responderam questões específicas relacionadas ao diagrama e ao texto. Os pesquisadores também pediram para estes pensarem em voz alta para posterior análise dos processos cognitivos, sendo este gravado. Como resultados, por exemplo, foi apresentada a correlação sobre informações consideradas como corretas e incorretas pelos participantes e o grau de certeza destas. Em relação aos dois artefatos, os autores afirmaram que os participantes utilizaram os dois para terem certeza da resposta.

Sobre as falhas de comunicação a partir de artefatos, Bordin e De Angeli (2016) [P004] argumentam que falhas de comunicação podem se tornar pontos focais no design centrado no usuário. Eles apresentam um estudo de caso que relata falhas de comunicação que afetam: o design centrado no usuário a partir do envolvimento com usuário, os artefatos desenvolvidos (protótipos e mockups), sincronização das interações e entendimento da equipe sobre as tarefas que devem ser realizadas. Os autores relatam que os profissionais afirmaram que documentação tornou-se cada vez mais relevante para manter a equipe de desenvolvimento alinhada quando esta se tornou distribuída devido à introdução de outros profissionais trabalhando em horários diferentes. Além disso, Nakajo *et al.* (1993) [P013]

apresentam erros de comunicação entre engenheiros envolvidos no desenvolvimento de software a partir da *documentação* de design de interface.

3.2.4 Discussão dos Resultados do Mapeamento Sistemático

Em relação às pesquisas relatadas nos artigos identificados, foram observadas pesquisas na indústria de software, na academia e parcerias da academia com a indústria de software. A Figura 3.11 apresenta um resumo dos tipos de estudos apresentados nas publicações identificadas neste mapeamento sistemático.

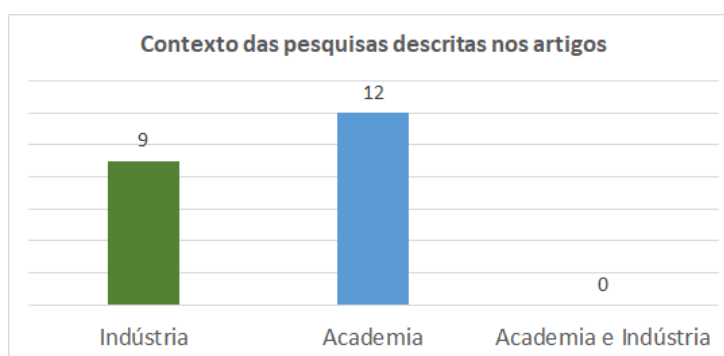


Figura 3.11: Contexto dos Estudos apresentados nas Publicações encontradas.

Os participantes destas pesquisas foram alunos e profissionais. Ressalta-se que algumas publicações não tiveram participantes relatados por serem propostas ou discussões dos autores sobre um determinado assunto.

Em relação à questão de pesquisa principal deste mapeamento: “*Qual o estado da arte sobre artefatos de software como apoio à comunicação no desenvolvimento de software?*”, foram identificadas especificações de requisitos, diagramas e representações visuais da interface que apoiam a comunicação, como o diagrama de classes da UML, diagrama de sequência e mockups. Esses artefatos expressam as ideias dos engenheiros de software sobre a análise e projeto de software. Relatos de uso na indústria e pesquisas com esses artefatos como apoio à comunicação foram identificados neste mapeamento sistemático. Pesquisas relacionadas à comunicação via artefato e pesquisas que contribuem para este tipo de comunicação também foram identificadas.

A partir dos resultados deste mapeamento sistemático, foram identificadas oportunidades de pesquisa para a melhoria do conteúdo dos artefatos e os envolvidos neste tipo de comunicação. A Figura 3.12 sumariza pontos que mostram tais oportunidades.

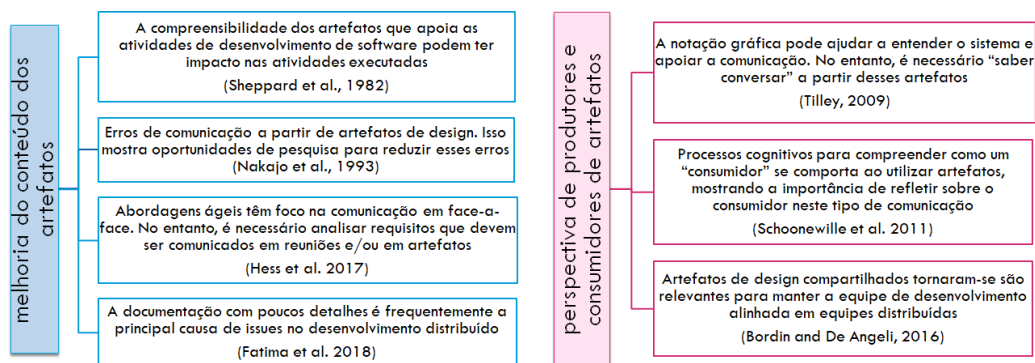


Figura 3.12: Oportunidades de Pesquisa para a Melhoria da Comunicação via Artefato.

Sobre a melhoria do conteúdo dos artefatos, Sheppard *et al.* (1982) e Nakajo *et al.* (1993) mostraram que a falta de entendimento mútuo entre produtores e consumidores de artefatos pode causar falhas na comunicação e isso impacta nas atividades executadas no desenvolvimento de software. Hess *et al.* (2017) mostraram que é importante analisar os requisitos que devem ser comunicados de forma oral e escrita. No trabalho de Fatima *et al.* (2018), os autores mostraram que poucos detalhes na documentação causaram inconsistências no desenvolvimento distribuído. Com isso, surgem oportunidades de pesquisa para o desenvolvimento de propostas que apoiem os produtores na análise da quantidade de informações e informações relevantes que um artefato deve ter para apoiar a comunicação, melhorando assim a sua compreensibilidade.

Sobre as oportunidades de pesquisa em relação à perspectiva de produtores e consumidores de artefatos de software, Schoonewille *et al.* (2011) e Tilley (2009) mostram a importância de refletir sobre o consumidor de artefatos de software. Tais artefatos são importantes para manter o entendimento mútuo de equipes, como equipes distribuídas (Bordin e De Angeli, 2016). Com isso, é importante ter uma proposta para que o produtor reflita sobre o consumidor dos artefatos, contribuindo para que as informações nos artefatos sejam compreendidas pelos demais membros da equipe. Com o desenvolvimento de tais propostas, problemas que afetam a comunicação entre produtores e consumidores podem ser reduzidos.

3.3. Considerações sobre o Capítulo

Neste capítulo foi apresentado um mapeamento sistemático da literatura com o objetivo de responder à questão de pesquisa “Qual o estado da arte sobre artefatos de software como apoio à comunicação no desenvolvimento de software?”. Os resultados

mostraram artefatos que têm apoiado a comunicação de equipes, pesquisas que estão relacionadas com a comunicação via artefato e pesquisas que contribuem para este tipo de comunicação. Oportunidades de pesquisa foram observadas em relação às considerações que devem ser realizadas por produtores para que ocorra a compreensão recíproca a partir de artefatos, evidenciando o problema investigado nesta tese.

CAPÍTULO 4 - ESTUDOS PARA A COMPREENSÃO DO PROBLEMA

Este capítulo apresenta estudos experimentais realizados com o objetivo de compreender a comunicação via artefato e os problemas que podem afetá-la.

4.1 Introdução

Para uma melhor compreensão da comunicação entre produtores e consumidores a partir de artefatos de software, foram conduzidos diferentes estudos exploratórios. Esses estudos estão relacionados à etapa ‘identificação do problema’ da metodologia utilizada, como mostra a Figura 4.1. Nestes estudos exploratórios, teorias que investigam diferentes formas de comunicação foram utilizadas para apoiar um melhor entendimento da comunicação via artefato. Além disso, os artefatos investigados nos estudos foram escolhidos por conveniência.

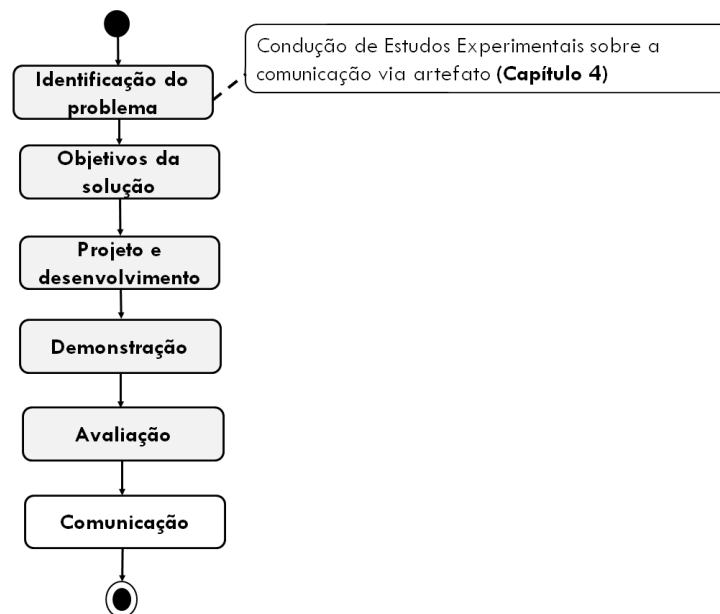


Figura 4.1: Condução de Estudos Experimentais sobre a Comunicação via Artefato.

A Seção 4.2 apresenta um estudo *in vivo* realizado com o objetivo de compreender as atividades realizadas e a comunicação via artefato em uma equipe de projeto. Esta equipe utilizou artefatos para apoiar sua comunicação. Em seguida são apresentados estudos *in vitro*.

Na Seção 4.3, é apresentado um estudo exploratório realizado com o objetivo de compreender a comunicação de projetistas, de um projeto real, para os participantes que assumiram o papel de desenvolvedores a partir de casos de uso da UML. Devido ao tempo de execução do estudo, esses participantes realizaram apenas o desenvolvimento de mockups a partir dos casos de uso. Os resultados mostraram problemas de comunicabilidade que causaram falhas de comunicação.

Na Seção 4.4 é apresentado um estudo exploratório para analisar a comunicação de diferentes duplas através de casos de uso, as quais atuaram como projetistas, para outras duplas, que atuaram como desenvolvedores de software. Neste estudo, foi possível analisar as intenções pretendidas pelos produtores que podem causar, ou não, problemas de comunicabilidade. Neste estudo também foram identificados problemas de comunicabilidade que causaram falhas de comunicação.

4.2 Investigando a Comunicação via Artefato de uma Equipe de Projeto

Esta seção apresenta um estudo *in vivo* realizado em um projeto de cooperação da indústria e academia para o entendimento de artefatos como apoio à comunicação. Este estudo foi apresentado em Lopes *et al.* (2016) e Lopes *et al.* (2018a). As contribuições pretendidas neste estudo também visam adicionar à literatura a importância das práticas adotadas com modelos de software em relação à comunicação de equipes no desenvolvimento de software.

Este sistema foi desenvolvido por três equipes distribuídas, nomeado como *Home Care Development Project* (HCDP). A equipe responsável pela HCDP, isto é, o cliente, definiu todas as funcionalidades a serem desenvolvidas e os artefatos que deveriam ser entregues. Uma equipe realizou as atividades relacionadas à codificação do sistema. Outra equipe realizou as atividades relacionadas à análise e projeto do sistema, composta por cinco engenheiros de software e um gerente do projeto.

O estudo apresentado nesta seção foi realizado na equipe de análise e projeto do HCDP. Os membros da equipe de análise e projeto possuíam experiência em análise e projeto de aplicações. A equipe adotou práticas da metodologia Scrum (Schwaber e Shuterland, 2011). As etapas de análise e projeto da aplicação foram divididas em Sprints de duração fixa. O projeto teve duração de 6 meses, sendo dividido em 6 Sprints. A

pesquisadora principal desta tese também atuou no projeto, especialmente para observar as atividades realizadas pela equipe. Na etapa de projeto, os artefatos de software solicitados pela equipe responsável foram:

- **Modelos de interação** - A equipe desenvolveu estes modelos com apoio da linguagem MoLIC (*Modeling Language for Interaction as Conversation*) (Barbosa e Paula, 2003) para representar todos os caminhos de interação entre o usuário e o sistema. Os elementos de um diagrama MoLIC e exemplo de modelagem podem ser consultados em Lopes *et al.* (2016).
- **Especificação de casos de uso da UML** – Na especificação de casos de uso, a equipe descreveu os casos de uso considerando os diferentes fluxos de execução.
- **Mockups** - Foram desenvolvidos também os protótipos necessários para representar a interface.
- **Diagramas de atividades da UML** - Desenvolvidos para fornecer uma visualização do comportamento do sistema através da sequência de ações nas atividades.

Os diferentes membros da equipe possuíam alto conhecimento para a elaboração de cada um dos artefatos, sendo estes responsáveis pela produção de cada um, conforme sua especialidade. Após o desenvolvimento destes artefatos, estes foram analisados por engenheiros de softwares diferentes, ou seja, o engenheiro de software que construiu um modelo não foi o mesmo que analisou. Tal prática teve como objetivo identificar inconsistências nas informações representadas nos artefatos. As etapas de análise e projeto da aplicação foram divididas em Sprints de duração fixa. O projeto teve duração de seis meses, sendo dividido em seis Sprints. Na Sprint 1 foi realizada a etapa de análise e nas demais Sprints foram desenvolvidos os artefatos de software. Na Sprint 1 também foram priorizadas as funcionalidades a serem projetadas da HCDP (dividas em partes, como A, B, C, D e E). Ao final de cada Sprint foi realizada a atividade de reunião de retrospectiva da equipe, sendo discutidas as lições aprendidas.

A seguir são descritas as atividades realizadas na análise (Sprint 1) e produção dos artefatos de software (Sprints 2 e 3). A descrição das demais Sprints não estão descritas neste texto devido à sua semelhança com a Sprint 3.

4.2.1 Atividades de Etapa de Análise da HCDP na Sprint 1

Na Sprint 1, Personas (Cooper, 1999) (Cooper et al., 1999) e Cenários (Jarke, 1999) foram criados com o objetivo de compreender diferentes tipos de usuário para a aplicação a ser projetada. Personas são representações fictícias, específicas e concretas de usuários-alvo. Estas podem ajudar engenheiros de requisitos na compreensão dos usuários-alvo. Um Cenário é um conjunto ordenado de interações, geralmente entre um sistema e um conjunto de atores externos ao sistema. Os cenários podem assumir várias formas e fornece vários tipos de informação em diferentes níveis de abstração.

A HCDP deveria controlar as atividades de uma pessoa idosa (ir ao médico, tomar remédios, fazer refeições). Assim, as personas criadas representavam familiares e cuidadores do idoso. Para a criação das personas, utilizamos um *template* composto pelas seguintes informações: *Dados de identificação, condições de moradia, nível de experiência com tecnologias, nível de responsabilidade, estilo de aprendizagem, aceitação de mudança e o contexto de uso de da aplicação*. As personas criadas pela equipe foram validadas pelos engenheiros de software com usuários através de pesquisas sobre o domínio do problema, pois estes mostraram interesse em obter uma aplicação com o mesmo propósito da HCDP. Após isto foram desenvolvidos cenários, que descreviam os detalhes da relação das personas com os dispositivos móveis.

Para elaboração dos cenários, foi utilizado um *template* composto pelas seguintes informações: Contexto, objetivos, eventos e ações. A partir deste cenário foram identificados requisitos para a construção da aplicação, por exemplo: “*Avisar que as atividades da rotina principal do idoso foram cumpridas*” – Requisito 1; “*Alertar que as atividades da rotina principal do idoso não foram cumpridas*” – Requisito 2.

A partir dos requisitos, desenvolveu-se o diagrama de casos de uso da HCDP. Após isto, a equipe responsável pelo HCDP e a equipe de análise e projeto definiram em conjunto a priorização de desenvolvimento dos casos de uso que seriam desenvolvidos nas demais Sprints.

4.2.2 Construção dos Artefatos de Software na Sprint 2

Na Sprint 2, a ordem de construção adotada para mockups, diagramas MoLIC, especificação de casos de uso e diagramas de atividades é apresentada na Figura 4.2.

Nota-se que a construção de mockups, diagramas MoLIC e especificação de casos de uso foi realizada de forma paralela, apenas com base no diagrama de caso de uso, requisitos e no cenário. Em seguida, o diagrama de atividades foi construído, com base nos três artefatos desenvolvidos. Ao final da Sprint 2, após o desenvolvimento de todos os artefatos, foi realizada a reunião de retrospectiva da equipe de projeto da HCDP. A seguir são discutidas algumas lições identificadas na Sprint 2.

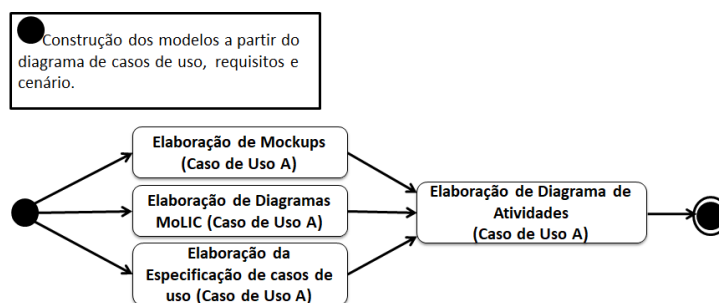


Figura 4.2: Atividades Realizadas para a Construção dos Artefatos na Sprint 2.

Na Sprint 2, existem 15 documentos que relatam as lições aprendidas codificadas com auxílio da abordagem PABC-Pattern (Rabelo *et al.*, 2014), com um total de sete documentos relacionados com a construção dos artefatos. Foram selecionadas as três principais lições aprendidas que relatam problemas na construção dos artefatos. Uma destas lições aprendidas tem por **Título** Retrabalho ao manter o padrão para os artefatos, a qual foi gerada pela seguinte **Situação**: “Ao desenvolver os artefatos separados após a reunião, existe um trabalho excessivo em padronizar os casos de usos, mockups, modelagem de interação e diagrama de atividade.”. A equipe de projeto definiu a seguinte estratégia de **Ação**: “Uma análise em conjunto onde será definida a ideia geral dos requisitos antes de criar os artefatos através de um documento formal”. Assim, a equipe concluiu que pode-se reduzir o retrabalho em relação às inconsistências entre tais artefatos.

Nesse contexto, foi selecionada a lição aprendida que tem por **Título** Dependência entre atividades pode ocasionar atrasos no projeto, que ocorreu pela seguinte **Situação**: “A melhoria do caso de uso dependia da entrega dos mockups. A entrega dos mockups chegou atrasada”. Diante disto, foi observada a seguinte **Consequência da Situação**: “Retrabalho. Atraso da atividade seguinte que deveria ser entregue a outro participante”. Sobre as estratégias de **Ação**, para esta codificação, foi definido: “Quando a atividade de uma pessoa depende da atividade de outra pessoa, deve-se haver um cuidado e esforço

para que não se atrase”. Assim, percebe-se que houve atraso na especificação de casos de uso, uma vez que tal artefato deveria representar as mesmas informações nos mockups e diagramas MoLIC.

Ainda neste contexto, foi selecionada a lição aprendida que tem por **Título** Falta de rastreabilidade entre os mockups e o caso de uso, a qual ocorreu pela seguinte **Situação**: *“Falta de rastreabilidade ou mapeamento para entender o que está no mockup e o que isto representa no caso de uso”*. A equipe de projeto definiu a seguinte estratégia de **Ação**: *“Ver quais são os mockups representados no caso de uso e integrar mais os dois modelos”*. Portanto, houve falta de entendimento de alguns engenheiros de software em relação à compreensão dos requisitos para o desenvolvimento de diferentes modelos. Diante disto, houve inconsistências ao representar as informações da HCDP entre os artefatos desenvolvidos apenas com base nos requisitos, diagrama de caso de uso e cenário.

Ressalta-se que na Sprint 2 os diagramas MoLIC não foram utilizados como base para a construção dos demais artefatos. Portanto, nesta Sprint, não houve o uso de um artefato que focasse nas informações essenciais na interação usuário-sistema para a construção dos outros artefatos. Por esta razão, houve retrabalho da equipe em relação à consistência dos artefatos. Assim, a equipe realizou uma mudança na ordem de construção dos artefatos. Esta mudança é apresentada na próxima subseção.

4.2.3 Construção dos Artefatos de Software na Sprint 3

Na Sprint 3, a ordem de construção em relação aos diagramas MoLIC, mockups, especificação de casos de uso e diagramas de atividades é apresentada na Figura 4.3. Esta ordem de construção é diferente em relação à ordem construção adotada na Sprint 2. A construção de mockups, diagrama de atividades e especificação de casos de uso foi realizada com base nos diagramas MoLIC, requisitos, diagramas de caso de uso, além do cenário. Esta mudança ocorreu para usar os diagramas MoLIC como base para a construção de outros artefatos.

Na Sprint 3, a equipe de projeto realizou a elaboração conjunta da modelagem de interação, ou seja, o engenheiro de software responsável pela modelagem da interação construía os diagramas MoLIC e os demais engenheiros acompanharam e discutiram

sobre a modelagem realizada, projetando soluções alternativas de interação para o usuário. Com a construção de artefatos de interação de maneira conjunta, foi possível identificar possíveis problemas na interação usuário-sistema, sendo desenvolvidas soluções para que tais problemas não ocorram. Após isto, os mockups, diagrama de atividades e especificação de casos de uso foram construídos com base nos diagramas MoLIC desenvolvidos. Ao final da Sprint 3, também foi realizada a reunião de retrospectiva da equipe de projeto da HCDP.

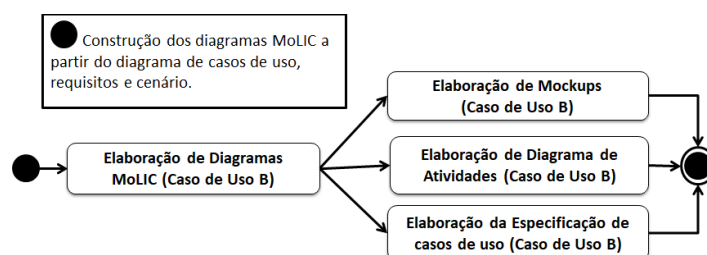


Figura 4.3: Atividades Realizadas para a Construção dos Artefatos na Sprint 3.

Em relação às lições aprendidas da equipe nesta Sprint, existem 12 documentos que relatam a codificação das lições aprendidas, com um total de quatro documentos relacionados com a construção dos artefatos. Foram selecionadas as três principais lições aprendidas sobre o uso de diagramas MoLIC como base para a construção dos demais artefatos pela equipe de projeto.

Uma lição aprendida selecionada sobre ao uso da MoLIC na Sprint 3 tem por **Título** Uso de diagramas MoLIC no início, a qual foi ocasionada pela seguinte **Situação**: “O uso dos diagramas MoLIC no início possibilitou a reflexão sobre o caso de uso. E também ver que é necessário o uso em conjunto com a interface”. Foi atribuída a esta lição aprendida codificada a seguinte **Consequência da Situação**: “Com base nos diagramas MoLIC, a equipe acabou obtendo tempo e ao final da sprint as modificações foram finalizadas de maneira simples”. No entanto, as modificações realizadas nos mockups refletem em modificações que devem ser realizadas no diagrama MoLIC, uma vez que tais artefatos podem ser projetados em conjunto (Lopes *et al.*, 2015). Em relação às estratégias de **Ação**, para esta codificação, não houve um plano de ação a ser definido, pois tal experiência foi considerada como positiva.

Além disso, outra lição aprendida selecionada tem por **Título** Reflexão sobre consumo de tempo em atividades do projeto, a qual ocorreu pela seguinte **Situação**: “Benefícios em atividades que consumiam tempo de projeto e não eram bem

compreendidas pelos participantes”. A equipe de projeto definiu a seguinte estratégia de **Ação**: “*Participantes tiveram que refletir o seguinte: Quando não se tinha um padrão de interface, começar pela modelagem de interação fazia sentido*”. Portanto, a equipe de projeto concluiu que apesar do uso da modelagem de interação inicial ser útil para compreender as funcionalidades do sistema, as alterações a partir dos *mockups* para a compreensão de todas as funcionalidades são úteis. Além disso, neste projeto houve uma melhor compreensão das funcionalidades a partir dos diagramas MoLIC em conjunto com os protótipos.

Com base nestas lições aprendidas observa-se como aspectos positivos: (a) o uso dos diagramas MoLIC no início para compreender as funcionalidades e (b) seu uso como base para a construção de outros artefatos. Além disso, a partir da Sprint 3, foi adotada a prática de inspeção dos artefatos por outros membros da equipe com o objetivo de aumentar a qualidade destes artefatos.

4.2.4 Limitações do Estudo

Existem limitações neste estudo de caso, como a expectativa do projetista responsável pelos diagramas MoLIC em relação aos benefícios da modelagem de interação. Isto pode ter influenciado o projetista responsável pela construção dos diagramas MoLIC em obter resultados positivos na etapa de projeto. Porém, ressalta-se que nenhum dos projetistas da equipe possuíam relação com a autoria da MoLIC. Além disso, as lições aprendidas foram codificadas por um membro da equipe que não tinha participado da construção dos diagramas MoLIC. Portanto, os resultados das lições aprendidas abordam apenas a conclusão da equipe sobre cada Sprint.

Também não é possível afirmar que os mesmos benefícios podem ser generalizados para diferentes tipos de aplicação, como desktop. É importante compreender o uso dos diagramas para diferentes tipos de aplicações. Além disso, este é o resultado de uma única equipe de projeto. Portanto, os resultados não são generalizáveis, mas são considerados válidos para o contexto do projeto.

4.2.5 Discussão dos Resultados

Neste estudo foi analisado o uso da modelagem de interação nas Sprints 2 e 3 como base para o desenvolvimento de outros artefatos. Com isso, foi possível compreender que

o uso de modelos de interação para apoiar a comunicação da equipe, visto que esse modelo possuía aspectos que promoviam o entendimento do sistema para o desenvolvimento dos demais artefatos.

A prática adotada sobre a modelagem conjunta para o entendimento do sistema permitiu que a equipe discutisse sobre o sistema e as informações foram formalizadas no diagrama MoLIC. A partir do diagrama MoLIC, a equipe desenvolveu os demais artefatos. A prática realizada na Sprint 3 permitiu a redução do retrabalho obtido na Sprint 2. Através dos resultados deste estudo também foi possível obter indícios sobre os benefícios da utilização de diagramas MoLIC como apoio à comunicação.

Em relação à prática realizada na Sprint 2, com o apoio do espaço de comunicação apresentado no Capítulo 3, percebe-se que ocorreram falhas na comunicação da equipe devido às diferentes percepções do elemento “Contexto”. Apesar da equipe estar realizando a modelagem de um mesmo problema, diferentes contextos foram considerados. A discussão da modelagem inicial permitiu aos produtores e consumidores a compreensão sobre o “Contexto” a qual os artefatos deveriam ser elaborados. Portanto, percebe-se que apesar do consumidor compreender os códigos no artefato, essa comunicação pode falhar se o “Contexto” não estiver definido. Ressalta-se que a pesquisadora principal participou como membro da equipe e apenas observou as atividades em relação à comunicação a partir de artefatos.

Com os resultados observados neste estudo, sugere-se que o contexto de modelagem seja definido para equipes de projeto que tenham mais de um profissional responsável. Isso pode ser realizado através de reuniões ou do próprio uso de um artefato, como o modelo de interação utilizado no início do projeto. Também se sugere um artefato que concentre as informações necessárias para os demais membros da equipe realizarem suas atividades, evidenciando aqui o apoio dos artefatos na comunicação.

4.3 Investigando a Comunicação de Produtores de Casos de Uso para Desenvolvedores

Nesta seção apresenta-se um estudo *in vitro*, em um ambiente semelhante ao desenvolvimento distribuído, com o objetivo de responder à questão de pesquisa “*Que*

problemas de comunicabilidade de artefatos podem afetar a comunicação entre produtores e consumidores?”. Assim, pode-se obter a compreensão das informações expressas em artefatos que foram compreendidas ou não a partir dos artefatos investigados.

Em relação ao espaço de comunicação para produtores e consumidores de artefatos, apresentado no Capítulo 3, ressalta-se que este estudo foi direcionado para a “Mensagem” (conteúdo do artefato) em relação à maneira como os produtores expressaram sua intenção através do “Código” (elementos do artefato). Para os demais elementos, foram tomadas providências, como a escolha de “Receptores” que conheçam o artefato e as ferramentas que disponibilizaram estes, o “Canal”, além de uma explicação para estes sobre o “Contexto” do software projetado.

Os casos de uso da UML foram investigados por serem um dos artefatos que apoiam a comunicação em equipes de desenvolvimento, tais como analistas de sistemas, projetistas e desenvolvedores (Misbhauddin e Alshayeb, 2015). Assim, esta pesquisa foi direcionada pela seguinte questão de pesquisa específica *“Quais problemas de comunicabilidade de casos de uso podem afetar a comunicação entre produtores e consumidores?”*.

Para responder essa questão de pesquisa, foi realizado um estudo exploratório para analisar a comunicação entre produtores e consumidores de artefatos. Neste estudo exploratório, foram utilizados os casos de uso do projeto HCDP, apresentado na Seção anterior. Estes artefatos haviam sido inspecionados por outro membro da equipe projeto. Portanto, estes artefatos possuíam boa qualidade na perspectiva dos seus produtores em relação à quantidade de defeitos de software (Granda *et al.*, 2015).

Foram convidados 44 participantes para realizar em pares o desenvolvimento de mockups a partir do caso de uso, os consumidores do caso de uso. Eles receberam o diagrama e a especificação de casos de uso para criar os mockups com base nestes. Durante o desenvolvimento dos mockups, uma câmera de vídeo estava em execução para analisarmos posteriormente a conversa entre os participantes e as decisões para a produção dos mockups. No final do estudo, os participantes desenvolveram 22 mockups. Para a análise da comunicação entre produtores e consumidores, foi adotada a seguinte estratégia:

1 - Os problemas na forma de expressão do diagrama e da especificação de casos de uso foi analisado com o apoio do princípio cooperativo de Grice (1975). A partir das máximas de Grice, foram elaboradas questões que apoiam tal análise. Essas questões são apresentadas na Subseção 4.3.3.1. Com isso, problemas de comunicabilidade no caso de uso investigado foram identificados.

2 – Os mockups desenvolvidos pelos participantes foram analisados com a ferramenta SigniFYIng Message (SFYI Message) (De Souza *et al.*, 2016), a qual suporta o mapeamento das informações expressas em artefatos de software. O resultado desta análise foi comparado com o diagrama e especificação de casos de uso, resultando em inconsistências entre algumas informações. Com isso, potenciais falhas de comunicação foram identificadas.

3 – A conversa dos participantes foi analisada para confirmar se as potenciais falhas de comunicação identificadas a partir dos mockups eram de fato falhas de comunicação entre produtores e consumidores ou apenas deficiências no desenvolvimento dos mockups. Na análise da conversa dos participantes, também foram identificadas outras falhas de comunicação não identificadas a partir dos mockups.

4 – Triangulações entre os problemas de comunicabilidade e as falhas de comunicação foram realizadas.

Os resultados mostraram os problemas de comunicabilidade do caso de uso investigado que causaram falhas de comunicação entre os produtores e consumidores.

As contribuições pretendidas neste estudo visam também adicionar à literatura a importância da comunicabilidade de artefatos. Os resultados obtidos podem indicar a importância dos produtores de artefatos sobre como suas intenções podem impactar na compreensão de outros membros de uma equipe de software.

4.3.1 Planejamento do Estudo

No estudo exploratório, ao invés dos participantes realizarem a etapa de codificação, planejou-se para os participantes realizarem apenas a construção de mockups com o apoio da ferramenta Balsamiq³, devido ao tempo limitado dos participantes na execução do estudo. Foram convidados para participarem do estudo alunos finalistas de

³ <https://balsamiq.com/>

graduação do curso de Ciência da Computação, os quais já haviam cursado disciplinas de Engenharia de Software e Interação Humano-Computador. Portanto, a amostra do estudo foi selecionada por conveniência. Os participantes são descritos como P1, P2 e assim sucessivamente. O estudo foi planejado para ser realizado em duplas com o intuito de promover a discussão do diagrama e especificação de casos de uso (UC), incluindo a explicitação percebida dos consumidores durante a construção da mockups.

Em relação ao diagrama e a especificação de UC do projeto HCDP, os artefatos explorados neste estudo focavam na realocação de um determinado compromisso dos familiares/profissionais com o idoso. Assim, existiam os perfis de responsável pelo compromisso, novo responsável (usuário que poderia assumir o compromisso) e coordenador (responsável por verificar os impedimentos relacionados aos cuidados do idoso). Este sistema era apoiado pelo monitoramento de agentes inteligentes por meio de sensores como GPS e Agenda. Em relação ao diagrama e especificação de UC cedidos para este estudo, disponibilizados em um relatório técnico (Lopes *et al.*, 2019d), tais artefatos tinham as seguintes informações:

Diagrama de UC - foi disponibilizado pelos projetistas o diagrama completo do sistema interativo, o qual possui sete casos de uso (Realocar Compromisso e Monitorar Compromisso. Os demais foram omitidos devido às restrições do projeto) e três atores (Responsável, Coordenador e Smartphone);

Especificação de UC – este caso de uso possui as seguintes tarefas que podem ser realizadas pelo usuário: (i) a verificação de novos responsáveis para a realização de um compromisso, (ii) o pedido de realocação realizado pelo responsável do compromisso, (iii) pedido de realocação realizado de maneira automática pelo sistema. As seguintes informações estavam relacionadas com a especificação de UC: três funcionalidades relacionadas, quatro atores, sendo três atores representados de forma única no diagrama, pré-condição, 11 passos no fluxo principal, 11 passos no fluxo alternativo, fluxo de exceção, e sete regras de negócio.

Um laboratório de informática para a execução do estudo, com a instalação das ferramentas Balsamiq (para a construção de protótipos), Astah⁴ (para a leitura de

⁴ <http://astah.net/>

diagramas de casos de uso) e leitor para arquivos em pdf para a visualização da especificação de UC. Estas ferramentas também foram utilizadas no projeto do sistema Elderly Care. Além disso, a ferramenta Camtasia⁵ foi utilizada para capturar a interação dos participantes com o diagrama e especificação de casos de uso ao construir os mockups.

4.3.2 Execução do Estudo

Para a execução do estudo, primeiro foi realizada uma apresentação sobre o contexto do sistema HCDP e, em seguida, a apresentação da atividade a ser realizada no estudo, a qual foi realizada de maneira semelhante no desenvolvimento real do sistema (como as equipes de desenvolvimento do HCDP eram distribuídas, a equipe de codificação recebia o diagrama completo e a especificação de um dos casos de uso para desenvolvedor). Em seguida, foi entregue aos participantes o diagrama e especificação de um dos casos de uso. Foi informado aos participantes que o diagrama de UC foi entregue para eles compreenderem o contexto do sistema e foi pedido apenas a construção dos mockups relacionada à especificação entregue. Os participantes leram o diagrama e a especificação de UC para compreenderem as informações do sistema e iniciaram a construção da mockups com a ferramenta Balsmiq. Dois projetistas deste sistema também estavam presentes na execução do estudo para esclarecer eventuais dúvidas dos participantes sobre o diagrama e especificação de UC.

4.3.3 Resultados do Estudo

4.3.3.1 Análise do Casos de Uso

As Máximas de Grice [16] foram adaptadas e combinadas para a análise do UC, formando questões que apoiaram tal análise, como mostra a

Nosso objetivo com esta análise foi compreender a forma de expressão do UC, buscando identificar problemas que podem afetar a compreensão recíproca entre membros de uma equipe de desenvolvimento de software. Como resultados, foram identificados problemas de comunicabilidade do UC, apresentados na Tabela 4.2.

Tabela 4.1.

⁵ <https://www.techsmith.com/camtasia.html>

Máximas de Grice	Questões para a Análise dos Casos de Uso
Quantidade	- Informações necessárias descritas? - Excesso de informações descritas?
Relação	- Informações relevantes descritas?
Qualidade	- Existem informações que não são verdadeiras em relação ao software?
Modo	- Existem informações que poderiam ser descritas de maneira mais objetiva? - Existem informações organizadas? - Existem informações obscuras?
Quantidade e Modo	- A falta de informações causa ambiguidade? - O excesso de informações causa equívoco?
Quantidade e Relação	- As informações necessárias e relevantes estão descritas? - Caso exista o excesso de informações, isto é relevante?
Quantidade e Qualidade	- As informações necessárias contêm declarações que não são verdadeiras?

Nosso objetivo com esta análise foi compreender a forma de expressão do UC, buscando identificar problemas que podem afetar a compreensão recíproca entre membros de uma equipe de desenvolvimento de software. Como resultados, foram identificados problemas de comunicabilidade do UC, apresentados na Tabela 4.2.

Tabela 4.1: Máximas de Grice adaptadas para a Análise do UC.

Máximas de Grice	Problemas de Comunicabilidade Identificados no Caso de Uso Investigado
Quantidade	- Falta da descrição dos atores na especificação de UC, como Responsável Original, No Responsável e Coordenador. - Falta da descrição de cada um dos agentes inteligentes na especificação de UC, como GPS e Agenda. - Falta de descrição das diferenças entre notificações, avisos e lembrete.
Modo	- Falta de padronização na descrição dos agentes inteligentes no diagrama e na especificação de UC. - A desorganização da especificação do UC pode impactar na compreensão das informações. As informações poderiam ser organizadas em outras especificações de UC.

Tabela 4.2: Problemas de Comunicabilidade no Caso de Uso.

4.3.3.2 Análise dos Mockups

As informações dos mockups foram mapeadas com a ferramenta SFYI Message, como mostra a Figura 4.4 da análise dos mockups produzidos por P1 e P2. Essa análise foi realizada para os outros conjuntos de mockups desenvolvidos pelos participantes. A análise completa está disponível em um relatório técnico (Lopes *et al.*, 2019d).

Dupla P1 e P2
SFYI Message: I. O que o desenvolvedor sabe sobre o usuário...
Sabem que os familiares e profissionais relacionados aos cuidados do idoso podem realocar um compromisso que estes não possam cumprir. Isso pode ser realizado de forma manual ou automática no sistema. Caso a tentativa de realocação não seja realizada, o usuário nomeado como Coordenador (responsável por verificar o andamento dos compromissos do idoso) será informado que o idoso não possui um responsável para a realização de um determinado compromisso.
SFYI Message: II. A intenção e expectativa do desenvolvedor...
Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela de compromissos. O usuário seleciona um compromisso e é exibido os detalhes do compromisso com as opções para realocar o compromisso e voltar para a tela anterior. A partir do botão realocar, é exibida a lista de familiares e profissionais disponíveis para assumirem o compromisso. O usuário pode selecionar um novo responsável e realocar. O usuário é informado do pedido de realocação enviado e da tentativa de realocação através da aplicação. Caso este seja concluído, o usuário é informado do novo responsável de tal compromisso. O usuário também pode não ter realizado a tentativa de realocação. Então, caso o usuário tente fazer a realocação, este é informado que o tempo para realizar a realocação para outro responsável finalizou (<u>Os mockups não ficaram tão claros em relação ao usuário não selecionar um novo responsável quando o tempo de realocação tiver sido finalizado</u>). Caso o compromisso não seja realocado, o Coordenador é informado que o idoso não possui um responsável para a realização de tal compromisso.
SFYI Message: III. Apoio e suporte do desenvolvedor...
A funcionalidade realocar compromisso, caso o usuário não faça a realocação, a aplicação fará automaticamente a tentativa de realocação. Se a tentativa de realocação não for concluída, o Coordenador dos compromissos do idoso será notificado. Este poderá resolver a situação sem o uso do sistema (realizando o compromisso ou ligando para alguém realiza-lo).

Figura 4.4: Análise do Mockup Produzido por P1 e P2 com SFYI Message.

Com essa análise, foram identificadas inconsistências entre os mockups e o UC. Essas inconsistências foram consideradas como possíveis falhas, porque as informações expressas nos mockups são diferentes da proposta dos produtores de UC. Essas inconsistências são apresentadas na Tabela 4.3.

Inconsistência nos mockups	Informação no Caso de Uso
Não ficou clara a conclusão do tempo de realocação por meio do mockup	A proposta de realocação do UC inclui a tentativa de realocação pelo usuário e pelo próprio sistema, o último ocorre se o usuário não selecionar a realocação
O botão de detalhes do compromisso não foi expresso no mockup	A proposta do UC possui um botão para visualizar o compromisso
Os detalhes do compromisso estão incompletos no mockup	A proposta do UC possui todos os detalhes do compromisso que será realocado

Inconsistência nos mockups	Informação no Caso de Uso
O tempo de conclusão da realocação não foi expresso, além das outras operações após esta	A proposta do UC apresenta o tempo para o usuário executar a realocação e outras operações pelo próprio sistema, se a realocação não for executada
As informações para desativar o botão de realocação após o final do tempo não estão claros no mockup	A proposta do UC mostra o botão de realocação desabilitado se o tempo para realocação do usuário não for realizado
Não ficou claro que o Coordenador será notificado sobre os compromissos que não foram realocados	Na proposta do UC, o Coordenador deve ser notificado se a realocação não for realizada
Não ficou claro como o principal responsável selecionará o usuário para a realocação do compromisso	A proposta do UC inclui detalhes da seleção do usuário para a tentativa de realocação
O usuário não é informado da solicitação de realocação enviada e da tentativa de realocação por meio do sistema	A proposta do UC mostra que a solicitação de realocação deve ser apresentada ao usuário, incluindo a tentativa de realocação pelo próprio sistema
A lista de usuários foi projetada de maneira diferente das regras de negócio	A lista de usuários está proposta de maneira diferente nas regras de negócio
A lista de usuários não é exibida	A proposta do UC mostra informações que os usuários capazes de realocar devem ser disponibilizados

Tabela 4.3: Análise das Informações nos Mockups em Comparação com o UC.

4.3.3.3 Análise da Comunicação Via Artefato

Nesta etapa, a conversa dos participantes foi analisada para verificar as potenciais falhas de comunicação observadas nos mockups. Triangulações entre os problemas de comunicabilidade nos casos de uso com falhas de comunicação foram realizadas.

Em relação à análise da conversa, as atividades realizadas pelos participantes que desenvolveram os mockups foram analisadas, identificando as potenciais falhas de comunicação. Um exemplo das atividades realizadas pelos consumidores P1 e P2 na Figura 4.5, incluindo a forma como foi realizada a análise da conversa dos participantes.

ATIVIDADES	FLUXO DE INTERAÇÃO DOS PARTICIPANTES																																						
Compreensão do Diagrama de Casos de Uso (Astah)	B	C*	B	D																							X												
Compreensão da Especificação de Caso de Uso (PDF)		B*	A*	D	E	F	G	E	H	B	F	X	E	X	E	F	F	X	F	G	F	F	E	F	F	F	G	L	M	H	F	E	F	G	M	E	F	M	B*
Construção de Protótipos (Balsamiq)													J	J	J	J	F	F	G	X	F	E	F	F	H	I	G	L	M	G	F	F	E	L	M	F	M		
A. Funcionalidades. B. Atores. C. Relacionamentos (include e extend). D. Pré-condições. E. Fluxo Principal. F. Fluxo Alternativo. G. Regras de negócio. H. Fluxo de Exceção. I. Compreensão de elementos da interface descritos no caso de uso. J. Análise de elementos no balsamiq para serem inseridos na interface. L. Comparação entre o caso de uso e os mockups criados. M. Correção para verificar a consistência dos protótipos com os casos de uso. *. Dúvida com o elemento. #. Relato de dúvida para um pesquisador. +. Uso de papel e caneta.																																							

Figura 4.5: Mapeamento de Atividades dos Participantes na Construção de Mockups a partir de Casos de Uso – P1 e P2.

A Tabela 4.4 apresenta os símbolos utilizados para registrar tal mapeamento. Por exemplo, a dupla com os participantes P1 e P2, na Figura 4.5, iniciaram a atividade compreendendo o diagrama de UC, discutindo sobre os atores (símbolo B) e relacionamentos (símbolo C) dos casos de uso. Percebeu-se que o símbolo C possui '*', indicando dúvida dos participantes sobre o relacionamento. Em seguida a dupla faz a leitura da especificação de UC, tendo dúvidas sobre os atores (símbolo B com o '*' que indica dúvida sobre o símbolo B, ou seja, os Atores). Para isto, a dupla consultava novamente o diagrama de UC e compreende a descrição dos atores relacionados ao UC (uma sequência de símbolos B* e B indicam o esclarecimento da dúvida). Caso contrário, uma sequência de A* e D na Figura 4.5, indica que a dúvida se mantém.

Categoria dos Símbolos	Descrição dos Símbolos
Diagrama e Especificação de UC	(A) Descrição das Funcionalidades relacionadas com o caso de uso; (B) Atores, (C) Relacionamentos, (D) Pré-condições, (E) Fluxo Principal, (F) Fluxo Alternativo, (G) Regras de negócio, (H) Fluxo de Exceção.
Construção dos Mockups	(J) Análise de elementos no Balsamiq para serem inseridos nos mockups,
Práticas Realizadas na Atividade	(I) Compreensão de elementos da mockups descritos no caso de uso (L) Comparação entre o caso de uso e os protótipos criados, (M) Correção para verificar a consistência dos protótipos com os casos de uso, e (N) Verificação da finalização da construção dos protótipos de acordo com o caso de uso, (X) Analisam os artefatos sem mencionar elementos específicos, (*) Discussões sobre dúvidas.

Tabela 4.4: Símbolos utilizados no Mapeamento de Atividades dos Participantes.

A análise dessa conversa foi importante não somente para a confirmação das falhas de comunicação a partir dos mockups. Com a análise da conversa dos participantes, observou-se outras falhas de comunicação não detectadas com a análise dos mockups com o apoio da ferramenta SFYI Message. Essas falhas de comunicação são apresentadas abaixo, com algumas considerações:

Os atores no diagrama da UC não foram totalmente compreendidos. Os participantes entenderam as informações completas desses atores a partir da especificação da UC. Os produtores descreveram o ator “Smartphone” no diagrama de UC e “GPS” e “Agenda” na especificação de UC como parte do “Smartphone”. Foi observado que os produtores pretendiam resumir os dois atores em apenas um ator no diagrama e detalhá-los

na especificação.

Os participantes não entenderam alguns termos na especificação do UC, como a diferença entre a notificação e os avisos enviados pelo sistema. A intenção dos produtores era que todos os consumidores entendessem a diferença desses termos.

A quantidade excessiva de informações e o formulário da organização na especificação da UC afetaram seu consumo. Foi observado que alguns participantes tiveram dificuldade em entender a especificação da UC devido à quantidade de informações e a forma como a especificação de UC foi organizada. Na especificação de UC foi necessária uma grande quantidade de informações devido ao objetivo do caso de uso, mas a sua organização poderia ser melhorada para não afetar seu consumo.

A Tabela 4.5 apresenta triangulações entre os problemas de comunicabilidade e as falhas de comunicação. Nesta tabela, são apresentadas somente as falhas de comunicação entre produtores e consumidores do UC. São descritas as falhas de comunicação observadas a partir dos mockups (com “M#” ao início da descrição) e as falhas de comunicação observadas na conversação dos participantes (com a letra “C#” ao início da descrição). Também são apresentados exemplos da conversação dos participantes que caracterizaram essas falhas de comunicação. Além disso, usamos a classificação de falhas de comunicação com base na Engenharia Semiótica, como completas (C), parciais (P) e temporárias (T). (coluna F na Tabela 4.5), conforme apresentado no início do Capítulo 3.

ID	Problemas de Comunicabilidade no UC	Descrição das Falhas de Comunicação	F	Citações
1	Falta da descrição dos atores na especificação de UC.	NÃO HOUVERAM FALHAS DE COMUNICAÇÃO A PARTIR DESSE PROBLEMA		
2	Falta da descrição de cada um dos agentes inteligentes na especificação de UC, como GPS e Agenda.	NÃO HOUVERAM FALHAS DE COMUNICAÇÃO A PARTIR DESSE PROBLEMA		
3	Falta de descrição das diferenças entre notificações, avisos e lembrete	C1: Os participantes não entenderam alguns termos na especificação do UC	C T	“Neste lembrete, você precisa ter alguns dados. Qual é a necessidade de colocar um aviso? (P8); No caso da notificação, acho que é quando a pessoa abre o telefone e há uma mensagem (P7)”
4	Falta de padronização na descrição dos agentes inteligentes no diagrama e na especificação de UC.	C2: Os atores no diagrama da UC não foram totalmente compreendidos.	P	“Fiquei em dúvida se o GPS avisa a todos os envolvidos (P7); Esse ator Smartphone faz o quê? (P8)”
5	A desorganização da especificação do UC pode impactar na compreensão das informações.	M1: Não ficou claro como o principal responsável selecionará o usuário para a realocação do compromisso	C	“Vamos criar uma tela para mostrar a notificação (P11); Vamos mostrar a lista de compromissos. A notificação mostrará uma lista de usuários (P13)”
		M2: As informações para desativar o botão de realocação após o final do tempo não estão claros no mockup	P	“O que podemos fazer com esse contador de tempo? (P9); Vamos colocar uma imagem com o tempo disponível (P10)”
		M3: Não ficou claro que o Coordenador será notificado sobre os compromissos que não foram realocados	C	“Existe uma tela de notificação? (P12); “[Deve haver] uma tela de notificação de confirmação empilhada (P11)”.
		M4: O tempo de conclusão da realocação não foi expresso, além das outras operações após esta	C	“Está certo? (P3); “Sim. Mas não é possível entender esse horário de recebimento? (P4)”.
		M5: O usuário não é informado da solicitação de realocação enviada e da tentativa de realocação por meio do sistema	C	“Tempo de sugestão finalizado. Exibir os detalhes do compromisso e o botão de realocação normalmente (P5); Sim (P6); Ok (P5)”.
		M6: A lista de usuários foi projetada de maneira diferente das regras de negócio	C	“Então você tem que colocar a seleção de um novo responsável. Ou seja, uma pessoa responsável que você irá sugerir (P17); Pode ser um menu para colocar esses usuários (P18); Mas o menu não será legal. Pensei em colocar o botão de realocação aqui (P17)”
		C1: A quantidade excessiva de informações e o formulário da organização na especificação da UC afetaram seu consumo	C T	“Eu já me perdi (P2); É aqui (P1); Está bem! Esse negócio de ir e voltar na especificação de casos de uso é irritante (P2)”.

Tabela 4.5: Falhas de Comunicação entre Produtores e Consumidores a partir do UC.

A partir da Tabela 4.5, o número de falhas de comunicação entre produtores e consumidores em relação aos problemas de comunicabilidade do UC foram agrupados.

A Figura 4.6 mostra o número de falhas de comunicação (M1 a M6; e C1 a C3 conforme apresentou a Tabela 4.5) relacionadas aos problemas de comunicabilidade (CP1 a CP5 como mostrou a Tabela 4.5).

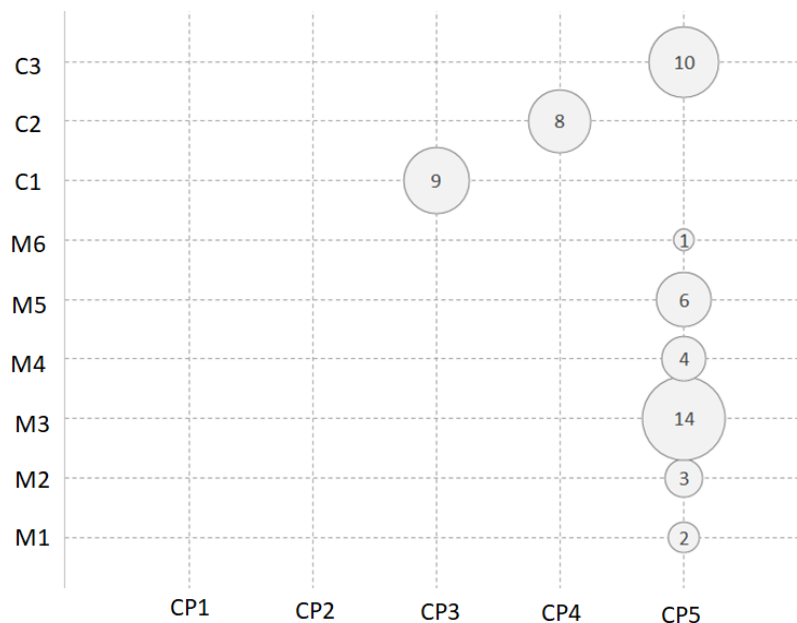


Figura 4.6: Número de Falhas de Comunicação Relacionadas aos Problemas de Comunicabilidade do UC.

Foi observado que nem todos os problemas de comunicabilidade identificados no UC causaram falhas de comunicação, pois os problemas CP1 e CP2 não causaram tais falhas. Foram observadas falhas de comunicação Temporárias e Completas, como C1. Portanto, os problemas de comunicabilidade no UC podem causar falhas de comunicação entre produtores e consumidores.

4.3.4 Limitações do Estudo

Existem limitações no estudo exploratório, como o fato dos participantes serem estudantes de graduação e o estudo ser conduzido em um ambiente acadêmico. Sobre isto, Fernandez *et al.* (2010) afirmam que estudantes que não têm experiência na indústria podem, no entanto, ter habilidades semelhantes aos profissionais menos experientes. Portanto, apesar da limitação imposta pela participação de estudantes e não de profissionais no estudo, acredita-se que os resultados encontrados não devem ser considerados inválidos. Em relação à comunicação verbal dos projetistas para os

participantes, caso os participantes tivessem a necessidade de solicitar esclarecimentos sobre os casos de uso que os impedissem de concluir suas atividades no estudo, dois destes estavam presentes durante a execução do estudo. Porém, nenhum dos participantes solicitou tais esclarecimentos e com isso, foi possível caracterizar os problemas de comunicabilidade entre os projetistas e os participantes através da análise mockups e da conversação entre os participantes.

Além disso, é importante ressaltar uma característica de todo estudo de caráter qualitativo, como é o caso deste. A pesquisa qualitativa não tem por objetivo a ‘replicabilidade’, seja de análise, seja de resultados, pois não trabalha com números, e sim com significados. A não replicabilidade está associada ao fato de que não se pode esperar que diferentes analistas tenham exatamente a mesma interpretação dos fatos e situações. Como significados resultam de interpretações, a replicabilidade não está em pauta em pesquisas qualitativas.

4.3.5 Discussão dos Resultados

Através da condução deste estudo exploratório foi possível identificar evidências para a questão de pesquisa “*Que problemas de comunicabilidade de casos de uso podem afetar a comunicação entre produtores e consumidores?*”. A seguir são discutidos apenas os problemas de comunicabilidade que afetaram a comunicação entre produtores e consumidores deste estudo:

(i) Falta de descrição das diferenças entre notificações, avisos e lembrete no sistema: Em relação às falhas de comunicação que ocorreram a partir desse problema de comunicabilidade, por exemplo, alguns participantes tiveram dificuldade em entender as diferenças entre esses termos, levando ao uso de elementos incorretos nos mockups. Como esses termos precisam de mais informações para que os consumidores possam compreender a diferença entre estes, seria interessante se os produtores tivessem adicionado informações para esclarecer tais diferenças.

(ii) Falta de clareza na descrição de agentes inteligentes no diagrama e especificação da UC: em relação às falhas de comunicação por causa desse problema de comunicabilidade, por exemplo, alguns participantes não entenderam que o ator “Smartphone” no diagrama da UC é representado como “GPS” e “Agenda” na especificação da UC, mas outros participantes entenderam. Em relação aos participantes

que entenderam essas informações, percebemos que era possível obter implicações conversacionais do “GPS” e “Agenda” relacionados ao ator “Smartphone” e regras de negócios. Seria interessante se os produtores tivessem adotado o mesmo padrão na descrição de agentes inteligentes ou tivessem anotado as diferenças no diagrama ou especificação da UC.

(iii) A desorganização na especificação de UC pode afetar o entendimento das informações – sobre falhas de comunicação, percebemos que a desorganização das informações prejudicou o consumo de casos de uso. Nesse caso, ocorrem falhas de comunicação porque os consumidores não compreenderam o objetivo do UC devido ao conteúdo desorganizado neste artefato. No entanto, a organização de informações dessa maneira é um padrão adotado pela equipe de desenvolvimento, como observamos em outros trabalhos que as equipes adotam seus próprios padrões na modelagem (Petre, 2013). No entanto, se esses casos de uso foram usados por novos funcionários para executar a manutenção, por exemplo, esse problema de comunicabilidade afetaria a compreensão destes consumidores.

4.4 Investigando a Comunicação de Produtores e Consumidores de Casos de Uso com o Apoio do Ferramenta SigniFYIng Message

O estudo *in vitro* apresentado nesta Seção foi realizado para compreender a intenção de comunicação dos produtores de artefatos e a compreensão do consumidor sobre os artefatos. Este estudo foi apresentado em Lopes *et al.* (2017). Na Seção anterior foi apresentado um estudo *in vitro* sobre a comunicação de produtores e consumidores de casos de uso. No entanto, não foi possível explorar a intenção dos produtores. Diante disto, o objetivo deste estudo foi responder à questão de pesquisa “*Como as intenções de produtores são expressas nos artefatos para que se estabeleça a comunicação com os consumidores?*”.

A ferramenta SFYI Message (de Souza *et al.*, 2016) foi usada para estruturar o conteúdo de parte do projeto de *metacomunicação*, isto é, a comunicação dos projetistas sobre como e por que usuários podem ou devem se comunicar com um sistema. A ferramenta auxiliou o contraste entre o pretendido pelo produtor e o interpretado pelo consumidor do artefato. Foram também empregadas as máximas do Princípio de

Cooperação de Grice (Grice, 1975) para analisar parte das condições de sucesso da comunicação entre produtores e consumidores.

As contribuições pretendidas nesta Seção também visam adicionar à literatura a importância da comunicabilidade de artefatos em relação à comunicação de equipes de desenvolvimento de software. Neste estudo, a comunicabilidade de artefatos foi explorada de uma forma diferente em relação aos métodos adotados na seção anterior.

4.4.1 Planejamento do Estudo

O estudo foi planejado para ser realizado em duas etapas. Na primeira etapa, os participantes produziram os casos de uso. Na segunda etapa, diferentes participantes consumiram informações dos casos de uso anteriormente produzidos.

Foram selecionados 34 participantes que eram alunos finalistas do curso de Ciência da Computação e tinham desenvolvido casos de uso para diferentes sistemas em disciplinas do curso. Foram definidas duplas de participantes para a realização das atividades do estudo, totalizando 17 duplas. A razão de fazer o estudo em duplas está relacionada com o intuito de promover a discussão dos casos de uso e a explicitação das intenções e significados pretendidos de quem criou e compreendeu estes artefatos. Os participantes tinham cursado uma disciplina introdutória de IHC e estavam cursando uma disciplina de Engenharia de Software durante a execução do estudo. Foram elaborados os formulários, com base na SFYI Message, para apoiar a análise da intenção de duplas produtoras (formulário de metacomunicação) e a compreensão de duplas consumidoras dos casos de uso (estrutura de metacomunicação).

4.4.2 Execução do Estudo

Na primeira etapa do estudo, foi solicitado aos participantes o desenvolvimento da especificação de apenas um caso de uso do diagrama de casos de uso, cujos sistemas eram de interesse dos participantes. Ressalta-se que cada parte desenvolvida dos sistemas nas especificações de casos de uso tratam de diferentes domínios, como aplicativos para entretenimento, educacionais e financeiros. As duplas desenvolveram os casos de uso e descreveram o contexto do software projetado em tal artefato. Ainda nesta etapa, as duplas produtoras preencheram o formulário de metacomunicação para expressar mais

explicitamente suas intenções e significados pretendidos quanto aos casos de uso para outros desenvolvedores (ver Figura 4.7).

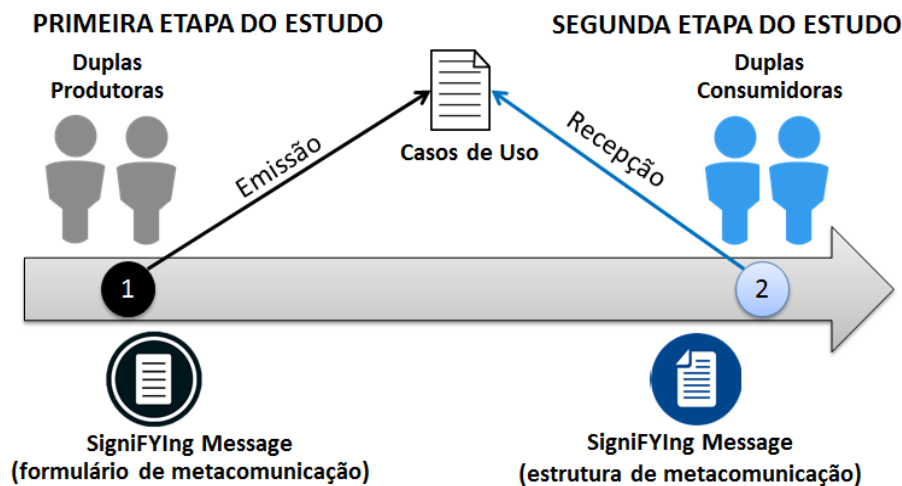


Figura 4.7: Etapas Realizadas no Estudo.

Na segunda etapa do estudo, foram entregues apenas as especificações de casos de uso para diferentes duplas interpretarem e descreverem o que tinham compreendido. Antes da entrega dos casos de uso para as duplas consumidoras, foi esclarecido aos participantes a mudança de seus papéis em relação ao desenvolvimento de software. Assim, na segunda etapa do estudo, os participantes representaram desenvolvedores de software, sendo estes agora as duplas consumidoras. As duplas consumidoras preencheram seu entendimento da comunicação dos produtores com o apoio do formulário de estrutura de metacomunicação. Nesta etapa, os consumidores não receberam o formulário de metacomunicação.

É importante frisar que os casos de uso consumidos não se referiam ao mesmo objeto dos casos de uso produzidos pelas duplas. As duplas consumidoras descreveram as informações que foram compreendidas para a codificação a partir destes artefatos produzidos. Ao final, os casos de uso e a estrutura de metacomunicação preenchida com base na interpretação das duplas consumidoras foram recolhidas.

4.4.3 Resultados do Estudo

Os casos de uso produzidos pelos participantes foram analisados e comparados com o formulário de metacomunicação. Em seguida foram triangulados os casos de uso, o

formulário de metacomunicação (de produtores) e a estrutura de metacomunicação (de consumidores). As adaptações das máximas de Grice (Grice, 1975) apresentadas na

Nosso objetivo com esta análise foi compreender a forma de expressão do UC, buscando identificar problemas que podem afetar a compreensão recíproca entre membros de uma equipe de desenvolvimento de software. Como resultados, foram identificados problemas de comunicabilidade do UC, apresentados na Tabela 4.2.

Tabela 4.1 foram utilizadas em todos os estágios de análise deste estudo.

4.4.3.1 Máximas de Grice Adaptadas para a Análise de Rupturas de Comunicação em Casos de Uso

As máximas violadas na escrita dos casos de uso deste estudo foram: Quantidade (Quant), Quantidade X Modo (QM), Quantidade X Relação (QR) e Qualidade (Quali). A Figura 4.8 apresenta a quantidade de máximas violadas pelas duplas produtoras.

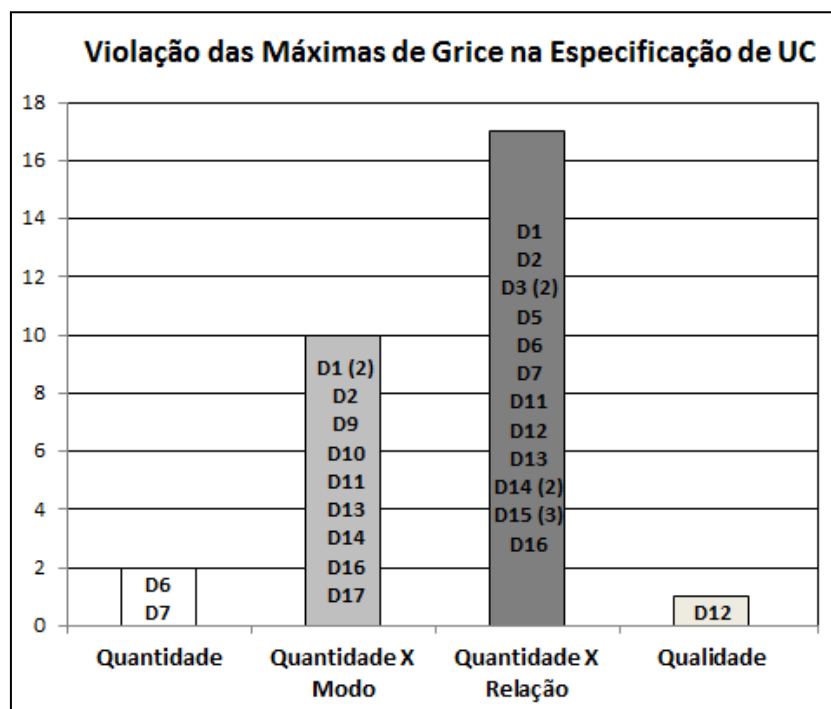


Figura 4.8: Total de Violações das Máximas de Grice na Especificação de Caso de Uso de Cada Dupla Produtora.

Na Tabela 4.6, por exemplo, a intenção dos produtores, a partir da leitura do formulário de metacomunicação preenchidos pelas duplas, não ficou clara em relação às representações necessárias nos casos de uso para os consumidores, resultando em um maior número de violações.

DP	Máximas violadas por duplas produtoras de Especificação de Casos de Uso (UC)
D1	- Omissão torna a especificação de UC ambígua - QM . - Fluxos alternativos que não estão referenciados no caso de uso, tornando o caso de uso desorganizado - QM . - Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR .
D2	- Omissão torna a especificação de UC ambígua - QM . - Fluxos alternativos que não estão referenciados no caso de uso, tornando o caso de uso desorganizado - QM . - Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR .
D3	- Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR . - Não foi descrita a pré-condição necessária - QR .
D4	- Não foi descrita a pré-condição necessária - QR .
D5	- Os detalhes do perfil do usuário não foram representados na descrição dos atores - QR .
D6	- Não foi descrito o nome do caso de uso - Quanti . - Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR .
D7	- Não foi descrito o nome do caso de uso - Quanti . - Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR .
D8	- Não foi descrito o contexto da especificação de casos de uso, o que compromete o entendimento do sistema - QR .
D9	- Fluxos alternativos que não estão referenciados no caso de uso, tornando a especificação de UC desorganizada - QM .
D10	- Fluxos alternativos que não estão referenciados no caso de uso, tornando o caso de uso desorganizado - QM . - Omissão torna a especificação de UC ambígua - QM .
D11	- Omissão torna a especificação de UC ambígua - QM .
D12	- Não foi descrita a pré-condição necessária - QR . - Os participantes consideraram um perfil de ator que não estava relacionado com o sistema - Quali . - Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR .
D13	- Fluxos alternativos que não estão referenciados no caso de uso, tornando o caso de uso desorganizado - QM . - Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR .
D14	- Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR . - Os detalhes do perfil do usuário não foram representados na descrição dos atores - QR . - Fluxos alternativos que não estão referenciados no caso de uso, tornando o caso de uso desorganizado - QM .
D15	- Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR . - Os detalhes do perfil do usuário não foram representados na descrição dos atores - QR . - Não foi descrita a pré-condição necessária - QR .
D16	- Não foram descritos os signos necessários para a compreensão da interface nas regras de negócio - QR . - Fluxos alternativos que não estão referenciados no caso de uso, tornando o caso de uso desorganizado - QM .
D17	- Fluxos alternativos que não estão referenciados no caso de uso, tornando o caso de uso desorganizado - QM .

Tabela 4.6: Descrição das Máximas de Grice Violadas por cada Dupla Produtora na Especificação de Casos de Uso.

Em seguida foi feita a triangulação dos casos de uso enviados pelas duplas produtoras, comparados ao formulário de metacomunicação e a estrutura de metacomunicação. Nossa atenção voltou-se para o entendimento das duplas consumidoras a partir da intenção manifesta das duplas produtoras. Ao final da análise

foram observadas: (i) máximas violadas que não prejudicaram o princípio cooperativo; (ii) máximas violadas que romperam o princípio cooperativo, resultando em rupturas de comunicação entre produtores e consumidores. As próximas subseções apresentam tal análise para os dois aspectos.

4.4.3.2 Princípio Cooperativo Estabelecido a partir da Especificação de Casos de Uso

A Figura 4.9 apresenta a quantidade de máximas de Grice que foram violadas pelas duplas produtoras e que ainda assim não impediram o estabelecimento do princípio cooperativo. Por exemplo, mesmo informações desorganizadas e que foram omitidas foram compreendidas (ver a Figura 4.10).

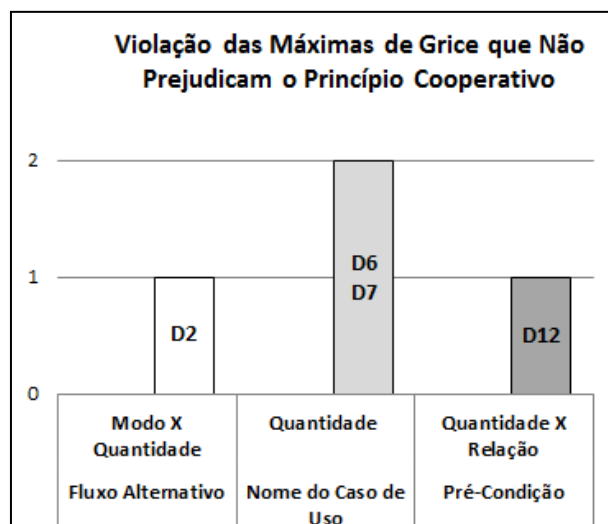


Figura 4.9: Violações das Máximas de Grice na Especificação de Casos de Uso que não Comprometeram o Princípio Cooperativo.

Em relação ao elemento fluxo alternativo, a dupla produtora D2 não usou as referências necessárias no fluxo principal do caso de uso (QM). No entanto, a dupla consumidora D6 conseguiu compreender a relação do fluxo alternativo a partir do fluxo principal, pois o domínio do sistema era conhecido. Em relação ao nome do caso de uso (Q) e pré-condição (QR), essas informações foram compreendidas a partir do fluxo principal. Percebe-se que a manutenção do princípio cooperativo neste caso se deve à implicatura conversacional a partir do elemento de fluxo principal.

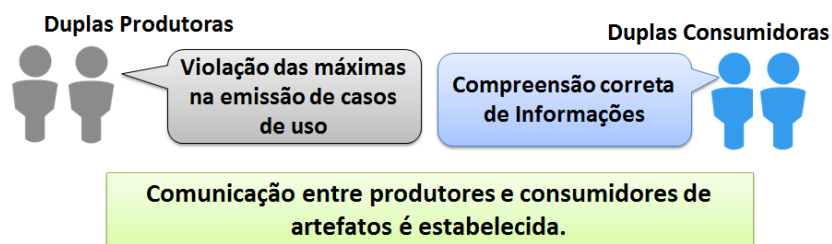


Figura 4.10: Princípio Cooperativo não Prejudicado q Partir de Máximas Violadas.

4.4.3.3 Princípio Cooperativo não Estabelecido a partir da Especificação de Casos de Uso

A Figura 4.11 apresenta a quantidade de máximas violadas pelas duplas produtoras que resultaram em falhas de comunicação a partir de casos de uso. Por exemplo, algumas duplas descreveram elementos de maneira incompleta, prejudicando a compreensão de informações.

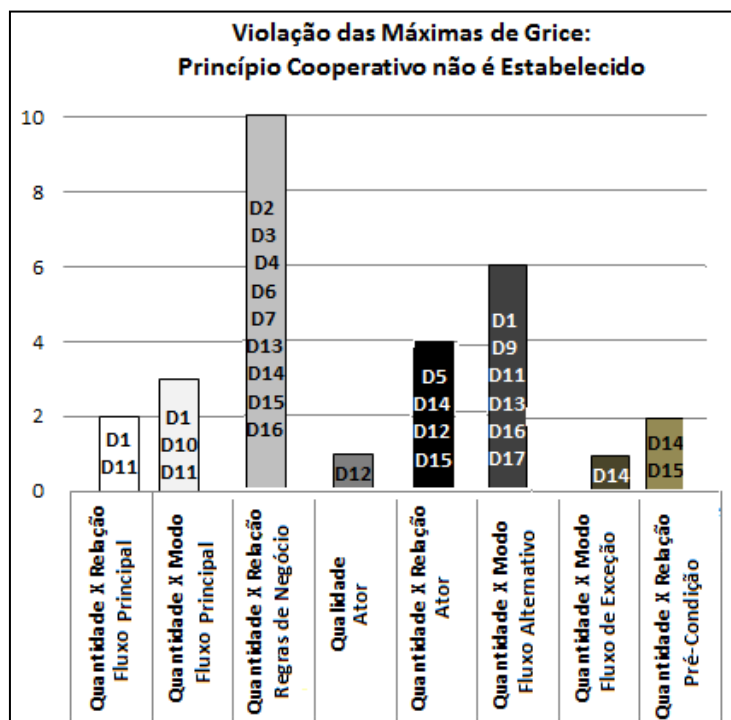


Figura 4.11: Violações das Máximas de Grice na Especificação de Casos De Uso que não Estabeleceram o Princípio Cooperativo.

Em relação ao elemento fluxo principal, houve a omissão dos passos no fluxo principal. Tais passos são relevantes para o consumidor da especificação de casos de uso (QR). Algumas omissões também tornaram a especificação de casos de uso ambígua

(QM). Sobre o elemento ator, a dupla consumidora D14 leu as informações da dupla produtora D12 e considerou um perfil de ator que não estava relacionado com o sistema (Quali). Os detalhes do perfil do usuário também não foram compreendidos a partir da descrição de alguns atores (QR). No que se refere ao fluxo alternativo e fluxo de exceção, algumas duplas fizeram a especificação destes elementos, mas não os relacionaram no fluxo principal, prejudicando a compreensão de duplas consumidoras. Já os elementos regras de negócio e pré-condição não foram compreendidos, pois informações necessárias não estavam descritas na especificação de casos de uso. Estas informações eram relevantes para a compreensão do sistema interativo (QR).

4.4.3.4 Consequências Observadas

A partir dos resultados expostos até aqui, observou-se que as máximas de Grice permitem inferir outros fatos. Nossa análise sugere que as violações das máximas de Grice, de maneira consciente ou não, podem resultar em defeitos de software. Esta conclusão resultou de uma análise com base nos diferentes tipos de defeitos apresentados por Granda *et al.* (2015), apresentada na Tabela 4.7.

Tipo	Descrição
Omissão	Informações necessárias omitidas.
Fato Incorreto	Algumas informações no artefato contradizem a lista de requisitos ou do conhecimento geral do domínio do problema.
Inconsistência	As informações em uma parte do artefato estão inconsistentes com outras.
Ambiguidade	As informações no artefato são ambíguas. Isto pode levar a diferentes interpretações de informações.
Informação Estranha	As informações fornecidas não são necessárias no artefato.
Redundância	Informações repetidas no artefato.

Tabela 4.7: Tipos de Defeitos (adaptados de Granda *et al.*, 2015).

A análise é apresentada na Tabela 4.8. A partir dos resultados apresentados na Tabela 4.8 percebe-se um mapeamento consistente o qual nos leva a concluir que a violação das máximas pode estar sistematicamente relacionada com a causa de defeitos de software. Ou seja, falhas de comunicação entre o produtor e o consumidor de artefatos pode modificar a intenção inicial de designers quanto à metacomunicação que pretendem

realizar para os usuários finais. Esta possibilidade afetaria negativamente a qualidade de sistemas interativos.

D#	Defeitos cometidos por duplas consumidoras
D1: D17	Omissão → Ambiguidade: <i>Omissão de passos no fluxo principal tornou a especificação ambígua.</i>
	Ambiguidade: <i>Fluxos alternativos não referenciados tornaram a especificação ambígua.</i>
	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
D2: D6	Omissão → Ambiguidade: <i>Omissão de passos no fluxo principal tornou a especificação ambígua.</i>
	Ambiguidade: <i>Fluxos alternativos não referenciados tornaram a especificação ambígua.</i>
	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
D3: D5	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
	Omissão: <i>Omissão da pré-condição necessária.</i>
D4: D1	Omissão: <i>Omissão da pré-condição necessária.</i>
D5: D8	Omissão: <i>O perfil do usuário não foi compreendido.</i>
D6: D9	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
D7: D11	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
D8: D2	- Não compreenderam a especificação de casos por falta detalhes sobre o contexto do sistema.
D9: D12	Ambiguidade: <i>Fluxos alternativos não referenciados tornaram a especificação ambígua.</i>
D10: D16	Omissão → Ambiguidade: <i>Omissão de passos no fluxo principal tornou a especificação ambígua.</i>
D11: D7	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
	Ambiguidade: <i>Fluxos alternativos não referenciados tornaram a especificação ambígua.</i>
D12: D14	Informação Estranha: <i>Os participantes consideraram um perfil de ator que não estava relacionado na especificação de casos de uso.</i>
	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
D13: D15	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
	Ambiguidade: <i>Fluxos alternativos não referenciados tornaram a especificação ambígua.</i>
D14: D4	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
	Omissão: <i>O perfil do usuário não foi compreendido.</i>
	Ambiguidade: <i>Fluxos de exceção não referenciados tornaram a especificação ambígua.</i>
D15: D13	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
	Omissão: <i>O perfil do usuário não foi compreendido.</i>
	Omissão: <i>Omissão da pré-condição necessária.</i>
D16: D3	Omissão: <i>Omissão de signos para a interface não foram compreendidos em regras de negócio.</i>
	Omissão: <i>O perfil do usuário não foi compreendido.</i>
D17: D10	Ambiguidade: <i>Fluxos alternativos não referenciados tornaram a especificação ambígua.</i>

Tabela 4.8: Descrição dos defeitos Cometidos por Cada Dupla Consumidora da Especificação de Casos De Uso.

4.4.4 Limitações do Estudo

Neste estudo exploratório, existem, porém, limitações. Em relação à falta de experiência da maioria dos participantes, Fernandez *et al.* (2010) afirmam que estudantes que não têm experiência na indústria podem, no entanto, ter habilidades semelhantes aos profissionais menos experientes. Nesta linha, apesar de o estudo ter sido conduzido em ambiente acadêmico, os participantes executaram atividades que são realizadas no

processo de desenvolvimento, ao construir artefatos de software e consumir informações para o desenvolvimento de outros artefatos.

Em relação à comunicação verbal dos produtores e consumidores de casos de uso neste estudo, caso os participantes tivessem a necessidade de solicitar esclarecimentos sobre os casos de uso que os impedissem de concluir suas atividades no estudo, foi solicitado que estes relatassem para os pesquisadores presentes durante a execução do estudo. Porém, nenhum dos participantes relatou impedimentos, sendo possível caracterizar os problemas de comunicabilidade entre os projetistas e os participantes com o apoio da ferramenta SFYI Message.

Outro aspecto desta mesma limitação é que as máximas violadas nos casos de uso também podem estar relacionadas à aprendizagem dos estudantes, uma vez que eles estavam cursando a disciplina de Engenharia de Software. Porém, uma vez mais, a interpretação de artefatos de maneira incompatível também causa defeitos na indústria de software. Portanto, embora reconhecendo a limitação imposta pela participação de estudantes, acredita-se que os resultados encontrados devam ser considerados válidos.

4.4.5 Discussão dos Resultados

Através dos resultados deste estudo foi possível compreender a natureza dos conteúdos pretendidos que estão ou deveriam estar em artefatos de desenvolvimento de software. Em relação à questão de pesquisa – *“Como as intenções de produtores são expressas nos artefatos para que se estabeleça a comunicação com os consumidores?”* – os resultados observados neste estudo ressaltam as interpretações incompatíveis, ou seja, as falhas de comunicação, que podem ser obtidas a partir de casos de uso devido à maneira em que as informações são expressas em artefatos, como os conteúdos que podem fornecer mais ou menos informações do que o necessário ou os conteúdos não explicitados que podem não ser percebidos pelos consumidores dos artefatos, mesmo que conscientemente pretendidos e reconhecidos pelo seu produtor.

Isso foi observado a partir da violação das máximas de Quantidade, Quantidade e Modo, Quantidade e Relação e Qualidade por parte dos produtores. Destes resultados, apenas uma das duplas, a D4, descreveu informações de maneira implícita de maneira proposital. A dupla descreveu que existiam opções de CRUD (*Create, Read, Update e Delete*) no fluxo principal sem detalhar cada operação, com a intenção de que estas

informações fossem compreendidas pelos consumidores. A descrição de informações implícitas de maneira proposital ou por falta de atenção das duplas produtoras foi constatada através da comparação entre as especificações de casos de uso e o respectivo formulário de metacomunicação preenchido por cada dupla produtora.

Algumas violações por falta de informações explícitas não causaram falhas na comunicação, como é o caso das duplas D2, D6, D7 e D12 que conseguiram compreender informações que deveriam estar no fluxo alternativo, nome do caso de uso e pré-condição a partir das informações no fluxo principal. Verifica-se aí o papel da implicatura conversacional. Com isto, estes resultados levam ao seguinte achado nesta pesquisa: *É possível obter implicatura conversacional de outros elementos a partir do fluxo principal de casos de uso.* No entanto, as demais violações levaram às falhas de comunicação, que também causaram defeitos de software.

4.5 Considerações sobre o Capítulo

Neste capítulo foram apresentados diferentes estudos para compreender o que pode afetar a comunicação entre produtores e consumidores de diferentes artefatos desenvolvidos nas etapas iniciais do desenvolvimento de software. As contribuições neste capítulo auxiliaram em um melhor entendimento da comunicação de produtores e consumidores a partir de artefatos.

O estudo *in vivo* mostrou que algumas práticas da equipe de software podem contribuir para a utilização de artefatos como meio de comunicação, sendo este importante para concentrar as informações que devem ser desenvolvidas em outros artefatos de software. Dois diferentes estudos *in vitro* demonstraram que a comunicabilidade de artefatos impacta na comunicação de produtores e consumidores. Com estes estudos foi possível observar diferentes falhas de comunicação a partir de artefatos, que podem auxiliar na definição dos riscos de falhas de comunicação.

CAPÍTULO 5 – DIRETIVAS DE COMUNICABILIDADE

Este capítulo apresenta as Diretivas de Comunicabilidade e três estudos experimentais realizados para sua avaliação.

5.1. Introdução

O objetivo da solução proposta nesta tese foi fundamentado nos resultados obtidos na etapa de ‘identificação do problema’ da metodologia seguida nesta pesquisa, apresentada no Capítulo 1. A partir do mapeamento sistemático, foram observadas oportunidades para o desenvolvimento de soluções que melhorem o conteúdo dos artefatos de software, de forma que esta solução também promova uma melhor comunicação sob a perspectiva de produtores e consumidores de artefatos (conforme apresentou a Figura 3.12 no Capítulo 3 para as oportunidades de pesquisa para a comunicação via artefato). Em relação aos estudos experimentais apresentados no Capítulo 4, foi observado que a comunicabilidade de artefatos impacta a comunicação de produtores e consumidores, sendo importante ter uma solução que auxilie na redução de problemas de comunicabilidade.

Diante disto, na etapa ‘objetivos da solução’, as Diretivas de Comunicabilidade (DCs) foram propostas para melhorar a comunicabilidade de artefatos de software com o objetivo apoiar a comunicação eficaz via artefato, como mostra a Figura 5.1. Isso poderia ser alcançado através da reflexão dos produtores de artefatos sobre o seu conteúdo para que haja redução dos riscos de falhas de comunicação e um aumento potencial de compreensão recíproca entre estes produtores e os demais membros de uma equipe de software.

As DCs são fundamentadas na Engenharia Semiótica (de Souza, 2005; de Souza *et al.*, 2016), no Princípio de Cooperação de Grice (Grice, 1975). Além disso, os resultados dos estudos experimentais apresentados no Capítulo 3 também forneceram ideias para o desenvolvimento desta proposta.

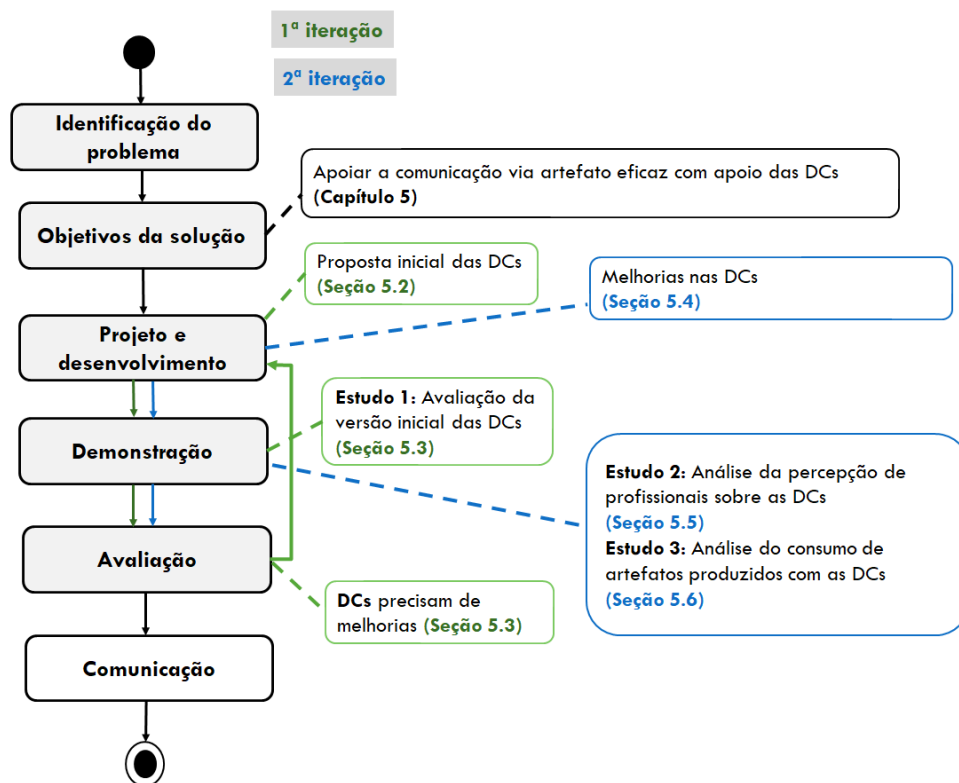


Figura 5.1: Proposta das Diretivas de Comunicabilidade e Três Estudos Iniciais.

A proposta inicial das DCs, desenvolvida na etapa ‘projeto e desenvolvimento’, foi avaliada em um primeiro estudo experimental da etapa de ‘demonstração’ da metodologia seguida. Os resultados deste estudo mostraram que elas promoveram a redução de riscos de falhas de comunicação, o que contribui para uma melhor comunicabilidade, como mostra a Figura 5.1. No entanto, percebeu-se oportunidades de melhorias na etapa de ‘avaliação’. Com isso, a partir da etapa ‘avaliação’ retornou-se para a etapa ‘projeto e desenvolvimento’ com o objetivo de melhorar a proposta das DCs, considerada como uma segunda iteração na metodologia. Foram realizadas melhorias na versão das DCs, a qual foi avaliada em um segundo estudo experimental com o objetivo de analisar a percepção de profissionais sobre o seu uso, conforme mostra a Figura 5.1. Os resultados obtidos neste segundo estudo experimental mostraram que as DCs continuaram contribuindo para a redução de riscos de falhas de comunicação, promovendo uma melhor comunicabilidade dos artefatos. A partir das percepções dos participantes, foi observado que as DC podem beneficiar a comunicação via artefato.

Com o objetivo de obter evidências a respeito das contribuições das DCs para a comunicação via artefato, foi realizado mais um estudo com o objetivo de analisar o consumo de artefatos produzidos com e sem o apoio das DCs, como mostra a Figura 5.1. Os resultados mostraram que os consumidores dos artefatos produzidos com o auxílio das DCs promoveram uma menor quantidade de falhas de comunicação, comparados aos consumidores dos artefatos produzidos sem as DCs. Assim, pode-se afirmar que as DCs promoveram uma melhor aproximação para apoiar a comunicação eficaz via artefato.

5.2. Proposta Inicial das Diretivas de Comunicabilidade

A proposta das DCs foi baseada na adaptação das quatro máximas do Princípio Cooperativo de Grice (Grice, 1975). A ideia de usar as Máximas de Grice na proposta das DCs também está relacionada com o segundo estudo exploratório apresentado no Capítulo 4. Com base nos resultados do segundo estudo exploratório foram observados problemas de comunicabilidade no artefato investigado em relação à falta de informações para um melhor esclarecimento de elementos representados (que está relacionado à Máxima de Quantidade), falta de clareza na descrição de informações (que está relacionada à Máxima de Modo) e a desorganização de informações (que está relacionada à Máxima de Relevância). As diretivas apresentadas na Tabela 5.1 foram propostas e consideradas mais adequadas para a comunicabilidade de artefatos. Nesta primeira versão da proposta, a interação entre todas as Máximas de Grice não foram pensadas para a criação das diretivas.

O emprego das DCs foi projetado com base no template de *metacomunicação* da Engenharia Semiótica (De Souza *et al.*, 2016), pois estas diretivas podem auxiliar os produtores a pensar nos consumidores antes do desenvolvimento do artefato. Além disso, os resultados do terceiro estudo exploratório, apresentado no Capítulo 4, mostraram que, quando os produtores não refletem sobre a sua comunicação com os consumidores, podem ocorrer falhas de comunicação. Além disso, foi observada uma oportunidade de pesquisa para investigar a comunicação recíproca entre produtores e consumidores de artefatos no mapeamento sistemático apresentado no Capítulo 3. O template original de *metacomunicação* da Engenharia Semiótica foi adaptado para:

“Este é o meu entendimento, como produtor de artefatos, de quem você, consumidor, é (para quem estou projetando o artefato), do que aprendi sobre o que você precisa fazer

no desenvolvimento do sistema (sobre o quê deve ser abordado no artefato). Esta, portanto, é a solução do sistema que projetei para você realizar as suas atividades”.

DCs	Descrição
DC1	As informações no artefato não devem conter conteúdo de algo para o qual não exista evidência adequada (<u>máximas de Qualidade</u>).
DC2	As informações necessárias no artefato devem ser descritas de maneira que sejam compreendidas pelos consumidores (<u>máxima de Quantidade</u>).
DC3	Não devem ser introduzidas informações que não vêm ao caso no artefato (<u>máxima de Relação</u>).
DC4	As informações no artefato devem estar organizadas (<u>máximas de Relação</u>).
DC5	As informações devem ser descritas de maneira clara para uma melhor compreensão das informações (<u>máxima de Modo</u>).
DC6	As informações no artefato devem ser descritas com a quantidade necessária a fim de evitar a ambiguidade na interpretação dos consumidores (<u>máximas combinadas de Quantidade e Modo</u>).

Tabela 5.1: Versão Inicial das Diretivas de Comunicabilidade.

Com base no template de *metacomunicação* da Engenharia Semiótica, foram desenvolvidas as seguintes questões para auxiliar os produtores a refletirem sobre os consumidores do artefato antes do uso das DCs:

(i) Para quem se está projetando o artefato? – “*O conteúdo do artefato pode ser compreendido para que o consumidor realize seus objetivos?*” - com o objetivo de apoiar o produtor a refletir se nas informações do artefato que podem ser compreendidas por seus consumidores, como desenvolvedores e gerentes, ou apenas desenvolvedores.

(ii) Sobre o que se está abordando no artefato? – “*Qual o conteúdo deve ser abordado sobre o domínio do problema/solução do sistema no artefato?*” - com o objetivo de estimular o produtor a refletir sobre o conteúdo que ele deseja que seja compreendido a partir do artefato, como as tarefas que um usuário pode realizar no sistema. A Figura 5.2 apresenta uma caracterização das etapas necessárias para a reflexão dos produtores a partir das questões fundamentadas na Engenharia Semiótica (etapa 1 na Figura 5.2) e do uso das DCs para algum artefato de software (etapa 2 na Figura 5.2).

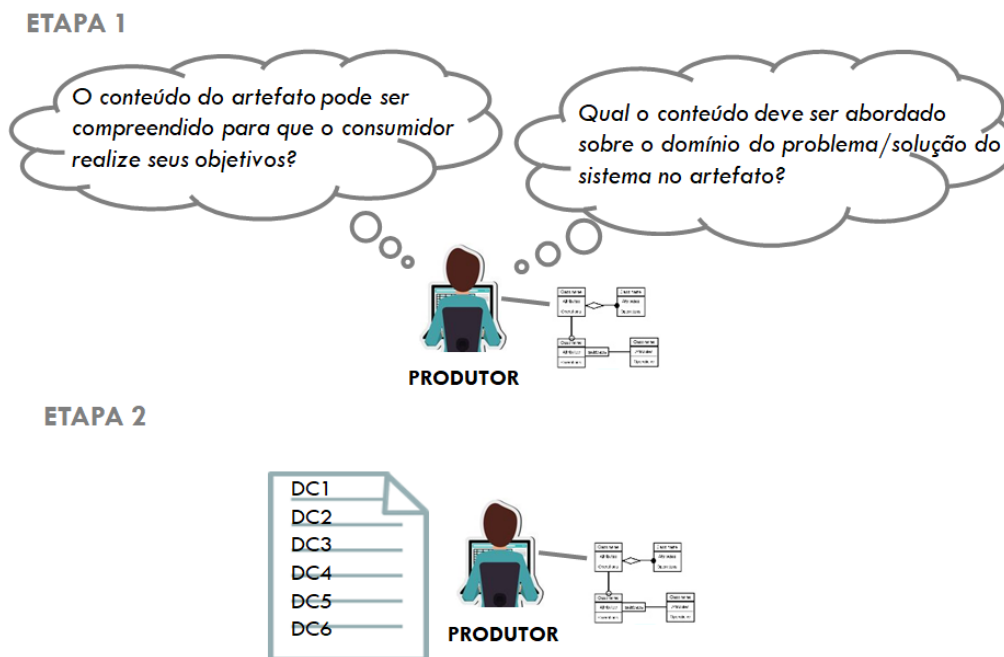


Figura 5.2: Exemplo de uso das Diretivas de Comunicabilidade.

5.3. Primeiro Estudo Experimental: Análise da Proposta das DCs

Esse estudo foi apresentado em Lopes *et al.* (2019c), um estudo *in vitro* para avaliar a versão inicial das DCs com o objetivo de compreender seu uso na produção de artefatos com o objetivo de reduzir os riscos de falhas de comunicação. Neste estudo foi analisado o uso das DCs para a produção de diagramas com diferentes propósitos e que os participantes tivessem conhecimento sobre a modelagem destes. Os diagramas explorados nesse estudo foram os diagramas classes da UML (OMG, 2015), *Business Process Modeling and Notation* (BPMN) (OMG, 2014) e *Interaction Flow Modeling Language* (IFML) (Brambilla e Fraternali, 2014). Estes diagramas possuem os seguintes propósitos de comunicação durante o desenvolvimento de software:

- **Classes da UML** – comunicar as informações que o sistema deve representar e gerenciar em relação ao domínio do problema.
- **BPMN** - comunicar informações sobre os passos a serem seguidos nas diferentes atividades do sistema, além dos perfis envolvidos e dos produtos resultantes.
- **IFML** - comunicar informações sobre os aspectos essenciais da interação do usuário com a interface da aplicação independentemente da plataforma de implementação.

Os diagramas de Classes, BPMN e IFML podem ser consumidos por diferentes profissionais para o desenvolvimento do sistema, por exemplo: McBride (2014) relata que BPMN pode ser usado para compreender o problema antes de sua codificação entre analistas de negócios e equipes de desenvolvimento. Brambilla e Fraternali (2014) relatam que através de diagramas IFML a interface pode ser codificada, além desta ser discutida por designers e desenvolvedores. As classes são comumente utilizadas pela equipe de desenvolvimento para a compreensão do domínio do problema antes de sua codificação (Reggio *et al.*, 2014).

5.3.1 Planejamento do Estudo

O estudo foi planejado para ser realizado com 24 alunos de graduação do curso de Ciência da Computação e Engenharia de Software da UFAM. Os alunos haviam cursado as disciplinas de Introdução à Engenharia de Software (IES) e Análise e Projeto de Sistemas (APS). Os alunos foram caracterizados como projetistas novatos, pois possuíam apenas experiência acadêmica com a modelagem de sistemas. Além disso, os participantes realizaram a modelagem em duplas. Tais alunos tinham conhecimento sobre a modelagem de diagramas de classe, BPMN e IFML.

Foram analisados dois grupos de modelagem, o Grupo DCs e o Grupo de Controle (grupo que não utilizou as DCs). Os grupos foram balanceados pelas notas de provas obtidas nas disciplinas de IES e APS. Cada grupo possuía seis duplas. Foram definidas as duplas para que os participantes pudessem discutir melhor suas intenções sobre a modelagem. Ambos os grupos utilizaram o mesmo cenário do problema para a modelagem dos artefatos. O cenário do problema de modelagem estava relacionado ao processo de submissão de artigos em uma aplicação web. Também foi utilizado um questionário pós-estudo para a compreensão da percepção dos participantes que utilizaram as DCs. O cenário do problema está disponível no Apêndice B.

5.3.2 Execução do Estudo

Inicialmente, ambos os grupos estavam em uma mesma sala para receberem as informações sobre a realização da modelagem com foco na redução de falhas de comunicação. Em seguida, os participantes do Grupo de Controle foram liberados do treinamento. Seguindo com o processo de treinamento, todos os participantes do Grupo

DCs receberam um treinamento de aproximadamente 20 minutos sobre o uso das diretivas. Neste treinamento, foram apresentados exemplos de uso das DCs e das questões fundamentadas na Engenharia Semiótica. Após isso, todos os participantes tiveram um prazo de sete dias para a entrega dos diagramas. O estudo foi conduzido de maneira por ser semelhante às atividades realizadas por projetistas na indústria de software, em que estes possuem um prazo para realizar a modelagem.

Ao entregarem os diagramas, todos os participantes responderam o questionário pós-estudo online. Os questionários tinham as seguintes questões: “**Q1.** *Qual diagrama foi desenvolvido primeiro?*”; “**Q2.** *Um dos diagramas foi usado como base para a modelagem de outro?*”. Nós utilizamos estas questões para compreendermos a maneira como foi o desenvolvimento da modelagem. O Grupo DCs tinha uma questão extra para compreendermos a percepção dos participantes com tais diretrizes, sendo esta: “**Q3.** *Qual a sua percepção com as diretivas de comunicabilidade?*”.

5.3.3 Resultados do Estudo

Antes de analisar os diagramas produzidos pelos participantes, o contexto em que os participantes realizaram a modelagem foi analisado a partir das questões Q1 e Q2 do questionário online, sumarizadas na Tabela 5.2. As duplas D1 a D6 realizaram a modelagem com o emprego das DCs e as duplas D7 a D12 realizaram a modelagem sem o uso de tais diretrizes. Apenas uma dupla de cada grupo não respondeu o questionário, mas estes enviaram os diagramas e suas percepções sobre a atividade em um relatório.

No que se refere ao contexto da modelagem, alguns participantes realizaram a modelagem em paralelo para compreender as informações relacionadas ao projeto do sistema, como as duplas D5, D7 e D11. Outros começaram a modelagem com o diagrama BPMN, como as duplas D1, D3 e D8, pois afirmaram que com este diagrama é possível compreender as atividades que são realizadas no sistema e assim é possível realizar mais facilmente a modelagem dos demais diagramas.

Em relação às duplas que iniciaram a modelagem com o diagrama de classes, tais diagramas não foram utilizados como base para o desenvolvimento dos demais. No entanto, percebeu-se que a maioria dos diagramas estavam consistentes com o conteúdo que pode ser relacionado nos diferentes diagramas, como classes do domínio (do diagrama de classes) que estavam relacionadas com as tarefas que podem ser executadas

(no diagrama BPMN). Assim, o contexto de modelagem adotado pela maioria dos participantes pode não ter influenciado na produção dos diferentes modelos.

Duplas	Q1	Q2
D1	BPMN	<i>“O BPMN foi usado como base. Ele ajudou na definição de atributos de algumas classes de domínio e na quantidade de telas e sequência de interação do IFML”</i>
D2	Classes	<i>“Não”</i>
D3	BPMN	<i>“Caso não tivéssemos usado o BPMN, provavelmente não conseguiríamos identificar todas as classes de domínio do problema”</i>
D4	Classes	<i>“Os diagramas não foram usados como base um para o outro, a base utilizada foi a descrição do problema”</i>
D5	Classes e BPMN	<i>“O diagrama de classe e BPMN desenvolvidos e a partir disso construímos o diagrama IFML”</i>
D6	-	-
D7	Classes e BPMN	<i>“Primeiro tentamos o diagrama de classes, como surgiram algumas dúvidas, começamos a fazer o BPMN para ajudar mais no que estava envolvido no processo”.</i>
D8	BPMN	<i>“Foi utilizado como base o BPMN para modelar o diagrama de classes e IFML. A modelagem do diagrama tornou-se mais fácil para compreender a interação”</i>
D9	Classes	<i>“Foram feitos totalmente independentes”</i>
D10	Classes	<i>“Os três foram feitos independente”</i>
D11	Classes e BPMN	<i>“O de classes ajudou em relação aos atributos, e o BPMN ajudou na compreensão das atividades”</i>
D12	-	-

Tabela 5.2: Versão Inicial das Diretivas de Comunicabilidade.

Em seguida, os artefatos foram analisados em relação aos riscos de falhas de comunicação, isto é, os problemas de comunicabilidade nos artefatos do ponto de vista da pesquisadora principal, os quais foram discutidos com a orientadora desse trabalho. Estes riscos também foram analisados quanto à qualidade dos artefatos. Essa análise foi baseada na classificação de defeitos apresentado por Granda *et al.* (2015), que foi apresentada na Tabela 4.7.

5.3.3.1 Análise do Diagrama de Classes

A Tabela 5.3 apresenta a descrição destes riscos com a classificação de possíveis falhas de comunicação (coluna SD), tipos de defeitos que podem ocorrer caso estes riscos se concretizem e o número de riscos cometidos pelos participantes de cada grupo (GD – Grupo DCs e GC – Grupo de Controle). A avaliação dos diagramas de classes resultou em um total 17 riscos para o Grupo DCs e 28 riscos para o Grupo de Controle.

SD	Defeitos que afetam a qualidade	Descrição dos riscos	GD	GC
Completa	Omissão	Falta de classes do domínio do problema	8	9
	Informação Estranha	Classes que não fazem parte do contexto do domínio do problema	0	1
	Informação Estranha	Relacionamentos que não fazem parte do contexto do domínio do problema	0	1
	Informação Estranha	Atributos que estão fora do domínio do problema	0	4
	Omissão	Falta de atributos importantes nas classes	0	1
	Informação Estranha	Cardinalidade que não fazem parte do contexto do domínio do problema	0	4
	Omissão	Falta de relacionamentos entre as classes	9	6
	Ambiguidade	Classes com descrições ambíguas	0	1
Temporária	Fato Incorreto	Relacionamentos errados na sintaxe	1	1

Tabela 5.3: Riscos de Falhas de Comunicação em Diagramas de Classes

A Figura 5.4 mostra o total de riscos relacionados às possibilidades de concretizar falhas de comunicação em Temporária, Parcial e Completa. De uma maneira geral, apesar dos participantes do Grupo DC terem utilizado as diretivas, os riscos relacionados às falhas Completas poderiam ser mitigados com as DCs da seguinte forma: **DC1** – análise de informações que estão fora do cenário do problema; e **DC2** - análise de informações necessárias a partir do cenário do problema para o diagrama de classes. Sobre o risco relacionado à falha Temporária, esses poderia ser mitigado com: **DC3** – correção de erros de sintaxe, causando incoerência na organização de informações no diagrama.

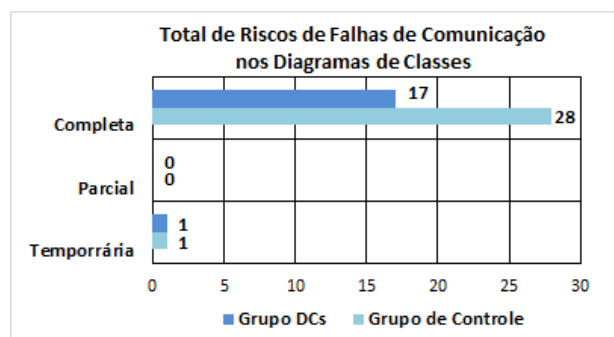


Figura 5.3: Total de Riscos de Falhas de Comunicação nos Diagramas de Classes.

A maioria dos riscos identificados podem causar falhas de comunicação Completa, os quais podem ocorrer devido à falta de contexto para a comunicação entre

produtores e consumidores. Também foram identificados riscos que podem causar falhas Temporárias, estes estão relacionados à erros de sintaxe. Isto foi considerado como possíveis falhas Temporárias devido ao conhecimento do consumidor em relação à linguagem de modelagem, o que não os leva à inferir informações incorretas.

5.3.3.2 Análise do Diagrama BPMN

A Tabela 5.4 apresenta a descrição destes riscos identificados nos diagramas BPMN. A interpretação da a Tabela 5.4 é semelhante à Tabela 5.3. A avaliação dos diagramas BPMN resultou em um total 12 riscos para o Grupo DCs e 17 riscos para o Grupo de Controle. A Figura 5.4 mostra o resultado total dos riscos identificados nos diagramas de BPMN de acordo com o tipo de falha de comunicação que poderia ocorrer, sendo apenas falhas Completas.

SD	Defeitos que afetam a qualidade	Descrição dos riscos	GD	GC
Completa	Omissão	Falta de algumas atividades no diagrama	3	3
	Inconsistência	Sequência incorreta de atividades no diagrama	0	1
	Informação Estranha	Atividades que estão fora do domínio do problema	2	3
	Omissão	Não foram representados elementos para informações importantes no diagrama	7	10

Tabela 5.4: Riscos de Falhas de Comunicação nos Diagramas BPMN.

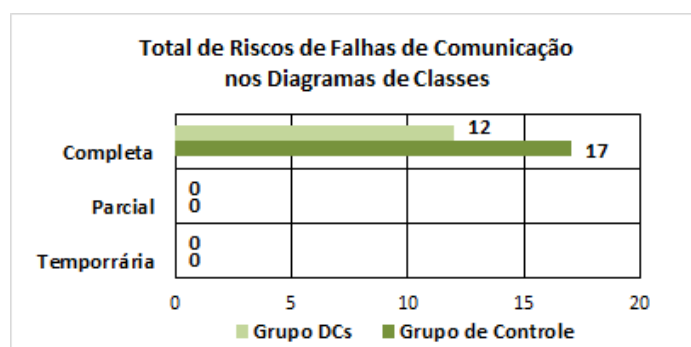


Figura 5.4: Total de Riscos de Falhas de Comunicação nos Diagramas BPMN.

Os riscos relacionados às falhas Completas poderiam ser mitigados com: **DC1** – análise das atividades que não fazem parte do domínio do problema; **DC2** – análise da

falta de atividades necessárias do cenário do problema para o diagrama BPMN; e **DC3** – análise da sequência lógica de atividades.

5.3.3.3 Análise do Diagrama IFML

A Tabela 5.5 apresenta a descrição dos riscos identificados nos diagramas IFML. A interpretação da Tabela 5.5 é semelhante à Tabela 5.3. A avaliação dos diagramas IFML resultou em um total 5 riscos para o Grupo DCs e 11 riscos para o Grupo de Controle. A Figura 5.5 mostra o resultado total dos riscos identificados nos diagramas IFML de acordo com o tipo de falha de comunicação que poderia ocorrer, sendo também neste tipo de diagrama apenas falhas Completas.

SD	Defeitos que afetam a qualidade	Descrição dos riscos	GD	GC
Completa	Omissão	Faltou detalhar a parte do cadastro dos usuários	2	4
	Omissão	Faltou detalhar a parte do envio de convites aos usuários	3	4
	Ambiguidade	Faltou detalhar campos específicos, prejudicando a compreensão da interface	0	1
	Inconsistência	Inconsistência em funcionalidades de diferentes perfis	0	1
	Ambiguidade	Faltou detalhar a sequência de atividades que devem ser realizadas antes da execução de uma atividade específica	0	1

Tabela 5.5: Riscos de Falhas de Comunicação nos Diagramas IFML.

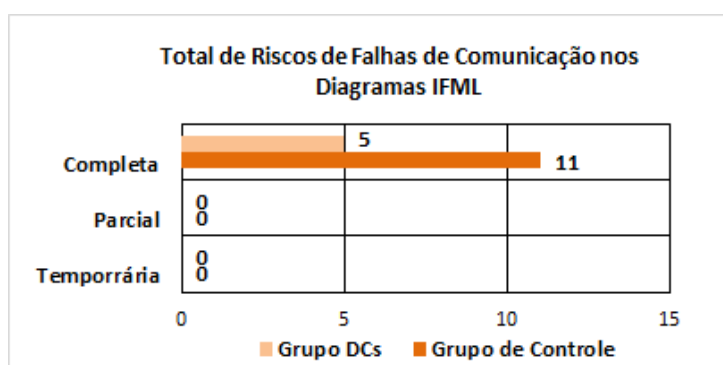


Figura 5.5: Total de Riscos de Falhas de Comunicação nos Diagramas IFML.

Os riscos relacionados às falhas Completas poderiam ser mitigados com: **DC1** – análise das funcionalidades para os diferentes perfis, pois os participantes usaram informações que não são verdadeiras a modelagem; **DC2** – análise da falta de tarefas do

usuário com o sistema especificadas no cenário do problema; **DC3** - análise da sequência de atividades que devem ser realizadas antes da execução de uma atividade específica.

5.3.3.4 *Análise do Uso das Diretivas de Comunicabilidade*

Em relação ao emprego das DCs, através do relato dos participantes no questionário pós-estudo, percebeu-se que os participantes as utilizaram para decisões de projeto e princípios de modelagem. As *decisões de projeto* referem-se à maneira como os produtores devem representar as funcionalidades a partir do cenário do problema. Os *princípios de modelagem* referem-se à maneira como os elementos dos diferentes diagramas são utilizados para representadas a intenção dos produtores. A Tabela 5.6 sumariza o modo em que as duplas empregaram as DCs.

Duplas	Uso das DCs	
	Princípios de Modelagem	Decisões de Projeto
D1	X	X
D2	X	-
D3	X	-
D4	X	-
D5	X	X
D6	X	-

Tabela 5.6: *Emprego das Diretivas de Comunicabilidade na Modelagem.*

As duplas D1 e D5 utilizaram as DCs tanto para decisões de projeto e princípios de modelagem (coluna Uso das DCs). A D1, por exemplo, afirmou a seguinte citação que compreendemos como decisões de projeto: *“Com apoio das diretivas, colocamos algumas informações complementares nos diagramas para garantir as funcionalidades do sistema”*. Em relação aos princípios de modelagem, a D1 citou: *“Acreditamos que as diretivas ajudam de maneira significativa na modelagem, evitando ambiguidade, inconsistência. São boas práticas para um profissional durante a modelagem”*. A maioria dos participantes utilizaram as DCs com a perspectiva de guia de modelagem e como auxílio na construção de artefatos. Sobre as citações dos participantes sobre as DCs como guia de modelagem, a D5 relatou: *“As diretivas são boas para a orientação durante a modelagem... Foi tranquilo utilizar as diretivas”*. Em relação à verificação do artefato,

D6 relatou: *“Acreditamos que o potencial das diretivas seria melhor alcançado se um profissional que não elaborou a modelagem realizasse a inspeção”*.

Sobre o uso das DCs, as duplas D4 e D1 mencionaram o uso das diretrizes para os diagramas de classes e BPMN. Nenhum dos participantes mencionou como eles usaram as DCs para modelar diagramas IFML.

Sobre a modelagem de diagramas de classes, em relação à DC1 (informações verdadeiras), D4 relatou: *“Procuramos não adicionar classes fora do domínio do problema. Nós tínhamos definido uma classe para o Relatório, mas ao olhar a heurística concordamos que era possível que o sistema gerasse o relatório e que a classe criada era desnecessária”*. D4 também relatou: *“Nós tomamos cuidado de colocar descrições em todos os relacionamentos entre classes, bem como as respectivas cardinalidades em todos eles”*. Percebe-se que a D4 relatou o uso da DC2 para que os consumidores pudessem compreender as informações relacionadas ao artefato. A D4 buscou utilizar apenas informações verdadeiras do domínio do problema. A dupla também relatou o seguinte sobre a DC5: *“Revisamos todos os relacionamentos que definimos, tanto em cardinalidade quanto em descrição (nome). Isso auxiliou a resolver uma ambiguidade nos relacionamentos das classes Coordenador e Revisor com a classe Convite”*. Sobre diagramas BPMN, D1 relatou sobre a DC2: *“O sistema precisava desses dados para atingir a meta, então adicionamos essas informações”*. Observamos que esses pares usavam apenas as informações necessárias do domínio do problema.

Em relação ao emprego das DCs, a maioria dos participantes considerou fácil a sua utilização na produção de artefatos. No entanto, algumas dificuldades sobre o uso das DCs também foram relatadas, por exemplo, a D6 e D2 relataram o seguinte, respectivamente: *“Consideramos redundantes algumas diretivas e achamos que poderiam ser sintetizadas”* e *“As diretivas 1 e 4 são ambíguas”*. Com isso, percebemos que as diretivas propostas também precisam de melhorias.

5.3.4 Limitações do Estudo

Como em todo estudo, neste existem limitações que podem afetar a sua validade, como a expectativa do primeiro principal em relação aos benefícios das DCs na modelagem para a melhoria da comunicação entre produtores e consumidores. A análise das diretivas aplicadas nos diagramas foi revisada por mais duas pesquisadoras. Os

resultados apresentados abordam apenas a modelagem realizada pelos participantes e a percepção destes com as DCs.

Outra limitação é o fato de os participantes serem estudantes de graduação e o estudo ser conduzido em um ambiente acadêmico. No entanto, estudantes podem ter habilidades semelhantes aos profissionais menos experientes (Fernandez *et al.*, 2012). Nesta linha, apesar de o estudo ter sido conduzido em ambiente acadêmico, os participantes executaram a modelagem de um cenário real de desenvolvimento de sistema. Além disso, as DCs também são voltadas para profissionais novatos em modelagem de software. Portanto, a amostra reflete a população do estudo.

Por fim, esses resultados foram obtidos a partir da modelagem de um único sistema web. Portanto, os resultados não são generalizáveis, mas são considerados válidos para o contexto do projeto de tal sistema.

5.3.5 Discussão dos Resultados

Em relação ao problema investigado neste trabalho, a partir dos resultados deste estudo Lopes *et al.* (2019c)⁶, percebeu-se que as DCs auxiliaram os produtores na reflexão do conteúdo dos artefatos que serão consumidos durante o desenvolvimento de software. O uso das DCs influenciou na redução dos riscos de falhas de comunicação dos artefatos produzidos pelo Grupo DCs em comparação com o Grupo de Controle. Percebeu-se também que as DCs podem auxiliar na redução de defeitos cometidos por falhas de comunicação.

O emprego das DCs na modelagem forneceu direcionamento para o conteúdo adequado dos artefatos a partir dos princípios de modelagem adotados pelos participantes e também auxiliam na reflexão sobre decisões de projeto. Em relação ao apoio nas decisões de projeto, este benefício pode ser alcançado quando as DCs são utilizadas como guias na modelagem.

Em relação ao uso das DCs, a maioria dos participantes considerou fácil utilizar tais diretivas. No entanto, alguns participantes consideraram as DCs redundantes e ambíguas.

⁶ Nota de esclarecimento: Nesta publicação, houve uma falha da minha parte em esclarecer que este estudo foi realizado com a versão inicial das Diretivas de Comunicabilidade. A versão citada em tal publicação inclui as melhorias realizadas nas diretivas após a execução deste primeiro estudo, a qual é apresentada na Seção 5.4. Ambas as versões possuem o mesmo embasamento teórico.

Isso mostrou a necessidade de realizar melhorias na proposta inicial. Além disso, as pesquisadoras também refletiram sobre a necessidade de realizar uma melhor combinação das quatro máximas de Grice após discussões com uma pesquisadora colaboradora desta pesquisa, especialista em Engenharia Semiótica.

5.4. Melhorias na Versão das Diretivas de Comunicabilidade

Para a realização das melhorias nas diretivas, primeiro foi realizado o mapeamento das quatro máximas de Grice para as DCs com expressões que podem caracterizar a comunicação do produtor para os consumidores. A seguir as diretivas são apresentadas, assim como a máxima de Grice na qual cada diretiva se baseia.

“Diga a verdade!” - DC1: Use informações verdadeiras. Não use informações que afetam a qualidade da informação no artefato (com base na Máxima de Qualidade).

“Diga o que é necessário e não mais que o necessário” - DC2: Use o conteúdo necessário no artefato. Não use conteúdo desnecessário no artefato (com base na Máxima de Quantidade).

“Diga de forma lógica” – DC3: Organize as informações no artefato de forma coerente (com base na Máxima de Relação).

“Diga de forma clara” – DC4: Organize as informações no artefato de forma clara (com base na Máxima de Modo).

A Tabela 5.7 apresenta os passos para o uso das DCs, incluindo possíveis combinações destas a partir das diferentes máximas de Grice. Conforme a Tabela 5.7, é possível utilizar as principais DCs e combiná-las.

Essa proposta ajuda os produtores e preocuparem com as DCs anteriores. Por exemplo (para a combinação de D2.1), ao usar as DC2, sobre uma informação necessária que está “faltando” no diagrama, o modelador também deve se preocupar em não incluir uma informação que não seja verdadeira, conforme a DC1.

P#	Forma de Uso
Passo 1	Apenas a DC1 (Máxima de Qualidade) .
Passo 2	DC2 (Máxima de Quantidade) , com as seguintes combinações: D2.1 – Em relação às informações necessárias e não mais que necessárias no artefato, não inclua informações que afetam a qualidade do artefato. (combinação de Quantidade e Qualidade) .
Passo 3	DC3 (Máxima de Relação) , com as seguintes combinações: D3.1 - Caso seja relevante o uso de informações incompletas ou extras, justifique

	(combinação de Relação e Quantidade). D3.2 - Caso seja relevante o uso de informações que não são verdadeiras, justifique (combinação de Relação e Qualidade).
Passo 4	DC4 (Máxima de Modo) , com as seguintes combinações: D4.1 - Mantenha a concisão, sem o sacrifício da coerência (combinação de Modo e Relação). D4.2 - Mantenha a concisão, sem o sacrifício do que é necessário (combinação de Modo e Quantidade). D4.3 - Mantenha a concisão, sem o sacrifício da qualidade (combinação de Modo e Qualidade).

Tabela 5.7: Passos para Utilizar a Segunda Versão das Diretivas de Comunicabilidade.

As DCs estão apresentadas, de maneira geral e foram elaboradas instâncias das DCs para alguns artefatos, no Apêndice C. O processo de instanciação inicia pela reflexão sobre o conteúdo informacional do artefato. No diagrama de classes da UML, por exemplo, o conteúdo informacional está representado através das classes criadas, com seus atributos específicos, incluindo os respectivos relacionamentos. Após isso, as quatro principais diretivas são adaptadas para o artefato. No caso da DC1 para diagramas de classes, por exemplo, as informações que contém informações verdadeiras estão relacionadas à descrição das classes que fazem parte do domínio do problema e de seus relacionamentos. Caso isso não seja verdadeiro, tal modelagem não estará representando o domínio do problema e os consumidores poderão ter falhas na comunicação sobre estas informações. No caso da DC2, deve-se analisar os aspectos relacionados à quantidade de informações do artefato. No caso do diagrama de classes, a quantidade de classes representadas na modelagem devem ser analisada, removendo ou atribuindo novas classes para a compreensão do domínio do problema. Para a DC3, deve-se analisar a forma como o artefato permite organizar as informações no artefato, buscando manter uma sequência lógica para tal compreensão. No diagrama de classes, isso pode ser obtido através da organização das classes em uma sequência que represente um melhor entendimento do domínio do problema. Por fim, sobre a DC4, deve-se pensar os termos que podem ser usados de forma clara no conteúdo informacional do artefato. No caso do diagrama de classes, os nomes das classes devem ser definidos de forma que estas não sejam ambíguas ou não permitam ter a compreensão sobre a diferença de uma classe para outra. Como exemplo das DCs específicas para diagramas de classe, abaixo são apresentadas as DCs específicas para tal diagrama.

“Diga a verdade!” - DC1: Use informações verdadeiras. Não use informações que afetam a qualidade da informação no artefato.

- Não insira classes no diagrama que estejam fora do domínio do problema.
- Remova as informações desatualizadas.

“Diga o que é necessário e não mais que o necessário” - DC2: Use o conteúdo necessário no artefato. Não use conteúdo desnecessário no artefato.

- Insira todas as classes do domínio do problema no diagrama.
- Insira todos os relacionamentos necessários entre as classes.
- Não use conteúdo desnecessário no diagrama.

“Diga de forma lógica” – DC3: Organize as informações no artefato de forma coerente.

- Organize as classes de forma que estas sejam visualizadas da melhor maneira.
- Organize os relacionamentos de forma que estes sejam visualizadas da melhor maneira entre as classes.

“Diga de forma clara” – DC4: Organize as informações no artefato de forma clara.

- Evite termos nos nomes das classes ou atributos que causem ambiguidade.
- Evite interpretação implícita.

Além da melhoria das DCs, foi realizada uma proposta para análise das condições de comunicação via artefato. A partir da adaptação dos elementos do espaço de comunicação (Jakobson, 1960) e do template de *metacomunicação* da Engenharia Semiótica (De Souza *et al.*, 2016), percebeu-se que podem ser analisados três fatores que impactam a comunicação a partir de artefatos, sendo:

Artefato como Meio de Comunicação - com base nos elementos de comunicação *Contexto*, no qual deve-se compreender se o artefato pode representar de maneira clara o domínio do problema, e *Canal* que refere-se ao que se deve compreender se o artefato, na forma em que este é disponibilizada, pode representar as informações necessárias para apoiar a equipe em suas atividades, como codificação e teste;

Conteúdo Comunicativo - com base nos elementos *Mensagem* e *Código* do template de *metacomunicação* da Engenharia Semiótica, os quais são necessários para estabelecer a comunicação recíproca entre produtores e consumidores;

Consumidores do Artefato - com base no elemento *Receptor*. Sobre este último fator, ele pode ser dividido em relação à capacidade de conhecimento do consumidor sobre o artefato, isto é, se ele é capaz de compreender o conteúdo comunicativo do produtor, e se o consumidor considera tal artefato útil para apoiá-lo em suas atividades.

Foram elaborados itens para análise de cada um destes fatores. Os itens relacionados aos fatores ‘Artefato como Meio de Comunicação’ e ‘Conteúdo Comunicativo’ devem ser respondidos pelos produtores dos artefatos. Os itens relacionados ao fator ‘Consumidores do Artefato’ deve ser respondido pelos próprios consumidores. Abaixo são apresentados os itens para análise de cada um destes fatores.

- **Artefato como Meio de Comunicação**

- **M1.** Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.
- **M2.** Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.
- **M3.** Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.
- **M4.** Este artefato me auxilia no que eu gostaria de “falar” para a equipe sobre o domínio do problema.
- **M5.** Este artefato permite que o meu projeto para o software seja representado de forma clara para equipe.
- **M6.** Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).
- **M7.** Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.

- **Conteúdo Comunicativo**

- **Cont1.** Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.
- **Cont2.** Não foram introduzidas informações desorganizadas no artefato.
- **Cont3.** Não foi introduzido o excesso de informações no artefato.

- **Cont4.** A quantidade de informações no artefato é suficiente para o entendimento da equipe.
- **Cont5.** Acredito que o conteúdo do artefato está claro e é de fácil compreensão.

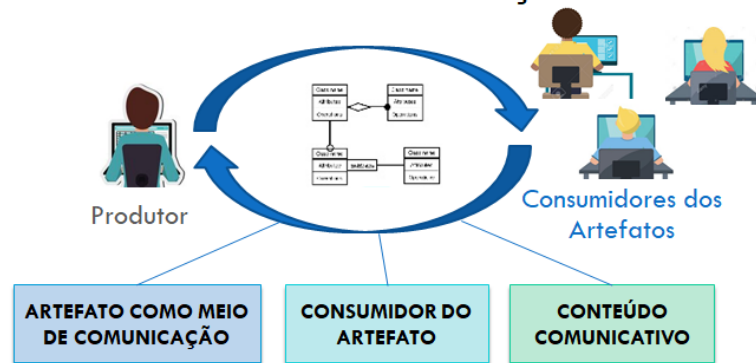
- **Consumidores do Artefato**
 - **PC1.** Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.
 - **PC2.** Eu acho fácil compreender este artefato.
 - **PC3.** Eu irei adotar esse artefato para compreender as atividades de desenvolvimento
 - **PC4.** Eu compreenderei melhor o software com este artefato.
 - **PC5.** Eu acho fácil usar este artefato.

Todos os itens devem ser avaliados com um formato de resposta baseado em uma escala Likert de sete pontos, com opções de resposta desde Concordo Totalmente a Discordo Totalmente, incluindo a escala intermediária. A escala de sete pontos foi escolhida porque esta permite uma melhor discriminação e permite o ganho de consistência interna e confiabilidade (Cicchetti *et al.*, 1985), critérios importantes para análise de variáveis que mensuram fatores. As respostas positivas dos respondentes indicam que a comunicação via artefato pode ser efetiva e as respostas negativas indicam possíveis riscos de falhas de comunicação. Os itens foram organizados em questionários e estão disponíveis no Apêndice D.

A Figura 5.6 apresenta uma caracterização das etapas realizadas para a reflexão do(s) produtor(es) sobre os fatores que impactam a comunicação via artefato e DCs aplicadas para a melhoria da comunicabilidade do artefato.

ETAPA 1

FATORES QUE INFLUENCIAM A COMUNICAÇÃO VIA ARTEFATO



ETAPA 2

DIRETIVAS DE COMUNICABILIDADE NA PRODUÇÃO DOS ARTEFATOS

DC1 - Diga a verdade!

DC2 - Diga o que é necessário e não mais que o necessário

DC3 - Diga de forma clara

DC4 - Diga de forma lógica

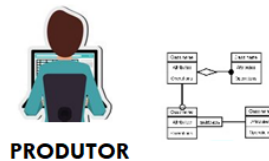


Figura 5.6: Exemplo de uso das Propostas Apresentadas nesta Tese.

A avaliação destas melhorias foram avaliadas nos demais estudos realizados nesta pesquisa, como o estudo experimental realizado para avaliar se as melhorias realizadas nas DCs continuaram promovendo a redução dos riscos de falhas de comunicação, apresentado na próxima subseção. Os demais estudos são apresentados no Capítulo 6.

5.5.Segundo Estudo Experimental: Análise da Percepção de Profissionais sobre o Uso das DCs

Foi realizado um segundo estudo, *in vitro*, com o objetivo de compreender as melhorias realizadas nas DCs para a redução dos riscos de falhas de comunicação, além de avaliar a percepção de profissionais da indústria sobre as DCs. Em relação à avaliação dos fatores que podem impactar na comunicação via artefato, foi decidido pelas pesquisadoras que tal avaliação seria realizada nos estudos *in vivo* a fim de obter dados que possam expressar a percepção de produtores e consumidores em projetos de desenvolvimento, sendo este apresentado no Capítulo 6.

5.5.1 Planejamento do Estudo

O estudo foi planejado para ser executado em um único dia. Foram selecionados 15 profissionais da indústria para a produção de diagramas de casos de uso da UML (diagrama e especificação de casos de uso) com o apoio das DCs. Tais profissionais estavam cursando a disciplina Fundamentos de Engenharia de Software do curso de pós-graduação em Engenharia de Software com ênfase em Desenvolvimento de Software do Centro Universitário do Norte (UNINORTE). Tal disciplina foi ministrada em dois dias, no período da manhã e tarde. A pesquisadora principal planejou a realização do estudo no segundo dia em que a disciplina foi ministrada. No período da manhã, antes da execução do estudo, os participantes haviam obtido um treinamento de aproximadamente duas horas para estes exercitarem a modelagem de casos de uso. Ressalta-se que todos estes profissionais tinham conhecimento na modelagem de casos de uso.

O estudo foi planejado para ser executado em um período de aproximadamente três horas no período da tarde. Foi reservado um laboratório para a execução do estudo, o qual tinha notebooks para serem utilizados pelos participantes. Com o objetivo de observar a discussão dos participantes em relação ao desenvolvimento dos artefatos, foi planejado o desenvolvimento de quatro grupos de modelagem. A pesquisadora principal enviou um e-mail para os participantes responderem o nível de experiência na indústria. No entanto, a maioria dos participantes não respondeu o e-mail, impossibilitando o balanceamento dos grupos. Com isso, os grupos foram elaborados de maneira aleatória.

Foram elaborados quatro cenários de modelagem simples. A escolha dos diferentes propósitos de sistema foi realizada para compreender o emprego das DCs em diferentes sistemas e para que os participantes pudessem concluir as atividades do estudo no tempo planejado. A descrição do propósito dos cenários de modelagem está disponível no Apêndice B. Os objetivos dos sistemas descritos nos cenários de modelagem estão descritos abaixo:

- Gerenciamento de professores e alunos em escolas de reforço para disciplinas básicas;
- Gerenciamento de eventos;
- Apoio aos lojistas no ramo de vendas em geral;
- Disponibilização de informação dos itinerários fluviais de Manaus.

5.5.2 Execução do Estudo

Antes da execução do estudo, os participantes tiveram um treinamento com aproximadamente 20 minutos sobre a proposta das DCs para serem aplicadas na modelagem de casos de uso. Foram apresentadas as DCs gerais e DCs específicas para casos de uso. Em seguida, a pesquisadora principal pediu aos participantes para se posicionarem de acordo com os grupos formados para realizarem a atividade do estudo. Os participantes estavam no mesmo laboratório, porém os grupos estavam distanciados. Após isso, os cenários de modelagem e as DCs foram entregues para os grupos.

A pesquisadora principal pediu para os participantes elaborarem o diagrama de casos de uso de maneira conjunta, discutindo aspectos relevantes do sistema. Também foi pedido aos participantes a especificação de apenas um caso de uso por cada participante. Os participantes utilizaram a ferramenta Astah para a modelagem de casos de uso. A Tabela 5.8 apresenta os quatro grupos formados pelos participantes e o objetivo de cada sistema nos cenários de modelagem.

Grupos	Participantes	Objetivos dos Sistemas a partir dos Cenários
Grupo 1	P4, P5, P6 e P7	Gerenciamento de professores e alunos em escolas de reforço para disciplinas básicas
Grupo 2	P8, P9, P10 e P11	Gerenciamento de eventos
Grupo 3	P12, P13, P14 e P15	Apoio aos lojistas no ramo de vendas em geral
Grupo 4	P1, P2 e P3	Disponibilização de informações dos itinerários fluviais de Manaus

Tabela 5.8: Grupos de Modelagem de Casos de Uso com as DCs.

Em relação ao uso das DCs, a pesquisadora principal esclareceu que os participantes poderiam empregá-las de acordo como estes considerassem mais adequado, como o uso destas durante a modelagem ou após estes realizarem uma proposta de modelagem. A pesquisadora observou que os participantes de todos os grupos realizaram a modelagem de casos de uso dos sistemas e em seguida aplicaram as DCs para apoiá-los nas decisões de projeto. Após o estudo, todos os participantes responderam um questionário pós-estudo para fornecerem as suas percepções sobre as DCs, incluindo a experiência de cada um dos participantes no desenvolvimento de software. O questionário pós-estudo está

disponível no Apêndice B. Sobre a duração do estudo, este foi finalizado antes do tempo planejado.

5.5.3 Resultados do Estudo

O nível de experiência dos participantes com desenvolvimento de software foi coletado a partir do questionário pós-estudo, sendo este apresentado na Tabela 5.9. Esses dados são apresentados para o esclarecimento dos participantes em cada grupo.

Profissional Júnior (1 – 3 anos de experiência)	Profissional Pleno 4 – 8 anos de experiência	Profissional Sênior > 9 anos de experiência
P1, P2, P3, P4, P6, P7, P10, P12, P13	P8, P9, P15	P5, P11, P14

Tabela 5.9: Nível de Experiência dos Participantes do Segundo Estudo Experimental.

Os artefatos produzidos pelos grupos foram analisados em relação aos riscos de falhas de comunicação do ponto de vista da pesquisadora principal, os quais foram discutidos com a orientadora dessa trabalho. Os riscos de falhas de comunicação identificados nos casos de uso são apresentados na Tabela 5.10, incluindo a quantidade total dos diferentes riscos observados ao lado de cada descrição. A avaliação dos artefatos produzidos pelos grupos resultou em poucos riscos de falhas de comunicação. Porém, tais riscos podem causar possíveis falhas de comunicação Completas.

Sobre a reflexão dos participantes com base nas DCs para a produção dos casos de uso, a pesquisadora principal observou que a maioria dos participantes empregou a DC2 para avaliar quantidade de informações necessárias que devem ser representadas e a DC3 para a organização de informações de maneira lógica nos casos de uso.

Grupos	Classificação	Descrição dos Riscos de Falhas de Comunicação
Grupo I	Completa	- Falta de relacionamento no diagrama de casos de uso (1) - Diferentes padrões na organização das informações de especificação de casos de uso (3) - Falta de informação em regras de negócio (5)
Grupo II	Completa	- Especificação de caso de uso inconsistente com o diagrama (1) - Falta de relacionamento no diagrama de casos de uso (1) - Falta de informação em regras de negócio (4)
Grupo III	Completa	- Falta de informação em regras de negócio (2) - Falta de passos na especificação do caso de uso (2)
Grupo IV	Completa	- Falta de passos na especificação do caso de uso (2) - Falta de informação em regras de negócio (5)

Tabela 5.10: Riscos de Falhas de Comunicação nos Artefatos Desenvolvidos.

Em relação ao questionário pós estudo, os participantes responderam a seguinte questão: “*Q1. Qual a sua percepção sobre as diretivas de comunicabilidade?*”. As respostas foram organizadas em diferentes categorias, as quais representam as percepções dos participantes. A Tabela 5.11 apresenta estas categorias e o relato dos participantes.

Categoria	Citação
DCs promovem a reflexão do design do sistema	P12 - <i>"As diretivas ajudam a organizar o pensamento na hora de projetar o sistema"</i>
	P5 - <i>"Ajudam a obter informações relevantes para o projeto"</i>
	P10 - <i>"As diretivas ajudam na reflexão do que deve ser desenvolvido, evitando incoerências"</i>
	P7 - <i>"As diretivas fornecem um apoio grande para a produção do software"</i>
	P14 - <i>"As diretivas ajudam no entendimento do sistema, ajudando a identificar possíveis erros"</i>
	P9 - <i>"Facilita na identificação de problemas na modelagem"</i>
	P15 - <i>"Ajuda a melhorar consideravelmente o entendimento geral de um sistema"</i>
DCs promovem a organização das informações	P4 - <i>"Ajudam a organizar melhorar as informações necessárias para criar um sistema"</i>
	P10 - <i>"Auxiliam na organização das ideias juntamente com o time de desenvolvimento"</i>
DCs podem promover a comunicação efetiva via artefato	P11 - <i>As DCs servem como um roteiro para organizar as ideias na comunicação de forma clara e lógica"</i>
	P6 - <i>"Ajudam a pensar em como se comunicar com os colegas"</i>
	P13 - <i>"Ajuda na comunicação de forma correta no desenvolvimento de software"</i>
DCs promovem uma melhor comunicabilidade	P8 - <i>"As diretivas ajudam a reduzir a possibilidade de uma mesma ideia ser interpretada de várias maneiras, pois as ideias devem ser passadas de uma maneira que todos compreendam"</i>
Dificuldades com o uso das DCs	P2 - <i>"Não é fácil compreender as diretivas, exigiu mais do meu esforço mental"</i> P5 - <i>"Não é fácil de aplicar, acredito que depende da experiência do utilizador"</i> P6 - <i>"As diretivas demandam tempo para o entendimento"</i>

Tabela 5.11: Percepção dos Participantes sobre as DCs na Produção de Casos de Uso.

A maioria das respostas na Tabela 5.11 mostraram que o propósito das DCs foi percebido pelos participantes, sendo possível observar através das respostas dos participantes que as ‘DCs promovem a reflexão no design do sistema’ e as ‘DCs podem promover a comunicação efetiva via artefato’. Isso também foi observado para o conteúdo dos artefatos a partir das respostas dos participantes, indicando que as ‘DCs promovem a organização das informações’ e as ‘DCs promovem uma melhor comunicabilidade’. Alguns participantes também relataram ‘Dificuldades com o uso das DCs’, o qual pode estar relacionado à reflexão destes em estar aplicando corretamente ou não o conceito

principal de cada diretiva ao que cada produtor deseja comunicar. Isso faz parte do processo de reflexão dos produtores que as DCs promovem, uma vez que estes devem se questionar a respeito de sua comunicação através dos artefatos. Outros dados foram coletados a partir do questionário pós-estudo, mas esses são discutidos no Capítulo 6.

5.5.4 Limitações do Estudo

Em relação à condução do estudo, existe a limitação do tempo para a condução da atividade. Para isso foi planejada a modelagem de sistemas com domínio do problema simples para que os participantes pudessem realizar o estudo em um período máximo de três horas. Com isso, os participantes puderam realizar a atividade e fornecerem suas percepções sobre as DCs para projetos de software. As percepções dos participantes mostram que as diretivas contribuíram para uma melhor comunicabilidade dos artefatos. Sobre a análise dos riscos de falhas de comunicação nos casos de uso, esta foi revisada por mais uma pesquisadora para diminuir a ameaça relacionada à expectativa da pesquisadora principal. Em relação à generalização dos resultados obtidos com as DCs, estes foram aplicados apenas para a produção de casos de uso. Os resultados obtidos neste estudo podem não ser obtidos na produção de outros artefatos para apoiar a comunicação dos envolvidos na produção dos sistemas.

5.5.5 Discussão dos Resultados

Este segundo estudo foi realizado com o objetivo de compreender as melhorias realizadas nas DCs e analisar a percepção de profissionais da indústria sobre o uso das diretivas. Os resultados indicaram que as DCs promoveram a reflexão dos produtores sobre a comunicação via artefato, melhorando a comunicabilidade destes. Além disso, a maioria das respostas dos participantes sobre as DCs não indicaram percepções negativas sobre o seu uso na indústria. Com estes resultados, as DCs foram consideradas aptas para serem avaliadas em relação ao seu apoio à comunicação via artefato. Assim, os próximos estudos foram planejados para analisar a compreensão recíproca entre produtores e consumidores de artefatos durante o desenvolvimento de seus projetos.

5.6. Terceiro Estudo Experimental: Análise do Consumo de Artefatos Produzidos com as DCs

Os estudos anteriores foram realizados com o propósito de avaliar se as DCs

promovem a redução de riscos de falhas de comunicação e se estas podem ser empregadas pelos produtores de software. Como os resultados foram positivos em relação ao propósito dos estudos anteriores, estudos para avaliar os efeitos das DCs na comunicação entre produtores e consumidores via artefato foram planejados.

Essa subseção apresenta um estudo realizado para obter evidências sobre o uso das DCs na melhoria da comunicação entre produtores e consumidores de artefatos. Esse estudo foi apresentado em Lopes *et al.* (2019a). Tal estudo foi realizado em duas etapas: (1) produção de artefatos por dois diferentes produtores, em que um produtor utilizou as DCs e o outro produtor não; (2) estes artefatos foram consumidos por outros participantes para o desenvolvimento de mockups.

Na primeira etapa, um diagrama de classes e diagrama de atividades foram produzidos com o apoio das DCs e outro sem. Esses diagramas foram produzidos por diferentes engenheiros de software, os quais possuíam o mesmo nível de experiência em modelagem de software. Estes produtores foram considerados como experientes devido à sua experiência com modelagem de software (com mais de três anos).

Os produtores foram convidados para realizar a modelagem de um sistema que contribua para um melhor controle de financeiro de seus usuários. O sistema deveria apoiar a criação de planejamentos que ajudarão o usuário a se organizar, além de auxiliar na economia de seu dinheiro. De acordo com o cliente e analistas, o aplicativo proposto visa: (i) fornecer o controle de cartões de crédito e de contas bancárias; (ii) ajudar o usuário a planejar suas finanças através do planejamento de gastos por despesas, com sua categoria, além do acompanhamento da evolução das despesas; (iii) informar o usuário quando o gasto, relacionado ao planejamento, atingir mais de 80% do limite estipulado. Com estas informações, um modelador utilizou as DCs e o outro não. Estes diagramas estão disponíveis no Apêndice B.

Na segunda etapa, os participantes projetaram mockups com o apoio dos diagramas desenvolvidos na primeira etapa. Foram selecionados 30 participantes, divididos em dois grupos (Grupo Experimental e Grupo de Controle), para criarem mockups a partir dos diagramas de classes e atividades. Os participantes do Grupo Experimental utilizaram os artefatos desenvolvidos pelo produtor que utilizou as DCs e os participantes do Grupo de Controle utilizaram os artefatos desenvolvidos pelo produtor que não utilizou as DCs. Os

participantes foram alunos de graduação do curso de Ciência da Computação e Engenharia de Software da UFAM, os quais haviam cursado as disciplinas de IES APS. Os grupos foram balanceados pelas notas de provas obtidas nas disciplinas de IES e APS e os participantes possuíam o mesmo nível de experiência com a construção de mockups.

5.6.1 Resultados do Estudo

Os mockups produzidos por cada participante foram analisados com o objetivo de compreender as informações expressas nestes a partir dos diagramas. Esta análise foi realizada com o apoio da ferramenta SigniFYI Message (de Souza *et al.*, 2016). A pesquisadora principal desta tese analisou os conjuntos de mockups. Os resultados desta análise foram discutidos com mais uma pesquisadora (orientadora deste trabalho), os quais estão apresentados no Apêndice E.

Em cada grupo, os participantes desenvolveram um conjunto de 15 mockups, sendo que cada conjunto possuiu de três a cinco telas. Em relação aos mockups desenvolvidos, no que se referem às informações expressas a partir dos diagramas de classes e diagrama de atividades, não foram observadas dificuldades na análise destes. Ressalta-se que os mockups não foram avaliados em outros aspectos de qualidade de uso, como usabilidade e comunicabilidade de interface (Barbosa e Silva, 2010).

As falhas de comunicação entre os produtores e seus consumidores, analisadas a partir dos conjuntos de mockups desenvolvidos, são resumidas na Tabela 5.12. A descrição destas falhas de comunicação a partir dos artefatos também é apresentada. Os defeitos que impactam a qualidade de software devido às falhas de comunicação também são apresentados, conforme diferentes tipos de defeitos (Granda *et al.*, 2015).

A partir da Tabela 5.12 é possível ver as falhas de comunicação que ocorreram entre produtores e consumidores dos diagramas de classes e atividades de forma única, com a quantidade de repetições são apresentadas ao lado da descrição de cada falha. O Grupo Experimental obteve um total 16 falhas de comunicação, sendo 5 falhas de comunicação classificadas como “Completa”. O Grupo de Controle obteve um total de 24 falhas, sendo 15 falhas classificadas como “Completas”.

Grupo	Falhas de Comunicação	Mockups Afetados por Problemas de Comunicabilidade	Defeitos de Software
Experimental	Temporária	Nenhum	-

Grupo	Falhas de Comunicação	Mockups Afetados por Problemas de Comunicabilidade	Defeitos de Software
(consumo de artefatos desenvolvidos com apoio das DCs)	Parcial	Não foram projetadas alternativas para o usuário (5)	Omissão
		Não foi representado os gastos por categoria (6)	Omissão
	Completa	O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos (5)	Fato Incorreto
Controle (consumo de artefatos desenvolvidos sem as DCs)	Temporária	Nenhum	-
	Parcial	Não foram projetadas alternativas para o usuário (7)	Omissão
		Não foi representado os gastos por categoria (2)	Omissão
	Completa	Não é possível que usuário acompanhe a evolução dos gastos (2)	Omissão
		O usuário não é informado quando o planejamento ultrapassa 80% dos gastos e sim o limite total (1)	Inconsistência
		O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos (2)	Fato Incorreto
		A opção para o cadastro de cartões não foi representada (2)	Omissão
		A opção para o cadastro de contas não foi representada (3)	Omissão
A opção para o cadastro de despesas não foi representada (5)	Omissão		

Tabela 5.12: Propagação de Problemas de Comunicabilidade do UC para mockups.

No Grupo Experimental, percebe-se que não ocorreram falhas “Temporárias”. Como falhas “Parciais”, foi observado que cinco participantes não representaram alternativas nos mockups para os usuários. Por exemplo, não é possível os usuários realizarem alterações no cadastro de cartões ou no planejamento. Isso pode ter ocorrido devido ao fato destes participantes não considerarem essas alternativas importantes. Ainda nesta categoria, seis participantes não representaram os gastos por categorias, talvez, pelo fato de os participantes terem considerado as despesas como as próprias categorias. Essas falhas de comunicação podem causar a omissão de funcionalidades. Em relação às falhas “Completas”, por exemplo, foi observado que dois participantes não representaram nos mockups opções para o usuários alterarem o planejamento somente quando este ultrapasse 80% dos gastos. Isso pode ter ocorrido pelo fato de os participantes não observarem o diagrama de classes, vendo apenas as informações no diagrama de

atividades. Esse tipo de falha de comunicação pode causar defeitos que afetem o uso correto das funcionalidades.

No Grupo de Controle, percebeu-se que não ocorreram falhas “Temporárias”. Sobre as falhas “Parciais”, por exemplo, não foram representadas alternativas nos mockups para os usuários alterarem o cadastro de cartões ou alterações no planejamento. Isso pode ter ocorrido devido ao fato dos consumidores dos artefatos não considerarem essas alternativas importantes ao criarem os mockups. Esta falha pode causar a omissão de funcionalidades. Em relação às falhas “Completas”, por exemplo, foi observado que dois participantes não representaram nos mockups o acompanhamento dos gastos. Talvez esses participantes não tenham compreendido essas informações a partir do diagrama de classes, o que pode causar a omissão desta funcionalidade. Não foi representado em um conjunto de mockups desenvolvido por um participante o fato dos usuários serem informados do limite quando este atinge 80% de seus gastos, os usuários nos mockups serão informados apenas quando ultrapassa o limite total. Tal participante como consumidor dos artefatos não deve ter compreendido tal informação a partir do diagrama de atividades, o que pode causar inconsistência na funcionalidade.

5.6.2 Limitações do Estudo

Sobre as limitações deste estudo, existe o fato de os participantes serem estudantes de graduação e o estudo ser conduzido em um ambiente acadêmico. No entanto, estudantes podem ter habilidades semelhantes aos profissionais menos experientes (Fernandez *et al.*, 2012). Outra limitação é o uso de apenas dois diagramas da UML, pois estes não representam todos os artefatos de software que podem ser usados como meio de comunicação. Sobre a análise das falhas de comunicação nos casos de uso, esta foi revisada por mais uma pesquisadora para diminuir a ameaça relacionada à expectativa da pesquisadora principal. Além disso, tais diagramas foram modelados por apenas por dois engenheiros de software. Portanto, não é possível generalizar os esses resultados.

5.6.3 Discussão dos Resultados

Este estudo foi realizado com o objetivo de compreender os benefícios das DCs em relação à melhoria da comunicação a partir de dois diferentes artefatos de software. Os resultados indicaram que os participantes do Grupo Experimental, que projetaram os

mockups a partir de sua compreensão dos artefatos desenvolvidos com o apoio das DCs, obtiveram um menor número de falhas de comunicação em comparação com os participantes do Grupo Controle, os quais projetaram os mockups a partir de artefatos desenvolvidos sem as DCs. Esses resultados mostram indícios de que o uso das DCs proporciona benefícios na melhoria da comunicação entre produtores e consumidores.

5.7.Considerações sobre o Capítulo

Este capítulo apresentou as DCs, propostas para melhorar a comunicabilidade de diferentes artefatos de software. Foi observado em dois estudos experimentais, realizados para avaliar as DCs, que estas promoveram a redução de riscos de falhas de comunicação. Com isso, as DCs foram consideradas aptas para serem avaliadas em relação ao apoio da comunicação recíproca entre produtores e consumidores de artefatos de software.

Este capítulo também apresentou um estudo realizado com o objetivo de iniciar a investigação sobre a comunicação recíproca entre produtores e consumidores de artefatos. Os resultados indicaram que os artefatos de software desenvolvidos pelos consumidores a partir dos diagramas produzidos com o apoio das DCs resultaram em um menor número de falhas de comunicação, em comparação com os consumidores dos diagramas produzidos sem o uso das DCs. Portanto, o uso das DCs mostrou indícios de melhoria na comunicação entre produtores e consumidores de artefatos. Assim, pode-se afirmar que as DCs promoveram uma melhor aproximação para a resolução do problema, no que se refere à comunicação eficaz via artefato. Com isso, considerou-se que as DCs podem ser utilizadas em um ambiente real de desenvolvimento de software.

CAPÍTULO 6 – ESTUDOS NA INDÚSTRIA COM AS DIRETIVAS DE COMUNICABILIDADE

Este capítulo apresenta cinco estudos experimentais que exploraram as Diretivas de Comunicabilidade durante o desenvolvimento de software.

6.1 Introdução

Os estudos experimentais para avaliação das Diretivas de Comunicabilidade (DCs), apresentados no Capítulo 5, mostraram que estas podem ser utilizadas em um ambiente real de desenvolvimento de software. Com isso, foi planejado um estudo *in vivo* com o objetivo de investigar o uso das DCs na produção de artefatos e os benefícios destas para quando ocorre a comunicação via artefato durante o desenvolvimento de software. Diante do exposto até aqui, ainda na etapa de ‘demonstração’ da metodologia seguida nesta pesquisa, como mostra a Figura 6.1, foram realizados dois estudos na indústria.

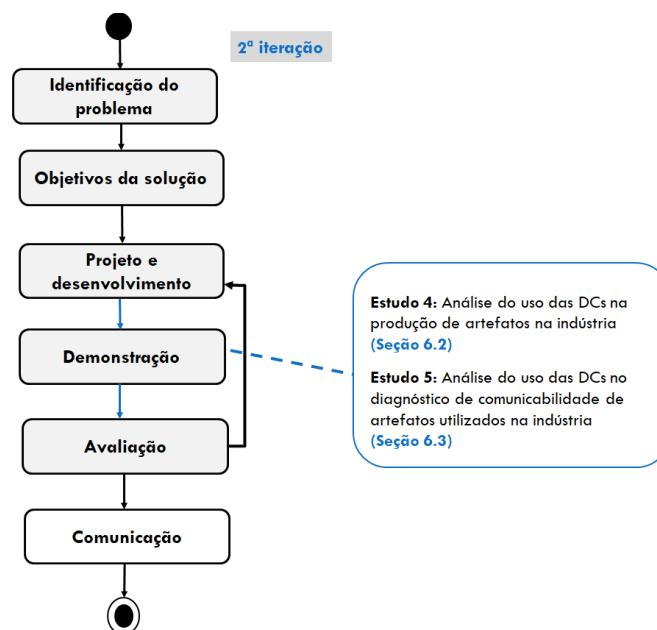


Figura 6.1: Estudos na Indústria com as Diretivas de Comunicabilidade.

Este capítulo apresenta dois estudos realizados na indústria de software com o objetivo de avaliar o uso das DCs em artefatos utilizados por diferentes equipes de desenvolvimento. O primeiro estudo teve por objetivo avaliar a aplicação das DCs durante a produção de artefatos antes do consumo destes pelos consumidores. O segundo estudo

foi realizado com o objetivo de avaliar se as DCs poderiam ser aplicadas para o diagnóstico de problemas de comunicabilidade. Os fatores relacionados à comunicação via artefato também foram avaliados nestes estudos.

Em ambos os estudos, foram realizadas diferentes entrevistas para o entendimento da comunicação das equipes com apoio dos artefatos e uso das DCs. Por esta razão, os dados qualitativos coletados foram analisados com procedimentos de *Grounded Theory* (Bandeira-de-Mello e Cunha, 2003), com auxílio da ferramenta Atlas.ti⁷. O *Grounded Theory* (ou Teoria Fundamentada nos Dados) (GT) é um método qualitativo que utiliza um conjunto de procedimentos sistemáticos de coleta e análise dos dados para gerar, elaborar e validar teorias substantivas sobre fenômenos essencialmente sociais, ou processos sociais abrangentes.

O método GT surgiu a partir da publicação de Glaser e Strauss (1967). No entanto, seus autores divergiram sobre alguns pontos e o método dividiu-se em duas vertentes. Uma das vertentes é defendida por Glaser (1992) e dá ênfase a característica emergente do método e aos processos indutivos. A outra vertente foi desenvolvida por Strauss (1987) e consolidada em (Strauss e Corbin, 1998), com o objetivo de sistematizar o método de coleta e análise de dados. A linha proposta por Strauss, GT é baseado na ideia de codificação (*coding*), que é o processo de analisar os dados. Durante esta codificação são identificados conceitos (ou códigos) e categorias. Um conceito (ou código) dá nome a um fenômeno de interesse para o pesquisador, abstraído de um evento, objeto, ação, ou interação que tenha um significado para o pesquisador (Strauss e Corbin, 1998). Categorias são agrupamentos de conceitos unidos em um grau de abstração mais alto. O processo de codificação pode ser dividido em três fases:

Codificação aberta - Envolve a quebra, a análise, a comparação, a conceituação e a categorização dos dados. Segundo Bandeira-de-Mello e Cunha (2006), nas fases iniciais da codificação aberta, o pesquisador explora os dados examinando minuciosamente aquilo que lhe parece relevante devido à leitura intensiva dos textos. Na fase de codificação aberta os incidentes ou eventos são agrupados em códigos através da comparação incidente–incidente. Os códigos gerados podem ser classificados como:

⁷ Atlas.ti – The Knowledge Workbench, Scientific Software Development – <http://www.atlasti.com>

códigos de primeira ordem, diretamente associados às citações (chamados códigos *in vivo*), códigos abstratos ou teóricos, associados a outros códigos, sem necessariamente estarem ligados a alguma citação. Também na codificação aberta, é realizada a criação de categorias que agregam os códigos para reduzir o número de unidades com que o pesquisador irá trabalhar.

Codificação axial - Após a identificação de categorias conceituais pela codificação aberta, a codificação axial examina as relações entre as categorias que formam as proposições da teoria substantiva (Bandeira-de-Mello e Cunha, 2006). Explicitam-se causas e efeitos, condições intervenientes e estratégias de ação, em proposições que devem ser testadas novamente nos dados. As relações entre os códigos (conectores segundo Glaser (1992)) podem ser definidas pelo próprio pesquisador. Na linha proposta por Strauss e Corbin (1998), essas relações formam o que os autores denominam de paradigma: condições causais, intervenientes, consequências e estratégias de ações/interações. A Tabela 6.1, adaptada de (Bandeira-de-Mello 2006), apresenta uma sugestão de conectores.

Codificação seletiva - Refina todo o processo identificando a categoria central da teoria, com a qual todas as outras estão relacionadas. A categoria central (*core category*) deve ser capaz de integrar todas as outras categorias e expressar a essência do processo social que ocorre entre os envolvidos. Esta categoria central pode ser uma categoria existente, ou uma nova categoria pode ser criada.

Rótulo	Descrição
<i>Is part of</i> (É parte de)	O código-origem é uma parte que compõe juntamente com outras partes o código-destino.
<i>is associated with</i> (É associado com)	O código-origem e o código-destino têm conceitos específicos relacionados.
<i>Is property of</i> (É propriedade de)	O código-origem é propriedade da categoria (código-destino)
<i>Is a</i> (É um)	O código-origem é um tipo, ou forma, do código-destino. É definido por um padrão de variação dimensional ao longo das propriedades da categoria (código-destino).

Tabela 6.1: Conectores para a Codificação Axial (adaptado de Bandeira-de-Mello (2006)).

Os procedimentos de GT foram aplicados para uma melhor compreensão dos dados qualitativos dos estudos apresentados neste capítulo, com base na linha proposta por Strauss (Strauss, 1987; Strauss e Corbin, 1998). Ao analisar as respostas dos participantes, na fase da codificação aberta, foram criados códigos (conceitos relevantes

da percepção dos participantes sobre as DCs) através de abstrações dos relatos fornecidos pelos diferentes membros das equipes. Em seguida, na fase de codificação axial, os códigos foram analisados e relacionamentos entre eles foram estabelecidos.

Como não se pretendia criar uma teoria a respeito destes dados, não foi realizada a codificação seletiva (Bandeira-de-Mello, 2006). As etapas de codificação aberta e axial foram suficientes para entender a percepção dos membros das diferentes equipes sobre a comunicação via artefato, incluindo o emprego das DCs nos dois estudos.

6.2 Estudo 1 – Uso das DCs durante o Desenvolvimento de Software

O objetivo deste estudo foi investigar o emprego das DCs na produção de artefatos que apoiam a comunicação de equipes durante o desenvolvimento de software. A pesquisadora principal entrou em contato com algumas empresas de software para a condução deste estudo. Após isso, duas empresas foram selecionadas devido: ao uso de artefatos como apoio à comunicação de equipes, ao seu interesse pela melhoria deste tipo de comunicação e a sua disponibilidade para a execução do estudo. Este estudo foi realizado em duas equipes de desenvolvimento de software de diferentes empresas.

Uma das equipes de desenvolvimento (Equipe 1) é da empresa privada C1, um centro tecnológico privado que desenvolve projetos nas áreas de software e hardware. A outra equipe (Equipe 2) é da empresa C2, uma empresa privada que atua no desenvolvimento de novos produtos, serviços e soluções tecnológicas inovadoras. Ambas equipes adotam a metodologia ágil Scrum em seus projetos e estas são apresentadas na Tabela 6.2.

Papel da Equipe 1	Experiência	Papel da Equipe 2	Experiência
analista de sistemas (<i>product owner - PO</i>)	5 anos	analista de sistemas (<i>product owner - PO</i>)	12 anos
designer	2 anos	designer	3 anos
desenvolvedor 1	10 anos	desenvolvedor 1	2 anos
desenvolvedor 2	13 anos	desenvolvedor 2	4 anos
desenvolvedor 3	11 anos	desenvolvedor 3	2 anos

Tabela 6.2: Equipe 1 e Equipe 2 do Estudo 1.

Na Equipe 1, na *Sprint* que foi investigada, os requisitos já haviam sido definidos e o produtor produziria apenas os protótipos para serem codificados. Além disso, este também produziria uma documentação para apoiar o entendimento da equipe e do cliente.

Portanto, o produtor, o designer da equipe, poderia aplicar as DCs na ‘Externalização’ dos artefatos, os quais são compartilhados na equipe para o entendimento do sistema (Chaudron *et al.*, 2018). Em seguida, os artefatos seriam utilizados pelos consumidores para apoiar a implementação do sistema.

Na Equipe 2, os requisitos ainda seriam definidos. A identificação dos requisitos nesta equipe era apoiada pelo uso de protótipos no início da *Sprint*, juntamente com o cliente. Assim, os produtores, analista de sistemas e designer, poderiam aplicar as DCs na ‘Ideação e Externalização’ dos artefatos, que consiste em criar artefatos para explorar e combinar ideias de desenvolvimento para o sistema e em seguida disponibilizá-los na equipe para o entendimento do sistema (Chaudron *et al.*, 2018). Em seguida, os artefatos seriam utilizados para apoiar a implementação do sistema.

6.2.1 Planejamento e Execução do Estudo

O estudo foi planejado para ser realizado em três etapas: (1) compreensão de problemas que afetam a comunicação a partir de artefatos, análise dos fatores que impactam este tipo de comunicação e análise do relato dos participantes sobre o uso dos artefatos, (2) emprego das DCs durante a elaboração dos artefatos e (3) análise da comunicação via artefato das equipes – com isso, foi possível analisar se houve redução dos problemas identificados na etapa 1. Essas etapas estão representadas na Figura 6.2, sendo estas conduzidas em cada empresa que as equipes atuam.

Na etapa 1, foi realizada uma reunião com cada equipe e a pesquisadora principal deste trabalho. Nesta reunião, cada equipe apresentou uma visão geral de suas atividades e ressaltaram os principais artefatos utilizados como meio de comunicação. Em seguida, a pesquisadora apresentou um link com os questionários que as equipes deveriam responder como primeira etapa da pesquisa. Foram enviados questionários específicos para produtores e consumidores de artefatos nas equipes para coletar a percepção das equipes sobre problemas que afetam a comunicação via artefato e analisar os fatores que impactam esta comunicação, como mostra o Apêndice B.



Figura 6.2: Etapas para a Condução do Estudo 1.

No início da etapa 2, a pesquisadora conduziu um treinamento com os produtores de artefatos sobre o uso das DCs e seus benefícios antes dos consumidores utilizarem estes em suas atividades. Um produtor da Equipe 1 e outro da Equipe 2 receberam treinamento com as DCs gerais e específicas. Houve a inclusão de mais um produtor na Equipe 2 após este treinamento; no entanto, este novo produtor apenas consultou as DCs gerais para a produção de artefatos, conforme sugerido pelo produtor que recebeu o treinamento das DCs.

Na etapa 2, os produtores aplicaram as DCs nos artefatos no início da *Sprint* de desenvolvimento dos projetos. A Equipe 1 produziu apenas protótipos e a Equipe 2 produziu requisitos e protótipos. Após os produtores disponibilizarem os artefatos para as equipes realizarem suas atividades, foi realizada uma reunião com a pesquisadora e estes produtores para eles relatarem sua percepção sobre o uso das DCs. Os produtores também preencheram um questionário pós-estudo.

Na etapa 3, após a finalização da *Sprint* de desenvolvimento do projeto, foi realizada uma entrevista para coletar informações a respeito da mitigação, ou não, dos problemas relacionados à comunicação via artefato. Os dados obtidos na entrevista foram triangulados com os problemas que afetavam tal comunicação, identificados na etapa 1.

As Subseções 6.2.2 e 6.2.3 apresentam resultados obtidos com cada equipe investigada em relação ao resultado da análise da comunicação e dos benefícios das DCs para os artefatos desenvolvidos nestas. A Subseção 6.2.4 apresenta o resultado qualitativo sobre as percepções dos participantes com as DCs.

6.2.2 Resultados da Equipe 1

Esta subseção apresenta os resultados obtidos com a investigação realizada na Equipe 1 da empresa privada C1, que inclui a análise qualitativa e os resultados da análise dos fatores ‘Artefato como meio de Comunicação’, ‘Conteúdo Comunicativo’ e ‘Consumidores do Artefato’.

Para a análise qualitativa, os procedimentos de GT foram aplicados para melhor compreender os dados qualitativos. Ao analisar as respostas dos participantes, na fase de codificação aberta, foram criados códigos através de abstrações das citações dos participantes nas diferentes etapas realizadas no estudo. A Figura 6.2 mostra parte dos códigos associados às citações dos participantes. Tal análise foi realizada com apoio da ferramenta ATLAS.ti⁸.

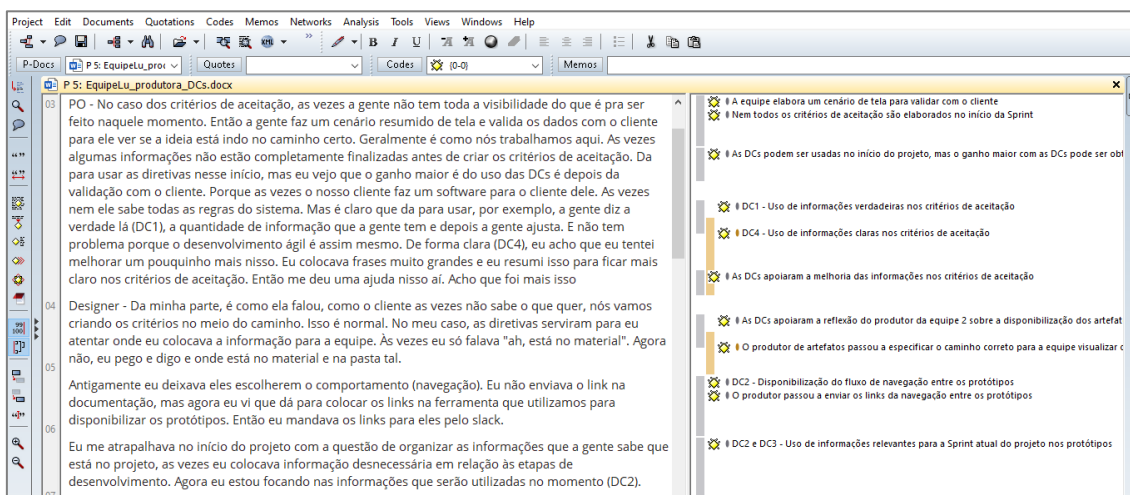


Figura 6.3: Associação dos Códigos e Dados qualitativos coletados durante o Estudo.

Após a codificação aberta, foi realizada a codificação axial. Na codificação axial, os códigos foram analisados e relacionamentos entre eles foram estabelecidos, formando categorias a partir destes códigos. Como o objetivo dessa análise não é criar uma teoria a respeito das DCs, não foi realizada a codificação seletiva. A codificação aberta e

⁸ <https://atlasti.com/>

codificação axial foram suficientes para entender a percepção dos participantes sobre o apoio das DCs na comunicação via artefato. Estes dois procedimentos foram revisados pela orientadora desta tese.

6.2.2.1 Resultado da Comunicação via Artefato da Equipe 1

Com os resultados do emprego de GT na análise dos dados qualitativos, foram identificadas categorias a partir dos códigos (com o uso do termo “CAT”) e subcategorias (subcategoria identificada com “SUB”). Em relação à comunicação da Equipe 1, como mostra a categoria representada Figura 6.4, é possível observar que a equipe 1 considera os artefatos como um dos meios de comunicação. Alguns participantes relataram: “*Os protótipos e user stories são essenciais para a comunicação, pois contém basicamente os pré-requisitos do cliente e detalhes de validação*” – Desenvolvedor 1 da Equipe 1 e “*Os protótipos e user stories são centralizadores das informações solicitadas pelo cliente/usuário. Com isso, o time pode consultar as regras de negócio e validar os cenários*” – PO da Equipe 1 .

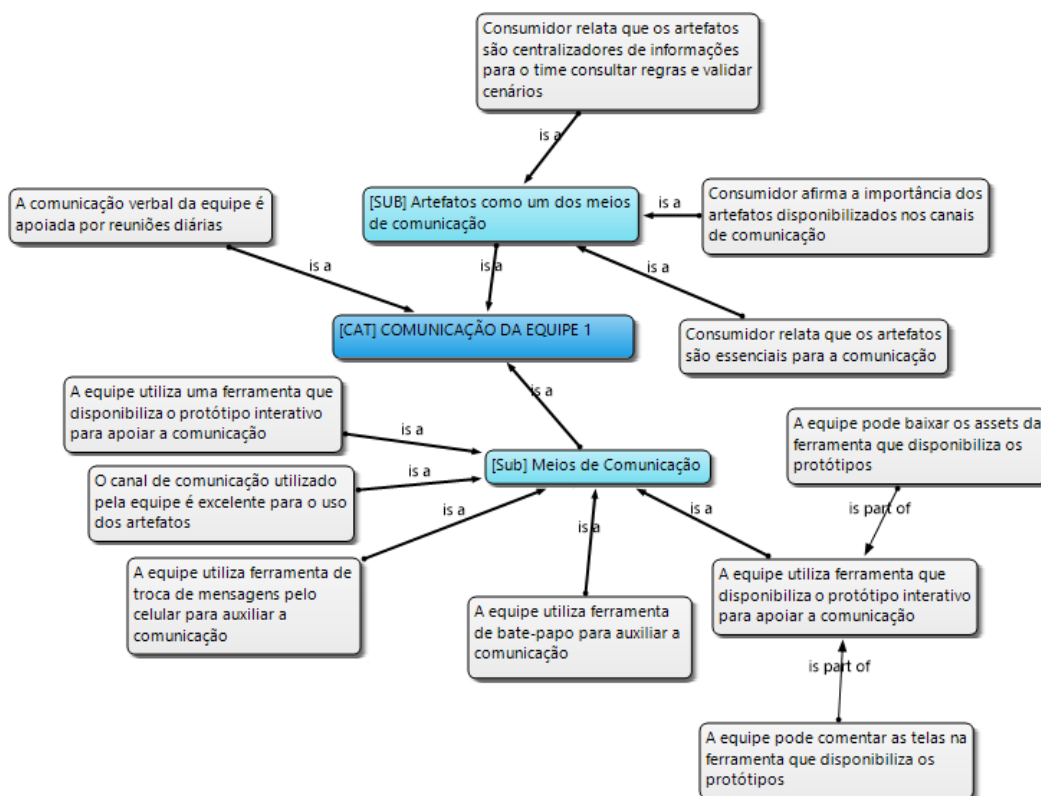


Figura 6.4: Categoria que Representa a Comunicação da Equipe 1.

Além disso, a comunicação da equipe é apoiada por reuniões diárias e outros meios de comunicação, como ferramenta de bate papo e ferramentas que disponibilizam os artefatos. O Desenvolvedor 3 da Equipe 1 relatou: “*A ferramenta ‘X’ (omitida nesta descrição) é de extrema importância, pois além das regras para o desenvolvimento das user stories, recebo imediatamente a notificação, o que agiliza a minha atividade*”.

Em relação aos fatores que impactam a comunicação via artefato, os resultados indicam que a comunicação via artefato pode ser eficaz nesta equipe, como mostra a na Figura 6.5. As respostas dos fatores ‘Artefato como meio de Comunicação’ e ‘Conteúdo Comunicativo’ foram respondidos pelo produtor. O fator ‘Consumidores do Artefato’ representa as respostas dos três Desenvolvedores da Equipe 1.

ARTEFATO COMO MEIO DE COMUNICAÇÃO (RESPONDIDO PELO PRODUTOR DO ARTEFATO)	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
M1. Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.		1					
M2. Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.	1						
M3. Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.			1				
M4. Este artefato me auxilia no que eu gostaria de “falar” para a equipe sobre o domínio do problema.	1						
M5. Este artefato permite que o meu projeto para o software seja representado de forma clara para a equipe.	1						
M6. Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).		1					
M7. Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.		1					
CONTEÚDO COMUNICATIVO (RESPONDIDO PELO PRODUTOR DO ARTEFATO)							
Cont1. Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.		1					
Cont2. Não foram introduzidas informações desorganizadas no artefato.		1					
Cont3. Não foi introduzido o excesso de informações no artefato.		1					
Cont4. A quantidade de informações no artefato é suficiente para o entendimento da equipe.	1						
Cont5. Acredito que o conteúdo do artefato está claro e é de fácil compreensão.	1						
CONSUMIDORES DO ARTEFATO (RESPONDIDO PELOS CONSUMIDORES DO ARTEFATO)							
PC1. Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.	1	2					
PC2. Eu acho fácil compreender este artefato.	1	1	1				
PC3. Eu irei adotar esse artefato para compreender as atividades de desenvolvimento	1	2					
PC4. Eu compreenderei melhor o software com este artefato.		2	1				
PC5. Eu acho fácil usar este artefato.	2	1					

Figura 6.5: Fatores Relacionados à Comunicação via Artefato da Equipe 1.

Sobre a análise dos artefatos como meio de comunicação na Etapa 1 deste estudo, foram observados problemas de comunicabilidade em relação à falta de informações para um melhor entendimento do sistema e informações desatualizadas. No entanto, um destes problemas pode estar relacionado com o impacto das atividades da Equipe 1 nas informações dos artefatos, como mostra a categoria apresentada na Figura 6.6. Tal figura mostra a opção de não atualizar os protótipos devido à grande quantidade de atualização, como relata o PO da Equipe 1: “A desatualização muitas vezes ocorre devido à grande quantidade de telas ou quando a mudança na regra de negócio ocorre de forma hábil para verbal e não há tempo atualização”. Tal citação foi codificada como “Com a grande quantidade de atualização de telas, a equipe opta por Technical Debt” por ser a decisão tomada pela equipe para não afetar o custo da entrega do software em cada *Sprint*.

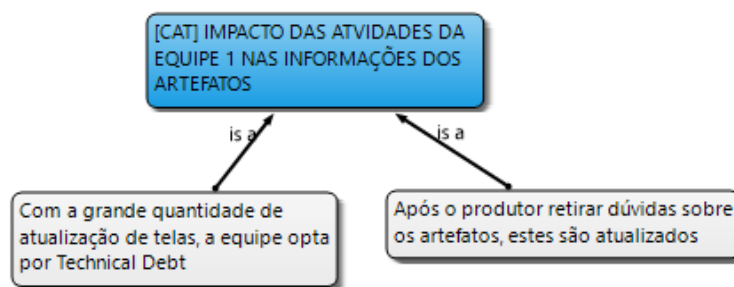


Figura 6.6: Impacto das Atividades da Equipe 1 nas Informações dos Artefatos.

6.2.2.2 Resultado dos Benefícios das DCs na Comunicação via Artefato da Equipe 1

A Tabela 6.3 apresenta os problemas de comunicabilidade nos artefatos identificados na etapa 1 do estudo em comparação com o relato do Designer da Equipe 1 após o emprego das DCs. Ressalta-se que o relato dos problemas de comunicabilidade não foi fornecido ao produtor dos artefatos. É possível perceber que as DCs forneceram apoio para reflexão do produtor na redução dos problemas de comunicabilidade.

Na etapa 1, foi realizada uma reunião com cada equipe e a pesquisadora principal deste trabalho. Nesta reunião, cada equipe apresentou uma visão geral de suas atividades e ressaltaram os principais artefatos utilizados como meio de comunicação. Em seguida, a pesquisadora apresentou um link com os questionários que as equipes deveriam responder como primeira etapa da pesquisa. Foram enviados questionários específicos para produtores e consumidores de artefatos nas equipes para coletar a percepção das

equipes sobre problemas que afetam a comunicação via artefato e analisar os fatores que impactam esta comunicação, como mostra o Apêndice B.

Problemas de Comunicabilidade	Emprego das DCs pelo Designer da Equipe 1
Falta de informações para o entendimento do sistema	Inclusão de mais informações nos protótipos – DC2 “Como entraram membros novos na equipe, eu acho que eu não consegui passar o que eu realmente queria nas primeiras telas, então adicionei outras informações”
	Uso de informação necessária e clara para a equipe e cliente – DC2 e DC4 “Em relação à documentação, eu procurei passar somente o necessário e procurei usar uma linguagem que todos compreendam porque essa documentação para vai o cliente e também fica disponibilizada para a equipe...”
Falta de informações atualizadas	Uso de informações corretas (devido à atualização do sistema) - DC1 “Como teve uma mudança na resolução da tela, eu fiz as telas com a resolução correta e as interações na ferramenta que utilizamos”
	Organização nas informações atualizadas – DC3 “Teve uma mudança em uma tela e com uma diretriz, eu fiquei pensando em como eles poderiam entender a minha lógica no protótipo e assim eu tentei organizar essas informações”

Tabela 6.3: Problemas de Comunicabilidade VS Benefícios obtidos com as DCs relatados pelo Produtor de Artefatos da Equipe 1.

Ainda sobre o apoio das DCs na comunicação via artefato, a Figura 6.7 apresenta a categoria que representa os benefícios das DCs para a comunicação via artefato da Equipe 1. De acordo com o produtor, as DCs reforçaram a comunicação entre os membros da equipe, uma vez que as DCs promoveram um olhar crítico deste sobre a sua por meio do protótipo, como mostra a seguinte citação: “As diretivas no meu ponto de vista só têm pontos positivos, porque elas reforçam a comunicação entre o time, que é essencial para o projeto. Utilizar as diretivas me levou a ter um olhar crítico em relação a minha comunicação com o time” – Designer da Equipe 1.

As DCs também promoveram a reflexão do produtor sobre os artefatos que deveriam ser disponibilizados para todos os membros da equipe nos canais de comunicação. O seguinte trecho mostra isso: “As diretivas me ajudaram a refletir, porque eu fui colocando em evidência as telas em todos os canais que eu poderia colocar” - Designer da Equipe 1. Ainda sobre a reflexão do produtor na melhoria da comunicação via artefato com base nas DCs, o produtor procurou perguntar da equipe as informações que poderiam estar disponíveis no artefato.

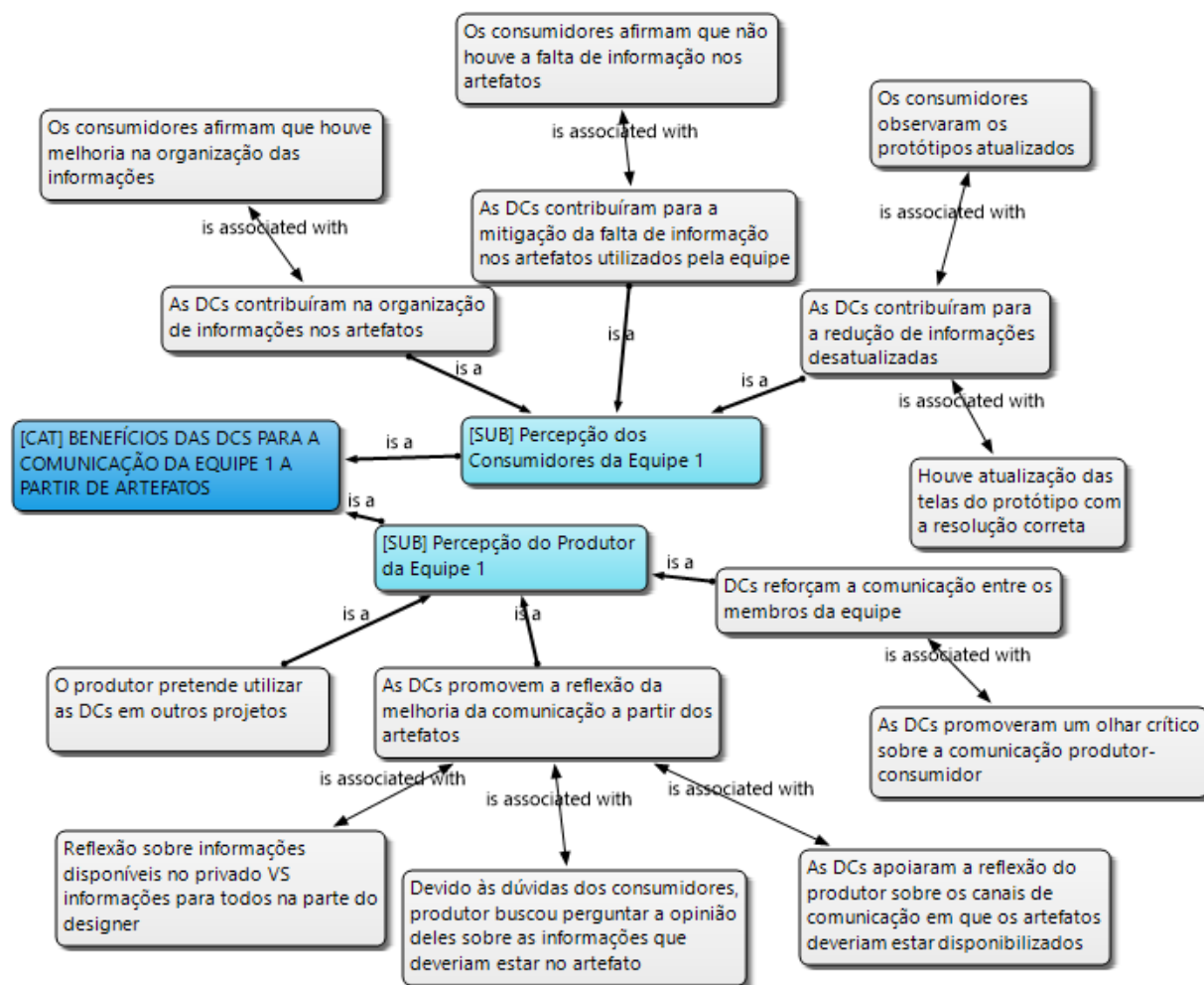


Figura 6.7: Benefícios obtidos para a Equipe 1 com a Comunicação via Artefato.

Sobre a percepção dos consumidores, estes observaram os protótipos atualizados, percebendo a redução de informações desatualizadas em comparação com as *sprints* anteriores: “*Os protótipos estão atualizados, mas a desatualização muitas vezes ocorre devido à grande quantidade de telas ou quando ocorre a mudança da regra ocorre de forma verbal e não há tempo hábil para atualização*” - Desenvolvedor 1 da Equipe 1.

Os consumidores também afirmaram que não houve falta de informação nos artefatos, percebendo que as DCs contribuíram para a mitigação da falta de informação nos artefatos, como mostra a seguinte citação: “*Nessa sprint não houve a falta de informação, mas a maioria das vezes faltam informações*” – Desenvolvedor 2 da Equipe 1. As DCs também contribuíram para uma melhor organização de informações nos

artefatos, como relata os consumidores: “*Nesta Sprint as informações estão organizadas, mas nem sempre foi organizada*” – Desenvolvedor 2 da Equipe 1.

6.2.3 Resultados da Equipe 2

Esta subseção apresenta os resultados obtidos com a investigação realizada na Equipe 2, uma equipe investigada em uma outra empresa, a qual foi chamada de empresa C2 no início deste capítulo. A análise dos dados da Equipe 2 foi realizada de maneira semelhante à análise realizada na Equipe 1.

6.2.3.1 Resultado da Comunicação via Artefato da Equipe 2

A comunicação da Equipe 2, de forma geral, é apresentada na Figura 6.8, representando a categoria de comunicação. A equipe considera o protótipo útil como meio de comunicação, pois é o principal artefato que apoiam a comunicação da equipe com o cliente e apoia a compreensão da equipe sobre as atividades de desenvolvimento, como destacaram os seguintes participantes: “*O protótipo é usado como o principal meio de comunicação entre o cliente e o time. É nele que o PO detalha os desejos do cliente*” – Designer da Equipe 2 e “*O PO organiza as informações e auxilia o entendimento do produto a ser desenvolvido*” – Desenvolvedor 2 da Equipe 2.

Além disso, a comunicação da equipe é apoiada por reuniões diárias, onde é possível retirar dúvidas, e ferramentas que são utilizadas como meio de comunicação, como ferramenta de bate papo e ferramentas que disponibilizam os artefatos, como os protótipos e requisitos. Sobre as ferramentas que disponibilizam os artefatos, o PO da Equipe 1 relatou: “*Através da ferramenta ‘X’ (omitida nesta descrição) são disponibilizados os protótipos, user stories e outros arquivos que usamos. Sobre os protótipos, todos da equipe podem acessar o link de casa protótipo nessa ferramenta*”.

Em relação aos fatores que impactam a comunicação via artefato, os resultados se mostraram positivos. A Figura 6.9 mostra as respostas contabilizadas dos produtores, PO e Designer da Equipe 2, para os fatores ‘Artefato como meio de Comunicação’ e ‘Conteúdo Comunicativo’. As respostas contabilizadas dos quatro Desenvolvedores da Equipe 2 para ‘Consumidores do Artefato’ também são apresentadas na Figura 6.9.

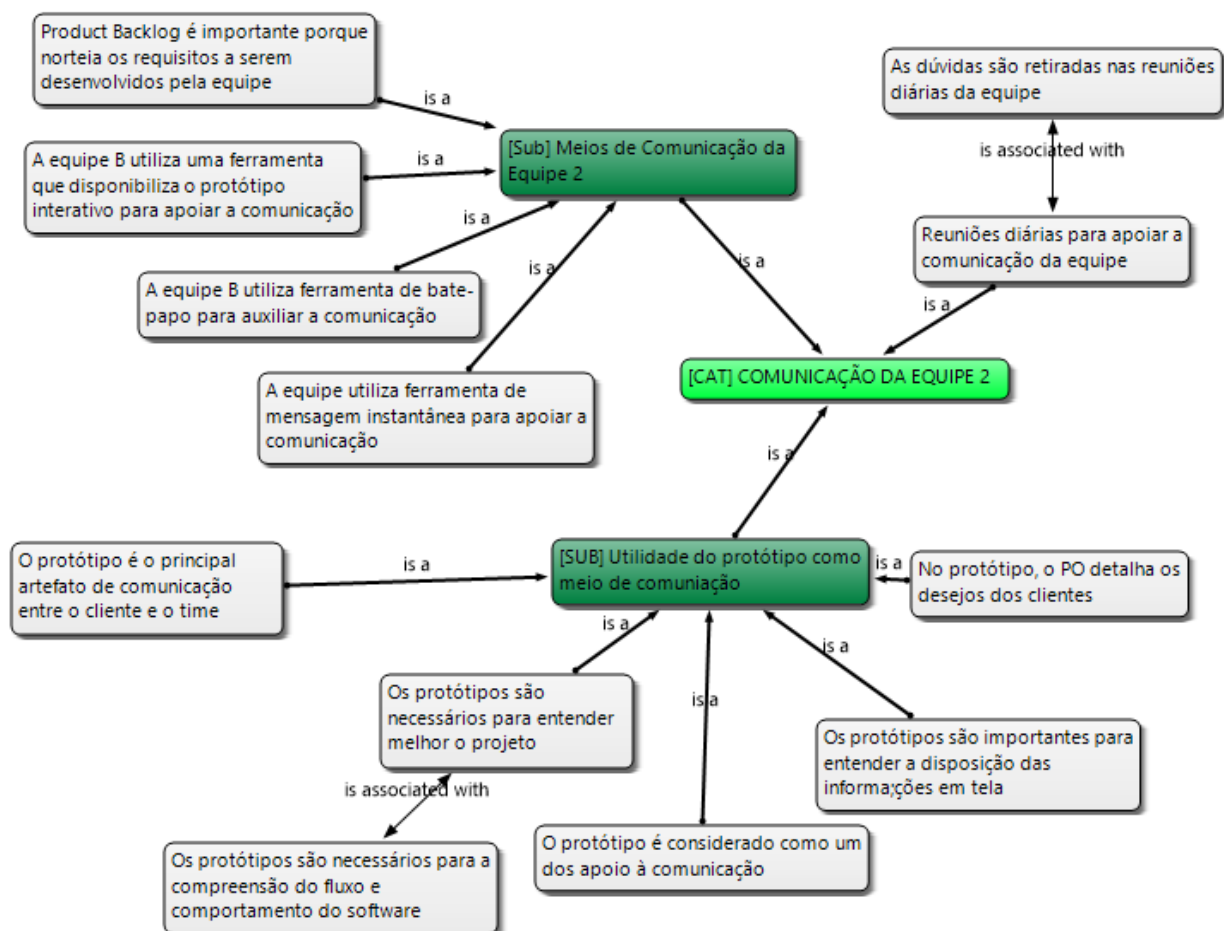


Figura 6.8: Categoria que Representa a Comunicação da Equipe 2.

Sobre a análise dos artefatos como meio de comunicação na Etapa 1 deste estudo, foram observados problemas de comunicabilidade em relação à falta de informações para um melhor entendimento do sistema e informações desatualizadas. No entanto, um destes problemas pode estar relacionado com um dos códigos relacionados com a categoria impacto das atividades da Equipe 2 nas informações dos artefatos, apresentada na Figura 6.10, que é a falta de todos os critérios de aceitação no início da Sprint, como relata o PO da Equipe 2 “*No caso dos critérios de aceitação, as vezes a gente não tem toda a visibilidade do que é para ser feito naquele momento. Então a gente faz um cenário resumido de tela e valida os dados com o cliente para ele ver se a ideia está indo no caminho certo*”.

ARTEFATO COMO MEIO DE COMUNICAÇÃO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
M1. Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.		2					
M2. Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.		2					
M3. Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.		2					
M4. Este artefato me auxilia no que eu gostaria de “falar” para a equipe sobre o domínio do problema.		2					
M5. Este artefato permite que o meu projeto para o software seja representado de forma clara para equipe.	2						
M6. Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).	2	1					
M7. Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.		1	1				
CONTEÚDO COMUNICATIVO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)							
Cont1. Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.		2					
Cont2. Não foram introduzidas informações desorganizadas no artefato.	1				1		
Cont3. Não foi introduzido o excesso de informações no artefato.	1	1					
Cont4. A quantidade de informações no artefato é suficiente para o entendimento da equipe.	1		1				
Cont5. Acredito que o conteúdo do artefato está claro e é de fácil compreensão.		2					
CONSUMIDORES DO ARTEFATO (RESPONDIDO PELOS CONSUMIDORES DO ARTEFATO)							
PC1. Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.	1	2					
PC2. Eu acho fácil compreender este artefato.	1	1	1				
PC3. Eu irei adotar esse artefato para compreender as atividades de desenvolvimento	1	2					
PC4. Eu compreenderei melhor o software com este artefato.		2	1				
PC5. Eu acho fácil usar este artefato.	2	1					

Figura 6.9: Fatores Relacionados à Comunicação via Artefato da Equipe 2.

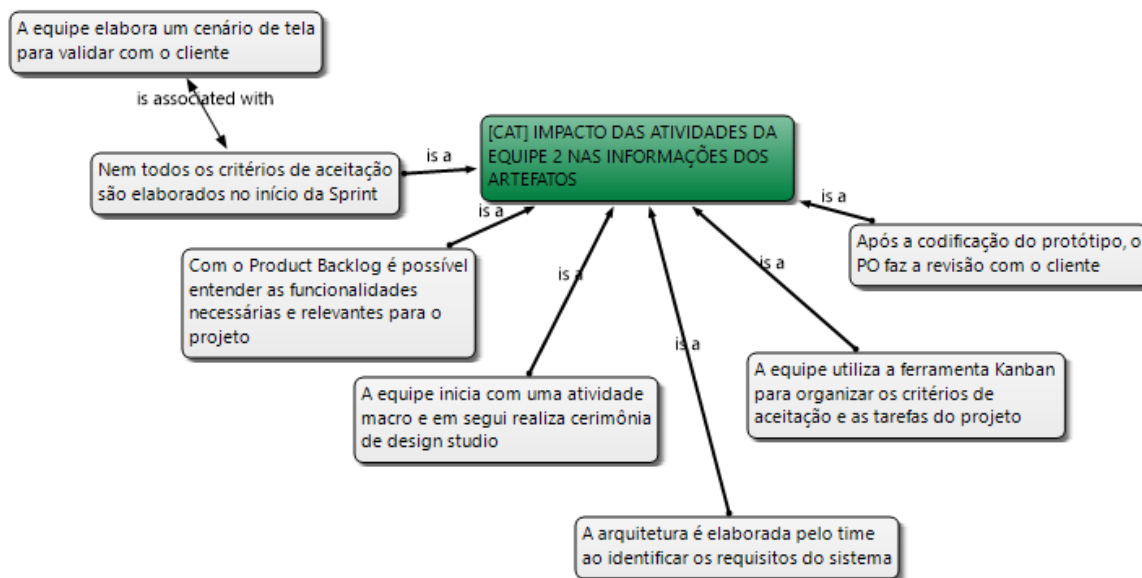


Figura 6.10: Impacto das Atividades da Equipe 2 nas Informações dos Artefatos.

6.2.3.2 Resultado dos Benefícios das DCs na Comunicação via Artefato da Equipe 2

A Tabela 6.4 apresenta os problemas de comunicabilidade dos artefatos, identificados na etapa 1 do estudo, em comparação com a percepção do produtor após o emprego das DCs.

Problemas de Comunicabilidade	Emprego das DCs pelos Produtores da Equipe 2
Falta de informações para o entendimento do sistema	Disponibilização dos links de navegação no protótipo – DC2 “Antes eu deixava os desenvolvedores escolherem o comportamento (navegação). Eu não enviava o link na documentação, mas agora eu vi que dá para colocar os links na ferramenta que utilizamos para disponibilizar os protótipos. Antes eu mandava os links para eles pelo slack” – Designer da Equipe 2.
Falta de organização de informações	Uso de informações organizadas na sequência adequada no Kanban – DC3 “Nós buscamos usar a questão da sequência das tarefas. A gente já teve projetos que teve esse problema na sequência de informações” – Analista de Sistemas da Equipe 2.

Tabela 6.4: Problemas de Comunicabilidade VS Benefícios obtidos com as DCs na Equipe 2 relatados pelos Produtores de Artefatos da Equipe 2.

Com os relatos apresentados, é possível perceber que as DCs forneceram apoio para o produtor reduzir os problemas de comunicabilidade nos artefatos produzidos na Sprint investigada. Ainda sobre o apoio das DCs na comunicação via artefato, a Figura 6.11

apresenta a categoria que representa os benefícios das DCs para a comunicação via artefato da Equipe 2.

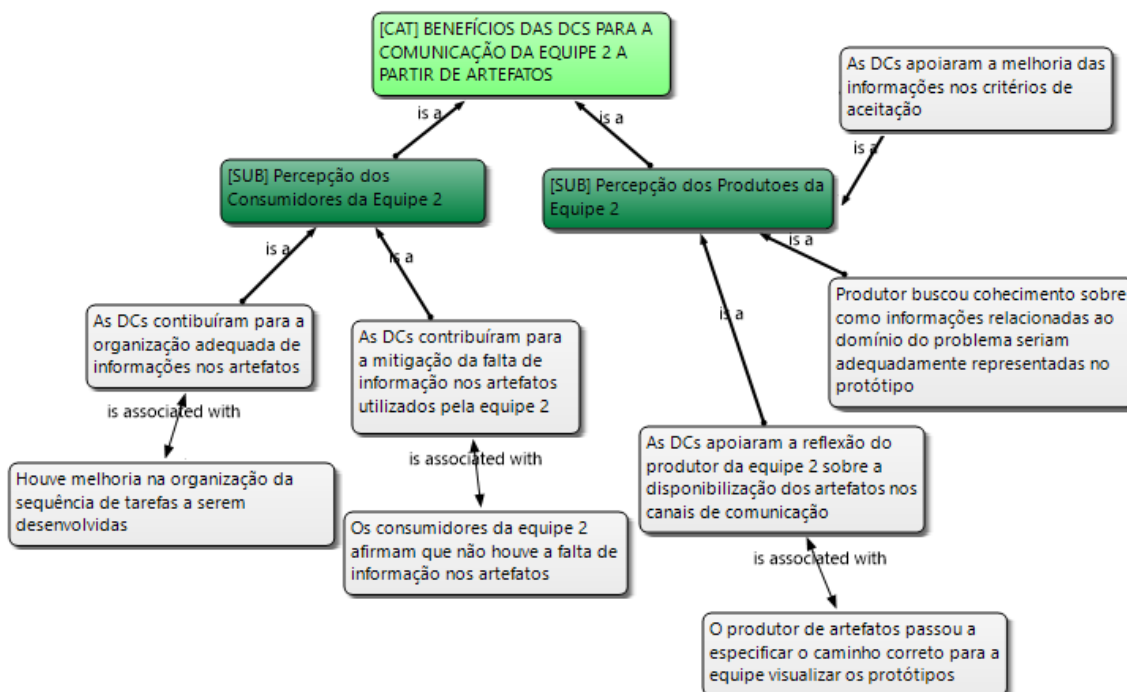


Figura 6.11: Benefícios obtidos para a Equipe 2 com a Comunicação via Artefato.

De acordo com os produtores, as DCs apoiaram a melhoria de informações nos critérios de aceitação e da inserção de informações adequadas nos protótipos, além de apoiarem a reflexão dos produtores sobre a disponibilização dos artefatos nos canais de comunicação, como mostram as seguintes citações: “*Eu tentei melhorar um pouquinho mais sobre a questão de forma clara. Eu colocava frases muito grandes e nesta Sprint eu resumi para os critérios de aceitação ficarem mais claros*” – Designer da Equipe 2; “*As diretivas serviram para eu refletir sobre onde eu deveria colocar informações para a equipe*” – Analista de Sistemas da Equipe 2.

Sobre a percepção dos consumidores, estes observaram as informações organizadas e afirmaram que não houve a falta de informação nos artefatos, percebendo que as DCs contribuíram para a mitigação da falta de informação nos artefatos, como mostra a seguinte citação: “*Informações sobre telas eu não senti falta. Não houve falta de informação, mas quando eu tenho dúvida, eu tiro dúvida com a designer.*” – Desenvolvedor 1 da Equipe 2.

6.2.4 Percepções sobre o Uso das DCs

Essa subseção apresenta os resultados qualitativos em relação ao uso e percepções dos produtores da Equipe 1 e Equipe 2 sobre as DCs. Em relação ao uso das DCs, tais resultados indicaram a forma como os produtores aplicaram as DCs nos artefatos de software, como a documentação, protótipos e organização das tarefas no Kanban com objetivo de melhorar a sua comunicabilidade. A Figura 6.12 e Figura 6.13 apresenta as variações de uso das DCs para a categoria uso das DCs para a melhoria da comunicabilidade dos artefatos.

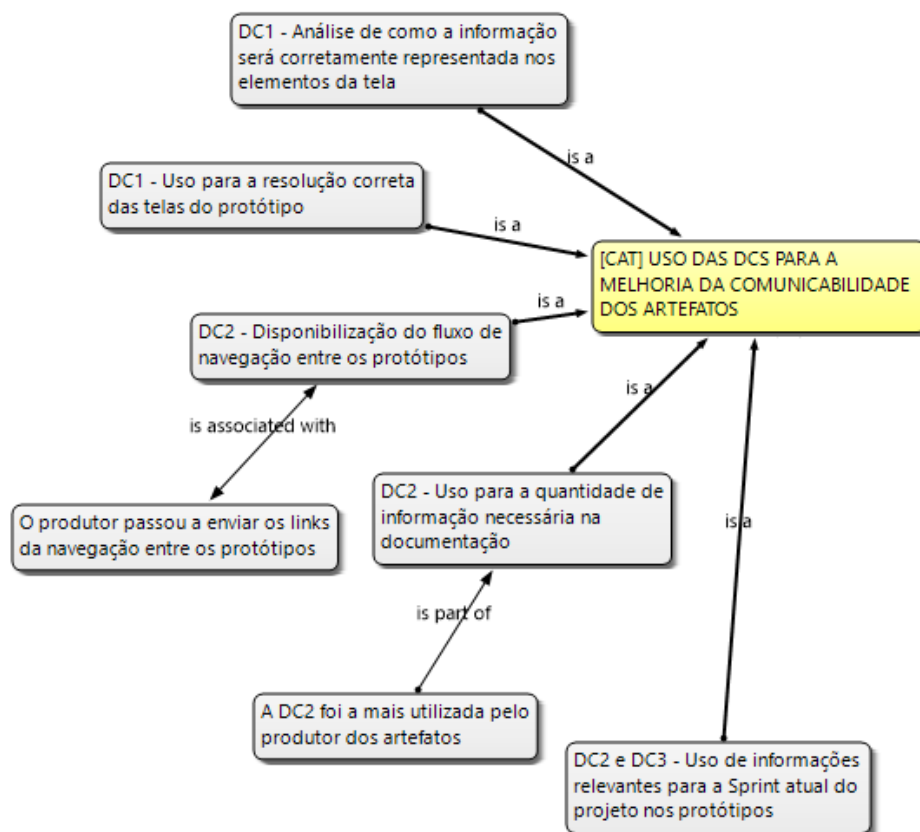


Figura 6.12: Variações do Uso da DC1, DC2 e combinação da DC2 e DC3 nos Artefatos.

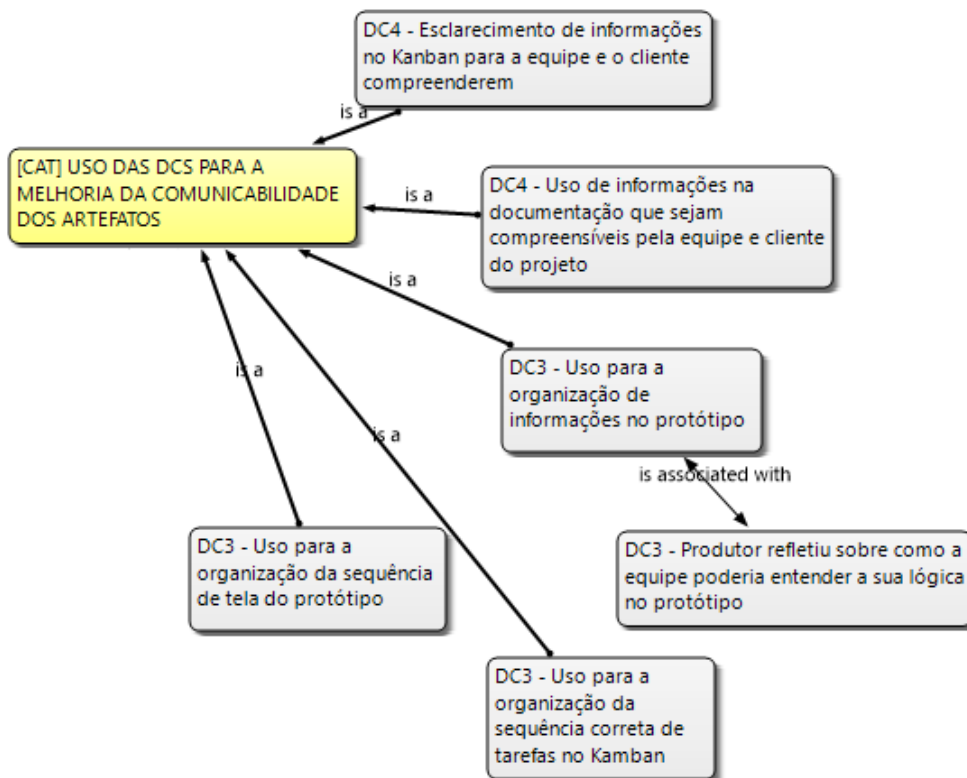


Figura 6.13: Variações do Uso da DC3 e DC4 nos Artefatos.

Em relação à DC1, essa foi aplicada somente nos protótipos para o produtor inserir apenas informações corretamente apresentadas nas telas em relação à disponibilização de elementos e resolução, como mostra a seguinte citação: *“Como teve mudança na resolução da tela, eu fiz todas as telas com a resolução correta”* – Designer da Equipe 1.

Em relação à DC2, essa foi aplicada nos protótipos, incluindo a disponibilização de informação da navegação, na documentação, para o uso de informação necessária, como mostra a citação do Designer da Equipe 1: *“Na documentação, tem uma diretriz que fala somente o necessário. Eu procurei passar somente informação necessária”*. Além disso, houve uma combinação da DC2 e DC3 para a inserção de informações relevantes na documentação da *Sprint* atual, pois o mesmo designer relatou: *“Eu me atrapalhava no início do projeto com a questão de organizar as informações que estão no projeto. As vezes eu colocava informação desnecessária em relação às etapas de desenvolvimento. Agora estou usando as informações necessárias que serão utilizadas no momento”*.

A DC3 foi aplicada nos protótipos e no Kanban. Nos protótipos, esta foi utilizada para a organização da sequência das telas e organização de informações. No Kanban, esta

foi utilizada para organizar as tarefas adequadas, como relatado pelo PO da Equipe 2: “*Nós buscamos organizar a sequência das tarefas. Nós já tivemos projetos com esse problema*”.

Por fim, sobre o uso da DC4, essa foi aplicada na documentação e nas informações do Kanban. Tanto na documentação e nas informações disponíveis no Kanban, os produtores utilizaram esta diretiva para tornar as informações do projeto claras para a equipe e para o cliente. O Designer da Equipe 2 citou: “*Procurei usar uma linguagem que todos compreendam porque essa documentação vai para o cliente e fica disponibilizada para a equipe*”.

Em relação às percepções dos produtores, a Figura 6.14 e Figura 6.15 apresentam a categoria que mostra as percepções sobre o uso das DCs, a qual possui três subcategorias: apoio na melhoria do conteúdo dos artefatos, apoio na reflexão do produtor sobre a comunicação via artefato e pontos a serem melhorados na proposta das DCs.

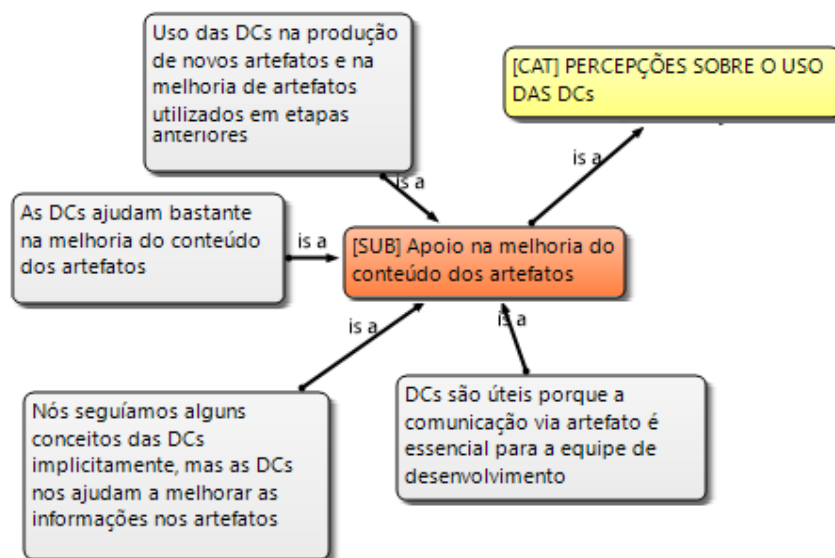


Figura 6.14: Percepções sobre as DCs: Apoio na melhoria do conteúdo dos artefatos.

Em relação à subcategoria apoio na melhoria do conteúdo dos artefatos, na Figura 6.14, os produtores afirmaram que as DCs ajudam bastante nesta questão e isto é útil porque a comunicação via artefato é essencial para a equipe de desenvolvimento, como mostram as seguintes citações: “*Eu acho que tudo que está nas diretivas é válido porque como tem muitos desenvolvedores é essencial essa comunicação*” - Designer da Equipe 1; “*Na minha opinião, acho que elas ajudam bastante. Não é algo que demanda tempo*”

para aplicar nos artefatos que utilizamos” - Analista de Sistemas da Equipe 2. Sobre as DCs em si, foi relatado pelos produtores que alguns seguiam os conceitos destas de forma implícita, como mostra a seguinte citação: “Sobre as diretivas, a gente já até fazia algumas delas. Olhando para as diretivas, é possível enxergar como a informação vai ficar redonda na tela” - Designer da Equipe 2. Além disso, um produtor decidiu aplicar as DCs em artefatos desenvolvidos nas sprints anteriores com o objetivo de melhorar o conteúdo destes: “Como entraram membros novos na equipe, eu acho que eu não consegui passar o que eu realmente queria nas primeiras telas, então adicionei outras informações” - Designer da Equipe 1.

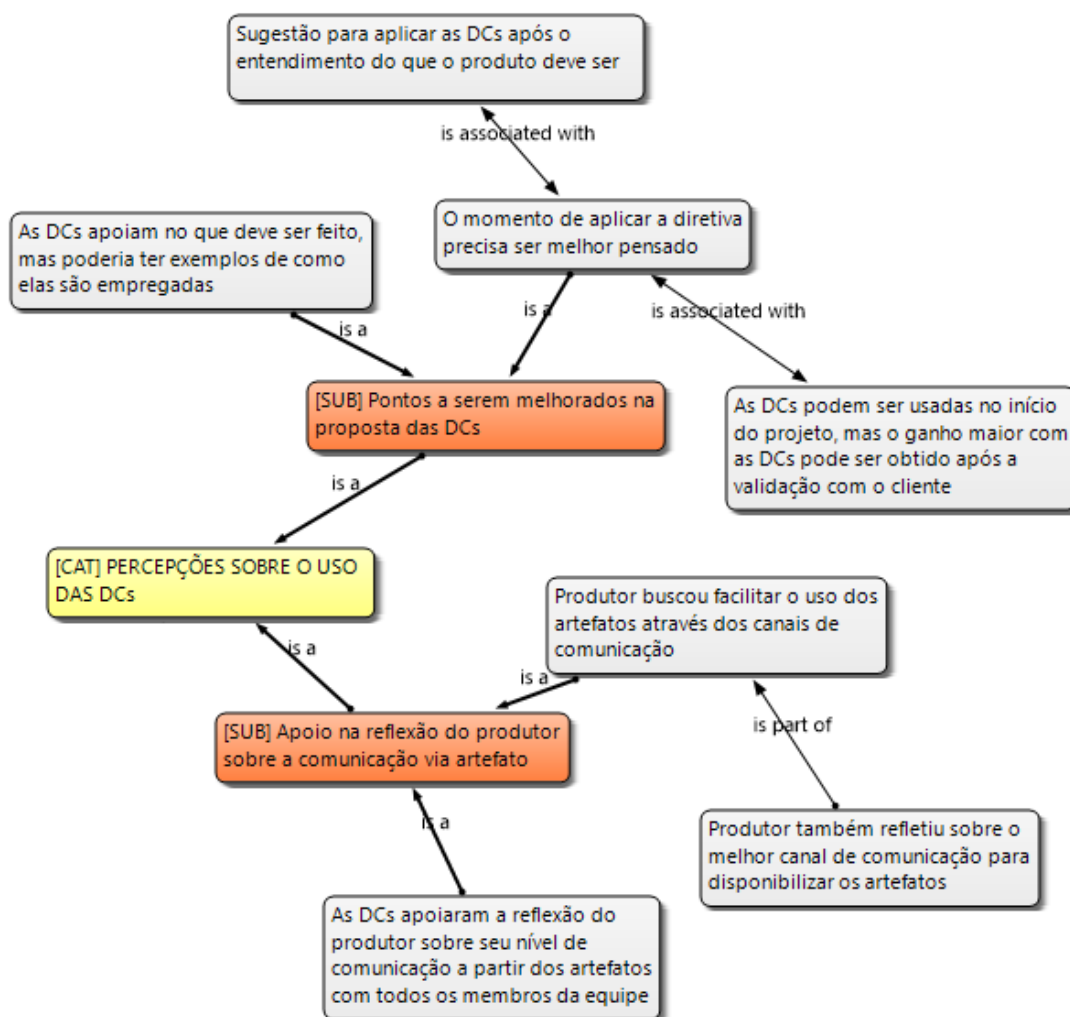


Figura 6.15: Percepções sobre as DCs: Apoio na reflexão do produtor sobre a comunicação via artefato.

A subcategoria apoio na reflexão do produtor sobre a comunicação via artefato, na Figura 6.15, percebeu-se que as DCs apoiaram a reflexão dos produtores sobre: o seu

nível de comunicação com os demais membros da equipe e os canais de comunicação em que os artefatos são utilizados. A seguinte citação ressalta isso: *“Na ferramenta X (omitido – ferramenta que a equipe utiliza para disponibilizar os artefatos) tem um grupo do projeto e aí eu procurei passar a documentação mais no grupo de design. Com as diretivas, eu percebi que não estava passando as mesmas informações para todo mundo. As diretivas me ajudaram a atentar sobre as informações que são disponibilizadas e como eu posso passar essas informações”* - Designer da Equipe 1.

A subcategoria pontos a serem melhorados na proposta das DCs, na Figura 6.15, mostrou as percepções dos produtores sobre como as DCs devem ser compreendidas, pois o código as DCs apoiam no que deve ser feito, mas poderia ter exemplos de como elas são empregadas mostra que os produtores podem utilizar as DCs gerais e DCs específicas. Esse código representa a percepção do Designer da Equipe 2, o qual utilizou apenas as DCs gerais, como mostra a citação: *“Ela mostra o que deve ser feito, mas poderia ter exemplo de como devem ser empregas. Acredito que um case cairia bem”*. Portanto, as DCs podem ser melhor compreendidas quando o produtor tem conhecimento destas de forma genérica e específicas, as quais mostram exemplos do seu emprego. O código o momento de aplicar a diretiva precisa ser melhor pensado mostra que as DCs devem ser empregadas na etapa de Externalização dos artefatos, apesar dos produtores relatarem que é possível utilizar as DCs na etapa de Ideação, como mostra a seguinte citação: *“Talvez fique melhor a aplicação delas quando tivermos todo o entendimento do que o produto deve ser, e muitas vezes esse entendimento vem após muitas interações com o cliente. No mais, elas estão boas e simples de entender”* - Analista de Sistemas da Equipe 2.

6.2.5 Discussão dos Resultados

Este estudo foi realizado com o objetivo de compreender os benefícios das DCs em relação à comunicação eficaz de duas equipes a partir de artefatos de software. Os resultados indicaram que as DCs apoiaram na reflexão dos produtores, em relação ao conteúdo informacional e dos canais que viabilizam esse tipo de comunicação, sobre sua comunicação com os demais membros da equipe. Isso demonstra que as DCs alcançaram o seu objetivo, pois isto proporcionou a comunicação eficaz via artefato nas equipes. Com este estudo, também foi possível observar o emprego das DCs para diferentes pontos de

melhorias em relação à comunicabilidade dos artefatos, além de ressaltar que as DCs podem ser melhor empregadas na etapa de Externalização dos artefatos, conforme a definição apresentada por Chaudron *et al.* (2018), descrita no Capítulo 3.

6.3 Estudo 2 – Apoio das DCs para o Diagnóstico de Artefatos

O objetivo deste estudo foi investigar o emprego das DCs no diagnóstico de artefatos para analisar se estas podem apoiar a reflexão de produtores e consumidores de artefatos em relação aos problemas de comunicabilidade em projetos de software concluídos. Com esse reflexo, os membros das equipes de software poderão atentar para a importância desse tipo de comunicação e adotar as DCs durante a produção de artefatos em novos projetos. As DCs foram adaptadas para tal diagnóstico de forma que o investigador pudesse responder questões que indicassem tais problemas, como mostra o exemplo das questões abaixo baseadas na DC1. A adaptação realizada para o diagnóstico de comunicabilidade dos artefatos investigados nas equipes está no Apêndice F.

As informações no artefato contêm declarações que não são verdadeiras?

- **Existem informações que não fazem parte do domínio do problema?**
 - ❖ Caso positivo, informações que não fazem parte do sistema podem ter sido codificadas.
- **Existem informações inconsistentes?**
 - ❖ Caso positivo, isto pode ter causado a codificação de funcionalidades inconsistentes/ou incorretas.
- **Existem informações desatualizadas?**
 - ❖ Caso positivo, isto pode ter causado a codificação de funcionalidades não necessárias/ incorretas.

Este estudo foi realizado com três equipes de desenvolvimento de software: Equipes 3 e 4 da empresa C1 e Equipe 5 da organização C3 (a Equipe 5), uma universidade brasileira em parceria com o exército brasileiro para o desenvolvimento de um sistema. Ressalta-se que estas equipes não estão envolvidas no estudo apresentado na seção anterior. A caracterização das equipes está descrita na Tabela 6.5.

Equipe 3

Experiência

Equipe 4

Experiência

Equipe 5

Experiência

analista de sistemas (product owner- PO)	9 anos	analista de sistemas (product owner - PO)	5 anos	analista de sistemas (product owner- PO)	20 anos
designer	6 anos	designer	5 anos	designer	9 anos
desenvolvedor 1	4 anos	desenvolvedor 1	13 anos	desenvolvedor 1	7 anos
desenvolvedor 2	6 anos	desenvolvedor 2	3 anos	desenvolvedor 2	20 anos
desenvolvedor 3	20 anos	desenvolvedor 3	12 anos	desenvolvedor 3	5 anos
desenvolvedor 4	6 anos	desenvolvedor 4	2 anos	desenvolvedor 4	16 anos
		desenvolvedor 5	8 meses	desenvolvedor 5	4 anos
				desenvolvedor 6	4 anos
				desenvolvedor 7	19 anos
				desenvolvedor 8	12 anos
				desenvolvedor 9	12 anos
				desenvolvedor 10	3 anos
				desenvolvedor 11	10 anos
				desenvolvedor 12	3 anos

Tabela 6.5: Equipe 3, Equipe 4 e Equipe 5 do Estudo 2.

6.3.1 Planejamento e Execução do Estudo

O estudo foi planejado para ser realizado em duas etapas: (1) compreensão de problemas que afetam a comunicação a partir de artefatos, análise dos fatores que impactam este tipo de comunicação e análise do relato dos consumidores dos artefatos e (2) uso das DCs adaptadas para o diagnóstico dos artefatos – realizado pelos produtores. Além disso, variações sobre o uso das DCs no diagnóstico dos artefatos poderiam ser identificadas. As etapas realizadas no estudo foram esquematizadas na Figura 6.16.

Na etapa 1, foi realizada uma reunião com cada equipe e a pesquisadora principal deste trabalho. Nesta reunião, cada equipe apresentou uma visão geral de suas atividades e ressaltaram os principais artefatos utilizados como meio de comunicação. Em seguida, a pesquisadora apresentou um link com os questionários que as equipes deveriam responder como primeira etapa da pesquisa. Foram enviados questionários específicos para produtores e consumidores de artefatos nas equipes para coletar a percepção das equipes sobre problemas de comunicabilidade e analisar os fatores que impactam esta comunicação, como mostra o Apêndice B. Em seguida, a pesquisadora conduziu um treinamento com o(s) produtor(es) de artefatos sobre o uso das DCs e seus benefícios antes dos consumidores utilizarem estes em suas atividades. Além disso, a pesquisadora apresentou as DCs para o diagnóstico de problemas de comunicabilidade que podem afetar a comunicação.

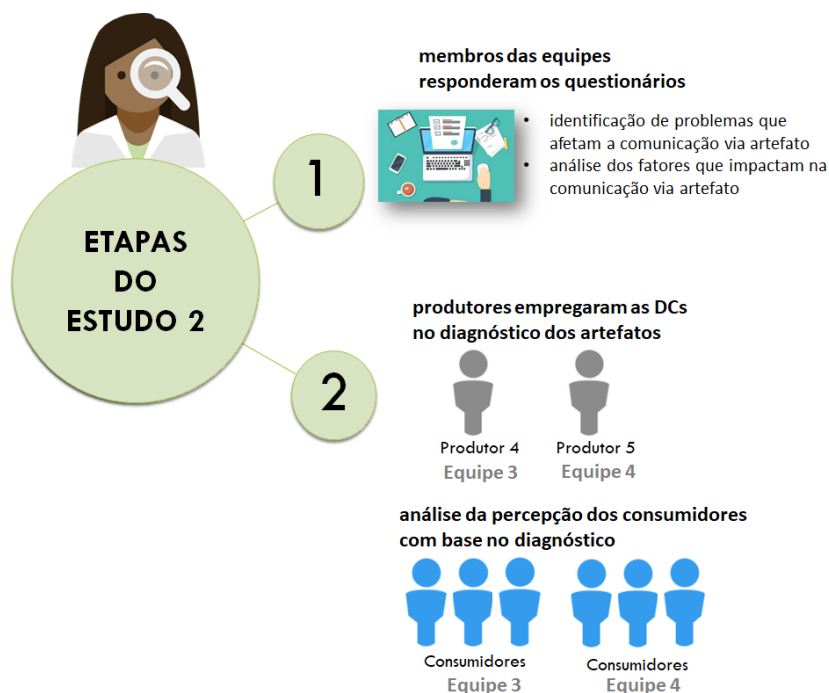


Figura 6.16: Etapas para a Condução do Estudo 2.

Na etapa 2, os produtores das equipes aplicaram o diagnóstico com as DCs nos artefatos utilizados na última *Sprint* de desenvolvimento realizado pelas equipes. Além disso, a pesquisadora principal realizou uma entrevista com a equipe, utilizando perguntas com base nas DCs adaptadas para o diagnóstico de artefatos com o objetivo de capturar a percepção dos consumidores dos artefatos. A Equipe 3 e Equipe 4 utilizaram protótipos como apoio à comunicação e a Equipe 5 utilizou protótipos e casos de uso.

As Subseções 6.3.2, 6.3.3 e 6.3.4 apresentam resultados de duas equipes investigada na empresa C1 e uma equipe da empresa C3, descritas no início deste capítulo. Nestas três subseções, são apresentados os resultados relacionados à comunicação das equipes e os resultados do diagnóstico de artefatos com apoio das DCs. A Subseção 6.3.5 apresenta o resultado das percepções dos participantes sobre as DCs para o diagnóstico de artefatos.

6.3.2 Resultados da Equipe 3

6.3.2.1 Resultado da Comunicação via Artefato da Equipe 3

Em relação ao apoio dos artefatos na comunicação da equipe 3, da empresa C, este é apresentado na Figura 6.17. Nesta categoria, foi possível observar a percepção e a forma de utilizar estes artefatos pela Equipe 3. A equipe considera o protótipo importante para as atividades de desenvolvimento, como relatado pelo Desenvolvedor 3 “*O protótipo é*

essencial para o projeto, uma ferramenta de apoio assim como os critérios de aceitação” . Além disso, o protótipos são úteis como meio de comunicação: “Acredito ser um bom meio de comunicação, pois além de oferecer a informação de como a tela deve ficar após a implementação, mostra também a navegabilidade” – PO da Equipe 3.

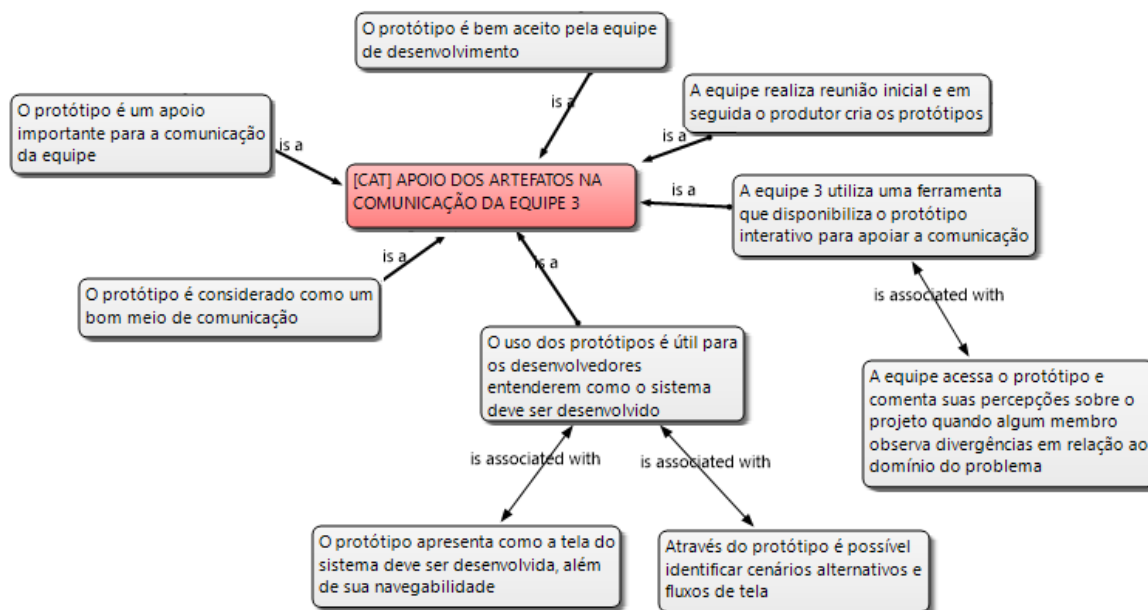


Figura 6.17: Categoria que Representa o Apoio dos Artefatos na Comunicação da Equipe 3.

Em relação aos fatores que impactam a comunicação, os produtores e consumidores da Equipe 3 responderam as questões que avaliam o ‘Artefato como meio de Comunicação’, ‘Conteúdo Comunicativo’ e ‘Consumidores do Artefato’, cujas respostas estão contabilizados na Figura 6.18. Os resultados indicam que estes fatores não impactaram na comunicação via artefato da Equipe 3.

A Figura 6.19 apresenta a categoria impacto das atividades da equipe 3 nas informações dos artefatos. Essa categoria representa práticas da equipe que podem impactar nas informações dos artefatos, como a falta de telas com resoluções diferentes, como relata o Desenvolvedor 3 da Equipe 3: “às vezes falta protótipos de telas diferentes devido à limitação da ferramenta de design. No caso da variação de tela, quando tem algo complexo, pedimos para o designer criar um protótipo”. A desatualização de informação por novas propostas de desenvolvimento também ocorre nessa equipe: “Tem uma falha, que é de toda equipe, inclusive é uma lição aprendida. Às vezes tomamos

decisões diferentes sobre o projeto e algum membro da equipe não está presente. Isso fica desatualizado nos protótipos” – Desenvolvedor 4 da Equipe 3.

ARTEFATO COMO MEIO DE COMUNICAÇÃO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
M1. Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.	1						
M2. Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.	1						
M3. Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.	1						
M4. Este artefato me auxilia no que eu gostaria de “falar” para a equipe sobre o domínio do problema.	1						
M5. Este artefato permite que o meu projeto para o software seja representado de forma clara para equipe.	1						
M6. Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).	1						
M7. Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.		1					
CONTEÚDO COMUNICATIVO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)							
Cont1. Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.	1						
Cont2. Não foram introduzidas informações desorganizadas no artefato.	1						
Cont3. Não foi introduzido o excesso de informações no artefato.	1						
Cont4. A quantidade de informações no artefato é suficiente para o entendimento da equipe.		1					
Cont5. Acredito que o conteúdo do artefato está claro e é de fácil compreensão.	1						
CONSUMIDORES DO ARTEFATO (RESPONDIDO PELOS CONSUMIDORES DO ARTEFATO)							
PC1. Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.	2	2					
PC2. Eu acho fácil compreender este artefato.	1	2	1				
PC3. Eu irei adotar esse artefato para compreender as atividades de desenvolvimento	2	2					
PC4. Eu compreenderei melhor o software com este artefato.	2	2					
PC5. Eu acho fácil usar este artefato.	1	3					

Figura 6.18: Fatores Relacionados à Comunicação via Artefato da Equipe 3.

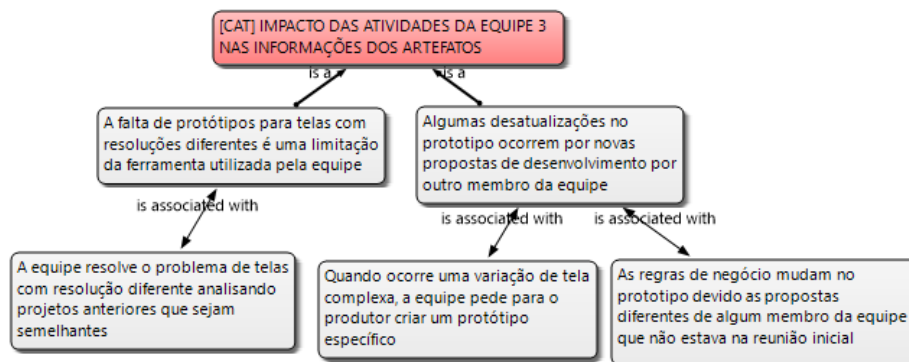


Figura 6.19: Impacto das Atividades da Equipe 3 nas Informações dos Artefatos.

6.3.2.2 Resultado do Diagnóstico dos Artefatos com o Apoio das DCs

Os resultados do diagnóstico dos protótipos em relação aos problemas de comunicabilidade são caracterizados na Figura 6.20, pois os protótipos representados nesta figura não pertencem de fato ao projeto elaborado pela Equipe 4. O diagnóstico foi realizado com base no resultado da autoavaliação do designer e da percepção dos desenvolvedores sobre os protótipos, tendo por base as DCs, sendo parte das citações destes representadas na Figura 6.20.

DC1: Falta de Atualização nos Protótipos

“Algumas vezes surgem muitas mudanças e não tem como atualizar todo o protótipo, mas o time entrou em acordo com a seguinte estratégia: quando existe alguma desatualização no protótipo, é sinalizado com um comentário”
Designer da Equipe 3

DC2: Falta de Fluxo de Navegação entre as Telas dos

“Às vezes falta algum fluxo de interação, aí a equipe fala com a designer para resolver”
Desenvolvedor 1 da Equipe 3



Figura 6.20: Diagnóstico dos Protótipos da Equipe 4.

Os resultados mostraram os problemas de comunicabilidade que causaram interrupções na comunicação da Equipe 3, como falta de fluxo de navegação entre as telas e informações desatualizadas nos protótipos. Em relação à falta de fluxo de navegação, esse problema de comunicabilidade pode não ter causado falhas totais de comunicação porque a equipe trabalha no mesmo ambiente e os membros da equipe retiraram suas dúvidas. Além disso, o produtor realizou a inserção de informação relacionada ao fluxo de navegação. Sobre as informações desatualizadas, a equipe adotou o uso de comentários para esclarecer isso.

6.3.3 Resultados da Equipe 4

6.3.3.1 Resultado da Comunicação via Artefato da Equipe 4

A comunicação da Equipe 4, da empresa C1, apoiada pelos artefatos é apresentada na Figura 6.21. Nesta categoria, foi possível observar que a equipe considera o protótipo como um apoio importante para o desenvolvimento de software, sendo este essencial para o projeto, como relata o Desenvolvedor 4: *“O protótipo é essencial para validar as regras de negócio e testar antes de programar”*.

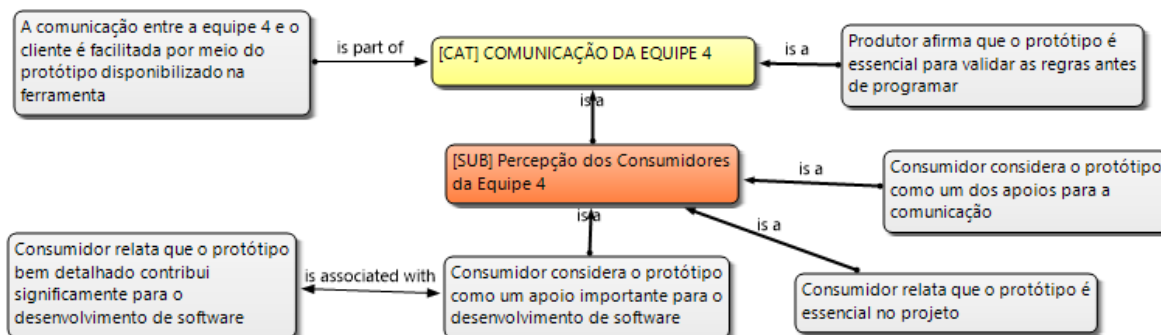


Figura 6.21: Categoria que Representa a Comunicação da Equipe 4 Apoiada por Artefatos.

A equipe afirmou que os protótipos auxiliam a comunicação da equipe com o cliente: *“Como nosso cliente fica totalmente externo, a comunicação é complicada e é difícil ele passar alguma ideia para nós. Às vezes não dá para ele estar em uma reunião de videoconferência. Com a ferramenta X (ferramenta onde o protótipo fica disponibilizado), é mais fácil para ele comentar as ideias dele sobre o projeto”* – Desenvolvedor 2. A Figura 6.22 apresenta a categoria que representa práticas da equipe.

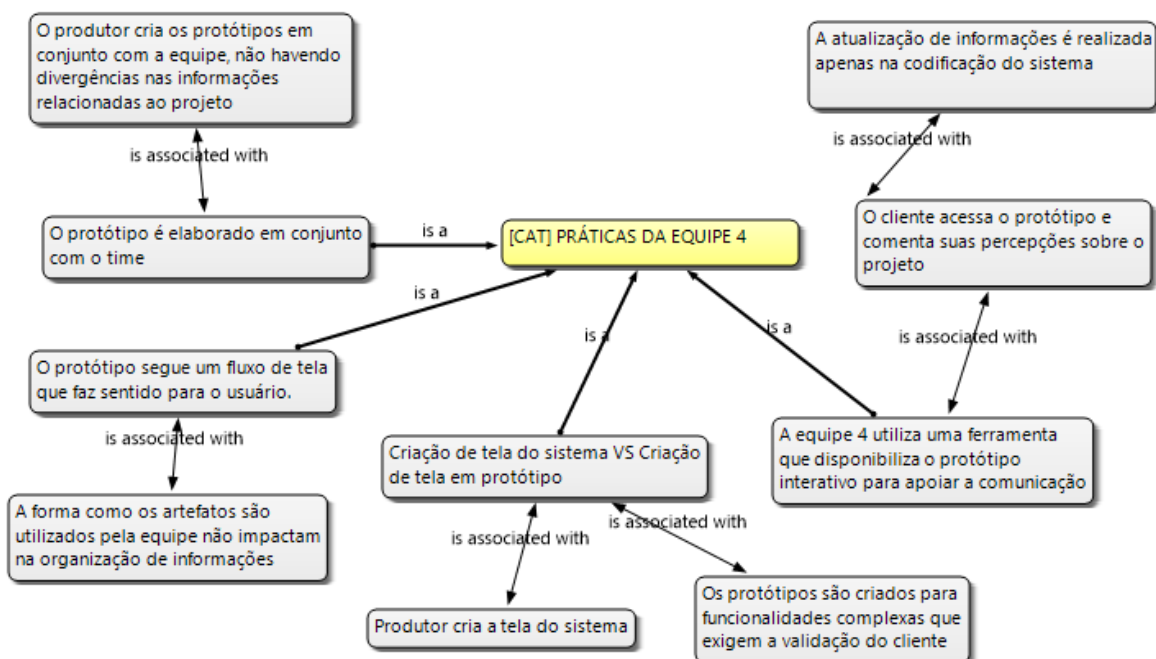


Figura 6.22: Práticas da Equipe 4 que Impactam a Comunicação via Artefato.

Em relação aos fatores que impactam a comunicação via artefato, as respostas dos produtores e consumidores foram contabilizadas e representadas na Figura 6.23.

ARTEFATO COMO MEIO DE COMUNICAÇÃO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
M1. Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.	1						
M2. Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.	1						
M3. Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.	1						
M4. Este artefato me auxilia no que eu gostaria de “falar” para a equipe sobre o domínio do problema.	1						
M5. Este artefato permite que o meu projeto para o software seja representado de forma clara para equipe.	1						
M6. Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).	1						
M7. Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.	1						
CONTEÚDO COMUNICATIVO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)							
Cont1. Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.	1						
Cont2. Não foram introduzidas informações desorganizadas no artefato.	1						
Cont3. Não foi introduzido o excesso de informações no artefato.	1						
Cont4. A quantidade de informações no artefato é suficiente para o entendimento da equipe.	1						
Cont5. Acredito que o conteúdo do artefato está claro e é de fácil compreensão.	1						
CONSUMIDORES DO ARTEFATO (RESPONDIDO PELOS CONSUMIDORES DO ARTEFATO)							
PC1. Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.	2	3					
PC2. Eu acho fácil compreender este artefato.	2	2	1				
PC3. Eu irei adotar esse artefato para compreender as atividades de desenvolvimento	2	3					
PC4. Eu compreenderei melhor o software com este artefato.	2	2	1				
PC5. Eu acho fácil usar este artefato.	3	2					

Figura 6.23: Fatores Relacionados à Comunicação via Artefato da Equipe 4.

Como aspectos positivos sobre a comunicação via artefato, esta equipe cria a ideia dos protótipos em conjunto, facilitando o entendimento mútuo dos membros da equipe: “O designer está sempre conosco nas reuniões de alinhamento. Ele faz o protótipo e nos apresenta para darmos opinião. Com isso, toda a equipe acaba criando o protótipo junto com o designer” – PO da Equipe 4. Os protótipos são criados apenas para funcionalidades complexas: “Nós só prototipamos no projeto quando a tela é muito complexa e precisamos ter o feedback do cliente... Como o designer tem a experiência com HTML, ele faz a tela e nós integramos com o back-end” - Desenvolvedor 2 da Equipe 4.

6.3.3.2 Resultado do Diagnóstico dos Artefatos com o Apoio das DCs

Os resultados do diagnóstico dos protótipos da Equipe 4 em relação aos problemas de comunicabilidade são caracterizados na Figura 6.24, uma vez que as informações do projeto não pode ser representado aqui nesta tese. O diagnóstico foi realizado com base no resultado da autoavaliação do designer e da percepção dos desenvolvedores sobre os protótipos com apoiadas DCs. Parte das citações destes são representadas na Figura 6.24.



Figura 6.24: Diagnóstico dos Protótipos da Equipe 4.

Foram observados problemas de comunicabilidade em relação à falta de informações, informações desatualizadas e informações explícitas. Outros problemas foram identificados no diagnóstico, mas estes não afetaram a comunicação. Sobre essas interrupções causadas na comunicação da Equipe 4, tal equipe buscou manter a compreensão recíproca através da comunicação verbal e da consulta de informações nos requisitos. Portanto, esses problemas de comunicabilidade permaneceram nos artefatos utilizados pela equipe.

6.3.4 Resultados da Equipe 5

6.3.4.1 Resultado da Comunicação via Artefato da Equipe 5

A comunicação da Equipe 5, da empresa C3, apoiada pelos artefatos é apresentada na Figura 6.25. Nesta categoria, foi possível observar que a equipe utilizava mais o protótipo como apoio para o desenvolvimento de software, em comparação com o uso dos casos de uso: *"Talvez eu tenha colocado na proposta do protótipo informações além do necessário e isso fez com que os casos de uso não fossem consultados"* – PO da Equipe 5. Os protótipos também auxiliaram a comunicação da equipe com o cliente.

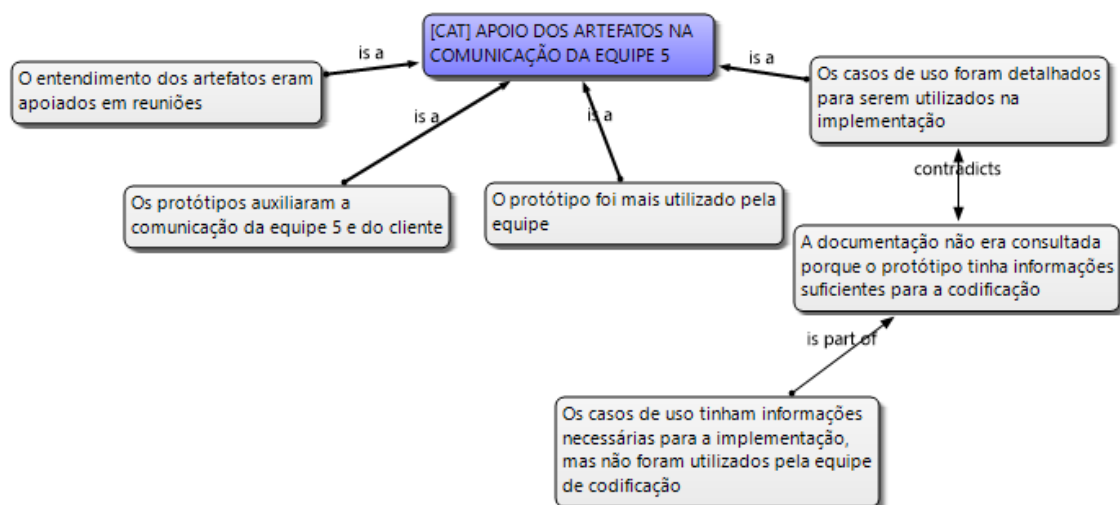


Figura 6.25: Categoria que Representa a Comunicação da Equipe 5 Apoiada por Artefatos.

Em relação às respostas dos participantes em relação aos fatores que impactam a comunicação via artefato, estas foram contabilizadas e apresentadas na Figura 6.26. Foram observadas discordâncias nos fatores ‘Artefato como Meio de Comunicação’ e ‘Consumidores do Artefato’. A partir dos dados coletados nos estudos, foi possível fazer algumas considerações sobre os fatores que impactam a comunicação via artefato.

Em relação ao ‘Artefato como Meio de Comunicação’, o Designer da Equipe 5 afirmou que existem limitações nestes artefatos para representarem todas as informações do sistema: *“Os artefatos contêm as informações necessárias que a equipe precisa para entender o problema. Contudo, existem algumas limitações e informações que não conseguem ser transmitidas nos artefatos. Por exemplo, o ‘mockup descartável’ apresenta apenas uma ideia de como será a interface com os possíveis campos do sistema, mas não apresenta como isso será feito, ou até mesmo a interação do usuário com o sistema”*.

Sobre o fator ‘Consumidores do Artefato’, foi percebido que a maioria dos consumidores não consideram os casos de uso úteis para a implementação, sendo os protótipos considerados como úteis para tal atividade: *“Existia excesso de informação na documentação. Foram colocados tantos detalhes e sem ser de forma direta que acabou gerando várias diferenças na documentação para implementação, em cima detalhes mínimos que não afetavam em si a funcionalidade do sistema... Com o uso do protótipo foi mais fácil entender as necessidades do usuário solicitante, e assim as dúvidas que eu tinha sobre o funcionamento do sistema foram sanadas”* – Desenvolvedor 11 da Equipe

5 e “Com o protótipo metade das funcionalidades do sistema estavam bem definidas, faltando apenas as regras de negócio, que não poderiam ser modeladas visualmente” – Desenvolvedor 12 da Equipe 5.

ARTEFATO COMO MEIO DE COMUNICAÇÃO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
M1. Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.		1				1	
M2. Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.	1		1				
M3. Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.		1				1	
M4. Este artefato me auxilia no que eu gostaria de “falar” para a equipe sobre o domínio do problema.		1				1	
M5. Este artefato permite que o meu projeto para o software seja representado de forma clara para equipe.		1				1	
M6. Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).		1		1			
M7. Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.		1		1			
CONTEÚDO COMUNICATIVO (RESPONDIDO PELOS PRODUTORES DO ARTEFATO)							
Cont1. Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.	1	1					
Cont2. Não foram introduzidas informações desorganizadas no artefato.	1	1					
Cont3. Não foi introduzido o excesso de informações no artefato.	1	1					
Cont4. A quantidade de informações no artefato é suficiente para o entendimento da equipe.	1	1					
Cont5. Acredito que o conteúdo do artefato está claro e é de fácil compreensão.		2					
CONSUMIDORES DO ARTEFATO (RESPONDIDO PELOS CONSUMIDORES DO ARTEFATO)							
PC1. Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.	5	6	1				
PC2. Eu acho fácil compreender este artefato.	4	8					
PC3. Eu irei adotar esse artefato para compreender as atividades de desenvolvimento	4	2		3	2	1	
PC4. Eu compreenderei melhor o software com este artefato.	4	4		1	1	2	
PC5. Eu acho fácil usar este artefato.	6		1	1	3	1	

Figura 6.26: Fatores Relacionados à Comunicação via Artefato da Equipe 5.

A Figura 6.27 apresenta a categoria que representa práticas da equipe que podem impactar na comunicação via artefato. Como aspectos positivos sobre a comunicação via artefato, esta equipe realizava as atualizações dos protótipos e casos de uso após a reunião com o cliente. Porém, ao longo do desenvolvimento deste sistema, os casos de uso não foram atualizados, gerando um impacto negativo na comunicação via artefato: “Os modelos UML eram altrados à medida que as mudanças nos requisitos era realizada” –

Desenvolvedor 4 da Equipe 5 e “Ao longo do desenvolvimento, acredito que os artefatos tenham ficado desatualizados em relação às necessidades dos usuários e implementação do sistema” - Desenvolvedor 4 da Equipe 5.

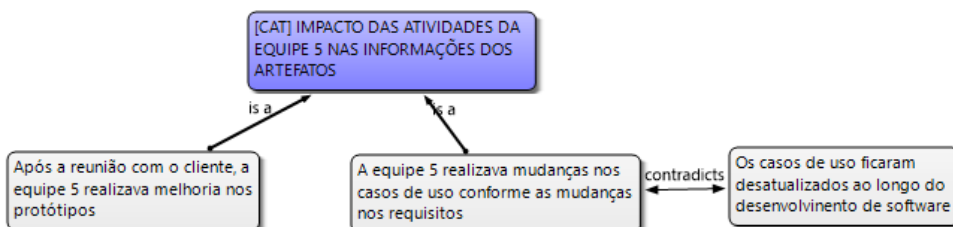


Figura 6.27: Práticas da Equipe 5 que impactam a Comunicação via Artefato.

6.3.4.2 Resultado do Diagnóstico dos Artefatos com o Apoio das DCs

Os resultados do diagnóstico dos protótipos e dos casos de uso da Equipe 5, foram observados problemas de comunicabilidade e caracterizados na Figura 6.28, juntamente com as citações dos participantes, uma vez que as informações do projeto não pode ser representado aqui nesta tese. O diagnóstico foi realizado de maneira semelhante a Equipe 3 e Equipe 4. Nesta análise, foram observados apenas problemas de comunicabilidade nos casos de uso.

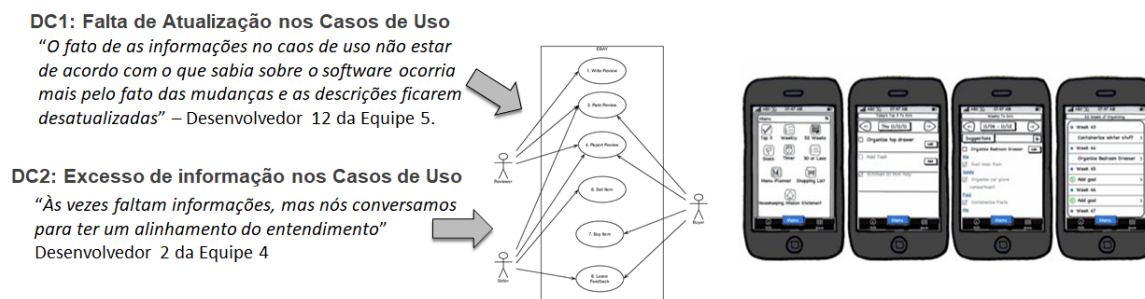


Figura 6.28: Variações do Uso das DCs no Diagnóstico dos Artefatos.

Foram observados problemas de comunicabilidade em relação à falta de atualização de informações e excesso de informação nos casos de uso. Esses problemas causaram interrupções na comunicação da Equipe 5. Sobre essas interrupções causadas na comunicação da Equipe 5, tal equipe buscou manter a compreensão recíproca através da comunicação verbal e da consulta de informações nos requisitos. Portanto, esses problemas de comunicabilidade permaneceram nos artefatos utilizados pela equipe.

6.3.5 Percepções sobre o Uso das DCs no Diagnóstico de Artefatos

Sobre os resultados obtidos com o uso das DCs no diagnóstico dos protótipos, a Figura 6.29 apresenta a categoria uso das DCs para diagnóstico de problemas de comunicabilidade nos artefatos. Os produtores analisaram se a melhoria da informação no artefato poderia ser realizada, indicando um possível problema de comunicabilidade, ou não. Além disso, tal categoria mostra a percepção dos consumidores com base nas DCs.

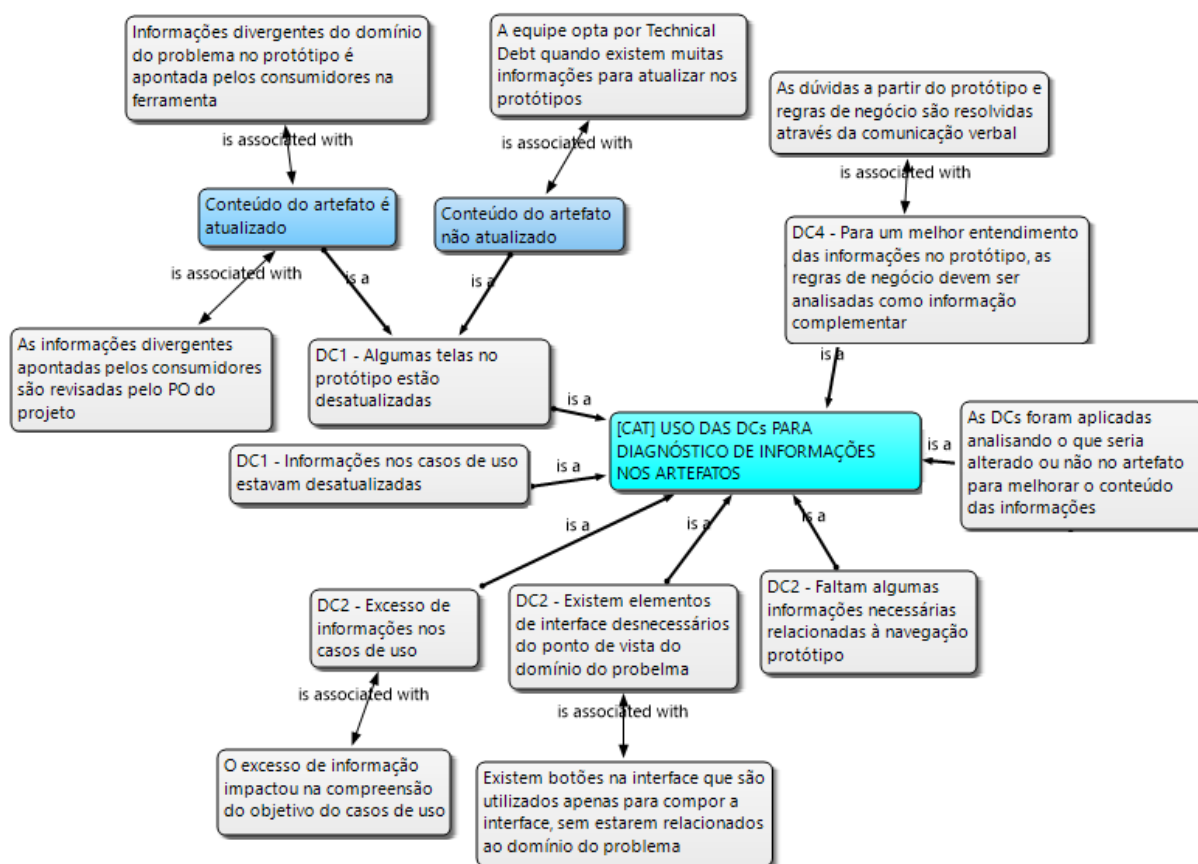


Figura 6.29: Variações do Uso das DCs no Diagnóstico dos Artefatos.

Em relação à DC1, foram identificadas telas que estão desatualizadas devido às seguintes atividades: o conteúdo do artefato é atualizado após informações divergentes serem apontadas pelo *Product Owner* (PO na Figura 6.29) ou pelos demais membros da equipe de desenvolvimento: “Quando nós identificamos que algo não está de acordo com o domínio do problema, nós deixamos uma comentário na ferramenta X (omitido – ferramenta utilizada para disponibilizar os protótipos)” – Desenvolvedor 5 da Equipe 3; o conteúdo do artefato não é atualizado, uma prática de adoção de *technical debt* pela

equipe. Também foram identificadas informações desatualizadas em casos de uso devido às mudanças realizadas nos requisitos ao longo do desenvolvimento de software.

Em relação à DC2, foram identificados elementos de interface desnecessários nas telas e a falta de informações relacionadas à navegação entre as telas: “*Existem algumas informações desnecessárias, como botões que não fazem parte do domínio do problema e que estão ali para a composição da interface*” – Designer da Equipe 4. Também foi identificado o excesso de informação nos casos de uso. Sobre a DC3, não foram identificados problemas de comunicabilidade com esta durante o diagnóstico.

Por fim, foram identificadas informações implícitas nos protótipos com o apoio da DC4, as quais devem ser compreendidas com informação complementar nas regras de negócio. Nesse caso, quando ocorrem dúvidas em relação às regras de negócio, estas são resolvidas através da comunicação verbal: “*As informações do protótipo são complementadas com descrições das regras de negócio e quando surgem dúvidas, basta uma conversa com os envolvidos para resolver isso*” – Designer da Equipe 4.

Em relação às percepções dos produtores no diagnóstico, a Figura 6.30 apresenta a categoria percepção das DCs, a qual possui três subcategorias: facilidade de uso, utilidade e sugestão de melhoria.

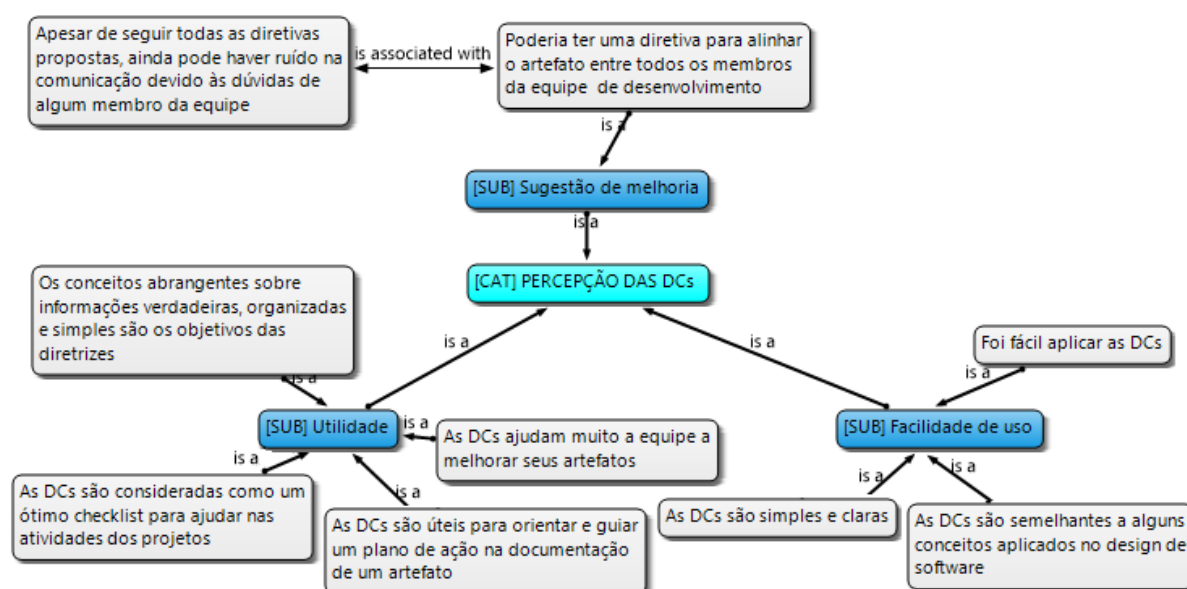


Figura 6.30: Percepções dos Produtores ao Aplicarem as DCs no Diagnóstico dos Artefatos.

Em relação à subcategoria facilidade de uso, as DCs foram consideradas fáceis de aplicar e semelhantes a conceitos utilizados no projeto de artefatos, como mostra a

seguinte citação: “As diretivas são muito parecidas com alguns conceitos que já seguimos no design de software, então foi bem tranquilo entender e ter facilidade de aplicar” - Designer da Equipe 4. Em relação à subcategoria utilidade, os produtores afirmaram que as DCs ajudam a equipe a melhorar os seus artefatos, pois fornecem suporte que orientam a produção de artefatos. As seguintes citações ressaltam isso: “Um designer sempre estará mais preocupado em como os usuários reagirão ao uso de um produto e como este será documentado pela equipe de desenvolvimento. As diretivas apoiam muito no sentido de ajudar a equipe a melhorar seus artefatos” – Designer da Equipe 4; “As diretivas são bem simples e claras, é um ótimo checklist que ajudaria muito na execução das atividades dos projetos” – Designer da Equipe 3.

Por fim, a subcategoria sugestões de melhoria mostra que poderia ter uma diretiva para alinhar o conteúdo do artefato produzido para melhorar ainda mais a comunicação: “Acho que poderia ser acrescentado uma nova diretiva que fale sobre estar alinhado com toda a equipe. Por mais que um designer siga direitinho todas as diretivas, ainda assim pode haver ruído na comunicação. Então acho que seria legal ter uma diretiva que fale em alinhar os artefatos com a equipe” – Designer da Equipe 3.

6.3.6 Discussão dos Resultados

Este estudo foi realizado com o objetivo de compreender os benefícios das DCs em relação ao diagnóstico de problemas de comunicabilidade nos artefatos utilizados por três equipes de desenvolvimento de software. O uso das DCs mostrou os principais problemas de comunicabilidade que impactaram negativamente na comunicação entre produtores e consumidores destes artefatos. Os resultados deste estudo também podem evidenciar o benefício de empregar as DCs na produção de artefatos, pois os problemas de comunicabilidade identificados nos artefatos destas equipes poderiam ser reduzidos.

6.4 Limitações dos Estudos

Em relação às limitações destes estudos, no que se refere aos contextos em que os estudos foram realizados, existe a diversidade limitada das empresas. No entanto, as empresas selecionadas representam típicas organizações de desenvolvimento de software. Sobre o treinamento realizado com as DCs, existe a ameaça sobre a curta duração deste. Para mitigar essa ameaça, a pesquisadora principal ressaltou que os produtores poderiam

entrar em contato caso estes tivessem dúvidas sobre a aplicação das diretivas. Em relação ao emprego das DCs e os resultados obtidos com estas, foram analisados dados a partir das diferentes etapas da condução dos estudos com o objetivo de obter evidências sobre o uso das DCs para apoiar a comunicação a partir de artefatos e caracterizar problemas de comunicabilidade que podem impactar neste tipo de comunicação. Assim, foi possível obter evidências sobre os benefícios das DCs para a comunicação via artefato.

6.5 Considerações sobre o Capítulo

Este capítulo teve como objetivo apresentar o uso das DCs na indústria de software por quatro equipes de desenvolvimento de software em dois estudos. No primeiro estudo, as DCs foram empregadas na produção de artefatos que apoiam a comunicação de equipes durante o desenvolvimento de software. Os resultados mostraram que as DCs apoiaram a mitigação dos problemas que podem causar falhas de comunicação via artefato nas duas equipes de desenvolvimento. Além disso, foi possível entender que as DCs podem ser melhor empregadas na etapa de Externalização dos artefatos, que consiste na representação do sistema que será construído com o objetivo de apoiar o entendimento da equipe (Chaudron *et al.*, 2018).

No segundo estudo, as DCs foram empregadas como diagnóstico de problemas de comunicabilidade que impactaram negativamente na comunicação via artefato em projetos de software concluídos. O uso das DCs mostrou a razão de falhas de comunicação que ocorreram entre produtores e consumidores de artefatos. Os resultados deste estudo também podem evidenciar o benefício de empregar as DCs na produção de artefatos, pois os problemas de comunicabilidade identificados nos artefatos destas equipes poderiam ser reduzidos.

Ainda sobre o uso das DCs por diferentes engenheiros de software, a pesquisadora principal também analisou percepção destes em relação à aceitação das DCs como apoio à melhoria de comunicabilidade. Também foi realizada uma análise das afirmativas empregadas para avaliar os fatores que impactam a comunicação via artefato para verificar se essas são consideradas confiáveis e se estão medindo corretamente os fatores propostos. Tanto a análise da aceitação das DCs e das afirmativas empregadas para avaliar os fatores serão apresentadas no próximo capítulo.

CAPÍTULO 7 – AVALIAÇÃO DA ACEITAÇÃO DAS DIRETIVAS DE COMUNICABILIDADE E ANÁLISE DOS FATORES QUE IMPACTAM A COMUNICAÇÃO

Este capítulo apresenta a avaliação da aceitação das Diretivas de Comunicabilidade no desenvolvimento de artefatos. Tal avaliação mostra a intenção dos participantes, profissionais da indústria, que utilizaram as diretivas nos estudos em adotá-las em futuros projetos. Adicionalmente, este capítulo apresenta a avaliação da confiabilidade das afirmativas utilizadas para a avaliação dos fatores que impactam a comunicação via artefato, os quais são aplicados juntamente com as diretivas.

7.1 Introdução

No cenário de desenvolvimento de software, gerentes e líderes precisam de informações que indiquem a razão de os engenheiros de software adotarem tecnologias no seu ambiente de trabalho (Sultan e Chan, 2000). Portanto, é necessário entender os fatores que podem influenciar na aceitação de uma nova tecnologia (Venkatesh e Bala, 2008). A avaliação aceitação das Diretivas de Comunicabilidade (DCs) torna-se importante para compreender aspectos relevantes da intenção que um produtor de artefatos tem em utilizá-las ou não em seus projetos de software.

Este capítulo apresenta a avaliação da aceitação das DCs por parte dos engenheiros de software, os quais que participaram do segundo estudo experimental para a avaliação das DCs e dos estudos realizados na indústria de software. Adicionalmente, sobre os fatores que devem ser considerados na comunicação via artefato, uma análise exploratória foi realizada para verificar se as afirmativas utilizadas para a avaliação destes fatores são consideradas confiáveis e se estão medindo corretamente os fatores propostos. Assim é possível avaliar a validade das afirmativas propostas.

7.2 Avaliação da Aceitação das Diretivas de Comunicabilidade

O Modelo de Aceitação de Tecnologia (do inglês *Technology Acceptance Model* – TAM) é um dos modelos mais adotados dentre as várias teorias e modelos propostos para

coletar informações sobre a decisão de aceitar ou rejeitar tecnologias (Marangunić e Granić, 2015). O TAM foi projetado para compreender a relação causal entre variáveis externas de aceitação dos usuários e o uso real da tecnologia, buscando entender o comportamento destes usuários através do conhecimento da utilidade e da facilidade de uso percebida por eles (Davis, 1989). Sendo assim, o TAM está fundamentado basicamente em dois construtos: a utilidade e a facilidade de uso percebida, sendo que ambos mensuram completamente os efeitos das variáveis externas, como características do sistema, processo de desenvolvimento, treinamento e etc. Davis (1989) define os dois principais determinantes do modelo TAM da seguinte maneira:

Facilidade de Uso: Define o grau em que um usuário acredita utilizar uma tecnologia específica com pouco esforço.

Utilidade: Define o grau em que um usuário acredita que utilizar uma tecnologia específica melhoraria seu desempenho no trabalho.

A intenção comportamental do usuário de usar uma tecnologia (*intenção de uso*) é determinada pela utilidade percebida e pela facilidade de uso percebida. Os questionários pós-estudo foram elaborados com base nas afirmativas do modelo TAM. As afirmações contidas nos questionários pós-estudo, adaptadas de Venkatesh e Bala (2008), em relação à facilidade de uso, utilidade e intenção de uso das DCs foram:

Facilidade de Uso

- E1.** Minha interação com as Diretivas de Comunicabilidade foi clara e compreensível.
- E2.** Interagir com as Diretivas de Comunicabilidade não exigiu muito do meu esforço mental.
- E3.** Considero fácil utilizar as Diretivas de Comunicabilidade.
- E4.** Eu acho fácil utilizar as Diretivas de Comunicabilidade para melhorar a minha comunicação com a equipe a partir dos artefatos.

Utilidade

- U1.** Usar as Diretivas de Comunicabilidade torna o meu desempenho melhor sobre as informações que precisam ser compreendidas pela equipe a partir dos artefatos.
- U2.** Usar estas diretivas em meu trabalho poderá melhorar a minha produtividade, pois não terei que corrigir informações que não seja compreendidas por colegas (do projeto de desenvolvimento).
- U3.** Usar as Diretivas de Comunicabilidade melhora a efetividade da minha comunicação com a equipe a partir dos artefatos.
- U4.** Eu considero as Diretivas de Comunicabilidade úteis para melhorar a minha comunicação com a equipe a partir dos artefatos.

Intenção de Uso

- I1.** Supondo que eu tenho tempo suficiente para desenvolver artefatos de software, eu utilizaria estas diretivas.
- I2.** Levando em conta que eu tenho domínio para escolher qualquer suporte para o desenvolvimento de artefatos de software, eu prevejo que eu irei usar estas diretivas

I3. Eu pretendo utilizar as diretivas nos próximos meses.

Como respostas para as afirmativas, foi utilizada a escala de Likert *et al.* (1993) com sete categorias sobre o grau de concordância e discordância com opções de resposta desde Concordo Totalmente a Discordo Totalmente. Os participantes, engenheiros de software que participaram do segundo estudo experimental e dos estudos realizados na indústria de software, responderam o seu grau de concordância em relação às afirmações sobre a utilidade e a facilidade de uso das DCs na produção de artefatos. Como o principal objetivo das DCs é apoiar a produção de artefatos, as DCs não foram em relação ao diagnóstico de artefatos. Este questionário está no Apêndice B e as respostas estão no Apêndice E.

A Figura 7.1 apresenta o grau de aceitação dos participantes, em usar as DCs na produção de artefatos, em relação à facilidade de uso (variáveis E1 à E4), utilidade (variáveis U1 à U4) e intenção de uso (variáveis I1 à I3). Observa-se que a maioria dos participantes concordaram com as afirmativas em relação à facilidade de uso, utilidade e intenção de uso, o que indica que as DCs foram consideradas fáceis de utilizar e úteis para a melhoria da comunicabilidade de artefatos.

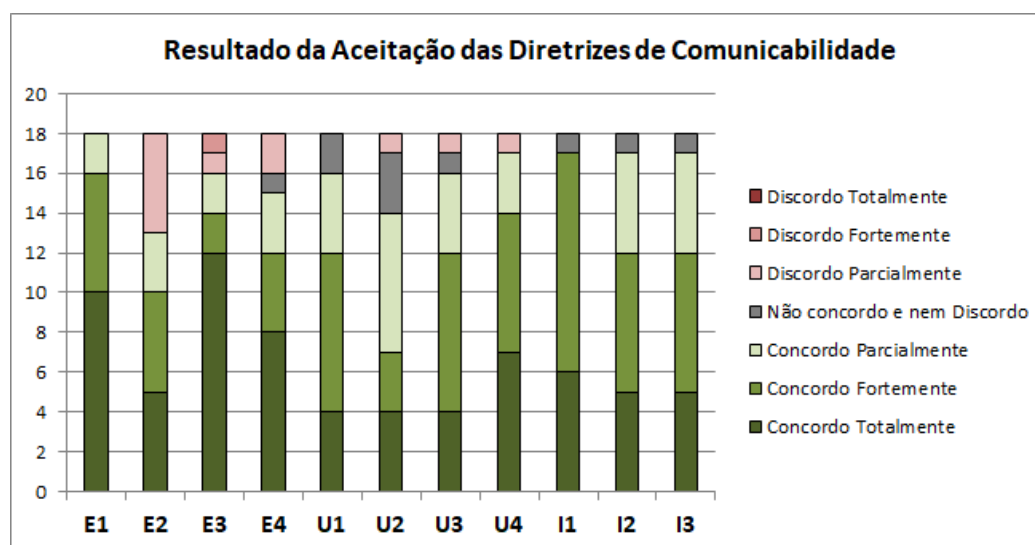


Figura 7.1: Grau de Aceitação dos Participantes com relação ao uso das DCs na produção de artefatos.

Sobre a facilidade de usar as DCs e sua utilidade, foi observado o seguinte relato durante a coleta de dados qualitativos: “As diretivas são bem simples e fáceis de entender. Elas são adequadas com o que é feito pelo nosso time no dia a dia. É uma ótima

ferramenta para lembrar e manter as boas práticas, pois algumas boas práticas vão se perdendo no dia a dia. Tê-las como referência me parece um bom lembrete para deixar os artefatos mais claros e entendíveis” - Analista de Sistemas da Equipe 2 do estudo na indústria.

Sobre a facilidade de uso, foi observado o seguinte relato: *“As diretivas são simples e diretas, o que facilita o entendimento”* - P1 do segundo estudo experimental. Sobre os participantes que discordaram das afirmativas relacionadas à facilidade de uso, foram observados os seguintes relatos: *“Não é fácil de aplicar, acredito que depende da experiência do utilizador”* - P4 do segundo estudo experimental; *“As DCs demandam tempo para o entendimento”* - P6 do segundo estudo experimental.

Os dados quantitativos e qualitativos em relação à aceitação dos engenheiros de software que utilizaram as DCs mostraram que estas podem ser adotadas na indústria de software. Os profissionais da indústria consideraram as DCs úteis e estes possuem a intenção de utilizá-las em seus projetos.

7.3 Avaliação dos Fatores que Impactam na Comunicação Via Artefato

Foi realizada uma análise exploratória para verificar se as afirmativas propostas são consideradas confiáveis e se estão medindo corretamente os fatores propostos na análise das condições de comunicação, a qual é empregada juntamente com o uso das DCs. Para isso, foi empregado o teste Alfa de Cronbach para analisar a consistência das afirmativas. A análise fatorial também foi realizada para verificar se as afirmativas estão medindo corretamente os fatores (Sánchez, 2010).

Essa avaliação é realizada em um conjunto de dados obtidos após a aplicação do questionário para avaliação de um determinado fator, que está disponibilizado no Apêndice E. Assim, as respostas dos engenheiros de software na indústria foram avaliadas para os itens dos questionários associados aos fatores de Artefato como Meio de Comunicação, Conteúdo Comunicativo e Consumidores do Artefato. Esta análise foi realizada com o apoio da ferramenta SPSS v.23⁹.

Em relação à confiabilidade das afirmativas, os resultados mostraram que essas são confiáveis para o Artefato como Meio de Comunicação ($p=0.876$) e Conteúdo

⁹ IBM SPSS Software – www.ibm.com/software/br/analytics/spss/

Comunicativo ($p=0.940$). Em relação à análise fatorial, os resultados são apresentados na Tabela 7.1. Observa-se que todas as afirmativas do fator ‘Meio de Comunicação’ estavam relacionadas ao Componente 1 na ferramenta SPSS, ou seja, tais variáveis estão medindo adequadamente um fator. Além disso, as afirmativas do fator ‘Conteúdo Comunicativo’ estão concentradas no Componente 2 na ferramenta SPSS, indicando também que as demais afirmativas estão mensurando outro fator. Com isso, pode-se interpretar que as variáveis estão mensurando adequadamente os fatores Artefato como Meio de Comunicação e Conteúdo Comunicativo.

	Fator 1 (Artefato como Meio de Comunicação)	Fator 2 (Conteúdo Comunicativo)
M1	0,892	-0,229
M2	0,935	-0,148
M3	0,919	0,258
M4	0,756	-0,337
M5	0,889	-0,320
M6	0,919	0,258
M7	0,875	-0,243
Cont1	-0,179	0,939
Cont2	-0,093	0,951
Cont3	-0,241	0,845
Cont4	0,116	0,910
Cont5	-0,132	0,963

Tabela 7.1: Análise das Variáveis que Mensuram os Fatores ‘Artefato como Meio de Comunicação’ e ‘Conteúdo Comunicativo’.

Em relação à confiabilidade das afirmativas relacionadas ao fator ‘Consumidores do Artefato’, o resultado mostrou que essas são confiáveis ($p=0.700$). Em relação à análise fatorial, os resultados mostram que as afirmativas estão relacionadas aos dois subfatores, como mostra a Tabela 7.2. Ressalta-se que este é o único fator que é mensurado por dois objetivos: capacidade de conhecimento do consumidor sobre o artefato e percepção de aceitação dos consumidores sobre o artefato para apoiá-lo em suas atividades. As variáveis propostas estão relacionadas com os diferentes fatores.

	Consumidores do Artefato	
	Fator 1 (Compreensão)	Fator 2 (Utilidade)
PC1	0,916	0,160
PC2	0,901	-0,038
PC3	0,167	0,915

PC4	-0,254	0,838
PC5	0,083	0,907

Tabela 7.2: Análise do Fator ‘Consumidores do Artefato’.

De acordo com os resultados apresentados na Tabela 7.1 e Tabela 7.2, pode-se concluir que tais afirmativas mensuraram adequadamente os fatores propostos. As respostas destes questionários estão no Apêndice E. Portanto, os questionários para análise dos fatores podem ser considerados como válidos.

Em relação às limitações desta análise, do ponto de vista estatístico, a quantidade de participantes não é considerada ideal, sendo este um problema conhecido em estudos experimentais de ES (Carver et al., 2003). Portanto, há limitação nos resultados.

7.4 Considerações sobre o Capítulo

Este capítulo apresentou a avaliação da aceitação das DCs. Os resultados indicaram que as DCs foram consideradas fáceis de utilizar e úteis para a melhoria da comunicabilidade de artefatos. Este capítulo também apresentou uma análise da validade das afirmativas utilizadas para a avaliação dos fatores. Os resultados mostraram que as afirmativas são confiáveis e estão mensurando adequadamente os fatores propostos para avaliar o impacto destes na comunicação via artefato.

CAPÍTULO 8 – CONTRIBUIÇÕES DAS DIRETIVAS DE COMUNICABILIDADE PARA ASPECTOS RELACIONADOS À COMUNICAÇÃO VIA ARTEFATO

Este capítulo apresenta as contribuições das Diretivas de Comunicabilidade para outros aspectos da comunicação via artefato.

8.1 Introdução

Neste capítulo são relatadas algumas contribuições das Diretivas de Comunicabilidade (DCs) para outros aspectos relacionados à comunicação via artefato. A primeira contribuição refere-se ao uso das DCs para apoiar a compreensão de modelos BPMN (*Business Process Modeling and Notation*) (OMG, 2014) durante a manutenção do software, uma vez que a compreensão de artefatos faz parte da manutenção. Essa contribuição está apresentada na Seção 8.2.

A segunda contribuição refere-se ao uso da base de conhecimento das DCs para a produção de técnicas que auxiliam na inspeção de artefatos de software. Estas técnicas devem ser aplicadas após a produção dos artefatos por profissionais que não sejam responsáveis pela produção do artefato. Tais técnicas são apresentadas na Seção 8.3. Por fim, a Seção 8.4, apresenta o uso das DCs no ensino de modelagem com foco na comunicação via artefato. As DCs foram aplicadas no ensino para uma investigação inicial sobre o fator de aprendizagem de produtores e consumidores ser uma das razões que podem impactar na comunicação via artefato.

8.2 Contribuições das DCs para a Compreensão de Modelos BPMN

Os modelos que apoiam o desenvolvimento e a manutenção de software, o modelo de processos de negócio, apesar de não ser um modelo específico para software, auxilia a compreensão do processo organizacional no qual um software está inserido. Por isso, esses modelos são muito úteis na manutenção de softwares corporativos (*enterprise applications*) (Shishkov *et al.*, 2002; Weitlaner *et al.*, 2013). Tais modelos podem ser desenvolvidos com a notação BPMN (*Business Process Modeling and Notation*), que é a notação padrão para a modelagem gráfica de processos, recomendada pela OMG (Object

Management Group) (OMG, 2014) e utilizada nas mais diversas empresas pelo mundo. Cruz *et al.* (2015) afirmam que os modelos de processos de negócio são de grande relevância durante o processo de desenvolvimento, pois proporcionam entendimento do software que está sendo desenvolvido ou modificado na manutenção.

Neste contexto, o fato de modelos de processos de negócios BPMN estarem desatualizados está relacionado com a falta de conhecimento sobre esta notação por parte dos profissionais que deveriam realizar tal atividade. Com isso, a técnica e-VOL BPMN com o objetivo de facilitar a evolução de modelos BPMN por profissionais com pouco conhecimento sobre a notação (Campos *et al.*, 2019). Com isso, espera-se que o modelo evoluído possua consistência com as novas funcionalidades do software e assim, represente corretamente as atividades apoiadas pelo software.

A e-VOL BPMN possui duas perspectivas para apoiar os profissionais: (i) compreensão do modelo BPMN e (ii) modificação do modelo BPMN. A compreensão relaciona-se com o entendimento do funcionamento do artefato de software e a modificação está relacionada com a alteração do artefato em si. É importante que o artefato seja bem compreendido antes de ser modificado, pois a compreensão é considerada por vários autores como um fator que influencia na manutenção (Lange *et al.*, 2007; Manso *et al.*, 2007; Nugroho, 2009). De fato, um artefato de software deve ser bem compreendido antes de qualquer alteração relacionada à manutenção ser realizada no mesmo (Fernandez-Saez *et al.*, 2013), visto que modificar um software sem primeiramente compreendê-lo, pode ocasionar modificações que geram falhas ou impactam em outros módulos do software, aumentando os custos e o tempo da manutenção.

Na compreensão do modelo com apoio da técnica e-VOL BPMN, esta conduz a pessoa que vai realizar a manutenção do modelo BPMN a compreendê-lo antes de passar para a tarefa de modificação do modelo. Com base nas DCs, foram criadas questões que apoiam a compreensão do modelo durante sua manutenção, que é apresentado abaixo:

- **DC1** - para apoiar a compreensão de informações que não são verdadeiras em relação ao domínio do problema nos modelos, foi criada a questão: *“Todas as informações no modelo são verdadeiras? As informações no modelo não necessitam de atualização?”*.

- **DC2** - para apoiar a compreensão do conteúdo necessário (e não mais que o necessário) nos modelos, foram criadas as questões: *“As informações necessárias do cenário de modelagem estão representadas no modelo?”* e *“Existem informações desnecessárias no modelo?”*.
- **DC3** - para apoiar a compreensão de informações que não são relevantes para os modelos, foi criada a questão: *“As atividades estão representadas de maneira lógica e coerente? Você identifica com facilidade qual é o próximo passo que deve ser realizado em cada parte do diagrama?”*.
- **DC4** - para apoiar a compreensão de informações que precisam ser esclarecidas no modelo, foi criada a questão: *“As atividades estão representadas de forma clara? Você identifica claramente no diagrama quem são os atores responsáveis pelas atividades? As atividades estão nomeadas de maneira clara? Você identifica claramente o início e o fim do processo?”*

Essas questões resultaram em um questionário de avaliação da compreensão, que também sugere correções que podem ser feitas no modelo. Esse questionário foi incluído como parte inicial da e-VOL BPMN (Campos *et al.*, 2019). A Figura 8.4 mostra o questionário de compreensão que foi incluído como passo inicial da e-VOL BPMN. Quando todas as respostas do questionário forem verdes, supõe-se que o artefato já está compreendido e corrigido pela pessoa que deve realizar as modificações no modelo. Com isso, ela pode seguir para a próxima fase que é a modificação do modelo propriamente dita.

Em relação à modificação, foram criados cenários de modificação para os elementos da notação BPMN. Para a criação dos cenários de modificação, foram investigados elementos da notação BPMN em contextos de aplicação real. Assim a e-VOL apresenta:

Cenário de mudança: representa o cenário de modificação identificado no domínio do problema, no processo real.

Modificação: representa a instrução para realizar as modificações no diagrama.

Como era: representa um exemplo de cenário antigo, que pode estar representado no diagrama atual (caso já exista um diagrama inicial).

Como passa a ser: um exemplo de como o diagrama pode ficar após a execução das instruções, isto é, o diagrama evoluído.

PROCEDIMENTOS PARA A COMPREENSÃO DOS MODELOS BPMN	
1. Todas as informações no modelo são verdadeiras? As informações no modelo não necessitam de atualização?	
<input type="checkbox"/> Sim	<input type="checkbox"/> Não. Neste caso, adicione informações verdadeiras!
2. As informações necessárias do cenário de modelagem estão representadas no modelo?	
<input type="checkbox"/> Yes	<input type="checkbox"/> Não. Neste caso, adicione as informações que estão faltando.
3. Existem informações desnecessárias no modelo?	
<input type="checkbox"/> Não	<input type="checkbox"/> Sim. Remova as informações desnecessárias e torne-o objetivo.
4. As atividades estão representadas de maneira lógica e coerente? Você identifica com facilidade qual é o próximo passo que deve ser realizado em cada parte do diagrama?	
<input type="checkbox"/> Sim	<input type="checkbox"/> Não. Organize as informações de maneira lógica para que o diagrama apresente informações com coerência.
5. As atividades estão representadas de forma clara? Você identifica claramente no diagrama quem são os atores responsáveis pelas atividades? As atividades estão nomeadas de maneira clara? Você identifica claramente o início e o fim do processo?	
<input type="checkbox"/> Sim	<input type="checkbox"/> Não. Faça as correções necessárias para deixar o diagrama com informações claras.
Todas as correções foram realizadas? As respostas acima são todas VERDES? Então, vamos evoluir o diagrama!	

Figura 8.1: Questionário de Compreensão da e-VOL BPMN.

A Figura 8.2 mostra parte dos cenários de modificação para diferentes elementos da notação BPMN. Na Figura 5, analisando o cenário “C12”, que trata do evento de início do tipo Timer, o cenário de mudança questiona se a modificação que ocorreu no mundo real é referente a um processo que sempre inicia em um determinado dia do mês. Se sim, então o usuário deve realizar a seguinte modificação: “Troque o evento inicial padrão por um evento inicial temporal”. Então, na coluna “Como era” é apresentado um exemplo onde um processo de pagamento inicia, mas não é especificado quando ele inicia. Então,

a coluna “Como passa a ser” apresenta a modificação que o usuário deve fazer para que o processo passe a especificar quando este inicia. No caso do exemplo, no 5º dia útil de cada mês.

Cenário de Mudança	Modificação	Como era	Como passa a ser
C12. Se um processo inicia sempre em um determinado dia do mês.	- Troque o evento inicial padrão por um evento inicial temporal		
C18. Se uma atividade só inicia se uma determinada condição for satisfeita	- Inclua um evento condicional para representar a condição antes da atividade.		

Figura 8.2: Exemplos de Cenários de Modificação.

Desta forma, a e-VOL BPMN demonstra não somente o contexto de utilização do elemento, mas também a forma que o elemento pode ser utilizado em um diagrama através de exemplos visuais, o que colabora para o aprendizado da notação BPMN. No total, a e-VOL BPMN possui 23 cenários de modificação que abrangem os elementos mais utilizados da notação BPMN, dentre elementos básicos e avançados.

8.2.1 Estudo Experimental com a técnica e-VOL BPMN

Campos *et al.* (2019) apresentam um experimento realizado com a e-VOL BPMN com o objetivo de avaliar seu apoio na manutenção de modelos BPMN. Neste estudo, a e-VOL BPMN foi comparada com o Pôster BPMN 2.0. O Pôster BPMN foi escolhido devido à sua popularidade e facilidade de acesso. Além do mais, em uma entrevista com dois profissionais da indústria, estes relataram que o Pôster BPMN 2.0 era o material sobre BPMN que a empresa tinha disponível.

Foram selecionados 55 estudantes para o experimento. Alguns desses participantes eram estudantes de graduação cursando a disciplina “Análise e Projeto de Sistemas” (APS) do Curso de Ciência da Computação e outros participantes eram alunos de graduação cursando a disciplina “Processo de Desenvolvimento de Software (PSD)” do Curso de Engenharia de Software, ambos da UFAM. Antes da execução do experimento, ambos os participantes receberam o mesmo treinamento sobre BPMN, com exercícios

práticos. Os participantes foram divididos em dois em dois grupos, um grupo que utilizaria a e-VOL BPMN e outro grupo que utilizaria o Pôster BPMN.

Cada participante deveria receber o diagrama inicial, a descrição de evolução do processo e a e-VOL BPMN ou o Pôster BPMN, dependendo do grupo em que o participante estava inserido. O diagrama BPMN era sobre um processo de produção de Mix de Cereais.

8.2.2 Resultados do Estudo

Os modelos produzidos neste experimento foram avaliados de acordo com a completude e corretude. A completude define o quanto um artefato apresenta as informações necessárias de acordo com o seu contexto (Marques *et al.*, 2017). A corretude define quanto o artefato emprega os elementos e relacionamentos de acordo com a sintaxe da notação e descreve corretamente o domínio da aplicação de acordo com as informações disponíveis (Marques *et al.*, 2017). Para avaliar a completude dos diagramas, foram avaliadas as funcionalidades que deveriam estar representadas no diagrama BPMN. Essas funcionalidades foram classificadas como:

Básicas – funcionalidades que deveriam ser representadas com elementos básicos da notação BPMN. Essas funcionalidades receberam peso 1, normal.

Complexas – funcionalidades que deveriam ser representadas com elementos avançados da notação BPMN. Essas funcionalidades receberam peso 2.

Quando uma funcionalidade não estava representada no diagrama, era atribuído um defeito de completude à funcionalidade relacionada. Quando uma funcionalidade estava representada, mas estava representada fora de ordem, incompleta ou com erro de sintaxe da notação era atribuído um defeito de corretude à funcionalidade relacionada. A Figura 8.3 apresenta o gráfico boxplot para a corretude e completude dos diagramas.

Para analisar estatisticamente a corretude e completude dos diagramas, foi aplicado o teste Shapiro Wilk (Shapiro, 1965) para verificar se a distribuição dos valores da amostra seguia uma distribuição normal. Com relação à completude, o resultado do teste Shapiro Wilk mostrou que os valores de completude dos diagramas não seguiam uma distribuição normal (*p-value* para o Grupo que utilizou a e-VOL = 0,001, *p-value* para o Grupo que utilizou o Pôster = 0,000). Assim, foi utilizado o teste de Mann Whitney para analisar a completude dos diagramas dos dois grupos. O resultado do teste de Mann

Whitney mostrou que não houve diferença significativa entre os dois grupos para a completude ($p = 0,952$). Com relação à corretude, novamente, o teste Shapiro Wilk foi utilizado para verificar se a distribuição dos valores da amostra segue uma distribuição normal. O resultado do teste Shapiro Wilk mostrou que os valores de corretude dos diagramas não seguiam uma distribuição normal (p -value para o Grupo que utilizou a e-VOL = 0.000, p -value para o Grupo que utilizou o Pôster = 0,152). Assim, foi utilizado o teste de Mann Whitney para analisar a corretude dos diagramas dos dois grupos. O resultado do teste de Mann Whitney mostrou que houve diferença significativa entre os dois grupos para a corretude ($p = 0,009$).

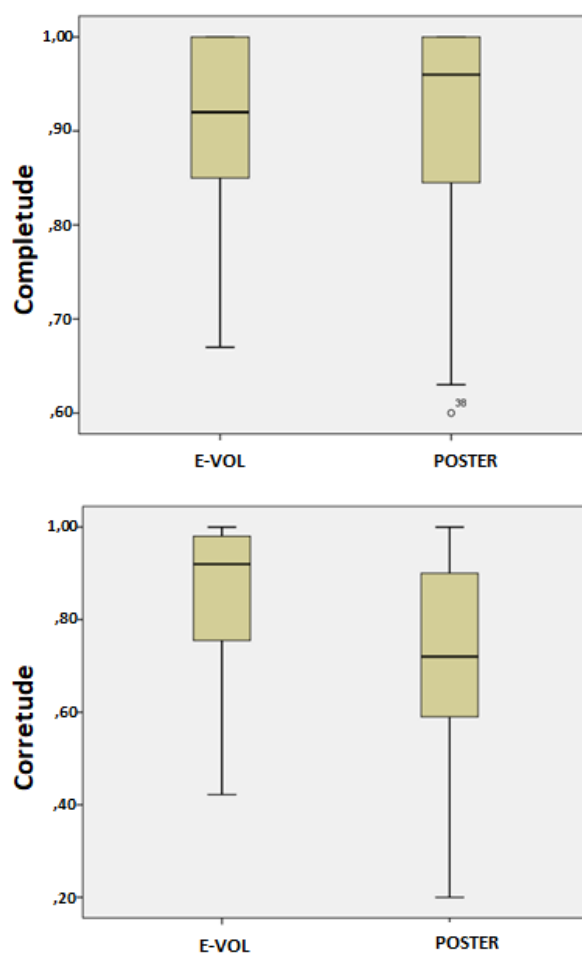


Figura 8.3: Gráfico Boxplot para Completude e Corretude dos modelos BPMN.

Observa-se na Figura 8.3, em relação à completude, que a mediana do grupo que usou o e-VOL BPMN estava abaixo da mediana do grupo que usou o pôster. No entanto, não foi observada diferença significativa entre os grupos. A mediana da corretude dos

diagramas do grupo que usou o e-VOL BPMN estava acima da mediana do grupo que usou o Pôster, o que mostra, junto com o resultado do teste estatístico que apresentou diferença estatística, que o grupo que usou o e-VOL apresentou modelos mais corretos que o grupo que usou o Pôster BPMN 2.0.

A análise dos dados mostrou que os modelos do grupo que utilizou a e-VOL BPMN corretos do que os modelos do grupo que utilizou o Poster BPMN na manutenção destes. Portanto, a e-VOL BPMN pode apoiar profissionais que possuam pouco conhecimento em BPMN a manterem tais modelos atualizados, o que conseqüentemente, irá diminuir o tempo para compreensão do software e custos no tempo de manutenção do software. Outros resultados podem ser consultados em (Campos *et al.*, 2019).

8.2.3 Limitações do Estudo

Em relação às limitações deste estudo, pode-se destacar o efeito de aprendizagem que os participantes experimentaram entre as sessões. Isso foi atenuado perguntando aos participantes se os mesmos já haviam tido algum tipo de experiência com a utilização da notação BPMN. Os participantes alegaram que não tiveram conhecimento prévio da notação. Antes da execução do experimento, todos os alunos participaram de um mesmo treinamento sobre BPMN, em duas aulas.

Os participantes foram estudantes de graduação, para equipará-los à profissionais de empresas e o estudo ser conduzido em um ambiente acadêmico. Em relação à questão 1, sob certas condições, não há grande diferença entre este tipo de alunos e profissionais (Carver *et al.*, 2003) e eles poderiam, portanto, ser considerados como a próxima geração de profissionais. Além disso, segundo Carver et al. (Carver *et al.*, 2003), os estudantes que não têm experiência na indústria podem ter habilidades semelhantes aos profissionais menos experientes. Em relação à questão 2, a descrição do cenário utilizado era referente a um processo real e também, o estudo procurou apresentar condições similares a manutenções / evoluções que são realizadas na indústria. Neste cenário, normalmente, existe um sistema legado que é utilizado e com o tempo o sistema precisa se adequar às mudanças ocorridas no processo, dessa forma, é necessário realizar a evolução do software e conseqüentemente, dos seus diagramas de projeto.

Para avaliar o experimento com relação à evolução dos diagramas, foram utilizadas as métricas de corretude e completude que são métricas utilizadas para avaliação de diagramas (Marques *et al.*, 2017). Em relação à homogeneidade da amostra, os participantes são todos estudantes da mesma instituição. Devido a esse fato, há uma limitação nos resultados, que devem ser considerados indicadores e não conclusivos.

8.3 Contribuições das DCs para o Desenvolvimento da Família de Técnicas ComD2

O propósito da família de técnicas ComD2 é apoiar a identificação de defeitos considerados como riscos de falhas de comunicação, o que pode melhorar a comunicabilidade de artefatos de software (Lopes *et al.*, 2018b). Inicialmente, foram desenvolvidas três técnicas específicas para inspeção de diagramas de classes, atividades e máquina de estados da UML. As técnicas foram desenvolvidas para estes modelos por serem uns dos mais utilizados na indústria (Reggio *et al.*, 2014).

Foram criados itens de verificação para apoiar a identificação de defeitos considerados como riscos de falhas de comunicação com base nas quatro DCs. A seguir é apresentada a maneira como os itens foram criados, utilizando um exemplo para o diagrama de classes:

- **DC1** - para a identificação de informações que não são verdadeiras em relação ao domínio do problema nos modelos, como: *“As classes possuem conteúdo que afeta a qualidade do modelo?”*.
- **DC2** - para o conteúdo necessário (e não mais que o necessário) nos modelos, como: *“Todas as classes necessárias do domínio do problema estão no diagrama?”*.
- **DC3** - para a identificação de informações que não são relevantes para os modelos, por exemplo: *“As classes são relevantes para a modelagem do sistema?”*.
- **DC4** - para a identificação de informações que não estão claras no modelo, como: *“Há classes e relacionamentos com descrições que não são claras?”*.

Como apresentado em um dos estudos do Capítulo 4 (Subseção 4.4), foi observado que quando existem problemas de comunicabilidades, eles podem causar defeitos no software, ou seja, defeitos que prejudicam a qualidade dos artefatos de software. Isso ocorre devido à falta de compreensão dos consumidores sobre a intenção dos produtores em

relação ao modelo. Assim, para cada item de verificação, foi usada a classificação de defeitos apresentada por Granda *et al.* (2015).

A partir dos itens de verificação, é possível classificar os defeitos. Por exemplo, para o item de verificação que é baseado na máxima de Quantidade, relacionamos este item da seguinte forma: *“Todas as classes necessárias do domínio do problema estão no diagrama? Caso contrário, isso pode ser uma discrepância de Omissão”*.

Além disso, cada item de verificação analisa as informações expressas em modelos, que podem estar nas seguintes dimensões de representação das informações: Sintático (relação do modelo com a linguagem de modelagem), Semântico (relação do modelo com o domínio do problema) e Pragmático (relação do modelo com os stakeholders) (Kamthan, 2009). Essas dimensões também podem ajudar na compreensão de informações que causam as falhas de comunicação a partir dos modelos. Portanto, podem existir riscos relacionados à forma de representação estão associados à dimensão Sintática, além de riscos relacionados ao conteúdo da informação estão associados às dimensões Semântica e Pragmática. As diferentes dimensões de representação relacionadas às informações nos modelos foram adicionadas aos itens de verificação. O item de verificação a seguir mostra a dimensão da representação afetada por um defeito considerado como risco de comunicação, o qual se refere ao conteúdo das informações do sistema na modelagem: *“Todas as classes necessárias do domínio do problema estão no diagrama? Caso contrário, isso pode ser uma discrepância de Omissão (Semântica)”*.

Em relação aos itens de verificação, estes podem indicar mais de um defeito de software. Nas técnicas da família ComD2, existem itens de verificação para todos os elementos de um modelo, como o seguinte item para o diagrama de classes: *“Há elementos com descrições que não são claras? Em caso afirmativo, provavelmente há discrepância de Ambiguidade (Pragmática)”*. Também existem itens de verificação para elementos exclusivos nos modelos, como o seguinte item para o elemento Associação no diagrama de classes: *“De acordo com o domínio do problema, todos os relacionamentos de associação são estabelecidos entre as classes? Caso contrário, isso pode ser uma discrepância de Omissão (Semântica e Pragmática)”*.

A Figura 8.4 apresenta alguns itens de verificação para diagramas de classes, mostrando a estrutura das técnicas da família ComD2. No relatório técnico (Lopes *et al.*,

2018c) pode ser visualizado os detalhes para a criação de outras técnicas para outros modelos com base na família ComD2, além das técnicas criadas.

Técnica ComD2 para Diagrama de Classe	
As classes possuem conteúdo que afeta a qualidade do modelo?	
Classes	Todas as classes necessárias do domínio do problema estão no diagrama? Caso contrário, isso pode ser uma discrepância de Omissão (Semântica).
Todas as classes necessárias do domínio do problema estão no diagrama?	
Classes	As classes possuem conteúdo que afeta a qualidade do modelo? Em caso afirmativo, provavelmente existem discrepâncias de Inconsistência e/ou Informação Estranha (Semântica).
As classes são relevantes para a modelagem do sistema?	
Classes	As classes são relevantes para modelagem de sistema? Caso contrário, provavelmente existem discrepâncias de Redundância e/ou Informação Estranha (Semântica).
Há classes e relacionamentos com descrições que não são claras?	
Classes	Existem classes e relacionamentos com descrições que não são claras? Em caso afirmativo, provavelmente há discrepância de Ambiguidade (Pragmática).

Figura 8.4: Parte da técnica ComD2 para diagramas de classes.

8.3.1 Estudo Experimental com as técnicas ComD2

Com o objetivo de avaliar as técnicas iniciais da família ComD2, foi realizado um estudo *in vitro*. Neste estudo, foram analisadas as medidas de eficácia (razão entre o número de defeitos detectados e o número total de defeitos existentes) e eficiência (razão entre o número de defeitos por tempo de inspeção) de cada participante para as diferentes técnicas. As medidas de eficiência e eficácia adotadas são frequentemente usadas em estudos que investigam técnicas de inspeção (Fernandez *et al.*, 2012; Valentim *et al.*, 2015). Também foi analisada a percepção dos participantes.

Foram selecionados 30 participantes para o estudo. Os participantes possuíam conhecimento sobre modelagem de software com diagrama de classes, atividades e máquina de estados. Foram selecionados diagramas UML de um projeto de desenvolvimento web real. Além disso, todos os artefatos necessários foram preparados, como formulários para os participantes reportarem as discrepâncias encontradas e questionários pós-estudo.

Na etapa de execução do estudo, primeiro foram ministrados treinamentos para todos os participantes sobre as técnicas ComD2. Em seguida os participantes realizaram a inspeção dos diagramas de maneira individual. Após a inspeção, foram entregues os

questionários pós-estudo para os participantes responderem. Duas pesquisadoras participaram da execução deste estudo.

8.3.2 Resultados do Estudo

Após a execução do estudo, foi analisado se as técnicas alcançavam o objetivo de detectar defeitos. O oráculo de defeitos continha um total de 25 defeitos, incluindo os defeitos nos três diagramas (sendo 10 defeitos no diagrama de classes, 7 defeitos no diagrama de máquina de estados e 8 defeitos no diagrama de atividades).

A Tabela 8.1 apresenta os participantes (coluna P#), o número de defeitos encontrados por cada participante (colunas CD para o diagrama de classes, AD para o diagrama de atividades e SD para diagrama de máquina de estados), com o tempo de inspeção (tempo (horas)) e a eficácia dos participantes com as técnicas (colunas EfC para a técnica de diagrama de classes, EfA para a técnica de diagramas de atividades e a EfS para diagrama de máquina de estados). Como os participantes realizaram a inspeção dos modelos desenvolvidos em um mesmo momento, sendo os diagramas de classes, atividades e máquina de estados, a eficiência individual de cada participante com as técnicas foi avaliada (colunas EfT).

Analisando o indicador de eficácia, foi observado que os inspetores conseguiram identificar uma média de 40% dos defeitos com a técnica para diagrama de classes, 35,83% com a técnica para diagrama de atividades e 25,83% com a técnica para diagrama de máquina de estados. Este é um bom resultado em termos de eficácia de identificação de defeitos, comparados aos indicadores alcançados por outras técnicas de inspeção para modelos (Valentim *et al.*, 2015). Os resultados mostraram indícios de que as técnicas ComD2 alcançam o propósito de apoiar a detecção de defeitos. Em relação à eficiência, os participantes encontraram uma média de 10,94 defeitos por hora com o conjunto de técnicas. Entretanto, como o número de defeitos é diretamente dependente dos modelos inspecionados, não é indicado realizar uma comparação entre os resultados de eficiência deste estudo de viabilidade.

P#	CD	AD	SD	Tempo (horas)	EfC (%)	EfA (%)	EfS (%)	EfT (%)
P1	1	3	2	1,33	10	37,5	25	24,1
P2	6	4	3	1,21	60	50	37,5	49,1
P3	4	2	1	0,95	40	25	12,5	25,8
P4	5	3	1	0,81	50	37,5	12,5	33,3

P5	3	2	1	1,26	30	25	12,5	22,5
P6	4	3	4	1	40	37,5	50	42,5
P7	4	4	3	1,16	40	50	37,5	42,5
P8	8	2	3	0,63	80	25	37,5	47,5
P9	7	3	3	1,48	70	37,5	37,5	48,3
P10	7	1	1	0,83	70	12,5	12,5	31,6
P11	5	2	1	0,81	50	25	12,5	29,1
P12	9	2	1	0,81	90	25	12,5	42,5
P13	6	4	1	1,3	60	50	12,5	40,8
P14	9	3	3	0,95	90	37,5	37,5	55
P15	8	4	1	1	80	50	12,5	47,5
P16	4	4	2	1,01	40	50	25	38,3
P17	3	4	3	1,13	30	50	37,5	39,1
P18	4	4	4	0,96	40	50	50	46,6
P19	8	0	3	0,26	80	0	37,5	39,1
P20	5	2	1	1,16	50	25	12,5	29,1
P21	6	2	3	1,33	60	25	37,5	40,8
P22	2	3	1	1,06	20	37,5	12,5	23,3
P23	8	2	3	1,05	80	25	37,5	47,5
P24	4	1	2	1,06	40	12,5	25	25,8
P25	5	7	3	1,01	50	87,5	37,5	58,3
P26	4	3	2	1,05	40	37,5	25	34,1
P27	3	4	1	1,06	30	50	12,5	30,8
P28	7	2	2	1,21	70	25	25	40
P29	5	3	1	1,21	50	37,5	12,5	33,3
P30	5	3	2	1,21	50	37,5	25	37,5
Média de Eficácia e Eficiência					53	35,8	25,8	10,94

Tabela 8.1: Resultados por participante com a família ComD2.

Para a compreensão das percepções dos participantes, foram analisados os questionários pós-estudo, os quais tinham três questões abertas. A primeira questão era: “Q. Qual a sua percepção com o uso das técnicas?”. Foram relatadas as seguintes respostas dos participantes que tiveram mais de 40% de eficácia na detecção de defeitos com a família ComD2 (considerando a eficácia das três técnicas):

“As técnicas são bem práticas. Os níveis de representação facilitam a revisão e ajudam a entender a intenção do autor do artefato. Com esta classificação, também é possível corrigir os defeitos de maneira mais fácil” (P12).

“Com o uso das técnicas é mais fácil classificar os defeitos, permitindo resolvê-los e comunicá-los mais facilmente. Através das técnicas é possível entender as diferentes correções que devem ser feitas, como uma correção simples ou complexa nos modelos” (P13)

As citações dos participantes mostram que os defeitos considerados como riscos de falhas de comunicação, que estão nas dimensões de representação Semântica e Pragmática, são os tipos de defeitos que mais podem prejudicar na compreensão de um modelo, pois os defeitos no nível Sintático podem não comprometer tanto a compreensão do consumidor que conhece a linguagem de modelagem.

Para compreender se os participantes tiveram dificuldades com a família de técnicas ComD2, o questionário pós-estudo tinha a terceira questão: “*Q. Quais as dificuldades com o uso das técnicas?*”. A seguir são relatadas algumas citações.

“Apesar das técnicas ajudarem na classificação das dimensões de representação, tive dúvidas com a classificação de Semântico e Pragmático” (P6)

“Certos defeitos se enquadram em mais de um nível, logo é preciso avaliar e interpretar cada caso para não haver equívocos” (P29)

“Existem tópicos semelhantes para auxiliar a inspeção, com isso fica difícil escolher os tópicos que se adequam na inspeção dos modelos” (P30)

Em relação às citações dos participantes, percebeu-se que existem dificuldades com entendimento das dimensões de representação das informações e os defeitos relacionados. Apesar das técnicas auxiliarem na classificação da dimensão de representação associado aos defeitos, podem ser realizadas melhorias nas técnicas em relação aos exemplos das diferentes dimensões de representação. Isso mostra que é necessário adicionar na proposta inicial da família ComD2, descrições das dimensões de representação para facilitar o seu uso por profissionais.

Sobre as anotações das pesquisadoras durante a execução do estudo, foi percebido que alguns participantes consideraram que as informações desnecessárias, irrelevantes, ambíguas e informações falsas afetam a comunicação de equipes a partir de modelos. Esses tipos de informações violam as quatro máximas de Grice e indicam que esta teoria é adequada para análise da comunicação entre produtores e consumidores.

Os resultados obtidos com as técnicas mostraram que os defeitos considerados como riscos de falhas de comunicação nas dimensões Semântica e Pragmática podem afetar em tal comunicação, pois afetam diretamente no domínio do problema e na relação da compreensão do modelo com os stakeholders. Os defeitos na dimensão Sintática nem

sempre afetam a compreensão dos consumidores, pois estão relacionados à sintaxe da linguagem utilizada para a modelagem.

Defeitos na dimensão Pragmática podem não ser percebidos pelos consumidores devido à falta de contexto informacional. Nesse caso, falhas de comunicação podem ocorrer porque os consumidores podem não entender a intenção dos produtores.

Portanto, os resultados do estudo forneceram indícios de que a família de técnicas ComD2 pode apoiar a inspeção de riscos de falhas de comunicação, os quais podem ser corrigidos por seus produtores e assim melhorar a comunicação de equipes de desenvolvimento que utilizam modelos como apoio à comunicação.

8.3.3 Limitações do Estudo

No estudo de viabilidade da família ComD2, há limitações, como o fato dos participantes serem alunos de graduação e o estudo ser realizado em ambiente acadêmico. Sobre essa limitação, Fernandez *et al.* (2010) afirmam que alunos de graduação que não têm experiência na indústria podem ter habilidades semelhantes a profissionais menos experientes e um dos objetivos das técnicas é auxiliar profissionais sem experiência no processo de inspeção de modelos. Outra limitação é que os modelos inspecionados faziam parte de um projeto de desenvolvimento, assim, não é possível afirmar que esses modelos representam todos os tipos de diagramas de classes, diagramas de atividades e diagramas de máquina de estados. Portanto, são necessários mais estudos em outros contextos com essa família de técnicas. Quanto aos indicadores de eficácia e eficiência adotados, eles são frequentemente utilizados em estudos que investigam técnicas de inspeção (Valentim *et al.*, 2015).

8.4 Contribuições para o Ensino de Modelagem com foco na Comunicação

Em relação às falhas de comunicação por meio de artefatos de software, este pode estar relacionado ao tipo de treinamento recebido pelos engenheiros de software.

Geralmente, os professores enfatizam a sintaxe e a representação do domínio do problema em sua produção. No entanto, os alunos também devem aprender sobre quais conteúdos nos são adequados para apoiar a comunicação.

Dado esse contexto, foi realizado um estudo com o objetivo de compreender o apoio das DCs no ensino de modelagem com foco na comunicação. Este estudo foi publicado em Lopes *et al.* (2019b). Neste estudo foi analisado o uso das DCs para a produção de diagramas de classes de maneira individual. Foram selecionados 36 alunos de graduação do curso de Ciência da Computação da UFAM. Estes alunos haviam cursado a disciplina de Engenharia de Software e estavam cursando a disciplina Análise e Projeto de Sistemas. Ressalta-se que estes alunos não participaram dos estudos realizados para avaliar as DCs. Para o uso das DCs, um treinamento com as DCs (como um *Learning Object*) e DCs impressas como suporte adicional foram elaborados para serem utilizados no contexto de ensino. Todos os participantes receberam treinamento com o *Learning Object*, mas apenas alguns participantes optaram por utilizar o suporte adicional. Devido a esta escolha, dois grupos foram formados: Grupo 1 (que obteve o treinamento e o uso das DCs de forma impressa como suporte adicional) e o Grupo 2 (que obteve apenas o treinamento sobre o apoio das DCs com foco na melhoria da comunicação entre produtores e consumidores).

Foram estabelecidas hipóteses nulas e alternativas para analisar a diferença entre o número de defeitos considerados como riscos de comunicação, sendo:

H_{0B} : Não há diferença estatística entre o Grupo 1 e o Grupo 2 em relação ao número de riscos.

H_{1B} : Há diferença estatística entre o Grupo 1 e o Grupo 2 em relação ao número de riscos.

8.4.1 Resultados do Estudo

Foram analisados os modelos produzidos pelo Grupo 1 e o Grupo 2. A avaliação do diagrama de classes resultou em um total de 49 riscos para o Grupo 1 e 68 riscos para o Grupo 2. A Figura 8.5 mostra o gráfico de boxplot com a distribuição dos riscos observados nos diagramas de classes em relação aos grupos. Observa-se que a mediana do Grupo 2 está abaixo do Grupo 1, e houve uma maior dispersão do número de defeitos

cometidos pelo Grupo 2. Devido à dispersão dos boxplots mostrados na Figura 8.5, o uso das DCs impressas resulta em defeitos mais homogêneo.

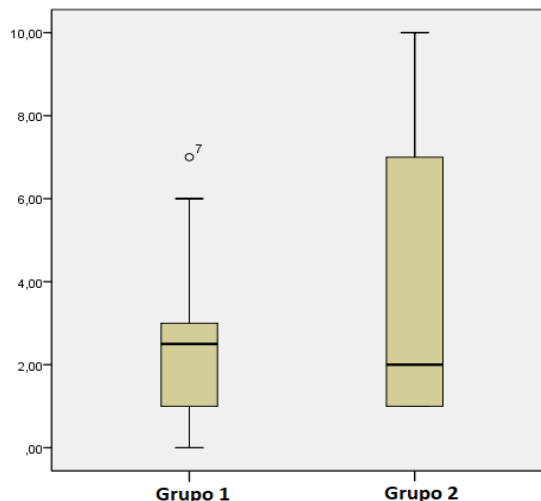


Figura 8.5: Número de Defeitos no Diagrama de Classes.

Para analisar o número de riscos cometidos pelos dos grupos, análises estatísticas foram realizadas usando a ferramenta SPSS, onde foi realizado o teste de normalidade de Shapiro-Wilk com $\alpha=0,05$. O teste de normalidade mostrou que a distribuição não é normal (com $p=0,107$ para o Grupo 1 e $p=0,000$ para o Grupo 2). Devido a esses resultados (McCrum-Gardner, 2008), o teste não paramétrico de Mann-Whitney foi utilizado. O resultado do teste de Mann-Whitney mostrou que não houve diferença significativa no número de riscos cometidos pelos grupos ($p=0,743$), suportando a hipótese H_{0B} .

Ainda sobre gráfico de boxplot apresentado na Figura 8.5, ambos os grupos têm baixas medianas de riscos cometidos na modelagem. O Grupo 2 tem a mediana mais baixa, com cinco participantes acima da mediana. Por outro lado, o Grupo 1 teve uma mediana de três riscos por diagrama, com apenas dois participantes acima da mediana. Portanto, as DCs podem contribuir para a modelagem de um diagrama de classes com uma média baixa de riscos por diagrama. A

Tabela 8.2 apresenta a descrição destes riscos com a classificação de possíveis falhas de comunicação (coluna SD), os tipos de defeitos que podem ocorrer caso estes riscos se concretizem e o número de riscos cometidos pelos participantes de cada grupo (G1 – Grupo 1 e G2 – Grupo 2).

SD	Tipo de defeito (Qualidade)	Descrição do defeito	G1	G2
Completo	Omissão	Falta de classes do domínio do problema	7	3
	Informação Estranha	Classes que não fazem parte do contexto do domínio do problema	4	7
	Informação Estranha	Relacionamentos que não fazem parte do contexto do domínio do problema	1	6
	Omissão	Falta de cardinalidade	14	21
	Omissão	Falta de atributos importantes nas classes	0	5
	Informação Estranha	Cardinalidade que não fazem parte do contexto do domínio do problema	9	9
	Omissão	Falta de relacionamentos entre as classes	7	10
	Ambiguidade	Classes com descrições ambíguas	0	2
Parcial	Redundante	Classes desnecessárias	2	0
Temporário	Fato Incorreto	Relacionamentos errados na sintaxe	5	5

Tabela 8.2: Riscos de Falhas de Comunicação em Diagramas de Classes

Houve um número maior de riscos de falhas Completas. Também houveram riscos Parciais e Temporários, como o uso de classes desnecessárias e relacionamentos errados na sintaxe, respectivamente. A

Tabela 8.3 sumariza o modo em que as duplas empregaram as DCs. Em relação aos princípios de modelagem, alguns participantes relataram que: “A modelagem com foco na comunicação me ajudou a escolher os elementos que eu deveria incluir na modelagem. Com o uso da DC1, evitei incluir elementos que representassem informações fora do escopo” (P3) e “O uso das diretivas me ajudaram a pensar nas classes relevantes para o sistema” (P14).

Sobre as decisões de design, os participantes relataram que: “A DC2 me ajudou a entender sobre o que deve estar na modelagem do sistema. Informações desnecessárias podem atrapalhar no entendimento do sistema” (P2) e “A modelagem com as DCs me ajudaram na resolução do problema. Elas me ajudaram a decidir o que eu deveria abstrair ou não do cenário de modelagem” (P10).

Alguns participantes relataram suas percepções sobre a modelagem com foco na comunicação de equipes. Por exemplo, “A experiência foi boa para modelar os

diagramas para a comunicação de equipes. Ela ajudou a produzir modelos mais objetivos. Fiz a modelagem com os detalhes que achei necessários” (P12) e “A modelagem é um fator importante para facilitar a comunicação com a equipe” (P20).

Participantes	Uso das DCs	
	Princípios de Modelagem	Decisões de Projeto
P1	-	-
P2	-	X
P3	X	-
P4	-	X
P5	X	-
P6	X	-
P7	X	-
P8	-	X
P9	X	-
P10	X	-
P11	X	-
P12	X	-
P13	X	X
P14	-	X
P15	-	-
P16	-	-
P17	-	-
P18	-	X

Tabela 8.3: Emprego das Diretivas de Comunicabilidade na Modelagem.

Após isso, o professor da disciplina na qual o estudo foi realizado foi entrevistado. Com essa entrevista, foi obtido o seguinte feedback sobre a proposta das DCs no treinamento de modelagem: *“Percebi que os alunos entenderam melhor o que o diagrama comunicava após o treinamento e o uso dos DCs. Alguns alunos me disseram que entenderam melhor como deveriam modelar os diagramas após a apresentação do Learning Object”*. Portanto, os professores podem adotar a proposta das DCs no ensino de modelagem de software. O material usado para apoiar o uso de DCs no ensino de modelagem de software é detalhado em (Lopes *et al.*, 2019e).

Os resultados indicaram que o *Learning Object* e DCs impressas como material adicional contribuem para a modelagem de software com foco na redução de riscos de falhas de comunicação. No entanto, também percebeu-se que o Grupo 1, o que utilizou as DCs impressas durante a modelagem, ainda obteve um menor número de defeitos comparados ao Grupo 2.

8.4.2 Limitações do Estudo

Neste estudo existem limitações em relação à modelagem com o apoio das DCs ter sido realizada apenas para diagramas de classes, pois este diagrama não representa todos os diagramas usados para dar suporte à comunicação. Assim, é importante que as DCs sejam exploradas neste contexto em novos estudos experimentais. Além disso, a modelagem foi realizada para apenas um cenário de um sistema. Portanto, os resultados não são generalizáveis, mas são considerados válidos para o contexto de projeto de tal sistema.

Outra limitação é a expectativa da pesquisadora principal quanto aos benefícios das DCS na modelagem de diagramas de classes para redução dos riscos de falhas de comunicação. Para minimizar esta limitação, a avaliação dos riscos de falhas de comunicação nos diagramas de classes foi avaliada por mais três pesquisadores.

8.5 Considerações sobre o Capítulo

Este capítulo apresentou as contribuições da proposta das DCs para outros aspectos relacionados à comunicação via artefato. As DCs apoiaram na compreensão de modelos BPMN antes da sua evolução, sendo a compreensão e evolução duas etapas da manutenção de artefatos de software. Tal apoio faz parte da técnica e-VOL BPMN (Campos *et al.*, 2019), uma técnica apoia a manutenção de modelos BPMN, cuja avaliação experimental mostrou que esta técnica apoia em tal atividade.

As DCs também forneceram base de conhecimento para o desenvolvimento das técnicas ComD2 (Lopes *et al.*, 2018b), técnicas que auxiliam na inspeção de defeitos que são considerados como riscos de falhas de comunicação a partir de artefatos de software. Os resultados da avaliação experimental mostraram que tais técnicas apoiaram a identificação eficaz destes tipos de defeitos.

Por fim, o uso das DCs no ensino de modelagem com foco na comunicação via artefato indicou melhorias na comunicabilidade dos diagramas produzidos pelos alunos (Lopes *et al.*, 2019b). Os alunos produziram modelos com conteúdo claro, objetivo, relevante e relacionado à solução adequada para o sistema na perspectiva dos professores. Com esta proposta, espera-se treinar estudantes de engenharia de software que reflitam sobre o conteúdo dos modelos que serão consumidos durante o processo de desenvolvimento de software.

CAPÍTULO 9 – CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS

Este capítulo apresenta as considerações finais desta tese. Além disso, são discutidas perspectivas futuras de trabalho com as Diretivas de Comunicabilidade.

9.1 Considerações sobre o Trabalho

O objetivo desta tese consistiu na elaboração e demonstração de uso das Diretivas de Comunicabilidade (DCs) para apoiar a comunicação efetiva entre produtores e consumidores de artefatos desenvolvidos nas etapas iniciais do processo de desenvolvimento, respondendo a questão que guiou esta pesquisa (*Como apoiar a comunicação efetiva de produtores e consumidores de artefatos desenvolvidos nas etapas iniciais do desenvolvimento de software?*). Em relação às conjecturas teóricas, definidas nesta pesquisa, com base no ciclo de *Design Science Research* (DSR), estas se mostraram válidas, pois as DCs foram úteis para apoiar a reflexão dos produtores de artefatos sobre a comunicação a partir dos artefatos. Isso promoveu uma melhor comunicabilidade dos artefatos e compreensão dos fatores, por parte dos produtores, que impactam neste tipo de comunicação. Além disso, foi possível observar que as DCs podem ser melhor empregadas nos artefatos que representam “como o sistema será construído”, o qual é compartilhado com uma equipe, ao invés de empregá-las quando os artefatos estão sendo elaborados para explorar ideias para o desenvolvimento do sistema.

As DCs podem ser empregadas por profissionais e pesquisadores interessados na comunicação eficaz via artefato de software. A seguir são apresentados os resultados relacionados aos objetivos específicos, que foram descritos no Capítulo 1:

- **Fornecer evidências sobre o estado arte da comunicação via artefato para compreender os trabalhos na literatura que apoiam este tipo de comunicação** – No Capítulo 3 foi apresentado um mapeamento sistemático da literatura realizado com o objetivo de responder à questão de pesquisa “*Qual o estado da arte sobre artefatos de software como apoio à comunicação no desenvolvimento de*

software?”. Os resultados mostraram artefatos de software que tem apoiado a comunicação de equipes, pesquisas relacionadas com a comunicação via artefato e pesquisas que contribuem para este tipo de comunicação. Além disso, foram observadas oportunidades de pesquisa em relação às considerações que devem ser realizadas por parte dos produtores de artefatos para que ocorra a compreensão recíproca entre estes e consumidores de artefatos, evidenciando a justificativa para a proposta das DCs.

- **Fornecer evidências experimentais acerca da comunicação entre produtores e consumidores de artefatos de software** – No Capítulo 4 foi apresentado três estudos experimentais realizados para uma melhor compreensão da comunicação via artefato. Os resultados do primeiro estudo mostraram que algumas práticas de equipes de desenvolvimento de software podem contribuir para a utilização de artefatos como meio de comunicação, como o uso de um artefato que concentre as informações que devem ser desenvolvidas em outros artefatos de software. Os outros dois estudos demonstraram que a comunicabilidade de artefatos impacta na comunicação de produtores e consumidores.
- **Criar uma proposta e demonstrar que esta pode nortear os produtores sobre os fatores que podem impactar este tipo de comunicação e na comunicabilidade de artefatos para que a comunicação via artefato seja eficaz** – Nos Capítulos 5 e 6, a proposta das DCs e os resultados dos diferentes estudos experimentais mostraram que estas alcançaram o propósito de promover a reflexão dos produtores de artefatos, incluindo apoio para o entendimento dos fatores que podem impactar este tipo de comunicação. Portanto, as DCs contribuem para a comunicação efetiva entre produtores e consumidores de artefato.

Sobre as contribuições desta tese para outras pesquisas alinhadas à comunicação entre produtores e consumidores de artefatos, as DCs foram utilizadas para desenvolver parte da técnica e-VOL BPMN (Campos *et al.*, 2019), uma técnica que apoia a manutenção de modelos BPMN. A manutenção de modelos BPMN consiste na compreensão de modelos BPMN, pois é importante que o artefato seja bem compreendido antes de ser modificado, e na modificação do modelo BPMN, está relacionada com a alteração do artefato em si. Em relação à compreensão dos modelos com apoio da técnica

e-VOL BPMN, foram criadas questões que apoiam a compreensão do modelo com base nas DCs. As DCs também forneceram base de conhecimento para o desenvolvimento das técnicas ComD2 (Lopes *et al.*, 2018b), técnicas que auxiliam na inspeção de defeitos que são considerados como riscos de falhas de comunicação a partir de artefatos de software.

Além disso, as DCs foram exploradas no ensino de modelagem com foco na comunicação via artefato, indicando melhorias na comunicabilidade dos artefatos desenvolvidos pelos alunos (Lopes *et al.*, 2019b). Estes resultados indicam que a adoção das DCs não se limita às atividades da produção de artefatos, como previamente definido, mas abre novas perspectivas para a comunicação via artefato.

Devido aos benefícios que podem ser obtidos com esta pesquisa, esta tese se posiciona na fronteira entre as áreas de IHC (Interação Humano-Computador) e de Engenharia de Software. No que se refere à IHC, com base na Engenharia Semiótica, a mensagem do designer para os usuários, sobre como os usuários podem ou devem utilizar o sistema, pode ser melhor compreendida e codificada no sistema pelos desenvolvedores. Assim, é possível diminuir ruídos na comunicação pretendida pelo designer com os usuários através do sistema. Do ponto de vista da Engenharia de Software, os sistemas poderão ser desenvolvidos com uma menor quantidade de defeitos advindos de falhas de comunicação, reduzindo custos no retrabalho da equipe (de Mello *et al.*, 2014).

9.2 Perspectivas Futuras

A realização desta pesquisa resultou no desenvolvimento das DCs, a qual promove a reflexão dos produtores de artefatos sobre sua intenção de comunicação através destes para os consumidores de equipes de desenvolvimento. As DCs também podem ser adaptadas para serem utilizadas no contexto de diagnósticos de problemas de comunicabilidade de artefatos. Além disso, as DCs podem contribuir para outros aspectos de comunicação via artefato, como mostrou o Capítulo 6. A versão final das diretivas está no Apêndice C. Os resultados obtidos proporcionam novas perspectivas de pesquisa, que podem ser exploradas em trabalhos futuros. Algumas dessas perspectivas são:

Explorar as DCs em outros artefatos de software: isso pode ser realizado para que se compreenda o uso das DCs em artefatos de software não investigados nesta tese. Assim, sugere-se que novos estudos sejam realizados.

Investigação dos benefícios das DCs em outros contextos do desenvolvimento de software: através de novos estudos experimentais, as DCs podem ser exploradas nos artefatos que são utilizados por equipes de Desenvolvimento Distribuído. Com isso, pode-se identificar novas evidências sobre o uso das DCs.

Desenvolvimento de apoio ferramental para elaboração dos artefatos: para tornar a elaboração dos artefatos mais prática e evitar o esquecimento do uso das DCs durante a produção dos artefatos. Os produtores poderão assinalar as DCs empregadas em um determinado artefato.

Explorar o material elaborada com base nas DCs para o ensino de modelagem de novos diagramas com foco na comunicação via artefato: isso pode ser realizado para compreender os benefícios das DCs na formação de futuros engenheiros de software em relação sobre este tipo de comunicação, visto que diversos artefatos são utilizados durante o desenvolvimento de software.

REFERÊNCIAS

Abdelzad, V., Lethbridge, T. C. & Hosseini, M. (2016). The Role of Semiotic Engineering in Software Engineering. *Proceedings of the 5th International Workshop on Theory-Oriented Software Engineering*, 15-21.

Alzoubi, Y. I.; Gill, A. Q. & Al-Ani, A. (2016). Empirical Studies of Geographically Distributed Agile Development Communication Challenges: A Systematic Review. *Information & Management*, 53(1), 22-37.

Bandeira-de-Mello, R. & Cunha, C. (2003). Operacionalizando o Método da Grounded Theory nas Pesquisas em Estratégia: Técnicas e Procedimentos de Análise com Apoio do Software ATLAS/TI. *Encontro de Estudos em Estratégia*.

Bandeira-de-Mello, R. (2006). Softwares em Pesquisa Qualitativa. .In: Godoi, C. K., Bandeira-de-Mello, R., Silva, A. B. d. (eds), *Pesquisa Qualitativa em Estudos Organizacionais: Paradigmas, Estratégias e Métodos*, Chapter 15, São Paulo, Saraiva.

Barbosa, S. D. J., & Paula, M. G. (2003). Designing and evaluating interaction as conversation: a modeling language based on semiotic engineering. *Interactive Systems. Design, Specification and Verification, 10th DSV-IS Workshop*, 16–33.

Barbosa, S. D. J., & Silva, B. S., 2010. *Interação Humano-Computador*. Rio de Janeiro: Elsevier.

Brambilla, M. & Fraternali, P. (2014.) *Interaction Flow Modeling Language: Model-driven UI Engineering of Web and Mobile apps with IFML (1st ed.)*. Morgan Kaufmann.

Brown, J. M., Lindgaard, G., & Biddle, R. (2011). Collaborative events and shared artefacts: Agile interaction designers and developers working toward common aims. *Agile Conference (AGILE)*, 87–96.

Bordin, S., & De Angeli, A. (2016). Focal Points for a more User-Centred Agile Development. *International Conference on Agile Software Development*, 3-15.

Carver, J., Jaccheri, L., Morasca, S., Shull, F. (2003). Issues in Using Students in Empirical Studies in Software Engineering Education. *Proceedings of 9th International Symposium on Software Metrics (METRICS)*, 239 – 249.

Campos, U., Lopes, A., Oliveira, E., Gadelha, B., & Conte, T. (2019). e-VOL BPMN: a technique to support the evolution and learning of BPMN diagrams. *IET Software*. Doi: 10.1049/iet-sen.2019.0132

Chaudron, M. R., Fernandes-Saez, A., Hebig, R., Ho-Quang, T., & Jolak, R. (2018). Diversity in UML Modeling Explained: Observations, Classifications and Theorizations. *International Conference on Current Trends in Theory and Practice of Informatics*, 47-66.

Cicchetti, D. V., Showalter, D., Tyrer, P. J. (1985). The effect of number of rating scale categories on levels of interater reliability: a Monte Carlo investigation. *Applied Psychological Measurement*, 9(1), 31-36.

Conradi, R., Mohagneghi, M., Arif, T., Hedge L.C., Bunde, G. A., & Pendersen, A. (2003). Object-Oriented Reading Techniques for Inspection of UML Models – An Industrial Experiment. *Proceedings of the Object-Oriented Programming (ECOOP'03)*, 483-501.

Cooper, A. (1999). *The Inmates Are Running the Asylum: Why High Tech Products Drive Us Crazy and How to Restore the Sanity*. Sams

Cooper, A., Reimann, R. and Cronin, D. (2007). *About Face 3: The Essentials of Interaction Design*. Wiley

Cruz, E., Machado, R. & Santos M. (2015). Bridging the Gap between a set of Interrelated Business Process Models and Software Models. *Proceedings of the 17th International Conference on Enterprise Information Systems (ICEIS 2015)*, 338 – 345.

Cruz-Lemus., J. A., Maes, A., Genero, M., Poels, G. & Piattini, M. (2010). The Impact of Structural Complexity on The Understandability of UML Statechart Diagrams. *Information Sciences*, 180 (11), 2209-2220.

de Souza, C. S. (2005). *The semiotic engineering of human-computer interaction (Acting with Technology)* (1st ed.). The MIT Press.

de Souza, C. S., Cerqueira, R. F. G., Afonso, L. M., de Mello Brandão, R.R., & Ferreira, J.J. (2016). *Software developers as users: semiotic investigations in human-centered software development* (1st ed.). Springer International Publishing Switzerland. DOI 10.1007/978-3-319-42831-4.

Davis, F. D. (1989). Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology. *MIS Quarterly*, 319-340.

de Mello Brandão, R.R., Moreno, M.F., Ferreira, J.J. & Cerqueira, R. (2016) Communicability Issues on PaaS Application Development. *Proceedings of the 15th Brazilian Symposium on Human Factors in Computing Systems*, 1-10.

de Mello, R. M., Nogueira, E., Schots, M., Werner, C. M. L., & Travassos, G. H. (2014). Verification of Software Product Line Artefacts: A Checklist to Support Feature Model Inspections. *Journal of Universal Computer Science*, 20(5), 720-745.

Diel, E., Marczak, S., & Cruzes, D. S. (2016). Communication challenges and strategies in distributed devops. *Proceedings of the 11th International Conference on Global Software Engineering (ICGSE)*, 24-28.

Dresch, A., Lacerda, D. P., Antunes, & J. A. V. (2015). *Design Science Research: Método de pesquisa para avanço da ciência e tecnologia*. Porto Alegre: Bookman.

Eco, U. (1983). *Semiotics and the Philosophy of language*. Bloomington: Indiana University Press.

Fantechi, A. S., Gnesi, G. L. and Maccari, A. (2003). Applications of linguistic techniques for use case analysis. *Proceedings of the 10th International Requirements Engineering Conference (RE'02)*, 161-170.

Fabbri, S., Hernandez, E. M., Di Thommazo, A., Belgamo, A., Zamboni, A., & Silva, C. (2012). Managing literature reviews information through visualization. *Proceedings of the 14th International Conference on Enterprise Information Systems (ICEIS 2012)*, 36-45.

Fernandez, A., Abrahão, S., & Insfran, E. (2010). Towards to the validation of a usability evaluation method for model-driven web development. *Proceedings of the 4th*

ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '10), 54-57.

Fernandez, A., Insfran, E., & Abrahão, S. (2011). Usability evaluation methods for the web: A systematic mapping study. *Information and Software Technology*, 53(1), 89-817.

Fernandez, A., Abrahão, S., Insfran, E., & Matera, M. (2012). Further Analysis on the Validation of a Usability Inspection Method For Model-Driven Web Development. *International Symposium on Empirical Software Engineering and Measurement (ESEM '12)*, 153-156.

Fernández-Sáez, A., Genero, M., Chaudron, M., 2013. Empirical Studies Concerning the Maintenance of UML Diagrams and their Use in the Maintenance of Code: A Systematic Mapping Study. *Information and Software Technology*, 55 (7), 1119-1142.

Ferreira, J. J. (2015). Comunicação Através de Modelos no Contexto do Desenvolvimento de Software. Tese de Doutorado. PUC-Rio.

Ferreira, J. J., de Souza, C. S., & Cerqueira, R. (2014). Characterizing the Tool-Notation-People Triplet in software Modeling Tasks. *Proceedings of the 13th Brazilian Symposium on Human Factors in Computing Systems*, 31-40.

Garcia, A., Silva da Silva, T., & Selbach Silveira, M. (2017). Artifacts for Agile User-Centered Design: A Systematic Mapping. *Proceedings of the 50th Hawaii International Conference on System Sciences (HICSS.2017)*, 5859–5868.

Génova, G., Valiente, M. C., & Marrero, M. (2008). On the Difference Between Analysis and Design, and Why it is Relevant for the Interpretation of Models in Model Driven Engineering. *Journal of Object Technology*, 8 (1), 107-127.

Glaser, B. (1992). *Basics of Grounded Theory Analysis*. Mill Valley: The Sociology Press.

Glaser, B. & Strauss, A. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research*. New York: Aldine Transaction.

Gomaa, H. (2006). Designing concurrent, distributed, and real-time applications with UML. *Proceedings of the 28th international conference on Software engineering (ICSE '06)*, 1059-1060. DOI: <https://doi.org/10.1145/1134285.1134504>

Granda, M. F., Condori-Fernández, N., Vos, T. E., & Pastor, O. (2015). What do we know about the defect types detected in conceptual models?. *Proceedings of the 9th International Conference on Research Challenges in Information Science (RCIS)*, 88-99.

Grice, H. P. (1975). Logic and conversation. In *Syntax and Semantics 3: Speech arts*, ed. Peter Cole and Jerry Morgan, 41–58.

Henttonen, K. & Blomqvist, K. (2005). Managing distance in a global virtual team: the evolution of trust through technology-mediated relational communication. *Strategic Change*, 14(2), 107–119

Herbsleb, D. & Mockus, A. (2003). An empirical study of speed and communication in globally-distributed software development. *IEEE Transactions on Software Engineering*, 29(6), 1–14

Hess, A., Diebold, P., & Seyff, N. (2017). Towards Requirements Communication and Documentation Guidelines for Agile Teams. *Proceedings of the IEEE 25th International Requirements Engineering Conference Workshops (REW)*, 415-418.

Hevner, A. & Chatterjee, S. (2010). *Design research in information systems: theory and practice*, Springer Science & Business Media.

Jarke, M. (1999). Scenarios for modeling. *Communications of the ACM*, 42 (1), 47-48.

Jakobson, R. (1960). Linguistics and poetics. T. A. Sebeok (ed.), *Style in Language*. Cambridge, MA: The MIT Press, 350–377.

Käfer, V. (2017). Summarizing Software Engineering Communication Artifacts from Different Sources. *Proceedings of the 11th Joint Meeting on Foundations of Software Engineering (ESEC/FSE 2017)*, 1038-1041.

Kamthan, P. (2009). A framework for understanding and addressing the semiotic quality of use case models. *Model-Driven Software Development: Integrating Quality Assurance*, 327-351.

Khatwani, C., Jin, X., Niu, N., Koshoffer, A., Newman, L., & Savolainen, J. (2017). Advancing Viewpoint Merging in Requirements Engineering: A Theoretical Replication

and Explanatory Study. *Requirements Engineering*, 22(3), 317-338.

Kitchenham, B., & Charters, S. (2007). Guidelines for Performing Systematic Literature Reviews in Software Engineering. Version 2.3, EBSE Technical Report (EBSE-2007-01), Keele University and University of Durham.

Kitchenham, B., Budgen, D., & Brereton, O P. (2010). The value of mapping studies: a participant observer case study. *Proceedings of the 14th international conference on Evaluation and Assessment in Software Engineering (EASE'10)*, 25-33.

Likert, R., Roslow, S. and Murphy, G. (1993). A simple and reliable method of scoring the Thurstone attitude scales. *Personnel Psychology*, 46(3), pp.689-690.

Landis, J. R. e Koch, G.G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, (33), 159.

Lange, C. F. J., Wijns, M. A. M., & Chaudron, M. R. V. (2007). Supporting Task-Oriented Modeling Using Interactive UML Views. *Journal of Visual Languages and Computing*, 18(4), 399–419

Lange, C. F. & Chaudron, M. R. (2006). Effects of defects in UML models: an experimental investigation. *Proceedings of the 28th International Conference on Software Engineering*, 401-411.

Lopes, A., Marques, A. B., Barbosa, S. D. J. & Conte, T. (2015). Evaluating HCI Design with Interaction Modeling and Mockups: A Case Study. *Proceedings of International Conference on Enterprise Information Systems (ICEIS 2015)*, 79-87.

Lopes, A., Valentim, N., Moraes, B., Zilse, R., & Conte, T. (2016). Utilizando Modelagem de Interação para Projetar uma Aplicação Móvel Multiusuário: Um Relato de Experiência. *Anais do XV Simpósio Brasileiro de Qualidade de Software*, 374-388.

Lopes, A., Conte, T., & de Souza, C. S. (2017). Analyzing the Use Case Communicability. *Proceedings of the XVI Brazilian Symposium on Human Factors in Computing Systems (IHC 2017)*. ACM, New York, NY, USA, Article 13, 10 pages. DOI: <https://doi.org/10.1145/3160504.3160539>

Lopes, A., Valentim, N., Moraes, B., Zilse, R., & Conte, T. (2018a). Applying user-centered techniques to analyze and design a mobile application. *Journal of Software*

Engineering Research and Development, 6(1), Article 5, 23 pages.

Lopes, A., Campos, U., Conte, T., & de Souza, C. S. (2018b). ComD2: Family of Techniques for Inspecting Defects in Models that Affect Team Communication. *Proceedings of the 30th International Conference on Software Engineering and Knowledge Engineering*, Article 155, 6 pages.

Lopes, A., Campos, U., Conte, T., & de Souza, C. S. (2018c). ComD2 (Communication between Designer and Developers): A Family of Techniques for Inspecting Defects that Affect Communication from Models. *USES Research Group Technical Report. TR-USES-2018-0003*. Available: <http://uses.icomp.ufam.edu.br/wp-content/uploads/2018/03/TR-USES-2018-0003.pdf>

Lopes, A., Oliveira, E., Conte, T. & Souza., C. S. (2019a). Directives of Communicability: Towards Better Communication Through Software Models. *Proceedings of the 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE '19)*, 45-48.

Lopes, A., Oliveira, E., Conte, T. & Souza., C. S. (2019b). Directives of Communicability: Teaching Students How to Improve Communication Through Software Modeling. *Proceedings of 2019 IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSECompanion)*, 306-307.

Lopes, A., Conte, T. & Souza., C. S. (2019c). Reducing the Risks of Communication Failures Through Software Models. *Proceedings of XVIII Brazilian Symposium on Human Factors in Computing Systems (IHC 2019)*, 10 pages.

Lopes, A., Conte, T., de Souza, C. S. (2019d). TR-USES-2019-0012. An Exploratory Study on the Communication Between Producers and Consumers of Use Cases. USES Research Group Technical Report. Available: <http://uses.icomp.ufam.edu.br/wp-content/uploads/2019/12/TR-USES-2019-013.pdf>

Lopes, A., E. Oliveira, T. Conte and C. S. de Souza. (2019e). Teaching Students How to Communicate Better Through Software Models. Technical Report. Available: <http://uses.icomp.ufam.edu.br/wp-content/uploads/2019/01/TR-USES-2019-0001.pdf>

[Misbhauddin, M. & Alshayeb, M. \(2015\). Extending the UML use case metamodel](#)

with behavioral information to facilitate model analysis and interchange. *Software & Systems Modeling*, 14 (2), 813-838.

McBride, T. (2014). The role of boundary objects in the fuzzy front end of it development. *Proceedings of the 23rd Australian Software Engineering Conference*, 162-169.

Mafra, S., Barcelos, R., & Travassos, G. (2006). Aplicando uma metodologia baseada em evidência na definição de novas tecnologias de software. *Anais do XX Simpósio Brasileiro de Engenharia de Software (SBES 2006)*, 239- 254.

Manso, P., M. E., Cruz-Lemus, J. A., Genero, M., & Piattini, M. (2009). Empirical Validation of Measures for UML Class Diagrams: A Meta-Analysis Study. *Proceedings of the International Conference on Model Driven Engineering Languages and Systems (MODELS'09)*, 303–313.

Marangunić, N., & Granić, A. (2015). Technology Acceptance Model: a Literature Review From 1986 to 2013. *Universal Access in the Information Society*, 14(1), 81-95.

Marques, A.B., Barbosa, S.D.J. & Conte, T. (2017). Defining a notation for usability oriented interaction and navigation modeling for interactive systems. *SBC Journal on Interactive Systems*, 8(2), 35-49.

Massago, Y., Balancieri, R., Penteadó, M., Moriya Huzita, E. H., & Vivian, R. L. (2013). Um Mecanismo de Captura de Informações Contextuais em um Ambiente de Desenvolvimento Distribuído De Software. *XVIII Congreso Argentino de Ciencias de la Computación*.

McCrum-Gardner, E. (2008). Which is the Correct Statistical Test to Use?. *British Journal of Oral and Maxillofacial Surgery*, 46 (1), 38–41.

Nakamura, Y., Sakamoto, K., Inoue, K., Washizaki, H. & Fukazawa, Y. (2011). Evaluation of Understandability of UML Class Diagrams by Using Word Similarity. *Joint Conference of the 21st Int'l Workshop on and 6th Int'l Conference on Software Process and Product Measurement (IWSM-MENSURA)*, 178-187

Nugroho, A. (2009). Level of Detail in UML Models and its Impact on Model Comprehension: A Controlled Experiment. *Information and Software Technology*,

51(12), 1670–1685.

OMG, 2014. Business Process Model and Notation (BPMN). Documents Associated with BPMN 2.0.2.

OMG. (2011). Business Process Model and Notation (BPMN) Version 2.0.

OMG. (2015). Unified Modeling Language TM (UML) Version 2.5.

Peppers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007) A design science research methodology for information systems research. *Journal of Management Information Systems*, 4 (3), 45-77.

Petre, M. (2013). UML in practice. *Proceedings of the 2013 International Conference on Software Engineering (ICSE 2013)*, 722-731.

Pikkarainen, M., Haikara, J., Salo, O., Abrahamsson, P., & Still, J. (2008). The impact of agile practices on communication in software development. *Empirical Software Engineering*, 13(3), 303-337.

Pimentel, M. (2017). Design Science Research e Pesquisas com os Cotidianos Escolares para fazer pensar as pesquisas em Informática na Educação. *Simpósio Brasileiro de Informática na Educação (SBIE)*, 28 (1), 414 - 424.

Rabelo, J., Viana, D., Santos, G. & Conte, T. (2014). Usando PABC-Pattern para codificar o conhecimento: um estudo experimental. Em *Anais do Simpósio Brasileiro de Qualidade de Software (SBQS, 2014)*, 1-15.

Reggio, G., Leotta, M., Ricca, F., & Clerissi, D. (2014). What are the used activity diagram constructs? A survey. *Proceedings of the 2nd International Conference on Model-Driven Engineering and Software Development (MODELSWARD 2014)*, 87–98.

Rittel, H.W. & Webber M.M. (1973). Dilemmas in a general theory of planning. *Policy sciences*, 4(2), 155-69.

Sánchez, R. A., & Hueros, A. D. (2010). Motivational Factors That Influence The Acceptance of Moodle Using TAM. *Computers in Human Behavior*, 26(6), 1632–1640.

Schoonewille, H. H., Heijstek, W., Chaudron, M. R. & Kühne, T. (2011). A cognitive

perspective on developer comprehension of software design documentation. *Proceedings of the 29th ACM International Conference on Design of Communication*, 211-218.

Schwaber, K. & Shuterland, J. (2011). *The Scrum Guide: The Definitive Guide to Scrum, The rules of the game*, s.l.: Scrum.org.

Sebe, N. (2010). Human-centered computing. Nakashima, H., Aghajan, H., & Augusto, J (Eds.), *Handbook of Ambient Intelligence and Smart Environments*, 349–370, doi: 10.1007/978-0-387-93808-0_13.

Seddig-Raufie, D., Jansen, L., Schober, D., Boeker, M., Grewe, N., & Schulz, S. (2012). Proposed Actions are no Actions: Re-Modeling an Ontology Design Pattern with a Realist Top-Level Ontology. *Journal of Biomedical Semantics*, 3, (2), S2.

Shapiro, S. S., & Wilk, M. B. (1965). An Analysis of Variance Test for Normality (Complete Samples). *Biometrika*, 52(3/4), 591-611.

Shishkov, B., Xie, Z., Lui, K., & Dietz, J. (2002). Using Norm Analysis to Derive Use Case from Business Processes. *Proceedings of 5th Workshop on Organizations Semiotic OSs*, 187-195.

Shull, F., Carver, J. & Travassos, G. (2001). An empirical methodology for introducing software processes. *Proceedings of 8th European software engineering conference held jointly with 9th ACM SIGSOFT International Symposium on Foundations of Software Engineering (ESEC/FSE-9)*, 288-296.

Strauss, A., 1987. *Qualitative analysis for social scientists*. New York: Cambridge University Press.

Strauss, A.; Corbin, J., 1998. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. London: SAGE Publications.

Storey, M. A., Zagalsky, A., Singer, L. & German, D. (2017). How Social and Communication Channels Shape and Challenge a Participatory Culture in Software Development. *IEEE Transactions on Software Engineering*, 43(2), 185-204.

Sultan, F., & Chan, L. (2000). The Adoption of New Technology: The Case Of Object-Oriented Computing in Software Companies. *IEEE Transactions on Engineering Management*, 47(1), 106-126.

Travassos, G. H. & Barros, M. (2003). Contributions of In Virtuo and In Silico Experiments for the Future of Empirical Studies in Software Engineering. *Proceedings of 2nd Workshop in Workshop Series on Empirical Software Engineering (WSESE 2003)*, 117-130.

Valentim, N. M. C., Rabelo, J., Oran, A. C., Marczak, S., & Conte, T. (2015). A controlled experiment with usability inspection techniques applied to use case specifications: comparing the MIT 1 and the uce techniques. *Proceedings of the 18th International Conference on Model Driven Engineering Languages and Systems, (MODELS 2015)*, 206-215.

Venkatesh, V., Bala, H. (2008). Technology Acceptance Model 3 and a Research Agenda on Interventions. *Decision Sciences*, 39(2), 273-315.

Venkatesh, V., Morris, M. G., Davis, G. B. & Davis, F. D. (2003). User Acceptance of Information Technology: Toward a Unified View. *MIS Quarterly*, 425-478.

Vivian, R. L., Huzita, E., Balancieri, R., Galdamez, E. V. C., et. al (2018). Improving Coordination and Communication in Distributed Software Development through Context-Based Software Artifacts Awareness: A Controlled Experiment. *Journal of Computer Science*, 14(11), 1531-1545. DOI: 10.3844/jcssp.2018.1531.1545

Voigt, S. (2017). A Method for Documenting Agile Software Projects. *European Conference on Knowledge Management*, 1035-1044.

Weitlaner, D., Guettinger, A., & Kohlbacher, M. (2013). Intuitive Comprehensibility of Process Models. *International Conference on Subject-Oriented Business Process Management*, 52-71.

Wieringa, R. (2014). Design Science Methodology for Information Systems and Software Engineering. Springer-Verlag Berlin Heidelberg.

APÊNDICE A – Artigos Identificados no Mapeamento Sistemático

[P001] Abdullah, N. N. B., Honiden, S., Sharp, H., Nuseibeh, B., & Notkin, D. (2011). Communication Patterns of Agile Requirements Engineering. *Proceedings of the 1st Workshop on Agile Requirements Engineering*, 1-4.

[P002] Aranguren, M. E., Antezana, E., Kuiper, M., & Stevens, R. (2008, April). Ontology Design Patterns for Bio-Ontologies: A Case Study on the Cell Cycle Ontology. *BMC Bioinformatics*, 9(S5), S1.

[P003] Bernonville, S., Kolski, C., Leroy, N., & Beuscart-Zéphir, M. C. (2010). Integrating the SE and HCI Models in the Human Factors Engineering Cycle For Re-Engineering Computerized Physician Order Entry Systems for Medications: Basic Principles Illustrated by a Case Study. *International Journal of Medical Informatics*, 79(4), e35-e42.

[P004] Bordin, S., & De Angeli, A. (2016). Focal Points for a more User-Centred Agile Development. *International Conference on Agile Software Development*, 3-15.

[P005] Brown, J., & Johnston, C. (2001). Extending the Value of Prototypes with Panorama a Tool to Browse Software Artefacts. *Proceedings of the Symposium on Computer Human Interaction*, 15-20.

[P006] Davies, I., Green, P., Rosemann, M., Indulska, M., & Gallo, S. (2006). How do Practitioners Use Conceptual Modeling in Practice? *Data & Knowledge Engineering*, 58(3), 358-380.

[P007] Fatima, A., Rasool, T., & Qamar, U. (2018). GDGSE: Game Development with Global Software Engineering. *IEEE Games, Entertainment, Media Conference (GEM)*, 1-9.

[P008] Hess, A., Diebold, P., & Seyff, N. (2017). Towards Requirements Communication and Documentation Guidelines for Agile Teams. *IEEE 25th International Requirements Engineering Conference Workshops (REW)*, 415-418.

[P009] Ho-Quang, T., Hebig, R., Robles, G., Chaudron, M. R., & Fernandez, M. A. (2017). Practices and Perceptions of UML Use in Open Source Projects. *IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*, 203-212.

[P010] Käpyaho, M., & Kauppinen, M. (2015). Agile Requirements Engineering with Prototyping: A Case Study. *IEEE 23rd International requirements engineering conference (RE)*, 334-343.

[P011] Kautz, K. (1993). Communication Support for Prototyping Projects. *Information and Software Technology*, 35(11-12), 647-651.

[P012] Mirel, B., Olsen, L. A., Prakash, A., & Soloway, E. (1999). Improving Quality in Software Engineering through Emphasis on Communication. *Computer Standards & Interfaces*, 2(21), 159.

[P013] Nakajo, T., Azuma, I., & Tada, M. (1993). A Case History Development of a Foolproofing Interface Documentation System. *IEEE Transactions on Software Engineering*, 19(8), 765-773.

[P014] Paech, B., Dorr, J., & Koehler, M. (2005). Improving Requirements Engineering Communication in Multiproject Environments. *IEEE Software*, 22(1), 40-47.

[P015] Savolainen, J., & Mannisto, T. (2010). Conflict-centric Software Architectural Views: Exposing Trade-Offs in Quality Requirements. *IEEE Software*, 27(6), 33-37.

[P016] Schoonewille, H. H., Heijstek, W., Chaudron, M. R., & Kühne, T. (2011). A Cognitive Perspective on Developer Comprehension of Software Design Documentation. *Proceedings of the 29th ACM international conference on Design of communication*, 211-218.

[P017] Sheppard, S. B., Kruesi, E., & Bailey, J. W. (1982). An Empirical Evaluation of Software Documentation Formats. *Proceedings of the 1982 Conference on Human Factors in Computing Systems*, 121-124.

[P018] Tilley, S. (2009). Documenting Software Systems with Views VI: Lessons Learned from 15 years of Research & Practice. *Proceedings of the 27th ACM international conference on Design of communication*, 239-244.

[P019] Vivian, R. L., Huzita, E. H. M., & Leal, G. C. L. (2013). Supporting Distributed Software Development Through Context Awareness on Software Artifacts: The DISEN-Collaborar Approach. *Proceedings of the 28th Annual ACM Symposium on Applied Computing*, 765-770.

[P020] Vivian, R. L., Huzita, E., Balancieri, R., Galdamez, E. V. C., et. al (2018). Improving Coordination and Communication in Distributed Software Development through Context-Based Software Artifacts Awareness: A Controlled Experiment. *Journal of Computer Science* 14(11), 1531-1545. DOI: 10.3844/jcssp.2018.1531.1545

[P021] Voigt, S. (2017). A Method for Documenting Agile Software Projects. *European Conference on Knowledge Management*, 1035-1044.

APÊNDICE B – Material de Apoio Utilizado nos Estudos

ESTUDO 1: PRIMEIRO ESTUDO EXPERIMENTAL PARA AVALIAÇÃO DAS DCs

• Cenário do problema de modelagem

O Coordenador do Comitê de Programa de um Simpósio de Qualidade de Software precisa de um sistema para gerenciar a submissão de artigos. É possível submeter dois tipos de artigos para este simpósio:

- **Trabalhos técnicos:** artigos de alta qualidade descrevendo resultados pioneiros de pesquisa na área de qualidade de software com contribuição acadêmica.

- **Relatos de experiência:** artigos de alta qualidade descrevendo e analisando a aplicação de processos, métodos ou ferramentas de qualidade de software, contextualizando a experiência, mostrando os resultados obtidos e lições aprendidas de uma experiência prática com contribuição para a indústria de software.

O Coordenador do Comitê de Programa deve inicialmente: definir os tópicos de interesse do simpósio (válidos para os dois tipos de artigos), definir os campos dos formulários para avaliação dos artigos e definir a data limite para submissão dos artigos. Após isso, ele convida os revisores que irão compor o Comitê de Programa (pessoas que avaliam os artigos submetidos). Os revisores podem aceitar ou não o convite.

Após as atividades iniciais (cadastro de tópicos e formulários de avaliação, convite a revisores e definição da data limite), o Coordenador “abre” o sistema para a submissão de artigos. A partir desta data até a data limite de submissão de artigos, os autores podem submeter seus artigos para o simpósio. Cada autor deve informar: o título do artigo, nome, e-mail e afiliação de cada autor, o tipo de artigo (se é relato de experiência ou trabalho técnico) e os tópicos da conferência relacionados com o artigo, além de fazer o *upload* do arquivo do artigo em formato PDF.

Logo após a data limite para submissão, o Coordenador deve atribuir os artigos para os revisores do Comitê de Programa. Cada artigo deve ser avaliado por três revisores do Comitê de Programa. O **sistema** deve sugerir uma atribuição de artigos aos revisores a partir dos tópicos de interesse escolhidos por cada revisor (definidos no momento da aceitação do convite para compor o Comitê). A atribuição de artigos é feita relacionando os tópicos definidos para cada artigo com os tópicos de interesse de cada revisor. O Coordenador pode verificar e alterar a sugestão de atribuição de artigos feita pelo sistema. Após fechar a atribuição de todos os artigos, o Coordenador deve acionar uma opção do sistema para avisar cada revisor por e-mail de quais foram os artigos atribuídos a ele e qual o prazo para a revisão.

Os revisores devem acessar o sistema para obter o arquivo PDF de cada artigo e visualizar o formulário de avaliação. Cada revisor marca suas notas e comentários (pontos fortes, pontos fracos e sugestões de melhoria) sobre cada artigo nos itens cadastrados no formulário de avaliação.

Quando termina o período de revisão, o Coordenador acessa as avaliações de cada artigo. Caso um artigo possua avaliações conflitantes (uns revisores avaliaram o artigo positivamente e outros avaliaram negativamente o mesmo artigo), ele pode mandar mensagens para os revisores, pedindo para que eles discutam o artigo (através de outras mensagens) e cheguem a um consenso. Neste período de consenso, um revisor pode alterar sua nota para um artigo.

Terminado o período de consenso, o Coordenador solicita ao sistema um “**Relatório de Avaliações dos Artigos**” mostrando todos os artigos ordenados pelas notas (do maior para o menor). Este relatório deve apresentar para cada artigo: o título do artigo, o tipo, os autores, além dos comentários de cada revisor. Analisando este relatório, o Coordenador atribui a cada artigo o estado “Aprovado” ou “Não Aprovado”. Ao finalizar esta classificação, o Coordenador cadastra os textos de e-mails parabenizando os autores de artigos aprovados e informando os autores de artigos não aprovados da sua situação. O Coordenador dispara o envio destes e-mails pelo próprio sistema.

Ao receber o e-mail de artigo aprovado, o autor deve preparar uma nova versão do artigo, chamada “Camera Ready”, no qual ele busca melhorar o artigo de acordo com as sugestões dos revisores. O autor deve fazer um novo upload da versão Camera Ready no sistema até a data limite informada pelo Coordenador. Por fim, após o Coordenador coletar a versão Camera Ready dos artigos, estes passam a estar inclusos nos anais do evento.

ESTUDO 2: SEGUNDO ESTUDO EXPERIMENTAL PARA AVALIAÇÃO DAS DCs

- **Propósito dos sistemas modelados**

E1. Minha interação com as Diretivas de Comunicabilidade foi clara e compreensível.							
E2. Interagir com as Diretivas de Comunicabilidade não exigiu muito do meu esforço mental.							
E3. Considero fácil utilizar as Diretivas de Comunicabilidade.							
E4. Eu acho fácil utilizar as Diretivas de Comunicabilidade para melhorar a minha comunicação com a equipe a partir dos artefatos.							

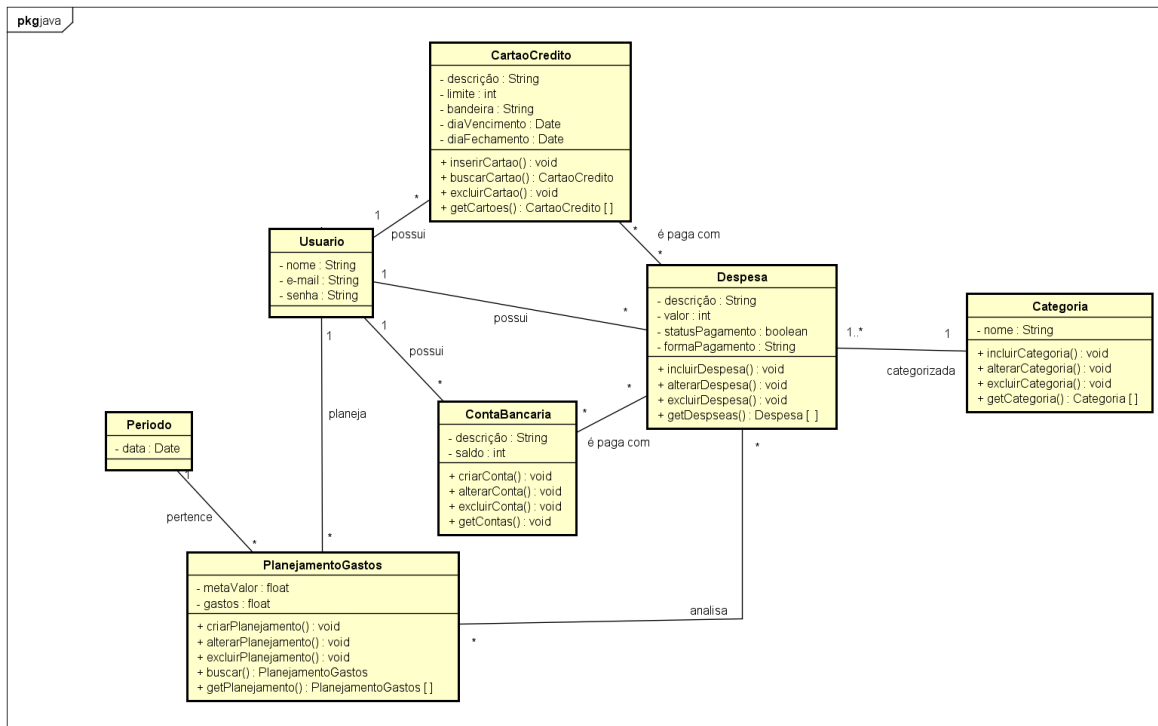
Utilidade Percebida							
U1. Usar as Diretivas de Comunicabilidade torna o meu desempenho melhor sobre as informações que precisam ser compreendidas pela equipe a partir dos artefatos.							
U2. Usar estas diretivas em meu trabalho poderá melhorar a minha produtividade, pois não terei que corrigir informações que não seja compreendidas por colegas (do projeto de desenvolvimento).							
U3. Usar as Diretivas de Comunicabilidade melhora a efetividade da minha comunicação com a equipe a partir dos artefatos.							
U4. Eu considero as Diretivas de Comunicabilidade úteis para melhorar a minha comunicação com a equipe a partir dos artefatos.							

Intenção de Uso							
I1. Supondo que eu tenho tempo suficiente para desenvolver artefatos de software, eu utilizaria estas diretivas.							
I2. Levando em conta que eu tenho domínio para escolher qualquer suporte para o desenvolvimento de artefatos de software, eu prevejo que eu irei usar estas diretivas							
I3. Eu pretendo utilizar as diretivas nos próximos meses.							

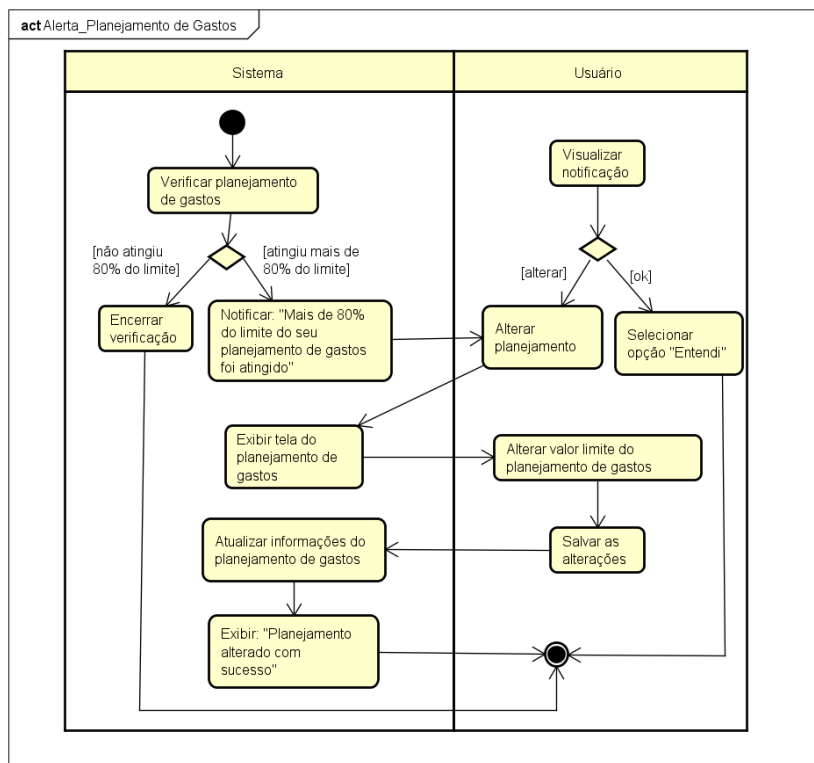
2. Cite pontos positivos e negativos relacionados às Diretivas de Comunicabilidade. Com sua resposta, melhorias poderão ser realizadas para apoiar outros profissionais na melhoria de sua comunicação a partir de artefatos de software.

ESTUDO 3: PRODUÇÃO DE ARTEFATOS COM AS DCS

- Modelo produzidos com as DCs (diagrama de classes e atividades da UML)

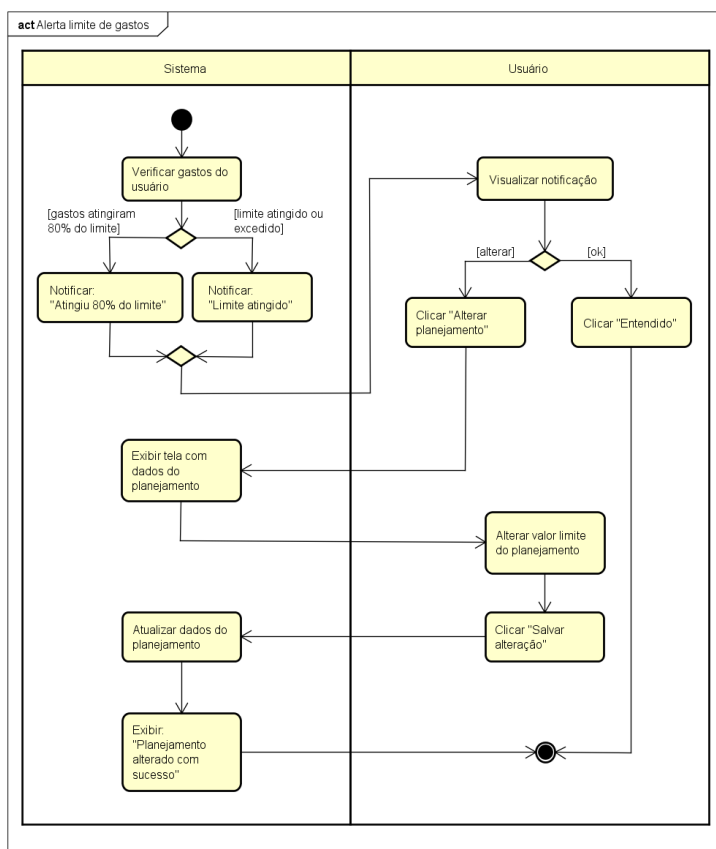
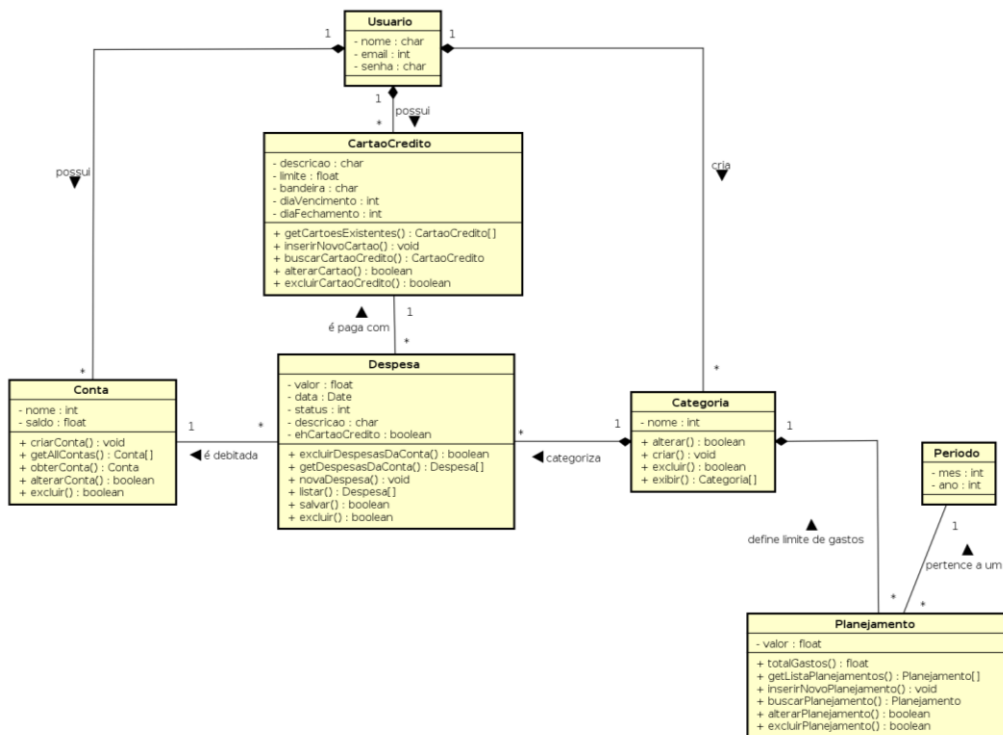


powered by Astah



powered by Astah

- Modelo produzidos sem as DCs (diagrama de classes e atividades da UML)



ESTUDO 4 e ESTUDO 5: QUESTIONÁRIO ENVIADO NA ETAPA 1

Produtores

Através das suas respostas para as questões das próximas seções, nós poderemos analisar os artefatos como "meio de comunicação".

Nome: _____

Endereço de e-mail: _____

Qual o nome do projeto? _____

Qual o seu papel na equipe de desenvolvimento deste projeto de software? _____

Quais os artefatos utilizados como meio de comunicação na equipe? (Por favor, responda esse questionário sobre cada um dos artefatos utilizados) _____

Qual a sua opinião sobre este artefato como meio de comunicação? _____

1. Em relação ao artefato de software como meio de comunicação, qual o seu grau de concordância em relação à:

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.							
Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.							
Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.							
Este artefato me auxilia no que eu gostaria de "falar" para a equipe sobre o domínio do problema.							
Este artefato permite que o meu projeto para o software seja representado de forma clara para a equipe.							
Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).							
Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.							

2. Em relação ao conteúdo comunicativo do artefato de software, qual o seu grau de concordância em relação à:

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.							
Não foram introduzidas informações desorganizadas no artefato.							
Não foi introduzido o excesso de informações no artefato.							
A quantidade de informações no artefato é suficiente para o entendimento da equipe.							
Acredito que o conteúdo do artefato está claro e é de fácil compreensão.							

Caso queira relatar algo sobre as questões dessa seção, por favor, fique à vontade.

Consumidores

Através das suas respostas para as questões das próximas seções, nós poderemos analisar os artefatos como "meio de comunicação" em sua equipe.

Nome: _____

Endereço de e-mail: _____

Qual o nome do projeto? _____

Qual o seu papel na equipe de desenvolvimento deste projeto de software? _____

Quais os artefatos utilizados como meio de comunicação na equipe? (Por favor, responda esse questionário sobre cada um dos artefatos utilizados) _____

Qual a sua opinião sobre este artefato como meio de comunicação? _____

1. Sobre esses artefatos, qual o seu grau de concordância em relação à:

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.							
Eu acho fácil compreender este artefato.							
Eu irei adotar esse artefato para compreender as atividades de desenvolvimento							
Eu compreenderei melhor o software com este artefato.							
Eu acho fácil usar este artefato.							

Caso queira relatar algo sobre as questões dessa seção, por favor, fique à vontade.

ESTUDO 4: QUESTIONÁRIO APLICADO APÓS O USO DAS DCs NA ETAPA 2

Avaliação das Diretivas de Comunicabilidade

Nome: _____

Endereço de e-mail: _____

Na sua opinião, quais são os pontos positivos de negativos da Diretivas de Comunicabilidade?

Você gostaria de relatar alguma sugestão de melhoria para o uso das Diretivas de Comunicabilidade?

3. Responda as questões a seguir considerando a sua experiência com as Diretivas de Comunicabilidade (DCs).

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
E1. Minha interação com as Diretivas de Comunicabilidade foi clara e compreensível.							
E2. Interagir com as Diretivas de Comunicabilidade não exigiu muito do meu esforço mental.							
E3. Considero fácil utilizar as Diretivas de Comunicabilidade.							
E4. Eu acho fácil utilizar as Diretivas de Comunicabilidade para melhorar a minha comunicação com a equipe a partir dos artefatos.							

Caso queira relatar algo sobre as questões dessa seção, por favor, fique à vontade.

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
U1. Usar as Diretivas de Comunicabilidade torna o meu desempenho melhor sobre as informações que precisam ser compreendidas pela equipe a partir dos artefatos.							
U2. Usar estas diretivas em meu trabalho poderá melhorar a minha produtividade, pois não terei que corrigir informações que não seja compreendidas por colegas (do projeto de desenvolvimento).							
U3. Usar as Diretivas de Comunicabilidade melhora a efetividade da minha comunicação com a equipe a partir dos artefatos.							
U4. Eu considero as Diretivas de Comunicabilidade úteis para melhorar a minha comunicação com a equipe a partir dos artefatos.							

Caso queira relatar algo sobre as questões dessa seção, por favor, fique à vontade.

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
I1. Supondo que eu tenho tempo suficiente para desenvolver artefatos de software, eu utilizaria estas diretivas.							
I2. Levando em conta que eu tenho domínio para escolher qualquer suporte para o desenvolvimento de artefatos de software, eu prevejo que eu irei usar estas diretivas							
I3. Eu pretendo utilizar as diretivas nos próximos meses.							

Caso queira relatar algo sobre as questões dessa seção, por favor, fique à vontade.

APÊNDICE C – Diretivas de Comunicabilidade

DIRETIVAS DE COMUNICABILIDADE PARA ARTEFATOS DE SOFTWARE

[Disponível aqui](#)

USES **serg**

Diretivas de Comunicabilidade

DC1 DC3
DC2 DC4

PRODUTOR → [Análise | Projeto] ← CONSUMIDOR → [Implementação | Teste]

DC1 – “Diga a verdade!”: Use informações verdadeiras

- Não use informações que estejam fora do domínio do problema
- Não use informações desatualizadas

DC2 – “Diga o que é necessário e não mais que o necessário”: Use o conteúdo necessário

- Insira todo o conteúdo necessário no artefato, isto é, a quantidade de informação adequada que você julgue como necessária para o entendimento desse artefato
- Não use conteúdo desnecessário no artefato

DC3 – “Diga de forma lógica”: Organize as informações de forma coerente

- Organize as informações de forma que estas sejam compreendidas da melhor maneira

DC4 – “Diga de forma clara”: Organize as informações de forma clara

- Use termos claros
- Evite a ambiguidade
- Evite interpretação implícita por parte da equipe

DIRETIVAS DE COMUNICABILIDADE PARA ARTEFATOS DE SOFTWARE ESPECÍFICOS

- Casos de Uso da UML - [Disponível aqui](#)



Casos de Uso da UML

DC1 – “Diga a verdade!”: Use informações verdadeiras na modelagem e especificação de casos de uso

- Não use informações nos casos de uso que estejam fora do domínio do problema
- Remova informações desatualizadas

DC2 – “Diga o que é necessário e não mais que o necessário”: Use o conteúdo necessário na modelagem e especificação de casos de uso

- Insira todos os casos de uso no diagrama que estão relacionados ao domínio do problema
- Não inclua casos de uso, atores ou relacionamentos desnecessários no diagrama
- Faça a especificação de casos de uso apenas com todas as informações necessárias
- Não inclua conteúdo desnecessário no fluxo principal, fluxo alternativo e regras de negócio

DC3 – “Diga de forma lógica”: Organize as informações de forma coerente

- Organize os atores com os casos de uso de forma que estes sejam visualizados da melhor maneira
- Use os relacionamentos adequados para representar uma melhor visualização das informações

DC4 – “Diga de forma clara”: Organize as informações de forma clara na modelagem e especificação de casos de uso

- Use termos claros e evite ambiguidade
- Use as referências que facilitem o entendimento dos fluxos alternativos no fluxo principal da especificação de casos de uso
- Evite a interpretação implícita

- Diagrama de Classe da UML - [Disponível aqui](#)

Diretivas de Comunicabilidade



Diagrama de Classes da UML

DC1 – “Diga a verdade!”: Use informações verdadeiras na modelagem do diagrama de classes

- Não insira classes no diagrama que estejam fora do domínio do problema
- Remova as informações desatualizadas

DC2 – “Diga o que é necessário e não mais que o necessário”: Use o conteúdo necessário na modelagem do diagrama de classes

- Insira todas as classes do domínio do problema no diagrama
- Insira todos os relacionamentos necessários entre as classes
- Não use conteúdo desnecessário no diagrama

DC3 – “Diga de forma lógica”: Organize as informações de forma coerente

- Organize as classes de forma que estas sejam visualizadas da melhor maneira
- Organize os relacionamentos de forma que estes sejam visualizadas da melhor maneira entre as classes

DC4 – “Diga de forma clara”: Organize as informações de forma clara na modelagem do diagrama de classes

- Evite termos nos nomes das classes ou atributos que causem ambiguidade
- Evite interpretação implícita

- Protótipos - [Disponível aqui](#)

Diretivas de Comunicabilidade



Mockups e Protótipos

DC1 – “Diga a verdade!”: Use informações verdadeiras na representação da interface

- Não use informações que estejam fora do domínio do problema
 - *As informações nos protótipos refletem as informações que devem ser desenvolvidas? Em caso negativo, insira apenas informações que estejam relacionadas ao sistema que será desenvolvido.*
- Não use informações desatualizadas
 - *As regras para o desenvolvimento do sistema mudaram? Em caso positivo, atualize as informações nos mockups e protótipos*

DC2 – “Diga o que é necessário e não mais que o necessário”: Use o conteúdo necessário na representação da interface

- Insira todo o conteúdo necessário
 - *A quantidade de informação nos protótipos é suficiente para o entendimento da equipe sobre o sistema? Em caso negativo, insira a quantidade de informação necessária para apoiar o melhor entendimento do sistema*
- Não use conteúdo desnecessário no modelo
 - *Existe o excesso de informação? Em caso positivo, se esse excesso for desnecessário para o entendimento do sistema, retire-o dos protótipos*

DC3 – “Diga de forma lógica”: Organize as informações de forma coerente

- Organize as telas de forma que estes sejam compreendidos da melhor maneira
 - *A ordem das telas está organizada de maneira que estas sejam melhor compreendidas pela equipe? Em caso negativo, organize a sequência das telas de modo que estas sejam melhor compreendidas pela equipe*

DC4 – “Diga de forma clara”: Organize as informações de forma clara para a representação da interface

- Use termos claros
 - *São utilizados termos conhecidos pela equipe? Em caso negativo, use termos que sejam conhecidos ou forneça um dicionário para o entendimento de tais termos*
- Evite interpretação implícita por parte da equipe
 - *São utilizadas informações com o objetivo de obter interpretação implícita pela equipe? Em caso negativo, reflita se tais informações devem ser expressas de forma explícita para evitar múltiplas interpretações*

APÊNDICE D – Questionários para Avaliação dos Fatores que Impactam na Comunicação via Artefato

Questionário para a avaliação de artefatos de software

<Nome do Projeto>

Através das questões abaixo você, projetista/designer de software, obterá a percepção dos artefatos de software como meio e comunicação. Responda as questões de cada item e veja se o modelo de software poderá auxiliar a equipe sobre o entendimento do software.

1. Em relação ao **artefato de software como meio de comunicação**, qual o seu grau de concordância em relação à:

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
M1. Acredito que seja necessário este artefato para apoiar a equipe no entendimento do domínio do problema.							
M2. Esse artefato possui elementos em sua notação para representar a melhor forma de comunicar a equipe sobre o domínio do problema.							
M3. Este artefato possui os elementos em sua notação que eu pretendo utilizar para falar com a equipe sobre, por exemplo, a documentação ou informação para a implementação do software.							
M4. Este artefato me auxilia no que eu gostaria de “falar” para a equipe sobre o domínio do problema.							
M5. Este artefato permite que o meu projeto para o software seja representado de forma clara para equipe.							
M6. Este é o artefato mais adequado para a equipe entender esta perspectiva do software que estou criando (interação, estrutural, comportamental e etc.).							
M7. Este artefato é suficiente para representar tal perspectiva para o desenvolvimento do software.							

2. Em relação ao **conteúdo comunicativo do artefato de software**, qual o seu grau de concordância em relação à:

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
Cont1. Acredito que os elementos de modelagem que estou utilizando são compreensíveis para a equipe.							
Cont2. Não foram introduzidas informações desorganizadas no artefato.							
Cont3. Não foi introduzido o excesso de informações no artefato.							
Cont4. A quantidade de informações no artefato é suficiente para o entendimento da equipe.							
Cont5. Acredito que o conteúdo do artefato está claro e é de fácil compreensão.							

Caso a equipe de software possa responder um questionário, aplique o questionário “Avaliação da Percepção dos Consumidores do Artefato”.

	Concordo Totalmente	Concordo Amplamente	Concordo Parcialmente	Não Concordo e Nem Discordo	Discordo Parcialmente	Discordo Amplamente	Discordo Totalmente
PC1. Eu sou capaz de reconhecer o conteúdo informativo representado neste artefato.							
PC2. Eu acho fácil compreender este artefato.							
PC3. Eu irei adotar esse artefato para compreender as atividades de desenvolvimento							
PC4. Eu compreenderei melhor o software com este artefato.							
PC5. Eu acho fácil usar este artefato.							

APÊNDICE E – Dados Analisados nos Estudos

ESTUDO 3: PRODUÇÃO DE ARTEFATOS COM AS DCS

- **Análise dos Mockups**

Grupo Experimental - “O que o desenvolvedor sabe sobre o usuário” (ver Tabela 9.1), “A intenção e expectativa do desenvolvedor sobre” (ver Tabela 9.2) e “Apoio e suporte do desenvolvedor para” (ver Tabela 9.3).

Grupo de Controle - “O que o desenvolvedor sabe sobre o usuário” (ver Tabela 9.4), “A intenção e expectativa do desenvolvedor sobre” (ver Tabela 9.5) e “Apoio e suporte do desenvolvedor para” (ver Tabela 9.6).

#P	SFYI Message
O que o desenvolvedor sabe sobre o usuário...	
P1	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P2	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P3	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P4	Sabe que o usuário pode cadastrar um plano de gastos. Não foram representadas as opções para: <u>cadastrar contas, cartões e despesas</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. <u>O usuário também poderá alterar esse plano de gasto somente se o plano tiver ultrapassado 80% do limite de gastos.</u>
P5	Sabe que o usuário pode cadastrar um plano de gastos, cartões e despesas. Não foi representada a opção para: <u>cadastrar contas</u> . <u>O usuário não é informado pelo sistema quando mais de 80% do limite de gastos for atingido.</u>
P6	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P7	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. <u>O usuário também poderá alterar esse plano de gasto somente se o plano tiver ultrapassado 80% do limite de gastos.</u>
P8	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P9	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P10	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P11	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P12	Sabe que o usuário pode cadastrar um plano de gastos, cartões, despesas e <u>históricos</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. <u>O usuário também poderá alterar esse plano de gasto somente se o plano tiver ultrapassado 80% do limite de gastos.</u>
P13	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P14	Sabe que o usuário pode cadastrar um plano de gastos, cartões, contas e despesas. Não foi representada a opção para: <u>cadastrar cartões</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P15	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. <u>As despesas por categorias não foram representadas</u> . <u>O usuário não é informado pelo sistema quando mais de 80% do limite de gastos for atingido.</u>
P16	Sabe que o usuário pode cadastrar um plano de gastos, contas e despesas. Não foi representada a opção para: <u>cadastrar cartões</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.

Tabela 9.1: Análise dos Mockups com a Ferramenta SFYI Message – Parte 1 (Grupo Experimental).

#P	SFYI Message
A intenção e expectativa do desenvolvedor sobre...	
P1	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. O usuário também pode acompanhar os gastos.
P2	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento e acompanhamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P3	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento e acompanhamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P4	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição do extrato das despesas da conta, o status de gastos realizados pelo usuário, o saldo da conta e um botão para visualizar a notificação do limite de gastos realizados. <u>Não foi representado os gastos por categoria. O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos.</u> Caso o planejamento não ultrapasse 80% dos gastos, o botão para visualizar detalhes do planejamento de gastos fica desabilitado.
P5	Projetar mockups para que o usuário acesse o sistema e logo seja exibida os gastos recentes, a categoria de gastos, o acompanhamento dos gastos e o planejamento de gastos. Na tela de acompanhamento, é exibido o quanto o usuário já gastou e o quanto ele ainda pode gastar. O usuário poderá alterar o planejamento, porém, <u>o usuário não é informado caso este ultrapasse 80% dos gastos.</u>
P6	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento e acompanhamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P7	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição das despesas da conta e cartão, além do detalhamento das despesas e de seu acompanhamento. <u>Não foi representado os gastos por categoria. O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos.</u>
P8	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento e acompanhamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P9	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com as opções de acompanhamento dos gastos e o planejamento de gastos. <u>Não foi representado os gastos por categoria.</u> Na tela estatística, é exibida o quanto o usuário já gastou e o quanto ele ainda pode gastar, incluindo a lista de gastos. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P10	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento e acompanhamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P11	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento e acompanhamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P12	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com as opções de cartões e contas, acompanhamento, vencimentos e históricos. <u>Não foi representado os gastos por categoria.</u> Na tela metas, é exibida o quanto o usuário já gastou a partir da lista de gastos. <u>O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos.</u>
P13	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento e acompanhamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P14	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com o acompanhamento dos gastos e o planejamento de gastos. <u>Não foi representado os gastos por categoria.</u> Na tela de acompanhamento, é exibida o quanto o usuário já gastou e o quanto ele ainda pode gastar, incluindo a lista de gastos. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P15	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com o acompanhamento dos gastos e o planejamento de gastos. <u>Não foi representado os gastos por categoria.</u> Na tela de acompanhamento, é exibida o quanto o usuário já gastou e o quanto ele ainda pode gastar. <u>O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos.</u>
P16	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com informações sobre o saldo atual e o quanto o usuário ainda pode gastar, gastos por categoria. Além disso, são exibidas as opções de cadastrar novo planejamento, cadastrar e visualizar despesas (acompanhamento) e cadastrar conta. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.

Tabela 9.2: Análise dos Mockups com a Ferramenta SFYI Message – Parte 2 (Grupo Experimental).

#P	SFYI Message
Apoio e suporte do desenvolvedor para...	
P1	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P2	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P3	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P4	<u>Não foram projetadas alternativas para o usuário.</u>
P5	<u>Não foram projetadas alternativas para o usuário.</u>
P6	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P7	<u>Não foram projetadas alternativas para o usuário.</u>
P8	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P9	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P10	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P11	<u>É possível que o usuário altere apenas os dados do planejamento.</u>
P12	<u>Não foram projetadas alternativas para o usuário.</u>
P13	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P14	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P15	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P16	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.

Tabela 9.3: Análise dos Mockups com a Ferramenta SFYI Message – Parte 3 (Grupo Experimental).

#P	SFYI Message
O que o desenvolvedor sabe sobre o usuário...	
P17	Sabe que o usuário pode cadastrar um plano de gastos, contas e cartões. Não foi representada a opção para: <u>o cadastro de despesas</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P18	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e categorias de despesas. Não foi representada a opção para: <u>o cadastro de despesas</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. <u>O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos</u> .
P19	Sabe que o usuário pode cadastrar um plano de gastos e despesas. Não foram representadas as opções para: <u>o cadastro de contas, cartões e categorias de despesas</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando este atinge 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P20	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P21	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. Não foram representadas as opções para: <u>categorias de despesas</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando este <u>atinge 75% e 80%</u> do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P22	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões, despesas e as categorias. Foram representadas as opções para: <u>consultar multas e consultar pendências</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P23	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P24	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P25	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P26	Sabe que o usuário pode cadastrar um plano de gastos, cartões e despesas. <u>Não foi representada a descrição das contas e as categorias dos gastos</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. <u>O usuário só poderá alterar o planejamento caso este ultrapasse 80% dos gastos</u> .
P27	Sabe que o usuário pode cadastrar um plano de gastos e categorias relacionadas às despesas. Não foram representadas as opções para: <u>o cadastro de contas, cartões e despesas específicas</u> . A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando este atinge 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P28	Sabe que o usuário pode cadastrar um plano de gastos, contas, cartões e despesas. A partir desse plano cadastrado, o usuário pode ser informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário também poderá alterar esse plano de gasto, caso este queira.
P29	Sabe que o usuário pode cadastrar um plano de gastos, contas e cartões. Não foram representadas as opções para: <u>categorias de despesas</u> . <u>O usuário não é informado pelo sistema quando mais de 80% do limite de gastos for atingido. O usuário não pode alterar o plano de gasto</u> .
P30	Sabe que o usuário pode cadastrar um plano de gastos, contas e cartões. Não foram representadas as opções para: <u>despesas</u> . <u>Isso ficou implícito na categoria</u> . <u>No entanto, todas as categorias são relacionadas às despesas</u> . <u>O usuário não é informado pelo sistema a porcentagem 80% do limite de gastos for atingido. Apenas é informado para o usuário que o usuário ultrapassou o limite</u> . O usuário também poderá alterar esse plano de gasto, caso este queira.

Tabela 9.4: Análise dos Mockups com a Ferramenta SFYI Message – Parte 1 (Controle).

#P	SFYI Message
A intenção e expectativa do desenvolvedor sobre...	
P17	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento de gastos para os usuários. Além disso, as opções de cadastro de cartões e contas são exibidas. <u>A opção para o cadastro de despesas não foi representada.</u> O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. usuário também pode acompanhar os gastos.
P18	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento de gastos para os usuários. Além disso, as opções de cadastro de cartões e contas são exibidas. <u>A opção para o cadastro de despesas não foi representada.</u> O usuário poderá alterar o planejamento somente se este ultrapassar 80% dos gastos. O usuário também pode acompanhar os gastos.
P19	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos, o planejamento de gastos para os usuários e acompanhamento dos gastos. <u>As opções para o cadastro de cartões, contas e categorias para as despesas não são apresentadas.</u> O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P20	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. <u>Não é possível que usuário acompanhe a evolução dos gastos.</u>
P21	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a lista de contas e cartões, plano de gastos e acompanhamento dos gastos. <u>As categorias não são apresentadas para as despesas.</u> O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. O usuário também pode cadastrar cartões e contas.
P22	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a lista de contas e cartões, plano de gastos, acompanhamento dos gastos e calendário com as datas de vencimento. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. O usuário também pode cadastrar cartões e contas. <u>As despesas e as categorias são cadastradas no plano de gastos.</u>
P23	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com as opções para verificar a descrição dos gastos e a categoria destes, o planejamento de gastos para os usuários, <u>consultar multas ou juros e consultar pendências.</u> Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. O usuário também pode acompanhar os gastos.
P24	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. O usuário também pode acompanhar os gastos.
P25	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos, a opção para acessar o planejamento de gastos para os usuários, editar conta e <u>o vencimento das faturas.</u> <u>A opção para cadastro de despesa não é exibida.</u> O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. O usuário também pode acompanhar os gastos.
P26	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com o acompanhamento dos gastos, a opção para acessar o planejamento de gastos dos usuários, editar conta e ver <u>o vencimento das faturas.</u> <u>A opção para cadastro de despesa não é exibida.</u> <u>O usuário poderá alterar o planejamento somente se este ultrapassar 80% dos gastos.</u>
P27	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com o acompanhamento dos gastos e a opção para acessar o planejamento de gastos dos usuários. <u>As opções para o cadastro de despesa, cadastro de contas e cartões não são exibidas.</u> O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não.
P28	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a descrição dos gastos e a categoria destes, o planejamento de gastos para os usuários. Além disso, as opções de cadastro de despesas, cartões e contas são exibidas. O usuário poderá alterar o planejamento caso este ultrapasse 80% dos gastos ou não. O usuário também pode acompanhar os gastos.
P29	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a lista de cartões, plano de gastos e despesas. <u>Não foi representado o acompanhamento dos gastos e contas.</u> O usuário não é informado quando o planejamento ultrapassa 80% dos gastos. <u>As categorias das despesas não foram representadas.</u>
P30	Projetar mockups para que o usuário acesse o sistema e logo seja exibida a tela com a lista de cartões, plano de gastos e saldo da conta. <u>Não foi representado o acompanhamento dos gastos e descrição das contas.</u> O usuário não é informado quando o planejamento ultrapassa 80% dos gastos e sim quando este ultrapassa o limite total. <u>Foram representadas apenas as categorias das despesas, sem a despesa específica.</u>

Tabela 9.5: Análise dos Mockups com a ferramenta SFYI Message – Parte 2 (Controle).

#P	SFYI Message
Apoio e suporte do desenvolvedor para...	
P17	<u>Não foram projetadas alternativas para o usuário.</u>
P18	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P19	<u>Não foram projetadas alternativas para o usuário.</u>
P20	<u>Não foram projetadas alternativas para o usuário.</u>
P21	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P22	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P23	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P24	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P25	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P26	<u>Não foram projetadas alternativas para o usuário.</u>
P27	<u>Não foram projetadas alternativas para o usuário.</u>
P28	É possível que o usuário altere os dados do planejamento, contas e cartões a qualquer momento.
P29	<u>Não foram projetadas alternativas para o usuário.</u>
P30	<u>Não foram projetadas alternativas para o usuário.</u>

Tabela 9.6: Análise dos Mockups com a Ferramenta SFYI Message – Parte 3 (Controle).

ESTUDO 4 e 5: ANÁLISE DO USO DAS DCS NA INDÚSTRIA

Quantidade de Respostas dos Produtores

Concordo Totalmente – 1 Concordo Amplamente - 2 Concordo Parcialmente – 3 Não Concordo e Nem Discordo – 4 Discordo Parcialmente -5 Discordo Amplamente -6 Discordo Totalmente - 7	M1	M2	M3	M4	M5	M6	M7
PRODUTOR 1 – Designer da Equipe 1	3,00	1,00	3,00	1,00	3,00	1,00	1,00
PRODUTOR 2 – Designer da Equipe 2	1,00	1,00	3,00	1,00	3,00	1,00	1,00
PRODUTOR 3 – Analista de Sistemas da Equipe 2	2,00	2,00	2,00	1,00	2,00	2,00	2,00
PRODUTOR 4 – Designer da Equipe 3	2,00	1,00	1,00	1,00	1,00	1,00	2,00
PRODUTOR 5 – Designer da Equipe 4	1,00	1,00	1,00	1,00	1,00	1,00	1,00
PRODUTOR 6 - Designer da Equipe 5	2,00	1,00	2,00	2,00	2,00	2,00	2,00
PRODUTOR 7 - Analista de Sistemas da Equipe 5	5,00	3,00	5,00	5,00	5,00	3,00	3,00

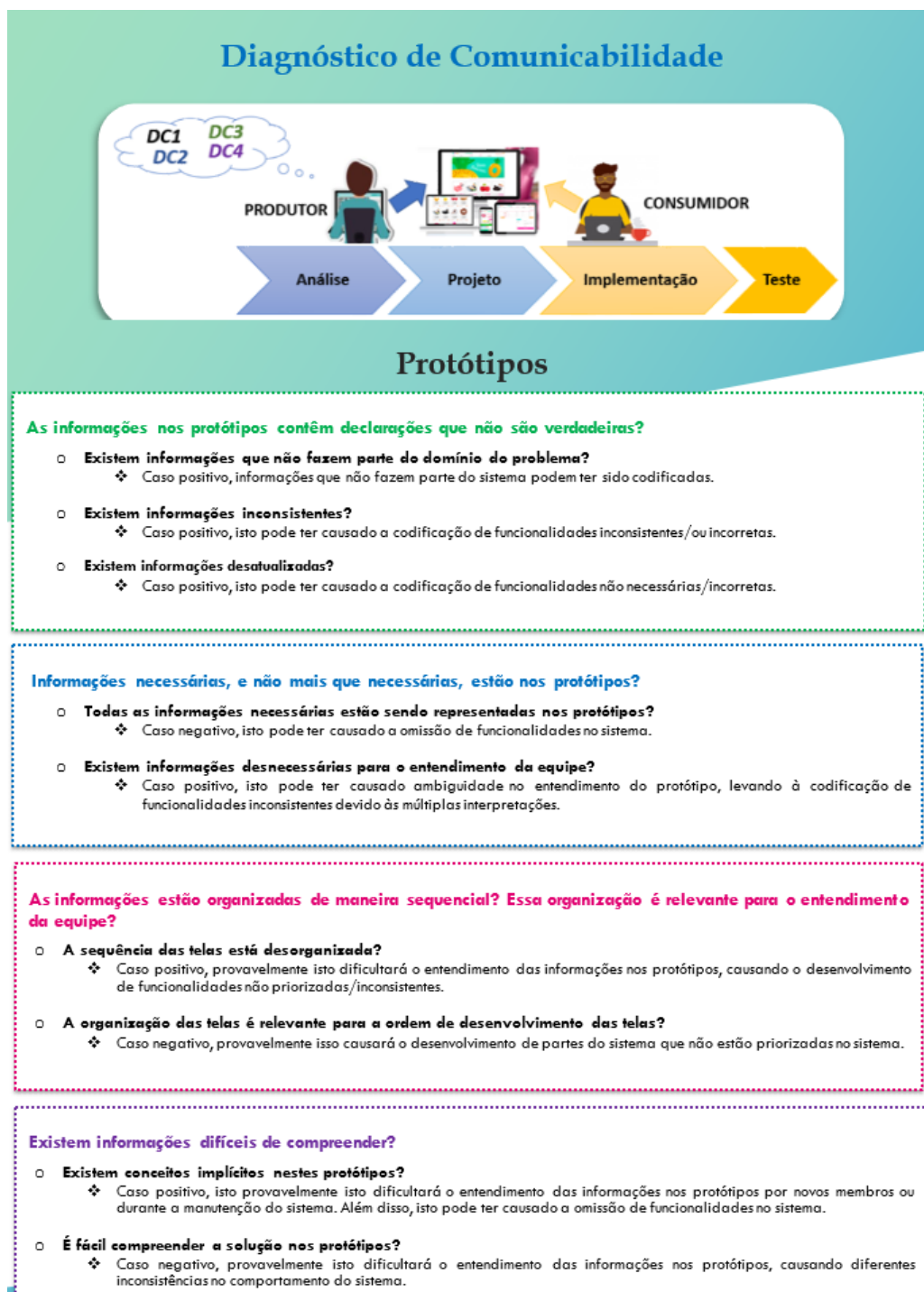
Concordo Totalmente – 1 Concordo Amplamente - 2 Concordo Parcialmente – 3 Não Concordo e Nem Discordo – 4 Discordo Parcialmente -5 Discordo Amplamente -6 Discordo Totalmente - 7	CONT1	CONT2	CONT3	CONT4	CONT5
PRODUTOR 1 – Designer da Equipe 1	1,00	1,00	3,00	1,00	3,00
PRODUTOR 2 – Designer da Equipe 2	5,00	5,00	1,00	3,00	5,00
PRODUTOR 3 – Analista de Sistemas da Equipe 2	2,00	2,00	2,00	1,00	1,00
PRODUTOR 4 – Designer da Equipe 3	2,00	1,00	1,00	2,00	1,00
PRODUTOR 5 – Designer da Equipe 4	1,00	1,00	1,00	1,00	1,00
PRODUTOR 6 - Designer da Equipe 5	2,00	2,00	2,00	2,00	2,00
PRODUTOR 7 - Analista de Sistemas da Equipe 5	1,00	1,00	1,00	1,00	2,00

Quantidade de Respostas dos Consumidores

Concordo Totalmente – 1 Concordo Amplamente - 2 Concordo Parcialmente – 3 Não Concordo e Nem Discordo – 4 Discordo Parcialmente -5 Discordo Amplamente -6 Discordo Totalmente - 7	PC1	PC2	PC3	PC4	PC5
CONSUMIDOR 1 – Desenvolvedor 1 da Equipe 1	3,00	3,00	3,00	1,00	2,00
CONSUMIDOR 2 - Desenvolvedor 2 da Equipe 1	2,00	2,00	2,00	1,00	2,00
CONSUMIDOR 3 - Desenvolvedor 3 da Equipe 1	2,00	1,00	2,00	3,00	1,00
CONSUMIDOR 4 - Desenvolvedor 1 da Equipe 2	2,00	2,00	1,00	1,00	2,00
CONSUMIDOR 5 - Desenvolvedor 2 da Equipe 2	1,00	1,00	1,00	1,00	1,00
CONSUMIDOR 6 - Desenvolvedor 3 da Equipe 2	2,00	1,00	2,00	1,00	1,00
CONSUMIDOR 7 - Desenvolvedor 1 da Equipe 3	1,00	1,00	2,00	2,00	2,00
CONSUMIDOR 8 - Desenvolvedor 2 da Equipe 3	2,00	3,00	2,00	2,00	2,00
CONSUMIDOR 9 - Desenvolvedor 3 da Equipe 3	1,00	2,00	1,00	1,00	2,00
CONSUMIDOR 10 - Desenvolvedor 4 da Equipe 3	2,00	2,00	1,00	1,00	1,00
CONSUMIDOR 11 - Desenvolvedor 1 da Equipe 4	2,00	2,00	2,00	1,00	2,00
CONSUMIDOR 12 - Desenvolvedor 2 da Equipe 4	2,00	3,00	2,00	3,00	2,00
CONSUMIDOR 13 - Desenvolvedor 3 da Equipe 4	2,00	2,00	2,00	2,00	1,00
CONSUMIDOR 14 - Desenvolvedor 4 da Equipe 4	1,00	1,00	1,00	2,00	1,00
CONSUMIDOR 15 - Desenvolvedor 5 da Equipe 4	1,00	1,00	1,00	1,00	1,00
CONSUMIDOR 16 - Desenvolvedor 1 da Equipe 5	2,00	2,00	1,00	1,00	1,00
CONSUMIDOR 17 - Desenvolvedor 2 da Equipe 5	1,00	1,00	1,00	1,00	1,00
CONSUMIDOR 18 - Desenvolvedor 3 da Equipe 5	2,00	2,00	4,00	1,00	1,00
CONSUMIDOR 19 - Desenvolvedor 4 da Equipe 5	2,00	2,00	1,00	2,00	1,00
CONSUMIDOR 20 - Desenvolvedor 5 da Equipe 5	2,00	2,00	1,00	2,00	1,00
CONSUMIDOR 21 - Desenvolvedor 6 da Equipe 5	1,00	1,00	4,00	5,00	4,00
CONSUMIDOR 22 - Desenvolvedor 7 da Equipe 5	1,00	1,00	4,00	5,00	4,00
CONSUMIDOR 23 - Desenvolvedor 9 da Equipe 5	1,00	2,00	2,00	2,00	1,00
CONSUMIDOR 24 - Desenvolvedor 9 da Equipe 5	2,00	2,00	3,00	2,00	4,00
CONSUMIDOR 25 - Desenvolvedor 10 da Equipe 5	3,00	2,00	5,00	4,00	6,00
CONSUMIDOR 26 - Desenvolvedor 11 da Equipe 5	2,00	2,00	2,00	1,00	3,00
CONSUMIDOR 27 - Desenvolvedor 12 da Equipe 5	1,00	1,00	3,00	3,00	4,00

APÊNDICE F – Diretivas de Comunicabilidade Adaptadas para o Diagnóstico de Comunicabilidade de Artefatos

[Disponível aqui](#)



[Disponível aqui](#)

