



UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE
CENTRO DE CIÊNCIAS EXATAS E DA TERRA
PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO
MESTRADO ACADÊMICO EM SISTEMAS E COMPUTAÇÃO



Visualização da Rastreabilidade em Projetos Ágeis através de Dados contidos em Ferramentas de Apoio à Gerência de Projetos

Gabriela Oliveira da Trindade

Natal-RN

Julho 2018

Gabriela Oliveira da Trindade

**Visualização da Rastreabilidade em Projetos Ágeis
através de Dados contidos em Ferramentas de Apoio à
Gerência de Projetos**

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Sistemas e Computação do Departamento de Informática e Matemática Aplicada da Universidade Federal do Rio Grande do Norte como requisito parcial para a obtenção do grau de Mestre em Sistemas e Computação.

Linha de pesquisa:
Engenharia de Software

Orientadora

Profa. Dra. Marcia Jacyntha Nunes Rodrigues Lucena

PPGSC – PROGRAMA DE PÓS-GRADUAÇÃO EM SISTEMAS E COMPUTAÇÃO
DIMAP – DEPARTAMENTO DE INFORMÁTICA E MATEMÁTICA APLICADA
CCET – CENTRO DE CIÊNCIAS EXATAS E DA TERRA
UFRN – UNIVERSIDADE FEDERAL DO RIO GRANDE DO NORTE

Natal-RN

Julho 2017 

Agradecimentos

Agradeço a Deus. Agradeço aos meus pais por sempre estarem presentes, pelo apoio e amor que sempre me deram, me ajudando a conseguir realizar este sonho. Aos meus irmãos e suas famílias, pelos seus ensinamentos e por todo o carinho. Agradeço ao meu namorado, marido, Emmerson Siqueira, pelas conversas, incentivos, amor e apoio. A toda minha família, agradeço a todos pela compreensão nos momentos que precisei me ausentar.

Agradeço a minha orientadora, Márcia Lucena, pelos conhecimentos compartilhados, pela amizade e carinho que sempre demonstrou ter.

Agradeço ao meu grupo LETS, em especial a Erica Cunha e João Carlos, pelas conversas, pelos momentos, sorrisos e por toda ajuda que me deram.

Agradeço a todas minhas amizadas.

Visualização da Rastreabilidade em Projetos Ágeis através de Dados contidos em Ferramentas de Apoio à Gerência de Projetos

Autor: Gabriela Oliveira da Trindade

Orientador(a): Profa. Dra. Marcia Jacyntha Nunes Rodrigues Lucena

RESUMO

A rastreabilidade de *software*, conhecida pelo relacionamento entre quaisquer artefatos da engenharia de *software*, traz grandes vantagens no processo de desenvolvimento. As informações que ela fornece ajudam em tomadas de decisão diante de uma alteração, melhor entendimento do artefato, reusabilidade, manutenção, previsão de custos e prazos, entre outros. Com ambientes cada vez mais adeptos às metodologias ágeis, com o cliente lado a lado dando *feedbacks* constantes, a adequação a essas mudanças solicitadas tem sido uma prática comum durante o desenvolvimento de sistemas. E, para que as mudanças sejam feitas de forma segura, informações de rastreabilidade ajudam em decisões, com o objetivo de que a alteração não traga inconsistências, introdução de erros e geração de falhas no sistema.

Algumas ferramentas de gerenciamento de projetos suportam elementos de rastreabilidade. Contudo com diversos dados que podem ser fornecidas com tal prática, fica difícil interpretá-los, principalmente quando são apresentados apenas textualmente. Sabendo que a visualização de informação traz possibilidade de uma análise com grandes volumes de dados de forma rápida e com mais clareza, oferecendo uma tomada de decisão mais segura e, permitindo descobrir informações antes não vistas, é possível verificar na literatura técnicas de visualização de informação de rastreabilidade. No entanto, tais técnicas necessitam de informações além desses dados, precisam considerar os pilares da informação expostos na academia (problemática, o que, quando e quem visualizar) para se ter uma visualização adequada.

Com este intuito, este trabalho realiza entrevistas na indústria para responder aos pilares da informação considerados na proposta de uma visualização. Em seguida, uma

análise baseada em Teoria Fundamentada é feita nos dados levantados. Então, no contexto montado de rastreabilidade, perfis definidos, necessidades e  problemas descritos e, artefatos gerados em ambientes ágeis, é estudada na bibliografia as visualizações de informação existentes.

Como resultado é realizada uma discussão e sugestão de visualização apropriada para informações de rastreabilidade com base nas sugestões da literatura e dos dados levantados da entrevista. Posteriormente, com Heurísticas criadas, é feita uma avaliação das ferramentas de gerenciamento de projeto que se integram a plataforma de hospedagem e versionamento de dados Github, para averiguar se fornecem a visualização de informação de rastreabilidade notada. 

Palavras-chave: Rastreabilidade de Software, Visualização da Rastreabilidade, Metodologias Ágeis.

Visualization of Traceability in Agile Projects through Data contained in Tools of Support to the Management of Projects

Author: Gabriela Oliveira da Trindade

Supervisor: Profa. Dra. Marcia Jacyntha Nunes Rodrigues Lucena

ABSTRACT

Software traceability, known for the relationship between any software engineering artifacts, brings great advantages to the development process. The information it provides helps in decision making in the face of a change, better understanding of the artifact, reusability, maintenance, forecasting of costs and deadlines, among others. With environments increasingly adept at agile methodologies, with client side-by-side giving constant feedbacks, a adequacy to these requested changes has been a common practice during system development. And, in order for changes to be made safely, traceability information helps in making decisions, with a goal in which the change does not bring inconsistencies, introduce errors, and generate system failures

Some project management tools support traceability elements. However with several data that can be provided with such a practice, it is difficult to interpret them, especially when they are presented only textually. Knowing that the visualization of information brings the possibility of an analysis with large volumes of data in a fast and clear way, offering a safer decision making and, allowing to discover information previously unseen, it is possible to verify in the literature techniques of visualization of information of traceability. However, such techniques require information in addition to these data, need to consider the pillars of information exposed in the academy (problematic, what, when and who to view) to have an adequate visualization.

With this purpose, this work performs interviews in the industry to respond to the pillars of information considered in the proposal of a visualization. Then, an analysis based on Grounded Theory is done on the data collected. Then, in the assembled context

of traceability, defined profiles, needs and problems described, and artifacts generated in agile environments, the existing information visualizations are studied in the bibliography.

As a result, a discussion and suggestion of appropriate visualization for traceability information is made based on the suggestions in the literature and data collected from the interview. Later, with Heuristics created, an evaluation of the project management tools that integrate the platform of hosting and versioning of data Github is made, to see if they provide the noticed visualization of traceability information.

Keywords: Software Traceability, Traceability Visualization, Agile Methodologies.

Lista de figuras

1	Linha do tempo da criação das metodologias ágeis. Adaptado de Bassi (2008)	p. 22
2	Exemplo de Rastreabilidade de Software	p. 28
3	Pré, Inter e Pós-rastreabilidade de requisitos. Fonte: Adaptado de Genvigir (2009) e Gotel (1995)	p. 28
4	Rastreabilidade Horizontal e Vertical. Fonte: Genvigir (2009) e Gotel (1995)	p. 29
5	Processo de Codificação e criação de memorando realizado na aplicação da Teoria Fundamentada	p. 56
6	Processo de análise das entrevistas	p. 58
7	Pilares aliciadores da informação que podem impactar na sugestão da visualização da informação da rastreabilidade.	p. 74
8	Exemplo da técnica de visualização Matriz de Rastreabilidade. Fonte: Pohl e Rupp (2012).	p. 81
9	Exemplo da Técnica  Visualização Matriz de Rastreabilidade sugerida por Franceschi e Duarte (2011). Fonte: Franceschi e Duarte (2011). . .	p. 81
10	Exemplo da técnica de visualização Árvore. Fonte: Herman, Melançon e Marshall (2000)	p. 82
11	Exemplo da técnica Alternativa (1) de visualização de Árvore. Fonte: Holten (2006)	p. 82
12	Exemplo da técnica Alternativa (2) de visualização de Árvore. Fonte: Chen, Hosking e Grundy (2012)	p. 83
13	Exemplo da transformação de Árvore em Treemap. Fonte: Wijk e Wetering (1999)	p. 84
14	Exemplo da técnica de Visualização Treemap. Fonte: Wijk e Wetering (1999)	p. 84

15	Exemplo de técnica Alternativa (1) de visualização do Treemap. Fonte: Holten (2006).	p. 85
16	Exemplo de técnica Alternativa (2) de visualização do Treemap. Fonte: Chen, Hosking e Grundy (2012).	p. 86
17	Exemplo da técnica de Visualização Radial Tree (a) e Balloon Tree (b). Fonte: Holten (2006).	p. 87
18	Exemplo da técnica de Visualização Radial Tree (a) e Balloon Tree (b). Fonte: Herman, Melançon e Marshall (2000).	p. 87
19	Exemplo da técnica de Visualização Sunburst. Fonte: Merten, Jüppner e Delater (2011).	p. 88
20	Exemplo da técnica de Visualização Netmap. Fonte: Merten, Jüppner e Delater (2011).	p. 89
21	Exemplo de grafo gerado no Neo4j a partir de dados do Github.	p. 95
22	Exemplo de filtragem do por usuário no grafo gerado no Neo4j a partir de dados do Github.	p. 96
23	Exemplo de filtragem do por issue no grafo gerado no Neo4j a partir de dados do Github.	p. 96
24	Suporte à criação de links de rastreabilidade nas ferramentas.	p. 107
25	Dados de artefato e sua forma de armazenamento nas ferramentas.	p. 108
26	Apoio de visualizações para a tomada de decisão para planejamento e controle dos artefatos	p. 111
27	Auxílio à manipulação e à naveabilidade nas informações de rastreabilidade	p. 112
28	Apoio na redução de carga cognitiva	p. 113

Listas de tabelas

1	Tipos de Rastreabilidade	p. 27
2	Técnicas comuns de Visualização de Rastros	p. 33
3	As três gerações dos sistemas de controle de versão. Retirado de Sink (2011).	p. 39
4	 Relação entre Perguntas da Entrevista na seção de Metodologia de Desenvolvimento e Gerenciamento de requisitos com Perguntas levantadas por Gotel e Finkelstein (1997)	p. 51
5	Algumas Questões de Pesquisa presentes no Roteiro da Entrevista . . .	p. 52
6	Relação de categorias e heurísticas	p. 70
7	Respostas aos pilares da informação	p. 78
8	Relação de categorias e heurísticas	p. 99
10	Heurísticas e questões de pesquisa relacionadas à categoria de Visualização da Rastreabilidade	p. 101
9	Heurísticas e questões de pesquisa relacionadas à categoria de Rastreabilidade	p. 104
11	Comparação das fases, práticas, papéis e artefatos de cada processo e metodologia ágil	p. 129

Sumário

1 Introdução	p. 12
1.1 Motivação	p. 14
1.2 Apresentação do Problema e Questões de Pesquisa	p. 15
1.3 Objetivo Geral e Específicos	p. 16
1.4 Metodologias	p. 17
1.5 Organização do Trabalho	p. 18
2 Fundamentação Teórica	p. 20
2.1 Métodos Ágeis	p. 21
2.2 Rastreabilidade em Métodos Ágeis	p. 26
2.3 Visualização da Rastreabilidade	p. 31
2.4 Sistema de Controle de Versão	p. 38
2.5 Trabalhos Relacionados	p. 41
3 Entrevista e suas Análises baseadas em Teoria Fundamentada	p. 46
3.1 Participantes	p. 54
3.2 Análises	p. 55
3.3 Resultados	p. 58
3.4 Discussão	p. 68
3.4.0.1 Ameaças à validade e as limitações do estudo	p. 72
3.5 Considerações	p. 72
4 Visualização da Informação da Rastreabilidade	p. 73

4.1	Propostas de Visualização da Literatura considerando os Pilares da Informação	p. 74
4.1.0.1	Pilares Aliciadores da Informação	p. 75
4.1.0.2	Propostas de Visualização	p. 79
4.2	Recomendação de Visualização	p. 90
4.3	Considerações	p. 94
5	Análise de Ferramentas de Gerenciamento de Projetos	p. 98
5.1	Metodologia e critérios para a seleção e avaliação das ferramentas	p. 99
5.2	Aplicação das Heurísticas	p. 100
5.3	Análise dos Resultados da Aplicação das Heurísticas	p. 105
5.4	Considerações	p. 114
6	Considerações finais	p. 116
6.1	Principais contribuições da pesquisa	p. 117
6.2	Limitações	p. 118
6.3	Trabalhos futuros	p. 118
6.4	Conclusão Final	p. 119
Referências		p. 120
Apêndice A – Processos e Metodologias Ágeis		p. 128
Apêndice B – Roteiro para Entrevista Presencial/Online		p. 132
Apêndice C – Roteiro para entrevista por <i>e-mail</i>		p. 135
Apêndice D – Modelos de <i>e-mails</i> para Entrevista		p. 136

1 Introdução

A rastreabilidade de software, segundo Cysneiros e Zisman (2017) e Gotel et al. (2012), pode ser definida como o relacionamento entre quaisquer artefatos da Engenharia de Software. Apesar de diversas definições tratarem a rastreabilidade com foco em requisitos, como por exemplo: "habilidade de acompanhar e descrever a vida de um requisito" (GOTEL; FINKELSTEIN, 1994); uma ampliação dessas definições foi dada em COEST¹ (*Center of Excellence for Software and Systems Traceability*).

Nesta ampliação, a rastreabilidade de *software* (RS) passa a "abrir a rastreabilidade multidirecional centrada em diversos artefatos" (COEST, 2018). Portanto, desde então, este termo pode exceder a rastreabilidade de requisitos para considerar as ligações entre os próprios requisitos ou dos requisitos a artefatos. Dito isto, este trabalho considerou de forma mais ampla essa definição de RS, para abranger, e interrelacionar qualquer artefato da Engenharia de Software a quaisquer outros tipo de artefato (CYSNEIROS; ZISMAN, 2017; GOTEL et al., 2012). Sendo o próprio requisito um artefato para esta pesquisa.

A razão para isso é fundamentada pelo fato dos relacionamentos gerarem informações, que ajudam na análise de impactos, acompanhamento da evolução do artefato e suas dependências, segundo estes mesmos autores (CYSNEIROS; ZISMAN, 2017; GOTEL et al., 2012). Além disso, a rastreabilidade pode facilitar atividades como manutenção do sistema, compreensão dos requisitos e mudanças ocorridas (POHL; RUPP, 2012). Por outro lado, a diversidade¹ volume de informações disponíveis podem dificultar a assimilação, distorcer os fatos e, por conseguinte, obstaculizar o melhor julgamento para tomada de decisão.

Diversidade e um grande volume de informações são bem comuns em ambientes, que adotam metodologias de desenvolvimento ágil. Ressalta-se que, para Franceschi e Duarte (2011), as metodologias de desenvolvimento ágil têm se tornado bastante populares,

¹<http://www.coest.org/>

sendo uma proposta alternativa às metodologias tradicionais. Sabendo ainda que o processo ágil "abraça mudanças", e que essas atividades podem ajudar nas alterações do *software*, além dos já citados, diferentes desafios são encontrados ao aplicar tal rastreabilidade em métodos ágeis (MA) (ANTONINO et al., 2014; QUSEF, 2013; GHAZARIAN, 2008, 2010; KHAN; ARSHAD, 2012; TAROMIRAD; PAIGE, 2012; LEE; GUADAGNO; JIA, 2003; URQUIZA-YLLESCAS; MARTINEZ-MARTINEZ, 2011; DUC, 2013). Isto ocorre principalmente pela rastreabilidade ser considerada como uma atividade "pesada", enquanto MA focam em atividades "leves", que permitam a agilidade no processo.

Alves e Alves (2009) afirmaram que um profissional com domínio do negócio e do sistema poderia mapear o impacto da mudança. Entretanto, estes autores defendem que pode haver a necessidade de uma rastreabilidade mínima, por exemplo, com o objetivo de relacionar o requisito ao seu fornecedor; pois, para eles: "*Como garantir que haverá uma percepção profissional do impacto sem o apoio de uma matriz de rastreabilidade?*". Julga-se essa apenas uma forma de encaminhar a rastreabilidade.

Estendendo o entendimento contido em (TRINDADE; LUCENA, 2016) para este contexto, os artefatos estão em constante mudança, contudo muitas das modificações são analisadas através de um conhecimento tácito que a equipe adquiriu do *software* e realizadas de forma rápida, com a documentação mínima exigida pelos métodos ágeis. Estas mudanças podem invalidar o sistema, para que isto não aconteça é importante o gerenciamento e rastreamentos dessas modificações, permitindo que uma análise mais profunda dos impactos a serem gerados pela mudança seja feita.

Para Espinoza e Garbajosa (2011), a falta de documentação adequada implica na rastreabilidade de requisitos (neste contexto, artefato), o que pode levar a sérios problemas para outras atividades  como na gestão de mudança, na tomada de decisão e na estimativa  ? onde uma visualização de dependências entre histórias e tarefas pode mostrar o que será alterado e uma estimativa do tempo pode ser melhor definida, por exemplo. Além disso, a rastreabilidade pode facilitar a manutenção do sistema, compreensão dos requisitos e mudanças ocorridas (POHL; RUPP, 2012).

Segundo Lucia e Qusef (2010), os projetos ágeis funcionariam melhor se incluíssem ferramentas de rastreabilidade juntamente com ferramentas de validação. As vantagens da rastreabilidade podem prover a qualidade do sistema (TRINDADE; LUCENA, 2016), com  podemos citar os seguintes padrões de qualidade que fazem referências às práticas de rastreabilidade: CMMI, ISO/IEC-15504 e IEEE-830; além das normas militares MIL-STD-2167A e MIL-STD-498 (GENVIGIR, 2009). Entretanto, como citado anteriormente,

ainda são encontrados desafios quanto a sua aplicabilidade em ambientes ágeis.

1.1 Motivação

Uma entrevista realizada por Jacobsson (2009) aponta os motivos pelos quais os agilistas não adotam a rastreabilidade, entre estes estão os fatos de que 'ninguém lê' e a 'falta de motivação'. Assim, por exemplo, documentos detalhados que descrevem a especificação e o projeto arquitetônico "nunca" são lidos ou consultados diariamente, e a rastreabilidade dos artefatos não é praticada pela 'falta de motivação' dos profissionais, por desconhecerem claramente seus benefícios.

Para amenizar um esforço a mais da equipe com ações relacionadas à promoção da rastreabilidade, que é vista como mais uma documentação a ser feita, o uso da visualização de informações poderá trazer benefícios, como sugerido por Thommazo et al. (2012), Hernandes (2009) e Gershon, Eick e Card (1998). Para estes autores, interfaces visualmente eficazes permitem a interação rápida com grandes volumes de dados, possibilitando a descoberta de características e padrões. Gershon, Eick e Card (1998) e Thommazo et al. (2012) entenderam a visualização como um processo de transformação de dados, informações e conhecimentos em forma visual, permitindo a capacidade visual natural dos seres humanos através de uma interface.

É importante lembrar, entretanto que para obter uma visualização precisamos de dados. Sabendo do uso crescente de plataformas para hospedagem de dados e controle de versão de diversos tipos de artefatos produzidos no processo de desenvolvimento de *softwares*, estas podem ser uma grande fonte de informações. Repositórios criados dentro dessas plataformas mantêm os artefatos e dados relacionados ao desenvolvimento de sistema, podendo estes serem atualizados a qualquer momento.

Cada repositório de código possui características próprias, mas todos encontram pelo menos uma estrutura mínima e mecanismos de visualização disponíveis para gerenciar e minerar informações relativas: a tempo, pendências, e recursos, como contribuidores e colaboradores do projeto; bem como o versionamento e a distribuição do código, dentre outras funcionalidades. Ainda assim, não é comum ter um mapeamento sobre informações antecedentes à criação dos requisitos, ou até mesmo visualizações adequadas para obtenção de informações relacionadas a rastros posteriores à criação do requisito, ou ainda visualizações que mostram dependências entre artefatos, principalmente quando se trata de equipes ágeis atuando nestes repositórios. 

1.2 Apresentação do Problema e Questões de Pesquisa

A visualização de rastros de informações sobre o *software*, visto o exposto, pode ajudar no entendimento de informações de rastreabilidade. Isto pode facilitar atividades, como a tomada de decisão e a análise de impacto por gerentes e líderes de equipe de desenvolvimento, por exemplo. Salientando que os ambientes ágeis buscam resolver problemas de forma eficaz e eficiente, quando uma mudança for solicitada, uma visualização poderá trazer as informações necessárias de forma rápida e mais segura, mostrando os impactos causados pela mudança, ajudando na estimativa de tempo e verificação dos envolvidos.

Existem visualizações tradicionais encontradas e discutidas na literatura recomendadas para este propósito, como por exemplo: matrizes, grafos e árvores. Todavia ainda cabe uma discussão sobre estas visualizações serem as melhores maneiras de visualizar as informações dos rastros. Sendo que Chen, Hosking e Grundy (2012) ainda acrescentaram como desafios a serem estudados dado a diversidade e o volume de dados em projetos de grandes sistemas de *software*: a escalabilidade e os problemas visuais de confusão. Trabalhos como Merten, Jüppner e Delater (2011), Voytek e Núñez (2011), Chen, Hosking e Grundy (2012) e Cysneiros e Zisman (2017) apresentam maneiras de visualizar de forma diferente às formas tradicionais.

Ademais, para melhor atendimento, julgou-se importante definir o público alvo ponderando que: i) há uma complexidade intrínseca a diversidade de papéis encontrados dentre os interessados em um projeto de software; ii) existem particularidades e responsabilidades inerentes a estes papéis; e iii) a concepção de uma visualização deve ser feita baseada em critérios. Dessa forma, o público alvo desta pesquisa foi delimitado como gerentes e líderes de projetos de *softwares*. Pressupõe-se que, ainda assim, outros papéis possam se beneficiar das possíveis contribuições produzidas.

Entendendo isto, o problema a ser investigado por esta pesquisa foi instrumentado a partir das duas questões de pesquisa (QP) a seguir:

QP1 - Quais informações da rastreabilidade de software, considerando os diferentes artefatos, são preponderantes as atividades de gestão e liderança de equipes de desenvolvimento de software presentes em ambientes ágeis?

QP2 - Quais visualizações da informação da rastreabilidade de software podem apoiar e otimizar atividades de gestão e liderança de equipes de desenvolvimento de software presentes em ambientes ágeis?

1.3 Objetivo Geral e Específicos

Pelos motivos enunciados e pela problemática apontada, o objetivo geral deste trabalho foi recomendar quais são as visualizações mais apropriadas para determinados tipos de rastros de artefatos e as informações  presentes em cada tipo. Isto, porque vislumbrou-se que a importância da rastreabilidade, independente do tipo de metodologia e processos aplicados, e os benefícios que uma visualização adequada das informações da rastreabilidade pode trazer aos ambientes ágeis de desenvolvimento de *software*.

Para atender este objetivo geral, o foco desta pesquisa foi direcionado, principalmente, para: i) os artefatos e as informações de um projeto gerados em plataformas de hospedagem de dados com controle de versão gerenciados por ferramentas; ii) as informações a serem consideradas do ponto de vista da Engenharia de Requisitos para rastreabilidade trazidas da literatura, a partir do trabalho de Gotel e Finkelstein (1997); e iii) as necessidades encontradas pela indústria. Isto está em consonância com o que foi sugerido por Spence (2014), que destacou ser necessário para a visualização a identificação de problemas, para em seguida ilustrar possíveis representações de dados.

Este objetivo geral foi pormenorizado em objetivos específicos, para que o processo de alcance deste fosse incremental e potencializado. A seguir, estes objetivos específicos são indicados:

1. Realizar entrevistas para verificar junto ao público alvo:
 - (a) Se as perguntas relacionadas à rastreabilidade encontradas na literatura ajudam/ajudariam no dia-a-dia,
 - (b) Como estas perguntas são/poderiam ser visualizadas,
 - (c) Se há novas perguntas que gostariam que fossem respondidas,
 - (d) Entender o processo utilizado para o rastreio de informações;
2. Estudar as metodologias e processos ágeis para verificar quais são os artefatos que podem ser gerados e priorizados;
3. Estudar a plataforma de hospedagem de dados com controle de versão, Github¹, no intuito de verificar o que já é oferecido e quais dados são possíveis obter a partir desta plataforma;

¹<https://github.com/>

4. Selecionar sistemas, que fazem uso de metodologia ou processo ágil, na plataforma citada, a fim de verificar a forma como a plataforma é utilizada, e quais dados são preenchidos;
5. Realizar um estudo na literatura sobre visualização de informações da rastreabilidade a partir dos resultados obtidos com a realização de entrevista com o público alvo e na bibliografia relacionada com rastreabilidade e metodologias ágeis;
6. Propor recomendações específicas a este tipo de visualização estudada, considerando os artefatos gerados nas metodologias ágeis, bem como aqueles disponibilizados na plataforma de hospedagem de dados com controle de versão escolhida;
7. Verificar, através de inspeções baseadas em heurísticas, as ferramentas de gerenciamento de projetos de software identificadas, que possibilitavam integração com o Github, se as visualizações oferecidas são eficazes ao público alvo em questão.

1.4 Metodologias

As metodologias utilizadas para alcançar os objetivos estabelecidos foram: estudo bibliográfico, entrevistas semi-estruturadas, análise baseada em teoria fundamentada, exploração e análise de ferramentas. Inicialmente, foi feito um estudo bibliográfico para fundamentar esta pesquisa. Este estudo baseou-se na investigação por material bibliográfico relevante às temáticas interessadas: rastreabilidade, metodologias ágeis, sistemas de controle de versão e visualização da informação. Nesta etapa, também foram levantados os primeiros trabalhos relacionados com o encaminhamento proposto para problemática definida.

Na sequência, foram realizadas entrevistas semi-estruturadas com diferentes perfis do ambiente de desenvolvimento. Sabendo que entrevistas são pesquisas qualitativas que possibilitam captar o ponto de vista de cada indivíduo (DENZIN; LINCOLN, 2006), o objetivo foi utilizá-la de forma semi-estruturada a fim de que discussões fossem geradas a partir das respostas dadas, além de tentar garantir que todos os entrevistados respondessem as mesmas perguntas. As questões realizadas foram abertas para aproximar-se mais da perspectiva dos entrevistados, dando a ele a liberdade de discorrer sobre o tema sugerido. Assim, foi obtido com um maior detalhamento a sua visão, verificando seu entendimento, opinião e utilização da visualização e rastreabilidade no processo de desenvolvimento.

A análise destas entrevistas foi feita baseada em uma abordagem metodológica de Te-

oria Fundamentada (TF) adaptada de Charmaz (2014), e possibilitou também a definição de heurísticas a serem utilizadas nas ações relacionadas com: i) definição de demandas e artefatos; ii) engenharia de requisitos; iii) gerenciamento de recursos; iv) participação dos *stakeholders*; v) rastreabilidade de artefatos; vi) uso de repositórios; vii) utilização de metodologias ágeis; e viii) visualização de informações da rastreabilidade.

Após este levantamento realizado pelas entrevistas, a exploração e uma análise foram realizadas na plataforma de hospedagem de dados, Github, devido ao seu grande uso na indústria, com a intenção de perceber o que esta oferece de informação sobre rastros, e ter acesso a projetos para análise de dados. Para isto, foram selecionados projetos de sistemas, que adotavam metodologias ágeis para desenvolvimento, para a partir destes obter dados sobre os artefatos do processo de desenvolvimento disponíveis. Além disto, foram verificadas quais as informações de rastros, que poderiam ser guardadas.

Em posse das informações destes estudos prévios, foi realizado outra investigação na literatura para verificar os tópicos já atendidos e os desassistidos pela academia e pela indústria. Esta investigação teve como critérios as definições e características identificadas no primeiro estudo bibliográfico, bem como às necessidades e aos anseios do público alvo emergidos nas entrevistas. Os resultados desta investigação serviram como sustentáculo para proposição de recomendações de visualização da informação da rastreabilidade, com o objetivo de promover melhorias à gestão de projetos ágeis de *softwares*.

Tendo no horizonte as recomendações feitas e as heurísticas propostas, conjecturou-se sobre a necessidade de exploração e análise das ferramentas utilizadas pelo público alvo. Isto foi feito a partir de questões decorrentes das heurísticas objetivando o atendimento das recomendações resultantes desta pesquisa. O objetivo deste estudo foi identificar a conformidade das ferramentas de gerenciamento de projetos descobertas, que permitiam a integração com Github, frente a essas questões definidas no âmbito desta pesquisa.

1.5 Organização do Trabalho

O texto reportando o trabalho concluído está organizado da seguinte forma: o Capítulo 2 apresenta a Fundamentação Teórica, sendo composta por definições e termos utilizados no decorrer deste trabalho, além dos trabalhos relacionados; o Capítulo 3 tem detalhes relacionados às entrevistas realizadas; enquanto que o Capítulo 4 traz as recomendações para visualização das informações da rastreabilidade contextualizada na gestão de projetos de software, e o caminho percorrido para isto; o Capítulo 5 é composto pela análise

das ferramentas de gerenciamento de projetos de *software* considerando as heurísticas definidas; e, por fim, no Capítulo 6, são feitas as considerações finais a partir de uma reflexão do que foi feito, das contribuições e das limitações, bem como são anunciados possíveis trabalhos futuros e a conclusão sintética desta pesquisa.

2 Fundamentação Teórica

O uso da rastreabilidade em ambientes de desenvolvimento ágil trazem diversos desafios (ANTONINO et al., 2014; QUSEF, 2013; GHAZARIAN, 2008, 2010; KHAN; ARSHAD, 2012; TAROMIRAD; PAIGE, 2012; LEE; GUADAGNO; JIA, 2003; URQUIZA-YLLESCAS; MARTINEZ-MARTINEZ, 2011; DUC, 2013). Apesar de ser considerada uma atividade "pesada", suas vantagens são tantas que podem compensar todo o trabalho em obtê-las. Dados trazidos com o uso da rastreabilidade ajuda na tomada de decisões relacionadas a mudanças, o que seria uma facilitador para gerentes e líderes inseridos em ambientes ágeis, visto que tais métodos e processos são conhecidos por se adequarem a mudanças (BECK et al., 2001a). Entretanto, com muitos dados a serem fornecidos é importante que estes sejam trabalhados e apresentados através do uso da visualização, facilitando a obtenção de informações antes não vistas.

Como podemos perceber, neste trabalho é abordado o uso da rastreabilidade de *software* através da visualização por gerentes e líderes de projetos que estão inseridos em ambientes de desenvolvimento ágil. As necessidades destes perfis em relação às informações provindas da rastreabilidade são melhor detalhadas no Capítulo 3, assim como a forma de apresentar essas informações no Capítulo 4. Aqui, são trabalhadas definições com o propósito de embasar a partir de conceitos centrais de outros autores aspectos teóricos deste trabalho.

Portanto, esta seção trata sobre: Métodos Ágeis 2.1, trazendo um pouco do seu histórico, tipos, vantagens e desvantagens; Rastreabilidade em Métodos Ágeis 2.2, onde são abordados os tipos, importância e estratégia de uso; a Visualização da Rastreabilidade 2.3, seus tipos e vantagens; Sistemas de Controle de Versão 2.4, com uma explicação de conceitos e apresentação de plataformas de hospedagem de dados e versionamentos; e por fim, apresentamos os Trabalhos Relacionados 2.5, onde são citadas as pesquisas mais correlatas ao presente trabalho.

2.1 Métodos Ágeis

Com um grande número de casos de fracasso que estavam ocorrendo na indústria de desenvolvimento de software, alguns líderes experientes faziam uso de modos de trabalho que se opunham aos principais conceitos tradicionais, onde com um tempo foram percebendo a eficiência de seus processos (BASSI, 2008). Após a aplicação destes processos em vários projetos, feitos aprimoramentos, percebeu que novas metodologias e processos estavam surgindo (BASSI, 2008). Segundo Bassi (2008), por não fazerem uso de formalidades que caracterizavam as metodologias tradicionais, e por evitarem a burocracia imposta pelo uso de uma documentação excessiva, tais metodologias ficaram conhecidas como leves, ganhando destaque nos ambientes industrial e acadêmico, entretanto gerando debates quanto à confiabilidade dos processos e qualidade do *software*.

Após muitos anos, percebendo a semelhança entre seus modos de trabalho (BASSI, 2008), o grupo formado pelos 17 líderes, intitulado de Aliança Ágil, deu origem ao Manifesto Ágil em 2001 (FOWLER; HIGHSMITH, 2001). Inicialmente o grupo discutiu suas formas de trabalho a fim de chegar em uma única metodologia (BASSI, 2008). Contudo, após alguns dias de debate perceberam que o desenvolvimento não poderia ser marcado por um único processo, devido a complexidade do desenvolvimento de *software* (BASSI, 2008). Logo, passaram a discutir as melhores maneiras de desenvolver *software*, sem tentar formar uma única metodologia, o que levou ao Manifesto Ágil criar princípios e valores baseados no bom senso de todos os 17 integrantes envolvidos, convidando aos que os liam a pensar diferente, aceitar mudanças, melhorar continuamente, adaptar-se e ter comprometimento (PUNTEL; PRASS, 2010). Assim nascia os métodos ágeis (MA), sendo considerados uma reação às formas tradicionais de desenvolvimento de *software* (JAQUEIRA, 2013), além de se tornarem uma alternativa a tais processos de desenvolvimento pesados e orientados por documentação (BECK et al., 2001a; COHEN; LINDVALL; COSTA, 2004).

É possível ver na Figura 1 todo esse trajeto descrito acima, desde a criação de metodologias até o acontecimento do Manifesto Ágil, onde criou um marco para criação dos princípios e valores seguidos pelos métodos ágeis.

Junto ao Manifesto Ágil, o método de desenvolvimento ágil também foi surgindo da crença de que a mudança é constante (LEE; GUADAGNO; JIA, 2003), sendo assim difícil obter os requisitos do sistema elicitados e estáveis antes do início da implementação (JAQUEIRA, 2013). Segundo Highsmith (2002), Cohen, Lindvall e Costa (2004), praticantes de métodos tradicionais (MT) ficavam insatisfeitos com as atividades de tais métodos,

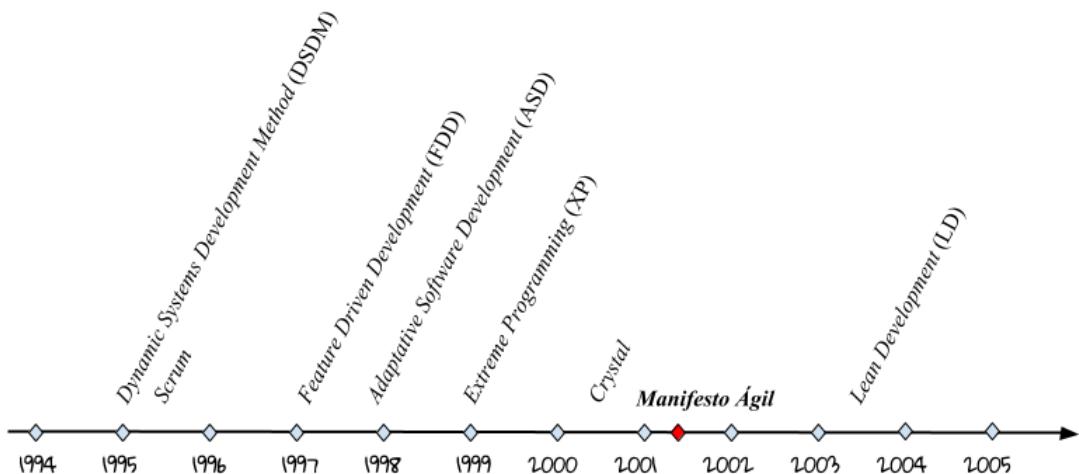


Figura 1: Linha do tempo da criação das metodologias ágeis. Adaptado de Bassi (2008).

que envolvem inicialmente a elicitação e documentação de um conjunto completo de requisitos, seguido de design arquitetônico, desenvolvimento e inspeção, chegando a pensar que talvez estas etapas fossem impossíveis.

Para Highsmith (2002) a agilidade é a "*capacidadade de criar e responder a mudanças para lucrar com um ambiente de negócios turbulentos*". Já Erickson, Lyytinen e Siau (2005) definiram a agilidade como a remoção de todo peso comumente associado às metodologias tradicionais de desenvolvimento de software para promover as rápidas respostas em ambientes em mudança. Williams e Cockburn (2003) afirmam que o desenvolvimento ágil está relacionado ao *feedback* de mudança, onde metodologias ágeis são desenvolvidas para abraçar as mudanças, ao invés de rejeitá-las. Segundo Paetsch, Eberlein e Maurer (2003) e Jaqueira (2013), MA trouxeram a minimização da formalidade e da documentação que era criada no início do desenvolvimento de qualquer projeto baseado em MT, passando então a focar mais no código e interação. Enquanto Nerur, Mahapatra e Mangalaraj (2005) afirmam que métodos ágeis lidam com mudanças, baseando-se nas pessoas, em vez de processos (COCKBURN; HIGHSMITH, 2001). Tais definições e modificações no processo são reflexos dos valores definidos no Manifesto para o desenvolvimento ágil, onde há a preocupação de mudanças no software e formalidade para sua construção. Estes valores são definidos da seguinte forma (BECK et al., 2001a):

- Indivíduos e interações acima dos processos e ferramentas;
- Software operando acima da documentação abrangente;
- Colaboração do cliente acima de negociações contratuais;

- Responder às mudanças acima de seguir um plano elaborado.

Os itens listados a esquerda de cada valor têm maior prioridade quando comparado aos itens da direita. Entretanto, isto não significa que eles não têm importância, apenas que, em MA, eles possuem uma importância secundária quando comparados com indivíduos e interações, *software* operando, colaboração do cliente e resposta às mudanças.

Apesar de ter valores e princípios diferentes entre MA e MT, há um importante aspecto que está relacionado com a Engenharia de Requisitos (ER). Para Lucia e Qusef (2010) a principal diferença na ER entre metodologias tradicionais e ágeis não está em como fazê-la, e sim em quando fazê-la. Para eles, enquanto os MT reúnem todos os requisitos, elaboram a documentação de especificação antes da fase do projeto, os MA têm a elaboração presente e até o final do ciclo de vida.

Segundo Jaqueira (2013), os defensores de MA afirmam que os requisitos são alterados de forma tão rápida que a documentação fica desatualizada logo após ser escrita, tornando o esforço desperdiçado. Logo, em tais métodos o envolvimento do cliente é constante (NERUR; MAHAPATRA; MANGALARAJ, 2005), tendo assim, requisitos elaborados de forma incremental de acordo com as necessidades do cliente, seguindo o modelo de histórias de usuário (*user stories*) e utilizando a linguagem natural.

É possível definir as histórias de usuário, segundo Lee, Guadagno e Jia (2003), como breves descrições de características ou comportamentos de sistema. Para Cao e Ramesh (2008), elas servem de âncora para discussões com o cliente, além de possuírem grande valor para eles (COHN, 2004). Uma das características é que elas possuem estimativas definidas pela equipe de desenvolvimento e prioridades estabelecidas de acordo com o valor que a história tem para o cliente.

Histórias de usuário são formas comuns entre os MA para definição das funcionalidades. Além dessa característica, Abrahamsson et al. (2003), Fowler (2005), Sillitti e Succi (2005), Bassi (2008), Ciscon (2009) definem um conjunto de práticas e comportamentos usuais nas metodologias ágeis, sendo elas: adaptabilidade às necessidades específicas da equipe; desenvolvimento incremental com ciclos de curtas iterações que buscam primeiramente resolver problemas que agregam mais valor para o cliente; *releases* frequentes gerando ao final de cada iteração uma versão liberada para o cliente testá-la; simplicidade no aprendizado; testes e implementação ocorrendo juntos; a estimativa e priorização de requisitos antes de cada iteração; grande envolvimento do cliente, sendo considerado parte da equipe; entre outras características (TRINDADE, 2015).

Como visto na Figura 1, algumas metodologias, processos, técnicas e práticas são listadas, sendo elas encontradas na literatura. Abaixo uma breve descrição sobre cada uma é apresentada de acordo com o livro de Highsmith (2002) - um dos integrantes da Aliança Ágil; onde ele as chama de Ecossistema de Desenvolvimento de *Software Ágil*.

- *Dynamic Systems Development Method* (DSDM): desenvolvido em meados da década de 1990, é uma consequência e extensão das práticas de desenvolvimento de aplicativos rápidos. Na Europa, a DSDM tem o treinamento e documentação melhor suportado quando comparada com as demais. Nos seus nove principípios temos: "*envolvimento ativo do usuário, entrega frequente, tomada de decisão da equipe, testes integrados ao longo do ciclo de vida do projeto e mudanças reversíveis no desenvolvimento*".
- *Scrum*: sendo inicialmente desenvolvido por Ken Schwaber e Jeff Sutherland, em que depois teve a colaboração de Mike Beedle, o *Scrum* fornece um *framework* para gerência de projetos. O desenvolvimento é feito em ciclos de *Sprints*, onde um conjunto de funcionalidades do *Backlog* são entregues. Seu nome foi inspirado pelo *Rugby* que, segundo Cohen, Lindvall e Costa (2003), *Scrum* ocorre quando os jogadores com vários fornecedores se amontoam juntos na tentativa de avançar no campo.
- *Feature Driven Development* (FDD): criado a partir da colaboração de Jeff De Luca e Peter Coad, o FDD tem processos breves e duas de suas funções-chave são Arquiteto-chefe e Programador-chefe. O processo é composto por 5 passos que se concentram no desenvolvimento de um modelo de objeto global, construção de uma lista de recursos seguido de planejamento por recurso, etapas iterativas de design por recurso e de construção por característica. Ele difere do XP em sua modelagem arquitetônica frontal leve.
- *Adaptative Software Development* (ASD): criação do próprio Jim Highsmith, o ASD fornece uma base filosófica para os MA, onde, ao invés de rejeitar mudanças, mostra a organizações de *software* como responder a turbulências do atual clima de negócios. Também possui uma filosofia de gerenciamento ágil - chamada de Gerenciamento de Liderança e Colaboração. Algumas de suas práticas são: desenvolvimento iterativo, planejamento baseado em recursos e revisões de grupo de foco do cliente.
- *Extreme Programming* (XP): desenvolvida por Kent Beck, Ward Cunningham e Ron Jeffries, preza pelos valores de comunidade, simplicidade, *feedback* e coragem, além

de fornecer um sistema de práticas dinâmicas. Tem como aspectos importantes: colaboração para modificar a visão quanto ao custo da mudança e ênfase na superioridade técnica através da refatoração e desenvolvimento de teste. Para Highsmith, o XP obteve o maior interesse de qualquer uma das abordagens ágil.

- *Crystal*: o método centrado nas pessoas - colaboração, boa cidadania e cooperação, foi desenvolvido por um "arqueólogo de metodologia", Alistair Cockburn. Com o objetivo de separar o que realmente funciona do que as pessoas dizem que deveriam funcionar, Alistair realizou uma entrevista com dezenas de pessoas. Alistair usa a criticidade, tamanho do projeto e os objetivos para criação de práticas adequadas a metodologia.
- *Lean Development* (LD): desenvolvido por Bob Charette, este é considerado por Highsmith o mais estratégico e o menos conhecido. Foi usado com sucesso em grandes projetos de telecomunicações na Europa. O LD é derivado dos princípios de produção *Lean*, a reestruturação na década de 90 da indústria japonesa automobilística. No LD, Bob ampliou "*a visão da metodologia tradicional de mudança com práticas de gestão restritivas para uma visão de mudança como a produção de oportunidades a serem realizadas usando o empreendedorismo de risco*".

No Apêndice A é apresentado, através da Tabela 11 alguns pontos importantes sobre cada metodologia ou processo que serão tratados neste trabalho. Nele é possível verificar dados como: fases, práticas, papéis e artefatos gerados.

Muitas vantagens dos MA são encontradas na literatura. Tomás (2009) e Trindade (2015) cita vantagens como: redução do tempo de entrega da primeira versão do *software*, possibilitando o cliente verificar o que foi produzido mais cedo, eliminando falhas; fácil adaptação a mudanças, diminuindo riscos; criação do software de forma rápida e segura, agregando cada vez mais valor ao cliente; aumento da visibilidade e adequação das medições e avaliações do estado das funcionalidades e tarefas; além da maior e melhor comunicação entre gestores e desenvolvedores.

Apesar de tantas vantagens traçadas, algumas desvantagens são discutidas, como o fato de terem maior dificuldade com times grandes (COCKBURN; HIGHSITH, 2001); dificuldade no controle de custos, pois nesta metodologia, muitas vezes o projeto termina apenas quando o cliente deixar de levantar funcionalidades (TOMÁS, 2009); pouca eficácia na comunicação com o cliente, pois depende da confiança do cliente no processo ágil e de sua disponibilidade (CAO; RAMESH, 2008). Outra desvantagem está relacionada à

rastreabilidade (ALVES; ALVES, 2009), que é uma atividade considerada importante no processo de desenvolvimento de *software*. Para Eberlein e Leite (2002) os MA não devem depender apenas de gerenciamento de configuração; segundo eles, deve-se adaptar práticas de gestão de requisitos a processos ágeis, proporcionando a rastreabilidade. Segundo Urquiza-Yllescas e Martinez-Martinez (2011) e Trindade (2015) os MA também geram documentação, porém nem sempre elas são suficientes para a rastreabilidade. Espinoza e Garbajosa (2011) afirmam que a falta de documentação adequada implica na rastreabilidade de um requisito, podendo criar sérios problemas para as atividades decorrentes da rastreabilidade, como a análise de impactos durante uma mudança.

2.2 Rastreabilidade em Métodos Ágeis

Definida como "*habilidade de acompanhar e descrever a vida de um requisito*" por Gotel e Finkelstein (1994), a rastreabilidade de requisitos (RR), segundo Kotonya e Sommerville (1998), possibilita descobrir quem sugeriu o requisito, o motivo de sua existência e quais suas dependências. Ramesh e Jarke (2001) também a definem como "*característica de um sistema no qual os requisitos estão claramente ligados às suas fontes e aos artefatos criados durante o ciclo de vida do desenvolvimento do sistema com base nessas exigências*". Genvigir (2009) afirma que a rastreabilidade está relacionada a capacidade de criar vínculos entre requisitos e artefatos que os satisfaçam. Neste trabalho será considerada além da habilidade de acompanhar e descrever a vida de um requisito, possibilitando descobrir os responsáveis, motivos de existência e modificações, irá também verificar informações de artefatos e seus vínculos.

Genvigir (2009), Neto (2013), Dall’Oglio, Silva e Pinto (2010) definem duas habilidades, sendo: Para frente (*forward*): habilidade de rastrear um requisito em direção aos seus refinamentos e; Para trás (*backward*): habilidade de rastrear um refinamento até sua origem. Segundo Genvigir (2009), caso um dessas habilidades não sejam executadas, o processo de rastreabilidade é falho.

Duas classificações gerais estão bastante presentes na literatura (GOTEL; FINKELSTEIN, 1994; GENVIGIR, 2009; DALL’OGLIO; SILVA; PINTO, 2010; POHL; RUPP, 2012; PEREIRA, 2011; NETO, 2013), sendo elas apresentadas e definidas na Tabela 1.

Apesar de não ser classificada na Tab. 1, uma ampliação da definição é apresentado no Coest¹ (*Center of Excellence for Software and Systems Traceability*), sendo a rastre-

¹<http://www.coest.org/>

Tabela 1: Tipos de Rastreabilidade

Rastreabilidade Pré, Inter e Pós-especificação	
Pré-especificação de Requisitos (pré-rastreabilidade):	Envolve artefatos que forma a base do requisito, ou seja, artefatos antes de serem incluídos no documento de requisitos, e os relacionamentos de rastreabilidade entre os requisitos.
Inter-requisitos (entre requisitos):	Traz o mapeamento das dependências entre os requisitos.
Pós-especificação de Requisitos (pós-rastreabilidade):	Contém as informações de rastreabilidade entre requisitos além dos artefatos posteriores a serem incluídos na especificação das próximas atividades do desenvolvimento.
Rastreabilidade Vertical e Horizontal	
Vertical (extrarrequisitos):	Realizada entre requisitos e artefatos produzidos pelo processo de desenvolvimento durante o ciclo de vida do projeto.
Horizontal:	Traz informações das diferentes versões ou variações dos requisitos ou outros artefatos em uma dada fase do ciclo de vida.

abilidade do *software*, onde considera "ampliar a definição para abranger a rastreabilidade multidirecional centrada em diversos artefatos", suportando a análise de segurança, análise de impacto de mudanças, inteligência de projeto e muito mais em toda a evolução, adaptando sistemas de *software*. Pensando da mesma maneira e fugindo da ideia de considerar somente rastreabilidade de requisitos para artefatos (ou vice-versa) ou entre requisitos, pensamos em considerá-la desta forma ampliada, incorporando a noção de rastreabilidade de software, que abrange e interrelaciona qualquer artefato de engenharia de software para outro (CYSNEIROS; ZISMAN, 2017; GOTEL et al., 2012). No contexto deste trabalho a rastreabilidade de software pode ser definida como:

Rastreabilidade de software : habilidade de prover relacionamentos entre artefatos gerados durante o processo de desenvolvimento do software.

Considerando o próprio requisito como artefato, a rastreabilidade de *software* tem o objetivo de contemplar todos os relacionamentos. Durante o texto do presente trabalho, trataremos este termo apenas como "rastreabilidade".

É possível verificar na Figura 2 um exemplo dado pela definição de rastreabilidade de *software*. Já a Figura 3 e Fig. 4 tem a intenção de demonstrar os tipos presentes na Tabela 1.

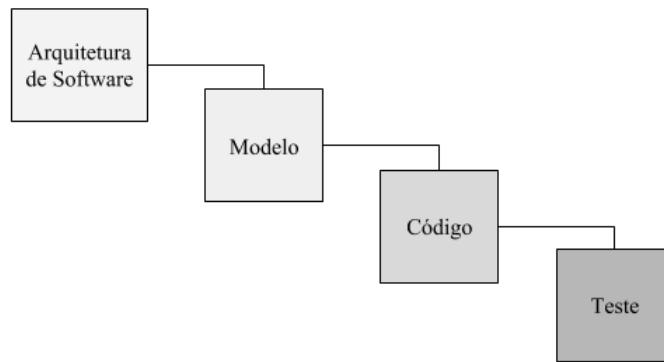


Figura 2: Exemplo de Rastreabilidade de Software

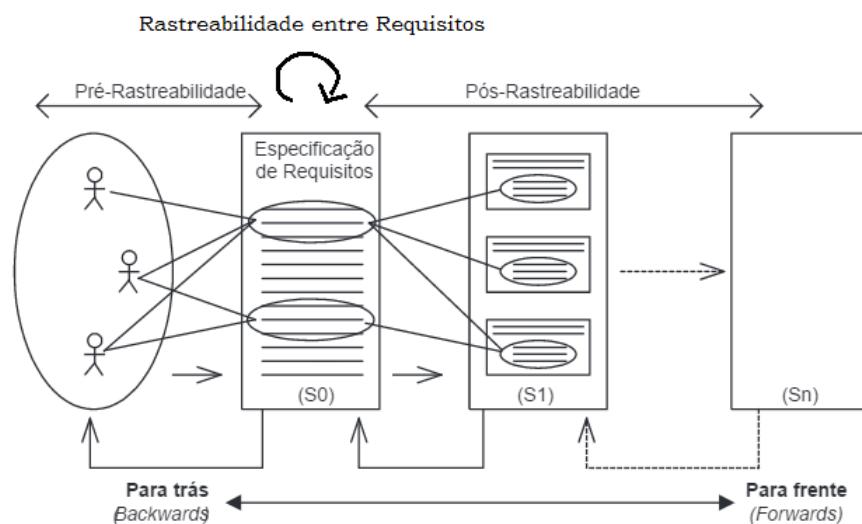


Figura 3: Pré, Inter e Pós-rastreabilidade de requisitos. Fonte: Adaptado de Genvigir (2009) e Gotel (1995)

Segundo Jacobsson (2009) e Lee, Guadagno e Jia (2003), a ideia de como a rastreabilidade pode trazer bons resultados para práticas de desenvolvimento ágil, é um tema em bastante evolução. Sabendo que o desenvolvimento ágil está sempre buscando utilizar melhores práticas da indústria projetadas para aceitar mudanças (LEE; GUADAGNO; JIA, 2003), a rápida análise de impacto que a alteração pode causar seria um grande benefício ao desenvolvimento ágil. Para Lee, Guadagno e Jia (2003), ser capaz de rastrear através dos artefatos de um *software*, o código-fonte, testes e requisitos é um fator relevante para o sucesso de grandes projetos.

Entretanto, Jacobsson (2009), através da entrevista realizada em sua pesquisa, apresenta motivos que levam os agilistas serem contra a iniciativa de aplicar a rastreabilidade

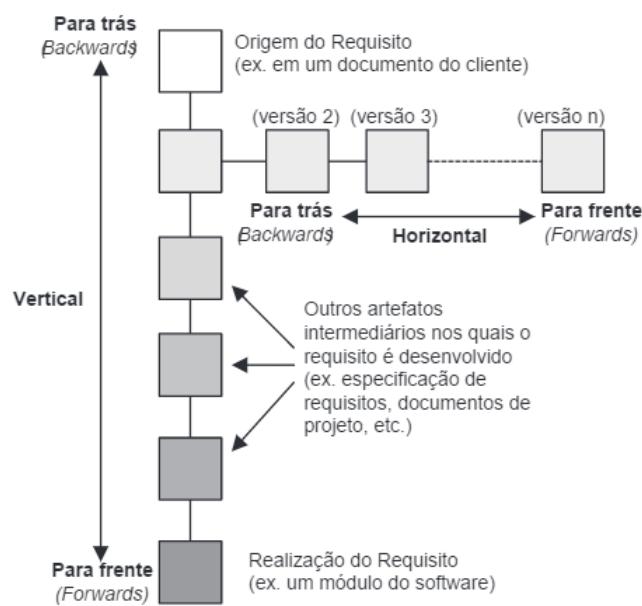


Figura 4: Rastreabilidade Horizontal e Vertical. Fonte: Genvigir (2009) e Gotel (1995)

em ambientes ágeis, alguns deles são: perda de criatividade, pois para os entrevistados a documentação e rastreabilidade limitam a criatividade, tendo que seguir decisões de projeto básico; conhecimento suficiente para guiar o projeto é possível através do Scrum, além de acreditarem que a documentação é contra a filosofia ágil e traz perda de tempo; rotina pesada para garantir a rastreabilidade, o que pode afetar o trabalho e produtividade do desenvolvedor; e, por fim, a falta de motivação, dada por não saberem claramente os benefícios que a rastreabilidade pode trazer.

O desinteresse da equipe em não apoiar a rastreabilidade pode ser decorrente dessa falta de conhecimento sobre seus benefícios, como citado por Jacobsson (2009). Como o ambiente ágil apoia e proporciona uma grande interação entre membros da equipe, muitas informações são trocadas por conversas informais, às vezes face a face, sem os registros delas. Entretanto, quando não documentadas, estas informações podem ser perdidas, seja pela troca de membro da equipe ou casos que existem equipes distribuídas (JACOBSSON, 2009), onde o contato se torna mais difícil; além não haver garantia de que o membro sempre lembrará de informações trocadas informalmente.

A falta de rastreabilidade em metodologias ágeis é um ponto difícil de ser tratado, segundo Alves e Alves (2009). Para eles e Trindade (2015), como a proposta ágil abraça a ideia de mudanças e tem entregas menores e incrementais do sistema, concluem que uma mudança afetará muitos artefatos produzidos e em produção. Portanto, a garantia de que há um profissional que pode analisar todo o impacto da mudança sem o apoio de recursos

que ajudem a fazer a análise de dependências é questionada por Alves e Alves (2009). O uso de ferramentas ágeis e automação para a rastreabilidade, é sugerida por Jacobsson (2009).

Muitas são as vantagens presentes no uso da rastreabilidade durante o processo de desenvolvimento do software. Autores como: Letelier (2002), Egyed e Grünbacher (2005), Genvigir (2009), Delater, Narayan e Paech (2012), Neto (2013) citam algumas dessas vantagens, sendo elas classificadas por Sayão e Leite (2005), Ghazarian (2008), Pohl e Rupp (2012), além de citadas em Trindade (2015). A seguir algumas dessas vantagens levantadas por esses autores:

- Verificabilidade: permite a análise da implementação do requisito no sistema, possibilitando verificar a alocação deles, assim como a identificação de requisitos que não foram testados;
- Identificação de propriedades desnecessárias do sistema: possibilita a identificação se certa propriedade colabora na implementação do requisito;
- Identificação dos requisitos desnecessários: proporciona a identificação de requisitos que não cooperam nas metas do sistema;
- Análise de impacto: colabora na investigação dos efeitos que a mudança em um requisito pode causar, verificando requisitos conflitantes;
- Correção de defeitos: ajuda na identificação do erro, informando quais artefatos devem ser revistos;
- Validação: ajuda na etapa final de validação do sistema, permitindo mostrar se a implementação atende ao conjunto de requisitos acordados com os clientes;
- Reusabilidade: facilita a reutilização de artefatos de requisitos em outros sistemas, onde selecionando um requisito, permite analisar o conjunto de artefatos que se originam dele;
- Determinação de responsabilidade: permite somar todos os esforços parciais para a implementação do requisito; 
- Manutenção: possibilita a cauda e o efeito de falhas; 
- Previsão de custos e prazos: a partir de componentes atingidos pela solicitação, ajuda na estimativa.

Apesar de todas essas vantagens, para Alves e Alves (2009) e Ghazarian (2008) a rastreabilidade ainda não se encaixa no paradigma ágil. Tal fato está relacionado com a dependência da rastreabilidade na criação de documentos formais de requisitos (KÄÄRIÄINEN et al., 2003; GHAZARIAN, 2008, 2010; KHAN; ARSHAD, 2012), em que normalmente tem as especificações de requisitos dos usuários omitidas (DUC, 2013). Logo, a rastreabilidade é uma prática difícil de ser implementada por completa em ambientes ágeis, segundo Trindade (2015) isto ocorre por muitas de suas atividades serem consideradas "pesadas", como por exemplo, a geração da matriz de rastreabilidade. Uma dificuldade citada por Duc (2013) está relacionada a forma de visualizar as dependências de um grande projeto, o que acaba tornando difícil devido a quantidade de relacionamentos que isto pode gerar. E este seria um grande facilitador para gerentes ou líderes de projetos ágeis, diante das decisões que precisarem tomar.

Apesar de existirem esse e mais outros desafios (verificar Trindade (2015)), sabemos que muitos são os benefícios da rastreabilidade para MA e alguns trabalhos estão tentando resolver estes pontos através de testes. Egyed (2003) advertiu que se o problema da rastreabilidade não fosse resolvido em ambiente ágeis, isso poderia causar sérios desafios, como: análise de impacto de mudanças, gerenciamento de mudanças de requisitos ou estimativa. Para Duc (2013), é fundamental uma abordagem de rastreabilidade nova e específica aplicada aos projetos ágeis. Através da pesquisa de Mader e Egyed (2012), vemos algumas porcentagens referentes ao uso da RR. Segundo eles, o uso da rastreabilidade resulta na execução 21% mais rápida das tarefas, em um índice 60% maior em soluções corretas, o que mostra que tal atividade pode produzir melhorias significativas. Apesar da importância e dos avanços na área de rastreabilidade, sua prática ainda traz desafios, e um deles está relacionado com a visualização de informações (CYSNEIROS; ZISMAN, 2017).

Os artefatos gerados em ambientes ágeis e, muitas vezes guardados em plataformas de hospedagem de dados de versionamento e ferramentas de gerenciamento de projeto, contêm informações importantes que podem ser extraídas e mostradas de forma eficiente e eficaz, respondendo aquilo que o gerente ou desenvolvedor procura.

2.3 Visualização da Rastreabilidade

Em Gotel, Marchese e Morris (2008) é possível verificar uma discussão sobre os termos "visualização da engenharia de software" e "visualização da informação". Eles citam que ambas são atividades intensivas em dados, mas com motivações diferentes, e; am-

bas apresentam desafios. Na visualização de engenharia de *software*, há complexidades semânticas e sintáticas na definição estrutural universalmente aplicada via UML (GOTEL; MARCHESE; MORRIS, 2008). Já na visualização de informações há problemas relacionados à necessidade de descobrir ou criar metáforas apropriadas para orientar na estruturação de conjuntos de dados (GOTEL; MARCHESE; MORRIS, 2008). Para eles, estes campos necessitam de padrões de eficácia, seja baseado na sobrevivência e uso amplo ou na definição ou cognição da linguagem visual. Este trabalho está voltado para a visualização da informação.

A visualização é dada como um processo de transformação de dados, informações e conhecimentos em forma visual, utilizando a capacidade visual natural dos seres humanos através de interfaces visuais efetivas (GERSHON; EICK; CARD, 1998). Já Spence (2014) define a visualização de informação como "*formação de uma modelo mental de algo*", onde considera modelo mental a representação interna da realidade externa. Em sua visão, o objetivo da visualização é amplificar o desempenho cognitivo e não apenas criar imagens interessantes. Segundo Jacko (2012) - citado por Spence (2014), o objetivo da visualização da informação também vai além de criar imagens; ainda acrescenta que "*a visualização da informação deve fazer para mente o que os automóveis faz para os pés*". Neste trabalho consideramos que a visualização tem o objetivo de ajudar ao visualizador entender os dados e seus relacionamentos de forma simples e rápida.

A medida que um conjunto de informações cresce, explorá-las se torna cada vez mais difícil (SHNEIDERMAN, 1996). Como um exemplo dado por Shneiderman (1996), é mais fácil explorar uma página de informações do que um livro inteiro, sendo difícil localizar itens conhecidos ou obter uma visão geral. Entretanto, ele afirma que designers estão descobrindo como usar filtros de cores rápidas e de alta resolução, ajudando a apresentar grandes quantidades de informação de maneira ordenada e controlada pelo usuário. Hernandes (2009) e Gershon, Eick e Card (1998) afirmam que interfaces visualmente eficazes permitem uma interação rápida com grandes volumes de dados, ajudando na descoberta de características, padrões e tendências camufladas. Psicólogos, estatísticos e designers gráficos fornecem orientações valiosas relacionadas a apresentação de informações estatísticas; entretanto, designers de interfaces de usuário vão muito além da sabedoria atual através da oportunidade de exibições dinâmicas (SHNEIDERMAN, 1996).

Para Gershon, Eick e Card (1998) muitos tipos de informações contém relacionamentos. Merten, Jüppner e Delater (2011) acreditam que relacionamentos podem ser adicionados entre artefatos, principalmente quando ferramentas de gerenciamento de co-

nhecimento são utilizadas. Estas ferramentas geralmente são compostas por vários tipos de visualização para os links de rastreabilidade, simplificando a compreensão (MERTEN; JÜPPNER; DELATER, 2011).

Sabemos que a rastreabilidade nada mais é que a criação de relacionamentos e, para Gershon, Eick e Card (1998), a visualização levou a novos conhecimentos e permitiu uma tomada de decisão mais eficiente. Logo, segundo eles, a visualização precisa possibilitar aos usuários obter informações rápidas, fazendo sentido e tomar decisões em um tempo relativamente curto (GERSHON; EICK; CARD, 1998). As vantagens que uma visualização adequada de rastros se unem aos benefícios trazidos pela rastreabilidade. Poder utilizá-la de forma eficiente, facilitando o trabalho ágil, pode proporcionar à equipe uma série de informações que muitas vezes ficam guardadas apenas na mente das pessoas.

Técnicas de visualização de rastreabilidade são tratadas na literatura. Através do estudo empírico realizado por Li e Maalej (2012), quatro visualizações comuns para apresentar informações de rastreabilidade foram identificadas, sendo elas: *hiperlinks*, listas, matrizes e grafos de rastreabilidade. Entretanto, algumas pesquisas, como: Merten, Jüppner e Delater (2011), Voytek e Núñez (2011), Chen, Hosking e Grundy (2012), Cysneiros e Zisman (2017) estão trabalhando em outras maneiras de visualizar rastros.

Na Tabela 2 é possível verificar as técnicas de visualização tradicionais (T) e alternativas (A) entre os trabalhos vistos Shneiderman (1996), Merten, Jüppner e Delater (2011), Heim et al. (2008), Chen, Hosking e Grundy (2012), Gotel, Marchese e Morris (2007), Li e Maalej (2012).

Tabela 2: Técnicas comuns de Visualização de Rastros

Técnica	Descrição	Classif.
Listas	Segundo Merten, Jüppner e Delater (2011), listas podem representar grandes quantidades de conhecimento estruturado de forma linear, sendo limitadas a uma única dimensão. Elas representam cada <i>link</i> de rastreabilidade em uma entrada, com informação de origem, artefatos de destino e outros atributos (LI; MAALEJ, 2012). Li e Maalej (2012) afirmam que muitas vezes as listas são usadas no processo de recuperação de <i>link</i> . Contudo, seu modelo linear limita o poder expressivo a uma única dimensão (HEIM et al., 2008).	T

Continuação na próxima página

Tabela 2 – continuação da página anterior

Técnica	Descrição	
Referências Textuais e <i>Hiperlinks</i>	Considerada por Pohl e Rupp (2012) uma maneira simples de representar informações de RR, esta representação consiste em anotar o artefato-alvo como uma referência textual no artefato inicial ou ainda, criar <i>hiperlinks</i> (POHL; RUPP, 2012). <i>Hiperlinks</i> permitem usuários navearem de um artefato "pulando" facilmente para outro (LI; MAALEJ, 2012). Segundo Li e Maalej (2012), <i>hiperlinks</i> conectam conceitos, palavras-chave ou frases relacionadas de forma natural.	T
Matriz	Para Li e Maalej (2012) e Cleland-Huang et al. (2014) a maneira mais comum de inserir e mostrar informações de rastreabilidade continua a ser a matriz. Chen, Hosking e Grundy (2012) diz que a matriz de rastreabilidade é uma grade bidimensional que exibe artefatos em linhas e colunas e representa <i>links</i> de rastreabilidade como marcações nas células geradas. Em termos de álgebra linear, Backes (2008) afirma que matrizes exibem o mapeamento entre fonte e alvo, gerando a matriz de dependências, que representa a relação de dependência entre estes elementos - sendo as linhas os elementos fonte, e as colunas, os elementos alvo. A marcação na célula da matriz vai indicar que o elemento fonte é mapeado para o elemento alvo e, de forma recíproca, isto significa que o elemento alvo depende do elemento fonte (BACKES, 2008; BERG; CONEJERO; HERNÁNDEZ, 2006). Segundo Pohl e Rupp (2012), apesar de ser uma técnica bastante comum, elas são de difícil manutenção e podem se tornar ilegíveis conforme o número de requisitos aumenta (MERTEN; JÜPPNER; DELATER, 2011). Além disso, ainda  deles, muitas matrizes precisam ser criadas para representar de forma clara as informações disponíveis, por exemplo, para cada tipo de relacionamento de rastreabilidade, uma matriz é gerada.	T

Continuação na próxima página

Tabela 2 – continuação da página anterior

Técnica	Descrição
Árvores	Para Merten, Jüppner e Delater (2011) as árvores são bastante intuitivas, entretanto, eles afirmam que não possuem capacidade de apresentar <i>links</i> adicionais de rastreabilidade entre os artefatos dos requisitos. Acrescentar traços ao lado da relação pai-filho mudaria a estrutura de dados da árvore para a de um grafo (MERTEN; JÜPPNER; DELATER, 2011). Exemplos desses traços são decisões racionais e conflitos. Apesar de mostrar duas dimensões por vez, duplicando seu poder expressivo quando comparadas com as listas, elas não possuem a capacidade de mostrar relações de requisitos multidimensionais (HEIM et al., 2008).
Grafos	Segundo Li e Maalej (2012), os grafos permitem a visualização de relações multidimensionais entre requisitos e outros artefatos. A interpretação de artefatos como nós e os <i>links</i> de rastreabilidade como arestas, podem possibilitar os rastros serem visualizados como grafos, segundo Merten, Jüppner e Delater (2011) e Chen, Hosking e Grundy (2012). Chen, Hosking e Grundy (2012) ainda acrescenta que os gráficos podem mostrar uma visão geral das relações de dependências entre artefatos, além de permitir facilmente uma navegação entre os <i>links</i> . Pohl e Rupp (2012) destacam que a distinção entre diferentes artefatos e tipos de rastreabilidade pode ser feita com o uso de diferentes atributos para os nós e arestas, respectivamente. Para Merten, Jüppner e Delater (2011), uma matriz de rastreabilidade pode ser interpretada como uma matriz de adjacência de tal grafo. Heim et al. (2008) afirmam que gráficos de rastreabilidade podem fornecer uma representação intuitiva das relações entre artefatos.
Sunburst	 Segundo Stasko et al. (2000), Sunburst é um gráfico onde os arquivos/artefatos estão dispostos em um <i>layout</i> radial. A raiz destes é mostrado no centro do visor e os níveis sucessivos são atraídos para mais longe do centro (STASKO et al., 2000). Para Merten, Jüppner e Delater (2011) estes arquivos/artefatos são considerados como nós que são desenhados em anéis adjacentes apresentando uma estrutura de árvore, onde cada filho é representado no mesmo espaço radiano que seu(s) pai(s).

Continuação na próxima página

Tabela 2 – continuação da página anterior

Técnica	Descrição	
Netmap	Na definição encontrada em Merten, Jüppner e Delater (2011), o Netmap é um gráfico que, assim como Sunburst, dispõe de um <i>layout</i> radial. Entretanto, sua diferença está no único anel que ele forma composto pelos nós representados por segmentos (MERTEN; JÜPPNER; DELATER, 2011). Além disso, segundo Merten, Jüppner e Delater (2011), cada nó é representado igualmente para evitar erros de interpretação. Outro detalhe são os <i>links</i> de rastreabilidade coloridos presentes no centro do gráfico, podendo ser direcionados ou não.	A
Treemap	Para Shneiderman (1992), Treemap é uma representação projetada para a visualização humana de estruturas de árvores tradicionais complexas. Segundo Wijk e Wetering (1999) o Treemap é construído a partir da subdivisão recursiva de um retângulo inicial. O tamanho de cada subretângulo é proporcional ao tamanho do nó; e, o tamanho dos pais são representados, por exemplo, pela soma do tamanho dos filhos (WIJK; WETERING, 1999). Então, o retângulo inicial é particionada em retângulos menores, de forma que o tamanho de cada retângulo reflete no tamanho da folha (WIJK; WETERING, 1999). Outra característica que o Treemap apresenta, segundo Wijk e Wetering (1999), é que a direção da subdivisão se alterna por nível, sendo o primeiro horizontal e o próximo vertical. Cores e anotações também podem ser representados na intenção de dar informações extras sobre as folhas (WIJK; WETERING, 1999).	A

É sempre bom lembrar que quando tratamos de visualização de dados devemos considerar que há o envolvimento do conhecimento sobre a percepção e cognição humana, além disso, desafios que exigem um pouco da atenção com variáveis, tela (número de dimensões), tarefas, design e contexto muitas vezes são enfrentados (SPENCE, 2014). Estas serão comentadas a seguir.

Considerações em relação à visualização são tratadas em Spence (2014). Segundo ele, como a visualização é uma atividade humana, características humanas como percepção e cognição, devem ser consideradas especialmente quando dados são transformados em imagens. Spence (2014) também declara que, pelo fato de dados serem compostos de muitas variáveis, isto se torna um grande desafio ao decidir como codificar estes dados em imagens. Além disso, ele cita um pouco sobre a tecnologia, onde trata sobre as dimen-

sões de uma imagem, destacando que a representação final pode ser estática, animada e interativa. O número de tarefas é um fator desafiante, quando uma série de subtarefas são requeridas para a realização de um objeto, fica mais difícil a abordagem; por exemplo, ao procurar uma casa, atividades como: exploração, pesquisa, atenção aos detalhes e supressão de dados indesejados, serão influenciados pela maneira como os dados são transformados em imagens (SPENCE, 2014).

Spence (2014) acredita que não há uma teoria da visualização existente que permita ser usada na orientação das decisões relacionadas a como comprimir grandes volumes de dados e transformá-los em imagens. Assim, ele destaca ser necessário identificar problemas, para em seguida ilustrar possíveis representações de dados e sua apresentação no espaço e tempo e então, fornecer uma estrutura para apoiar o design de ferramentas interativas. Por fim, ele julga o contexto da visualização de informação ser de grande importância, afetando no design da representação, apresentação e interação.

Como falado anteriormente, Shneiderman (1996) diz que o crescimento de informações a serem tratadas é proporcional a dificuldade em explorá-las; entretanto ele acredita que o uso de filtros facilite a amostra de grandes quantidades de informações. Para Shneiderman (1996) as habilidades de percepção dos seres humanos são muito subutilizadas em projetos da época. Escanear, reconhecer e recuperar imagens rapidamente, além de detectar mudanças no tamanho, cores, formas, movimentos ou texturas são características dos usuários (SHNEIDERMAN, 1996). Shneiderman (1996) afirma que apesar da existência de muitas diretrizes de *design* visual, o princípio básico pode ser simplificado como "*Mantra de Busca de Informação Visual: Visão geral primeiro, zoom e filtro, então detalhes sob-demanda*". Ele propõem uma taxonomia *Type by Task Taxonomy* (TTT) de visualização de informações, onde em todos os sete tipos de dados (1-, 2-, 3- dados dimensionais, dados temporais e multidimensionais e dados de árvore e rede), os itens têm atributos e, uma tarefa de pesquisa básica é selecionar todos os itens que satisfaçam os valores de um conjunto de atributos.

A visualização de rastros envolve três conceitos principais, segundo Li e Maalej (2012), sendo elas: a informação da rastreabilidade (o que visualizar); a técnica de visualização (como visualizar) e o contexto da tarefa (quando visualizar). Já Gotel, Marchese e Morris (2007) faz uma análise sobre a visualização do ponto de vista da engenharia de requisitos (ER). Eles revelam que a visualização tem sido usada para suportar três aspectos na ER, sendo eles: transmitir a estrutura e relacionamentos entre requisitos e artefatos, dar suporte a elicitação de requisitos e atividades relacionadas e, modelar subconjuntos dos

requisitos ou propriedades deles para fins analíticos específicos. É possível notar que o primeiro aspecto também está relacionado a rastreabilidade.

Na visão de Chen (2004), a visualização da informação está mais preocupada com métodos para encontrar e extrair estruturas a partir de um conjunto de informações e seguidamente gerar *layouts* espaciais e técnicas de desenho de gráfico. Tendo a mesma preocupação apontada por Chen (2004), o uso de ferramenta que contém dados a respeito de projetos será considerada pensando neste conjunto de informações. Logo, ferramentas de gerenciamento de projetos que se integram a uma plataforma de hospedagem de dados e controle de versão - sendo ela o Github, serão avaliadas ao final com o objetivo de verificar se oferece o suporte desejado pelos gerentes e líderes.

2.4 Sistema de Controle de Versão

Atualmente pessoas  atelosas não iniciam um projeto sem o uso de estratégias de *backup* (LOELIGER, 2009). Segundo Loeliger (2009), dados podem ser facilmente perdidos, seja através de mudanças erradas no código ou acidente de disco, por isso é aconselhável manter um arquivo "vivo" de todos os trabalho. Ainda em sua visão, a estratégia de *backup* inclui o controle de versão, ou rastreamento e gerenciamento de revisões. Contudo, antigamente, para executar um bom controle de versão no desenvolvimento de um *software*, era necessário muito esforço, além de bastante disciplina por parte dos desenvolvedores (GARCIA, 2011). Assim, foram criados sistemas de controle de versão que automatizam as tarefas de muito esforço que eram necessárias para a realização do *backup*. Sua evolução foi acontecendo com o passar do tempo, ficando cada vez mais adaptado às necessidades que eram descobertas.

Podendo ser chamadas de Sistema de Controle de Versão (ou VCS - do inglês *Version Control System*), Gerenciador de Código-fonte (ou SCM - do inglês *Source Code Management*), ou ainda, Sistema de Controle de Revisão (ou RCS - do inglês *Revision Control System*), essas ferramentas gerenciam e rastreiam diferentes versões do *software*, ou qualquer outro conteúdo/arquivo, com o objetivo de desenvolver e manter repositórios com os artefatos produzidos, fornecendo acesso a edições e histórico de alterações (LOELIGER, 2009). Para Sink (2011) o SVC (forma que será tratado neste trabalho) é um *software* que ajuda desenvolvedores de uma equipe a trabalharem juntos. Sink (2011) ainda apresenta três objetivos básicos de um SVC: possibilitar o trabalho simultâneo da equipe, não permitir conflitos em mudanças e, proporcionar o arquivamento de todas as versões que já

existiram - gravando o responsável, datas e motivos de mudanças. Três gerações do SVC são apresentadas por Sink (2011), sendo elas mostradas na Tabela 3.

Tabela 3: As três gerações dos sistemas de controle de versão. Retirado de Sink (2011).

Geração	Arquitetura	Operações	Concorrência	Exemplos
Primeira	-	Um arquivo por vez	<i>Locks</i>	RCS, SCCS
Segunda	Centralizada	Multi-arquivo	Merge antes do <i>commit</i>	CVS, SourceSafe, Subversion, Team Foundation Server
Terceira	Descentralizada	<i>Changesets</i>	<i>Commit</i> antes do merge	Bazaar, Git, Mer- curial

Na primeira geração, apenas uma pessoa por vez poderia trabalhar no arquivo. Na segunda, modificações simultâneas eram possíveis, entretanto era necessário a mesclagem antes de realizar o *commit*. Já as ferramentas da terceira geração permitem que a mesclagem (merge) e *commit* sejam feitos de forma separadas, podendo *commitar* antes do merge (SINK, 2011).

A diferença dos tipos de arquitetura centralizada e descentralizada interferem no funcionamento das ferramentas. A centralizada trabalha com um servidor central, onde ocorre o controle de versão. Neste tipo são permitidas diversas áreas de trabalho, tomando como base a arquitetura cliente-servidor. Nela, os usuários fazem cópias locais da versão atual no repositório, faz alterações localmente para depois enviarem para o servidor central, sendo os arquivos mantidos e armazenados neste servidor (GARCIA, 2011). Para casos de conflito no arquivo, o tratamento irá depender do VCS, podendo ser o bloqueio do arquivo até que o usuário libere para os demais ou a mesclagem (merge), onde mostra ao usuário os trechos que deram conflitos, permitindo que ele tome as decisões (GARCIA, 2011). Já a arquitetura descentralizada pode ter vários servidores, sendo um para cada área de trabalho. Ou seja, os usuários trabalham em seus próprios repositórios locais, contendo todo o histórico (GARCIA, 2011). Neles os usuários podem receber e enviar mudanças para outros repositórios de usuário (GARCIA, 2011). Outra característica, segundo Garcia (2011), está na possibilidade de criação de vários ramos principais quando, por convenção, é definido.

O uso de sistemas de controle de versão se torna mais eficaz quando adaptado aos hábitos de trabalho e objetivos da equipe (LOELIGER, 2009). Nele é possível criar e manter repositórios que servem para gerenciar um conjunto de artefatos que foram criados, possibilitando o compartilhamento com toda equipe, assim como o acompanhamento da vida de todo e qualquer artefato. Isto faz do SVC uma grande fonte de informação.

Um SVC que tem fornecido novos passos para o mercado e é o mais utilizado, segundo

Rao e Sekharaiah (2016), é o Git². Com o código aberto, o Git oferece uma estrutura leve e robusta, sendo ideal para gerenciar um conjunto completo de resultados de pesquisa, como conjunto de dados, código estatístico, figuras, entre outros (RAM, 2013). Ram (2013) afirma que, para pesquisadores individuais, o Git fornece de forma poderosa o rastreio e comparação de versões, além de recuperação de erros e exploração de novas abordagens de forma estruturada, mantendo uma trilha de auditoria completa. Outra característica do Git e seus serviços de hospedagem é a possibilidade de trabalho assíncrono, permitindo que todos os colaboradores fundem suas contribuições a qualquer momento (RAM, 2013).

Atualmente, há diversas plataformas de hospedagem de dados com controle de versão que usam a estrutura Git. Github³, Bitbucket⁴ e Gitlab⁵ são algumas delas. Sendo tratadas, através dos seus próprios sites, como podemos ver a seguir:

- Github: plataforma de desenvolvimento inspirada na maneira de trabalho. Possibilita hospedar e rever código, gerenciar projetos e criar *software* junto a milhões de outros colaboradores, desde *open source* ao negócio. Se integra com ferramentas como: Slack, Zenhub, Travis CI, Atom, Circle CI, Codeship, Code Climate. Considera-se a maior comunidade de código aberto do mundo, com 71 milhões de repositórios, 25 milhões de desenvolvedores e 117 mil *businesses*; além do uso por mais de 1 milhão de equipes. Empresas como Airbnb, IBM, Google, Facebook, Walmart, Paypal e Nasa utilizam-o.
- Bitbucket: ferramenta de gerenciamento de repositório Git para equipes profissionais. Possibilita a colaboração no código-fonte - entre outros arquivos, permite aprovações de revisão do código, assegura o fluxo de trabalho e proporciona a criação de um *pipeline* de lançamento. Tem integrações com Jira, Hipchat, Bamboo, Trello, etc. É utilizado por mais de 1 milhão de equipes e 6 milhões de desenvolvedores adoram o Bitbucket. Empresas como a Ford, Paypal, Starbucks e Pandora fazem uso desta ferramenta.
- Gitlab: Uma plataforma que permite o controle de versão usando a estrutura do Git, o Gitlab lança produtos a cada mês, através da colaboração da sua comunidade. Tem o núcleo aberto e continuamente melhorado. Cria o valor mais rápido com seu uso durante todo o desenvolvimento de software. Possui mais de 1800 contribuidores e

²<https://git-scm.com/>

³<https://github.com/>

⁴<https://bitbucket.org/>

⁵<https://about.gitlab.com/>

usado por mais de 100 mil organizações no mundo, o GitLab tem clientes como: Sony, Nasa, Bayer, Comcast e Nasdaq.

Por possuir uma maior quantidade de desenvolvedores que faz uso, como é possível ver acima, através dos dados coletados em cada um dos sites, o Github terá um destaque na seleção de ferramentas de gerenciamento de projetos, sendo este um dos critérios de seleção da ferramenta. Este trabalho teve a preocupação de selecionar para avaliação apenas as ferramentas de gerenciamento de projetos que se integram ao Github, já que esta plataforma de hospedagem de dados e versionamento é uma grande fonte de informações. Pensou nisso como forma de aproveitar estas informações dadas pelos desenvolvedores de forma a favorecer decisões dos gerentes e líderes.

2.5 Trabalhos Relacionados

A literatura traz diversos trabalhos que tratam sobre a visualização de rastreabilidade de software. Melhores tipos de visualização, abordagens e ferramentas são discutidos. Com o uso da técnica *snowball* (ou bola de neve), onde são verificadas referências de artigos previamente selecionados, foi conduzida a pesquisa por trabalhos relacionados. O foco foi dado àquelas pesquisas que tratassem de dados da rastreabilidade de *software* transcritos através de uma visualização. Entretanto, ao final é relatado dois trabalhos que, apesar de não focar na visualização, fazem uso dela e se preocupam em capturar dados de rastreabilidade de *software* em ferramentas de versionamento e gerenciamento, sendo este um assunto também abordado nesta pesquisa.

Trabalhos como Gershon, Eick e Card (1998), Marcus, Xie e Poshyvanyk (2005), Heim et al. (2008), Zhou et al. (2008), Pilgrim et al. (2008), Merten, Jüppner e Delater (2011), Voytek e Núñez (2011), Li e Maalej (2012), Thommazo et al. (2012), Chen, Hosking e Grundy (2012), Filho Lencastre (2012), Niu, Reddivari e Chen (2013), Kugele e Antkowiak (2016), Silva (2016), Rodrigues, Lencastre e Filho (2016), Naves e Furtado (2017), Cysneiros e Zisman (2017), Naves e Furtado (2017), são descritos a seguir.

Para Gershon, Eick e Card (1998), uma maneira de visualizar relacionamentos é através de grafos, onde nós representam entidades e as arestas criam as relações. Na visão deles, o grafo é bom para conjuntos com dezenas ou centenas de nós e *links*, mas quando este número cresce, tornam-se visivelmente confusos. Para aumentar a densidade de informações em grafos, Gershon, Eick e Card (1998) cita uma abordagem que permite que o visualizador veja detalhes e o contexto geral.

Heim et al. (2008) sugerem uma abordagem de foco e contexto que também usa a visualização baseada em grafo, onde representam requisitos como nós e relações como vértices. Assim como Gershon, Eick e Card (1998) afirmou sobre a desvantagem do uso do grafo, Heim et al. (2008) propõem que a abordagem seja para um conjunto limitado de requisitos, pois para grandes quantidades de dados, o uso do grafo é difícil de ser entendido. No seu trabalho, Heim et al. (2008) sugerem ampliar a análise e o gerenciamento tradicional de requisitos através da visualização gráfica, que permite representar de forma multidimensional as relações, além de facilitar a compreensão dos requisitos.

Outra abordagem que ao final faz uso do grafo, é a proposta por Thommazo et al. (2012). Thommazo et al. (2012) apresentam duas abordagens para automatizar a geração da matriz de rastreabilidade com suporte a visualização. Ao final, eles apresentam uma ferramenta criada (COCAR) que tem a intenção de transformar a visualização da matriz em grafos; pois, na sua visão, apesar de melhorar a visualização da matriz com o uso de cores, a dependência multidimensional dos requisitos é difícil de ser compreendida através dela.

Kugele e Antkowiak (2016) propõem uma metodologia baseada em modelo para facilitar a visualização da rastreabilidade em vários pontos de vista e níveis de abstração. Segundo eles, este conceito de visualização intuitiva permitirá verificar a importância do artefato e o impacto da mudança de um artefato em outros. Em seu trabalho, Kugele e Antkowiak (2016) também citam a desvantagem trazida por grafos, afirmando que ao expor engenheiros a grafos extensos, eles são sobrecarregados, quando deveria ser uma visualização para propor uma atuação rápido do engenheiro, de acordo com suas habilidades cognitivas.

Já Li e Maalej (2012) abordam técnicas comuns de visualização de rastros. Tal estudo se preocupa com o que visualizar, quando e como. Entretanto, não coletam todos esses dados na indústria; mas fazem uma análise de algumas visualizações tradicionais de acordo com o perfil. Logo, através de um estudo comparativo e entrevistas realizadas, eles descobriram que as matrizes e gráficos são visualizações mais comuns em atividades de gerenciamento, *hiperlinks* são bastante utilizados para suportar atividades de implementação e testes, enquanto listas parecem ser a técnica menos atraente. Ainda relatam que grafos são preferidos para navegar em artefatos vinculados, matrizes são apropriadas para uma visão geral e *hiperlinks* considerados adequados para informações finas. Uma das suas conclusões é que nem sempre os usuários são capazes de escolher a visualização mais adequada.

Merten, Jüppner e Delater (2011) fogem um pouco das visualizações comuns. Eles utilizam as visualizações Sunburst e Netmap como técnicas de visualizações alternativas às mais tradicionais. Eles implementaram as visualizações em conjunto com um *plugin* de requisitos para a plataforma Redmine. Segundo eles, o funcionamento da técnica é bom para grandes quantidades de artefatos e links de rastreabilidade. Tais visualizações ainda possibilitam a capacidade de apresentar dados derivados.

Voytek e Núñez (2011) mostram algo similar a Merten, Jüppner e Delater (2011). A técnica de visualização apresentada e desenvolvida por eles é baseada em um gráfico, com anéis mostrando artefatos de projeto e nós que representam relacionamentos. Cores específicas são utilizadas para indicar o tipo de relação.

Já Chen, Hosking e Grundy (2012) faz uma combinação com uma visualização considerada comum neste trabalho. Chen, Hosking e Grundy (2012) desenvolvem uma nova abordagem que combina o Treemap e técnicas de visualização de árvore hierárquica, com o objetivo de fornecer uma estrutura global de traços e uma visão detalhada deles. Segundo eles, ambos reduzem a desordem visual, além de serem altamente escaláveis e interativos. Mostram que a abordagem proposta pode suportar a compreensão, navegação e manutenção de *links* de rastreabilidade. O foco da sua pesquisa é na rastreabilidade entre código-fonte e seções em documentos escritos em linguagem natural.

Marcus, Xie e Poshyvanyk (2005) também trabalham no uso da visualização sobre rastreabilidade. Neste trabalho eles discutem elementos e propriedades de *links* de rastreabilidade e levantam um conjunto de requisitos de alto nível necessários para uma ferramenta de visualização que suporte a navegação entre *links* de rastreabilidade. Ao final, é apresentado um protótipo da ferramenta (TraceViz), no qual implementa alguns desses requisitos. O TraceViz exibe: elementos (fontes e destino) de forma hierárquica, relacionamentos de rastreabilidade para um elemento escolhido, e; informações relacionadas à propriedades e histórico de navegação de um relacionamento particular.

Zhou et al. (2008) preocupam-se com a quantidade de informação provinda da rastreabilidade, onde diz que os engenheiros de software são incapazes de destilar as informações necessárias eficientemente. Pensando nisto, o trabalho deles propõe uma técnica de visualização (ENVISION), baseada em árvore hiperbólica, onde visa facilitar a compreensão da rastreabilidade do software através da visualização dos rastros, navegação, pesquisa, foca, filtragem, entre outras funcionalidades. Em Pilgrim et al. (2008) é explorado como as informações de rastreabilidade disponíveis podem gerar diagramas adequados. Logo, eles propuseram uma técnica para visualizar todos os diagramas juntamente com suas infor-

mações de rastreabilidade em uma única visualização, utilizando um editor de diagramas 3D.

Niu, Reddivari e Chen (2013) realizam uma pesquisa na literatura sobre a análise de requisitos visuais, caracterizando seus principais componentes e relacionamentos. Então, analisam o apoio da ferramenta deles aprimorada através de um estudo de caso onde requisitos são processados, visualizados e analisado. Ao final, eles concluem que o aumento da interatividade da visualização dos requisitos pode levar a decisões acionáveis.

O trabalho de Silva (2016) responde de forma gráfica a perguntas que interessam a gerentes de projetos e equipe. O seu objetivo foi criar um modelo de referência de rastreabilidade em gestão de projetos que utilizam MA. Apesar do modelo de rastreabilidade ser o foco principal deste trabalho, ele se assemelha ao nosso ao pensar em responder perguntas vindas de problemas e desafios, sendo útil para gerentes de projeto. Para isto, utiliza a visualização de gráficos, contudo não apresentou justificativas para o tipo de gráfico utilizado. Outra semelhança está no uso de gerenciador de versionamento e de projetos para a coleta de informações.

Naves e Furtado (2017) também se tornam correlato por pensarem nas informações do gerenciador de versão e de projetos, conectando-as e apresentando com o uso da visualização. Outra semelhança está na preocupação da agilidade das MA. Eles ampliam a escalabilidade da técnica Trace++ (proposta por Furtado e Zisman (2016)), por meio da proposta de uma ferramenta para facilitar os processos de rastreabilidade, de maneira a diminuir a complexidade desta tarefa. Entretanto, o trabalho deles não foca em visualização, apesar de no final ser gerado um grafo para a exibição dos relacionamentos, seu foco não estava  possíveis tipos de gráficos.

Filho e Lencastre (2012) afirmam que técnicas de visualização, como matrizes, grafos e árvores, não se adaptam a grande volume de dados; portanto, apresentam em seu artigo um trabalho em andamento para o desenvolvimento de uma ferramenta de visualização, a fim de ampliar a abordagem de rastreabilidade. Além disso, apresentam um conjunto de requisitos de alto nível para uma ferramenta de visualização de rastreabilidade, é apresentado. Já em 2016, Rodrigues, Lencastre e Filho (2016), apresentam uma ferramenta de visualização de rastreabilidade (Multi-VisioTrace), onde possibilita ao usuário escolher entre 4 tipos de visualização a que mais se adequada à sua tarefa.

Cysneiros e Zisman (2017) apresentam o D3TraceView, uma ferramenta para a visualização de rastreabilidade que permite várias consultas relacionadas ao uso de informações de rastros e possibilita exibir informações de rastreabilidade de diferentes formas, a de-

pender da finalidade do uso e das informações a serem visualizadas. Funções e atividades a serem suportadas na ferramenta foram definidas através do OpenUp, que é um processo ágil para desenvolvimento de *software* (CYSNEIROS; ZISMAN, 2017). Segundo eles, o objetivo principal da ferramenta é fornecer vários tipos de visualização para dados de rastreamento específicos, através da entrada de artefatos de *software* e suas relações. Apesar da grande proximidade, uma diferença é que o trabalho presente vai a indústria na intenção de identificar os problemas enfrentados para em seguida ilustrar as possíveis representações de dados, além do foco em quem irá visualizar, como sugerido por Spence (2014).

Mesmo com muitos desses trabalhos se preocupando com a visualização, poucos tratam dos artefatos e objetivos da MA. Além disso, não há uma preocupação de ir na indústria identificar problemas e necessidades para então propor uma visualização adequada. E, como vimos na Seção 2.3, onde tratamos da visualização, o tipo de dado pode ser o ponto chave para a escolha da visualização, juntamente com saber quais problemas enfrentados e para quem a visualização será proposta (SPENCE, 2014). Logo, este trabalho se preocupa em oferecer a visualização mais eficaz para o tipo de dado oferecido pelas metodologias ágeis, para facilitar na resolução de problemas trazidos na indústria e com foco em perfis de gerência e liderança. Portanto, houve a realização de entrevistas com gerentes e líderes de projeto neste trabalho, no intuito de coletar desafios e problemas enfrentados por falta da rastreabilidade na indústria, além de trazer nas entrevistas perguntas sobre a rastreabilidade vindas da literatura, se as equipes já conseguem responder a essas perguntas, como e quais as vantagens de ter essas respostas. A preocupação da pesquisa está em oferecer informações que sejam de utilidade para o gerente de projeto e líderes, de maneira visual, trazendo agilidade e precisão nas informações.

3 Entrevista e suas Análises baseadas em Teoria Fundamentada

As entrevistas realizadas tiveram a intenção de entender como a visualização de rastreabilidade está presente na indústria, além de obter um levantamento dos possíveis problemas e das necessidades relacionadas à ela, dentro de uma equipe de desenvolvimento de sistemas. Tal motivação surgiu baseada na literatura e na experiência profissional da aluna em trabalhar em projetos de grande porte com um elevado número de usuários e instituições, utilizando a metodologia ágil. Através destes meios, percebeu um fenômeno relacionado com o processo da rastreabilidade dos artefatos e da visualização da informação provinda dessa rastreabilidade, além da sua influência na tomada de decisão para a evolução dos sistemas. Logo, foram elaboradas algumas questões de pesquisa a fim de identificar a relação causa-efeito e, possíveis teorias que fundamentam essa relação. Tais questões serão apresentadas no decorrer deste capítulo.

A investigação desse fenômeno junto ao público-alvo estimulou a realização de entrevistas fundamentadas na proposta de Sampieri, Collado e Lucio (2013) e; a utilização, para análise dos dados, de uma abordagem metodológica de Teoria Fundamentada (TF), sendo adaptada de Charmaz (2014), por acreditar que o fenômeno pesquisado foi resultado de construções coletivas e integradas à realidade vivida. Vale ressaltar que os dados foram coletados a partir de dois rounds de entrevistas - presencial (de natureza exploratório) e por *e-mail* (de natureza confirmatória); e, analisados sistematicamente seguindo uma estratégia indutiva baseada em teoria fundamentada, onde segundo Charmaz (2014) com os métodos do TF é possível, conduzir, controlar e organizar a coleta de dados, além de construir uma análise original dos mesmos.

Neste estudo algumas perguntas encontradas na literatura sobre rastreabilidade foram avaliadas pelos participantes a fim de verificar se ainda são importantes no cenário atual, enquanto outras foram elaboradas na intenção de entender as necessidades e possibilidades de visualização de rastros dentro do processo utilizado. Além disso, através da entrevista

foi possível identificar as utilidades diante de cada perfil no que se refere às ligações e visualizações entre informações, quais os problemas e uso da rastreabilidade e sua visualização, permitindo compreender um pouco dos processos e verificar quais tipos de rastros são possíveis para o cenário apresentado pelo participante. Como citado anteriormente, são realizadas dois rounds de entrevistas, sendo uma presencial e de natureza exploratória; enquanto a outra, por *e-mail* e de natureza confirmatória. O uso de entrevistas teve o objetivo coletar essas informações de forma mais detalhada.

Este método de coleta de dados tem uma característica qualitativa que, como definido por Denzin e Lincoln (2006), é uma atividade onde "localiza o observador no mundo". Para Denzin e Lincoln (2006), a pesquisa qualitativa envolve uma abordagem naturalista e interpretativa, onde os pesquisadores estudam seus objetos de pesquisa em ambientes naturais, tentando entender ou interpretar os fenômenos a partir dos significados que as pessoas lhe dão.

Sendo considerada mais íntima, flexível e aberta, quando comparada a um questionário, as entrevistas são definidas por Sampieri, Collado e Lucio (2013) como uma reunião na intenção de conversar e trocar informações entre o entrevistador e o(s) entrevistado(s) - sendo este último também tratado como participante. Ainda segundo a visão deles, as entrevistas são ferramentas para coletar dados qualitativos e empregadas quando o problema de estudo não pode ser observado ou é muito difícil observá-lo por ética ou complexidade, permitindo obter uma informação mais detalhada e pessoal. A finalidade da entrevista realizada neste trabalho se aproxima da definição dada por Sampieri, Collado e Lucio (2013).

Para conduzir este estudo, além dos objetivos apresentados, quatro questões de pesquisa foram elaboradas, a fim de entender melhor o contexto e a prática dos gerentes e desenvolvedores dentro dos temas citados; e, além disso, algumas perguntas levantadas por Gotel e Finkelstein (1997) ajudaram a perceber lacunas a mais a serem respondidas, motivando a busca por essa informação. Com isso, ao final da entrevista tais questões de pesquisa são respondidas com os dados coletados. As questões e seus objetivos são:

QP1. *"Como a documentação produzida, quando são adotadas metodologias ágeis pelas equipes, apoia o desenvolvimento e a evolução dos sistemas?"*

Com a intenção de identificar como a documentação mínima exigida pelas metodologias ágeis apoia os profissionais em sua tomada de decisão. Além disso, esta questão de pesquisa tenta verificar se as documentações são suficientes para as necessidades apresentadas pelos participantes.

QP2. *"Como as informações de rastreabilidade podem ser úteis no dia-a-dia dos agilistas?"*

Buscando compreender como são recuperados e utilizados os artefatos e suas informações, nos sistemas produzidos. Então, tenta verificar se os benefícios provindos da rastreabilidade estavam sendo aproveitados pela equipe.

QP3. *"Como a escolha dos instrumentos para comunicação entre os stakeholders podem afetar a rastreabilidade?"*

Tentando ponderar os efeitos causados pelo dinâmico envolvimento de diferentes interessados em todas as fases da evolução do sistema, sabendo que isto pode impactar diretamente na comunicação, documentação e rastreabilidade.

QP4. *"Como as visualizações da rastreabilidade influenciam na tomada de decisão para evolução de sistemas?"*

Empenhando-se em evidenciar as formas que as estratégias da visualização da rastreabilidade adotadas pelas equipes influenciam na tomada de decisão durante a evolução do sistema, esclarecendo os desafios e riscos inerentes ao processo.

Na sua formulação, as entrevistas possuem a classificação (SAMPIERI; COLLADO; LUCIO, 2013) em: estruturadas, semi-estruturadas e abertas. Para Sampieri, Collado e Lucio (2013), normalmente as primeiras entrevistas devem ser abertas e do tipo piloto para então irem se estruturando conforme o trabalho avança. Entretanto, na nossa pesquisa algumas perguntas já estavam pré-determinadas a partir de levantamento de questões feito por Gotel e Finkelstein (1997). Logo, o piloto já apresentava a classificação utilizada durante todo o processo, sendo a entrevista semi-estruturada. Onde, as semi-estruturadas são baseadas em um roteiro de assuntos ou perguntas, dando a liberdade ao entrevistador de fazer outras perguntas caso precise, para obter mais informações sobre o tema desejado.

Seguindo a classificação semi-estruturada nos dois momentos (rounds), para a entrevista presencial, tentou-se ainda que todas as perguntas pré-determinadas fossem respondidas, de acordo com o ritmo apresentado pelo entrevistado, ou seja, aplicando a ordem de perguntas que era mais conveniente diante das respostas obtidas, assim como, ajustando a forma de questionar. Outra característica foi que, dependendo das respostas, outras perguntas fora do roteiro poderiam surgir, sendo elas discutidas. Já para as entrevistas por e-mail, as perguntas eram enviadas e ajustadas com o vocabulário a depender do perfil do participante.

As entrevistas foram formuladas de forma individual a cada participante com per-

guntas abertas em ambas casos (presencial e por *e-mail*), na intenção de permitir que o entrevistado/participante discorra sobre o que foi perguntando e haver uma maior proximidade da perspectiva deles. O objetivo era obter um maior detalhamento da sua visão sobre o tema, observando seu entendimento, opinião, problemas e necessidades relacionadas à rastreabilidade e sua visualização no processo de desenvolvimento. No caso da entrevista presencial, como a discussão poderia se tornar longa, ao início de cada uma foi solicitada a sua gravação, sendo ela realizada apenas diante da permissão do entrevistado. Não houveram problemas quanto as gravações, todos os participantes permitiram, o que gerou áudios e vídeos (para casos de entrevistas a distância, foram realizadas via Skype ou Hangout).

Algumas recomendações dadas em Sampieri, Collado e Lucio (2013), para a estruturação e levantamento das perguntas foram seguidas, sendo elas:

1. Perguntas totalmente abertas (como já mencionado) e neutras;
2. Perguntas mais gerais inicialmente, para depois passar para as mais específicas, mesmo a ordem sendo flexível e esteja subordinada aos temas que surgirem;
3. As perguntas e a forma de apresentá-las tiveram a intenção de permitir ao participante compartilhar sua perspectiva e experiência sobre os temas rastreabilidade e visualização, já que ele é considerado o especialista;
4. Para uma mesma pergunta, várias formas de fazê-la foram pensadas, na intenção de ter como alternativa, caso não ficasse claro o entendimento da pergunta.

O roteiro completo das entrevistas realizadas podem ser encontrados no Apêndice B (entrevista presencial) e Apêndice C (entrevista por *e-mail*). Além dos dados demográficos, as temáticas que envolveram as questões se tornaram seções, que foram:

Metodologia de desenvolvimento e gerenciamento de requisitos: Esta seção teve o objetivo de entender o processo utilizado para o rastreio de informações específicas. Apesar de não ser classificada como rastreabilidade, neste momento tentou-se coletar se as perguntas listadas por Gotel e Finkelstein (1997) seriam importantes de serem respondidas - na visão do entrevistado. A intenção de não classificá-la como rastreabilidade foi no intuito de não surpreender o entrevistado com um termo que talvez ele não conhecesse, tendo a preocupação de não inibi-lo na entrevista. Portanto, nesta seção evitou-se o uso do termo ?rastreabilidade? com o objetivo de

deixá-lo mais confortável em usar termos e atividades que estavam presentes em seu dia-a-dia;

Rastreabilidade: Após o entrevistado apresentar um maior envolvimento com a entrevista, esta temática começa a ser trabalhada. Nela buscou-se entender a visão do entrevistado sobre o tema, sendo complementado pelo entrevistador caso necessário. Além dessas informações, de forma mais direta foram realizadas perguntas relacionadas a preocupação da rastreabilidade de *software* dentro do processo e sobre quais tipos eram utilizados;

Visualização: Aqui também foram coletadas informações sobre o entendimento do entrevistado na área, também questionou-se informações sobre tipos de visualização de rastros utilizados e quais os desejados, acrescentando perguntas sobre possíveis ferramentas utilizadas para geração de visualizações.

Algumas perguntas do primeiro round de entrevistas (presenciais), presentes na seção de metodologia de desenvolvimento e gerenciamento de requisitos, são um reflexo daquelas encontradas em Gotel e Finkelstein (1997) - como citado acima. Algumas das criadas foram divididas em duas na intenção de que o entrevistado falasse detalhadamente sobre cada questionamento. Outras foram unidas, como podemos ver na MG5, onde percebeu semelhança em duas perguntas de Gotel e Finkelstein (1997). Então decidiu que, para não ficar redundante, perguntou sobre o funcionamento do processo, e em sequência sobre a importância de ter cada informação daquele processo, conseguindo assim, responder às duas perguntas. Na Tabela 4 é possível verificar o mapeamento das perguntas efetuadas com aquelas encontradas na literatura. O ID utilizado se refere ao mesmo presente no Apêndice B e C.

Gotel e Finkelstein (1997) afirmam que através do estudo empírico realizado pelo seu próprio trabalho em 1994 (GOTEL; FINKELSTEIN, 1997), foi argumentado que a informação mais fundamental para aliviar os problemas relacionados a rastreabilidade de requisitos a longo prazo eram as relacionadas a identificação de fonte(s) humana(s). Eles descobriram que os problemas de rastreabilidade de requisitos tendem a surgir quando profissionais não conseguem responder a perguntas relacionadas ao pessoal envolvido. Justificam que isso ocorre devido as pessoas serem frequentemente a linha de base final sempre que os requisitos precisam ser reexaminados ou re-trabalhados.

É importante destacar que um dos objetivos desta entrevista é saber se essas perguntas são importantes para os entrevistados e como poderiam ajudá-los. Além disso, também

Tabela 4: Relação entre Perguntas da Entrevista na seção de Metodologia de Desenvolvimento e Gerenciamento de requisitos com Perguntas levantadas por Gotel e Finkelstein (1997)

ID	Perguntas da Entrevista	Perguntas em Gotel e Finkelstein (1997)
MG3	É possível saber quem são as pessoas envolvidas na produção do requisito? É importante ter essa informação? Por quê?	Quem esteve envolvido na produção deste requisito e como?
MG4	Como essas pessoas são envolvidas? Isso é importante pra você? Por quê?	
MG5	Se houver mudança de responsável, você teria como saber quem foi o responsável, quem é atualmente e quando houve a mudança? Como é o processo? Seria relevante ter essa informação? Por quê?	Quem foi originalmente responsável por este requisito, quem é atualmente responsável por isso e em que pontos em sua vida essa responsabilidade mudou de mãos? Em que pontos da vida deste requisito os acordos de trabalho de todos os envolvidos foram alterados?
MG6	Quem precisa ser envolvido ou informado na mudança ou produção do requisito? Como você verificaria isso (documentação, definidos informalmente pelo cliente ou pelo gerente)? É importante ter essa informação? Por qual o motivo?	Quem precisa estar envolvido ou informado das mudanças propostas a esse requisito?
MG7	Para você seria relevante ter a sistematização dos desdobramentos relacionados à perda de um membro da equipe? (Lembrando que a perda de um membro na equipe pode levar a perda de conhecimento) É importante ter essa informação? Por quê?	Quais são as ramificações, em relação à perda de conhecimento relacionado aos requisitos, se um indivíduo ou grupo específico deixar um projeto?

tentou entender os processos a fim de verificar prováveis problemas e uso da rastreabilidade e visualização. E, diferente de Gotel e Finkelstein (1997), a entrevista também relatou a rastreabilidade de *software*, sendo outro diferencial o foco na metodologia aplicada pelo entrevistado, tendo os métodos ágeis como principais na pesquisa. Assim, verificamos as perguntas a partir de um cenário mais atual e com essas informações será possível a construção de visualizações apropriadas que respondam às perguntas mais relevantes para os participantes no cenário ágil. Na Tabela 5 é possível verificar algumas questões encontradas na entrevista. Bem como na Tabela 4, o ID presente na Tabela 5 se refere ao mesmo presente no Apêndice B e C.

Lembrando que, como se trata de uma entrevista semi-estruturada, na entrevista presencial, a pergunta era feita conforme a discussão ia sendo gerada e de acordo com os termos utilizados pelo entrevistado durante a entrevista. Muitas vezes, algumas perguntas não precisaram ser feitas, pois os próprios entrevistados já discorriam nas respostas que

Tabela 5: Algumas Questões de Pesquisa presentes no Roteiro da Entrevista
Dados Demográficos

DD4	Formação
DD6	Tipo de cargo que ocupa atualmente
DD7	Tempo de experiência profissional neste cargo
Metodologia de desenvolvimento e gerenciamento de requisitos	
MG1	Quais metodologias de desenvolvimento são utilizadas em sua equipe?
MG5	Se houver mudança de responsável, você teria como saber quem foi o responsável, quem é atualmente e quando houve a mudança? Como é o processo? Seria relevante ter essa informação? Por quê?
Rastreabilidade	
R2	No seu projeto há preocupação com a rastreabilidade?
R8	Qual o custo dentro da sua equipe de manter a rastreabilidade dos artefatos obtidos ou gerados? É compensado? Se sim, como? Caso contrário, por quê?
R9	Qual o papel você atribui ao uso da rastreabilidade dentro dos seus projetos? Conte me como foi a experiência ou caso que você vivenciou.
Visualização	
V3	Você gostaria de ter algum tipo de visualização sobre alguns dados que não tem hoje e não foi perguntado sobre?
V5	Você faz uso de alguma estratégia ou ferramenta de visualização dos relacionamentos entre os artefatos do projeto? Qual o papel e em quais momentos são/foram importantes no gerenciamento do projeto? Alguma experiência para relatar? Caso contrário, ter uma visualização dos relacionamentos entre os artefatos seria útil no seu dia-a-dia? Por que?
V6	A visualização poderia ser relevante para lhe ajudar a identificar problemas, impacto das mudanças ou soluções? Por quê? Se sim, quais tipos de visualização seriam mais adequadas nas classes indicadas por você?

levavam a conclusões de perguntas futuras.

Mudando um pouco da estruturação para a execução, alguns detalhes também foram pensados. Os convidados foram selecionados com base em uma lista de contatos, pessoas indicadas por esses contatos, além de considerar os critérios de experiência profissional atual no uso de metodologias ágeis e o desempenho de papéis de liderança ou gerência e desenvolvimento em ambientes ágeis. Em ambas situações os convidados receberam convites à participação, onde nele foram expostos a temática do trabalho e o tempo médio de dedicação a entrevista para os casos presenciais. Os convites e troca de informações foram realizadas via *e-mail* ou *whatsapp*, tendo para convite e agradecimento um modelo padrão que foi ajustado de acordo com a proximidade do entrevistador e participante; contudo, as informações primordiais eram mantidas, como por exemplo, em mensagens relacionadas ao convite, o tempo da entrevista presencial era sempre informado. Estes modelos podem ser verificados no Apêndice D.

Outro cuidado considerado também relacionado à execução, entretanto, da entrevista presencial, foi o ambiente. Como houve a preocupação em realizar a entrevista em locais que o entrevistado ficasse confortável, a partir dos convites iniciais eram gerados informações sobre os possíveis ambientes para a realização da entrevista. Isto permitiu entrevistas

desde salas de reunião até cafeterias. Datas e horários também foram decididos a partir das disponibilidades dadas pelo participante através da troca de mensagens originadas pelo convite.

Ainda sobre a entrevista presencial, pensando na interpretação dada, com a intenção de obter mais de uma visão do que foi discutido, cada entrevista foi realizada com dois entrevistadores e um entrevistado - com exceção de duas, onde não foi possível ter a presença de um dos entrevistadores na sua execução. Apesar de ter presente dois entrevistadores, as entrevistas eram conduzidas apenas por um, sendo o outro suporte, caso o participante não entendesse a maneira do entrevistador principal se expressar o segundo entrevistador poderia atuar. Outra intenção dada ao segundo entrevistador era a possibilidade de discutir as visões das respostas ao final de cada entrevista. Nas entrevistas por *e-mail*, também houve a participação na elaboração e discussão sobre a resposta na perspectiva dos dois entrevistadores iniciais, tendo a mesma intenção da entrevista presencial - analisar sob pelo menos dois pontos de vista. Em todos os casos, a entrevista por *e-mail* foi analisada pelos dois entrevistadores.

Como está sendo detalhado, é possível perceber que métodos mistos foram adotados para a condução do estudo qualitativo. Além da entrevista presencial, obteve-se uma segunda bateria de entrevista feita por *e-mail* - o Apêndice D também apresenta modelos de *e-mails* para realizar perguntas e lembretes para responder a elas. Como citado anteriormente, tais métodos tiveram a natureza exploratória e confirmatória, respectivamente.

Para a entrevista presencial, estimou-se aproximadamente 1 hora de duração, sendo este tempo definido através da entrevista-piloto, onde tentou não extrapolar o combinado por mensagem e assim não prejudicar compromissos posteriores do entrevistado. Este tempo pode ser extrapolado ou minimizado, a depender do andamento da entrevista e principalmente da disponibilidade e sinais de fadiga apresentados pelo entrevistado, sendo neste último caso um resultado da maturação - onde, como definido por Wohlin et al. (2012), a maturação é o efeito de reação dos indivíduos à medida que o tempo passa, podendo afetar negativamente através do cansaço e aborrecimento, ou positivamente com o aprendizado.

A entrevista-piloto teve um tempo total de 1 hora e 36 segundos. Nela, o objetivo foi avaliar as questões apresentadas, estruturação da entrevista, tempo, viabilidade e eventos adversos. Após a realização desta etapa, discussões entre os entrevistadores foram realizadas e ajustes foram feitos com base no que foi observado. Além disso, ao final da entrevista-piloto, uma conversa rápida com o entrevistado aconteceu, onde um *feedback*

sobre suas impressões foram dados, desde possíveis perguntas que poderiam ter causado desconforto para serem respondidas, até a compreensão ou não do que foi questionado.

3.1 Participantes

No público-alvo envolveram pessoas que fazem parte do processo de desenvolvimento de *software* e que usam metodologias ou processos ágeis, sendo eles gerentes/líderes - identificados pela letra "M-", e desenvolvedores - identificados pela letra "D-". Inicialmente, a entrevista tinha o foco voltado para aqueles que possuíam um cargo de liderança ou gerencial; entretanto, em um segundo momento buscou pelos demais perfis de uma equipe de desenvolvimento, pois percebeu que a realização de entrevistas com desenvolvedores traria informações relevantes a respeito da execução do seu trabalho e como isso ajudaria a gerar informações suficientes para que as necessidades dos gerentes sejam cobertas. Outro ponto importante nas entrevistas com desenvolvedores foi a descoberta de suas intenções quando gerados artefatos produzidos por eles.

Dentre os perfis indicados, oito profissionais - sendo estes cinco gerentes e três desenvolvedores, aceitaram o convite para a entrevista presencial. Nas entrevistas por *e-mail*, sete convites de profissionais foram aceitos, sendo quatro participantes possuem o perfil gerencial e três têm o perfil de desenvolvedor. A classificação do perfil foi feita através de uma pergunta inicial sobre o cargo que ocupa, além de também ser analisada a postura do participante diante de perguntas que buscam entender o processo de desenvolvimento seguido. Notou que, mesmo alguns participantes afirmando ter um cargo gerencial ou de liderança, eles possuíam atividades voltadas para o desenvolvimento.

A média de idade dos participantes foi um pouco mais de 36 anos, variando entre 24 e 49 anos. Já a média relacionada ao tempo de experiência no último cargo é de um pouco mais de 6 anos, sendo de 1 a 15 anos sua variação. Pode-se dizer que foram coletados dados desde pessoas com mais experiência e tempo de trabalho até aqueles menos experientes e relativamente recentes no seu último cargo. Outra característica é que o sexo predominante foi o masculino, sendo ele 73,33% do conjunto de entrevistados.

O campo de atuação das empresas em que os participantes trabalhavam foi bastante diverso. Entretanto,  não será possível detalhá-los devido ao termo de confidencialidade dado nas entrevistas.. Estas empresas variam para médio a grande porte, a depender da quantidade de funcionário que nelas são empregados.

3.2 Análises

Na entrevista presencial, com a gravação sendo permitida por todos os entrevistados, após sua realização, o material produzido foi transscrito e formatado em planilhas. Já a entrevista por *e-mail* teve a intenção de validação do que havia sido tratado na entrevista de modelo anterior, ou seja, ela ocorreu somente após a entrevista presencial ser finalizada. Na sua abordagem, as perguntas foram enviadas via *e-mail*, obtendo uma resposta através do mesmo. Assim como nas presenciais, as entrevistas por *e-mail* foram colocadas em planilhas para análise. Com todos os dados em planilhas, separados por participante e modelo de entrevista, foi utilizada uma abordagem baseada em teoria fundamentada (TF), adaptada de Charmaz (2014). Portanto, quando tratados conceitos, tal referência é utilizada.

Os métodos do TF proporcionam a percepção dos dados sob uma nova perspectiva, além de possibilitar a exploração dos dados por meio de uma redação analítica já na fase inicial (CHARMAZ, 2014). Para Charmaz (2014), tais métodos baseiam-se em diretrizes sistemáticas e flexíveis para a coleta e análise de dados, construindo assim, teorias fundamentadas nos próprios dados. Logo, a partir dos dados coletados nas entrevistas, foi utilizada a TF para análise dos dados. Nesta etapa de análise, como sugerido ainda por Charmaz (2014), tenta-se descobrir o que ocorre nos ambientes de pesquisa que foram integrados e como é a vida dos participantes. Analisa-se a forma como eles explicam seus enunciados e ações, assim como é questionada a compreensão analítica que se tem sobre os participantes.

Uma das etapas encontradas na TF é a codificação. Para Charmaz (2014), a codificação é associação de marcadores a segmentos de dados que os representam; é uma categorização dos dados com uma denominação concisa. Ela refina dados, classifica-os e fornece uma maneira de poder comparar e debater sobre os segmentos codificados (CHARMAZ, 2014). Charmaz (2014) ainda a define como um elo fundamental entre a coleta dos dados e o desenvolvimento de uma teoria emergente para explicar tais dados, definindo o que ocorre com eles e debatendo sobre seus significados. Então, para a análise em TF, estudou os primeiros dados, começando a separar  classificar e sintetizar por meio da codificação qualitativa, como sugerido em Charmaz (2014).

Além da codificação, também foram realizados memorandos, sendo compostos de anotações analíticas sobre os códigos, comparações entre eles e qualquer outra reflexão que iam ocorrendo no decorrer da análise sobre os dados. A partir das comparações e criação

de memorandos, foram definidas ideias que melhor se ajustavam e interpretavam os dados, sendo as categorias analíticas provisórias (CHARMAZ, 2014). Com o andamento da análise, as categorias provisórias se tornavam mais sistematizadas, visto que conforme a codificação ocorria era sempre questionado sobre quais categorias teóricas aqueles enunciados podiam indicar. A Figura 6 representa este processo de codificação e criação de memorandos.

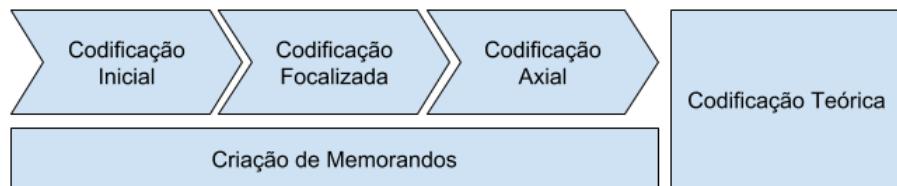


Figura 5: Processo de Codificação e criação de memorando realizado na aplicação da Teoria Fundamentada

Com os dois conjuntos de dados obtidos a partir das transcrições das entrevistas iniciais e respostadas dadas por e-mail nas entrevistas de validação, a codificação foi realizada em três fases, sendo elas (definidas a partir de Charmaz (2014)):

- Codificação Inicial: Sempre aberto à exploração de quaisquer possibilidades teóricas que possam ser reconhecidas nos dados, este passo inicial orienta para as decisões posteriores relativas à definição das principais categorias. Nesta fase, tentou observar atentamente a cada ação, codificando através delas. Tal sugestão é dada por Charmaz (2014), onde diz que ao codificar com base nas ações, refreia as tendências de realizar saltos conceituais e adoção de teorias existentes antes de obter o trabalho analítico realizado. Entretanto, apesar de Charmaz (2014) concordar com Glaser (1992) que durante a codificação inicial não deve ter em mente conceitos preconcebidos, como ela mesmo reconhece, existem pesquisadores que consideram ideias e experiências anteriores, onde no caso deste trabalho, estas foram consideradas. Assim como Charmaz (2014) cita, afirmado por Dey (1999), “há uma diferença entre uma mente aberta e uma cabeça vazia”, ou seja, apesar de considerar as próprias experiências, ainda sim, durante a codificação inicial, sempre tentou permanecer aberto para aprender mais durante a codificação. Outro detalhe importante a destacar nesta fase é que a codificação inicial utilizada neste trabalho foi a linha por linha, pois, como sugerido por Charmaz (2014), tal maneira funciona bem com dados detalhados sobre problemas ou processos empíricos fundamentais, o que reflete no que foi coletado.

- Codificação Focalizada: Sendo a segunda fase da codificação, ela significa utilizar os códigos anteriores mais significativos ou frequentes para analisar detalhadamente grande volume de dados (CHARMAZ, 2014). Nesta fase foi necessário decidir quais entre os códigos iniciais permitem uma compreensão analítica melhor, ajudando assim na categorização dos dados. Durante a codificação focalizada houveram momentos em que foi necessário retornar aos dados anteriores devido a um novo entendimento gerado naquele momento.
- Codificação Axial: Representando as propriedades e dimensões de uma categoria, a codificação axial relaciona categorias e subcategorias (CHARMAZ, 2014). Nesta fase, os dados fragmentados na codificação inicial foram reagrupados na intenção de dar coerência à análise, como sugerido por Charmaz (2014). Aqui foi possível perceber de forma mais clara as categorias que surgiram ao final da análise.

Junto a autora deste trabalho, outra pesquisadora realizou a codificação inicial de forma independente. Em seguida, essas pesquisadoras efetivaram a codificação focalizada, para então, de forma cooperativa realizar a codificação axial. Com o resultado do processo iterativo, surgiram 11 categorias principais que foram relacionadas na codificação teórica. Ao final, estas categorias foram associadas às temáticas das questões de pesquisa (Comunicação, Documentação, Rastreabilidade e Visualização).

Para a **QP1**, cuja a temática era Documentação foram relacionadas as categorias: documentação, repositórios, gerenciamento de equipe, diferenças em projetos e equipes, participação do *stakeholders*, demandas e rastreabilidade. Enquanto a **QP2**, com a temática de Rastreabilidade, teve as seguintes categorias relacionadas: rastreabilidade, metodologias de desenvolvimento, gerenciamento de projetos, diferenças em projetos e equipes, gerenciamento de equipe, requisitos e repositórios. A temática de Comunicação, **QP3**, obteve as seguintes categorias: gerenciamento de equipes, metodologias de desenvolvimento e participação dos *stakeholders*. E por fim, a **QP4** com a temática de Visualização, teve as seguintes categorias vinculadas: visualização da rastreabilidade, gerenciamento de projetos, gerenciamento de equipe, metodologias de desenvolvimento e repositórios.

Na codificação também foram utilizados códigos dos termos específicos utilizados pelos participantes, sendo estes conhecidos como códigos *in vivo*. Segundo Charmaz (2014), os códigos *in vivo* ajudam a conservar, na própria codificação, os significados dos participantes, relativos às suas opiniões e atitudes; servindo como marcadores do discurso, além dos significados dos participantes. Sua utilidade vai para: termos gerais que sinalizam significados condensados, porém expressivos; termo inovador do participante, que apreende

os significados ou experiências; e, termos abreviados característicos do campo e específicos de um grupo (CHARMAZ, 2014).

Para o tratamento dos códigos in vivo encontrados, foi criado um glossário onde os termos específicos e suas definições eram tratados. Neste glossário também foram definidos termos gerais, não sendo apenas os específicos de cada participante. Como relatado durante este capítulo até aqui, todo o processo de análise pôde ser representado na Figura 6.

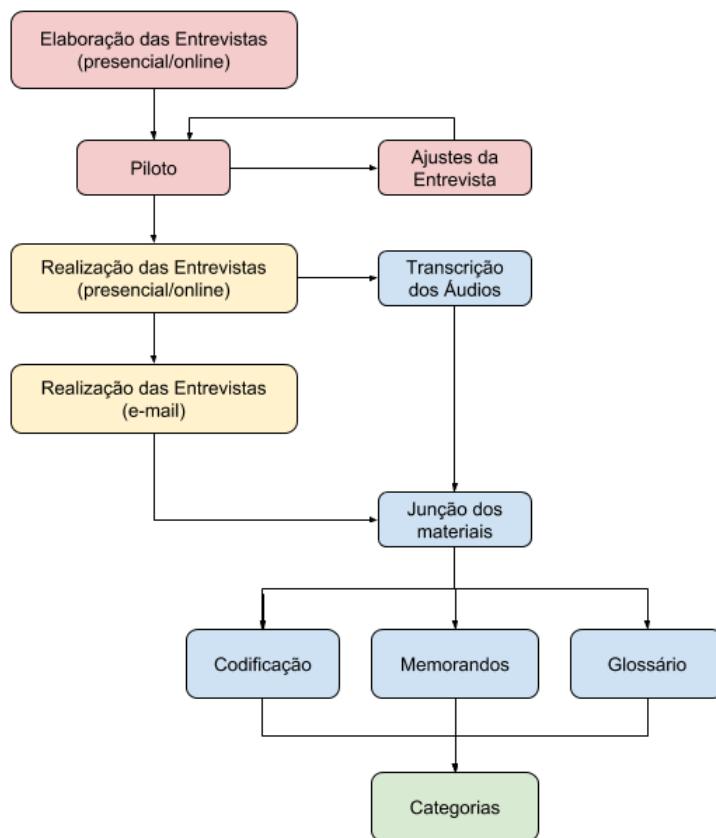


Figura 6: Processo de análise das entrevistas

3.3 Resultados

As equipes, que adotam metodologias ágeis, assumem os desafios inerentes a recuperação da informação, onde sumariamente há uma documentação mínima e distribuída em diferentes artefatos, para privilegiar um ambiente de desenvolvimento mais flexível às mudanças. Fato ocasionalmente inconsciente aos interessados, e potencializado quando não há um planejamento adequado à rastreabilidade dos artefatos disponíveis.

Por outro lado, o planejamento, a realização e a visualização apropriada da rastre-

abilidade podem apoiar substancialmente a correta tomada de decisão pelos gerentes e desenvolvedores, considerando os cuidados necessários à refatoração e à integração contínua na evolução de sistemas. As proposições de cada tema são demonstradas a seguir, a partir das temáticas trazidas com as questões de pesquisa.

QP1 - Documentação: *"Como a documentação produzida, quando são adotadas metodologias ágeis pelas equipes, apoia o desenvolvimento e a evolução dos sistemas?"*

A documentação em metodologias ágeis, não existindo um marco, como nas metodologias tradicionais, está sujeita aos artefatos de código-fonte, arquivos de log (logs) e eventuais registros de comunicação, na maioria das equipes.

M05: *"eles preferem codificar e não documentam o que fazem. Isso dificulta bastante, pois há uma rotatividade muito grande e o conhecimento é perdido".*

D02, demonstrando a preocupação de ter o código como uma fonte suficiente para dar continuidade a uma tarefa: *"que não está claro no código, entendeu?! a gente tenta criar um código mais legível possível (...) que fique claro ali o quê, enfim, cada coisa está fazendo, entendeu?! aí, enfim, o que cada... e a gente também tenta usar ao máximo o vocabulário do domínio dentro do código, entendeu?"*.

D03: *"as discussões, por exemplo, 'Ah, estou com dúvida nisso aqui' ou... é... a pessoa responder (uma dúvida) pelo Jira para manter um histórico para, caso eu não poder fazer aquela task, outra pessoa fazer e tal (...) então, acontecia, às vezes, de uma outra pessoa pegar aquele bug que eu não pude terminar, em uma outra sprint. Então, ela já tinha um histórico ali (na ferramenta), do quê estava sendo discutido antes..."*.

D01: *"o log tem tudo, né?! tipo, toda alteração tinha um log... mas não tinha uma padronização (na utilização)".*

M02: *"então, se eu quero saber qual o foi o desenvolvedor que atendeu aquele requisito, eu vou no (nome da ferramenta de gerenciamento de projetos) e eu tenho como saber qual foi o desenvolvedor que implementou aquele requisito, qual foi o testador que testou aquele requisito".*

Mesmo considerando apenas esses três tipos de artefatos, os agilistas encontram múltiplas dificuldades em construir e manter a documentação atualizada durante o ciclo de vida do sistema. A natureza dessas dificuldades estão relacionadas desde, por exemplo, a identificação dos interessados, dos responsáveis e de seus papéis; tamanho e variabilidade dos projetos; tamanho das equipes; pouca importância dada pelos interessados; priorização; ambiguidade textual; limitações das ferramentas utilizadas.

M02: "a gente já consegue identificar (o stakeholder), só que isso não fica registrado... é, não tem um registro dessa informação... e aí, assim, eu fico sabendo por isso, por essa procura (do stakeholder), porque é mediante a demanda, assim, é uma identificação minha por essa necessidade".

M04: "agora é, falando com relação a esse registros das pessoas que são responsáveis (...) ter esses tipos de registros precisava considerar o tamanho e a criticidade do que você tava fazendo".

D01 falando das limitações da ferramenta utilizada: "uma tarefa pode ter vários logs... então pra você saber quem mexeu naquilo, você tem que dá uma vassourada naqueles... ali, de um em um...".

M03: "com essa demanda aumentou (desses treinamentos), a documentação também ficou atrasada; então hoje eu não posso nem dizer que o que está...".

A importância da documentação e rastros de informação são recorrentes para os gerentes, independente do artefato a ser utilizado, como as trocas de *e-mails*, registros das atividades, arquivos de log, relatórios de testes e, até mesmo, *screenshots*. Enquanto que para os desenvolvedores, em sua maioria, os artefatos utilizados estão restritos ao código-fonte e aos arquivos de log. Essa importância é dada em situações de, por exemplo: entrada de um novo membro na equipe; identificação dos envolvidos; transferência de tecnologia; reuso do código-fonte; necessidade de evolução de requisitos ou do próprio sistema.

Algo a ser destacado é o fato de, frequentemente, ser uma demanda prioritária para os usuários de sistemas de gestão da informação. Isto também pode ser observado junto às equipes de desenvolvimento, que trabalham em cooperação à outras, assim como no tratamento das obrigações legais vigentes a serem atendidas pelo sistema.

M03: "a gente chegou ao ponto de, é? às vezes fazer um printscreen de tela, e essa era a documentação (...) então é preciso ter uma documentação, e que essa documentação, ela seja simples; é, mas tenha a sua., atenda a sua...né...ao seu propósito. (...) principalmente quando a gente está fazendo transferência de tecnologia em que várias outras instituições estão vendendo aquilo ali... então, é preciso ter uma documentação...".

M04, quando questionado sobre a importância de saber como as pessoas são relacionadas, quais os papéis: "como eu me mudei há oito meses pra cá, no momento que eu cheguei na equipe, eu não conhecia ninguém, então realmente nesse momento foi extremamente útil (o registro de papéis desempenhados)".

M07: "a rastreabilidade é sempre relevante, mas ganha uma importância muito maior,

sempre que os projetos sejam críticos ou envolvam obrigações legais e/ou normativas de conformidade".

D01: "Quando viu a possibilidade de exportar o software, percebeu a preocupação (com a documentação)".

Há um manancial de estratégias adotadas pelos profissionais, como: estabelecimento de um modelo; uso de comentários e *tags* no artefato gerado; criação de diferentes tipos de *links* de rastreabilidade (pré, inter, pós, horizontal e vertical) dos artefatos; registro de reuniões em atas; uso e customização de ferramentas; feitura de planos de ação e de contingenciamento.

M02: "quando você altera uma regra de negócio, altera um requisito. Você precisa ir no código, ir lá e dá um *ctrl+shift+g*, buscar referência daquele método, quem é que está usando aquele método... (...) têm os testes que estão relacionados a esses requisitos... é... e o código que está relacionado a esse requisito, mas eu não tenho a dependência entre eles...".

D01: "então era uma tarefa para cada desenvolvedor, com os links pra os requisitos...".

D02: "quando eu crio esse pull request, eu posso associar issues a esse pull request... entendeu?! então, tipo, se eu entrar na issue eu vou ver os pull requests que estão linkados aquela issue... se eu for no pull request eu vou ver as issues que estão linkadas ao pull request...".

D05: "Então, normalmente, o detalhamento do requisito era originado do registro de tarefas do tipo ?Reunião?, ou algo assim. Mas nem sempre o refinamento do requisito era relacionado com a tarefa que deu origem a ele".

D06: "Geralmente é relacionado com o requisito origem através de uma menção usando hashtag e o número da issue. Às vezes uma nova issue é criada para dar um overview de forma atualizada e mais limpa".

M07: "a gente tem que registrar que mudou (uma funcionalidade), é... tem que registrar na ferramenta que mudou".

RQ2 - Rastreabilidade: "Como as informações de rastreabilidade podem ser úteis no dia-a-dia dos agilistas?"

A rastreabilidade em metodologias ágeis é um problema recorrente na Engenharia de Requisitos. Frequentemente, quando é feito uso de ferramentas computacionais não há relacionamento entre os artefatos produzidos. Por outro lado, quando os instrumentos

utilizados envolvem apenas processos manuais, algumas informações importantes na perspectiva do gerente são perdidas. São inerentes a esses processos manuais certa dificuldade de historiar e armazená-los ao longo do ciclo de vida dos sistemas, assim como rastreá-los.

M03: "*assim são várias ferramentas na verdade pra você ter informações; então assim, isso eu acho que atrapalha um pouco...*".

M04: "*existe sempre o componente humano e, muitas vezes, quando você tem muitas mudanças e ele não é obrigada a fazer isso (rastreabilidade) (...) a pessoa sempre vai procurar o que é menos trabalhoso pra ele fazer, então ele acaba pulando. Ao invés de informar tudo o que tá acontecendo, só informar grupos de tempos em tempos. E isso depende de memória - as vezes ele perde informações importantes*".

M04: "*só que isso não tá assim, tudo numa única ferramenta; tem uma parte numa coisa, uma parte na outra, então a pessoa tem que, que sair procurando*".

Outra dificuldade presente nas equipes ágeis é o custo atribuído de realizar a rastreabilidade ao passo que a importância só percebida no momento de buscar informações relevantes, e já há um processo estabelecido, um volume grande informação e artefatos produzidos.

Algumas equipes contornam essa situação relacionada aos custos iniciais elencando apenas uma parte dos artefatos para serem rastreados, por entenderem que os custos podem ser compensados durante o ciclo de vida do sistema.

D07: "*mas levando em conta os projetos que pedem, o custo é compensado sim. (...) Tendo os problemas rastreados, é mais fácil de atacá-los, de analisar e buscar soluções*".

M01: "*a gente sabe que compensa (a rastreabilidade) né, a gente já leu até artigos que demonstrou isso...*".

M02: "*é, hoje eu não tenho isso... por mais que eu saiba que isso é importante, eu não tenho essa ferramenta...*".

A cultura de seus membros e a existência de uma política de rastreabilidade mostraram ser fatores preponderante para uma bem sucedida rastreabilidade de artefatos. Considerando a volatilidade da informação, quando não há um registro, informar os rastros da informação pode ser valoroso na ocasião de perda de um membro ou identificar possíveis impactos de uma modificação, por exemplo.

M03: "*então a gente tenta ao máximo educá-los, para que eles façam essas questões (informar rastros sobre mudanças), porque assim não tem um documento de aceite*".

D04: "sim, por causa de nós. (...) porque assim, é... primeiro a falta de costume (de manter a rastreabilidade) (...) então, este é o nosso maior problema, por isso que eu disse: por causa de nós. a nossa falta de hábito".

D05: "acho que as principais dificuldades são (de manter a rastreabilidade): (...) e a segunda, a implantação de uma cultura que considere a importância da rastreabilidade para a melhoria da gestão do processo de desenvolvimento de software".

D05: "Existia uma dependência muito forte da ?boa vontade? do autor do refinamento do requisito para que o registro fosse feito da forma apropriada".

A experiência de não possuir rastreabilidade entre os artefatos existentes foi relatada como negativo em vários sentidos, por exemplo, para tomada de decisão, iteração, aumento nos custos. Enquanto que, ao contrário, os benefícios da rastreabilidade são ressaltados por suas reais possibilidades de, por exemplo, melhor acompanhamento da evolução do sistema e do envolvimento da equipe, tomada de decisão mais acertada, priorização objetiva da demanda e redução do retrabalho.

M01: "eu acho que a vantagem é o acompanhamento, não é?! saber se a pessoa está com alguma dificuldade, não é?! o quê que produziu...".

M02: "porque senão fica sempre nessa área de você fazer um retrabalho, buscar essa informação, e fica muito pessoal, não é?! assim, se eu sair, alguém que entrar vai precisar fazer essa pesquisa novamente...".

M04: "ai ajuda (na resolução de problemas) porque a gente precisa identificar, é? qual foi a fonte, quem foi a fonte do requisito, pra voltar nessa pessoa e tirar as dúvidas, certo?! (...) como é muito grande, muito complexo, então, não dá, não cabe na cabeça de uma pessoa; tem que ter realmente uma forma de você descobrir quando, em que (trecho incompreensível) e como mudou (uma funcionalidade)".

D01: ?Acho que era bom pra isso, pra controlar as alterações que aconteceram no requisito e quem foi o responsável por cada alteração, tirar dúvidas e tal (...) para tirar alguma dúvida, pra... até pra evoluir aquilo ali não é?!?.

D03: "porque se eu precisasse olhar, por exemplo, (...) uma história que já foi concluída, ou...sei lá, minha história relacionada a uma outra que já foi concluída; isso (os impactos) ficava (registrado)".

D04: "porque eu conseguiria ter um controle maior sobre tudo o que está acontecendo nesse momento".

Diferentes estratégias para obter informações sobre a vida do artefato, a partir dos *links* de rastreabilidade, foram elegidas, de acordo com o tipo de artefato de interesse. São estratégias de uso comum, por exemplo, o emprego de: ferramentas computacionais; versionamento dos artefatos; mecanismos automatizados de busca, filtragem ou mineração; mesmo nome entre os diferentes artefatos relacionados; relacionamentos entre a origem e possíveis refinamentos, além do responsável pela solicitação, criação, alteração e pelo aceite.

Outra perspectiva, seriam os casos onde não há, ou é feita parcialmente, a rastreabilidade de artefatos, onde os profissionais são impelidos a operar a partir do uso de mensagens de erro; buscas manuais nos registros textuais ou tags dos artefatos; contato verbal ou textual com o responsável pela solicitação, criação, alteração ou pelo aceite.

M02: "*quando você altera uma regra de negócio, altera um requisito... você precisa ir no código, ir lá e dá um ctrl+shift+g, buscar referência daquele método, quem é que está usando aquele método...*".

M03: "*então sempre que a pessoa vai editar o documento, aí fica lá no histórico do próprio (nome da ferramenta), a pessoa, que editou. então é a última versão de quem editou, e quem editou anteriores tá lá também*".

M04: "*a gente precisa identificar é? qual foi a fonte, quem foi a fonte do requisito, pra voltar nessa pessoa e tirar as dúvidas, certo?!*".

D02: "*também a gente faz uso de flags, ne, de labels nas issues... tipo, a gente cria labels customizadas (...) então, tipo, eu consigo fazer essas filtragens...*".

D03: "*então, acontecia, as vezes, de uma outra pessoa pegar aquele bug que eu não pude terminar, em uma outra sprint... então, ela já tinha um histórico ali, do quê que tava sendo discutido antes... (...) toda vez que a gente fazia um commit, abria um PR, a gente tinha que pôr o ID relacionado a task que a gente tava fazendo, aí eles linkavam e atualizavam o status e tal; tipo, quando a gente fazia o merge o pull request, lá no Jira falava: 'ah, tipo, agora você pode fechar essa task que esse...PR tá fechado'*".

D03: "*não, isso (analisar impactos) a gente sabe através de olhar código. (...) e as vezes nem no código...*".

RQ3 - Comunicação: "*Como a escolha dos instrumentos para comunicação entre os stakeholders podem afetar a rastreabilidade?*"

A acepção da comunicação e da colaboração é social. Sendo que ambas são consideradas como instrumentos para a rastreabilidade.

radas motores para as equipes ágeis e, nesses casos, caracterizadas flexibilidade e informalidade dos meios e instrumentos. Quando são utilizadas ferramentas computacionais, há o registro das informações, auxiliando o processo de documentação e a busca por essas informações.

Em algumas metodologias, a priorização e a codificação são reflexos diretos da dinâmica de comunicação. Dessa forma, a disponibilidade para a dissolução de dúvidas, o informe de mudanças e o fornecimento de *feedbacks* são essenciais aos bons resultados na evolução de sistemas. A experiência estende a percepção das boas práticas de comunicação à aplicação de filtros relacionados aos papéis desempenhados. Ressaltando-se tal ação é relevante à medida que os meios e instrumentos são mais informais, podendo aumentar a distorção da mensagem.

M04: *"muitas vezes o registro é a aprovação do cliente da mudança, que foi implementado; porque pode ser, por exemplo, que a dúvida tenha sido tirada por uma ligação no telefone".*

D02: *"porque o que eu codifico, é o reflexo da comunicação que eu tenho com que mantém o produto... entendeu?! então eu preciso entender o que tá se passando pra poder escrever o código..."*.

D03: *"se tem alguma coisa, você vai no Slack do grupo (de funcionários) e resolve..."*.

D04: *"... porque ele fez questão de anotar. e ele me mandou três e-mails, não, não três e-mails, mas eu tenho a lista de e-mails onde toda essa comunicação rolou..."*.

D05: *"nunca tive a consciência de estar usando rastreabilidade nos projetos que participei. Os registros detalhados nos logs das tarefas sempre foi para facilitar a comunicação entre os membros da equipe. Hoje sei da importância da rastreabilidade no gerenciamento de projetos"*.

RQ4 - Visualização: *"Como as visualizações da rastreabilidade influenciam na tomada de decisão para evolução de sistemas?"*

É imperativo o planejamento da visualização da transformação dos dados obtidos a partir da rastreabilidade dos artefatos em informações relevantes ao entendimento pelos interessados do sistema e dos processos relacionados. A razão de haver esse planejamento está no princípio de uma visualização somente será adequada quando considerados o volume e a diversidade dos tipos de dados existentes no desenvolvimento de sistemas. Ademais, dada natureza dos ambientes ágeis, é crucial considerar a dinamicidade e agilidade desses dados e das decisões a serem tomadas.

M02, quando questionado sobre a importância da visualização de impactos: "*se um sistema é desenvolvido baseado nos requisitos, então, é essencial eu saber o impacto da alteração de um requisito, pra saber quais são as funcionalidades, não é?! então, se as funcionalidades agregam valor ao usuário, não é?!... eu sou um usuário, eu uso o quê?! Eu uso funcionalidades, funcionalidades que vão atender aos meus requisitos, às minhas necessidades... então, eu acho vital saber o impacto de uma alteração, porque uma alteração que eu faço, em um ponto pode afetar o usuário de outro ponto, o requisito de um usuário em outro lugar... seja ele funcional ou não funcional...*".

M04: "*porque facilitaria navegar nas informações. uma das grandes razões porque as pessoas não voltam para... não buscam essas informações, é porque elas são difíceis de garimpar*".

D05: "*Acredito que visualizações desses tipo (de relacionamentos) podem ser úteis para tomar decisões do tipo: 'Qual o impacto de uma determinada mudança no projeto?'; 'Qual o grau de acoplamento de um determinado módulo?'; etc.*".

Esse preceito mínimo de planejamento na gestão da informação é negligenciado, embora os desenvolvedores e os gestores relatem diferentes preocupações, como a imparcialidade, a conformidade e a corretude das informações visualizadas. Na maioria das situações, desenvolvedores e gerentes encontram dificuldades oriundas, por exemplo, da falta, inadequação ou complexidade dos artefatos ou das informação nestes e limitações das ferramentas utilizadas.

M01: "*(tem) esse desafio de mostrar visualmente, acompanhar projetos muito grandes, com vários usuários não é, vários participantes vamos dizer... então assim, eu sei que tem, mas eu nunca consegui acompanhar não...*".

M02: "*tenho como saber qual foi o desenvolvedor que implementou aquele requisito, qual foi o testador que testou aquele requisito... mas não sei assim, pra mim quando fala em visualizar... eu sempre penso em algo... (...) algo gráfico... (...) tenho que buscar... não tem nada que eu posso gerar e os requisitos. (...) eu queria principalmente fazer uma ferramenta gráfica que pudesse identificar os caras que tivesse atuando, e sem precisar ir procurando a informação no sistema...*".

M04: "*quando você tem que descobrir no código qual é o requisito que ele está implementando, pelos comentários do desenvolvedor, isso requer muito, assim, você tem que abrir o código, tem que ler... (...) isso é quase trabalho investigativo*".

D01: "*não tinha um relatório disso (de informações de rastros), não tinha um botão*

pra fazer isso (buscar rastros), mas eu acho que dava (obter a informação) fuçando, entendeu? pelos logs... ".

D04: *"eu gosto muito de ver aqueles desenhos (com rastros de informações e artefatos), muito, mas eu não tenho paciência pra fazer aquilo não".*

Os benefícios de uma visualização adequada podem estar relacionados com: a maior facilidade de compreensão dos interessados; a identificação do progresso, da atuação e da capacidade de produção das equipes, dos problemas e dos impactos possíveis das ações em análise; a priorização, a produção e a alteração dos artefatos de forma mais rápida e com maior qualidade; e a gestão mais eficaz e eficiente dos artefatos dos sistemas e das equipes.

M04: ?então, se a ferramenta conseguisse facilitar isso; você vai olhar, clica, ela já aponta, já leva pra o local onde tem a informação? (...) ?pra mim é muito mais fácil trabalhar com coisas visuais do que com? eu perco menos tempo do que abrindo texto e lendo ele para localizar qual foi o módulo que foi afetado, ou quem era a pessoa envolvida?.

M05: *"sabemos que a visualização ajudaria para suportar análise de impacto".*

M06: *"a visualização é muito importante para tomada decisões de melhoria de processos, forma de trabalho, análise de equipe, qualidade de serviço prestados, alinhamento da documentação com o produto entregue".*

M08: *"por ter uma relação entre eles, cada alteração poderia ser visualizada diretamente em um único plano (...) na verdade gostaria de ter isso implantado (...) seria útil para projetar o que seria afetado em uma mudança que desejo fazer".*

D01: *"... a visualização pode ajudar no... assim, pode ajudar o gestor a tomar uma decisão, não digo mais acertada, assim, mas eu acho que mais, de uma forma mais rápida... porque pode ter uma visão mais geral... (...) eu acho que dependendo da forma como eu vejo aquilo, eu posso tomar uma decisão diferente...".*

D03: *"acho que mais f* ainda ia ser se tivesse como mensurar isso (visualização de impactos) no código, do que na história...".*

D06, quando questionado sobre a importância da visualização: *"entender edge-cases no código a nível de infraestrutura e negócio seria muito interessante. Avaliar impacto de mudanças acho que seria o mais importante para mim".*

Para desfrutar desses benefícios e considerando um cenário, onde há diferentes papéis e interessados, além de diversas intenções, houve certa convergência nas sugestões de

visualizações desejadas à incorporação do cotidianos das equipes. Conjeturam-se que as sugestões ou as necessidades indicadas podem estar limitadas ao horizonte vislumbrado, e não ao horizonte possível na temática visualização das informações da rastreabilidade dos artefatos em metodologias ágeis.

M02: *"eu queria principalmente fazer uma ferramenta gráfica que pudesse identificar os caras que estivesse atuando, e sem precisar ir procurando a informação no sistema... (...) eu acho que uma das ideias seria um mapa dos requisitos... da dependência dos requisitos... (...) não sei... eu tenho uma árvore... enfim, qual fosse a disposição... mas em que eu pudesse identificar os requisitos e saber imediatamente onde é que eles estão relacionados... e aí, tivesse a informação, também nessa árvore, de quem foi que atendeu aquele requisito, de quem foi que... de todo o desenvolvimento daquele requisito..."*.

M06: *"(visualizar) o que está sendo entregue por release e para quem e com quais artefatos. roadmap de sistemas, anual, mensal, trimestral por sistema"*.

M07: *"visualizar as relações: código x requisitos. (...) código x testes. (...) testes x requisitos. (...) artefato x responsável. (...) Atividade de processo x responsável"*.

D02: *"um mapa mental, não sei... de alguma coisa assim específica"*.

D05: *"acredito que a utilização de gráficos que utilizam grafos em sua construção, pode ajudar na visualização dos relacionamentos. Porque através da utilização de nós e arestas (elementos fundamentais em um grafo) pode-se descrever claramente como dois ou mais objetos se relacionam"*

3.4 Discussão

Os resultados desse estudo apontaram as necessidades e estratégias de gerentes e de desenvolvedores agilistas em um cenário, onde há: i) um ambiente propício e flexível às mudanças e evolução dos sistemas; ii) muitos interessados e envolvidos nesse processo; iii) uma quantidade mínima bem pequena de artefatos exigidos nas metodologia utilizadas; iv) pouca ou nenhuma visualização da rastreabilidade dos artefatos existentes de um sistema.

Nos ambientes ágeis de desenvolvimento de sistemas é bem comum as informações existentes serem documentadas de forma distinta e informal, quando comparadas aos ambientes tradicionais, que privilegiam as especificações e os modelos. O código-fonte recebe um maior destaque para os agilistas como artefato de documentação do projeto. Entretanto o código-fonte não é um artefato legível para a maioria dos interessados, como

clientes e usuários.

Outras divergências a serem ressaltadas são concernentes aos procedimentos e prioridades dos desenvolvedores e gerentes agilistas. Ponderando essas divergências sob a ótica dos valores e princípios do (BECK et al., 2001b), a prioridade base está: i) na comunicação e nas interações mais valorizadas que processos e ferramentas; e ii) na colaboração com o cliente mais do que negociação de contratos.

Assim, muitos artefatos são gerados a partir das interações, colaboração e, principalmente, comunicação dos envolvidos. A falta ou o uso de instrumentos inadequados ao registro dos dados desses artefatos podem trazer riscos aos projetos e dificultar o rastreamento das informações. Isto impacta diretamente na tomada de decisão às mudanças, na entrada ou na saída de membros à equipe, no envolvimento dos interessados e na comunicação entre os envolvidos, além de aumentar os custos de produção e manutenção. Essa circunstância é corroborada por Espinoza e Garbajosa (2011) e pelas entrevistas realizadas.

Os desenvolvedores e gerentes, que utilizavam ferramentas computacionais em suas atividades, revelaram também a ordem de preferência na geração e na observação dos artefatos durante o processo de evolução dos sistemas: 1º código-fonte; e 2º registros de logs e da comunicação nas ferramentas. Todavia, quando foram questionados pela ordem de preferência na busca de informações frente aos artefatos mencionados figurou-se o próprio conhecimento tácito.

O risco da utilização desse tipo de conhecimento tácito é a não garantia de que todas as informações serão lembradas e consideradas diante de uma decisão. A justificativa para tal comportamento foi feita julgando as dificuldades encontradas. As principais dificuldades declaradas foram: i) a inexistência, a desatualização ou a inconsistência dos dados; ii) a utilização de diferentes ferramentas com o mesmo propósito; e iii) o esforço necessário para mineração desses dados.

Ter a informação sobre a vida de um artefato registrada foi considerado relevante na maioria dos casos, contudo percebeu que havia uma certa dificuldade em encontrar essas informações, muitas vezes por estarem registradas em ferramentas que não traziam o dado de forma direta. Enquanto que o uso concomitante de diversas ferramentas com a mesma intenção geram maior esforço de alimentação e manutenção dos dados, insatisfação dos responsáveis pela necessidade de retrabalho na inclusão dos dados, além da incerteza sobre a consistência desses dados.

Todo esse esforço pode ser minimizado, como visto, a partir das informações da rastreabilidade dos artefatos. Entretanto, os relatos levaram a constatação que sua importância, na maioria das ocasiões, foi percebida somente na presença de grandes volumes de dados e ferramentas envolvidas na análise, nas entradas ou nas saídas inesperadas de membros da equipe, nas mudanças, nos impasses entre opiniões e nas decisões mais graves do projeto ou em sistemas críticos.

Dessa forma, possuir os dados do desenvolvimento do projeto, a presença de ferramentas e a realização da rastreabilidade dos artefatos, isoladamente, podem não ser fatores de um bem sucedido projeto. As evidências alvejaram para conveniência de possuir um planejamento do uso das ferramentas utilizadas, da rastreabilidade dos artefatos e da adequada visualização. Visualização da informação, que segundo Spence (2014) e Li e Maalej (2012), deve considerar também o objetivo final para o qual servirá as informações.

Além das necessidades vistas da visualização da rastreabilidade em ferramentas, percebeu também nas entrevistas outros pontos a serem tratadas por estas que dão apoio à gerência de projetos. Com a ideia trazida da TF de codificar e analisar todo dado trazido da entrevista, algumas categorias além das relacionadas a visualização e rastreabilidade foram identificadas. Essas categorias inspiraram à criação de heurísticas que podem ajudar na avaliação das ferramentas de gerenciamento de projetos. Entretanto, como este trabalho foca na rastreabilidade e visualização, nem todas as heurísticas foram utilizadas. A Tabela 6 apresenta as categorias com as heurísticas levantadas.

Tabela 6: Relação de categorias e heurísticas

Categorias emergidas do TF	Heurísticas
Demandas	Identificação das demandas
	Definição de critérios de priorização
Diferenças	Flexibilidade as características dos projetos
	Flexibilidade as características das equipes
Documentação	Produção e transformação da informação
	Assistência ao tratamento das informações
	Armazenamento e base de conhecimento
	Apoio à recuperação da informação em diferentes artefatos
	Auxílio ao uso da informação
Gerenciamento de Equipe	Definição de perfis ou de papéis
	Flexibilidade à entrada ou à saída de membros

	Assistência ao trabalho colaborativo ou cooperativo Acompanhamento das atividades e do desempenho
Gerenciamento de Projetos	Sustentação ao planejamento do projeto Acompanhamento do projeto Gerenciamento de conflitos entre artefatos ou versão dos artefatos Assistência a múltiplos projetos Auxílio ao controle dos recursos, qualidade e riscos
Metodologias de desenvolvimento (Ágil)	Flexibilidade a criação e a manipulação de diferentes tipos de artefato Suporte a metodologia utilizada
Participação dos Stakeholders	Auxílio na comunicação Gerenciamento das informações de participação
Rastreabilidade	Apoio aos tipos de rastreabilidade (pré, inter e pós - rastreabilidade) Auxílio à rastreabilidade de propriedades de um artefato Assistência a rastreabilidade do projeto
Repositórios	Uso de repositório e sistemas de controle de versão
Requisitos	Apoio à especificação Assistência a evolução e reuso dos requisitos
Visualização da Rastreabilidade	Adaptabilidade aos pilares definidos da informação (problemas, o quê?, quando?, quem?) Assistência à seleção de artefatos e à identificação de repercussão das ações Apoio à tomada de decisão para planejamento e controle dos artefatos Apoio à tomada de decisão para gestão de recursos, incluindo humanos e tempo Flexibilidade nos pontos de vista do projeto Auxílio à manipulação e à naveabilidade nas informações de rastreabilidade Redução na carga cognitiva (por busca de informações)

3.4.0.1 Ameaças à validade e as limitações do estudo

As ameaças à validade e as limitações inerentes a esse estudo estão relacionadas ao contexto de aplicação dos resultados ser apenas àqueles explorados nesta pesquisa. A causa está subordinada ao fato dos códigos, categorias e teorias terem emergidos a partir dos dados coletados de forma empírica no mundo real.

Na sequência, com os dados levantados, analisados e discutidos, foram realizados estudos relacionados a visualização de informação de rastreabilidade, na intenção de verificar as possibilidades para geração de uma visualização, conhecendo os problemas relatados, os possíveis dados e as necessidades dos entrevistados. Portanto, seguindo a ideia de Spence (2014) e Li e Maalej (2012), antes de analisar as possíveis representações, foram levantados dados sobre problemas, o que visualizar e quando visualizar para então sugerir como visualizar.

3.5 Considerações

Com a realização desta pesquisa foram encontrados insumos para investigação de visualizações adequadas ao perfil de gerentes e líderes de equipes de desenvolvimento de *software*. Em sendo assim, no próximo capítulo, são demonstrados o planejamento e a execução de uma pesquisa bibliográfica para identificação e recomendação de visualizações a esse público alvo tendo como referencial os achados destas entrevistas.

4 Visualização da Informação da Rastreabilidade

Algumas técnicas de visualização foram desenvolvidas para suportar a representação de informações de rastreabilidade (MERTEN; JÜPPNER; DELATER, 2011). Matrizes, árvores e grafos são exemplos destas. Contudo, não foram encontrados estudos que levantassem o que é necessário para a criação de uma visualização em um dado contexto, como, por exemplo, um cenário ágil que produz artefatos de forma diferente ao tradicional - onde podem ser vistos no Apêndice A; além de coletar as necessidades da indústria.

Como tratado no capítulo de Fundamentação Teórica, o objetivo da visualização de informação não é somente de criar imagens, segundo Jacko (2012) . Para isto, não havendo uma teoria de visualização existente para ajudar na compressão de grandes volumes de dados e transformá-los em imagens, é necessário identificar as características das pessoas que usarão a visualização (MERTEN; JÜPPNER; DELATER, 2011; SPENCE, 2014), assim como os seus problemas, para então ilustrar as possíveis representações dos dados e sua apresentação no espaço e tempo (SPENCE, 2014).

Além disso, como este trabalho trata da visualização de informações acerca da rastreabilidade e artefatos produzidos em cenários ágeis, ainda como citado no capítulo de Fundamentação Teórica, também é importante considerar os conceitos principais citados por Li e Maalej (2012) e Winkler (2008), como: (i) a informação da rastreabilidade (o que visualizar) e (ii) o contexto da tarefa (quando visualizar); e (iii) a técnica de visualização (como visualizar), que é respondido neste capítulo. Sendo assim, as características e os problemas enfrentados por um perfil de usuário somados aos conceitos de informação da rastreabilidade (i) e contexto da tarefa (ii) são considerados como pilares aliciadores para a sugestão de visualizações adequadas ao contexto deste trabalho, que seria o conceito de como visualizar (iii).

Para a obtenção das informações referentes a esses pilares, foram considerados os resultados das entrevistas realizadas, sendo estes analisados e discutidos no capítulo an-

terior. Este capítulo, portanto, trata da técnica de visualização, o “como visualizar??”, diante do cenário de necessidades e interesses percebidos através dos dados trazidos da indústria. Considerando isto, neste capítulo, será sugerida uma visualização da informação adequada às necessidades encontradas de acordo com perfis de liderança e gerenciais. A Figura 7 apresenta os pilares aliciadores para a sugerir a visualização adequada no contexto presente.



Figura 7: Pilares aliciadores da informação que podem impactar na sugestão da visualização da informação da rastreabilidade.

4.1 Propostas de Visualização da Literatura considerando os Pilares da Informação

Os pilares aliciadores da informação, "o quê?" e "quando?", foram levantados, segundo as sugestões de Spence (2014), Li e Maalej (2012) e Winkler (2008) e a definição do público alvo - pilares "quem?" e "problemas", para a apresentação das propostas de visualização e discussão sobre a visualização adequada. Assim, nesta seção, são apresentadas respostas relacionando os pilares da informação e as visualizações existentes mais adequadas.

4.1.0.1 Pilares Aliciadores da Informação

A definição do público alvo, pessoas com perfis de liderança e gerência, foi feita a partir da identificação das necessidades desses perfis no uso rastreabilidade em ambientes ágeis. Esta percepção veio da experiência vivida e da literatura discutida, no Capítulo 2. Além disto, havia a intenção de melhor delinear as propostas de visualização de acordo com a problemática, o cenário real e coletar mais necessidades e dificuldades enfrentadas a serem consideradas. Essas informações foram obtidas com a realização de entrevistas, como relato no Capítulo 3.

Os problemas comumente percebidos durante as entrevistas estão relacionados ao apoio a decisões a serem tomadas diante de um cenário, que está em constante mudança, como é possível notar nos ambientes ágeis. Esses ambientes são ditos ágeis, justamente, pelo fato de seus métodos e processos terem a disponibilidade em abraçá-la (WILLIAMS; COCKBURN, 2003; PUNTEL; PRASS, 2010). As demandas (solicitações de alterações, correções ou novas funcionalidades) provindas dos clientes, que foram comentadas pelos entrevistados, apareciam de forma frequente, visto que há uma participação notória de clientes e usuários nestes ambientes, não sendo somente na fase inicial do projeto. Tal situação, segundo os entrevistados, gera a necessidade de um controle maior dessas modificações, como, por exemplo, desde a decisão em realizar as alterações até o envolvido que a implementou.

Além das modificações a serem realizadas nos artefatos, também foram abordadas mudanças associadas à gestão de tempo e de recursos, incluindo humanos. A partir dessas discussões, foi possível notar as diferentes preocupações do universo dos entrevistados, como, por exemplo, a alteração de membro na equipe, concentração de conhecimento em um único integrante, e mudanças no planejamento das atividades. Por conseguinte, as decisões a serem tomadas aqui tratadas, vão além daquelas solicitadas pelos clientes; elas também refletem em decisões para um plano, bem como decisões voltadas à gestão da equipe.

Para um maior controle das modificações, uma série de informações foram consideradas importantes à tomada de decisão pelos gerentes e líderes, como por exemplo: responsáveis, histórico de alterações, situação em caso de tarefas e esforço dado por um integrante. Tais informações são necessárias a depender do tipo de decisão imposta, se é relacionada a uma alteração no sistema, mudança de planejamento ou de equipe.

As informações relacionadas a alterações no sistema se cruzam com as encontradas

na literatura, onde Marcus, Xie e Poshyvanyk (2005) afirmam que ao lidar com links de rastreabilidade, é preciso rastrear várias informações. Segundo eles, origem, destino e os próprios *links* possuem propriedades importantes. Assim, para origem e destino, esses autores sugerem, por exemplo: nome do artefato, tipo, tempo de criação, tempo de atualização, versão; enquanto que, para os *links*: método de descoberta, quem indicou como o *link* foi identificado, tempo de criação e atualização, autor, descrição da lógica, versão, situação, dentre outras informações.

A necessidade de uma visualização de informações da rastreabilidade foi bastante abordada, principalmente para aqueles participantes que julgaram possuir projetos com uma relevante importância ou, ainda, aqueles que possuíam projetos com requisitos legais, ou seja, que envolviam leis e, por conseguinte, a necessidade de garantir a conformidade legal. Para estes casos, foi dada ênfase pelos entrevistados na obrigatoriedade de armazenar e manter as informações de rastreabilidade, e a própria consulta a elas, quando necessário. Por outro lado, para os outros participantes havia a necessidade da visualização de informações da rastreabilidade, mas não era dado o destaque ou importância que aquela informação traz para todos os membros da equipe. Observa-se que muitas vezes essas informações eram cadastradas sem saber que benefício elas poderiam trazer ao final e, assim sendo, elas eram pouco consultadas. Com isso, em alguns momentos foi possível perceber que havia um grande foco em codificação e produção, seguindo a risca um prazo curto, deixando de lado o mantimento ou até mesmo o armazenamento dessas informações, como por exemplo, o próprio caso do cadastro de uma solicitação.

Nota-se ainda que algumas vezes as informações de rastreabilidade não eram mantidas ou armazenadas devido a urgência de uma solicitação e da pouca visibilidade da informação registrada. Em muitos momentos, as decisões eram tomadas através do conhecimento tácito do gerente ou líder, devido a dificuldade de capturar a informação, que necessitava. Em contrapartida, quando foi levantada a importância de se obter aquela informação com a ajuda de uma visualização, a grande maioria julgou necessário para o progresso e desenvolvimento do sistema. Os próprios participantes apontaram os benefícios que uma visualização poderia trazer, além de fazerem sugestões diretas de visualização, que gostariam de ter.

O trabalho de Marcus, Xie e Poshyvanyk (2005) trazem uma afirmação que é possível percebê-la nas entrevistas deste trabalho. Para esses autores, os atributos provindos de artefatos e *links* da rastreabilidade fornecem informações úteis ao usuário em muitas tarefas de engenharia ou manutenção; portanto, a representação dessas informações em

conjunto com os *links* é bastante desejável. Com as entrevistas, foi possível verificar a real necessidade da visualização das informações de rastreabilidade diante de tantas decisões a serem tomadas por profissionais dos perfis de interesse desta pesquisa.

Percebeu-se que todo o cenário apresentado está em torno das alterações e mudanças constantes, que a própria literatura trás como um fator presente quando se trata de ágil (BECK et al., 2001b; ERICKSON; LYYTINEN; SIAU, 2005; WILLIAMS; COCKBURN, 2003; NE-RUR; MAHAPATRA; MANGALARAJ, 2005), isto pode ser observado ao definir agilidade no Capítulo 4, Seção 2.1. Outro ponto que é importante ressaltar, é a ajuda que a visualização das informações de rastreabilidade de *software* pode prover diante deste cenário de constantes decisões a serem consideradas. Como citado no capítulo anterior pelos próprios entrevistados, há necessidades referentes à visualização dessas informações, que ajudam a analisar possíveis impactos diante de uma alteração. Este tipo de visualização é bastante desejada, entretanto as ferramentas utilizadas por estes profissionais não forneciam essas informações de forma visual.

As necessidades referentes a visualização de informações de rastreabilidade eram justificadas com a facilidade no ataque aos problemas gerados pela mudança e também com a simplicidade na busca por soluções. Ainda, foram comentados transtornos ocorridos em situações onde foi considerado apenas o conhecimento tácito para solucionar problemas, como esquecimento de informações importantes relacionadas a possíveis impactos ou soluções anteriores para problemas similares. Então, mesmo tendo essas informações cadastradas, a dificuldade em obtê-las fazia com que muitos entrevistados buscassem informações com a própria equipe ou no conhecimento que já tinha. Isto, como comentado, tornava-se um grande problema, quando se tratava de um funcionário novo na equipe, por exemplo.

A Tabela 7 apresenta as respostas, de forma resumida, às perguntas presentes na Figura 7, que são necessárias, segundo a literatura, para sugerir uma visualização da informação. As necessidades apresentam uma amostra do que foi discutido nas entrevistas. Tais informações foram coletadas dessas entrevistas realizadas a partir de uma análise dos dados utilizando a Teoria Fundamentada.

Necessidades, como: i) análise do trabalho desenvolvido por um integrante, ii) análise do esforço de cada integrante, iii) dados de produção em um período de tempo definido e iv) quem precisa ser informado sobre uma alteração estavam mais voltadas para decisões relacionadas à gestão de tempo e de recursos, incluindo humanos. Estas foram identificadas em casos de constatação de conhecimento concentrado em um único participante, onde a

**Tabela 7: Respostas aos pilares da informação
Informações provindas das entrevistas**

Problema	Falta de um apoio adequado, que torne o processo de tomada de decisão mais ágil e seguro uma diante alterações de artefato, planejamento e equipe;
O que visualizar (necessidades)	Análise de impactos (entre os artefatos), que vão ser ocasionados pela alteração solicitada Histórico de alterações feitas em um artefato, com datas, motivos e versão do artefato Stakeholder responsável por uma mudança no artefato Análise dos stakeholders, que precisam ser informados de uma alteração solicitada Análise de dependências entre artefatos produzidos Análise das situações dos artefatos Avaliar cobertura de testes nos requisitos e código fonte Análise do trabalho desenvolvido por um integrante da equipe Análise de esforço de cada integrante da equipe por tempo O que foi produzido em um dado período de tempo, com seus relacionamentos, tempo e responsáveis
Quando visualizar	A visualização será feita antes de tomar a decisão, seja ela a nível de planejamento e controle dos artefatos ou ainda a nível de gestão de tempo e de recursos
Quem irá visualizar	O suporte da visualização é pensando em cargos de gerência e liderança, pois estes estão mais propícios a tomarem decisões

informação das atribuições de tarefas ajuda na: percepção da necessidade de disseminação de conhecimento; verificação na sobrecarga de esforço, onde a análise do esforço ajuda a definir as próximas atividades dos membros da equipe; e consulta à produtividade em um certo período, em que fornece um apoio para a estimativa de prazos e custos em outras atividades.

Já outras necessidades foram relacionadas a decisões voltadas ao planejamento e controle de artefatos frente às mudanças, sendo elas: i) histórico de alterações realizadas dado um artefato; ii) verificação de dependências entre os artefatos produzidos; iii) análise da situação de um artefato; e iv) avaliação de cobertura de testes nos requisitos e código. Nestes casos, a identificação dessas necessidades se deu diante de um cenário de constante mudança dos artefatos devido a grande demanda trazida pelos donos do produto. Tais informações foram tratadas como importantes para os momentos de tomada de decisão da alteração no artefato.

Outras necessidades ainda agruparam-se para dar apoio em ambas situações, a gestão de tempo e recursos incluindo humanos, e planejamento e controle de artefatos. Estas foram: i) *stakeholders* responsáveis por uma mudança; ii) *stakeholders* envolvidos no artefato; e iii) análise de impactos de uma alteração. A identificação dessas informações também foram realizadas a partir do cenário de constante mudança nos artefatos. Estas informações ajudavam aos gerentes e líderes a tomar decisões para estimativa de prazos e custos, e controle de pessoas na equipe.

Apesar do foco em gerência e liderança, as entrevistas também foram realizadas com desenvolvedores, e suas respostas ajudaram a perceber necessidades deste perfil, que levava a problemas de gestão, e não seriam notados apenas com as falas dos gerentes e líderes. Pode-se exemplificar com a necessidade dos desenvolvedores de enxergar os impactos de uma alteração no código fonte. Apesar dessa não ser vista como uma necessidade direta dos perfis alvos, os desenvolvedores fizeram perceber que essa análise ajudaria não somente nas decisões de implementação, como também na definição dos prazos relacionados ao tempo de resolução de uma tarefa, por exemplo. Com essa informação, a estimativa de prazos e custos pode ser melhor definida, e ajudar ao próprio gerente ou líder entrar em acordo com a equipe definindo um prazo de entrega mais adequado.

4.1.0.2 Propostas de Visualização

Para verificar as propostas de visualização, uma pesquisa na literatura foi realizada sobre visualização da informação, seus tipos e sugestões diante das necessidades levanta-

das. Apesar de, na Fundamentação Teórica, alguns casos já terem sido abordados, estes resumem-se a casos tradicionais e específicos utilizados por uma parcela de trabalhos, que envolvem a rastreabilidade. Não há, contudo, uma justificativa da escolha do tipo de visualização e os possíveis dados, que podem ser representados. Estas técnicas encontradas, e já citadas no Capítulo 2, foram discutidas a seguir, agora não trazendo a sua definição e sim apontando as vantagens e desvantagens perante as condições propostas na Figura 7, que foram julgadas como importantes na definição de uma visualização.

A **Matriz de Rastreabilidade**, apesar de ser uma técnica bem consolidada na literatura para a visualização de links de rastreabilidade, apresenta vantagens e desvantagens diante das necessidades trazidas das entrevistas. Com sua facilidade no entendimento, mesmo por usuários não técnicos (MERTEN; JÜPPNER; DELATER, 2011; WINKLER, 2008), esta técnica traz relacionamentos entre os artefatos de forma clara e simples, fornecendo uma rápida visão geral das relações entre dois artefatos - se o conjunto de artefatos for pequeno (CHEN; HOSKING; GRUNDY, 2012). É possível verificar um exemplo de Matriz de Rastreabilidade na Figura 8.

Em um cenário real, matrizes de rastreabilidade podem se tornar grandes e ilegíveis (WINKLER, 2008) - imagine que para representar o relacionamento entre N artefatos, são necessárias NxN células. Além desta desvantagem, que pode dificultar sua leitura, as matrizes também podem necessitar do uso de várias camadas de rastreabilidade, pois, muitas vezes, mais de uma matriz pode ser necessária para representar diferentes tipos de conexões entre os artefatos (MERTEN; JÜPPNER; DELATER, 2011), como por exemplo, ora se tem a matriz de dependências, ora se tem a matriz de pré-rastreabilidade.

Outro ponto negativo é que só é possível verificar os relacionamentos, sem a possibilidade de ter algumas informações como atributos, que acrescentam conhecimento, como a data de criação, o responsável ou a versão do artefato. Como dito por Winkler (2008), a separação de conteúdos e links é um obstáculo para qualquer tarefa avançada, que precisa ser realizada com o uso da matriz. Esta é uma grande desvantagem, visto que algumas informações foram traçadas como importantes para os gerentes ou líderes entrevistados.

Apesar de Franceschi e Duarte (2011) terem criado uma **Matriz de Rastreabilidade Alternativa (1)** que acrescenta dois atributos (ver Figura 9): sentido e força do relacionamento - através de seta e cores, respectivamente; estas informações não são suficientes diante dos dados trazidos das entrevistas. Esta matriz permanece com o problema de poder apresentar grandes proporções, falta de informações e necessidade de, para cada tipo de relacionamento, gerar uma nova matriz.

		Artefatos Alvo				
		Req-1	Req-2	Req-3	Req-4	Req-5
Artefatos Iniciais	Req-1		X			
	Req-2			X		
	Req-3					X
	Req-4			X		
	Req-5					

Figura 8: Exemplo da técnica de visualização Matriz de Rastreabilidade. Fonte: Pohl e Rupp (2012).

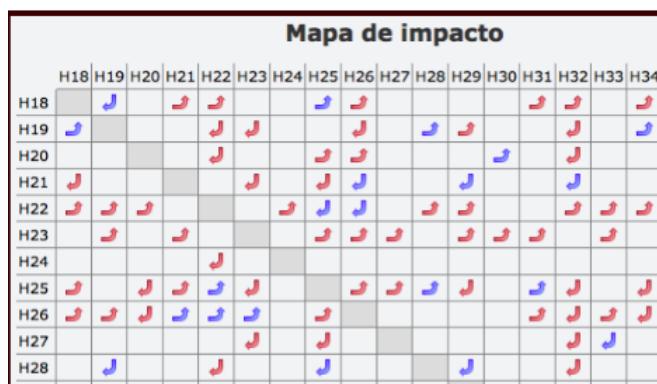


Figura 9: Exemplo da Técnica de Visualização Matriz de Rastreabilidade sugerida por Franceschi e Duarte (2011). Fonte: Franceschi e Duarte (2011).

Árvores, assim como as matrizes, são consideradas intuitivas (MERTEN; JÜPPNER; DELATER, 2011). Para Chen, Hosking e Grundy (2012), sua representação é de fácil entendimento até para leigos e comunica de forma clara a estrutura hierárquica (HOLTEN, 2006). Uma visualização com o uso de Árvore pode ser verificada em Figura 10.

Mesmo com suas vantagens, estas visualizações não possuem a capacidade de apresentar *links* adicionais de rastreabilidade entre requisitos, além de pais e filhos, como por exemplo *links* de decisões de lógica, conflitos, decomposições para refinamento e implementações (MERTEN; JÜPPNER; DELATER, 2011). Qualquer relacionamento ao lado da relação pai e filho mudaria a estrutura de árvore para um grafo.

Ainda que já existam, na literatura, formas sofisticadas para a visualização de árvores, como citadas em Wijk e Wetering (1999) e Herman, Melançon e Marshall (2000), tais layouts não resolvem o problema de ter relacionamento apenas com um pai, pois esta característica é fruto da definição de uma árvore. O problema assim continua sendo uma grande desvantagem provinda das árvores para o cenário descrito, pois nem sempre há

um único pai na representação de casos de *links* de rastreabilidade de *software*.

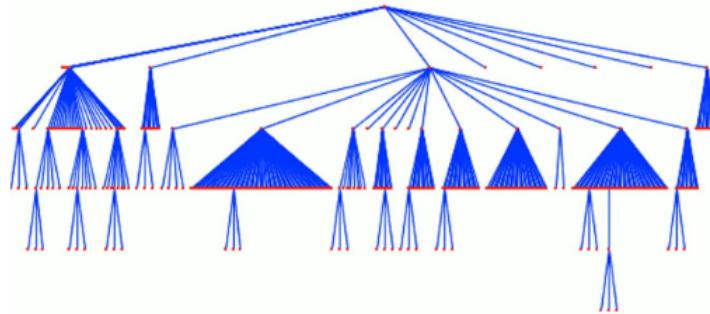


Figura 10: Exemplo da técnica de visualização Árvore. Fonte: Herman, Melançon e Marshall (2000).

Holten (2006) ainda apresenta uma maneira de visualizar uma árvore (**Árvore Alternativa (1)**) mostrando relacionamentos entre os nós folhas, indo além do tradicional relacionamento folha e pai. Apesar de resolver parte da desvantagem de uma árvore, esta solução traz densidade e sobreposição entre as arestas, tornando uma difícil leitura da origem e destino do relacionamento (CHEN; HOSKING; GRUNDY, 2012). Além disso, tais ligações estão presentes apenas nos nós folhas da árvores, quando na verdade, em um cenário real definido pelo contexto do presente trabalho, poderiam existir relacionamentos além desses traçados por Holten (2006). Na Figura 11, é possível verificar uma representação dada por Holten (2006).

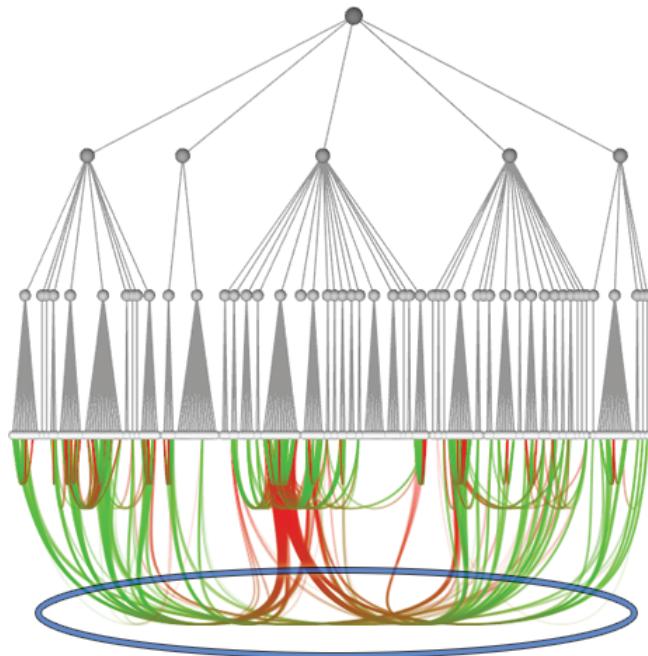


Figura 11: Exemplo da técnica Alternativa (1) de visualização de Árvore. Fonte: Holten (2006).

Chen, Hosking e Grundy (2012) cita outra técnica que representa uma árvore (**Árvore Alternativa (2)**). A diferença está na estrutura montada dos nós folhas. Nela, são diretamente adicionados os links de rastreabilidade, como é possível verificar nos quadrados em cinza da Figura 12. Na sua estrutura, se um nó filho estiver relacionado a outros três nós, por exemplo, adiciona-se três nós sobre o nó filho (CHEN; HOSKING; GRUNDY, 2012).

Com a mesma estrutura da árvore é possível ter as mesmas vantagens, acrescentando a vantagem de saber quantos relacionamentos o nó filho possui. Entretanto, não são disponibilizados mais detalhes sobre os nós pais e seus relacionamentos, dificultando assim a criação de links de rastreabilidade em nós que sejam pais. Isto pode ser uma limitação, quando se pensa em relacionamentos de rastreabilidade.

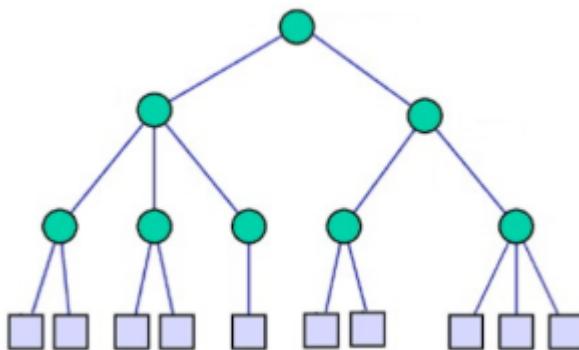


Figura 12: Exemplo da técnica Alternativa (2) de visualização de Árvore. Fonte: Chen, Hosking e Grundy (2012).

O **Treemap**, assim como as árvores, expressa-se de forma hierárquica. Seu uso é bastante eficaz em casos que o tamanho do recurso é o mais importante a ser exibido (WIJK; WETERING, 1999). Como exemplo dado por Wijk e Wetering (1999), é possível perceber sem esforço quais recursos são maiores, quando representados por esta técnica (ver Figura 13). Para Chen, Hosking e Grundy (2012), o *layout* do Treemap faz com que seja uma técnica ideal para exibir grandes árvores, usando efetivamente o espaço de exibição. Para estes autores, o Treemap pode transmitir uma estrutura global de alto nível de um sistema sob rastreamento, respondendo a perguntas como quantos itens um artefato possui, qual possui um maior número de itens, e como eles são organizados.

Algumas limitações são encontradas no Treemap, uma delas está na dificuldade de visualizar a estrutura da árvore que representa (WIJK; WETERING, 1999; CHEN; HOSKING; GRUNDY, 2012). A partir dela, é mais fácil enxergar aqueles elementos que não se relacionam, do que as relações que existem; e como o relacionamento é algo importante para

se verificar dado o contexto deste trabalho, tal técnica acaba tendo grande desvantagem diante outras (ver Figura 14, onde há uma representação da árvore em Treemap). Além disso, por representar uma árvore, os problemas apresentados na árvore para representação de *links* de rastreabilidade, refletem-se nesta mesma técnica.

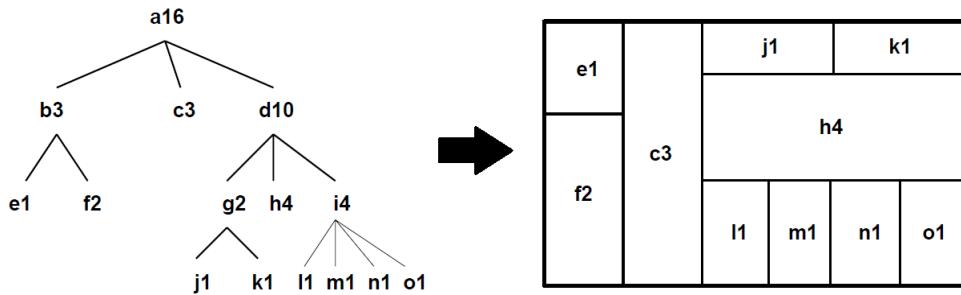


Figura 13: Exemplo da transformação de Árvore em Treemap. Fonte: Wijk e Wetering (1999).

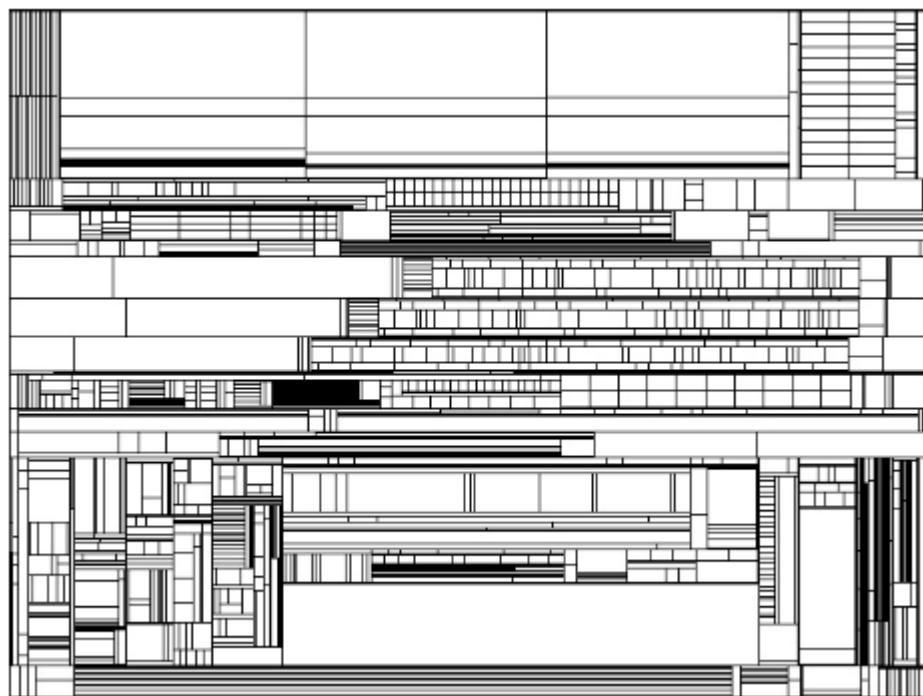


Figura 14: Exemplo da técnica de Visualização Treemap. Fonte: Wijk e Wetering (1999).

Apesar do Treemap não mostrar os relacionamentos, Holten (2006) sugere uma visualização baseada em Treemap com o acréscimo de tais *links* (arestas de adjacência) (**Treemap Alternativa (1)**). Tal efeito pode ser verificado, na Figura 15. As vantagens do Treemap refletem-se na técnica proposta por Holten (2006). Além das descritas no Treemap, Holten (2006) acrescenta a visualização de *links* de rastreabilidade. Como os *links* estão abertos para serem vinculados a qualquer retângulo que representa o nó, está visualização pode ser uma representação de um grafo.

Embora Holten (2006) resolva o problema de percepção dos relacionamentos causados no Treemap, segundo Chen, Hosking e Grundy (2012) tais relacionamentos são fáceis de se tornarem superlotados ao considerar um grande número de links, causando assim uma desvantagem relacionada à desordem visual com o grande número de arestas exibido. Os elementos mais próximos à borda têm pouco espaço para gerar a aresta, sendo limitado esse tamanho; enquanto que os elementos mais centrais podem possuir um grande volume de arestas passando acima, dificultando sua visualização (ver Figura 15.a).

Ainda sim, há o empacotamento de arestas que alivia a visualização para o problema de desordem visual se um grande número de arestas é mostrado (CHEN; HOSKING; GRUNDY, 2012) (ver Figura 15.b). Entretanto, o empacotamento de arestas deixa a região em que as relações cruzam densa, impedindo uma melhor visualização dos elementos, que estão abaixo dela por conta da sobreposição da região abaixo. Tais abordagens têm a dificuldade em identificar os itens de origem e destino de uma ligação, se não utilizarem uma técnica de ?foco e contexto? (CHEN; HOSKING; GRUNDY, 2012).

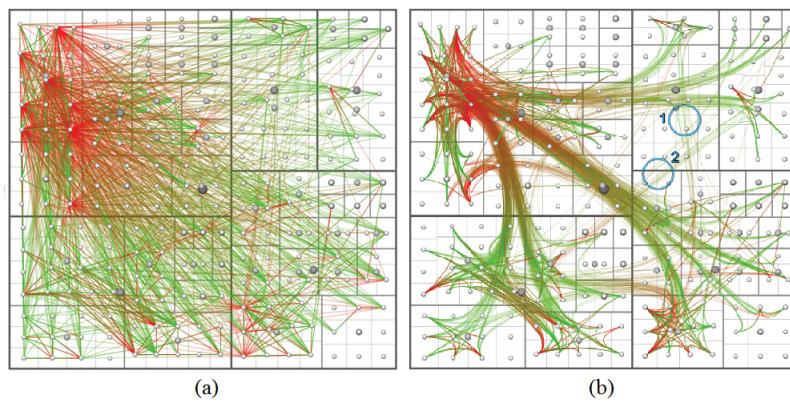


Figura 15: Exemplo de técnica Alternativa (1) de visualização do Treemap. Fonte: Holten (2006).

Pensando em amenizar o problema de densidade de relacionamentos, que na sugestão do Treemap com *links* dada por Holten (2006), Chen, Hosking e Grundy (2012) sugere que ao invés de desenhar arestas sobre o Treemap (**Treemap Alternativa (2)**), sejam introduzidas cores nas situações do relacionamento de cada nó, descrevendo se o nó possui *links* e quantos são. Para isso, Chen, Hosking e Grundy (2012) criam intervalo de cores, que representam as quantidades de relacionamentos.

Ao clicar no que representa o nó no Treemap, esses autores sugerem que seja mostrada uma árvore na vertical, onde tal árvore ilustra dois níveis de informação de dependência (CHEN; HOSKING; GRUNDY, 2012). O primeiro nível são os artefatos relacionados ao item selecionado, e o segundo nível são os artefatos, que dependem do artefato do primeiro nível.

Segundo Chen, Hosking e Grundy (2012), essa visualização mostra não somente artefatos relacionados ao item selecionado, como também informações de dependência entre esses artefatos. Cores também são usadas para mostrar os níveis de pontuação de similaridade dos links. Outra característica é poder comprimir ou expandir a árvore gerada para uma melhor visualização. Esta representação pode ser vista na Figura 16.

Apesar dessa ser uma solução interessante, não apresenta os atributos julgados importantes pelos gerentes e líderes para ajudar na tomada de decisão. Informações, como: responsáveis, versão e data de criação de um artefato, não são possíveis de serem identificadas. Percebeu que essa seria uma visualização mais apropriada quando se trata de código fonte, talvez sendo uma recomendação para os desenvolvedores.

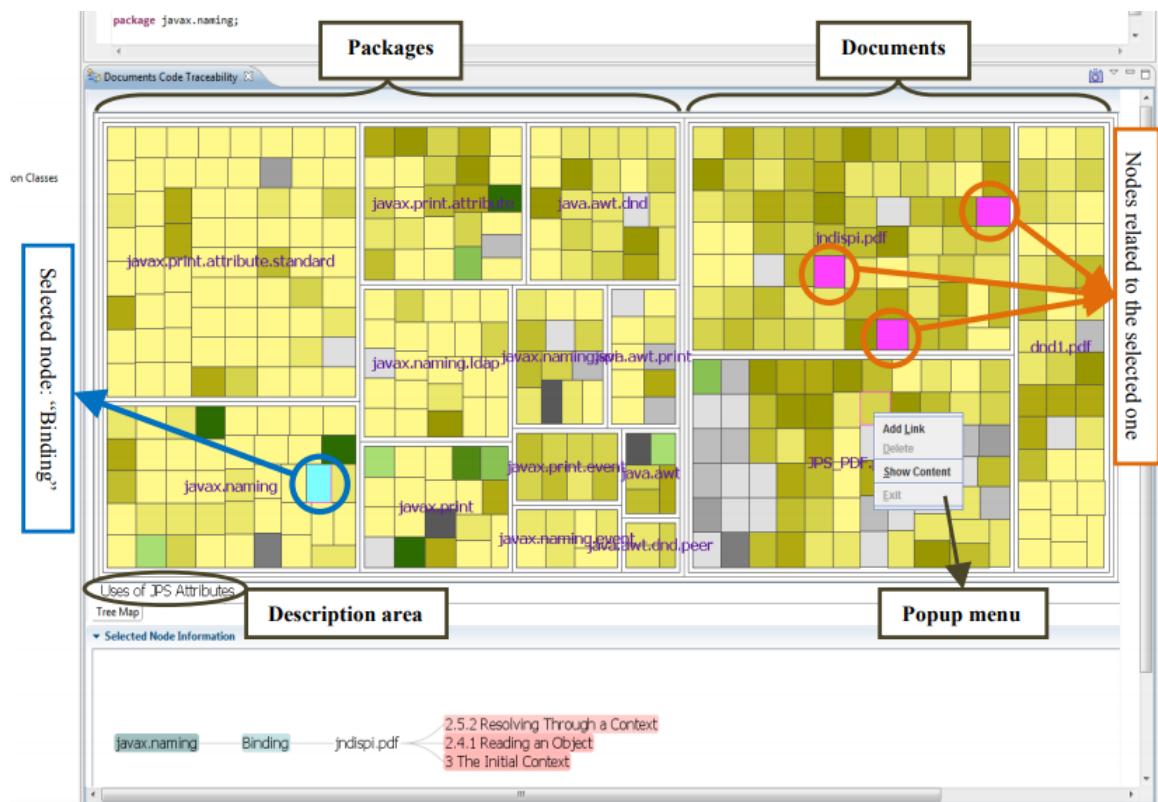


Figura 16: Exemplo de técnica Alternativa (2) de visualização do Treemap. Fonte: Chen, Hosking e Grundy (2012).

Holten (2006) indicou outras maneiras de representação de árvores com as arestas de adjacência acrescentadas, como as: **Radial Tree** e **Balloon Tree** (ver Figura 17.a 17.b respectivamente). Tais representações dos relacionamentos são difíceis de serem lidas pelas mesmas razões apresentadas para sua solução dada ao Treemap - relacionamentos se tornam densos e dificultam a leitura. Adicionalmente, estas técnicas também podem ser vistas em Herman, Melançon e Marshall (2000). Há apenas uma ressalva a ser feita: estes autores preocuparam-se com a sobreposição de arestas, mas seus exemplos resumem-se

em amostras pequenas (ver Figura 18).

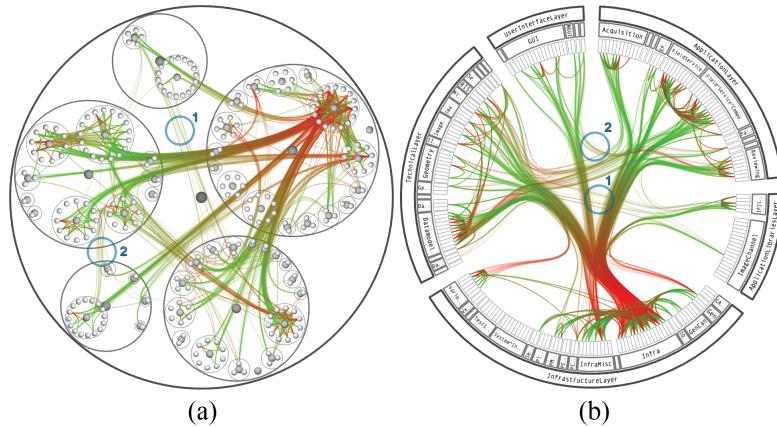


Figura 17: Exemplo da técnica de Visualização Radial Tree (a) e Balloon Tree (b). Fonte: Holten (2006).

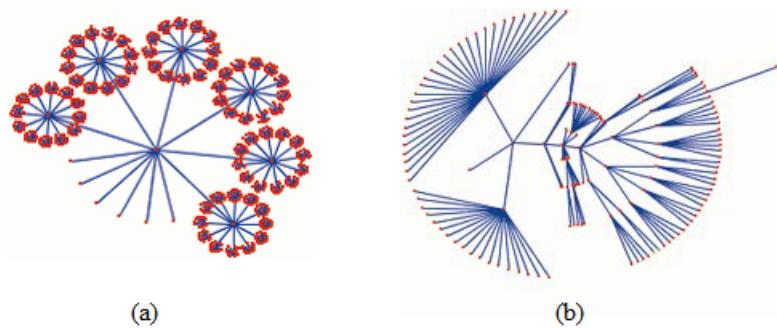


Figura 18: Exemplo da técnica de Visualização Radial Tree (a) e Balloon Tree (b). Fonte: Herman, Melançon e Marshall (2000).

O **Sunburst** é uma técnica apontada por Merten, Jüppner e Delater (2011), onde é feita uma comparação com árvores. Para estes autores, as árvores crescem rapidamente na direção vertical se muitos ramos forem expandidos, cabendo apenas uma parte da árvore, na tela do computador. Merten, Jüppner e Delater (2011) defenderam que o Sunburst é uma árvore em *layout* radial, o que pode criar dificuldades no crescimento de uma árvore, pois na técnica Sunburst a árvore cresce uniformemente em todas as direções, e não somente na vertical. A Figura 19 mostra um Sunburst.

Apesar de Stasko et al. (2000) comprovarem sua utilidade e comprehensibilidade, este foi um caso para a navegação em estrutura de sistemas de arquivos. No contexto de rastreabilidade de diferentes artefatos, a estrutura torna-se diferente, pois um link pode estar vinculado a mais de um artefato (pai). Sendo assim, as representações hierárquicas, como representações em árvores, não são adequadas por promoverem a duplicidade de elementos e não a visualização da multiplicidade de relacionamentos.

Além disso, Stasko et al. (2000) indicaram que um dos problemas do Sunburst é a ilegibilidade devido a tendência da última camada (mais externa) ter divisões menores, que ocorre em situações para nós que tem muitos filhos. Para mitigar este problema, Merten, Jüppner e Delater (2011) sugeriram o kit de ferramentas, mostrado em Belmonte (2011). Este conjunto de ferramentas permite o dobramento e o desdobramento do nó, onde o usuário oculta as ramificações, que não são do seu interesse, e foca nas informações que está procurando. Estas ações liberam espaço para as outras informações, e tornando os nós maiores, o que permite melhor visualização. O Sunburst, contudo, continua sendo uma representação de uma estrutura hierárquica do projeto sob rastreamento (CHEN; HOSKING; GRUNDY, 2012), levando então as mesmas desvantagens de uma árvore no contexto citado.

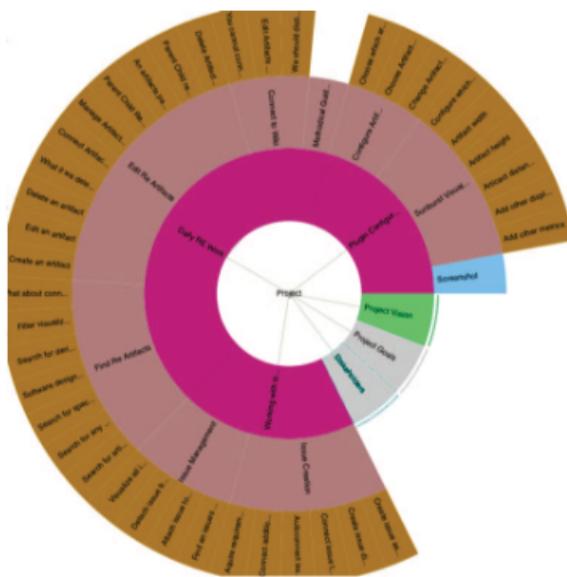


Figura 19: Exemplo da técnica de Visualização Sunburst. Fonte: Merten, Jüppner e Delater (2011).

Netmap é outra técnica mencionada por Merten, Jüppner e Delater (2011), como tratado na fundamentação teórica, que se aproxima do Sunburst. Assim como o Sunburst, o Netmap é representado por um *layout* radial, e sua diferença está na representação ser feita com apenas um anel. Uma de suas vantagens está em também apresentar informações adicionais para a representação de cada nó, sendo este um fator importante, visto que algumas das necessidades estavam relacionadas com dados adicionais aos artefatos. A partir da Figura 20 é possível um verificar o exemplo de Netmap sem (a) e com dados adicionais (b).

As características do Netmap (de um *layout* radial e com apenas um anel) restringem o espaço radial para cada nó, que é representado (MERTEN; JÜPPNER; DELATER, 2011).

Essa restrição faz com que os artefatos e *links* de rastreabilidade - dispostos ao meio do círculo - precisem ser filtrados por um atributo para tornar a pesquisa dos links mais eficiente (MERTEN; JÜPPNER; DELATER, 2011). Isto pode ser definida com uma limitação idêntica a identificada na matriz, onde são necessárias várias representações com base no tipo de relacionamento ou artefato, que se deseja. Outra desvantagem do Netmap encontra-se na grande possibilidade de ter a maioria dos relacionamentos transpassando a outros relacionamentos, o que pode gerar confusão visual ao lidar com um grande número de links de rastreabilidade (CHEN; HOSKING; GRUNDY, 2012). Além disso, como já citado, a quantidade de nós pode limitar o tamanho do gráfico e de cada célula presente no anel, dificultando a leitura.

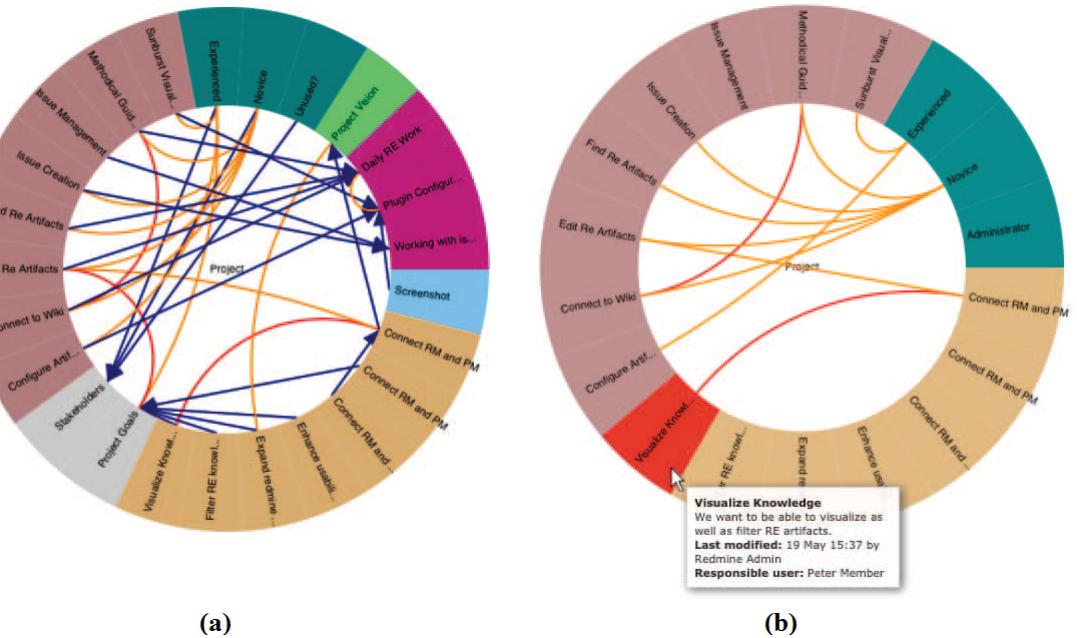


Figura 20: Exemplo da técnica de Visualização Netmap. Fonte: Merten, Jüppner e Delater (2011).

Para Heim et al. (2008) e Merten, Jüppner e Delater (2011), **Grafos de Rastreabilidade**, assim como as matrizes, podem fornecer uma representação intuitiva das relações entre artefatos. Merten, Jüppner e Delater (2011) diz que as pessoas tendem a interpretar mais os grafos. Além disso, diferentemente das matrizes, com grafos é possível visualizar todos os tipos de relacionamentos em uma única representação. Logo, esta técnica de visualização de rastreabilidade suporta uma visão geral das relações entre os artefatos, podendo ser usada para procurar links de rastreabilidade facilmente (CHEN; HOSKING; GRUNDY, 2012).

Apesar das suas vantagens, os grafos também apresentam a desvantagem ao se tornarem enormes, ocasionando uma difícil leitura, comprometendo a visibilidade ou usa-

bilidade (HERMAN; MELANÇON; MARSHALL, 2000). Em casos de elevada quantidade de elementos, os grafos podem ter seu desempenho comprometido ou até atingir limites na plataforma de visualização (HERMAN; MELANÇON; MARSHALL, 2000). Para casos de estruturas de grafos menores, é mais fácil a compreensão e análise detalhada (HERMAN; MELANÇON; MARSHALL, 2000).

Mesmo a academia apresentando outras representações de rastreabilidade, como citadas na Fundamentação Teórica (listas e referências textuais e *hiperlinks*), estas focam no uso da linguagem natural, caso este que não foi considerado neste trabalho como visualização. A representação considerada aqui, foca em elementos gráficos, sumarizando e relacionando as informações textuais. Portanto, na Fundamentação Teórica teve a preocupação de mostrar estas possibilidades; entretanto, visto a necessidade da indústria por elementos gráficos, não houve a intenção de aprofundar em tais técnicas.

4.2 Recomendação de Visualização

Diante de todo o cenário mostrado, com os pilares da informação discutidos e as propostas de visualizações da informação apresentadas, aqui é recomendada uma técnica de visualização para a compreensão dos dados que se deseja auxiliando o gerente e líder na tomada de decisão, seja ela para na gestão de tempo e de recursos - incluindo humanos, ou no planejamento e controle de artefatos. Mas para isto, como citado em UGGERMANN-KLEIN e Wood (1989), é essencial destacar o comum entendimento em design de que a estética e a legibilidade não coincidem necessariamente, sendo a legibilidade considerada mais importante que a estética, e assim tratada neste trabalho para a recomendação de visualização.

Como sugerido por Herman, Melançon e Marshall (2000), considerando o contexto, houve uma reflexão em cima da questão: "Existe uma relação inerente nos elementos de dados a serem visualizados?"; onde eles afirmam que caso haja a relação, os dados podem ser representados por nós e as ligações por *link*. Parece trivial tal consideração, mas ao mesmo tempo gera uma grande reflexão quando a literatura traz alternativas na visualização de links. E, foi percebido que estes links e nós podem ter suas representações com o uso de diversas técnicas, e uma mesma representação com vários *layouts*, trabalhados para facilitar o entendimento do visualizador ao ver a imagem.

Algumas dessas técnicas de visualização da informação de rastreabilidade propostas na literatura foram apresentadas na seção acima, onde cada uma delas apresentam van-



tagens e desvantagens diante do contexto montado com base nos pilares da informação levantados - ressaltando a importância de tal consideração para aqui sugerir uma visualização. Dentre elas, umas são mais tradicionais, enquanto outras mais inovadoras. E, ao se tratar de visualização de informação sobre rastreabilidade, os trabalhos mais recentes (CHEN; HOSKING; GRUNDY, 2012; KUGELE; ANTKOWIAK, 2016; RODRIGUES; LENCASTRE; FILHO, 2016; CYSNEIROS; ZISMAN, 2017) buscam as alternativas mais modernas.

Apesar de existir essas novas formas de visualizar na literatura, sendo elas utilizadas em trabalhos como (FRANCESCHI; DUARTE, 2011; HOLTEN, 2006; CHEN; HOSKING; GRUNDY, 2012; MERTEN; JÜPPNER; DELATER, 2011; STASKO et al., 2000; BELMONTE, 2011), para o cenário montado, com os pilares da informação levantados, percebeu que a utilização de uma forma tradicional com o grafo é mais eficiente, pois estas possibilitam uma representação do que foi levantado como necessário nas entrevistas de forma ágil e, principalmente, de fácil entendimento, sem a necessidade de um maior conhecimento sobre a técnica de visualização. Para Li e Maalej (2012), os grafos são intuitivos para representar e explorar relacionamentos de maneira informal, com compreensão de informações relativamente alta. E, com as respostas imediatas da maioria dos entrevistados, percebeu que a necessidade apresentada pelos gerentes e líderes era por grafos. Tal feito coincidiu com o resultado do estudo proposto neste trabalho.

Não diferente de pesquisas como Li e Maalej (2012), os grafos são vistos como boas representações para em cenários com tarefas de gerenciamento, onde há relações transitivas óbvias que ajudam a localizar artefatos influenciados. E, como dito por Li e Maalej (2012), grafos é uma visualização mais flexível devido a troca entre a visualização de informações locais e globais, ou seja, os artefatos podem ser representados como formas rotuladas, onde estes podem apresentar também informações que vão além da sua identificação. O que combina com as necessidades apresentadas pela indústria, ver além dos relacionamentos, como por exemplo, responsabilidades, datas e evolução.

A desvantagem do grafo na complexidade de mostrar informações para um elevado número de nós pode ser quebrada com os métodos ótimos para geração de *layouts* mais apropriados. Em Herman, Melançon e Marshall (2000) é possível verificar os possíveis *layouts* e algoritmos para um grafo e como ele pode facilitar a visualização, impedindo um maior sobreposição de arestas e nós. Onde, nós e arestas devem ser uniformemente distribuídos, sendo as arestas linha reta Herman, Melançon e Marshall (2000). Ou seja, apenas com a organização de nós e arestas é possível melhorar a compreensão, deixando a visualização mais limpa e consequentemente mais comprehensível e intuitiva.

Contudo, ainda sim pode haver uma grande massa de informações a ponto de dificultar a compreensão, tornando a aplicação de um bom *layout* ainda insuficiente para tratar a quantidade de dados. Como dito por Herman, Melançon e Marshall (2000), nenhum *layout* sozinho pode superar os problemas levantados por grandes tamanhos de grafos. Ainda na visão destes, as facilidades de navegação e interação são essenciais na visualização de informações. Para Shneiderman (1996) e Keim (2002), uma visualização inclui além dos tipos de dados, sendo técnicas de interação também consideradas. Com isso, visto que o fator humano é necessário para seleção da informação de rastreabilidade (MARCUS; XIE; POSHYVANYK, 2005), ainda há solução com a aplicação da mineração de dados, em casos de trabalhar com grande quantidade de dados.

Trabalhos como Keim (2002) e Shneiderman (1996) tratam da combinação de visualização de informação com mineração de dados. Além disso, segundo o Mantra de Busca de Informações Visuais (SHNEIDERMAN, 1996), os princípios básicos de diretrizes para design visual se resumem a: visão geral primeiro, zoom e filtro e, em seguida, detalhes sob demanda. O que leva a pensar não somente em soluções que mostre uma visão geral de forma eficaz, mas que também possibilite uma mineração dos dados necessários para o usuário naquele momento, ou seja, utilizando mecanismos de exploração para a navegação entre os dados.

Sendo conhecida como uma etapa no processo de descoberta de conhecimento, a mineração de dados revela padrões ocultos para uma avaliação (HAN; PEI; KAMBER, 2011). Entretanto, segundo Han, Pei e Kamber (2011), ainda percebe-se que na indústria e mídia o termo ?mineração de dados? é utilizado para se referir a todo o processo de descoberta e não somente a uma parte dele. Mas a mineração de dados também pode ser definida como o processo de descobrir não somente padrões, como também conhecimentos interessantes a partir de grande volume de dados (HAN; PEI; KAMBER, 2011). Tal uso, pode facilitar a seleção de informação que o gerente ou líder precisa naquele momento para uma melhor percepção de novos conhecimentos, antes não visto em uma visão geral dos dados.

Esses recurso especial de filtragem pode ajudar com problemas de escalabilidade (LI; MAALEJ, 2012; KEIM, 2002; HEIM et al., 2008). Para Li e Maalej (2012), os usuários devem ser capazes de selecionar e personalizar de forma interativa suas visualizações. Em um grafo de rastreabilidade, os usuários podem reorganizar os artefatos e seus *links* para satisfazer suas necessidades (LI; MAALEJ, 2012). Segundo Keim (2002), a visualização de informações e a mineração de dados visuais podem ajudar a lidar com o fluxo de informações, e a vantagem da exploração de dados visuais é o usuário estar diretamente

envolvido no processo de mineração. Ainda em sua visão, para uma mineração de dados eficaz, é importante incluir o humano no processo de exploração de dados, onde este aplica as suas habilidades perceptivas. Técnicas de mineração de dados visuais são de alto valor na análise exploratória de dados [Keim, 2002].

Sendo a mineração de dados uma maneira de solucionar a desvantagem dos grafos, focando em informações que se deseja, ela resulta em uma boa recomendação para gerentes e líderes buscarem informações de rastreabilidade que vão ajudar na tomada de decisão diante do constante cenário de mudanças percebido em ágeis. Além de todos esses pontos, as informações que vão além do relacionamento ainda podem ser vistas como atributos ou propriedades aos nós e links criados dentro do grafo. E, para Marcus, Xie e Poshyvanyk (2005) os elementos e propriedades dos links de rastreabilidade determinam muitas categorias, onde segundo eles, diferentes tarefas podem exigir acesso a um conjunto específico de links, com base em suas propriedades. Logo, através da mineração por conhecimentos específicos, as informações podem facilmente ser acessadas e verificadas, aumentando o conhecimento do gerente e líder com foco em determinado atributo desejado.

Além da interação, a customização também é necessária para ajudar procurar ou recuperar *links* (MARCUS; XIE; POSHYVANYK, 2005). E esses recursos podem ser facilmente atribuídos a estrutura de um grafo. Cores e destaque a diferentes tipos de *links* ou nós podem ajudar facilmente a evidenciar diferentes características entre os artefatos e seus relacionamentos antes mesmo de fazer uma filtragem, apenas através das evidências visuais. Para Voytek e Núñez (2011), a cor na visualização é uma maneira eficaz e atenta de chamar atenção para áreas de interesse. O tom é uma característica visual pré-atenta que é usada para identificar recursos importantes, permitindo que alguns aspectos de visualização se destaquem para o visualizador (VOYTEK; NÚÑEZ, 2011).

Além de cores, também é possível trabalhar com posicionamento e tamanho como forma de favorecer a compreensão humana na visualização e destacar informações, eliminando a complexidade que pode ser gerada em um grafo. Logo, além da filtragem ajudar a reduzir o esforço cognitivo do usuário, permitindo a concentração em um subconjunto de informações de *links* (LI; MAALEJ, 2012); o uso de características que dão destaque visual, trazendo informações antes não vistas, é também um ganho quando pensa em visualização.

Com isso, mesmo que a mineração possa ser aplicada ainda em outras técnicas aqui presentes, elas ainda apresentariam outras desvantagens que dificultariam nas necessidades traçadas pelos entrevistados. As técnicas como: árvores e suas extensões, treemap,

sunburst, dificultam um nó ter mais que um pai, sendo a solução de quebrar essa característica o uso de grafo ou a duplicidade de nó (informação) - para a cada pai que ele se relacionar, o nó está presente. Já as outras técnicas como: matrizes, técnicas estendidas com uso relacionamentos sugeridas por Holten (2006) (Radial Tree, Balloon Tree, Árvore, Treemap), netmap têm suas desvantagens quando apresentam a facilidade de sobreposição de arestas, mesmo para casos ainda menores, além de algumas não terem o tratamento ao acréscimo de informações em arestas e nós para enriquecer a visualização com dados antes não vistos e desejados pelos gerentes e líderes.

Para Herman, Melançon e Marshall (2000), qualquer sistema de visualização precisa fornecer uma interação quase que em tempo real para o usuário. Pensando nisso, para ilustrar a recomendação foi utilizado o banco de dados Neo4J¹, que permite que a interação do usuário através da mineração de dados, respondendo com um grafo. O Neo4j permite a utilização de cores, e por ser um banco de dados, é possível criar nós e arestas com propriedades definidas.

A Figura 21 apresenta um exemplo de grafo gerado no Neo4j a partir de um projeto aberto do governo, estando ele hospedado na plataforma Github. Algumas informações solicitadas pelos gerentes e líderes estão contidas neste grafo. Um exemplo de filtragem pode ser visto na 22, onde o filtro foi através do nome do usuário com suas issues (nós verdes) e *commits* (nós vermelhos) nos últimos 3 anos.

Outro exemplo se encontra na Figura 23 onde a busca foi realizada pela issue e seus relacionamentos. Nesta figura, o nó azul representa uma *label*; o verde, um *milestone*; o cinza, usuários e; o vermelho, *commit*.

Conclui-se que percebeu uma grande vantagem com grafos diante do cenário apresentado, com os pilares da informação considerados. Mas ainda sugere-se uma possível extensão para melhoramento do entendimento ou necessidades dos usuários, como por exemplo: incluir propriedades visíveis do nó sendo elas mostradas ao passar com o mouse em cima, incluir propriedades visíveis na aresta (representante do relacionamento), incluir cores nas arestas e tamanhos nos nós.

4.3 Considerações

Há uma grande importância em ir na indústria a partir das entrevistas verificar as necessidades por trás de quem usará a informação, pois analisar e entender por quem

¹<https://neo4j.com/>

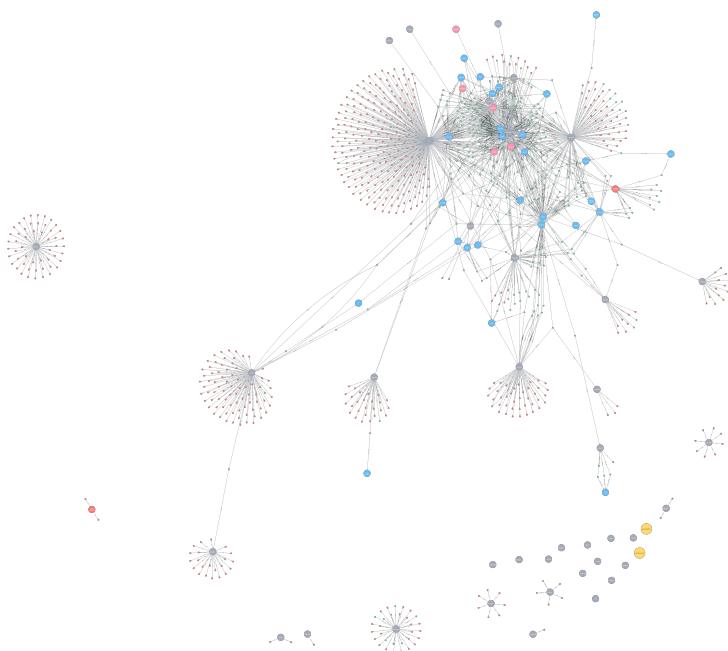


Figura 21: Exemplo de grafo gerado no Neo4j a partir de dados do Github.

uma visualização é usada e para quê ela é usada, antes mesmo da sua elaboração, não só resulta em uma execução mais eficiente, como também em uma aceitação mais ampla da visualização de rastreabilidade, como dito por Winkler (2008). Com isso, os pilares da informação para a visualização foram respondidos e analisados a ponto de refletir nas visualizações sugeridas pela literatura em cima do contexto montado.

Pode parecer trivial a tarefa de representar e navegar por links de rastreabilidade; entretanto, para Marcus, Xie e Poshyvanyk (2005) a visualização de links de rastreabilidade é uma tarefa importante e não trivial, sendo ela um suporte considerável e necessário para recuperar, navegar e manter esses links. Apesar disto, com os resultados das entrevistas percebeu que há uma falta de informações gráficas sobre dados de rastreabilidade, o que em parte acaba gerando sua má usabilidade, sendo esta uma razão para a pouca percepção dos benefícios que a rastreabilidade pode trazer (MARCUS; XIE; POSHYVANYK, 2005).

Uma visualização sofisticada não é necessária, segundo Marcus, Xie e Poshyvanyk (2005), pois dado um artefato de origem é fácil relacionar a todos os seus artefatos e propriedades. E para isto, é preciso diferenciar tipos de relacionamentos e artefatos, pois muitas vezes há uma necessidade de verificar ao mesmo tempo mais de um tipo de artefato e todos os links associados, logo, é desejável e importante fornecer diferenças visuais entre links e artefatos diferentes, bem como permitir aos usuários a mineração de dados visual, para que seja mostrado apenas informações de seu interesse (MARCUS; XIE; POSHYVANYK, 2005).



Figura 22: Exemplo de filtragem do por usuário no grafo gerado no Neo4j a partir de dados do Github.

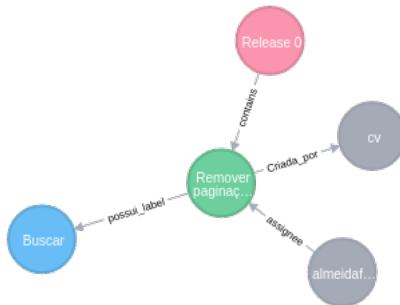


Figura 23: Exemplo de filtragem do por issue no grafo gerado no Neo4j a partir de dados do Github.

Muito se foi visto na literatura, desde novas a tradicionais tipos de visualização. A todo momento pensou nas necessidades trazidas nas entrevistas, traçando as vantagens e desvantagens de cada visualização com base nelas. Com todo esse estudo, percebeu que as visualizações de rastreabilidade se classificam em duas grandes classes; Árvores e Grafos. Suas representações que geram novas visualizações, mas com isso algumas delas trazem desvantagens no contexto deste trabalho, enquanto outras eram beneficiadas.

Grafos foi uma classe que ganhou destaque com sua possibilidade de criar vínculos sem regras relacionadas a quantidade - por exemplo, as árvores só permitem um pai ao nó filho. A regra geral da árvore dificulta a geração aberta de relacionamentos; logo, suas diferentes e inovadoras representações sofriam com a mesma desvantagem. Com isso, foram encontrados ainda casos de visualizações, como Treemap, que acrescentavam a

possibilidade de criar mais vínculos, não se restringindo a regra da árvore. Logo, não somente esse aspecto foi analisado, como também as outras necessidades citadas nas entrevistas.

Apesar do Grafo ter outras representações visuais, ainda percebeu que muitas destas sofriam com alguns aspectos, como: acréscimo de informações além dos relacionamentos e complexidade de leitura. A simplicidade na visualização é algo desejável para os entrevistados, o fato de não terem as informações de rastreabilidade de forma fácil e simples já é uma circunstância para o pouco uso da informação. Consequentemente, acrescentar visualizações complexas não irá favorecer o uso destes dados que ainda são tão pouco aproveitados devido a dificuldade de tratá-los.

Pensando na forma mais simplificada de representação de um grafo, percebeu apenas a desvantagem de difícil leitura em cenários de grande quantidade de dados.  , com a mineração de dados, seguindo os princípios básicos do mantra sugerido em Shneiderman (1996) (visão geral primeiro, zoom e filtro e, em seguida, detalhes sob demanda), a visualização em grafo ganhou mais destaque, pois sua desvantagem poderia ser resolvida com a visualização de sub-grafos resultantes. Ou seja, com a minimização dos dados, dando destaque apenas ao que interessa para o visualizador, as vantagens do grafo se destacam: na fácil leitura - até mesmo para leigos, simplicidade na representação de links e possibilidade do acréscimo de propriedades que representam as informações desejadas pelo perfil analisado.

Com toda essa análise, no próximo capítulo é estudada ferramentas de apoio a gerência de projetos que dão suporte às necessidades trazidas na entrevista, dando destaque ao o fornecimento de visualização de rastreabilidade com o uso do grafo ou não.

5 Análise de Ferramentas de Gerenciamento de Projetos

Ferramentas de gerenciamento de projetos são bastante utilizadas para o auxílio do planejamento e acompanhamento de um projeto, sendo a atividade de gerência de projetos um importante meio para atingir os objetivos traçados de maneira eficaz (FEITOSA, 2012). O gerenciamento de projetos é um campo de prática, construído socialmente que se desenvolve com uso destas ferramenta e técnicas desenvolvidas para dar suporte, além das interações entre os praticantes e pelos seus esforços deliberados (CRAWFORD, 2006). Portanto, sendo essa uma atividade paralela às demais na engenharia de *software*, ela ganha um grande destaque pela sua relevância e envolvimento com outras atividades.

É possível verificar que, independente de metodologia a ser aplicada no desenvolvimento de *software*, há ferramentas de gerenciamento de projetos que apoiam - desde as mais tradicionais, àquelas ágeis. E,  mesmo com o Manifesto Ágil (2001) tendo em um dos seus valores o destaque de indivíduos e interações mais que processos e ferramentas, o uso deste recurso acaba realçando junto às interações e indivíduos.

Ferramentas de gerenciamento de projetos se adaptaram facilmente aos princípios e valores ágeis. Processos como Scrum, quadros como Kanban, são naturalmente percebidos quando trata-se com foco nas metodologias ágeis. Buscando agilidade no processo, elas apoiam atividades do ágil de maneira simples e clara, se integrando a outras ferramentas que de alguma forma também dão suporte a tais metodologias, seja para interação entre equipes distribuídas, apoio ao versionamento e controle de código, notificação através sistemas de e-mail e até mesmo com outras ferramentas de gerenciamento de projetos para o carregamento de informações.

As interações entre essas ferramentas facilita a troca de dados  podem conter elementos referentes à rastreabilidade, e com isso um grande número  informações pode ser gerado e facilitado com o uso de uma visualização que ajude a manipulá-los. Uma grande fonte de informação pode ser obtida através da integração das ferramentas de gerencia-

mento de projetos com os próprios sistemas de controle de versionamento (SCV), visto que o uso de SCV nos tempos atuais torna-se uma característica indispensável a equipes cada vez mais distribuídas, além da segurança na hospedagem do material trabalhado.

5.1 Metodologia e critérios para a seleção e avaliação das ferramentas

Pensando no intenso uso de SCV e a grande variedade na indústria de ferramentas para gerenciamento de projetos, optou por buscar aquelas que se integram a um SCV, e como destacado no capítulo de fundamentação teórica, seção 2.4, foi utilizado como critério inicial para a seleção a integração com o Github, visto esta ser uma plataforma de destaque pela quantidade de usuários que apresenta (ver Capítulo 2 Seção 2.4). Além disso, também verificou as ferramentas de gerenciamento de projetos que apoiam as metodologias ágeis, em virtude deste trabalho está inserido no contexto ágil.

Considerando o cenário ágil, as plataformas de hospedagem de dados e versionamento, a rastreabilidade e visualização - trazidas dos debates nas entrevistas através dos pilares da informações, foram pesquisadas 19 ferramentas de gerenciamento de projetos, a fim de verificar seu suporte a rastreabilidade e visualização. A seleção destas ferramentas foi realizada com base também em sugestões de sites informativos e de opinião, como revistas especializadas para profissionais em desenvolvimento de *software* e *blogs*, e na experiência prévia de uso.

Outra consideração feita às ferramentas se deu com a disponibilidade de uma versão aberta, mesmo que com limitações. Logo, das 19 ferramentas listadas, 5 ferramentas (cujos ID são: 02, 06, 07, 11 e 18) foram desconsideradas por possuírem apenas opções demonstrativas - desde 7 dias até 30 dias; e 1 ferramenta (ID: 09) por ter a integração com o Github apenas na versão paga, restando assim, apenas 13 ferramentas (Tabela 8). Estas 13 foram analisadas de acordo com seus planos livre de pagamento, ou seja, elas ainda possuíam limitações, como por exemplo: livre para projetos abertos e *startups*; quantidade de usuários, projetos, integrações e tamanho do projeto limitados.

Tabela 8: Relação de categorias e heurísticas

ID	Ferramentas	Links	Licenciamento
01	Zube	https://zube.io/	Livre para 4 usuários
02	Codetree	https://codetree.com/	Demonstrativa por 14 dias

03	Zenhub	https://www.zenhub.com/	Livre para projetos open source
04	Issue.sh	https://issue.sh/	Livre para projetos open source
05	Waffle	https://waffle.io/	Livre para projetos open source
06	Jira	https://br.atlassian.com/software/jira	Demonstrativa por 7 dias
07	Redmine Easy	https://www.easyredmine.com/	Demonstrativa por 30 dias
08	Pivotal Tracker	https://www.pivotaltracker.com/	Livre para 3 usuários
09	Trello	https://trello.com/	Livre, mas integração ao Github paga
10	Pipefy	https://www.pipefy.com/	Livre para 5 usuários
11	Azendoo	https://www.azendoo.com/	Demonstrativa por 14 dias
12	Taiga	https://taiga.io/	Livre
13	Mingle	https://www.thoughtworks.com/mingle/	Livre para 5 usuários
14	Meister task	https://www.meistertask.com/pt	Livre com limitações
15	Wrike	https://www.wrike.com/pt-br/	Livre para 5 usuários
16	You track	https://www.jetbrains.com/youtrack/	Livre para 10 usuários, projetos open source, 5gb
17	Visual Studio Team Services	https://www.visualstudio.com/team-services/	Livre para 5 usuários
18	Basecamp	https://basecamp.com/	Demonstrativa por 30 dias
19	Acunote	http://www.acunote.com/	Livre para 5 usuários

A avaliação das ferramentas foi realizada de acordo com a criação de heurísticas com base nas categorias: Rastreabilidade e Visualização da Rastreabilidade, onde emergiram das entrevistas tratadas no Capítulo 3. Cada categoria, sendo ela refletida sobre os dados das entrevistas, inspiraram criação destas heurísticas que foram a base para questões a serem avaliadas de acordo com a ferramenta.

5.2 Aplicação das Heurísticas

Como tratado anteriormente, a partir das entrevistas surgiram categorias que levaram as heurísticas, sendo aqui tratadas somente àquelas relacionadas à rastreabilidade e

visualização. Ainda ressaltando, para a verificação da heurística proposta foram criadas questões de pesquisa e definidos critérios para a análise das ferramentas selecionadas. A Tabela 9 e 10 apresenta as categorias Rastreabilidade e Visualização da Rastreabilidade, respectivamente, e com as heurísticas, seguidas das questões de pesquisa. No Apêndice Y é possível verificar uma tabela geral (Tabela X) com a relação de categorias, heurísticas, questões e critérios.

Tabela 10: Heurísticas e questões de pesquisa relacionadas à categoria de Visualização da Rastreabilidade

Adaptabilidade aos pilares definidos da informação (problemas, o quê?, quando?, quem?)	Quais são os tipos de visualizações (representações) dos relacionamentos entre os artefatos presentes na ferramenta?
	Há presença de visualizações com informações das propriedades de um artefato?
	Há presença de visualizações com informações das propriedades de um relacionamento?
	É possível combinar mais de um tipo de visualização?
	É possível escolher a forma de visualização?
Assistência à seleção de artefatos e à identificação de repercussão das ações	É possível a identificação de artefatos afetados em uma manutenção ou evolução específica?
	As dependências entre artefatos produzidos podem ser analisadas?
	É possível identificar todos os stakeholders envolvidos no artefato?
	O stakeholder responsável por uma mudança no artefato pode ser verificado?
Apoio à tomada de decisão para planejamento e controle dos artefatos	O histórico de alterações feitas em um artefato, com datas, motivos e versão do artefato pode ser observado?
	As situações dos artefatos são representadas de forma gráfica?
	É possível avaliar a cobertura de testes dos requisitos no código fonte?

	<p>É possível verificar quem priorizou a tarefa?</p> <p>Possibilita a visualização de artefatos relacionados a uma participação?</p> <p>É possível identificar o tipo de classificação da demanda que ocorre?</p> <p>É possível a identificação de artefatos que não estão aloados ou não possuem relacionamento dentro do projeto?</p>
Apoio à tomada de decisão para gestão de recursos, incluindo humanos e tempo	<p>A ferramenta possibilita o acompanhamento de realização das atividades ou do desempenho por indivíduo?</p> <p>A ferramenta possibilita o acompanhamento de realização das atividades ou do desempenho por equipe?</p> <p>Existem visualizações que identifiquem com precisão e rapidez os atrasos e as inconformidades no projeto?</p> <p>É viável o monitoramento do tempo gasto nas tarefas do projeto?</p> <p>Atribuições de um membro são identificáveis e rastreáveis nos artefatos gerados?</p> <p>É possível fazer a análise dos stakeholders que precisam ser informados de uma alteração de artefato solicitada ou realizada?</p> <p>Permite a verificação do que foi produzido em um dado período de tempo, com seus relacionamentos e responsáveis?</p>
Flexibilidade nos pontos de vista do projeto	<p>É possível visualizar a dependência de interprojetos?</p> <p>Há quadros de tarefas com sumarização das informações personalizáveis?</p> <p>É possível a verificação de milestones e, consequentemente, vínculos destes com os artefatos?</p>
	<p>Há uma visualização do histórico de um projeto?</p>

Auxílio à manipulação e à naveabilidade nas informações de rastreabilidade	Há mecanismos que permitem a manipulação direta das informações de um artefato? Existem mecanismos que possibilitem a navegação entre as informações? É possível analisar artefatos relacionados apenas a pré-rastreabilidade de um artefato? É possível analisar artefatos relacionados apenas a pós-rastreabilidade de um artefato?
Redução na carga cognitiva (por busca de informações)	Há mecanismos de consulta e de filtragem das informações apresentadas na visualização? É permitido cadastramento de consultas recorrentes? Ao selecionar um artefato, é possível identificar suas relações e refinamentos? Existem possibilidades de configuração de cores, texturas, tamanhos ou posição para diferenciar tipos dos artefatos? Existem possibilidades de configuração de cores, texturas, tamanhos ou posição para diferenciar situações dos artefatos? Existem possibilidades de configuração de cores, texturas, tamanhos ou posição para criar agrupamentos de artefatos? Existem possibilidades de configuração de cores, texturas ou espessuras para os diferentes tipos de relacionamentos entre os artefatos? Existem possibilidades de configuração de cores, texturas ou espessuras para os diferentes graus de relacionamentos entre os artefatos? Há representações gráficas das informações (relacionados a rastreabilidade) ou diagramas construídos automaticamente?

As ferramentas foram avaliadas seguindo passos de acordo com as necessidades impostas para a análise. Ao início da exploração de cada ferramenta foram cadastradas

Tabela 9: Heurísticas e questões de pesquisa relacionadas à categoria de Rastreabilidade

Apoio aos tipos de rastreabilidade (pré, inter e pós - rastreabilidade)	<p>Possibilita a criação de rastros com artefatos criados antes do artefato instanciado?</p> <p>A criação de rastros com artefatos criados depois do artefato instanciado é possível?</p> <p>A evolução dos artefatos (versionamento) é viável?</p>
Auxílio à rastreabilidade de propriedades de um artefato	<p>É possível criar tipos diferentes para a classificação de demandas? (por exemplo: bug, correção, nova, teste, etc.)</p> <p>O armazenamento de quem priorizou é feito?</p> <p>É possível cadastrar dados referentes ao tempo considerando a evolução do artefato? (data de criação, data de sugestão de modificação, data de modificação)</p> <p>O cadastramento do tempo gasto nas tarefas é viável?</p> <p>É possível cadastrar os envolvidos no artefato?</p> <p>Possibilita o armazenamento da discussão ocorrida para a manutenção ou evolução do artefato?</p>
Assistência a rastreabilidade do projeto	<p>São mantidos os históricos dos projetos entregues?</p> <p>É possível a criação de milestones e, consequentemente, seu vínculo com artefatos?</p> <p>É possível configurar a dependência de interprojetos?</p> <p>Possibilita a criação de artefatos relacionados a uma participação?</p> <p>Qual é o tipo de integração feita com o Github?</p>

contas, onde após realizar o login, um projeto era criado - visto que para o próximo passo, algumas vezes era necessário um projeto. Com isso, a integração com o Github era efetuada, onde em alguns casos precisou que fosse configurada de forma mais manual, com o uso da própria plataforma do Github; e em outros, bastava um clique de permissão de acesso à plataforma de hospedagem de dados e versionamento a partir da ferramenta de gerenciamento de projetos.

Após a integração com o Github, já foi possível realizar ações para verificar o tipo de integração, sendo ele unilateral ou bilateral. Para isto, issues eram criadas na ferramenta e verificadas se estas se apresentavam no Github; da mesma forma o inverso foi testado, criando issues no Github e observando se eram trazidas na ferramenta de gerenciamento. Além disso, também testou se a atribuição a propriedades da issue eram mantidas em ambos os casos, tais propriedades foram desde comentários até atribuição de responsáveis. Esta ação respondeu a questão "Qual é o tipo de integração feita com o Github?", encontrada na heurística "**Assistência a rastreabilidade do projeto**", da categoria Rastreabilidade.

A integração fez parte dos passos iniciais devido a facilidade que ela trazia nos casos em que fosse bilateral, visto que com o carregamento de um projeto teste, *issues* já eram inseridas na ferramenta, diminuindo o esforço de criá-las. Após isto, seguiu um formulário com ações que precisaram ser tomadas para responder cada uma das questões, logo, nesse momento a ordem apresentada das questões na Tabela 9 e 10 deram sequência. Portanto, apenas a questão sobre a integração ao Github foi exceção ao momento de verificação.

Como apoio ao registro de dados, uma planilha foi criada, onde permitiu lançar as respostas correspondentes a cada questão de acordo com ferramenta analisada. Isto ajudou na criação de gráficos que pudessem apoiar às análises feitas, mostrando dados interessantes, sendo eles discutidos a seguir.

5.3 Análise dos Resultados da Aplicação das Heurísticas

Com as 13 ferramentas analisadas, alguns pontos foram tratados e analisados nesta seção. A análise foi feita por heurística, sendo refletidos as respostas às questões da heurística levantada que ganharam um maior destaque por alguma diferenciação das demais. Vale ressaltar que a análise foi feita na versão livre destas, então limitações são dadas, desde quantidades de usuário até a liberação de certas funcionalidades. Logo, o que está

sendo discutido é com base no plano livre de pagamento de taxas para uso.

Nas 13 ferramentas, que dão suporte a adoção de metodologias ou processos ágeis, uma similaridade foi percebida em artefatos gerados por cada. Quadros para o gerenciamento e planejamento do projeto; criação de *sprints*; *backlog*; gráfico *burndown*; histórias; tarefas; releases foram contatados em sua grande maioria.

Além dos artefatos gerados, funcionalidades em comum também foram percebidas - o que já era esperado visto a intenção e contexto em comum das ferramentas; entretanto algumas apresentam nomenclaturas diferentes, quando a intenção dada à funcionalidade é a mesma. Um exemplo disto foi a criação de cartão e criação de tarefa visto em ferramentas diferentes, onde a criação da tarefa era um cartão e vice-versa. Esta funcionalidade foi encontrada na sua grande maioria através dos quadros de tarefas, onde apenas uma ferramenta não apresentou possibilidade de quadro (ID: 19), onde deu somente a opção de verificar as tarefas através de uma lista.

Tais quadros tarefas para gerenciamento e planejamento de tarefas foram um destaque de semelhança. Tarefas - como será tratado, histórias de usuário, issues, cartões são termos utilizados com a mesma intenção dentro das ferramentas, sendo elas interpretadas como um importante artefato, pois todos os outros artefatos gerados dependiam de sua criação. Como exemplo tem código, documentos, testes e discussões, que podiam ser associados para o acompanhamento dentro das tarefas, seja através de um arquivo anexado para documentos, vínculo com o Github para código e testes ou fóruns para discussões.

Apesar das similaridades de funcionalidade, ainda foi possível perceber características específicas de cada ferramentas. Algumas delas não apresentaram funcionalidades básicas para o apoio à rastreabilidade, como por exemplo, o cadastro de relacionamentos (ver Figura 24); enquanto outras mostraram, além do cadastro manual, o cadastramento automático provindo do próprio Github. A Figura 24 representa a quantidade de ferramentas que permitem o vínculo, a depender do tipo de rastreabilidade, onde há um conjunto de ferramentas que permitiram o vínculo de forma manual e automática. Apenas uma (ID: 15) não apresentou a pré e pós-rastreabilidade de artefatos.

Em todas as ferramentas notou a possibilidade de fazer o acompanhamento da evolução dos artefatos através do uso das tarefas, a partir de comentários feitos pelos usuários sobre o versionamento do artefato na própria tarefa. Já para o versionamento da tarefa - onde também é considerado um artefato; através do seu histórico era possível acompanhar sua evolução. Entretanto, somente uma ferramenta (ID: 13) gerou um número de versão para certas modificações na própria tarefa. Nas outras, isso só era possível através de

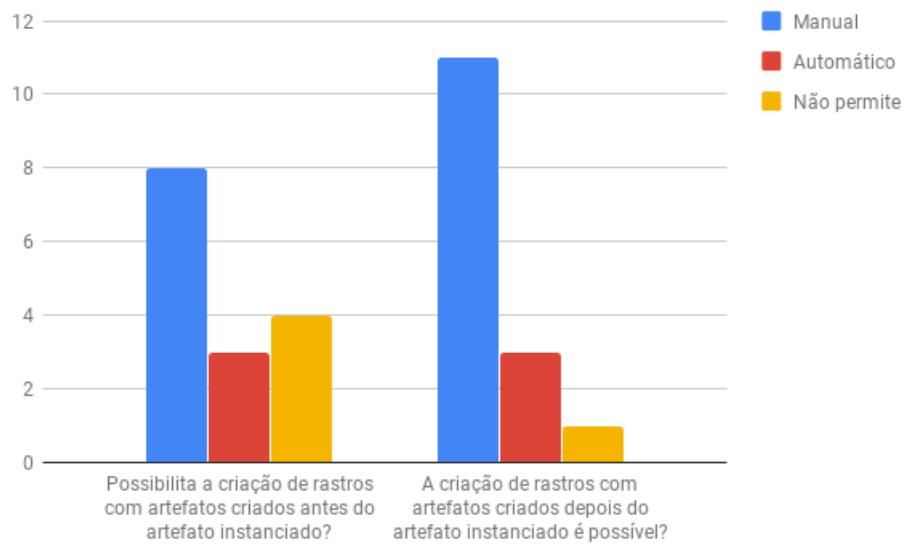


Figura 24: Suporte à criação de links de rastreabilidade nas ferramentas.

comentários dados pelos usuários. E, esta possibilidade de discussão (criação de comentários) a partir de uma tarefa criada foi dada em todas as ferramentas. Uma delas (ID: 13) apresentou uma interface semelhante a de chat dentro da tarefa, as outras tinham um aspecto mais aproximado a de um fórum.

A criação de vínculos de rastreabilidade foi julgada importante nas entrevistas para apoio a tomada de decisão, seja ela para gestão de recursos, incluindo humanos e tempo ou para planejamento e controle dos artefatos. É importante verificar na Figura 24 que em sua maioria, esta informação é de possível armazenamento. Entretanto, quando se tratou da evolução de um artefato com foco em versionamento, percebeu que a maioria das ferramentas possibilitava somente com uso de recursos gerais, como comentários. Portanto, observa-se que a heurística da categoria de rastreabilidade - "**Apoio aos tipos de rastreabilidade (pré, inter e pós - rastreabilidade)**", está presente na maioria das ferramentas quando tratado a criação de relacionamentos entre tarefas, mas ao pensar em rastrear o versionamento do artefato, foi preciso investigar outra funcionalidade que apoiava a atividade.

Ainda em se tratando dos dados de um artefato, viu a possibilidade de criação de tipos de demandas em 10 ferramentas, as outras 3 tinham apenas tipos pré-definidos. Para a informação do tempo ou esforço gasto em tarefas, ainda conseguiu verificar dentre as 13, 4 ferramentas que não possibilitaram o cadastro desta informação. Já para dados como: quem priorizou, datas referentes à evolução de um artefato e os envolvidos do artefato foram analisados a fim de verificar que a informação era cadastrada e, caso sim, se era

de forma manual. A Figura 25 apresenta o quantitativo de ferramentas que armazenam esses dados, seja de forma manual ou automática.

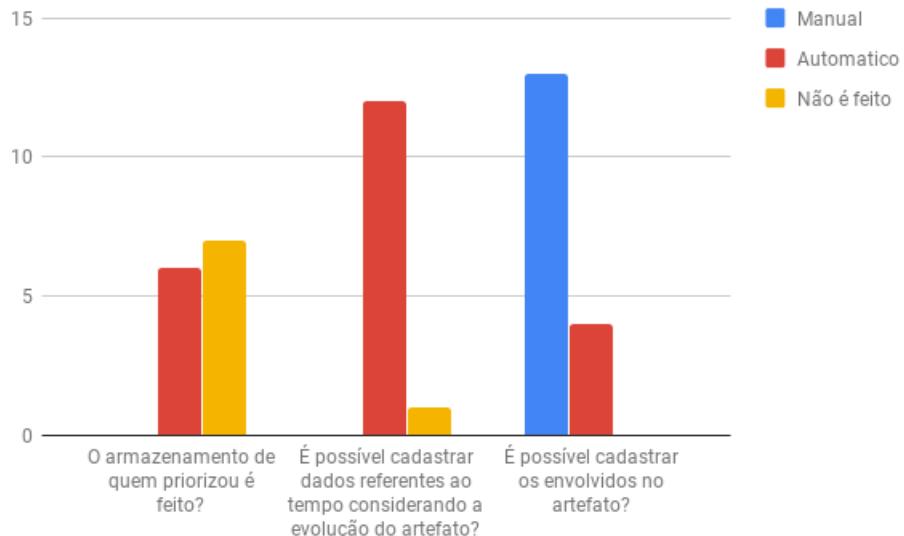


Figura 25: Dados de artefato e sua forma de armazenamento nas ferramentas.

Com isso, percebeu que a segunda heurística presente na categoria de Rastreabilidade, "**Auxílio à rastreabilidade de propriedades de um artefato**", as ferramentas em sua grande maioria a atendiam. A única exceção na divisão das ferramentas entre armazenar ou não quem priorizou o artefato, onde muitas vezes aconteciam apenas arrastando os mesmos, definindo uma ordem de priorização, e assim, o cadastro de quem fez essa alteração sendo perdido.

Agora, pensando na assistência a rastreabilidade do projeto, algumas questões foram levantadas. Entre estas, foi observado a integração da ferramenta de gerenciamento de projeto ao Github, onde apenas 4 delas ofereciam uma integração bilateral. Se tratando das unilaterais, algumas tinham serviços específicos de forma bilateral apenas para construção de uma *issue*, enquanto que na criação de dependências não havia essa passagem de informação. Logo, foram consideradas bilaterais aquelas que trocam informações em ambos os lados (da ferramenta para o Github e vice-versa), com as propriedades do artefato, ou seja, além de criação de *issues*, como dependências e responsáveis. Outras se integraram apenas através do *commit* no Github, permitindo que uma tarefa na ferramenta fosse fechada após o *commit* com uma mensagem da identificação da tarefa.

Também houve uma verificação das ferramentas no uso de *milestones*. Esta, sendo uma característica do Github, ainda não era contemplada em todas elas. Onde apenas duas possuíam o cadastramento deste artefato. Outras características pouco percebidas

nas ferramentas foram referentes a configuração de dependência de interprojetos e criação de artefatos relacionados a uma participação - pensando na colaboração do cliente. Logo, a heurística de "**Assistência a rastreabilidade do projeto**" deixou a desejar nas ferramentas apresentadas.

Na categoria de Visualização de Rastreabilidade, importantes informações foram coletadas a partir das heurísticas e suas questões. Pensando nos pilares da informação, para o apoio à representação de rastreabilidade, percebeu que apenas uma ferramenta (ID: 13) possibilitava a criação de uma árvore a partir dos cartões que representam tarefas. Entretanto, a criação dessa árvore se tornou manual e trabalhosa, não sendo gerada com base nos relacionamentos definidos nos cartões.

Algumas informações das propriedades de artefatos e de seus relacionamentos são, em sua grande maioria, dados percebidos a partir da visualização dos quadros de tarefas, como de Kanban e Sprint. Como dito anteriormente, sendo os cartões (retratando tarefas) a representação de todos os artefatos - com exceção de uma ferramenta (ID: 19); os quadros apresentaram estes cartões de tal forma que algumas propriedades poderiam ser verificadas antes mesmo de abri-los, como responsável, estimativa e dependência. Já os relacionamentos possuíam, em 7 das ferramentas, algum destaque visual, seja com uma figura no  cartões ou uma cor em cima da palavra que representava o relacionamento (por exemplo: *required*, *blocked*).

Apesar da possibilidade da dependência entre as tarefas ser verificada nos quadros de tarefas disponíveis, esse relacionamento era verificado através de uma propriedade da tarefa, não ficando claro o relacionamento com a origem e destino de forma visual, percebendo apenas existência de uma dependência. Contudo, ainda sim foram consideradas como visualizações, mas dentre as analisadas, nenhuma refletiu os possíveis cadastros de rastreabilidade em uma única visualização na intenção de verificar apenas relacionamentos.

As ferramentas apresentaram diversos tipos de visualização quando tratou-se de andamento do projeto, controle das tarefas, tempo gasto e produtividade dos usuários. Estas visualizações muitas vezes podiam ser combinadas. Entretanto, elas eram geradas com base em rastros de dados específicos de um usuário ou tarefa, não apresentando uma visualização de rastros entre artefatos.

Com isso, embora algumas ferramentas apresentem as características mencionadas nas questões apresentadas na heurística "**Adaptabilidade aos pilares definidos da informação**", houve pouco ou nenhuma diversidade relacionada a representação visual das

informações da rastreabilidade relacionada com as recomendações definidas no Capítulo 4. Isto pode trazer impactos significativos a uma tomada de decisão emergente, quando considerado o volume de informações textuais pouco relevantes ou representativas mostrados em quadros de tarefas.



Possibilidade de uma visualização para análise impactos diante uma alteração foi percebida apenas em uma ferramenta (ID: 13). Sendo ela a que disponibiliza a criação de uma árvore a partir de um cartão. Entretanto, esse impacto só podia ser analisado na estrutura hierárquica e de um cartão específico, resultando nos problemas que uma análise de impactos a partir das árvores pode ter (ver capítulo 4). Além disso, a geração dessa árvore de cartões era manual, não sendo gerada a partir dos relacionamentos previamente cadastrados nos próprios cartões.

Visualizações do envolvimento de *stakeholders* (desconsiderando o responsável pela execução do artefato) não foi possível em nenhuma das ferramentas. Contudo, quando tratado o responsável pela execução de uma tarefa, 5 ferramentas apresentaram visualizações, que podem ser novamente resumidas aos quadros de tarefas disponíveis, onde através de uma mostra dos cartões era possível verificar os responsáveis sem abri-los, apenas com a imagem do perfil do usuário representante.

Tais informações fazem parte da heurística de "**Assistência à seleção de artefatos e à identificação de repercussão das ações**", sendo esta não contemplada em parte por algumas ferramentas, pois possuem maneira de visualizar a informação tratada através do quadro, mas em alguns casos, para as perguntas levantadas, a maioria não possui respostas a estas; como por exemplo a identificação dos *stakeholders* envolvidos e análise de artefatos afetados diante uma alteração.

Outra característica relacionada a visualização e percebida nas ferramentas foi a possibilidade que todas apresentaram em demonstrar as situações de maneira gráfica. Em 12 delas isto era possível através da visualização do quadro de gerenciamento e planejamento de artefatos com a raias disponíveis de acordo com o status da tarefa, às vezes acrescentadas de cores. Em uma única ferramenta (ID: 19), apenas o uso de cores foi dado, por não apresentar o quadro.

Enquanto, visualizações que refletissem informações sobre por quem a tarefa foi priorizada, artefatos relacionados a uma participação e artefato que não estão alocados ao sistema, na maioria, não foram encontradas - como é possível ver na Figura 26. Já a identificação do tipo de demanda que ocorre era facilmente visualizada a partir dos quadros de tarefas já discutidos, com exceção de 3 ferramentas (ID: 12, 13, 15).

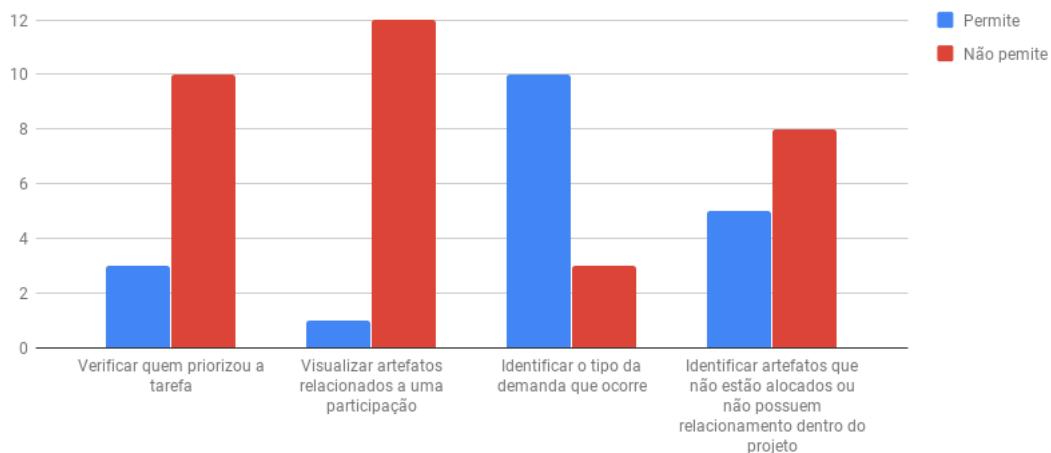


Figura 26: Apoio de visualizações para a tomada de decisão para planejamento e controle dos artefatos

Dessa forma, a heurística da categoria de Visualização da Rastreabilidade, "**Apoio à tomada de decisão para planejamento e controle dos artefatos**", não foi atendida por, em sua maioria, não permitir uma visualização com os dados julgados importantes nas entrevistas - como é possível ver na figura acima (Figura 26).

Em se tratando de tempo, em relação a visualização de acompanhamento sobre produtividade de um indivíduo, 10 ferramentas geravam visualizações que permitissem essa análise; outras 10 ferramentas possibilitavam a visualização da produtividade da equipe; 7 permitiram o monitoramento do tempo gasto nas tarefas; 12 possibilitavam a identificação de atribuições de um membro e 12 possuíam quadros de tarefas para o acompanhamento do projeto ou da Sprint. Enquanto que nenhuma delas permitiam fazer a análise dos *stakeholders* que precisam ser informados no caso de uma alteração solicitada em um artefato, ou proporcionou a visualização do que foi produzido em um dado período de tempo, com seus relacionamentos e responsáveis.

Apesar das informações de produtividade, trabalho realizado e acompanhamento da equipe serem importantes e discutidas nas entrevistas realizadas, visualizações às informações relevantes relacionadas ao conhecimento dos *stakeholders* que precisam ser informados sobre uma alteração e relacionamentos entre o que foi produzido com seus responsáveis num período de tempo não foram encontradas nas ferramentas analisadas. Com isso, a heurística "**Apoio à tomada de decisão para gestão de recursos, incluindo humanos e tempo**" não foi contemplada.

A visualização de dependências entre de interprojetos não foi identificada nas ferramentas, sendo a sua criação vista numa minoria - como citado inicialmente nesta seção.



Da mesma forma ocorreu ao analisar a visualização de *milestones*, sendo esse vista em apenas 4 ferramentas (ID: 01, 03, 04, 05), onde na realidade eram representadas a partir da realização de um filtro nos quadros de tarefas. A visualização de um histórico do projeto foi percebida por 10 ferramentas e, quadros de tarefas foram verificados em 12. Para a heurística "**Flexibilidade nos pontos de vista do projeto**" as ferramentas atenderem em partes, mesmo não contemplando alguns pontos tratados na heurística, dados relevantes ainda eram trabalhados, como visualização para histórico do projeto e quadros de tarefas que são bastante utilizados em cenários ágeis.

Refletindo a facilidade de interação do usuário com a visualização, perguntas voltadas para naveabilidade e manipulação de dados foram realizadas. Percebeu que a maioria das ferramentas apresentou mecanismos para manipulação direta de informações contidas nos artefatos, apenas 3 (ID: 01, 03, 19) não havendo esta facilidade. Muitas possibilitaram a navegação entre informações do artefato, por exemplo, a partir de uma tarefa era possível chegar a outras que estavam vinculadas apenas com um clique sobre elas. Para a navegação entre as informações somente 1 não possuía (ID: 04). A navegabilidade da pós-rastreabilidade dos artefatos foi mais comumente vista, quando comparada a pré. Tais fatos podem ser verificados na Figura 27.

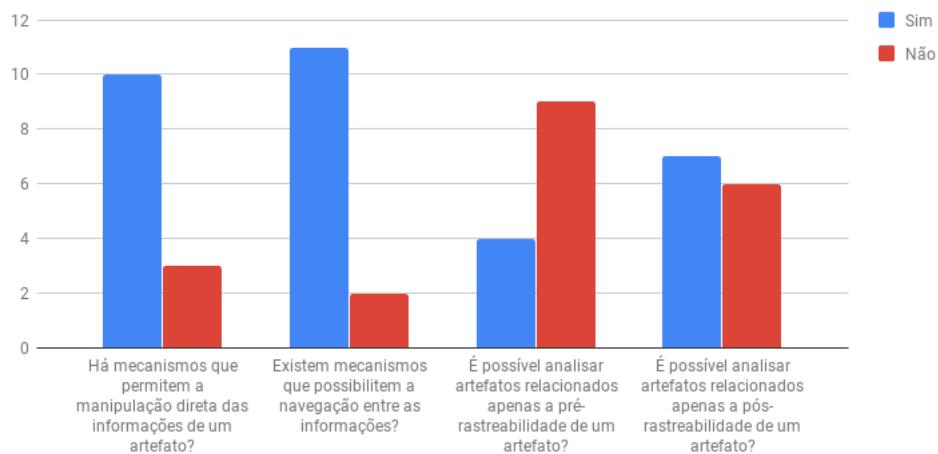


Figura 27: Auxílio à manipulação e à navegabilidade nas informações de rastreabilidade

Apesar da sua maioria oferecer mecanismos para manipulação direta de informações e navegação entre elas, algumas ferramentas deixaram a desejar quando tratado com foco em rastreabilidade entre os artefatos. Esta é uma característica importante na análise de impactos de um artefato e que foi tratada como faltante nas ferramentas também utilizadas pelos entrevistados. Logo, a heurística "**Auxílio à manipulação e à navegabilidade nas informações de rastreabilidade**" foi atendida apenas no auxílio à manipulação,

não sendo encontrada no apoio a navegabilidade das informações de rastreabilidade.

Por último, pontos com relação a redução da carga cognitiva para busca de informações foram analisados (ver Figura 28). Percebeu que representações para os graus de relacionamento não foram apresentados em nenhuma das ferramentas. Entretanto, foi visto que algumas ferramentas apresentavam elementos para diferenciar os tipos de relacionamentos entre artefatos utilizando ícones e cores; assim como em diferentes tipos de artefatos.

Apenas uma ferramenta (ID: 13) não tinha configurações de cores, texturas, tamanhos ou posições para suportar diferentes tipos do artefato. Em se tratando da diferenciação das situações dos artefatos, percebeu que o uso de quadros de tarefas, em 12 ferramentas, faziam a utilização da configuração de posicionamentos em raias com base nas situações. Entretanto uma delas utilizou cores para representar a situação do artefato (ID: 13). Agrupamentos, nas 10 ferramentas que possibilitaram o uso, foram verificados com a utilização de *labels* e *epics*. Essas *labels* ou *epics*, na sua maioria, possibilitavam a configuração de cores, dando-se um destaque quando os cartões se encontravam presentes no quadro de tarefas.

Existem possibilidades de configuração de cores, texturas, tamanhos ou posição para...

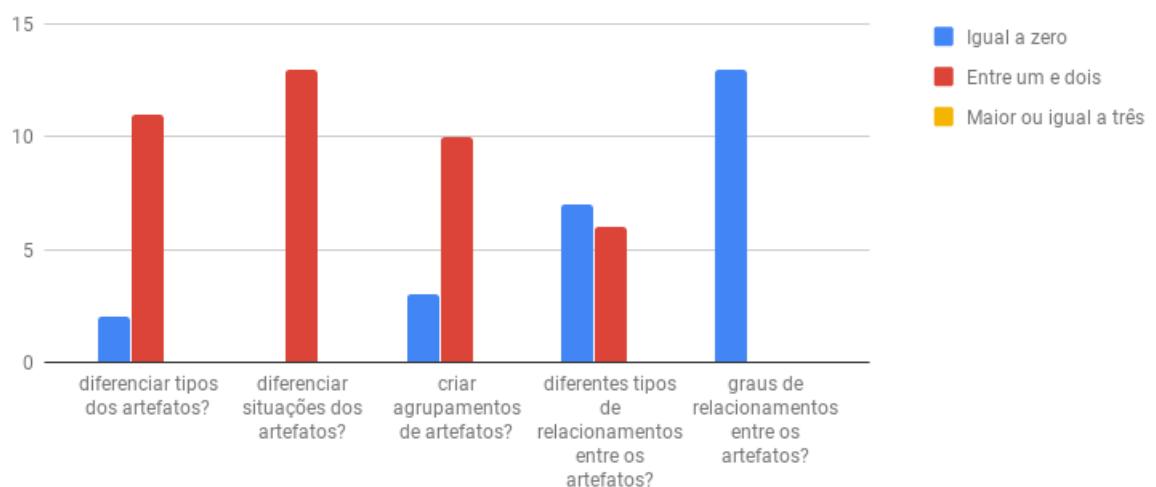


Figura 28: Apoio na redução de carga cognitiva

O uso dessas características também devem ser analisadas ao pensar na sobrecarga inicial quando o usuário se depara com mecanismos antes não utilizados. Apesar disso, sabe-se que após o aprendizado ou compreensão do usuário com os mecanismos utilizados,

a sobrecarga inicial é compensada. Mas a utilização de mecanismos consolidados podem facilitar que essa compreensão seja absorvida de forma mais rápida.

Além desses mecanismos, também atentou-se a possibilidades de consultas e filtragens das informações apresentadas na visualização. Apenas uma ferramenta (ID: 19) não foi considerada por não apresentar uma visualização dos artefatos, sendo apenas uma listagem. Somando-se a isso, analisou também a possibilidade de cadastrar consultas recurrentes, amenizando esforços de criação de consultas utilizadas com certa frequência. Apenas duas ferramentas (ID: 12, 13) possibilitaram o cadastramento dessas consultas, já as demais apresentavam apenas filtros pré-definidos ou campos gerais de busca.

A construção de diagramas e representações gráficas de forma automática se obteve em todas as ferramentas. Entretanto, o foco era mais voltado para produtividade, esforço e tempo relacionado a indivíduos e pessoas. Assim, não havendo uma geração automática de visualização própria para links de rastreabilidade. Apesar disso, com estes pontos tratados, a heurística "**Redução na carga cognitiva**" foi contemplada, pois todas as ferramentas, de alguma maneira, apresentaram recursos que ajudavam na visualização que está sendo apresentada; sendo eles desde ícones a cores e posicionamentos.

5.4 Considerações

Os resultados deste estudo apontaram que as heurísticas, cujas a temática foi rastreabilidade, foram parcialmente atendidas pelas ferramentas analisadas. Enquanto que as heurísticas identificadas para visualização das informações de rastreabilidade, praticamente, foram ignoradas.

Entre as visualizações possíveis na literatura e recomendadas por esta pesquisa, em sua totalidade, não foram contempladas nas visualizações empregadas pelas ferramentas analisadas. Isto pode impactar, e não permite que todo potencial do uso das informações de rastreabilidade seja usufruído para gestão de projetos de desenvolvimento de *software*.

O uso de quadros de tarefas foram um destaque quando tratado sobre a visualização de informações da rastreabilidade, pois este era um dos únicos recursos para a percepção de rastros. Mas, o foco destes quadros não estão relacionados à rastreabilidade, sendo está, algumas vezes apresentadas no quadro sem o destaque e a simplicidade na verificação de *links*, observando o artefato de origem e destino.

Assim, as necessidades trazidas da indústria não foram atendidas pelas ferramentas

avaliadas. O discurso de que as informações de rastreabilidade são difíceis de serem analisadas pela necessidade de navegar entre tarefas buscando estes *links*, é válido para as ferramentas aqui tratadas.

6 Considerações finais

A gestão de projetos ágeis de *software*, como quais outras, deveria ser feita com base em promover a participação, a responsabilidade e a autonomia dos interessados nas atividades de desenvolvimento de *softwares*. Outrossim, é preciso gerenciar outros recursos que não os humanos, para garantir que esse desenvolvimento seja mais eficiente e eficaz o possível. Nestes ambientes, o transitório e a urgência são recorrentes em todo o ciclo de vida esperado para um projeto de *software*. As ações, então, precisam ser controladas e precisas para obtenção dos melhores resultados.

Assim, as metodologias ágeis de desenvolvimento de *software* recebem novos adeptos por atender essas expectativas, como indicado por Franceschi e Duarte (2011). Entretanto, para que isto fosse possível, práticas e artefatos das metodologias tradicionais precisaram ser suprimidos ou adaptados. Além do mais, com a priorização de tarefas e de artefatos, algumas informações não são formalizadas, com o propósito flexível e universalmente de recuperação destas. Logicamente, houve impacto no gerenciamento da informação e do conhecimento relacionado ao projeto e seu possível reuso.

Em um passado não muito distante, a solução dada foi para os desafios da rastreabilidade de *software*. Esta rastreabilidade visava relacionar os artefatos existentes, e permitir algumas facilidades a, por exemplo, manutenção e evolução dos artefatos, entendimento dos requisitos, e verificação dos impactos de uma mudança. Sendo que, para profissionais de ambientes ágeis, consideraram as atividades relacionadas à esta prática burocráticas, exigindo um esforço comparado a feitura das documentações tradicionais, e desnecessárias, considerando que as informações produzidas muitas vezes não consultadas dada a dificuldade de entender e localizar o que se buscava dentro conteúdo armazenado (JACOBSSON, 2009).

Assim, a combinação do uso de ferramentas de gerenciamento de projetos (LUCIA; QUSEF, 2010) e a visualização das informações de rastreabilidade (THOMMAZO et al., 2012; HERNANDES, 2009; GERSHON; EICK; CARD, 1998), mostraram-se como uma alternativa a

este cenário.

6.1 Principais contribuições da pesquisa

As principais contribuições deste trabalho são: i) a confirmação da importância da rastreabilidade de *software* em ambientes ágeis e, de que a presença de visualizações das informações desta rastreabilidade adequadas podem trazer melhorias à gestão de projetos de *software* com base nas entrevistas realizadas; ii) as heurísticas definidas a partir da realização de uma análise das entrevistas com gerentes e líderes de equipes adaptada na abordagem metodológica de Teoria Fundamentada (Charmaz, 2014); iii) as recomendações de visualizações mais adequadas a informações de rastreabilidade, no contexto estudado e em sinergia com literatura; e iv) a verificação das ferramentas utilizadas para gerenciamento de projetos ágeis de *software*, de acordo com as heurísticas definidas e as recomendações feitas com o foco na rastreabilidade de *software* e na visualização das informações desta rastreabilidade.

Esta confirmação (i) foi necessária, para que não houvesse um investimento de tempo e de trabalho na trajetória planejada em vão. A investigação feita trouxe elementos que permitiram a reflexão sobre a relação existente entre necessidades, tempo e esforços investidos por profissionais da indústria de *software* quando na gestão de projetos. Além disto, evidenciou quais eram as estratégias, as dificuldades, os artefatos, as informações disponíveis e o ferramental empregados cotidianamente nas atividades desempenhadas por gerentes e líderes de projetos de *software*.

As heurísticas (ii) foram uma consequência das categorias emergidas da Teoria Fundamentada na análise das entrevistas, e não tiveram como intenção esgotar o assunto. Entretanto, estas podem ser ditas sensíveis para os profissionais, que participaram das entrevistas. Sendo assim, estas também foram consideradas para recomendações das visualizações das informações de rastreabilidade de *software* (iii), respeitando os pilares da informação: problemas, tipos de dados, intenção e interessado (WINKLER, 2008; LI; MAALEJ, 2012; SPENCE, 2014). Ressalta-se que estas recomendações também foram feitas com base nas justificativas presentes na literatura.

Com isto, foi feita a verificação das ferramentas utilizadas no contexto proposto (iv), cujo resultado indicou a existência de certa preocupação em facilitar a recuperação da informação e oferecer mecanismos de visualização das informações armazenadas. Todavia, esta verificação revelou que, por motivos desconhecidos, a maioria das informações

relativas à rastreabilidade de *software* foram desprezadas na proposição de visualizações adequadas para o perfil do público alvo na gestão de projetos de *software*.

Este achado nas ferramentas junto às discussões realizadas nas entrevistas deu destaque ao discurso de alguns entrevistados, onde demonstraram desanimados com o uso de dados da rastreabilidade pela dificuldade de se obter essas informações na própria ferramenta, preferindo assim, a busca pelo conhecimento tácito.

6.2 Limitações

As limitações desta pesquisa estão relacionadas com a quantidade e a regionalidade dos participantes das entrevistas, bem como a avaliação das recomendações efetuadas. Considerando o instrumento utilizado para investigação junto ao público alvo, a maior abrangência de participantes ineficiente com relação a números, entretanto mostrou-se conveniente ao propósito de investigação exploratória e na riqueza de informações obtidas. Desta forma, julga-se necessário expandir esta pesquisa a partir da aplicação de questionários online, por exemplo, para inclusive mitigar a regionalidade dos participantes.

Observando que foi realizada uma verificação das recomendações para visualização das informações da rastreabilidade de software em ferramentas utilizadas para gerenciamento de projetos pelo público alvo desta pesquisa, julga-se pertinente aplicar uma avaliação com estes profissionais para averiguar a conformidade destas recomendações com as necessidades e intenções levantadas pelas entrevistas. Ademais, poderia ser útil investigar os motivos pelos quais os resultados da verificação das ferramentas indicaram a falta de preocupação na visualização das informações da rastreabilidade de software especificamente.

6.3 Trabalhos futuros

Como trabalhos futuros, há uma intenção de expandir a pesquisa realizada na indústria com o objetivo de atingir uma maior diversidade de participantes, assim como um maior volume de informações. Para isto, pensa-se na realização de questionários online, na intenção de tingir uma maior público com a diversidade nas regionalidades. Além disso, pretende efetuar uma avaliação da visualização proposta junto ao público alvo, tendo a intenção de certificar se a visualização sugerida atenda às necessidades e propósitos trazidos pelas entrevistas.

Somando a isso, tenciona-se investigar os motivos pelos quais as ferramentas apresentadas nesta pesquisa não apresentam uma visualização voltada para a informação da rastreabilidade, quando em partes, tem a preocupação de coletar esses dados. E, por fim, também pretende-se aumentar a análise das ferramentas, acrescentando a elas: Asana, Aha!, Team Work Project, Podium e IceScrum

6.4 Conclusão Final

A pesquisa realizada respondeu às questões de pesquisas definidas para problemática observada, as delimitações e os recortes feitos. Considerando a complexidade e as diferentes dimensões possíveis desta problemática, a solução proposta é apenas uma primeira resposta. Assim, entende-se que a problemática não foi extinta, há muito a ser investigado. A saber, as informações da rastreabilidade de software, considerando os diferentes artefatos, preponderantes às atividades de gestão e liderança de equipes de desenvolvimento de software presentes em ambientes ágeis (QP1) foram aquelas relacionadas às ações de: i) definição de demandas e artefatos; ii) engenharia de requisitos; iii) gerenciamento de recursos; iv) participação dos stakeholders; v) rastreabilidade de artefatos; vi) uso de repositórios; vii) utilização de metodologias ágeis; e viii) visualização de informações da rastreabilidade. Estas informações foram apresentadas e discutidas no Capítulo 3.

Com relação às visualizações da informação da rastreabilidade de software que podem apoiar e otimizar atividades de gestão e liderança de equipes de desenvolvimento de software presentes em ambientes ágeis (QP2), estas visualizações foram identificadas e verificadas no Capítulo 4. Além disto, foi verificado a presença destas visualizações em ferramentas de gerenciamento de projetos. O resultado desta verificação apontou para a existência de informações relacionadas à rastreabilidade de software, entretanto foi observada a ausência de visualizações apropriadas, como defendida por esta pesquisa, para estas informações.

Referências

- ABRAHAMSSON, P. et al. Agile software development methods: Review and analysis. *arXiv preprint arXiv:1709.08439*, 2017.
- ABRAHAMSSON, P. et al. New directions on agile methods: a comparative analysis. In: IEEE. *Software Engineering, 2003. Proceedings. 25th International Conference on*. [S.l.], 2003. p. 244–254.
- AHMAD, G.; SOOMRO, T. R.; BROHI, M. N. Agile methodologies: Comparative study and future direction. *European Academic Research*, v. 1, n. 11, p. 3826–3841, 2014.
- ALVES, S. de R.; ALVES, A. L. Engenharia de requisitos em métodologias ágeis. 2009.
- ANTONINO, P. O. et al. A non-invasive approach to trace architecture design, requirements specification and agile artifacts. In: IEEE. *Software Engineering Conference (ASWEC), 2014 23rd Australian*. [S.l.], 2014. p. 220–229.
- BACKES, J. *Rastreabilidade semi-automática através do mapeamento de entidades*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Sul, 2008.
- BASSI, D. L. F. *Experiências com desenvolvimento ágil*. Dissertação (Mestrado) — São Paulo: IME-USP, Dissertação (Mestrado em Ciências da Computação), Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo, 2008.
- BECK, K. et al. *Manifesto for agile software development*. 2001. Disponível em: <<http://agilemanifesto.org/>>. Acesso em Dezembro 4, 2017.
- BECK, K. et al. *Manifesto for agile software development*. 2001. Disponível em: <<http://agilemanifesto.org/history.html>>. Acesso em Dezembro 4, 2017.
- BELMONTE, N. *The Javascript Visualization Toolkit*. 2011.
- BERG, K. van den; CONEJERO, J. M.; HERNÁNDEZ, J. Analysis of crosscutting across software development phases based on traceability. In: ACM. *Proceedings of the 2006 international workshop on Early aspects at ICSE*. [S.l.], 2006. p. 43–50.
- CAO, L.; RAMESH, B. Agile requirements engineering practices: An empirical study. *IEEE software*, IEEE, v. 25, n. 1, 2008.
- CHARMAZ, K. *Constructing grounded theory*. [S.l.]: Sage, 2014.
- CHEN, C. *Information visualization: Beyond the horizon*. [S.l.]: London: Springer-Verlag, 2004.

- CHEN, X.; HOSKING, J.; GRUNDY, J. Visualizing traceability links between source code and documentation. In: IEEE. *Visual Languages and Human-Centric Computing (VL/HCC), 2012 IEEE Symposium on*. [S.l.], 2012. p. 119–126.
- CISCON, L. A. *Um Estudo E Uma Ferramenta De Gerência De Projetos Com Desenvolvimento Ágil De Software*. Tese (Doutorado) — Tese De Mestrado. Universidade Federal De Minas Gerais, 2009.
- CLELAND-HUANG, J. et al. Software traceability: trends and future directions. In: ACM. *Proceedings of the on Future of Software Engineering*. [S.l.], 2014. p. 55–69.
- COCKBURN, A.; HIGHSMITH, J. Agile software development, the people factor. *Computer*, IEEE, v. 34, n. 11, p. 131–133, 2001.
- COHEN, D.; LINDVALL, M.; COSTA, P. Agile software development. *DACS SOAR Report*, v. 11, 2003.
- COHEN, D.; LINDVALL, M.; COSTA, P. An introduction to agile methods. *Advances in computers*, Elsevier, v. 62, p. 1–66, 2004.
- COHN, M. *User stories applied: For agile software development*. [S.l.]: Addison-Wesley Professional, 2004.
- CORAM, M.; BOHNER, S. The impact of agile methods on software project management. In: IEEE. *Engineering of Computer-Based Systems, 2005. ECBS'05. 12th IEEE International Conference and Workshops on the*. [S.l.], 2005. p. 363–370.
- COSTA, G. H. d. C. *Engenharia de Requisitos no Desenvolvimento de Software Ágil*. Dissertação (Mestrado) — Universidade Federal de Pernambuco, Centro de Informática, 2011.
- CRAWFORD, L. Developing organizational project management capability: theory and practice. *Project Management Journal*, SAGE Publications Sage CA: Los Angeles, CA, v. 37, n. 3, p. 74–86, 2006.
- CYSNEIROS, G. A. d. A. F.; ZISMAN, A. D3traceview: A traceability visualization tool. 2017. 
- DALL'OGLIO, P. *Uma Ferramenta para Gerenciamento de Requisitos em Projetos Baseados em Extreme Programming*. Tese (Doutorado) — Tese de Doutorado. Universidade do Vale do Rio dos Sinos, 2006.
- DALL'OGLIO, P.; SILVA, J. P. S. da; PINTO, S. C. C. da S. Um modelo de rastreabilidade com suporte ao gerenciamento de mudanças e análise de impacto. In: *WER*. [S.l.: s.n.], 2010.
- DELATER, A.; NARAYAN, N.; PAECH, B. Tracing requirements and source code during software development. In: *In Proc. 7th Int. Conf. of Software Engineering Advances, Lissabon*. [S.l.: s.n.], 2012. p. 274–282.
- DENZIN, N. K.; LINCOLN, Y. S. Introdução: a disciplina e a prática da pesquisa qualitativa. In: *O planejamento da pesquisa qualitativa: teorias e abordagens*. [S.l.]: Artmed, 2006.

- DEY, I. *Grounding grounded theory: guidelines for grounded theory inquiry*. [S.l.]: San Diego, USA: Academic Press, 1999.
- DUC, V. H. *Traceability in Agile software projects*. Dissertação (Mestrado) — Master of Science Thesis in Software Engineering Management, 2013.
- EBERLEIN, A.; LEITE, J. Agile requirements definition: A view from requirements engineering. In: *Proceedings of the International Workshop on Time-Constrained Requirements Engineering (TCRE'02)*. [S.l.: s.n.], 2002. p. 4–8.
- EGYED, A. A scenario-driven approach to trace dependency analysis. *IEEE Transactions on Software Engineering*, IEEE, v. 09, n. 2, p. 116–132, 2003.
- EGYED, A.; GRÜNBACHER, P. Supporting software understanding with automated requirements traceability. *International Journal of Software Engineering and Knowledge Engineering*, World Scientific, v. 15, n. 05, p. 783–810, 2005.
- ERICKSON, J.; LYYTINEN, K.; SIAU, K. Agile modeling, agile software development, and extreme programming: the state of research. *Journal of database Management*, IGI Global, v. 16, n. 4, p. 88–100, 2005.
- ESPINOZA, A.; GARBAJOSA, J. A study to support agile methods more effectively through traceability. *Innovations in Systems and Software Engineering*, Springer, v. 7, n. 1, p. 53–69, 2011.
- FAGUNDES, P. B. *Framework para comparação e análise de métodos ágeis*. Dissertação (Mestrado) — Universidade Federal de Santa Catarina, 2005.
- FEITOSA, M. G. G. Maturidade no gerenciamento de projetos: um estudo das práticas existentes nos órgãos do governo de pernambuco. *Revista de Gestão e Projetos-GeP*, v. 3, n. 2, p. 207–234, 2012.
- FILHO, A. d. A. G.; LENCASTRE, M. Towards a traceability visualisation tool. In: IEEE. *Quality of Information and Communications Technology (QUATIC), 2012 Eighth International Conference on the*. [S.l.], 2012. p. 221–223.
- FOWLER, M. *The new methodology*. 2005. Disponível em: <<http://www.martinfowler.com/articles/newMethodology.html>>. Acesso em Dezembro 1, 2015.
- FOWLER, M.; HIGHSMITH, J. The agile manifesto. *Software Development*, [San Francisco, CA: Miller Freeman, Inc., 1993-], v. 9, n. 8, p. 28–35, 2001.
- FRANCESCHI, R. A.; DUARTE, A. M. D. Uma abordagem para gerência de requisitos integrada com práticas ágeis de gerência de projetos. 2011.
- FURTADO, F.; ZISMAN, A. Trace++: A traceability approach to support transitioning to agile software engineering. In: IEEE. *Requirements Engineering Conference (RE), 2016 IEEE 24th International*. [S.l.], 2016. p. 66–75.
- GARCIA, J. A. E. Software development and collaboration: Version control systems and other approaches. 2011.

- GENVIGIR, E. C. *Um modelo para rastreabilidade de requisitos de software baseado em generalização de elos e atributos*. Tese (Doutorado) — São José dos Campos: Instituto Nacional de Pesquisas Espaciais, 2009.
- GERSHON, N.; EICK, S. G.; CARD, S. Information visualization. *interactions*, ACM, v. 5, n. 2, p. 9–15, 1998.
- GHAZARIAN, A. Traceability patterns: an approach to requirement-component traceability in agile software development. In: WORLD SCIENTIFIC AND ENGINEERING ACADEMY AND SOCIETY (WSEAS). *Proceedings of the 8th conference on Applied computer scince*. [S.l.], 2008. p. 236–241.
- GHAZARIAN, A. *Reliability in agile software engineering: A dilemma*. [S.l.], 2010.
- GLASER, B. G. *Basics of grounded theory analysis: Emergence vs forcing*. [S.l.]: Sociology press, 1992.
- GONÇALVES, E. dos S.; FILHO, H. B. dos R. Ferramenta para gerenciamento de requisitos em metodologias ágeis. *HÍFEN*, v. 32, n. 62, 2008.
- GOTEL, O. et al. Glossary of traceability terms (v1. 0). *Software and Systems Traceability*, Springer, p. 413, 2012.
- GOTEL, O.; FINKELSTEIN, A. Extended requirements traceability: Results of an industrial case study. In: IEEE. *Requirements Engineering, 1997., Proceedings of the Third IEEE International Symposium on*. [S.l.], 1997. p. 169–178.
- GOTEL, O. C.; FINKELSTEIN, C. An analysis of the requirements traceability problem. In: IEEE. *Requirements Engineering, 1994., Proceedings of the First International Conference on*. [S.l.], 1994. p. 94–101.
- GOTEL, O. C.; MARCHESE, F. T.; MORRIS, S. J. On requirements visualization. In: IEEE. *Requirements Engineering Visualization, 2007. REV 2007. Second International Workshop on*. [S.l.], 2007. p. 11–11.
- GOTEL, O. C.; MARCHESE, F. T.; MORRIS, S. J. The potential for synergy between information visualization and software engineering visualization. In: IEEE. *Information Visualisation, 2008. IV'08. 12th International Conference*. [S.l.], 2008. p. 547–552.
- GOTEL, O. C. Z. *Contribution structures for requirements traceability*. Tese (Doutorado) — Imperial College, 1995.
- HAN, J.; PEI, J.; KAMBER, M. *Data mining: concepts and techniques*. [S.l.]: Elsevier, 2011.
- HEIM, P. et al. Graph-based visualization of requirements relationships. In: IEEE. *Requirements Engineering Visualization, 2008. REV'08*. [S.l.], 2008. p. 51–55.
- HERMAN, I.; MELANÇON, G.; MARSHALL, M. S. Graph visualization and navigation in information visualization: A survey. *IEEE Transactions on visualization and computer graphics*, IEEE, v. 6, n. 1, p. 24–43, 2000.

- HERNANDES, E. *Automated Data Processing And Conceptualization With Support From Ontologies Visualization*"(in portuguese). 2009.
- HIGHSITH, J. A. *Agile software development ecosystems*. [S.l.]: Addison-Wesley Professional, 2002.
- HOLTEN, D. Hierarchical edge bundles: Visualization of adjacency relations in hierarchical data. *IEEE Transactions on visualization and computer graphics*, IEEE, v. 12, n. 5, p. 741–748, 2006.
- JACKO, J. A. *Human computer interaction handbook: Fundamentals, evolving technologies, and emerging applications*. [S.l.]: CRC press, 2012. 539 p.
- JACOBSSON, M. *Implementing traceability in agile software development*. Dissertação (Mestrado) — Department of Computer Science, Lund University, 2009.
- JAQUEIRA, A. d. O. P. *Uso de modelos i* para enriquecer requisitos em métodos ágeis*. Dissertação (Mestrado) — Universidade Federal do Rio Grande do Norte, 2013.
- KÄÄRIÄINEN, J. et al. Supporting requirements engineering in extreme programming: managing user stories. In: *Proc. 16th International Conference on Software & Systems Engineering and their Applications (ICSSEA 2003)*. [S.l.: s.n.], 2003.
- KEIM, D. A. Information visualization and visual data mining. *IEEE Transactions on Visualization & Computer Graphics*, IEEE, n. 1, p. 1–8, 2002.
- KHAN, H. M.; ARSHAD, I. Test-lists utilization in test driven development: The role of test-lists in requirements traceability. 2012.
- KODALI, M. *Traceability of Requirements in Scrum Software Development Process*. Dissertação (Mestrado) — Malardalen University, School of Innovation Design and Engineering; Vasteras, Sweden, 2015.
- KOTONYA, G.; SOMMERVILLE, I. *Requirements engineering: processes and techniques*. [S.l.]: Wiley Publishing, 1998.
- KUGELE, S.; ANTKOWIAK, D. Visualization of trace links and change impact analysis. In: IEEE. *Requirements Engineering Conference Workshops (REW), IEEE International*. [S.l.], 2016. p. 165–169.
- LEE, C.; GUADAGNO, L.; JIA, X. An agile approach to capturing requirements and traceability. In: *Proceedings of the 2nd International Workshop on Traceability in Emerging Forms of Software Engineering (TEFSE 2003)*. [S.l.: s.n.], 2003.
- LETELIER, P. A framework for requirements traceability in uml-based projects. In: *Proc. of 1st International Workshop on Traceability in Emerging Forms of Software Engineering*. [S.l.: s.n.], 2002. p. 173–183.
- LI, Y.; MAALEJ, W. Which traceability visualization is suitable in this context? a comparative study. *Requirements Engineering: Foundation for Software Quality*, Springer, p. 194–210, 2012.
- LOELIGER, J. *Version Control with Git*. [S.l.: s.n.], 2009.

- LUCIA, A. D.; QUSEF, A. Requirements engineering in agile software development. *Journal of Emerging Technologies in Web Intelligence*, Academy Publisher, PO Box 40 Oulu 90571 Finland, v. 2, n. 3, p. 212–220, 2010.
- MADER, P.; EGYED, A. Assessing the effect of requirements traceability for software maintenance. In: IEEE. *Software Maintenance (ICSM), 2012 28th IEEE International Conference on*. [S.l.], 2012. p. 171–180.
- MARCUS, A.; XIE, X.; POSHYVANYK, D. When and how to visualize traceability links? In: ACM. *Proceedings of the 3rd international workshop on Traceability in emerging forms of software engineering*. [S.l.], 2005. p. 56–61.
- MERTEN, T.; JÜPPNER, D.; DELATER, A. Improved representation of traceability links in requirements engineering knowledge using sunburst and netmap visualizations. In: IEEE. *Managing Requirements Knowledge (MARK), 2011 Fourth International Workshop on*. [S.l.], 2011. p. 17–21.
- NAVES, F.; FURTADO, F. Tracepp requirements traceability tool: Extending the scalability of the trace++ technique. In: IEEE. *Information Systems and Technologies (CISTI), 2017 12th Iberian Conference on*. [S.l.], 2017. p. 1–7.
- NERUR, S.; MAHAPATRA, R.; MANGALARAJ, G. Challenges of migrating to agile methodologies. *Communications of the ACM*, ACM, v. 48, n. 5, p. 72–78, 2005.
- NETO, E. S. *Rastreabilidade de Requisitos*. 2013.
- NIU, N.; REDDIVARI, S.; CHEN, Z. Keeping requirements on track via visual analytics. In: IEEE. *Requirements Engineering Conference (RE), 2013 21st IEEE International*. [S.l.], 2013. p. 205–214.
- PAETSCH, F.; EBERLEIN, A.; MAURER, F. Requirements engineering and agile software development. In: IEEE. *Enabling Technologies: Infrastructure for Collaborative Enterprises, 2003. WET ICE 2003. Proceedings. Twelfth IEEE International Workshops on*. [S.l.], 2003. p. 308–313.
- PEREIRA, J. Desenvolvimento de um software para métricas em rastreabilidade de requisitos de software. 2011.
- PILGRIM, J. von et al. Constructing and visualizing transformation chains. In: SPRINGER. *European Conference on Model Driven Architecture-Foundations and Applications*. [S.l.], 2008. p. 17–32.
- POHL, K.; RUPP, C. *Fundamentos da Engenharia de Requisitos: Um guia de estudo para o Exame CPRE-FL (Certified Professional for Requirements Engineering- Fundation Level)*. [S.l.]: International Requirements Engineering Board (IREB), 2012.
- POPPENDIECK, M.; POPPENDIECK, T. *Lean Software Development: An Agile Toolkit: An Agile Toolkit*. [S.l.]: Addison-Wesley, 2003.
- PUNTEL, M.; PRASS, F. Um método ágil híbrido. 2010.

- QUSEF, A. Test-to-code traceability: Why and how? In: IEEE. *Applied Electrical Engineering and Computing Technologies (AEECT), 2013 IEEE Jordan Conference on.* [S.l.], 2013. p. 1–8.
- RAM, K. Git can facilitate greater reproducibility and increased transparency in science. *Source code for biology and medicine*, BioMed Central, v. 8, n. 1, p. 7, 2013.
- RAMESH, B.; JARKE, M. Toward reference models for requirements traceability. *IEEE transactions on software engineering*, IEEE, v. 27, n. 1, p. 58–93, 2001.
- RAO, N. R.; SEKHARAIAH, K. C. A methodological review based version control system with evolutionary research for software processes. In: ACM. *Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies.* [S.l.], 2016. p. 14.
- RODRIGUES, A.; LENCASTRE, M.; FILHO, A. d. A. G. Multi-visiotrace: Traceability visualization tool. In: IEEE. *Quality of Information and Communications Technology (QUATIC), 2016 10th International Conference on the.* [S.l.], 2016. p. 61–66.
- SAMPIERI, R. H.; COLLADO, C. H.; LUCIO, P. B. *Metodologia de pesquisa.* [S.l.: s.n.], 2013.
- SAYÃO, M.; LEITE, J. C. S. do P. Rastreabilidade de requisitos. *RITA*, v. 13, n. 1, p. 57–86, 2005.
- SHNEIDERMAN, B. Tree visualization with tree-maps: 2-d space-filling approach. *ACM Transactions on graphics (TOG)*, ACM, v. 11, n. 1, p. 92–99, 1992.
- SHNEIDERMAN, B. The eyes have it: A task by data type taxonomy for information visualizations. In: IEEE. *Visual Languages, 1996. Proceedings., IEEE Symposium on.* [S.l.], 1996. p. 336–343.
- SILLITTI, A.; SUCCI, G. 14 requirements engineering for agile methods. *Engineering and Managing Software Requirements*, Springer, p. 309–326, 2005.
- SILVA, F.; HOENTSCH, S. C.; SILVA, L. Uma análise das metodologias ágeis fdd e scrum sob a perspectiva do modelo de qualidade mps. br. *Scientia Plena*, v. 5, n. 12, 2009.
- SILVA, R. P. d. *Modelo de rastreabilidade de requisitos aplicada a gestão de projetos em métodos ágeis.* Dissertação (Mestrado) — Brasil, 2016.
- SINK, E. *Version control by example.* [S.l.]: Pyrenean Gold Press Champaign, IL, 2011.
- SIQUEIRA, F. *Métodos Ágeis.* 2003.
- SPENCE, R. *Information visualization.* [S.l.]: Springer, 2014.
- STASKO, J. T. et al. *An evaluation of space-filling information visualizations for depicting hierarchical structures.* [S.l.], 2000.
- TAROMIRAD, M.; PAIGE, R. F. Agile requirements traceability using domain-specific modelling languages. In: ACM. *Proceedings of the 2012 Extreme Modeling Workshop.* [S.l.], 2012. p. 45–50.

- THOMMAZO, A. D. et al. Geração automática da matriz de rastreabilidade de requisitos com suporte de visualização. 2012.
- TOMÁS, M. R. Métodos ágeis: características, pontos fortes e fracos e possibilidades de aplicação. IET, 2009.
- TRINDADE, G. O. da. *Requisitos para Ferramenta de Gerenciamento de Requisitos em Desenvolvimento Ágil a partir de um Estudo Exploratório*. 2015.
- TRINDADE, G. O. da; LUCENA, M. Rastreabilidade de requisitos em metodologias ágeis: um estudo exploratório. 2016.
- UGGEMANN-KLEIN, A.; WOOD, D. Drawing trees nicely with tex. *Electronic Publishing*, v. 2, n. 2, p. 101–115, 1989.
- URQUIZA-YLLESCAS, J.; MARTINEZ-MARTINEZ, A. Trying to link traceability elements in a general agile model life cicle. 2011.
- VOYTEK, J.; NÚÑEZ, J. L. Visualizing non-functional traces in student projects in information systems and service design. *CHI 2011*, 2011.
- WIJK, J. J. V.; WETERING, H. Van de. Cushion treemaps: Visualization of hierarchical information. In: *InfoVis*. [S.l.: s.n.], 1999. v. 99, p. 73–78.
- WILLIAMS, L.; COCKBURN, A. Guest editors' introduction: Agile software development: It's about feedback and change. *Computer*, IEEE Computer Society Press, v. 36, n. 6, p. 39–43, 2003.
- WINKLER, S. On usability in requirements trace visualizations. In: IEEE. rev. [S.l.], 2008. p. 56–60.
- WOHLIN, C. et al. *Experimentation in software engineering*. [S.l.]: Springer Science & Business Media, 2012.
- ZHOU, X. et al. Facilitating software traceability understanding with envision. In: IEEE. *Computer Software and Applications, 2008. COMPSAC'08. 32nd Annual IEEE International*. [S.l.], 2008. p. 295–302.

APÊNDICE A - Processos e Metodologias Ágeis

A Tab. 11 foi montada com base em informações coletadas principalmente em trabalhos como Fagundes (2005) e Abrahamsson et al. (2017), e completadas e analisadas pelos trabalhos: Bassi (2008), Kodali (2015), Ahmad, Soomro e Brohi (2014), Gonçalves e Filho (2008), Dall’Oglio (2006), Costa (2011), Ciscon (2009), Coram e Bohner (2005), Silva, Hoentsch e Silva (2009), Poppendieck e Poppendieck (2003), Siqueira (2003).

Tabela 11: Comparação das fases, práticas, papéis e artefatos de cada processo e metodologia ágil

Nome	Fases	Práticas	Papéis	Artefatos Gerados
Método de Desenvolvimento Sistemas Dinâmicos (DSDM)	Pré-projeto; Ciclo de vida do projeto: estudo de viabilidade, estudo de negócios, iteração de modelo funcional, iteração de concepção e construção e implementação; Pós-projeto	(chamadas de Princípios) envolvimento ativo do usuário é imperativo; poder de tomar decisões; entrega frequente; aptidão para fins comerciais é o critério essencial de aceitação de entregas; desenvolvimento imortal e incremental é necessário para convergir numa solução de negócios precisa; mudanças são reversíveis; requisitos são basilizados em um alto nível; teste e integração ao longo do ciclo de vida; uma abordagem colaborativa e cooperativa compartilhada por todas as partes interessadas é essencial	Desenvolvedores; Desenvolvedores Sênior; Coordenador Técnico; Usuário Embaixador; Usuário Conselheiro; Patrocinador Executivo; Visionário	Sugere a produção de no máximo 15 artefatos, incluindo Protótipos. Apresenta apenas propósitos, critérios de qualidade e breves descrições para cada, deixando a cargo da empresa ou equipe personalizar estes artefatos
Scrum	Pré-jogo (Planejamento e Arquitetura); Desenvolvimento (Sprints e Reuniões Diárias); Pós-jogo (Entrega)	<i>Backlog</i> do Produto; Estimativa de Esforço; <i>Sprint</i> ; Reunião de Planejamento da <i>Sprint</i> ; <i>Sprint Backlog</i> ; Reuniões Diárias; Revisão da <i>Sprint</i>	Dono do Produto; Time; Scrum-Master e Cliente	<i>Backlog</i> do Produto; <i>Sprint Backlog</i> ; <i>Release Backlog</i> ; <i>Backlog</i> de Impedimentos; <i>Burn-down chart</i> ; Anotações da Retrospectiva
Desenvolvimento Adaptativo de Software e Aprendizado (ASD)	Especulação, Colaboração e Aprendizado	(não possui um conjunto de práticas e sim propriedades) Dirigido a Missões; Planejamento Baseado em Funcionalidades; Iterativo, Prazos Pré-fixados; Tolerância a Mudanças; Dirigido a Riscos; <i>Time-Boxed</i> ; Revisões do Grupo de Foco do Cliente	(não foram encontrados papéis bem definidos, mas sugestão) Patrocinador de Execução; Facilitador; Escrita; Gerente de Projeto; Representantes do Cliente e Desenvolvedores	[não foram encontrados dados na literatura]

Continuação na próxima Página

Tabela 11 – continuação da página anterior

Nome	Fases	Práticas	Papéis	Artefatos Gerados
Desenvolvimento Orientado a Funcionalidade (FDD)	Desenvolver um Modelo Global; Construir uma Lista de Funcionalidades; Planejar a Construção por Funcionalidades; Projetar cada Funcionalidade; Construir cada Funcionalidade	Modelagem de Objetos de Domínio; Desenvolvendo através de Funcionalidades; Propriedade Individual da Classe; Equipes de Funcionalidades; Inspeções; Construções Regulares; Administração de Configuração; Relatório dos Resultados	Gerente de Projetos, Versão, Domínio, Desenvolvimento; Arquiteto Principal; Programador Chefe; Proprietário de Classe; Especialista no Domínio, na Linguagem; Coordenador de Configuração; <i>Toolsmith</i> ; Administrador de Sistema; Testadores; Desenvolvedores; Escritores Técnicos	Plano de Projeto, Lista de Funcionalidades e o Diagrama de Seqüência
Programação Extrema (XP)	Exploração; Planejamento; Iterações para Entrega; Produção; Manutenção; Fim do Projeto	Jogo de Planejamento; Entregas Pequenas; Metáfora; Projeto Simples; Teste; Refatoração; Programação em Pares; Propriedade Coletiva; Integração Contínua; Semana de 40 horas; Cliente Presente; Padrões de Codificação; Espaço Aberto; Regras Próprias	Programador; Cliente; Testador; Rastreador; Treinador; Consultor; Gerente	Histórias; Tarefas; Restrições; Testes de Aceitação; Código; Releases; Projeto UML; Metáforas; Documentação; Padrões; Testes Unitários; Planejamento; Plano de <i>Release</i> ; Plano de Iteração; Resultados de Teste; <i>Spikes</i> (soluções); Recursos; Escopo; Qualidade; Tempo; Resultados de Rastreamento
Desenvolvimento de Software Lean (LSD)	Inicio; Estado de Crescimento Regular; Transição e Renovação	(não define práticas, define "ferramentas") Vendo resíduos; Mapeamento de fluxo de valor; Desenvolvimento baseado em conjunto; Sistemas de tração; Teoria das filas; Motivação; Medições; TDD	[não foram encontrados dados na literatura]	[não foram encontrados dados na literatura]

Continuação na próxima página

Tabela 11 – continuação da página anterior

Nome	Fases	Práticas	Papéis	Artefatos Gerados
Crystal	[não foram encontrados dados na literatura]	Encenação; Edição e Revisão; Monitoramento do Processo; Paralelismo e Fluxo; Estratégia de Diversidade Holística; Técnica de Metodologia de Ajustamento; Inspeções de Usuários; <i>Workshops</i> Reflexivos; Produtos de Trabalho; Padrões; Ferramentas	Clear: Patrocinador, Usuário, Programador, Documentador, Testador, Coordenador, Especialista de Negócio e Coletor de Requisitos; Orange: Designer de Interface Gráfica, Projetista de Banco de Dados, Especialista em uso, Facilitador Técnico, Analista de Negócios, Arquiteto, Mentor de Design e Reuse Point, Escritor e Testador Sequência de <i>Release</i> , Modelos de Objetos Comuns, Manual do Usuário, Casos de Teste e Código de Migração	Clear: Casos de Uso, Descrição de <i>Features</i> , Cronograma, Documentação de Design; Orange: Documento de Requisitos, Documentos de <i>Design UI</i> , Relatórios de Status, Agendamento

APÊNDICE B - Roteiro para Entrevista Presencial/Online

Dados Demográficos

DD1. Nome:

DD2. E-mail:

DD3. Idade:

DD4. Formação:

DD5. Instituição em que trabalha:

DD6. Tipo de cargo que ocupa atualmente: () Gerencial () Consultor interno () Consultor externo () Professor () Analista () Técnico () Administrativo () Outro:

DD7. Tempo de experiência profissional neste cargo: ano(s) e mês(es).

DD8. Trabalhou em outro cargo?

DD9. Qual(is)?

DD10. Tem alguma experiência em pesquisa(s) acadêmica(s)?

DD11. Que tipo(s) - iniciação científica, dissertação?

DD12. Qual(is) temáticas?

DD13. Você faz uso de repositório?

DD14. Se sim, qual?

DD15. Você faz uso de alguma ferramenta para gerenciamento de projetos ou de requisitos? Se sim, qual?

DD16. Você utiliza de alguma forma especial ou configurou a ferramenta para atender alguma demanda específica da sua equipe?

Metodologia de desenvolvimento e gerenciamento de requisitos

MG1. Quais metodologias de desenvolvimento são utilizadas em sua equipe?

MG2. Já teve a possibilidade de trabalhar ou conhecer a fundo outra metodologia?

– as próximas perguntas têm o objetivo de verificar sua importância e qual o motivo dela.

MG3. É possível saber quem são as pessoas envolvidas na produção do requisito? É importante ter essa informação? Por quê?

MG4. Como essas pessoas são envolvidas? - Isso é importante pra você? Por quê?

MG5. Se houver mudança de responsável, você teria como saber quem foi o responsável, quem é atualmente e quando houve a mudança? Como é o processo? Seria relevante ter essa informação? Por quê?

MG6. Quem precisa ser envolvido ou informado na mudança ou produção do requisito? Como você verificaria isso (documentação, definidos informalmente pelo cliente ou pelo gerente)? É importante ter essa informação? Por qual o motivo?

MG7. Para você seria relevante ter a sistematização dos desdobramentos relacionados à perda de um membro da equipe? (Lembrando que a perda de um membro na equipe pode levar a perda de conhecimento) É importante ter essa informação? Por quê?

MG8. Quem usa os requisitos levantados? Como utilizam? Saber isso é relevante para você? Por quê? Isso altera sua forma de tratar os requisitos?

MG9. Quais os papéis são desempenhados? Ter essa informação é importante? Por quê?

MG10. Em quais requisitos os stakeholders colaboram? É importante ter essa informação? Por quê?

MG11. Como e em que momento o cliente é envolvido? É relevante saber essa informação? Por quê?

Rastreabilidade

R1. Você tem conhecimentos sobre rastreabilidade? E sobre os tipos de rastreabilidade?

R2. No seu projeto há preocupação com a rastreabilidade?

R3. Quais tipos de rastreabilidade vocês usam? Por qual motivo?

Visualização

V1. O que você entende por visualização de rastros/links entre artefatos?

V2. Você utiliza algum tipo de visualização de dados para verificar rastros? Como: pessoas envolvidas, papéis desempenhados, mudanças de responsabilidades?

V3. Você gostaria de ter algum tipo de visualização sobre alguns dados que não tem hoje e não foi perguntado sobre?

Você aceitaria receber algumas questões por e-mail? () sim () não

Entrevistador(a): Data: / /

APÊNDICE C – Roteiro para entrevista por *e-mail*

Rastreabilidade

R4. Que tipos de rastros são verificados?

R5. Na sua opinião, qual a dificuldade de se manter a rastreabilidade dos requisitos?
Por quê?

R6. É importante para você saber a origem do requisito? Como você documenta(ria)?
Você faz uso de alguma ferramenta para auxiliá-lo nessa atividade?

R7. Quando é necessário refinar o requisito, é criado um relacionamento com o requisito origem?

R8. Qual o custo dentro da sua equipe de manter a rastreabilidade dos artefatos obtidos ou gerados? É compensado? Se sim, como? Caso contrário, por quê?

R9. Qual o papel você atribui ao uso da rastreabilidade dentro dos seus projetos?
Conte me como foi a experiência ou caso que você vivenciou.

Visualização

V4. Quais visualizações são importantes para você? Por qual motivo?

V5. Você faz uso de alguma estratégia ou ferramenta de visualização dos relacionamentos entre os artefatos do projeto? Qual o papel e em quais momentos são/foram importantes no gerenciamento do projeto? Alguma experiência para relatar? Caso contrário, ter uma visualização dos relacionamentos entre os artefatos seria útil no seu dia-a-dia?
Por que?

V6. A visualização poderia ser relevante para lhe ajudar a identificar problemas, impacto das mudanças ou soluções? Por quê? Se sim, quais tipos de visualização seriam mais adequadas nas classes indicadas por você?

APÊNDICE D – Modelos de *e-mails* para Entrevista

1. Convite para participação de entrevista

Assunto: Convite para participação de entrevista

Olá, <Nome do(a) convidado(a)>.

Eu me chamo Gabriela Trindade, e sou aluna do curso de mestrado em Sistemas e Computação, da Universidade Federal do Rio Grande do Norte (UFRN). Meu projeto de pesquisa está relacionado com a investigação, proposta e validação de uma abordagem de rastreabilidade e visualização de requisitos em metodologias ágeis.

Para esta fase da pesquisa, convido você a participar de uma entrevista sobre gerenciamento de requisitos em sistemas computacionais. As suas respostas serão úteis para definição da minha abordagem. Estimamos que essa entrevista deverá levar em torno de 1 hora. Assim, peço que escolha o melhor dia/horário para você.

Desde de já, agradeço a sua colaboração.

Gabriela Trindade.

2. E-mail de convite à entrevista por e-mail

Assunto: Algumas questões por e-mail

Olá, <Nome do(a) convidado(a)>.

Primeiramente, agradeço sua participação na entrevista para meu projeto de pesquisa. Como combinado na ocasião, solicito sua colaboração em responder às questões abaixo, considerando sua experiência profissional e acadêmica.

Agradeço sua colaboração e atenção,

Gabriela Trindade.

<Inserir as questões>

3.*E-mail* de reinteração de convite à entrevista por *e-mail* (INFORMAL)

[Resposta ao e-mail: Algumas questões por e-mail]

Olá, <Nome do(a) convidado(a)>.

Bem sei, que na correria cotidiana, algumas atividades ficam para depois. Entretanto como a mensagem também pode ter ido para sua caixa de spam, reencaminho minha solicitação de preenchimento das questões do e-mail enviado anteriormente.

Conto com a sua colaboração,

Gabriela Trindade.

4.*E-mail* de agradecimento pela participação

[Resposta ao e-mail: Algumas questões por e-mail]

Olá, <Nome do(a) convidado(a)>.

Agradeço sua participação na entrevista para meu projeto de pesquisa. Sua colaboração permitirá maior entendimento dos objetos desta pesquisa.

Cordialmente,

Gabriela Trindade.