



**MARIANA DE AZEVEDO SANTOS**

**AGILE UBPM FOR SCRUM: Modelo de Aprimoramento do Gerenciamento e  
Desenvolvimento Ágil Baseado na Percepção de Valor do Usuário**

**LAVRAS-MG  
2011**

MARIANA DE AZEVEDO SANTOS

AGILE UBPM FOR SCRUM: Modelo de aprimoramento do gerenciamento e desenvolvimento ágil baseado na percepção de valor do usuário

Monografia de graduação apresentada ao Departamento de Ciência da Computação da Universidade Federal de Lavras como parte das exigências do curso de Sistemas de Informação para obtenção do título de Bacharel em Sistemas de Informação.

Área de Concentração:  
Engenharia de Software

Orientador:  
Prof. Paulo Henrique de Souza Bermejo

Co-orientador:  
Prof. Marcelo Silva de Oliveira

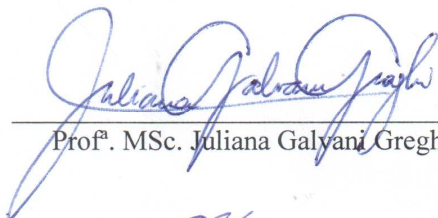
LAVRAS  
MINAS GERAIS - BRASIL  
2011

MARIANA DE AZEVEDO SANTOS

AGILE UBPM FOR SCRUM: Modelo de Aprimoramento do Gerenciamento e  
Desenvolvimento Ágil Baseado na Percepção de Valor do Usuário

Monografia de graduação apresentada ao Departamento  
de Ciência da Computação da Universidade Federal de  
Lavras como parte das exigências do curso de Sistemas  
de Informação para obtenção do título de Bacharel em  
Sistemas de Informação.

Aprovada em 16 de junho de 2011.



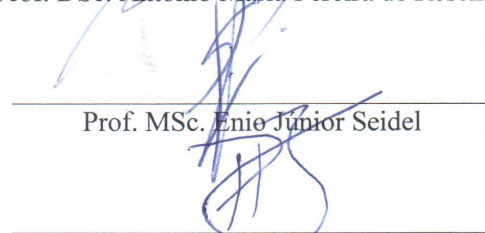
---

Prof. MSc. Juliana Galyani Greghi



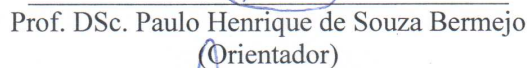
---

Prof. DSc. Antônio Maria Pereira de Resende



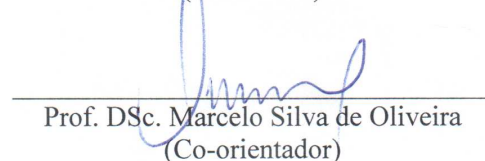
---

Prof. MSc. Enio Junior Seidel



---

Prof. DSc. Paulo Henrique de Souza Bermejo  
(Orientador)



---

Prof. DSc. Marcelo Silva de Oliveira  
(Co-orientador)

LAVRAS  
MINAS GERAIS-BRASIL

*Dedico este trabalho aos meus pais Paulo André dos Santos e  
Leila Maria de Azevedo Santos, e ao meu irmão Lucas de Azevedo Santos.*

## AGRADECIMENTOS

Ao Criador, a causa primária de todas as coisas, a quem dedico minha fé e confiança nesta minha jornada de buscas e aprendizado eterno.

Aos meus pais Paulo André dos Santos e Leila Maria de Azevedo Santos pelo amor, dedicação e paciência sempre, pelo exemplo luta e coragem. Vocês dois são a minha luz!

Ao meu irmão Lucas de Azevedo Santos, pelo companheirismo, a amizade, carinho e atenção.

À minha família, vó Anália, tias e tios queridos, primos e primas pelo apoio e carinho.

Aos amigos de tantos anos que já são parte da minha família, Carlos Tadeu, Maria, Germano, Guilherme e Giovani Siqueira, Carol, Diogo, Luiz Henrique, Susan, Talita, Lívia, Karla Monique, Felipe, Tiago, Maria José, Gilson, Michelle, Rafaela e Daniela, sempre pelo apoio, afeto e os bons momentos.

Aos amigos da UFLA, Clayton, Carla, Christiane, Alexsandra, Maísa, Juliana Villas Boas, Gabriela, Daniel Zanzini, Daniel Sannomia, Diego Bachim, Bruno Vítor, Ivayr Farah Netto, Henrique, Leandro Alonso, Pedro Guimarães, Cristiane Coelho, Yara, Giulia, Tiago Freire (Kuruma), Carlos Lino, Mailton, Jesimar, Luiz Faleiro, Júlio Farah e Sarah, pela amizade sincera e pelo crescimento que tivemos nos grupos de estudo, grupos de trabalho e é claro, nas inúmeras disciplinas que fizemos juntos.

Aos amigos do labGTI (Laboratório de Governança, Tecnologia e Inovação), Adriano, Everton, Ramon, Lucas, André, Danilo, Carla Evaristo, José Henrique, Guilherme e Danielle.

Aos professores orientadores Antonio Maria Pereira de Resende, Juliana Galvani Greggi e Paulo Henrique de Souza Bermejo, pelas oportunidades de trabalho em projetos de pesquisa, pelos quais muito aprendi e continuo aprendendo. Aos professores Marcelo Oliveira e Enio Seidel pela orientação, ajuda e apoio em Estatística durante todo esse trabalho. Foi um grande aprendizado!

E por fim a todos que contribuíram diretamente e indiretamente para a realização deste trabalho.

Obrigada!

## RESUMO

### AGILE UBPM FOR SCRUM: Modelo de Aprimoramento do Gerenciamento e Desenvolvimento Ágil Baseado na Percepção de Valor do Usuário

Com a crescente necessidade das organizações em entregar um produto de software com rapidez e qualidade, cada vez mais organizações em todo o mundo estão adotando metodologias ágeis de desenvolvimento de software, que em comparação com metodologias tradicionais, obtiveram melhores resultados quanto aos prazos, qualidade, escopo e custos. O trabalho, que ora se apresenta, objetiva a investigação e análise da percepção de valor de uso de práticas das metodologias ágeis para o desenvolvimento de sistemas de software através de um questionário direcionado a comunidade de software no mundo. Dentre os resultados encontrados destaca-se um levantamento de semelhança de práticas, análises estatísticas da percepção de valor das práticas utilizando Análise Fatorial e Análise de Cluster e a proposta de um modelo de aprimoramento para o gerenciamento e desenvolvimento ágil, chamado AGILE UBPM for *Scrum*, baseado na percepção de valor do usuário e aplicado na metodologia ágil *Scrum*.

**Palavras-chave:** Metodologias ágeis, práticas ágeis, percepção de valor, AGILE UBPM for *Scrum*.

## **ABSTRACT**

### **AGILE UBPM FOR SCRUM: Best Practice Model for Software Developing Based on User's Value Perception**

With the growing need for organizations to deliver a software product quickly and with quality, even more organizations in the worldwide are adopting agile software development, which in comparison with traditional methods, have obtained better results for the time, quality, scope and costs. The work presented here, aims to research and analysis of the value perception of the users on Agile practices for developing software systems through a questionnaire directed to the software community in the world. Among the findings highlighted a survey of similar practices, statistical analysis of the perception of value of the practices using the quantitative methods of factor analysis and cluster analysis and the proposal of a model for improving the management and agile development called AGILE UBPM for Scrum, based on the perceived value of the user and applied to the Scrum agile methodology.

**Keywords:** Agile methodologies, agile practices, perception of value, AGILE UBPM

# LISTA DE FIGURAS

Figura 2.1 – Ciclo de vida da metodologia XP .....	15
Figura 2.2 – Visão geral da metodologia Scrum .....	16
Figura 2.3 – Ciclo adaptativo da metodologia ASD .....	19
Figura 2.4 – Estrutura da metodologia FDD .....	22
Figura 2.5 – Ciclo de vida da metodologia DSDM .....	24
Figura 2.6 – Ciclo de vida da família da metodologia Crystal .....	26
Figura 2.7 – Ciclo de vida do TDD .....	29
Figura 3.1 – Desenho da pesquisa .....	36
Figura 4.1 – Semelhanças entre os conceitos das principais práticas ágeis utilizadas nas metodologias estudadas .....	43
Figura 4.2 – Estrutura do questionário.....	46
Figura 4.3 – Número respostas obtidas em cada questão de avaliação de práticas ágeis do questionário .....	47
Figura 4.4 – Países respondentes da pesquisa .....	48
Figura 4.5 – Cargo dos respondentes da pesquisa .....	49
Figura 4.6 – Característica da empresa .....	50
Figura 4.7 – Utilização de metodologias ágeis pelas empresas .....	50
Figura 4.8 – Estágio de adoção de metodologias ágeis na empresa .....	51
Figura 4.9 – Representação gráfica do fator Interação da equipe .....	57
Figura 4.10 – Representação gráfica do fator Liderança e custo .....	58
Figura 4.11 – Representação gráfica do fator Desenvolvimento .....	59
Figura 4.12 – Representação gráfica do fator Tempo de resposta .....	59
Figura 4.13 – Representação gráfica do fator Modelagem de processos .....	60
Figura 4.14 – Representação gráfica do fator Desenvolvimento solo .....	61
Figura 4.15 – Representação gráfica do fator Backlog com integração contínua .....	66
Figura 4.16 – Representação gráfica do fator Análise de requisitos .....	66
Figura 4.17 – Representação gráfica do fator Modelagem do processo de teste .....	67
Figura 4.18 – Representação gráfica do fator Prevenção de erros com casos de teste .....	67
Figura 4.19 – Representação gráfica do fator Documento de visão .....	68
Figura 4.20 – Representação gráfica do fator Equipes de teste multifuncionais .....	69
Figura 4.21 – Representação gráfica do fator Releases com tempo pré-fixado .....	75
Figura 4.22 – Representação gráfica do fator Testes e liberação contínua .....	76
Figura 4.23 – Representação gráfica do fator Equipes de liderança .....	76
Figura 4.24 – Representação gráfica do fator Histórias de usuário .....	77
Figura 4.25 – Representação gráfica do fator Documentação representativa .....	78
Figura 4.26 – Representação gráfica do fator Diagramação .....	78
Figura 4.27 – Representação gráfica do fator Desenvolvedor solo .....	79
Figura 4.28 – Representação gráfica do fator Desenvolvimento baseado em casos de teste .....	84
Figura 4.29 – Representação gráfica do fator Diagramas de erros .....	85



Figura 4.30 – Representação gráfica do fator Equipes pequenas lideradas pelo facilitador .....	85
Figura 4.31 – Representação gráfica do fator Funcionalidades definidas pelo cliente .....	86
Figura 4.32 - Representação gráfica do fator Reuniões em pé .....	86
Figura 4.33 – Representação gráfica do fator Releases frequentes .....	87
Figura 4.34 – Representação gráfica do fator Programador-líder .....	87
Figura 4.35 – Dendograma da análise de cluster para o Fator 1 – Interação da equipe, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	93
Figura 4.36 – Dendograma da análise de cluster para o Fator 2 – Liderança e custos, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	94
Figura 4.37 – Dendograma da análise de cluster para o Fator 3 – Desenvolvimento, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	95
Figura 4.38 – Dendograma da análise de cluster para o Fator 4 –Tempo de resposta, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	96
Figura 4.39 – Dendograma da análise de cluster para o Fator 5 – Modelagem de processos, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	97
Figura 4.40 – Dendograma da análise de cluster para o Fator 6 – Desenvolvimento solo, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	98
Figura 4.41 – Dendograma da análise de cluster para o Fator 1 – Backlog de integração contínua, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	99
Figura 4.42 – Dendograma da análise de cluster para o Fator 2 – Análise de requisitos, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	100
Figura 4.43 – Dendograma da análise de cluster para o Fator 3 – Modelagem do processo de teste, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	101
Figura 4.44 – Dendograma da análise de cluster para o Fator 4 – Prevenção de erros com casos de teste, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	102
Figura 4.45 – Dendograma da análise de cluster para o Fator 5 – Documento de visão, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	103
Figura 4.46 – Dendograma da análise de cluster para o Fator 6 – Equipes de teste multifuncionais, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	104
Figura 4.47 – Dendograma da análise de cluster para o Fator 1 – Release com tempo pré-fixado, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	105
Figura 4.48 – Dendograma da análise de cluster para o Fator 2 – Testes e liberação contínua, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	106
Figura 4.49 – Dendograma da análise de cluster para o Fator 3 – Equipes de liderança, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	107

Figura 4.50 – Dendograma da análise de cluster para o Fator 4 – Histórias de usuário, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	108
Figura 4.51 – Dendograma da análise de cluster para o Fator 5 – Documentação representativa, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	110
Figura 4.52 – Dendograma da análise de cluster para o Fator 6 – Diagramação, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	111
Figura 4.53 – Dendograma da análise de cluster para o Fator 7 – Desenvolvedor solo, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	112
Figura 4.54 – Dendograma da análise de cluster para o Fator 1 – Desenvolvimento baseado em casos de teste, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	113
Figura 4.55 – Dendograma da análise de cluster para o Fator 2 – Diagrama de erros, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	114
Figura 4.56 – Dendograma da análise de cluster para o Fator 3 – Equipes pequenas lideradas pelo facilitador, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	115
Figura 4.57 – Dendograma da análise de cluster para o Fator 4 – Funcionalidades definidas pelo cliente, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança .....	117
Figura 4.58 – Dendograma da análise de cluster para o Fator 5 – Reuniões em pé, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	119
Figura 4.59 – Dendograma da análise de cluster para o Fator 6 – Releases frequentes, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	121
Figura 4.60 – Dendograma da análise de cluster para o Fator 7 – Programador-líder, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.....	123
Figura 4.61 – Número de respondentes e as metodologias por eles utilizadas.....	128
Figura 4.62 – Visão geral do ciclo de vida do <i>Agile UBPM for Scrum</i> .....	130

# LISTA DE TABELAS

Quadro 2.1 - Descrição das práticas executadas durante a implementação da metodologia XP .....	14
Quadro 2.2 - Descrição dos papéis delegados aos funcionários de cada projeto implementando o Scrum .....	17
Quadro 2.3 - Descrição das práticas executadas durante a implementação da metodologia Scrum .....	17
Quadro 2.4 - Descrição das práticas executadas durante a implementação da metodologia ASD .....	20
Quadro 2.5 - Descrição das práticas executadas durante a implementação da metodologia FDD .....	22
Quadro 2.6 - Descrição das práticas executadas durante a implementação da metodologia DSDM ....	24
Quadro 2.7 - Descrição das práticas executadas durante a implementação da metodologias da família Crystal .....	27
Quadro 2.8 - Descrição das práticas executadas durante a implementação da metodologia TDD .....	29
Quadro 4.1 - Práticas avaliadas pelo critério custo, qualidade, prazo e escopo .....	52
Quadro 4.2 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem representativa do critério custo .....	54
Quadro 4.3 - Porcentagem da variância explicada para a análise representativa para o critério Custo. 55	
Quadro 4.4 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem pouco representativa .....	61
Quadro 4.5 - Porcentagem da variância explicada para a análise pouco representativa para o critério Custo .....	62
Quadro 4.6 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem representativa do critério qualidade .....	64
Quadro 4.7 - Porcentagem da variância explicada para a análise representativa para o critério Qualidade .....	64
Quadro 4.8 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem pouco representativa do critério qualidade .....	69
Quadro 4.9 - Porcentagem da variância explicada para a análise pouco representativa para o critério Qualidade .....	70
Quadro 4.10 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem representativa do critério Prazo .....	72
Quadro 4.11 - Porcentagem da variância explicada para a análise representativa para o critério Prazo 73	
Quadro 4.12 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem pouco representativa do critério Prazo .....	79
Quadro 4.13 - Porcentagem da variância explicada para a análise pouco representativa para o critério Prazo .....	80
Quadro 4.14 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem representativa do critério Escopo .....	82
Quadro 4.15 - Porcentagem da variância explicada para a análise representativa para o critério Escopo .....	83
Quadro 4.16 - Coeficiente KMO da análise fatorial e teste de esfericidade de Bartlett para amostragem pouco representativa do critério Escopo.....	88
Quadro 4.17 - Porcentagem da variância explicada para a análise pouco representativa para o critério Escopo .....	88
Tabela 4.1 - Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério custo .....	56

Tabela 4.2 - Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério custo .....	62
Tabela 4.3 - Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério qualidade .....	65
Tabela 4.4 - Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério qualidade .....	69
Tabela 4.5 - Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério prazo .....	74
Tabela 4.6 - Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério prazo .....	80
Tabela 4.7 - Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério escopo .....	83
Tabela 4.8 - Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério escopo .....	89

## SUMÁRIO

LISTA DE FIGURAS .....	i
LISTA DE TABELAS.....	iv
1 INTRODUÇÃO.....	1
1.1 Apresentação.....	1
1.2 Problemas de pesquisa.....	2
1.3 Objetivos.....	3
1.3.1 Objetivo Geral.....	3
1.3.2 Objetivos Específicos.....	3
1.4 Estrutura do Trabalho.....	4
2 REFERENCIAL TEÓRICO.....	6
2.1 Engenharia de Software.....	6
2.2 Metodologias ágeis.....	8
2.2.1 Extreme Programming (XP).....	12
2.2.2 Scrum.....	15
2.2.3 Adaptive Software Development (ASD).....	18
2.2.4 Feature Driven Development (FDD).....	20
2.2.5 Dynamic Software Development Method (DSDM) .....	23
2.2.6 Crystal methodologies .....	25
2.2.7 Test Driven Development (TDD).....	27
2.3 Valor e percepção de uso .....	30
2.4. Trabalhos relacionados.....	31
2.5. Considerações finais.....	33
3 METODOLOGIA.....	35
3.1 Método de pesquisa.....	35
3.2 Desenho da pesquisa.....	36
3.3 Coleta de dados para a pesquisa.....	37
3.4 Abrangência da pesquisa.....	38
3.5 Análise de dados da pesquisa.....	39
4 RESULTADOS .....	42
4.1 Levantamento e análise de semelhança das práticas.....	42
4.2 Instrumento para avaliação da percepção de valor em metodologias ágeis.....	44
4.3 Análise demográfica da amostra.....	48
4.4 Análises estatísticas.....	51
4.4.1 Análise Fatorial.....	54
4.4.1.1 Custo.....	54
4.4.1.2 Qualidade.....	63
4.4.1.3 Prazo.....	72

4.4.1.4 Escopo.....	82
4.4.2 Análise de Cluster.....	91
4.4.2.1 Custo.....	92
4.4.2.2 Qualidade.....	99
4.4.2.3 Prazo.....	105
4.4.2.4 Escopo.....	113
4.4.3 Discussão.....	124
4.5 Proposta de modelo de aprimoramento do gerenciamento e desenvolvimento ágil baseado na percepção de valor do usuário (User-based Perception Model – UBPM).....	127
5 CONCLUSÃO .....	132
5.1 Trabalhos futuros.....	133
REFERÊNCIAS BIBLIOGRÁFICAS.....	134
ANEXO A: QUESTIONÁRIO APLICADO EM PORTUGUÊS.....	138
ANEXO B: CARTA DE APRESENTAÇÃO DA PESQUISA.....	144

# 1 INTRODUÇÃO

## 1.1. Apresentação

A complexidade e o caráter dinâmico do processo de desenvolvimento de software tem tornado metodologias tradicionais de desenvolvimento de software dispendiosas e, muitas vezes, inadequadas para organizações que necessitam reagir com agilidade diante às mudanças (SCHWABER, 2004).

Essa necessidade e a questão de como o desenvolvimento de software deve se organizar para entregar um produto com rapidez e qualidade tem se tornado uma das principais discussões na área de engenharia de software (DYBÅ & DINGSØYR, 2008).

Nesse cenário, surgiram as metodologias ágeis, que enfatizam princípios associados à comunicação, objetividade, maior foco no desenvolvimento e interação com o cliente, de forma a proporcionar agilidade e eficiência no desenvolvimento (BECK *et al.*, 2011).

As metodologias ágeis são formadas por elementos individuais chamados de práticas, que incluem o uso de controle de versão, estabelecem normas de codificação, dando demonstrações semanais para todos os interessados (SHORE & WARDEN, 2007).

Nas práticas ágeis existentes, conceitos como relacionamento e interação da equipe, disposições do ambiente de trabalho, e outros procedimentos, estão presentes, aumentando o espírito de equipe (ABRAHAMSSON *et al.*, 2002).

Segundo Abrahamsson *et al.* (2002), cada vez mais organizações em todo o mundo estão adotando metodologias ágeis de desenvolvimento de software. Esse crescimento pode ser apoiado pelo resultado da pesquisa de Charette (2001), onde projetos utilizando métodos ágeis, em comparação com metodologias tradicionais, obtiveram melhores resultados quanto aos prazos, qualidade e custos.

Existem diversos exemplos de metodologias ágeis propostas na literatura e em geral elas se baseiam no conjunto de princípios estabelecidos no Manifesto Ágil (PFLEEGER & ATLEE, 2006). Mesmo adotando princípios comuns, as metodologias ágeis possuem

algumas divergências na utilização e conceituação de práticas de gerenciamento e desenvolvimento de software.

## 1.2. Problemas de Pesquisa

Buscando compreender os fatores de sucesso e insucesso de projetos de software, o *Standish Group* realizou uma recente pesquisa com a comunidade de TI no mundo. Segundo o Standish Group (2009), 32% dos projetos de software são bem sucedidos, sendo entregues no prazo, cumprindo o orçamento e de acordo com especificações prévias. Em contraste, aproximadamente 44% estão atrasados, com orçamento ultrapassado, e/ou com menos funções e funcionalidades requeridas.

Comparando os dados supracitados com pesquisa, realizada também pelo Standish Group em 2006, houve uma queda de 3% das taxas de sucesso no projeto. Ou seja, entre 2006 e 2009, não houve melhorias nas taxas de sucesso em projetos de software. Pelo contrário, notou-se que essas taxas sofreram leve queda.

De acordo com Koscianski e Soares (2007), uma abordagem adequada para evitar muitos dos insucessos nos projetos seria o desenvolvimento incremental, princípio conceitual básico das metodologias ágeis.

Considerando a crescente adoção de metodologias ágeis em organizações no mundo, pesquisas sobre o rendimento de projetos ágeis sob critérios de desempenho como custo, qualidade, prazo e escopo tem apresentado resultados benéficos de sua aplicação.

A pesquisa de Svensson e Höst (2005) retrata como a adoção de uma metodologia ágil em uma empresa contribuiu para a simplificação da comunicação entre os membros da organização e redução do custo excedente para 25%. Sillitti *et al.* (2005), afirmam que gerentes de projeto constataram que empresas que utilizam métodos ágeis tem escopo mais flexível do que as de documentação rígida e parecem ter uma relação mais satisfatória com o cliente.

Segundo Melkin e Maurer (2005), na visão do desenvolvedor, o trabalho em equipes ágeis promove o aumento qualitativo no que se diz respeito ao desenvolvimento de



habilidades profissionais, tais como comunicação, compromisso, cooperação e adaptabilidade.

Na perspectiva do cliente, o estudo de Ilieva *et al.* (2004) afirma que a introdução de uma metodologia ágil na processo de desenvolvimento da empresa estudada obteve aceitação e foi muito elogiada.

Dybå e Dingsøyr (2008) apontam que diversos trabalhos sobre metodologias ágeis já foram publicados, no entanto pouco se sabe sobre quais as principais práticas adotadas e que efeitos geram em projetos de software.

Diante da demanda por melhoria nas taxas de sucesso no desenvolvimento de software e dos crescentes relatos de melhorias alcançadas na adoção das metodologias ágeis de desenvolvimento, este trabalho propõe responder as seguintes questões de pesquisa:

*Q1: Quais as principais metodologias e práticas ágeis utilizadas para o desenvolvimento de software?*

*Q2: Dentre as principais práticas ágeis, quais possuem melhor avaliação de percepção de valor sobre os critérios de desempenho de projeto: custo, qualidade, prazo e escopo?*

## **1.3. Objetivos**

Os objetivos estão organizados na forma de objetivos geral e específicos para alcance e cumprimento das metas propostas durante a realização do projeto.

### **1.3.1 Objetivo Geral**

Investigar e analisar a percepção de valor de uso de práticas das metodologias ágeis para o desenvolvimento de sistemas de software.

### **1.3.2 Objetivos Específicos**

(i). Identificar e avaliar a semelhança das principais práticas de metodologias ágeis propostas para o desenvolvimento de produtos de software;

(ii). Analisar a percepção de valor do uso das principais práticas de metodologias ágeis no desenvolvimento de produtos de software;

(iii). Propor um modelo de boas práticas de modelo de aprimoramento do gerenciamento e desenvolvimento ágil baseado na percepção de valor do usuário.

## 1.4. Estrutura do trabalho

O presente trabalho divide-se em cinco capítulos e um apêndice, os quais visam à abordagem de questões relacionadas à proposição modelo de boas práticas para uso e aplicação de práticas de metodologias ágeis para o desenvolvimento de software de acordo com o perfil organizacional e características peculiares do projeto de desenvolvimento. Os capítulos seguintes estão estruturados como apresentado abaixo.

**Capítulo 1:** Introdução – o presente capítulo apresenta a introdução do trabalho, incluindo a apresentação, a contextualização, o objetivo geral e os específicos, a definição do problema e a estrutura do trabalho.

**Capítulo 2:** Referencial teórico – Capítulo em que é conceituado o assunto sobre metodologias ágeis de desenvolvimento, abordando os seus fundamentos, princípios além de relacionar os principais métodos ágeis, incluindo um comparativo entre eles. Também apresenta uma contextualização sobre valor e percepção de uso que consistem em variáveis importantes para a análise dos modelos propostos neste trabalho.

**Capítulo 3:** Metodologia - Neste capítulo é explicado o método de pesquisa, por meio da apresentação dos seus fundamentos de natureza e desenho da pesquisa, abrangência e técnicas e dos modelos estatísticos utilizados para a análise dos resultados.

**Capítulo 4:** Resultados: Capítulo que apresenta os resultados do trabalho: (i) levantamento de semelhanças das principais práticas ágeis; (ii) o instrumento de avaliação da percepção de valor de práticas ágeis; (iii) análise estatística das principais práticas de metodologias ágeis no desenvolvimento de produtos de software; (iv) proposta do modelo de aprimoramento do gerenciamento e desenvolvimento ágil de software baseado na percepção do usuário – AGILE UBPM for *Scrum* (*User-based Perception Model*).

**Capítulo 5:** Conclusão – Neste capítulo são apresentadas: (i) a discussão sobre os resultados obtidos na monografia, utilizando-se para isso uma síntese e a comparação do modelo proposto em relação aos estudos mais recentes e trabalhos relacionados; (ii) as conclusões sobre a monografia e (iii) as recomendações para trabalhos futuros.

## 2 REFERENCIAL TEÓRICO

Uma metodologia de desenvolvimento de software é um conjunto de atividades e práticas que auxiliam no desenvolvimento de um produto de software (KOSCIANSKI & SOARES, 2007).

As metodologias com especificações completas de requisitos, projeto, construção e teste de sistemas, não mais se adaptam ao desenvolvimento rápido, voltado para situações de constantes mudanças (SOMMERVILLE, 2008). Nesse cenário, surgiram as metodologias ágeis de desenvolvimento de software.

Mesmo com um histórico recente de aplicações em organizações, as metodologias ágeis já demonstram resultados efetivos (KOSCIANSKI & SOARES, 2007). Segundo Charette (2001) projetos utilizando metodologias ágeis obtiveram melhores resultados quanto aos prazos, qualidade e custos.

Para revisão teórica do assunto de metodologias ágeis de desenvolvimento de software, este capítulo foi estruturado da seguinte forma:

- 2.1) revisão teórica sobre a Engenharia de Software;
- 2.2) definição e contextualização das metodologias ágeis abordadas;
- 2.3) definição e contextualização sobre valor e percepção de uso de software;
- 2.4) revisão sobre trabalhos relacionados a proposta;

### 2.1. Engenharia de Software

Ao contrário do que muitas pessoas imaginam, o conceito de software abrange não só programas que executam instruções em computadores, mas sim todo o conteúdo que o caracteriza, como dados de documentação e configuração (SOMMERVILLE, 2008).

O software é um produto desenvolvido por profissionais da computação da área de Engenharia de Software, ramo da engenharia relacionada ao gerenciamento e processos de desenvolvimento de softwares computacionais a custos baixos e de alta qualidade (PRESSMAN, 2006).

O conceito de Engenharia de Software foi inicialmente proposto em 1968, na tentativa de solucionar a “crise do software”. O objetivo era propor soluções mais sistematizadas e controladas para um conjunto de problemas enfrentados no desenvolvimento de sistemas de software, baseados nos conhecimentos de práticas e métodos da computação (PFLEEGER & ATLEE, 2006).

Embora com a existência de várias metodologias diferentes, em suma, todas elas compartilham características comuns (SOMMERVILLE, 2008):

- **Especificação do Software:** definição da funcionalidade do software e suas restrições;
- **Projeto e implementação:** o produto de software deve atender à especificação;
- **Validação do software:** validação do software para a garantia de que ele faça o que o cliente deseja;
- **Evolução do software:** evolução do software para atender as necessidades do cliente.

Devido ao crescimento da importância do desenvolvimento de software nos últimos tempos, os profissionais e pesquisadores da Engenharia de Software começaram a perceber uma mudança de cenário: o desenvolvimento de software nos dias de hoje, passou a ser um grande esforço coletivo, onde a comunidade de software tem tentado criar esforços para que este desenvolvimento torne-se cada vez mais fácil, rápido, de alta qualidade e menos dispendioso (FUGGETTA, 2000; PRESSMAN, 2008).

Nesse sentido, metodologias com especificações completas de requisitos, projeto, construção e teste de sistemas, não estão adaptadas ao desenvolvimento rápido, voltado para esse cenário com situações de constantes mudanças (SOMMERVILLE, 2008).

Diante de um cenário mais competitivo e dinâmico, o desenvolvimento ágil tornou-se popular, por manter a equipe de desenvolvimento produtiva, ofertando valor e diminuindo custos. (SHORE & WARDEN, 2007).

## 2.2. Metodologias ágeis

As metodologias ágeis são processos que suportam e implementam a filosofia de dinamismo e agilidade no desenvolvimento de software (SHORE & WARDEN, 2007).

Constituem-se em um conjunto de práticas de desenvolvimento e gerenciamento de software que foram consolidadas, em 2001, por 17 profissionais experientes especialistas em processos de desenvolvimento de software, estabelecendo valores e princípios comuns (ÅGERFALK & FITZGERALD, 2006; KOSCIANSKI & SOARES, 2007).

Com base nos resultados, foi criada a Aliança Ágil e o estabelecimento do Manifesto Ágil. Os valores chave propostos no “Manifesto Ágil” são (BECK *et al.*, 2010):

- Indivíduos e interações ao invés de processos e ferramentas.
- Software executável ao invés de documentação.
- Colaboração do cliente ao invés de negociação de contratos.
- Respostas rápidas a mudanças ao invés de seguir planos.

A idéia do “Manifesto Ágil” é redefinir as prioridades em um projeto de software. Ainda há processos e ferramentas, documentação, negociação de contratos, porém de forma reduzida e mais objetiva. Em resumo, esse paradigma permite a minimização de riscos no desenvolvimento e gerenciamento de projetos de software.

Complementando os valores descritos no Manifesto Ágil, foram criados os 12 princípios que descrevem e explicam a filosofia ágil (AGILE ALLIANCE, 2011)<sup>1</sup>:

*Princípio 1: “Nossa maior prioridade é satisfazer o cliente através da entrega rápida e contínua de software de valor.”*

---

<sup>1</sup> Texto original em inglês, traduzido e comentado

Na visão ágil, o cliente deve ser o principal foco das equipes ao desenvolver um software e entregá-lo às partes interessadas de maneira rápida e eficiente.

*Princípio 2: “Mudanças de requisitos são bem-vindas, mesmo muito depois do início do desenvolvimento. Processos ágeis esperam que a mudança traga uma vantagem competitiva para cliente.”*

As mudanças devem ser recebidas, mas somente com o intuito de agregar valor e vantagens competitivas ao negócio do cliente. As mudanças aleatórias, baseadas no estado de humor dos envolvidos, devem ser rejeitadas, uma vez que não colaboram em nada para a evolução do projeto.

*Princípio 3: “Entregas frequentes de software funcional, de algumas semanas a alguns meses, com preferência para um período curto.”*

O tempo deve ser flexível às necessidades da equipe e do cliente. A equipe deve fazer tudo que for possível para tentar entregar um software ou parte dele que funciona, com qualidade, o mais breve possível.

*Princípio 4: “Pessoas de negócio e os desenvolvedores devem trabalhar juntos diariamente durante o projeto.”*

Os *stakeholders* e a equipe devem se comunicar frequentemente, mesmo que isso signifique que o *stakeholder* esteja presente na empresa acompanhando de perto a equipe de desenvolvimento, ou a equipe se estabeleça na empresa contratante.

*Princípio 5: “Construa projetos em torno de indivíduos motivados. Ofereça o ambiente e suporte que precisam, e confie neles para realizar o trabalho.”*

As equipes devem ter características auto-gerenciáveis e transmitir confiança ao líder e gerente do projeto. O mesmo deve se esperar do gerente e do líder, que tenham confiança na sua equipe. Quanto ao ambiente de trabalho este deve ser organizado e estimulante, para que a equipe possa crescer e manter a qualidade do trabalho.

*Princípio 6: “O método mais eficiente e efetivo de transmitir uma informação para uma equipe de desenvolvimento é conversando cara-a-cara.”*

A melhor maneira de entender um negócio é discutindo junto ao cliente. Uma vez que estas reuniões presenciais acontecem com maior frequência fica mais fácil entender o que o cliente espera do negócio, evitando inadequações e implementações desnecessárias.

*Princípio 7: “Software que funciona é a medida primária de progresso.”*

Entregar um software sem funcionamento é o mesmo que não entregar. O primordial é que a cada funcionalidade validada, esta seja implementada e entregue para o cliente. Se a equipe não conseguiu entregar valor junto à funcionalidade, significa que estas não foram bem definidas e devem ser revisadas.

*Princípio 8: “Processos ágeis promovem desenvolvimento sustentado. Os patrocinadores, desenvolvedores e usuários devem ser capazes de manter um ritmo constante indefinidamente.”*



Se existe comunicação, um ambiente organizado, as prioridades do negócio e do projeto estão bem definidas e os stakeholders estão envolvidos, o ritmo de entregas de novas funcionalidades é mantido, e com melhoria contínua.

*Princípio 9: “Atenção constante a excelência técnica e ao bom design aumenta a agilidade.”*

A qualidade, o bom design, o seguimento de padrões de projeto são pontos importantes para manter a excelência técnica da equipe. Se estes princípios são seguidos, é possível se ter agilidade, entregar software sem sacrifícios de padrões de qualidade.

*Princípio 10: “Simplicidade – a arte de maximizar a quantidade de trabalho não realizado – é essencial.”*

A equipe de simplificar seu trabalho, procurando soluções iniciais simples para os requisitos do cliente, e, em seguida, refatorar e aprimorar as soluções. A idéia é não se preocupar em excesso com uma arquitetura muito complexa, afetando na entrega do software, que se atrasada, pode não agregar valor algum ao cliente.

*Princípio 11: “As melhores arquiteturas, requisitos e designs emergem de equipes auto-organizadas.”*

Equipes auto-organizáveis são bem vistas na filosofia ágil porque possuem habilidade de se adaptar dinamicamente às mudanças de direção do negócio propostos pelos *stakeholders*. Assim, as equipes, procuram reinventar e reestruturar o negócio com criatividade para atender a demanda do cliente.

*Princípio 12: “Em intervalos regulares de tempo, a equipe reflete sobre como se tornar mais efetiva, então sintoniza e ajusta seu comportamento de acordo.”*

O processo de retrospectiva é importante para monitorar o projeto seja ao final de todo mini-ciclo de trabalho, ou ao final do projeto por completo. As retrospectivas servem para identificar processos falhos e desnecessários, bem como traçar planos de melhoria futura, propostas pela equipe.

Seguindo os 12 princípios estabelecidos no Manifesto Ágil, existem diversos exemplos de metodologias ágeis propostas na literatura, possuindo seu conjunto de práticas e papéis bem definidos (PFLEEGER & ATLEE, 2006).

Dentre as metodologias ágeis iniciais, criadas anteriormente ao Manifesto Ágil destacam-se o *Extreme programming* para desenvolvimento de software, *Scrum* para gerenciamento de projetos, *Adaptive Software Development*, *Feature Driven Development*, *Dynamic Systems Development Method* e *Crystal Methodologies*. Posteriormente, o *Test Driven Development* foi incorporado à lista de metodologias disponíveis.

### **2.2.1 Extreme Programming (XP)**

Criada por Kent Beck e Martin Fowler em 1997 e apresentada a comunidade de software na OOSPLA *Conference (Object-Oriented Programming, Systems, Languages & Applications Conference)* de 2000, o *Extreme Programming (XP)* é uma metodologia voltada para atender às necessidades específicas de desenvolvimento de software conduzido por equipes pequenas diante de requisitos vagos e da evolução das necessidades dos clientes (BECK & FOWLER, 2000).

O XP consiste na estratégia de acompanhamento constante e realização de vários pequenos ajustes durante o desenvolvimento de software. Essa estratégia se resume em cinco valores básicos: comunicação, *feedback* rápido, presumir simplicidade, mudanças incrementais e trabalho de qualidade (PFLEEGER & ATLEE, 2006).

A partir do conjunto de valores supracitados, a metodologia recomenda princípios de bom senso e práticas de desenvolvimento a níveis extremos baseadas nas experiências do autor (BECK, 1999). As práticas em conjunto formam a base do desenvolvimento XP, onde o ponto fraco de uma prática é compensado pelos pontos fortes de outras (BECK, 1999).

A aplicação das práticas do XP, detalhadas no Quadro 2.1, obedece duas classes: as práticas primárias, que são identificadas pela equipe como proposta de melhoria e podem ser utilizadas a qualquer momento no andamento do projeto, como times multidisciplinares, histórias, programação em pares, integração contínua, desenvolvimento baseado em testes, *design* incremental, *build* de 10 minutos; e, as práticas secundárias, que são complementares às primárias, mas sua aplicação depende do amadurecimento da equipe para com as sugestões iniciais.

Práticas	Descrição
Programação em pares	Dois desenvolvedores trabalham juntos no código, sendo um responsável por pensar nos detalhes do código e a outra para garantir o seguimento das normas de codificação necessárias.
Time multidisciplinar	Equipe de trabalho com colaboradores de diferentes áreas e funções distintas.
Pequenos releases ( <i>Potentially Shippable</i> )	A liberação de pequenas versões funcionais do projeto.
Integração contínua	Prática de desenvolvimento de software onde os membros de um time integram seu trabalho frequentemente.
Desenvolvimento baseado em testes primários	Prática que defende a criação de testes unitários primeiro e depois criação o código para que os testes funcionem.
Triagem de erros	Prática de auxílio na garantia de qualidade e programadores acompanhar bugs de software informados. Quando um erro é encontrado, testes são criados prevenindo sua repetição.
Jogos de planejamento	Reunião onde os desenvolvedores e o cliente reúnem-se para priorizar as funcionalidades.
Cartões CRC	É um produto de trabalho do projeto onde são agrupadas as classes/funcionalidades mais relevantes

	para o incremento de software da iteração.
Refatoração	Processo de aprimoramento do código existente, sem mudar seu comportamento externo.
Propriedade coletiva do código	Rotatividade do código entre os programadores.
Build de 10 minutos	Conversão do código em artefato utilizável com testes automatizado com tempo de execução de 10 minutos.
Histórias ( <i>User story</i> )	Atividade de planejamento em que são criados conjuntos de histórias e descrevem características e funcionalidades.
Testes unitários	Teste com foco no componente individual do software, abrangendo interface, estrutura de dados e a funcionalidade do componente.
<i>Spike solutions</i>	Protótipo operacional de uma parte do projeto considerada difícil, com o objetivo de diminuir riscos na validação de histórias.
Projeto <i>Velocity</i>	É uma forma de cálculo da velocidade do projeto. Representa a quantidade de histórias dos clientes implementadas.
Área de Trabalho Informativa	A área de trabalho deve proporcionar a boa comunicação entre o time, visibilidade do projeto e facilitar a utilização de práticas ágeis.
Ciclo semanal	Iteração com entrega de incremento semanal
Ciclo trimestral	Iteração com entrega de incremento trimestral
Cliente presente	O cliente deve estar presente e participar de todas as etapas do processo de desenvolvimento
Design Incremental	O design da aplicação surge de forma iterativa e incremental em projetos. Cria-se a solução mais simples possível, suficiente para implementar as funcionalidades de cada iteração.

Quadro 2.1 - Descrição das práticas executadas durante a implementação da metodologia XP

A aplicação dos valores, princípios e práticas da metodologia XP durante o desenvolvimento de software podem ser retratadas na Figura 2.1.

Após o desenho da metáfora inicial do sistema e do protótipo operacional de partes custosas do projeto, a equipe planeja os requisitos a serem implementados na versão, oriundas das histórias do usuário e dos Cartões de CRC com as funcionalidades planejadas e validadas nos testes de cenário. Durante a iteração, os desenvolvedores utilizam práticas de programação em pares e integração contínua para implementar o código, que testado continuamente até que receba a aprovação do cliente. Ao final deste ciclo, uma pequena e potencial versão funcional do software é entregue.

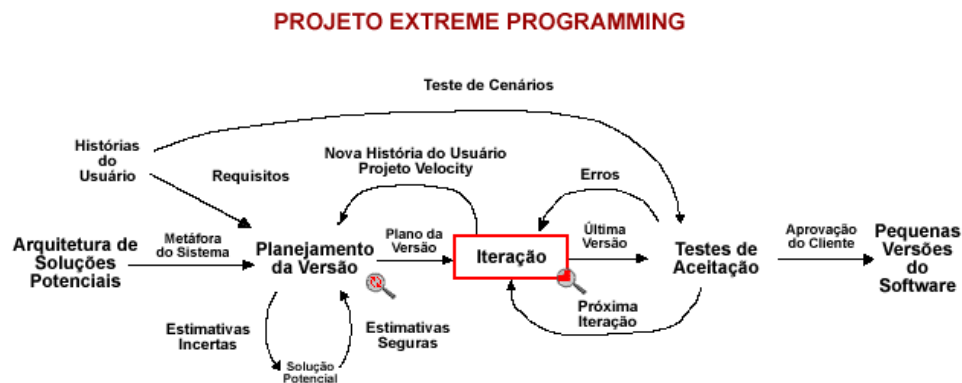


Figura 2.1 – Ciclo de vida da metodologia XP. Fonte: Adaptado de *The Rules of Extreme Programming* (2011)

## 2.2.2 Scrum

*Scrum* é uma metodologia ágil interativa e incremental de desenvolvimento e gerenciamento de projetos de software. A princípio, ela foi idealizada como um modelo de gerenciamento de projetos em indústrias automobilísticas e de produtos de consumo (TAKEUCHI & NONAKA, 1986).

O nome *Scrum* deriva-se de uma estratégia usada no rúgbi para o reinício do jogo, com características similares ao método de desenvolvimento proposto. Em 1993, a metodologia foi concebida e implementada por Jeff Sutherland, John Scumniotales e Jeff McKenna, sendo formalmente definida e documentada em 1995 por Ken Schwaber e Jeff Sutherland.

Segundo Schwaber e Beedle (2002), o *Scrum* tem como objetivo gerenciar e controlar processos de desenvolvimento de software, focado em pessoas e que seja indicada para ambientes com frequente mudança de requisitos.

Segundo Santos, Gregghi & Bermejo (2010), a metodologia se define em quatro papéis, conhecidos como *Product Owner*, *Scrum Master*, *Team* e o cliente; quatro cerimônias, o *Daily Meeting* ou *Daily Scrum*, *Sprint Review*, *Sprint Planning Meeting*, *Sprint Retrospective*; e, três produtos, *Product Backlog*, *Sprint Backlog* e o *Burndown Chart*.

O funcionamento do *Scrum* é estruturado em ciclos chamados *Sprints*, que são iterações de trabalho que duram de duas a quatro semanas. A metodologia estabelece também um conjunto de práticas e regras que devem ser cumpridas pela equipe, em seus respectivos papéis, para que o projeto seja bem sucedido (SANTOS, GREGHI & BERMEJO, 2010).

Na Figura 2.2, pode-se visualizar o funcionamento geral do método. Em primeiro lugar, o *Product Owner* e o cliente definem o *Product Backlog*. A partir dele, são determinados quais são os requisitos que devem ser priorizados no *Sprint*, sendo assim chamados de *Sprint Backlog*. Durante os *Sprints* são realizados os *Daily Meetings*, reuniões diárias onde o time compartilha uns com os outros o andamento do projeto. Ao final do *Sprint*, o *Scrum Team* entrega um incremento funcional do projeto ao cliente que junto ao *Product Owner*, redefine os requisitos, a priori, de acordo com o interesse do cliente.

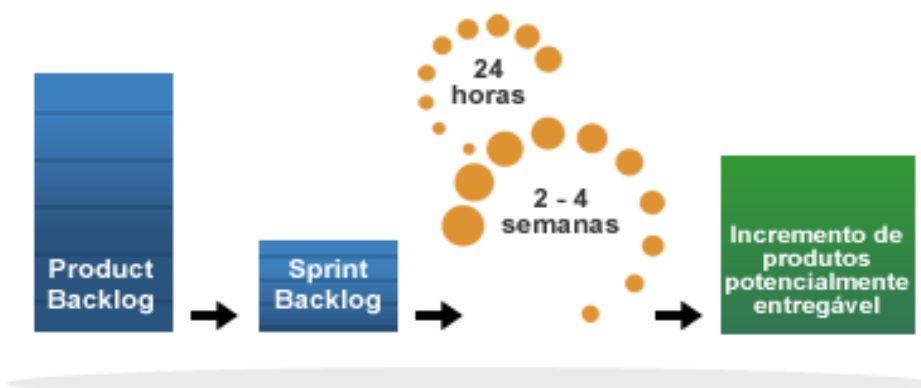


Figura 2.2 - Visão geral da metodologia Scrum. Fonte: Adaptado de SCRUMALLIANCE (2010)

As práticas e os papéis utilizados durante o ciclo do Scrum se encontram detalhadas no Quadro 2.2 e no Quadro 2.3.

Papéis	Descrição
<i>Product Owner</i>	É o moderador entre os interesses do time de desenvolvimento e do cliente. Sua responsabilidade é manter a equipe funcional e produtiva.
<i>Scrum Master</i>	Representante do cliente no projeto. Suas responsabilidades são definir funcionalidades de acordo com o valor de mercado, planejar e fazer a lista de prioridades.
Time	É o time responsável pelo desenvolvimento do projeto. Ele é multidisciplinar e é composto por um grupo de cinco a nove integrantes.
Cliente	Participa das tarefas relacionadas à implementação da lista de funcionalidades.

Quadro 2.2 – Descrição dos papéis delegados aos funcionários de cada projeto implementando o *Scrum*

Práticas	Descrição
<i>Sprint Planning</i>	Reunião em que o <i>Product Owner</i> planeja e faz a lista de prioridades que deverão ser cumpridas no projeto por completo.
<i>Daily Scrum</i>	Reunião diária onde cada membro do time responde o que já fez, o que pretende fazer e se há algum impedimento para a conclusão de sua tarefa.
<i>Sprint Review</i>	Reunião de balanço de tudo o que foi feito no <i>Sprint</i> .
<i>Sprint Retrospective</i>	Reunião de retrospectiva da <i>Sprint</i> onde são avaliados aspectos como o trabalho em equipe, os pontos positivos, negativos e como desenvolver estratégias de crescimento.
<i>Product backlog</i>	Lista de prioridades feita logo no início do projeto, com o objetivo de listar o que deve ser entregue ao cliente.

<i>Sprint Backlog</i>	Produto oriundo do <i>Sprint Planning Meeting</i> . É uma lista de tarefas específicas a serem desenvolvidas durante o <i>Sprint</i> .
<i>Burndown Chart</i>	Gráfico que estima o tempo gasto no andamento do trabalho dentro do <i>Sprint</i> . Ele é monitorado pelo time.
<i>Potentially Shippable</i>	Incremento de produtos potencialmente entregável.

Quadro 2.3 – Descrição das práticas executadas durante a implementação da metodologia Scrum

### 2.2.3 Adaptive Software Development (ASD)

Criada por Jim Highsmith em 1992, derivada de um processo iterativo baseado no processo RAD (*Rapid Application Development*) desenvolvido para auxiliar na comercialização de uma ferramenta para mainframe (HIGHSMITH, 2002).

É iterativo e incremental, focada em desenvolvimento de software e comumente aplicada em sistemas grandes e complexos (PRESSMAN, 2008). As práticas do ASD são direcionadas por uma crença na filosofia da adaptação e aceitação de mudanças de forma contínua.

O ASD possui seis princípios básicos que são aplicados em seu ciclo de vida (HIGHSMITH, 2002; PFLEEGER & ATLEE, 2006):

- **Focado em missão:** a missão é uma diretriz, que estabelece o destino do projeto, mas não prescrevendo o caminho a ser traçado.
- **Desenvolvimento baseado em componentes:** as funcionalidades são vistas como o ponto crucial de valor do cliente, de modo que o projeto é organizado em torno da construção de componentes para fornecer o produto ao cliente.
- **Iteração:** disposição estratégica das fases do projeto, onde são feitos planejamentos de retrabalho em que o tempo de revisão e melhorias de partes do sistema é pré-definido.
- **Tolerância às mudanças:** são vistas como ajustamentos às realidades do desenvolvimento de software.



- **Incremento de produtos:** prazos fixos de entrega para forçar os desenvolvedores na abrangência dos requisitos essenciais para cada versão produzida.
- **Orientação a riscos:** os riscos são abraçados pelos desenvolvedores, que são orientados a tratar os problemas mais difíceis primeiro.

A Figura 2.3 mostra o ciclo dinâmico de implementação e aprendizagem do ASD, que possui três fases: especulação, colaboração e aprendizado contínuo. Na etapa de especulação, o projeto é iniciado e são realizadas atividades relacionadas ao planejamento do ciclo adaptativo, como declaração das missões determinadas pelo cliente, definições de requisitos do usuário e restrições de projeto.

Depois de especulado, o projeto segue a fase de colaboração, onde a equipe desenvolve simultaneamente as funcionalidades definidas e integradas, aplicando conceitos de comunicação e confiança entre a equipe, bem como raciocínio colaborativo. Por fim, segue a etapa de aprendizado, onde a equipe libera o incremento de software, recebe o *feedback* do cliente, realiza revisões técnicas nos componentes desenvolvidos e análises em grupo com foco no interesse do cliente.

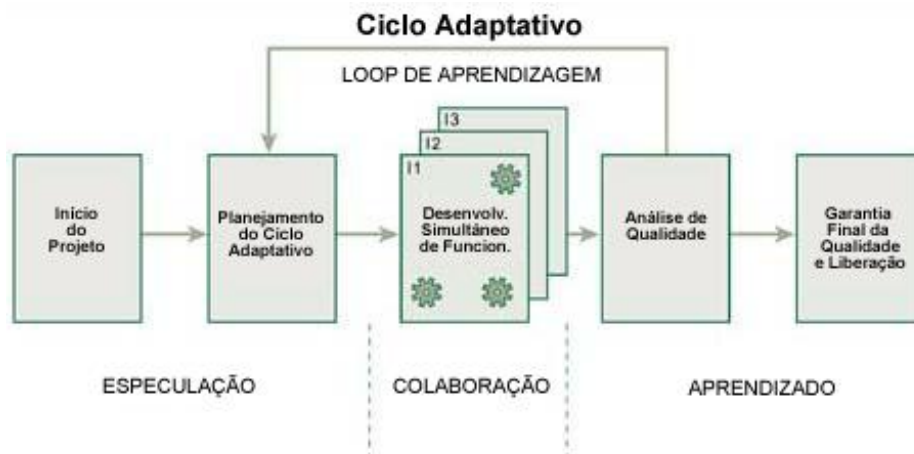


Figura 2.3 – Ciclo adaptativo da metodologia ASD. Fonte: Adaptado de (Highsmith, 2002)

No Quadro 2.4, são especificadas e descritas as principais práticas utilizadas para execução das atividades de desenvolvimento com a metodologia ASD.

Práticas	Descrição
Iteratividade	Ciclo de vida do projeto iterativo, com foco em refatoração e adaptação a mudanças.
Declaração de missão	Definição da missão da equipe, dos requisitos básicos a serem implementados.
Análise em grupo com foco no cliente	Análises de negócio e projeto com o cliente presente e focado em seus interesses.
Planejamento do ciclo adaptativo com base nos componentes	Planejamento dos ciclos do projeto baseado nos componentes incrementais.
Estimativa por missão	Planejamento do tempo gasto no andamento para a realização da missão determinada pelo cliente.
Relatório de progresso	Registro de progresso do projeto durante os ciclos.
Revisões técnicas	Adoção de técnicas de revisão e inspeção do projeto.

Quadro 2.4 – Descrição das práticas executadas durante a implementação da metodologia ASD

## 2.2.4 Feature Driven Development (FDD)

O *Feature Driven Development* – Desenvolvimento orientado a funcionalidades (FDD) é um processo projetado e testado para oferecer entregas de software frequentes e tangíveis, trabalhando em resultados repetidamente (PALMER & FELSING, 2002).

Deriva do Método COAD (análise, desenho e programação orientada a objetos) e das técnicas de gerenciamento iterativo, incremental e enxuto de projetos (PFLEEGER & ATLEE, 2006).

No processo FDD, uma funcionalidade é uma função de valor para o cliente que pode ser desenvolvida em um ciclo de uma ou duas semanas (PRESSMAN, 2008). Sendo assim, as fases do FDD se organizam na implementação de pequenas partes funcionais que podem ser entregadas e inspecionadas com maior facilidade.

O diferencial do FDD está em sua combinação de práticas de planejamento e gerencia de projetos, equilibrando conceitos de processos tradicionais e processos ágeis.

O processo do FDD possui cinco fases (HEPTAGON, 2011):

- **Desenvolvimento de um modelo abrangente:** estudo feito entre os membros do domínio do negócio e os desenvolvedores, sob a orientação de um arquiteto de software, para o entendimento do escopo do sistema. São oriundos dessa fase os modelos propostos pela equipe, sendo apenas um escolhido como modelagem orientada a objetos do sistema.
- **Construção da lista de funcionalidades:** Seguido ao processo de modelagem, os desenvolvedores-líderes transformam o domínio de negócio em atividades, que passam a ser tratadas como funcionalidades.
- **Planejamento por funcionalidade:** O gerente do projeto, juntamente com os desenvolvedores determinam a ordem de implementação das funcionalidades, de acordo com as dependências e a complexidade das mesmas.
- **Detalhamento por funcionalidade:** Nesta etapa é formada a equipe de funcionalidade, pelo desenvolvedor-líder, onde os proprietários de classe são selecionados e o diagrama de sequência da funcionalidade é construído.
- **Construção por funcionalidade:** Os desenvolvedores implementam os itens necessários para que o projeto seja funcional.

A Figura 2.4 ilustra o processo do FDD. Basicamente, a metodologia possui apenas duas fases: Concepção e planejamento do projeto, e, construção iterativa. Na fase de concepção, o negócio é estudado pela equipe com o objetivo de desenvolver um modelo abrangente do sistema orientado a objetos. Em seguida, cria-se uma lista de atividades oriundas do domínio do negócio, para que essas atividades possam ser implementadas de acordo com as prioridades definidas.

Depois do plano de desenvolvimento ser definido, a construção das funcionalidades começa a ser trabalhada. Os desenvolvedores selecionados para serem proprietários de classe são selecionados e o diagrama de progresso da implementação é construído para monitorar o andamento das atividades. Ao final do ciclo um incremento, que é um produto de software, é entregue ao cliente.

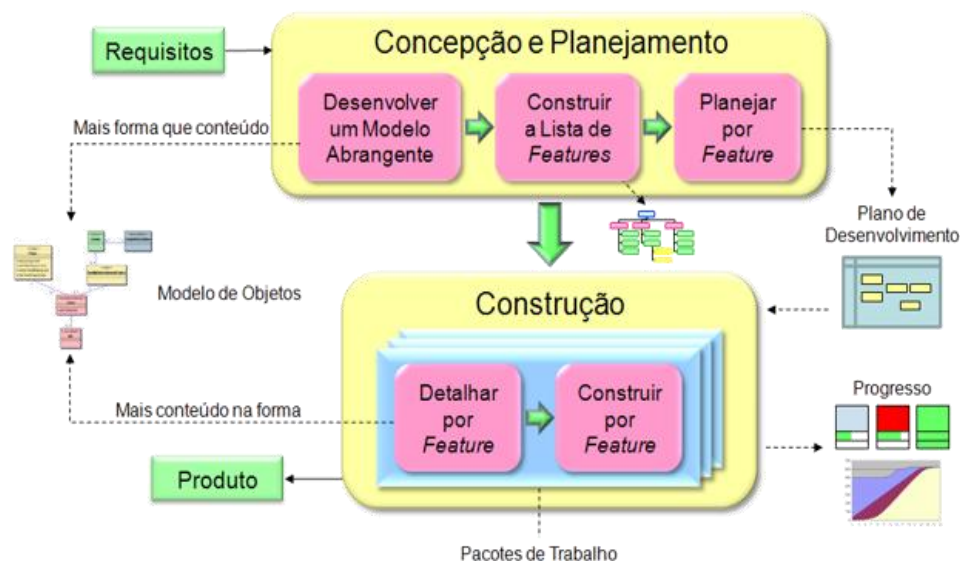


FIGURA 2.4 – Estrutura da metodologia FDD. Fonte: Adaptado de (HEPTAGON, 2011)

Esse conjunto de práticas utilizadas para o cumprimento do ciclo da metodologia FDD, é especificada e descrita no Quadro 2.5.

Práticas	Descrição
Modelagem abrangente do domínio	Modelo representativo dos requisitos do sistema (funcionalidades).
Desenvolvimento por funcionalidade	Desenvolvimento a partir de uma lista de funcionalidades de valor ao cliente.
Propriedade de classe individual	Individualidade do código entre os programadores (cada classe possui seu responsável)
Relatório de progresso	Registro de progresso do projeto durante a iteração.
Inspeção	Adoção de técnicas de inspeção.
Equipes de funcionalidade	Equipes de proprietários de classe.
<i>Potentially Shippable</i>	Incremento de produtos potencialmente entregável.

Quadro 2.5 – Descrição das práticas executadas durante a implementação da metodologia FDD

## 2.2.5 Dynamic Systems Development Method (DSDM)

O *Dynamic Systems Development Method* (DSDM) consiste em uma metodologia ágil de desenvolvimento incremental e iterativo para a construção de sistemas com restrição de prazo (PRESSMAN, 2006).

Foi baseada no processo RAD, utilizando como filosofia o princípio de Pareto: 80% das funcionalidades de um software podem ser entregues em 20% do tempo proposto pra entregar o sistema completo (PFLEEGER & ATLEE, 2006; PRESSMAN, 2006).

Atualmente, a metodologia está sob cuidados de uma comunidade mundial de empresas sem fins lucrativos, denominada *DSDM Consortium* (DSDM ATERN - THE AGILE PROJECT FRAMEWORK, 2011).

Na Figura 2.5, o ciclo de vida do DSDM é demonstrado. A metodologia possui cinco fases definidas em três ciclos iterativos: pré-projeto, ciclo de vida e pós-projeto (ABRAHAMSSON *et al.*, 2002; PRESSMAN, 2006; DYBÅ & DINGSØYR, 2008):

- **Estudo de viabilidade:** fase onde são estabelecidos os requisitos básicos e as restrições do negócio. Em seguida, é feita uma avaliação se o projeto proposto é viável à metodologia DSDM.
- **Estudo do negócio:** é definida a arquitetura do software a ser construído, bem como os requisitos funcionais e de maior valor ao projeto.
- **Iteração do modelo funcional:** nesta fase são produzidos protótipos que representam a funcionalidade requerida pelo cliente. Os protótipos tem como objetivo obter requisitos adicionais através do contato com o cliente.
- **Iteração de projeto e construção:** é executada uma revisão dos protótipos construídos durante a Iteração do modelo funcional, para agregar valor aos usuários finais do software.
- **Implementação:** codificam os protótipos como incrementos de software funcionais.



Figura 2.5 – Ciclo de vida da metodologia DSDM. Fonte: Adaptado de (PINEYWOODS TECH, 2011)

Para a implementação do DSDM, o conjunto de práticas sugeridas pela metodologia é descrito no Quadro 2.6.

Práticas	Descrição
Regra de Pareto (80%/20%)	80% das funcionalidades podem ser entregues em 20% do tempo proposto pra entregar o sistema completo
Tomada de decisão pela equipe	Democracia na tomada de decisões do projeto.
Cliente presente	O cliente deve estar presente e participar de todas as etapas do processo de desenvolvimento
<i>User story</i> iterativo	Atividade de planejamento em que são criados conjuntos de histórias e descrevem características e funcionalidades do sistema durante a iteração
Estudo do objetivo do negócio	Estudo do domínio do negócio, suas restrições e técnicas
<i>Potentially Shippable</i>	Incremento de produtos potencialmente entregável.
Mudanças reversíveis	Possibilidade de reversão de mudanças no projeto, caso haja problemas.
Testes integrados	Os testes são realizados a cada iteração.

Quadro 2.6 – Descrição das práticas executadas durante a implementação da metodologia DSDM

## 2.2.6 Crystal methodologies

É um conjunto de metodologias criado por Cockburn (2002), partindo da premissa de que todo projeto tem necessidades, convenções e uma metodologia diferente (PFLEEGER & ATLEE, 2006).

Para Cockburn (2002), o funcionamento do projeto é influenciado por fatores humanos, e há melhora neste quando os indivíduos produzem melhor, se comunicam melhor e os lançamentos frequentes reduzem a necessidade de construir produtos intermediários do processo.

Tal qual acontece com as propriedades geológicas dos cristais, cada uma das metodologias da família *Crystal* corresponde a uma cor diferente, dependendo da carga de comunicação entre a equipe, prioridade do projeto e sua criticidade (COCKBURN, 2006): *Crystal Clear*, *Crystal Yellow*, *Crystal Orange*, *Crystal Orange Web*, *Crystal Red*, *Crystal Magneta*, *Crystal Blue*.

A Figura 2.6 mostra o gráfico de análise da metodologia *Crystal* e os três fatores que influenciam a escolha da metodologia: a carga de comunicações (tamanho de equipe), a criticidade do sistema, e prioridades do projeto. Ao movimentar-se no eixo horizontal, seleciona-se a pauta o tamanho da equipe, ou seja, na medida em que se afasta do eixo, a comunicação face a face entre os membros da equipe reduz.

Ao movimentar-se no eixo vertical, seleciona-se a prioridade do projeto e sua criticidade, ou seja, na medida em que se afasta do eixo, o potencial do sistema em causar danos como perda de conforto, perdas financeiras ou perda no ciclo de vida do projeto aumenta.

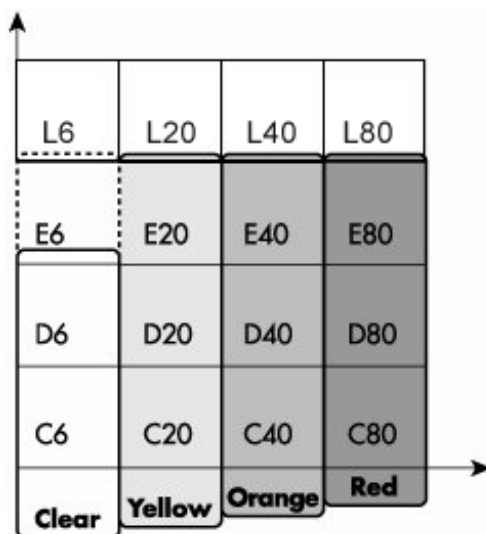


FIGURA 2.6 – Ciclo de vida da família da metodologia *Crystal*. Fonte: Adaptado de (Cockburn, 2004)

Das soluções propostas pela família *Crystal*, são descritas abaixo as duas mais utilizadas, o *Crystal Clear* e o *Crystal Orange* (HIGHSMITH, 2002):

- **Crystal Clear:** sua aplicação é dirigida a um domínio de projeto pequeno, de ciclo de vida não crítico, onde o número de pessoas na equipe é reduzido. Os papéis se restringem ao programador, o investidor, designer e o usuário. O planejamento feito é semelhante ao proposto pelo *Scrum*, com adição do desenvolvimento de casos de teste. A entrega do software é realizada de maneira incremental a cada dois ou três meses.
- **Crystal Orange:** sua aplicação é dirigida a um domínio de projeto grande, cujo ciclo de vida é crítico, onde o número de pessoas na equipe é até 40 pessoas.

No Quadro 2.7, as principais práticas utilizadas na família de metodologias *Crystal* são descritas. À medida que o projeto cresce em complexidade, essas práticas são adaptadas para atender a proposta do projeto.



Práticas	Descrição
Refino de metodologia	Seleção do tipo de metodologia da família <i>Crystal</i> , de acordo com o projeto.
<i>Holistic Diversity Strategy</i>	Elaboração de estratégias para montagem de equipes multifuncionais.
Documento de visão	Documento com as visões e idéias do usuário sobre o produto de software a ser construído.
Várias equipes trabalhando em paralelo	Múltiplas equipes podem ser utilizadas com o máximo de paralelismo.
Revisão e análise	A cada iteração são realizadas atividades de revisão e análise dos objetivos do incremento.
Desenvolvimento de casos de teste	Criar testes simulando condições de entrada e saída para verificar não somente as condições inválidas de execução, como também as condições válidas.
<i>Potentially Shippable</i>	Incremento de produtos potencialmente entregável.
Iterações fixas	Ciclo de desenvolvimento em iterações com tempo de entrega fixo.
Monitoramento de progresso	Monitoramento do progresso do projeto durante a iteração.
<i>Workshop</i> de análise	Workshops de reflexão das atividades da iteração, podendo ser realizadas antes ou depois da mesma.

Quadro 2.7 – Descrição das práticas executadas durante a implementação das metodologias da família *Crystal*

## 2.2.7 Test Driven Development (TDD)

O *Test Driven Development* (TDD) – Desenvolvimento Guiado a Testes é uma metodologia ágil de desenvolvimento de software baseado em casos de teste. Seus princípios foram elaborados por Kent Beck a partir da prática de *test-first* da metodologia ágil *Extreme Programming* (XP) (BECK, 1999).

A idéia por trás do TDD é aparentemente simples: escrever os testes para seu código antes de escrever o próprio código, não só com o objetivo de solucionar defeitos, mas para entender as funcionalidades de forma confiável e previsível (FREEMAN & PRYCE, 2010).

É amplamente utilizada na comunidade ágil, em conjunto com outras metodologias, como *Extreme Programming* (XP) (BECK, 1999), *Crystal Clear* (COCKBURN, 2004), e é frequentemente utilizado em projetos gerenciados por *Scrum* (SCHWABER, 2002).

Para Stellman e Greene (2005), o TDD também ajuda os programadores entender melhor os requisitos e garante que os testes unitários estão sempre corretos.

Uma vantagem na utilização do TDD é o encurtamento do ciclo de *feedback* sobre as decisões de *design*: ao invés de projetar e então esperar semanas ou meses pelo retorno do cliente, o *feedback* vem em segundos ou minutos, ao tentar traduzir suas idéias em uma interface plausível (BECK, 2002).

O ciclo de desenvolvimento com TDD possui seis fases:

- **Adição de teste:** criação de um caso de teste para uma nova funcionalidade, com auxílio do caso de uso ou de *User Story*.
- **Execução do teste procurando por falhas:** nesta fase os testes são validados verificando se todos os testes estão funcionando corretamente. Se o teste não passar, significa que a funcionalidade preterida ainda não foi implementada.
- **Escrever o código:** os desenvolvedores escrevem algum código que faz com que o teste seja validado.
- **Executar testes automatizados:** os desenvolvedores executam testes no código para averiguar se código possui todos os requisitos testados.
- **Refatoração:** A partir desta etapa, o código inicialmente escrito pode ser aprimorado buscando soluções mais coesas e limpas, sem perder a funcionalidade do código principal.
- **Repetição:** Com o código refatorado e validado pelos testes, o incremento é entregue ao cliente e outro teste é iniciado, repetindo o ciclo de desenvolvimento.

As seis fases descritas acima podem ser visualizadas e explicadas através da Figura 2.7.

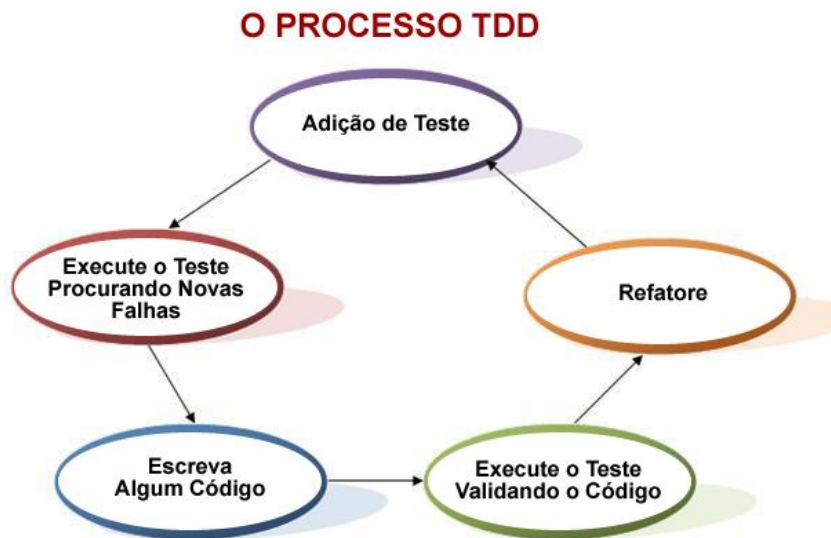


Figura 2.7 – Ciclo de vida do TDD. Fonte: Adaptado de (GESTÃO DE PROJETOS DEBLM, 2011)

As práticas recomendadas pelo TDD, descritas no Quadro 2.8, se assemelham com as propostas pela metodologia *Extreme Programming* (XP), porém sua utilização se restringe na fase de codificação de testes.

Práticas	Descrição
Pareamento	Escrever testes em pares, trocando idéias durante a programação do código.
Trabalhe descansado	A metodologia aconselha: trabalhe apenas quando tiver com a mente arejada e pare quando apresentar sinais de cansaço.
Integração contínua	Prática de desenvolvimento de software onde os membros de um time integram seu trabalho frequentemente.
<i>Design</i> simples	Cria-se a solução mais simples possível, suficiente para implementar as funcionalidades requeridas.
Refatoração	Processo de aprimoramento do código existente, sem mudar seu comportamento externo.
Entrega contínua	Se os testes com o TDD passam mais rapidamente, é possível a

	disponibilização de código em produção frequentemente, sem incomodar o cliente.
--	---

Quadro 2.8 – Descrição das práticas executadas durante a implementação da metodologia TDD

## 2.3. Valor e percepção de uso

Kötler (2000) define valor como um indicador da contribuição de uma empresa para com o seu cliente, baseado na coleção de produtos, serviços e intangíveis que a empresa oferece. Para o cliente, um produto representa valor se este responde positivamente aos seus desejos e suas necessidades são atendidas.

Com a crescente complexidade do processo de desenvolvimento de software e as pressões de mercado, as empresas estão buscando a verificação e validação de práticas que asseguram maior valor agregado de um produto de software (ABRAHAMSSON *et al.*, 2002; SCHWABER, 2004; BIFFL *et al.*, 2006).

De acordo com Kötler (2000), as percepções são opiniões ou experiências fundamentadas a partir do impacto de informações recebidas e processadas pelos indivíduos ou grupos sociais, que podem alterar ou manter suas opiniões em função de um efeito que o conteúdo analisado representa sobre os mesmos.

No ponto de vista da Engenharia de Software, a perspectiva de valor fornece uma boa maneira de olhar para o processo de desenvolvimento de um produto, criando estratégias para alcançar um crescimento rentável em longo prazo e uma vantagem competitiva sustentável para as empresas de software (BIFFL *et al.*, 2006).

Todavia, os desenvolvedores de software precisam considerar quais são os elementos-chave de valor em termos de como criar valor para seus atuais e futuros produtos de software, bem como entregar este valor para um cliente da forma mais rentável e sustentável possível (BIFFL *et al.*, 2006).

Portanto, a importância da avaliação da percepção de valor da comunidade de software no presente trabalho constitui em um passo fundamental para o aprimoramento de conhecimentos sobre o desempenho de práticas ágeis em empresas do mundo.

## 2.4. Trabalhos relacionados

Através de uma revisão de literatura, foram encontrados artigos relacionados ao trabalho proposto. A estratégia de pesquisa foi realizada em bases de dados acessadas por meio de um portal brasileiro de buscas de periódicos, o Portal Brasileiro de Periódicos da CAPES – Agência Brasileira de Apoio e Fomento a Pesquisa.

As pesquisas foram realizadas nas seguintes bases de dados eletrônicas:

- Biblioteca Digital da ACM
- *IEEE Xplore*
- *ScienceDirect – Elsevier*
- *SpringerLink*
- *Wiley Inter Science Journal Finder*

Para a busca de títulos, resumos e palavras-chave dos artigos incluídos nas bases de dados pesquisadas, foram utilizados os seguintes termos de pesquisa:

- *Agile software development*
- *Agile methods*
- *Agile methodologies*
- *Software Engineering*
- *Agile software management*
- *Agile software project*
- *Agility*
- *Agile practices*
- Engenharia de software
- Metodologias ágeis

A partir destes critérios de busca foram encontrados 60 artigos relacionados à área de Engenharia Software, especificamente sobre metodologias ágeis. Apenas quatro destes artigos encontrados estão diretamente relacionados ao tema do trabalho e suas características e resultados serão brevemente descritos.

Ahmed *et al.* (2010), executou um estudo das práticas que faz a adoção de princípios ágeis mais sucedida quanto ao aumento da produtividade e qualidade. O mapeamento destas práticas se deu através de um questionário que permaneceu online por oito dias e obteve 42 respondentes.

A partir dos resultados, os autores elaboraram um modelo de processo de desenvolvimento genérico, mais voltado aos princípios ágeis que a práticas em si. Em adição, a avaliação estatística utilizada para encontrar os padrões de princípios foi básica, principalmente devido ao tamanho da amostra.

Chow e Cao (2008) propuseram a identificação de fatores críticos de sucesso de projetos ágeis de software utilizando uma abordagem qualitativa sob quatro categorias de sucesso: qualidade, custo, prazo e escopo. Porém o trabalho se limita quanto ao tamanho da regionalidade e o numero pequeno da amostra trabalhada, totalizando 109 questionários respondidos.

O número reduzido de pesquisas empíricas sobre agilidade em termos de dimensões, determinantes e efeitos de desempenho sobre o desenvolvimento de software foi a motivação para os estudos de Lee e Xia (2010).

Usando uma abordagem de pesquisa integrada que combinam análises de dados qualitativos e quantitativos, a pesquisa busca examinar empiricamente as relações entre duas dimensões de desenvolvimento de software agilidade (amplitude de resposta da equipe de software e eficiência da resposta da equipe de software), dois antecedentes que podem ser controlados (autonomia da equipe e da diversidade da equipe), e três aspectos de desempenho do desenvolvimento de software (prazo final, no orçamento de conclusão e funcionalidade do software).

A pesquisa obteve 399 questionários respondidos por gerentes de projeto de software. Os resultados sugerem que a amplitude de resposta do time e eficiência de resposta do time impacta no desempenho da agilidade de desenvolvimento de software de

forma diferente: a eficiência de resposta afeta positivamente na conclusão no prazo, na conclusão orçamental e na funcionalidade do software, enquanto a amplitude de resposta afeta positivamente apenas a funcionalidade do software.

Para Cao *et al.* (2009), a utilização de práticas das metodologias ágeis devem ser adaptadas para atender às necessidades de diferentes contextos.

Utilizando a teoria da estruturação adaptativa, os autores examinaram como a estrutura da metodologia ágil *Extreme Programming* (XP), projetos e organizações afetam na adaptação destas metodologias. Os resultados foram descritos em várias estruturas que afetam a adoção de práticas ágeis, o conjunto de práticas apropriadas e suas características, e sua relação com processo de resultados.

## 2.5. Considerações Finais

A complexidade e o caráter dinâmico do processo de desenvolvimento de software tem tornado metodologias tradicionais de desenvolvimento de software dispendiosas e, muitas vezes, inadequadas para organizações que necessitam de desenvolvimento ágil e respostas rápidas a mudanças.

Nesse cenário, surgiram as metodologias ágeis, que enfatizam princípios associados à comunicação, objetividade, maior foco no desenvolvimento e interação com o cliente, de forma a proporcionar agilidade e eficiência no desenvolvimento.

Existem diversos exemplos de metodologias ágeis propostas na literatura e em geral elas se baseiam no conjunto de princípios estabelecidos no Manifesto Ágil. Portanto, mesmo com uma variedade de metodologias apresentadas, elas possuem valores e práticas muito semelhantes.

Segundo Charette (2001), projetos utilizando métodos ágeis obtiveram melhores resultados quanto aos prazos, qualidade e custos, tornando-se um interessante contexto de investigação de quais as principais práticas ágeis que trazem vantagem para o projeto, segundo estes mesmos critérios.

Além de Charette (2001), diversos estudos em Engenharia de Software têm investigado como práticas de metodologias ágeis são percebidas pelos praticantes utilizando como parâmetro os critérios de desempenho e dimensões de projetos como custo, qualidade, prazo e escopo, ou, tratando-os de forma individual.

Todavia, estes trabalhos se limitam em avaliar projetos utilizando metodologias específicas e de maior popularidade, tais como XP e *Scrum*. A amplitude e diversidade de práticas e metodologias ágeis, portanto, é desconsiderada na maioria dos estudos.

Em contraposição a estes estudos, este trabalho se propôs a investigar e avaliar todas as práticas utilizadas no desenvolvimento ágil, considerando a percepção do usuário destas práticas e o valor agregado às mesmas.



## 3 METODOLOGIA

A metodologia de pesquisa define o que foi pesquisado no trabalho e como este foi realizado, seguindo as etapas de concepção, planejamento, aplicação e conclusão.

### 3.1. Método de pesquisa

O método científico de uma pesquisa corresponde a um caminho para se chegar ao fim de um determinado trabalho (GIL, 1999).

Existem várias maneiras de se classificar uma pesquisa, sendo os pontos mais tradicionais: quanto à natureza da pesquisa, a forma de abordagem do problema, os seus objetivos e os procedimentos técnicos (SILVA & MENEZES, 2000).

Em relação à natureza, a pesquisa pode ser classificada como aplicada ou tecnológica, uma vez que o objetivo é gerar conhecimentos para aplicação prática, dirigidos a solução de um problema específico (JUNG, 2004).

Quanto aos objetivos, a pesquisa pode ser classificada como do tipo descritiva (JUNG, 2004; HAIR *et al.*, 2009). Para Jung (2004), na pesquisa do tipo descritiva visa-se a identificação, o registro e a análise de características, fatores ou variáveis que se relacionam com o fenômeno ou processo.

Quanto à forma de abordagem, pode-se enquadrar a pesquisa como de caráter quantitativo considerando a população de empresas que utilizam metodologias ágeis para desenvolvimento de produtos de software. (JUNG, 2004; HAIR *et al.*, 2009).

Em relação aos procedimentos técnicos, esta pesquisa caracteriza como um levantamento de dados mediante a uma pesquisa de campo, com aplicação de um questionário para coleta, análise e interpretação de dados (JUNG, 2004).

A metodologia da pesquisa pode ser mais bem esclarecida através de um desenho de pesquisa, onde são apresentados os seus detalhes nas fases do seu desenvolvimento, nas técnicas e nos instrumentos de coleta e análise dos dados.

### 3.2. Desenho da pesquisa

O desenho da pesquisa contempla os seus componentes que são demonstrados em seqüências lógicas (YIN, 2010). A representação gráfica do desenho da pesquisa pode ser visualizada na figura 1.

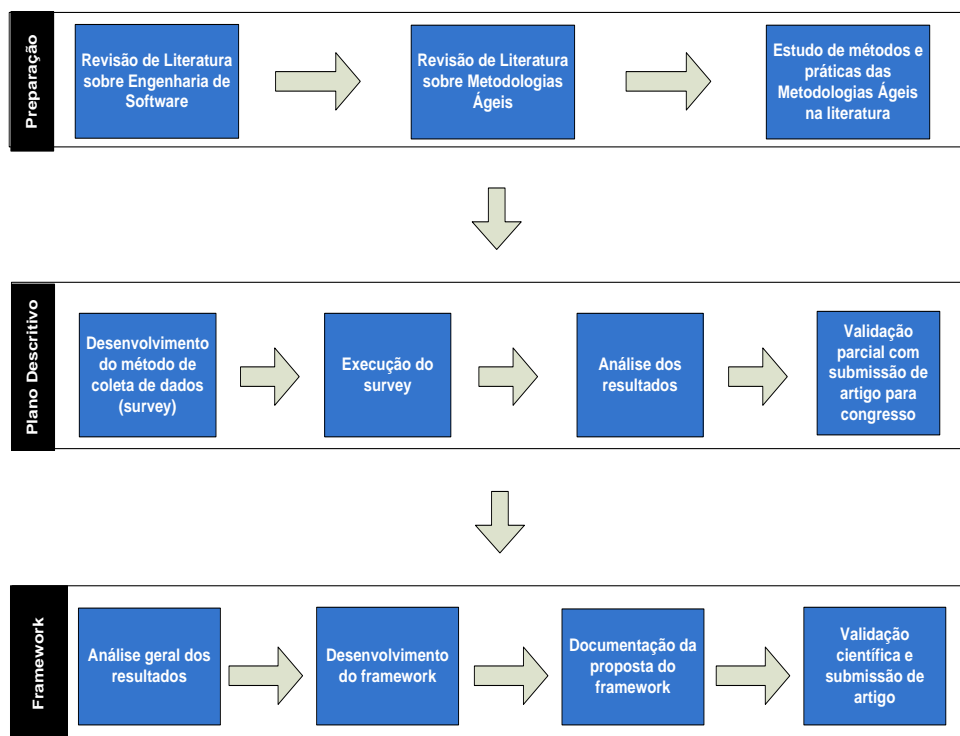


Figura 3.1 – Desenho da pesquisa

O desenho proposto para a pesquisa contempla três etapas, sendo cada uma delas dividida em um conjunto de fases que, por sua vez, dividem-se em atividades.

A etapa de Preparação caracteriza-se pela revisão de literatura, que teve por objetivo identificar em trabalhos científicos e livros, o conjunto de práticas utilizadas nas metodologias ágeis existentes: XP, *Scrum*, ASD, FDD, DSDM, TDD, *Crystal Methodologies*.

Tal como relatam Hair *et al.* (2009), uma completa revisão de literatura pode ser vantajoso para melhor compreensão de uma questão a ser investigada. Nesse sentido, essa fase tem como objetivo o levantamento de: (i) principais metodologias ágeis existentes; e (ii) as práticas descritas na literatura. As informações levantadas na literatura servirão de subsídio para o desenvolvimento do estudo descritivo.

Na segunda etapa da pesquisa, o Plano Descritivo, teve como objetivo investigar as práticas e a percepção de valor das empresas que utilizam metodologias ágeis no desenvolvimento de software sobre os critérios: custo, qualidade, prazo e escopo.

Para tanto, foi aplicado um questionário, através da ferramenta *online SurveyMonkey*, sobre a utilização de práticas de metodologias ágeis que, posteriormente, seus resultados foram analisados por métodos de estatística descritiva e exploratória, a fim de se verificar as hipóteses ou problemas de pesquisa levantados durante a etapa de revisão de literatura.

A análise dos resultados do questionário foi realizada através do software de análise estatística SPSS (*Statistical Package for the Social Sciences*), versão 18, para o sistema operacional Windows e do macro XLSTAT do *Microsoft Office Excel*.

Durante a etapa de Framework, foram documentados os resultados das análises da pesquisa e, a partir daí, o desenvolvimento do modelo de aprimoramento do gerenciamento e desenvolvimento ágil baseado na percepção de valor do usuário.

### **3.3. Coleta de dados para a pesquisa**

A pesquisa esteve disponível por quatro meses *online* no software de elaboração de questionários para a *web SurveyMonkey*, obtendo um total de 253 questionários respondidos. Em adição a essas abordagens, ao final do período de disponibilização do questionário, foi realizado o sorteio de um livro com a temática de Metodologia Ágil, previamente anunciado na descrição da pesquisa deixada nas comunidades participantes.

O questionário foi construído a partir da etapa de revisão de literatura, onde foram mapeadas as principais práticas utilizadas nas metodologias ágeis e continha um total de 15 perguntas, distribuídas em três partes: perfil do respondente e da empresa, avaliação de

práticas ágeis e avaliação de benefícios e desafios enfrentados na adoção de metodologias ágeis.

Entretanto, por ter como público alvo a comunidade de desenvolvimento de software, o questionário possuía um caráter bi-direcional. Caso o respondente não utilizasse metodologias ágeis no desenvolvimento de software em sua empresa, ele respondia um conjunto de perguntas relacionadas a benefícios da adoção de metodologias ágeis listados na literatura, que ele achava mais importantes e dificuldades ou barreiras, também mapeadas na literatura, encontrados que em sua visão, impedem a empresa de adotar metodologias ágeis.

### **3.4. Abrangência da pesquisa**

A abrangência ou delimitação da pesquisa estabelece os limites de investigação: população amostragem e seleção de participantes (MARCONI & LAKATOS, 2003).

O presente trabalho possui dois tipos de amostragem: especialista, cujos elementos são especialistas ou possuem conhecimento em uma determinada área; propagação geométrica (*snowball*), onde se pretende incluir na amostra sujeitos pouco acessíveis ou com determinado atributo difícil de encontrar (HAIR *et al.*, 2009).

Quanto à abordagem de amostragem especialista consiste em praticantes e não praticantes de metodologias ágeis do Brasil e do mundo cujo acesso à pesquisa se deu em comunidades de redes sociais, listas de discussão e fóruns voltados para o desenvolvimento de software e Engenharia de Software.

Quanto à abordagem de amostragem por propagação geométrica, a utilização ocorreu durante a execução do questionário, onde, ao final das 15 perguntas, os respondentes eram questionados sobre a possibilidade de indicar para a equipe de pesquisa uma pessoa que este achasse adequada para responder o questionário.

Para a construção do banco de dados da pesquisa foram considerados apenas os questionários que concluíram todas as respostas, totalizando um número de 253 respondentes.

### 3.5. Análise de dados da pesquisa

Segundo Appolinário (2006), na pesquisa quantitativa a análise obedece a um plano pré-estabelecido, confirmando as hipóteses da pesquisa ou descobertas por dedução, utilizando dados que representam uma população específica (amostra), a partir da qual os resultados são generalizados.

Na etapa da análise, a percepção de valor dos usuários foi analisada utilizando técnicas de análise estatística exploratórias.

De acordo com Hair *et al.* (2009), a análise exploratória é utilizada quando se deseja descobrir tendências, relações e padrões ocultos ao pesquisador em uma coleção de dados analisados. Dentre as técnicas utilizadas neste trabalho para entender as relações entre as práticas ágeis e a sua percepção de uso por aqueles que as utilizam em seus projetos, estão a Análise Fatorial e a Análise de *Cluster*.

A Análise Fatorial é uma técnica de análise exploratória de dados que possui como objetivo a análise da estrutura de um conjunto de variáveis inter-relacionadas, tornando-se possível a construção de uma escala de medidas para fatores, que podem diretamente ou não, controlar as variáveis da pesquisa (HAIR *et al.*, 2009; MAROCO, 2010).

A finalidade de sua utilização consiste na redução de dados ou na simplificação da sua estrutura, na classificação e no agrupamento, na investigação de dependência entre variáveis, na predição e na elaboração de hipóteses e testando-as (JOHNSON & WICHERN, 2007).

No entanto, segundo Mingoti (2005), este conceito não é utilizado para prever uma variável dependente, mas sim para maximizar e sumarizar o poder de explicação do conjunto inteiro das variáveis originais encontrando os fatores ou variáveis latentes.

A Análise fatorial inclui alguns modelos análise, sendo os mais comuns a análise de componentes principais e análise dos fatores comuns.

A matriz de componentes inicial gerada destes modelos indica a relação entre os fatores e as variáveis individuais, mas raramente resulta em fatores que possam ser interpretados.

A Análise Fatorial torna-se mais ou menos útil em função da sua capacidade de produzir fatores, através dos métodos de rotação da matriz, que transforma a matriz de fatores em uma matriz rotacionada mais simples e fácil de interpretar (MALHOTRA, 2001).

Segundo Hair *et al.* (2009), o efeito final da matriz fatorial rotacionada é redistribuir a variância dos primeiros fatores para os últimos com o objetivo para atingir um padrão fatorial mais simples e mais significativo.

Há diferentes tipos de rotações possíveis de aplicação. Para este trabalho foi utilizado o método de rotação ortogonal VARIMAX, que é o mais utilizado por se concentrar na máxima simplificação das colunas da matriz fatorial através da maximização da soma de variâncias de cargas exigidas da matriz fatorial (MAROCO, 2010).

Uma vez executada a análise, foi possível a identificação das práticas correlacionadas que representam maiores valores segundo a percepção dos respondentes, sendo redimensionadas em fatores de impacto.

Apenas os fatores encontrados utilizando como variáveis as práticas com  $n > 35$ , foram considerados para a elaboração de um modelo de desenvolvimento e gerenciamento ágil. Esse critério foi considerado por representar uma amostra mais significativa para o trabalho.

Como etapa complementar a Análise Fatorial, foi executada a Análise de *Cluster*, com o objetivo de identificar semelhanças na percepção de valor dos respondentes da pesquisa para com as práticas, de acordo com o cargo por eles ocupado em sua organização.

A Análise de *Cluster* ou análise de agrupamentos tem por finalidade obter grupos distintos entre si, mas que, cada cluster contenha elementos de características similares com baixa dispersão interna (JOHNSON & WICHERN, 2007; HAIR *et al.*, 2009).

O método de agrupamento utilizado foi o método hierárquico de *Ward*, que visa sempre minimizar a variabilidade interna dentro dos *clusters*. O procedimento de *Ward*, a seleção dos agrupamentos é baseada em qual combinação de agregados minimiza a soma interna de quadrados no conjunto completo (HAIR *et al.*, 2009). Segundo Mingoti (2005),

esse procedimento tende a produzir grupos com aproximadamente o mesmo número de elementos ou observações.

A medida de similaridade utilizada foi a medida de distância euclidiana quadrada ou absoluta, que é recomendada para métodos de agrupamento de Ward (HAIR *et al.*, 2009).

O produto desta análise é uma árvore hierárquica, onde os sujeitos da análise, que representam cada *cluster*, são sumarizados em um padrão de agrupamento (MAROCO, 2007).

## 4 RESULTADOS

Este capítulo apresentará o resultado dos quatro estudos oriundos dos problemas de pesquisa e dos objetivos do trabalho:

- 4.1) Levantamento e análise de semelhança de práticas;
- 4.2) Instrumento para a avaliação da percepção de valor em metodologias ágeis;
- 4.3) Análise demográfica da amostra;
- 4.4) Análises estatísticas;
- 4.5) Proposta de modelo de aprimoramento do gerenciamento e desenvolvimento ágil baseado na percepção de valor do usuário (*User-based Perception Model – UBPM*)

### 4.1. Levantamento e análise de semelhança das práticas

Como resultados da etapa de revisão de literatura foram identificados as principais práticas aplicadas nas metodologias ágeis estudadas: XP, *Scrum*, ASD, FDD, DSDM, ASD, *Crystal Methodologies* e TDD.

As técnicas dessas metodologias foram analisadas e interpretadas visando à verificação de semelhança entre práticas relacionadas às etapas de formação de equipes, planejamento do projeto, reuniões, técnicas de desenvolvimento e monitoramento do projeto e seu contexto de utilização na metodologia.



Através desta análise tornou-se possível a criação de um quadro de práticas, abordando os conceitos semelhantes executados nas etapas do ciclo ágil. Esses conceitos estão identificados na Figura 4.1.

Etapas do projeto	Práticas (das metodologias)									
	XP	Scrum	ASD	FDD	DSDM	Crystal	TDD			
Planejamento	Design Incremental <i>Spike Solutions</i>	<i>Sprint Planning</i>	Ciclo adaptativo	Modelo abrangente	Estudo do objetivo do negócio	Refino de metodologia	-			
Análise de requisitos	Cartões de classes e funcionalidades (CRC) <i>User story</i>	<i>Product Backlog</i> <i>Sprint Backlog</i>	Declaração de missão	Lista de funcionalidades	<i>User story</i> iterativo	Documento de visão	-			
Regras	Build de 10 minutos	Ciclos de 2-4 semanas	-	Desenvolvimento por funcionalidade	Pareto (80%/20%) Mudanças reversíveis	Iterações fixas <i>Holistic Diversity Strategy</i>	Trabalhe descansado			
Equipes	Duplas	Multidisciplinar	Multidisciplinar	Equipes de funcionalidades	Equipes pequenas	Várias equipes trabalhando em paralelo	Solo/Duplas			
Codificação	Refatoração Integração contínua Programação em pares Propriedade coletiva	-	Revisões técnicas	Propriedade de código individual Inspeção de código	-	-	Pareamento Refatoração Integração contínua			
Estimativas	Jogos de planejamento	<i>Sprint Planning</i>	Estimativa por missão	Estimativa por funcionalidade	Estimativa por funcionalidade	Estimativa por funcionalidade	-			
Reuniões	Reuniões em pé	Reuniões diárias <i>Sprint review</i>	Análise em grupo com foco no cliente	Desenvolvimento do modelo abrangente	Revisão do negócio	Workshop de análise	-			
Monitoramento	Projeto <i>Velocity</i>	<i>Burndown chart</i>	Relatório de progresso	Relatório de progresso	Relatório de progresso	Relatório de progresso	-			
Testes	Testes primários Triagem de erros	-	Testes integrados	Testes integrados	Testes integrados	Desenvolvimento por casos de teste	Desenvolvimento por casos de teste			
Versionamento	Frequente	Frequente	Frequente	Frequente	Frequente	Frequente	Frequente			

Figura 4.1 – Semelhanças entre os conceitos das principais práticas ágeis utilizadas nas metodologias estudadas

A Figura 4.1 representa um mapeamento das práticas de metodologias ágeis descritas na literatura. Como resultado desta etapa, foi obtido uma tabela de práticas com conceito de aplicação e significação semelhantes durante a etapa de ciclo de vida de um projeto.

O objetivo do levantamento de semelhanças de práticas foi reduzir o número de práticas listadas e agrupá-las em uma quantidade satisfatória para serem avaliadas pelos participantes da pesquisa, durante a aplicação de um questionário.

## **4.2. Instrumento para avaliação da percepção de valor em metodologias ágeis**

O instrumento para avaliação da percepção de valor em metodologias ágeis utilizado foi um questionário, construído a partir da etapa de revisão de literatura, onde foram mapeadas as principais práticas utilizadas nas metodologias ágeis.

O questionário continha um total de 15 perguntas, divididas em três fases: perfil demográfico, avaliação da percepção de práticas ágeis e percepção de benefícios e dificuldades.

Na fase de descrição do perfil demográfico os respondentes, informavam seus dados pessoais (nome, idade, *email*), caracterização da empresa em que trabalha e o cargo ocupado.

A descrição dos dados, com exceção do cargo e do tamanho da empresa não era uma questão obrigatória, mas o respondente que não informasse seus dados, não poderia concorrer ao sorteio do livro. O objetivo desta fase do questionário era de caracterização da amostra.

A segunda fase constitui na contextualização de avaliação de práticas ágeis sobre os critérios custo, qualidade, prazo e escopo. Os respondentes indicavam a sua percepção de valor de uso de práticas através de uma escala *Likert* de seis pontos, onde se avaliava as práticas pelos valores: muito alto (6), alto (5), satisfatório (4), regular (3), baixo (2) e muito baixo (1). Os respondentes não eram obrigados a avaliar todas as práticas, apenas as que

ele já utilizou ou utiliza. Essa medida foi tomada para evitar respostas viciadas, devido ao não conhecimento da prática especificada.

Outra característica importante do questionário é o seu caráter bi-direcional, uma vez que o mesmo foi disponibilizado em comunidades sociais abertas, com uma população heterogênea. Caso o respondente não utilizasse metodologias ágeis no desenvolvimento de software em sua empresa, ele respondia um conjunto de perguntas relacionados à percepção de valor em relação à benefícios e dificuldades na adoção de metodologias ágeis mais citados na literatura.

Ao final do questionário, caso fosse de interesse do respondente, este poderia indicar uma nova pessoa que estaria apta a responder o questionário.

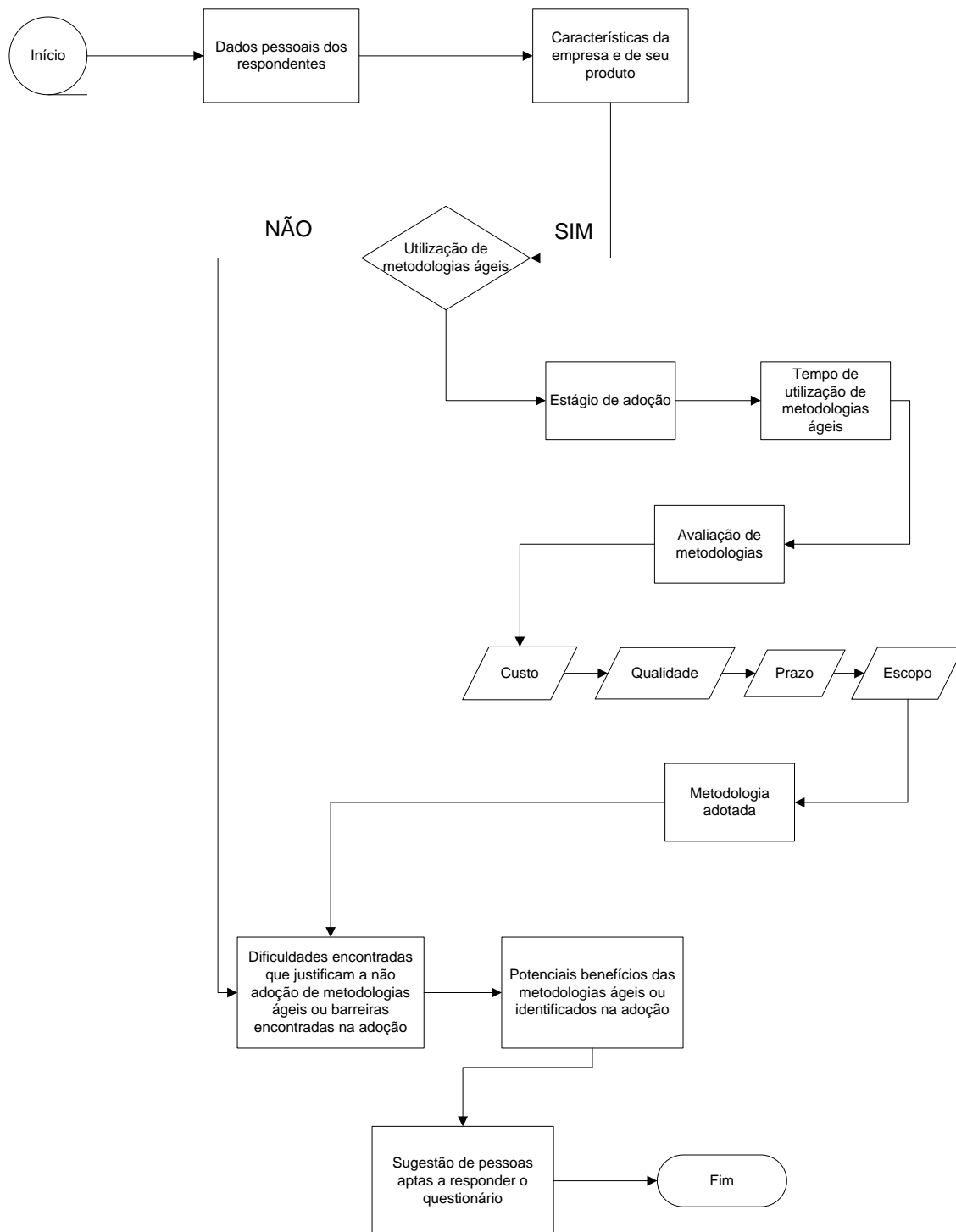


Figura 4.2 – Estrutura do questionário

Desta forma, o questionário da pesquisa foi construído para que, de maneira sucinta, tornasse possível a investigação do valor percebido na utilização de práticas de

metodologias ágeis na população de indivíduos que trabalham no desenvolvimento e gerenciamento de software.

Todavia, pode-se perceber que ainda que o universo da pesquisa esteja devidamente especificado em 15 perguntas, conduzidas em três fases (caracterização demográfica, percepção de práticas e percepção de benefícios e dificuldades), houve uma desestimulação por parte dos respondentes ao responderem as questões sobre práticas ágeis.

A Figura 4.3 mostra que as práticas listadas em primeiro lugar foram as mais votadas, em detrimento das práticas listadas por último, que receberam menos votos. Entretanto, a curva em análise, apresenta picos em valores mais altos, o que pode representar, hipoteticamente, dois possíveis comportamentos:

Um possível interesse do respondente em avaliar positivamente uma determinada prática, que ele utiliza ou já utilizou.

Um possível interesse do respondente em avaliar negativamente uma determinada prática, que ele utiliza ou já utilizou.

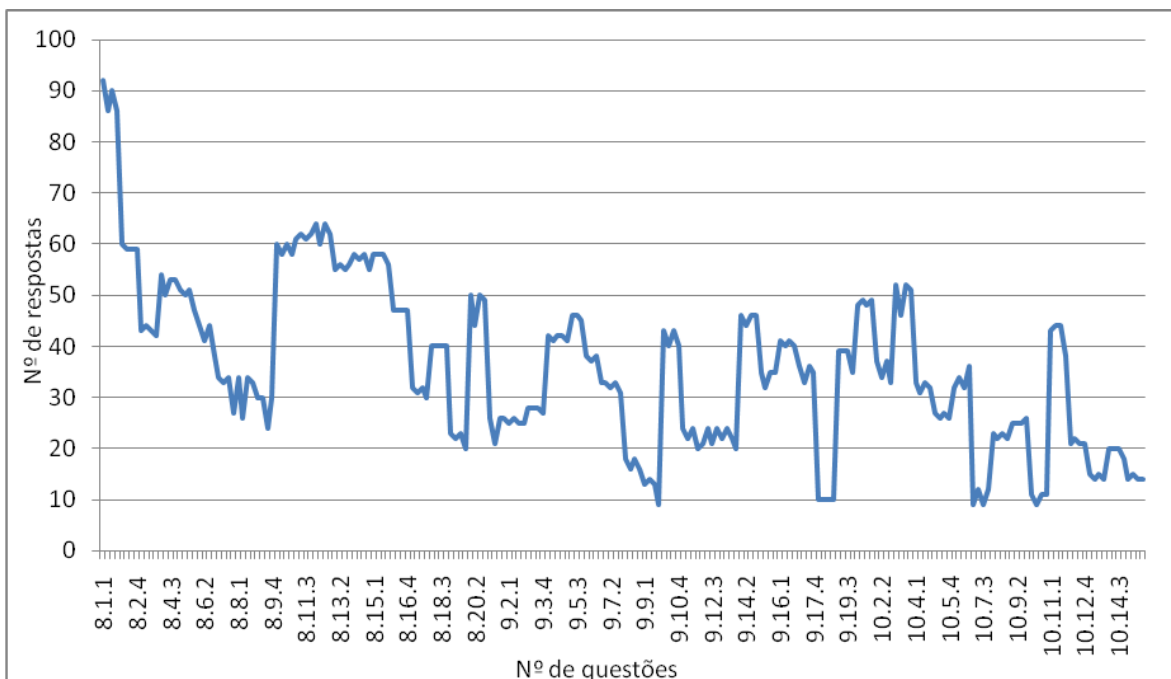


Figura 4.3 – Número respostas obtidas em cada questão de avaliação de práticas ágeis do questionário.

Uma vez que o foco do presente estudo é avaliar a percepção de valor do praticante em metodologias ágeis, o estudo das hipóteses sugeridas pode ser executado com maior

profundidade em trabalhos futuros, visto que necessita de embasamento em análises psicológicas e de comportamento humano.

Portanto, pode-se concluir que apesar do apoio estatístico e dos fundamentos técnicos utilizados para a elaboração do questionário, o mesmo precisa ser aprimorado para atender de forma mais eficiente, a necessidade do usuário e os objetivos da pesquisa.

### 4.3. Análise demográfica da amostra

A pesquisa obteve um total de 253 respondentes de países do mundo, como representada na Figura 4.4, sendo 214 respondentes do Brasil, 21 dos Estados Unidos, 2 da Bélgica e 1 respondente para cada um dos respectivos países: Alemanha, Argentina, Austrália, Bulgária, Canadá, Colômbia, Dinamarca, Espanha, França, Índia, Itália, Luxemburgo, Polônia, Suíça, Sérvia e Venezuela.

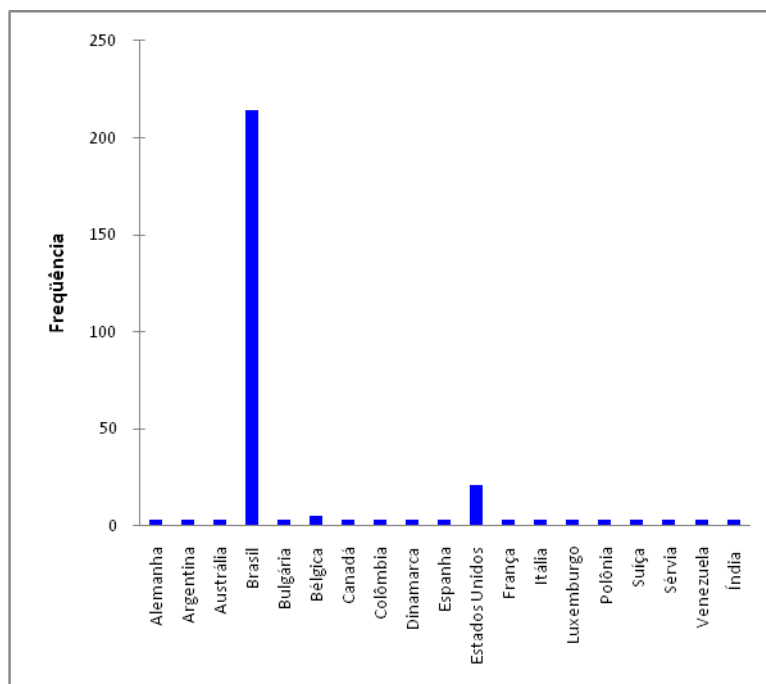


Figura 4.4 – Países respondentes da pesquisa

Quanto à descrição dos cargos ocupados na empresa, representados na Figura 4.5, 43 respondentes são Analistas de Sistema, representando 17% da amostra. Em seguida, 42 respondentes afirmam serem Gerentes de Projeto, representando 16,6% da amostra. Os

programadores representam 11% da amostra, com 28 respondentes. Os Analistas de TI representam 6,3%, com 16 respondentes. Em contrapartida, caso o respondente ocupasse algum cargo não citado na listagem, este era especificado na categoria Outros, que representou 14,2% da amostra, com 36 respondentes.

Os outros cargos listados (Administrador de banco de dados, administrador de redes, analista de requisitos, analista de teste, analista de negócio, analista de suporte, arquiteto de software, CIO, CEO, gerente de teste, líder técnico, líder de projeto, testador e professor/pesquisador) obtiveram menos de 6% da contagem de respostas.

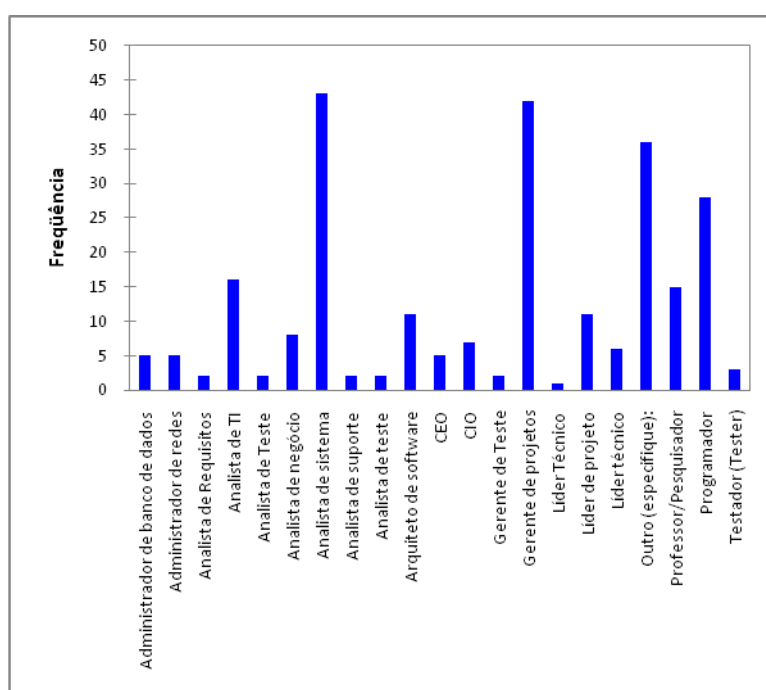


Figura 4.5 – Cargo dos respondentes da pesquisa

A Figura 4.6 mostra os dados em relação ao tamanho da empresa, 120 respondentes afirmam trabalharem em empresa de grande porte (acima de 200 funcionários). As empresas de médio porte (de 50 a 200 funcionários) obtiveram 50 respostas. Em seguida, 43 respondentes caracterizaram sua empresa como de porte micro (1 a 9 funcionários) e os outros 40 como empresa de pequeno porte ( de 10 a 49 funcionários).

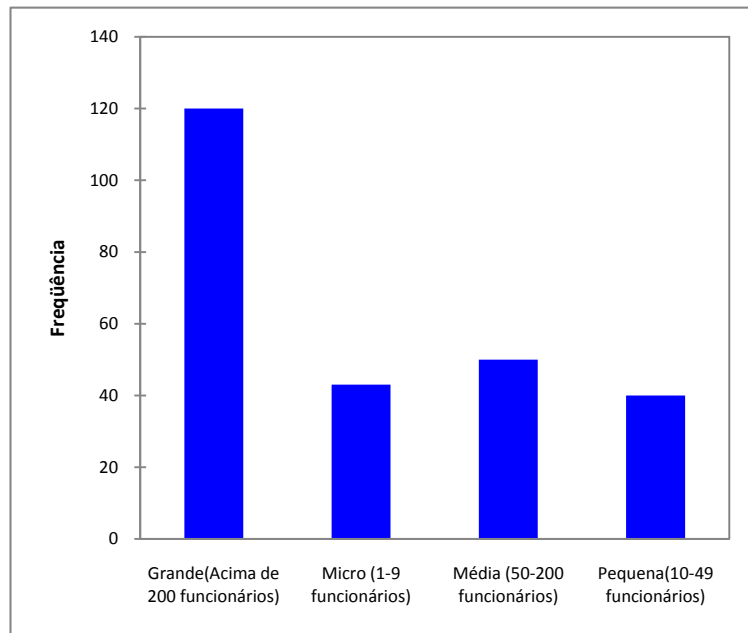


Figura 4.6 – Característica da empresa

Na Figura 4.7, em relação à utilização de metodologias ágeis pela empresa, 145 responderam que não utilizam metodologias ágeis para desenvolver software e 109 responderam que utilizam metodologias ágeis.

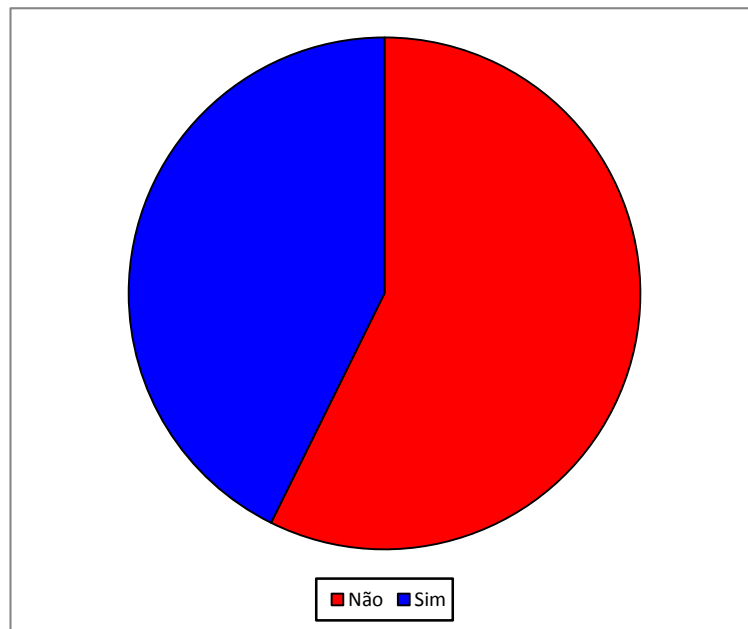


Figura 4.7 – Utilização de metodologias ágeis pelas empresas

Quanto ao estágio de adoção em metodologias ágeis, representado na Figura 4.8, 34% responderam que a utilização de metodologias ágeis em sua empresa está bem



definida, mas sem medições formais de desempenho das práticas em relação aos critérios de controle de projetos como custo, qualidade, escopo e prazo.

Cerca de 24% dos respondentes afirmaram que o processo de metodologias ágeis está parcialmente implementado em sua empresa. A categoria definido com medições formais e melhoria contínua obteve 23% das respostas. A categoria Inicial/ad hoc obteve 19% das respostas.

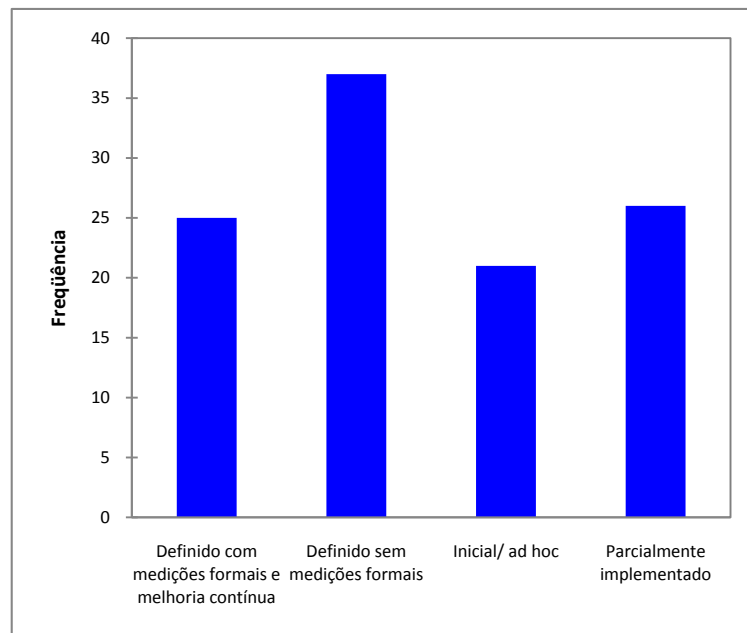


Figura 4.8 – Estágio de adoção de metodologias ágeis na empresa

Uma vez que o objetivo do trabalho é a avaliação da percepção de valor dos respondentes em relação às principais práticas ágeis utilizadas, foi utilizada apenas parte da amostra obtida referentes aos utilizadores de metodologias ágeis. Essa amostra corresponde a 109 respondentes, cujas informações obtidas foram analisadas através de técnicas estatísticas.

## 4.4. Análises Estatísticas

Nesta pesquisa em questão, o objetivo desta etapa consistiu em tratar dados coletados, uma vez que a avaliação da percepção de valor de todas as práticas ágeis listadas era optativa.

O Quadro 4.1 mostra as práticas avaliadas e os dados válidos considerados para análise advindos do processo de tratamento das variáveis.

Práticas	Nº respostas (custo)	Nº respostas (qualidade)	Nº respostas (prazo)	Nº respostas (escopo)
Equipes pequenas	92	86	90	86
Equipes multifuncionais	60	59	59	59
Programador solo	43	44	43	42
Representante do cliente	54	50	53	53
Facilitador do time	51	50	51	47
Programador-líder	44	41	44	38
Equipes de funcionalidades	34	33	34	27
Várias equipes trabalhando em paralelo	34	26	34	33
Pequenos times em grandes equipes	24	30	24	30
Reunião diária	60	58	60	58
Reuniões em pé	61	62	61	62
Reunião de planejamento da iteração	64	60	64	62
Retrospectiva da iteração	55	56	55	56
Lista de prioridades do produto	58	57	58	55
Cliente participativo	58	58	58	56
Casos de uso	47	47	47	47
Uso de cenários	32	31	32	30
Documento de visão	40	40	40	40
Documento evocativo	23	22	23	20
<i>User story</i>	50	44	50	49
Arquitetura de soluções potenciais	26	21	26	26
Modelo de domínio	25	26	25	25
Estudo do objetivo de negócio	28	28	28	27
Diagramas UML	42	41	42	42
Refatoração	46	41	46	45
Programação em pares	38	37	38	36
Propriedade coletiva do código	33	32	33	31
Triagem de erros	43	40	43	40
Gestão da configuração	24	22	24	20
Desenvolva depois teste	21	24	21	24
Teste primeiro depois desenvolva	22	24	22	20
Testes funcionais	46	44	46	46
Desenvolvimento baseado em teste	35	35	35	35
Testes unitários	41	40	41	40
Automação de testes	34	33	36	35
Integração contínua	39	39	39	35
Iterações fixas	48	49	48	49
Gráficos de andamento	34	34	37	33
<i>Kanban</i>	52	46	52	51
Jogos de planejamento	33	31	33	32
Projeto <i>Velocity</i>	27	26	27	26
Estimativa por funcionalidade	32	34	32	34
Relatório de progresso	23	22	23	22

Lista de verificação	25	25	25	26
Lançamento de versão	43	44	44	38
Técnico ágil presente	21	22	21	21
Análise em grupo com foco na visão do cliente	20	20	20	18

Quadro 4.1 – Práticas avaliadas pelo critério custo, qualidade, prazo e escopo.

Devido à aleatoriedade da amostra do trabalho, a etapa de preparação de dados consistiu na seleção e tratamento dos dados dos respondentes, eliminação dos *Outliers*, que são as avaliações consideradas fora do padrão de valores obtidos (pontos extremos de máximo e mínimo), e a separação de práticas a serem analisadas através do número de respondentes.

Nesse sentido, a percepção de práticas ágeis foi avaliada sob dois critérios de análise: representativa e pouco representativa.

As práticas classificadas para a análise representativa são as práticas que obtiveram valores maiores que 35 ( $n > 35$ ) em número de respostas. Nesse contexto, os resultados obtidos para esse grupo de práticas, poderão representar um comportamento mais aproximado de validação estatística, dando subsídios para a proposta de um modelo de aprimoramento do desenvolvimento e gerenciamento de práticas ágeis baseadas nas percepções dos respondentes.

As práticas classificadas para a análise pouco-representativa são as práticas que obtiveram valores entre 20 e 35 ( $20 \leq n < 35$ ) em número de respostas. Para esse grupo de práticas, os resultados podem ser considerados hipóteses a serem verificadas em trabalhos futuros, uma vez que necessitam de um número maior de respondentes para explorar seu comportamento na amostra.

As práticas que obtiveram um número de respostas  $n < 20$  foram excluídas da análise por se classificar como amostra não representativa.

A primeira etapa da análise consistiu em verificar se a amostra selecionada estava apta para ser executada utilizando a técnica de Análise Fatorial. A partir desta técnica, é explicada a variabilidade nas respostas do questionário que deram origem aos fatores descobertos.

A segunda etapa da análise estatística consistiu na verificação de semelhanças e características de percepção comuns entre um conjunto de categorias de respondentes, utilizando a técnica de Análise de *Clusters*.

Os resultados de ambas as análises são demonstrados sob a perspectiva de quatro critérios de desempenho de projetos: custo, qualidade, prazo e escopo.

### 4.4.1 Análise Fatorial

Para identificar quais práticas estão correlacionadas e representam maior valor na percepção dos respondentes, foi utilizada no trabalho a Análise Fatorial.

O objetivo da análise fatorial é reduzir o número de variáveis originais, ou seja, as práticas, em dimensões que podem ser explicadas através da existência de correlação entre as variáveis originais. Essa nova dimensão é chamada de Fator.

Os fatores encontrados são definidos pelo valor de seus índices, cuja consideração é dependente do tamanho da amostra. Para esta pesquisa, as práticas que possuem cargas fatoriais de 0,55 ou mais são consideradas significantes, uma vez que o número da amostra é 109 respondentes (HAIR *et al.*, 2009).

#### 4.4.1.1 Custo

- **Amostra representativa**

Com objetivo de verificar a validação da análise da variável prática com  $n > 35$ , para o critério custo, foram extraídos indicadores, mostrando que a matriz fatorial construída pode ser considerada.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin</i>		0,870
Teste de esfericidade	Aproximação $Q^2$	26,494
<i>Bartlett</i>	DF	300
	Sig.	0,000

Quadro 4.2 – Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem representativa do critério custo

O Quadro 4.2 resume as informações de validação da análise fatorial. O primeiro índice é chamado de KMO, e define a qualidade da amostra para a Análise Fatorial, onde o  $KMO = 0,870$ , aproximadamente 0,9, possui uma recomendação excelente. O teste de *Bartlett* responde se as variáveis selecionadas são correlacionadas ou não. Neste caso, o teste de *Bartlett* apresenta um valor de significância  $p < 0,0001$  e portando concluí-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados no Quadro 4.3, a Análise Fatorial mostrou uma estrutura de seis fatores que explicam 64,9 % da variância de respostas do questionário, considerando autovalores superiores ou iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).

Fatores	Autovalores iniciais		
	Total	% de Variância	Cumulativa %
1	9,842	37,854	37,854
2	1,714	6,592	44,446
3	1,604	6,168	50,614
4	1,431	5,505	56,119
5	1,176	4,523	60,641
6	1,121	4,310	64,951

Quadro 4.3 – Porcentagem da variância explicada para a análise representativa para o critério Custo

A Tabela 4.1 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.1 – Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério custo

	F1	F2	F3	F4	F5	F6
Equipes pequenas	0,156	<b>0,594</b>	0,049	<b>0,560</b>	-0,078	0,021
Equipes multifuncionais	0,222	<b>0,661</b>	0,137	0,483	0,022	0,066
Programador solo	-0,037	-0,003	-0,027	0,044	0,005	<b>0,815</b>
Representante do cliente	0,079	<b>0,737</b>	0,301	0,030	0,209	0,221
Facilitador do time	0,486	<b>0,640</b>	0,012	0,070	0,168	0,154
Programador-Líder	0,254	<b>0,564</b>	0,224	0,090	-0,030	0,121
Reunião Diária	<b>0,747</b>	0,131	0,068	0,277	0,103	0,050
Reunião em pé	<b>0,721</b>	0,154	0,213	-0,007	0,025	0,074
Reunião de planejamento da iteração	<b>0,726</b>	0,239	0,259	0,171	0,051	0,079
Retrospectiva da iteração	<b>0,726</b>	0,222	0,276	0,230	-0,036	0,109
Lista de prioridades do produto	<b>0,557</b>	0,438	0,163	0,129	0,262	0,101
Cliente participativo	0,160	<b>0,587</b>	0,176	0,272	0,151	0,263
Casos de uso	0,078	0,022	0,062	0,024	<b>0,761</b>	0,366
Documento de visão	0,508	0,036	0,139	0,003	0,390	0,363
<i>User story</i>	<b>0,584</b>	0,365	0,190	0,272	0,173	0,288
Diagramas UML	0,022	0,159	-0,119	0,132	<b>0,683</b>	0,268
Refatoração	0,173	0,125	<b>0,628</b>	0,453	-0,061	0,149
Programação em pares	0,232	0,137	0,541	0,187	0,453	0,042
Triagem de erros	0,147	0,054	0,210	<b>0,607</b>	0,234	0,262
Testes funcionais	0,264	<b>0,558</b>	0,195	-0,004	0,460	0,197
Desenvolvimento por casos de teste	0,372	0,300	<b>0,693</b>	0,125	-0,062	0,011
Testes unitários	0,269	0,497	0,564	-0,152	0,038	0,291
Integração contínua	0,400	0,239	<b>0,635</b>	0,140	0,086	0,124
Iterações fixas	<b>0,579</b>	0,161	0,424	0,130	0,106	0,145
<i>Kanban</i>	0,406	0,193	0,090	<b>0,619</b>	0,085	0,150
Lançamento frequente de versões	0,157	0,191	0,357	0,450	0,429	0,045

A partir do resultado encontrado na análise fatorial, foi possível a extração de seis fatores:

**Fator 1 – Interação da equipe:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas ágeis relacionadas às reuniões em que a equipe planeja, controla seu progresso no projeto e revisa seus resultados no ciclo de desenvolvimento. Os maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para práticas que compõem e caracterizam o fator, ou seja, que o estabelecimento de um ciclo fixo para o cumprimento das tarefas, com especificação de tarefas através do *user story* e interação da equipe, em relação ao critério custo agrega valor ao projeto com alto desempenho.

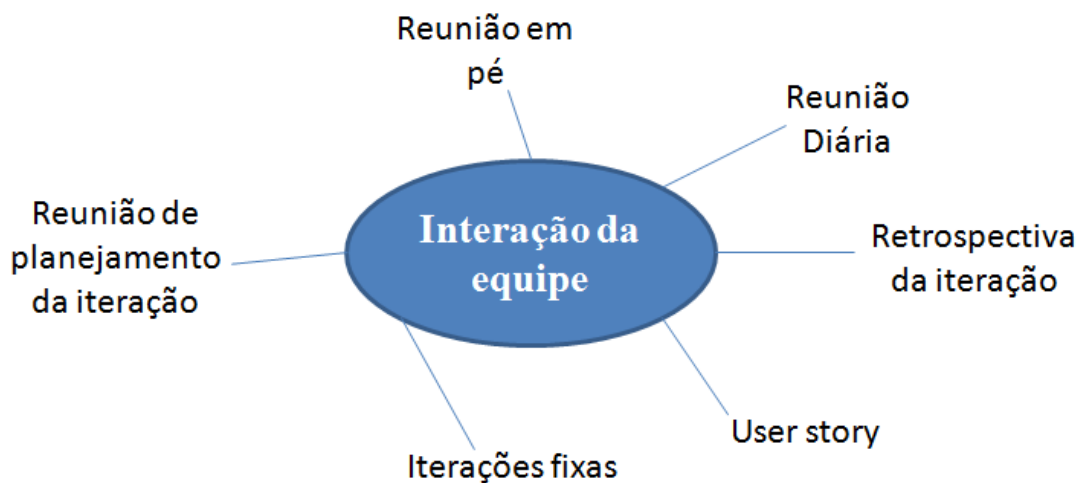


Figura 4.9 – Representação gráfica do fator Interação da equipe

**Fator 2 – Liderança e custos:** este segundo fator apresenta pesos fatoriais elevados significativos em práticas ágeis relacionadas à formação e liderança em equipes. Os maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para práticas em que as pessoas do projeto ocupam papéis de liderança de equipes, se responsabilizando pelas atividades de testes funcionais junto ao cliente. Essa percepção é observada em sob a ótica do critério custo, agregando valor ao projeto com alto desempenho.

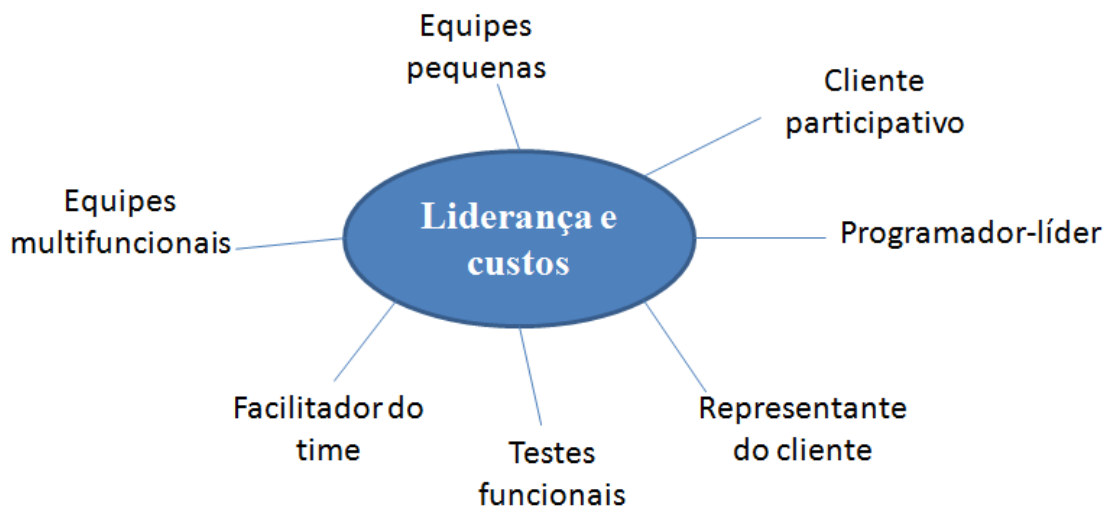


Figura 4.10 – Representação gráfica do fator Liderança e custos

**Fator 3 – Desenvolvimento:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas ágeis relacionadas à etapa de desenvolvimento. Os maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para integração de código, testes e refatoração, ou seja, em que cada vez que a equipe submete seu código a um repositório de arquivos do projeto, um *build* funcional é construído automaticamente e os testes começam a ser executados em um curto tempo de resposta, evitando falhas do sistema. Essa percepção é observada em sob a ótica do critério custo, agregando valor ao projeto com alto desempenho.



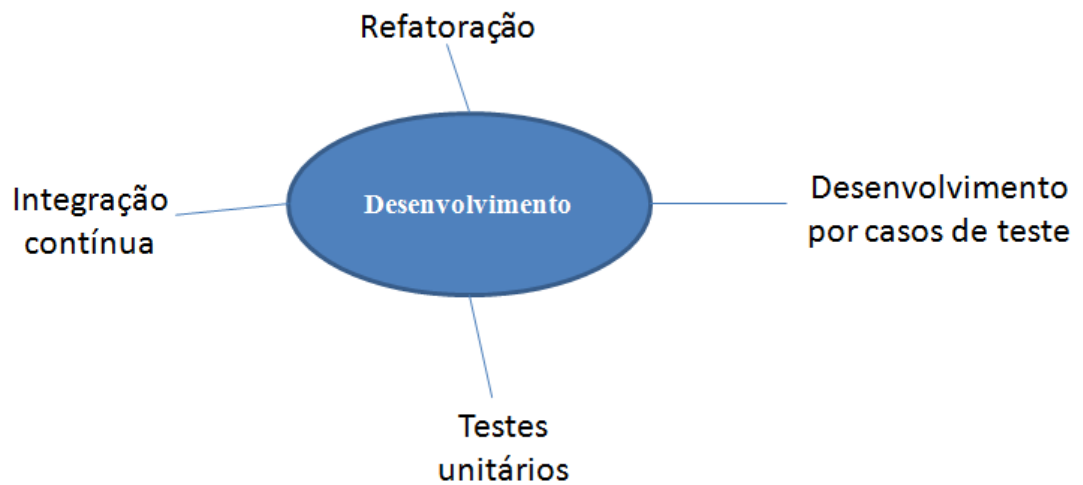


Figura 4.11 – Representação gráfica do fator Desenvolvimento

**Fator 4 – Tempo de resposta (*Just-in-time*):** este quarto fator apresenta pesos fatoriais elevados significativos em práticas ágeis relacionadas ao *feedback* instantâneo. Os maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para um cenário de práticas em que as pessoas do projeto utilizam o *Kanban* como monitoramento para a implementação de soluções para os resultados dos testes de aceitação. Essa percepção é observada em sob a ótica do critério custo, agregando valor ao projeto com alto desempenho.

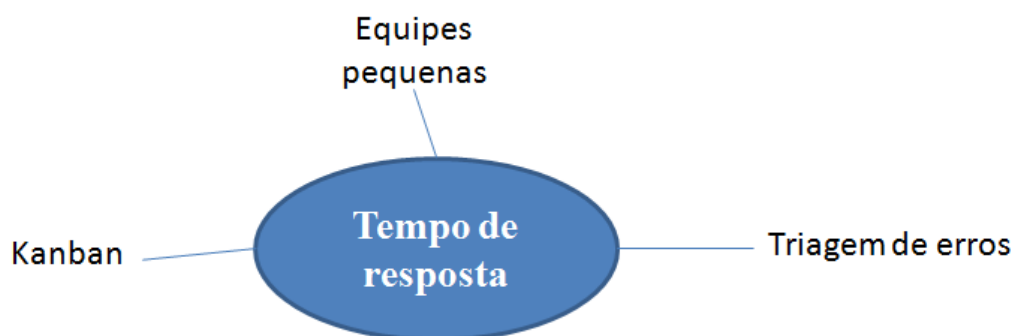


Figura 4.12 – Representação gráfica do fator Tempo de resposta

**Fator 5 – Modelagem de processos:** este quinto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à etapa de modelagem dos casos de uso e classes a serem implementadas no software. Os maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para práticas de modelagem de processos, realizadas na etapa de planejamento, para construção do Modelo de domínio. Essa percepção é observada em sob a ótica do critério custo, agregando valor ao projeto com alto desempenho.

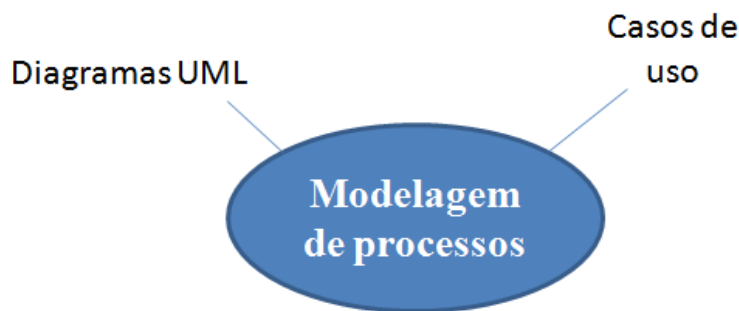


Figura 4.13 – Representação gráfica do fator Modelagem de processos

**Fator 6 – Desenvolvimento solo:** este sexto fator apresenta pesos fatoriais elevados significativos especificamente na prática Programador-solo. O Programador-solo trabalharia como um único responsável pelas tarefas de modelagem, teste e validação do software. Os maiores pesos fatoriais justificam a percepção de valor “Muito alta” dos respondentes para esse papel em relação ao critério custo, agregando valor ao projeto com alto desempenho.



Figura 4.14 – Representação gráfica do fator Desenvolvimento solo

- **Amostra pouco representativa**

Para tanto, foram extraídos indicadores provenientes da análise, mostrando que a matriz fatorial construída pode ser considerada.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin</i>	0,770
Teste de esfericidade <i>Bartlett</i>	Aproximação Q <sup>2</sup>
	df
	Sig.
	-23,425
	190
	0,000

Quadro 4.4 – Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem pouco representativa do critério Custo

O Quadro 4.4 resume as informações de validação da análise fatorial. O índice KMO = 0,770, aproximadamente 0,8, diz que a análise possui uma recomendação ótima. O teste de *Bartlett* indicou um valor de significância  $p < 0,0001$  e portanto concluí-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados, pode-ser observar no Quadro 4.5 que a Análise Fatorial mostrou uma estrutura de seis fatores que explicam 67,1% da variância

de respostas do questionário, considerando autovalores superiores ou iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).

Fator	Autovalores iniciais		
	Total	% of Variância	Cumulativa %
1	6,319	31,596	31,596
2	2,153	10,763	42,359
3	1,473	7,366	49,726
4	1,246	6,230	55,956
5	1,169	5,844	61,800
6	1,061	5,303	67,102

Quadro 4.5 – Porcentagem da variância explicada para a análise pouco representativa para o critério Custo.

A Tabela 4.2 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.2 – Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério custo

	F1	F2	F3	F4	F5	F6
Equipes de funcionalidades	0,053	-0,070	<b>0,881</b>	0,063	0,001	0,084
Várias equipes trabalhando em paralelo	-0,252	0,246	0,470	<b>0,446</b>	0,242	0,157
Pequenos times me equipes grandes	-0,204	0,045	-0,014	0,184	0,483	<b>0,566</b>
Uso de cenários	0,204	0,135	0,223	<b>0,758</b>	0,094	-0,150
Documento evocativo	0,042	0,036	<b>0,840</b>	0,050	-0,053	-0,071
Arquitetura de soluções potenciais	<b>0,770</b>	0,276	-0,076	0,119	0,082	0,039
Modelo de domínio	<b>0,692</b>	0,096	0,191	0,006	0,018	-0,155
Estudo do objetivo do negócio	0,267	0,169	-0,007	<b>0,600</b>	0,169	0,128
Propriedade coletiva do código	0,468	0,053	-0,057	<b>0,672</b>	-0,051	0,030
Desenvolva depois teste	0,046	-0,016	0,042	-0,077	-0,145	<b>0,861</b>
Teste primeiro depois desenvolva	<b>0,750</b>	0,099	-0,019	0,263	-0,016	0,157
Automação de testes	<b>0,644</b>	0,363	-0,245	0,285	0,128	0,018
Gráficos de andamento	0,370	<b>0,606</b>	-0,103	0,194	0,302	-0,155
Jogos de planejamento	<b>0,641</b>	0,481	0,012	0,156	-0,009	0,026
Projeto <i>Velocity</i>	<b>0,690</b>	0,441	-0,061	0,046	0,197	0,056
Estimativa por funcionalidade	0,133	0,185	-0,051	0,110	<b>0,795</b>	-0,047
Relatório de progresso	<b>0,552</b>	<b>0,560</b>	0,204	-0,055	0,209	0,042
Lista de verificação	0,542	-0,162	0,290	0,075	0,466	-0,136
Técnico ágil presente	0,179	<b>0,727</b>	0,040	0,251	-0,085	-0,105
Análise em grupo com foco no cliente	0,213	<b>0,755</b>	-0,013	0,067	0,112	0,160

A partir do resultado encontrado na análise fatorial, foi possível a extração de três fatores:

**Fator 1 – Modelagem e estimativa de processos:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas classificadas como práticas de modelagem de projetos e testes das soluções propostas. As práticas que obtiveram maiores pesos fatoriais hipoteticamente justificam a percepção de valor dos respondentes para com a etapa de modelagem em relação ao critério custo agregam alto valor ao projeto neste critério de desempenho.

**Fator 2 – Monitoramento com técnico ágil presente:** este segundo fator apresenta pesos fatoriais elevados significativos em práticas classificadas como práticas de análise e revisão de etapas. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor dos respondentes para com a etapa de análise e revisão do produto intermediário ao final de cada fase do projeto em relação ao critério custo agrega alto valor ao projeto neste critério de desempenho.

**Fator 3 – Planejamento por funcionalidades:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas com características peculiares para formação de equipes e documentação de projetos: que estas estejam alinhadas às funcionalidades, ou seja, a criação de equipes e documentação específicas para entregar uma determinada funcionalidade. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor dos respondentes para com esta etapa em relação ao critério custo agrega alto valor ao projeto.

#### **4.4.1.2 Qualidade**

- **Amostra representativa**

Para a validação da análise para o critério qualidade, foram extraídos indicadores, mostrando que a matriz fatorial construída pode ser considerada.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin</i>		0,680
Teste de esfericidade	Aproximação Q <sup>2</sup>	-1,475
<i>Bartlett</i>	df	300
	Sig.	0,000

Quadro 4.6 – Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem representativa do critério Qualidade

As informações descritas no Quadro 4.6 indicam que o índice KMO = 0,680, aproximadamente 0,7, valida a análise com uma recomendação boa. O teste de *Bartlett* apresenta um valor de significância  $p < 0,0001$  e portanto concluí-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados, a Análise Fatorial, representada no Quadro 4.7, mostrou uma estrutura de seis fatores que explicam 64,3%, aproximadamente 64% da variância de respostas do questionário, considerando autovalores superiores ou iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).

Fatores	Autovalores iniciais		
	Total	% da Variância	Cumulativo %
1	5,542	39,116	39,116
2	2,234	6,232	45,347
3	2,003	5,314	50,662
4	1,558	4,678	55,340
5	1,329	4,606	59,946
6	1,097	4,389	64,335

Quadro 4.7 – Porcentagem da variância explicada para a análise representativa para o critério Qualidade.

A Tabela 4.3 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.3 – Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério qualidade

	F1	F2	F3	F4	F5	F6
Equipes pequenas	0,168	0,292	0,099	-0,053	0,146	0,278
Equipes multifuncionais	0,015	0,378	-0,111	0,053	0,195	<b>0,557</b>
Programador solo	-0,058	-0,156	0,026	0,040	-0,076	0,061
Representante do cliente	0,291	0,224	0,157	0,176	0,028	<b>0,626</b>
Facilitador do time	0,226	0,104	0,494	0,259	0,264	0,182
Programador-Líder	0,096	0,163	0,203	0,253	0,140	-0,003
Reunião Diária	0,207	0,174	0,004	-0,001	-0,103	-0,068
Reunião em pé	0,286	-0,294	0,139	-0,120	-0,037	0,374
Reunião de planejamento da iteração	<b>0,664</b>	0,153	-0,036	0,051	0,081	-0,028
Retrospectiva da iteração	<b>0,784</b>	0,108	-0,102	0,114	0,272	0,147
Lista de prioridades do produto	<b>0,587</b>	0,239	0,244	-0,040	0,032	0,071
Cliente participativo	0,291	<b>0,692</b>	-0,049	0,245	-0,044	0,070
Casos de uso	-0,076	-0,062	<b>0,636</b>	0,024	0,395	0,235
Documento de visão	0,279	0,181	0,141	-0,098	<b>0,705</b>	-0,028
<i>User story</i>	0,320	<b>0,623</b>	0,029	0,065	0,404	0,317
Diagramas UML	0,032	0,005	<b>0,847</b>	-0,047	-0,021	-0,047
Refatoração	0,196	0,249	-0,422	0,008	-0,115	0,291
Programação em pares	-0,027	0,119	-0,037	0,468	0,118	0,213
Triagem de erros	0,285	0,051	0,119	<b>0,783</b>	-0,106	0,026
Testes funcionais	0,063	<b>0,576</b>	0,556	0,165	-0,116	-0,040
Desenvolvimento por casos de teste	0,051	0,287	-0,032	<b>0,725</b>	0,035	0,170
Testes unitários	0,013	-0,031	-0,061	0,452	-0,185	<b>0,587</b>
Integração contínua	<b>0,663</b>	0,068	0,010	0,291	-0,184	0,308
Iterações fixas	0,516	0,098	0,191	0,195	0,131	0,321
<i>Kanban</i>	0,039	0,466	0,002	0,125	-0,012	0,155
Lançamento frequente de versões	0,011	0,031	0,157	0,022	0,105	0,149

A partir do resultado encontrado na análise fatorial, foi possível a extração de seis fatores:

**Fator 1 – Backlog com integração contínua:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas relacionados à etapa planejamento das funcionalidades do projeto. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para integração contínua das funcionalidades priorizadas na reunião de planejamento da iteração e avaliadas na reunião de retrospectiva. Essa percepção é observada sob a ótica do critério qualidade, agregando alto valor ao projeto.

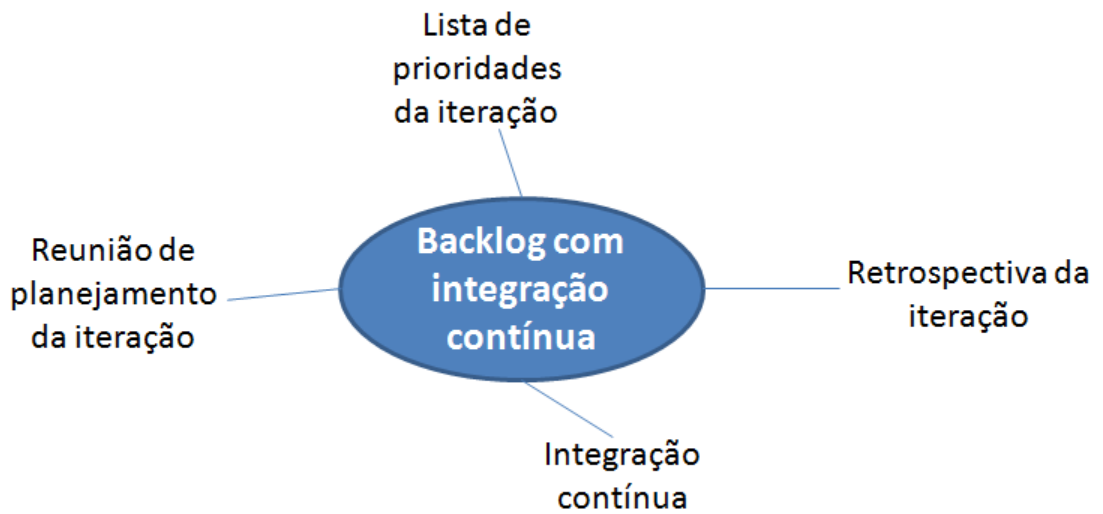


Figura 4.15 – Representação gráfica do fator Backlog com integração contínua

**Fator 2 – Análise de requisitos:** este segundo fator apresenta pesos fatoriais elevados significativos em práticas relacionados à etapa análise, seleção e execução de testes das funcionalidades especificadas no documento de requisitos do software com a participação do cliente. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para o papel participativo do cliente, *user story* e testes funcionais. Essa percepção é observada sob a ótica do critério qualidade, agregando alto valor ao projeto.



Figura 4.16 – Representação gráfica do fator Análise de Requisitos



**Fator 3 – Modelagem do processo de teste:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas a fase de modelagem de testes. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para com as práticas de geração de caso de uso, diagramas UML e a execução de testes funcionais. Essa percepção é observada sob a ótica do critério qualidade, agregando alto valor ao projeto.

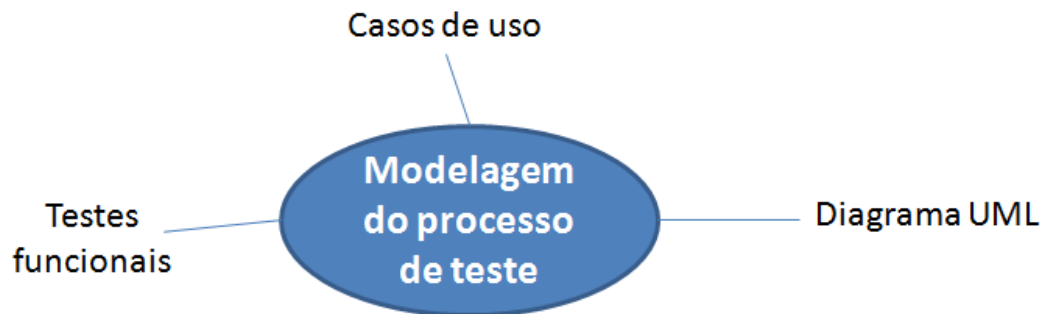


Figura 4.17 – Representação gráfica do fator Modelagem do processo de teste

**Fator 4 – Prevenção de erros com casos de teste:** este quarto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à utilização de casos de teste para a correção de erros oriundos dos testes de aceitação. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para com as práticas de triagem de erros dos testes de aceitação e desenvolvimento por casos de teste em relação ao critério qualidade agrega valor ao projeto.

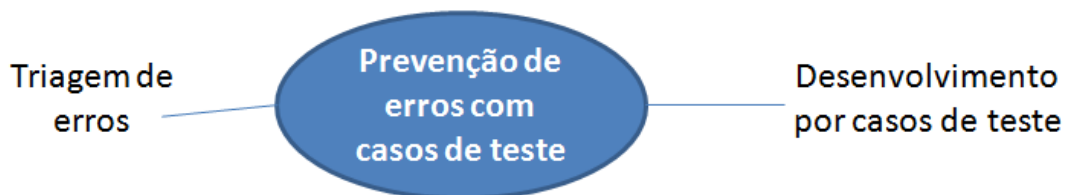


Figura 4.18 – Representação gráfica do fator Prevenção de erros com casos de teste

**Fator 5 – Documento de visão:** o quinto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à utilização de um documento de visão como

artefato que relata perspectivas do sistema, processo antecedente à análise do modelo de domínio. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para com a prática de construção de um documento de visão em relação ao critério qualidade agrega alto valor ao projeto.



Figura 4.19 – Representação gráfica do fator Documento de visão

**Fator 6 – Equipes de teste multifuncionais:** este sexto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à formação de uma equipe multifuncional para testes. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para com as práticas equipes multifuncionais, representante do cliente e testes unitários em relação ao critério qualidade agrega alto valor ao projeto.

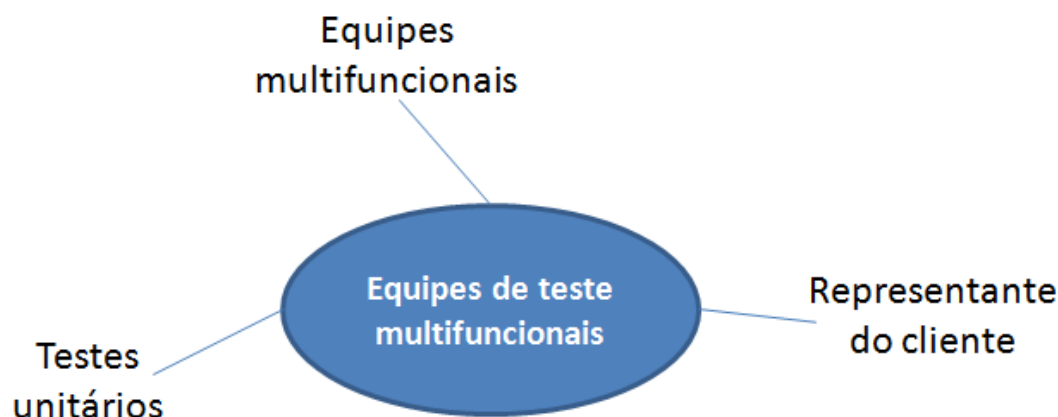


Figura 4.20 – Representação gráfica do fator Equipes de teste multifuncionais

- **Amostra pouco representativa**

A primeira etapa da análise exploratória consistiu em verificar se a amostra selecionada estava apta para ser executada utilizando a técnica de Análise Fatorial.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin</i>		0,708
Teste de esfericidade <i>Bartlett</i>	Aproximação $Q^2$	-38,750
	df	171
	Sig.	0,000

Quadro 4.8 - Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem pouco representativa do critério Qualidade

O Quadro 4.8 resume as informações de validação da análise fatorial. O índice  $KMO = 0,708$ , aproximadamente 0,7, possui uma recomendação boa. O teste de *Bartlett* para essa análise apresenta um valor de significância  $p < 0,0001$  e portanto concluí-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados, pode-ser observar no Quadro 4.9 que a análise mostrou uma estrutura de cinco fatores que explicam 59,893%, aproximadamente 60% da variância de respostas do questionário, considerando autovalores superiores ou

iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).

Fator	Autovalores iniciais		
	Total	% da Variância	% Cumulativa
1	5,074	26,703	26,703
2	2,189	11,523	38,226
3	1,557	8,192	46,418
4	1,384	7,283	53,702
5	1,176	6,191	59,893

Quadro 4.9 – Porcentagem da variância explicada para a análise pouco representativa para o critério Qualidade.

A Tabela 4.4 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.4 – Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério qualidade

	F1	F2	F3	F4	F5
Equipes de funcionalidades	0,137	0,149	0,049	<b>0,766</b>	-0,040
Várias equipes trabalhando em paralelo	-0,063	-0,005	0,087	0,397	0,587
Pequenos times em equipes grandes	0,129	0,451	0,124	<b>0,619</b>	0,118
Uso de Cenários	0,074	0,472	0,048	0,122	0,263
Documento evocativo	0,109	<b>0,857</b>	0,188	0,308	0,121
Arquitetura de soluções potenciais	0,080	0,069	0,006	-0,029	0,264
Modelo de domínio	<b>0,640</b>	-0,120	0,080	0,078	0,073
Estudo do objetivo de negócio	0,313	-0,031	0,025	-0,065	0,286
Propriedade coletiva do código	0,124	-0,121	0,039	0,083	0,544
Desenvolva depois teste	0,090	<b>0,637</b>	0,337	0,002	-0,143
Teste primeiro depois desenvolva	0,259	0,017	<b>0,609</b>	-0,006	0,068
Automação de testes	-0,025	-0,066	0,367	0,000	0,372
Gráficos de andamento	0,043	0,069	0,259	-0,059	<b>0,546</b>
Jogos de planejamento	<b>0,748</b>	0,077	0,057	0,103	0,318
Projeto <i>Velocity</i>	<b>0,657</b>	0,042	0,182	0,144	0,277
Estimativa por funcionalidade	0,176	0,026	0,063	0,110	0,375
Relatório de progresso	0,438	0,257	<b>0,566</b>	0,297	0,209
Lista de verificação	0,277	0,161	0,404	0,232	0,169
Técnico ágil presente	<b>0,604</b>	0,101	0,349	0,120	0,060
Análise em grupo com foco no cliente	<b>0,611</b>	0,208	0,321	0,186	<b>0,640</b>

A partir do resultado encontrado na análise fatorial, foi possível a extração de cinco fatores:

**Fator 1 – Modelagem e estimativa de processos:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas classificadas como práticas de modelagem de projetos e testes das soluções propostas. As práticas que obtiveram maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com a etapa de modelagem em relação ao critério custo agrega alto valor ao projeto.

**Fator 2 – Documentação simples para testes:** este segundo fator apresenta pesos fatoriais elevados significativos para práticas Documento de evocativo e desenvolva depois teste, o que representa uma documentação simples, sem linguagem técnica, do processo de teste. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes em relação ao critério custo agrega alto valor ao projeto com alto desempenho neste critério.

**Fator 3 – Progresso do desenvolvimento por testes:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas ao controle de progresso do processo de teste: o relatório controla o progresso das atividades de teste, onde são indicadas as novas funcionalidades aprovadas pelo teste e as que ainda estão sendo testadas. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério custo agrega alto valor ao projeto com alto desempenho neste critério.

**Fator 4 – Grande equipe de funcionalidade:** este quarto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas com características peculiares para formação de grandes equipes: que estas estejam alinhadas às funcionalidades, ou seja, a criação de equipes e documentação específica para entregar uma determinada funcionalidade. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério custo agregam alto valor ao projeto com alto desempenho neste critério.

**Fator 5 – Monitoramento e análise de equipes paralelas:** este quinto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas o monitoramento

de atividades do projeto de uma equipe subdividida em pequenas sub-equipes trabalhando em paralelo. Porém as análises ao final do ciclo, cujo foco são as requisições do cliente, são realizadas com o grande grupo. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério custo agrega alto valor ao projeto com alto desempenho neste critério

### 4.4.1.3 Prazo

- **Amostra representativa**

Com objetivo de verificar a validação da análise para o critério prazo, foram extraídos indicadores, mostrando que a matriz fatorial construída pode ser considerada.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin</i>		0,791
Teste de esfericidade <i>Bartlett</i>	Aproximação Q <sup>2</sup>	-1,663
	df	300
	Sig.	0,000

Quadro 4.10 – Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem representativa do critério Prazo

Para fins de validação da análise, representada no Quadro 4.10, foi extraído o índice KMO = 0,791, aproximadamente 0,8, possui uma recomendação ótima. O teste de *Bartlett* mostrou um valor de significância  $p < 0,0001$  e, portanto concluí-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados, a Análise Fatorial mostrou uma estrutura de oito fatores que explicam 62,966 %, aproximadamente 63% da variância de respostas do questionário, considerando autovalores superiores ou iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).

Fator	Autovalores iniciais		
	Total	% da Variância	Cumulativa %
1	6,740	26,962	26,962
2	2,073	8,293	35,255
3	1,854	7,415	42,670
4	1,524	6,097	48,767
5	1,302	5,209	53,976
6	1,206	4,822	58,798
7	1,042	4,168	62,966

Quadro 4.11 – Porcentagem da variância explicada para a análise representativa para o critério Prazo.

A Tabela 4.5 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.5 – Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério prazo

	F1	F2	F3	F4	F5	F6	F7
Equipes pequenas	0,482	0,185	0,270	0,135	<b>-0,238</b>	-0,031	0,464
Equipes multifuncionais	0,092	0,077	<b>0,817</b>	0,046	0,045	0,010	-0,059
Programador solo	0,054	-0,016	0,094	-0,001	-0,001	0,112	<b>0,814</b>
Representante do cliente	0,143	0,326	<b>0,562</b>	0,139	0,166	0,253	0,085
Facilitador do time	0,090	0,214	0,117	0,273	-0,022	0,532	0,092
Programador-Líder	0,266	0,054	<b>0,677</b>	0,030	0,200	0,189	0,177
Reunião Diária	0,138	0,176	0,183	0,393	-0,209	0,449	-0,441
Reunião em pé	0,167	-0,384	0,273	0,195	-0,039	0,033	0,419
Reunião de planejamento da iteração	0,128	0,206	0,006	<b>0,699</b>	-0,174	0,194	0,247
Retrospectiva da iteração	0,078	-0,161	0,093	<b>0,692</b>	0,174	0,119	0,015
Lista de prioridades do produto	-0,077	-0,025	0,086	0,346	-0,024	0,164	0,034
Cliente participativo	0,222	0,196	<b>0,623</b>	0,287	0,126	0,031	-0,015
Casos de uso	0,004	0,111	0,192	0,009	<b>0,812</b>	0,070	0,012
Documento de visão <i>User story</i>	0,329	-0,032	0,066	0,434	<b>0,646</b>	-0,120	-0,073
<i>User story</i>	-0,002	0,173	0,240	<b>0,682</b>	0,143	0,008	-0,158
Diagramas UML	0,252	-0,223	0,160	-0,068	0,431	<b>0,587</b>	0,144
Refatoração	-0,009	0,410	0,009	0,371	0,335	0,373	0,068
Programação em pares	0,453	0,233	0,214	0,066	0,466	0,304	-0,184
Triagem de erros	<b>0,705</b>	0,171	-0,041	-0,025	0,128	0,159	0,022
Testes funcionais	0,466	0,318	0,179	-0,071	0,148	0,050	0,132
Desenvolvimento por casos de teste	0,093	<b>0,807</b>	0,172	-0,016	0,019	0,167	-0,142
Testes unitários	0,322	<b>0,730</b>	0,125	-0,078	0,079	0,029	0,135
Integração contínua	0,318	<b>0,649</b>	0,143	0,301	0,056	-0,079	0,115
Iterações fixas	<b>0,644</b>	0,123	0,263	0,207	0,011	-0,028	0,136
<i>Kanban</i>	<b>0,682</b>	0,057	0,187	0,131	-0,103	0,177	-0,058
Lançamento frequente de versões	<b>0,688</b>	0,112	0,147	-0,014	0,246	-0,038	0,087

A partir do resultado encontrado na análise fatorial, foi possível a extração de sete fatores:

**Fator 1 – Releases com tempo pré-fixado:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à entrega do incremento em um tempo pré-determinado. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas *Kanban*, triagem de erros, iterações fixas e lançamento frequente de versões. Essa percepção é observada sob a ótica do critério prazo, agregando valor ao projeto.



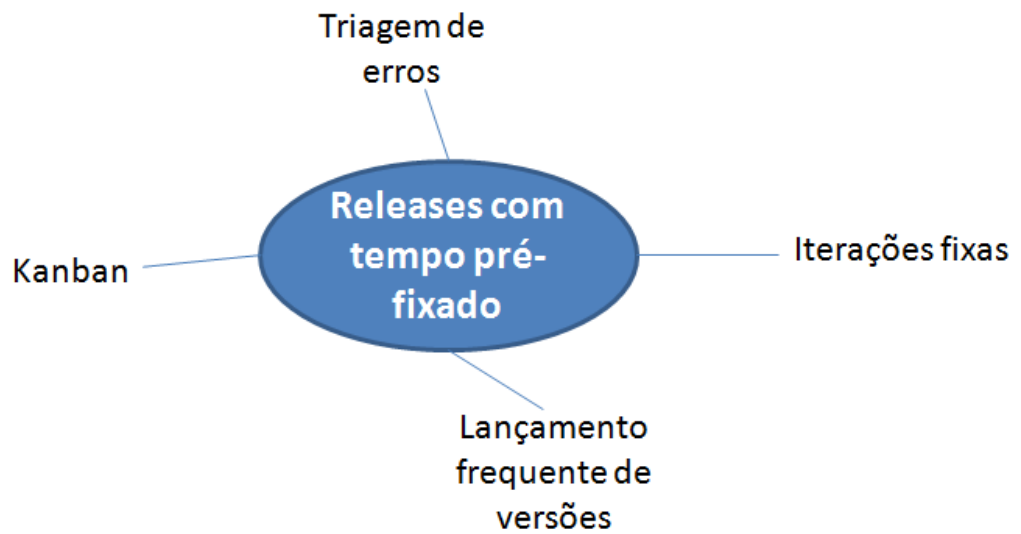


Figura 4.21 – Representação gráfica do fator Releases com tempo pré-fixado

**Fator 2 – Testes e liberação contínua:** este segundo fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à entrega aos testes e liberação de versão utilizando integração contínua. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas desenvolvimento por casos de teste, testes unitários e integração contínua. Essa percepção é observada sob a ótica do critério prazo, agregando alto valor ao projeto.

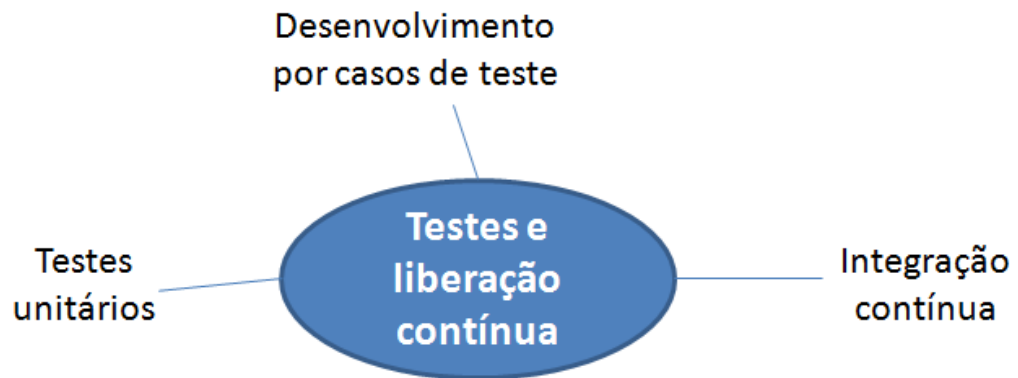


Figura 4.22 – Representação gráfica do fator Testes e liberação contínua

**Fator 3 – Equipes de liderança:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à formação de equipes multifuncionais, com posições que favorecem a liderança, altamente voltada para desempenho das pessoas do projeto e *feedback* com o cliente. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas: equipes multifuncionais, representante do cliente, programador-líder, cliente participativo. Essa percepção é observada sob a ótica do critério prazo, agregando alto valor ao projeto.

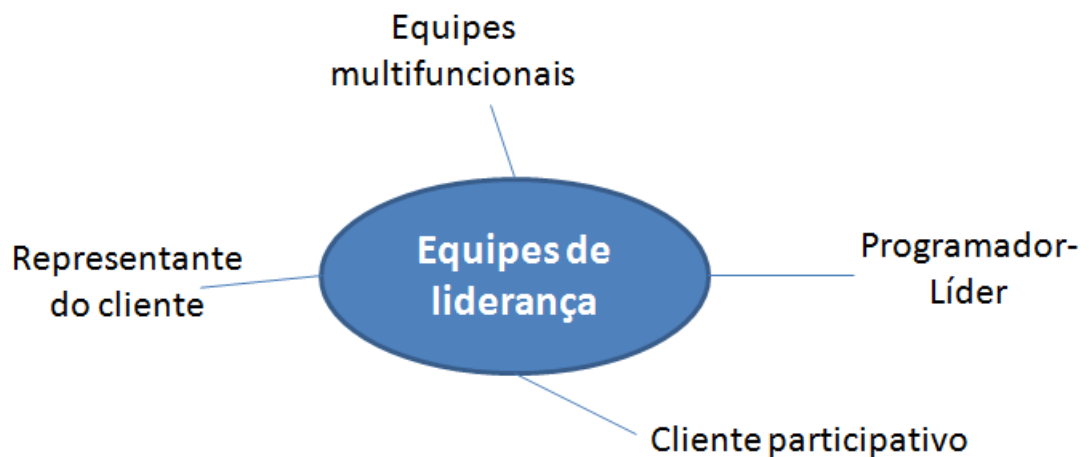


Figura 4.23 – Representação gráfica do fator Equipes de liderança

**Fator 4 – Histórias de usuário:** este quarto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à organização dos requisitos em histórias dos usuários, definidas nas reuniões de planejamento de cada *Sprint*, ou ciclo de trabalho, e revisadas na cerimônia de retrospectiva. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas *user story*, reunião de planejamento da iteração e retrospectiva da iteração. Essa percepção é observada sob a ótica do critério prazo, agregando alto valor ao projeto.



Figura 4.24 – Representação gráfica do fator Histórias de usuário

**Fator 5 – Documentação representativa:** este quinto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à elaboração de uma documentação representativa, de fácil entendimento da equipe e do cliente. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas de criação de diagramas de caso de uso e do documento de visão. Essa percepção é observada sob a ótica do critério prazo, agregando alto valor ao projeto.

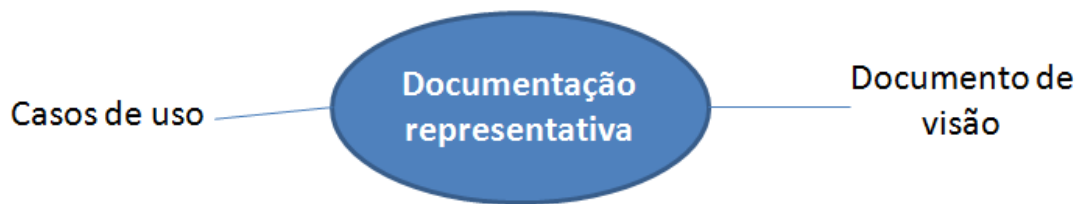


Figura 4.25 – Representação gráfica do fator Documentação representativa

**Fator 6 – Diagramação:** este sexto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à especificação utilizando diagramas UML. A prática com maior peso fatorial justifica a percepção de valor “Muito alta” dos respondentes para a prática de diagrama UML. Essa percepção é observada sob a ótica do critério prazo, agregando alto valor ao projeto.

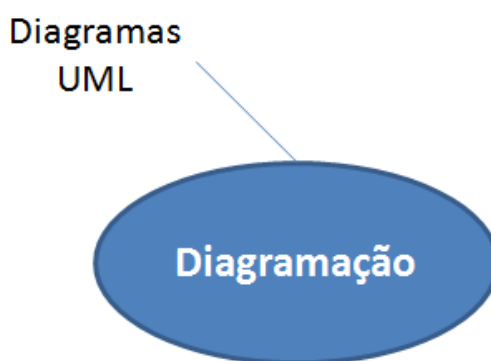


Figura 4.26 – Representação gráfica do fator Diagramação

**Fator 7 – Desenvolvedor solo:** este sétimo fator apresenta pesos fatoriais elevados significativos em práticas relacionadas ao papel de desenvolvedor solo. A prática com maior peso fatorial justifica a percepção de valor “Muito alta” dos respondentes para o desenvolvimento solo. Essa percepção é observada sob a ótica do critério prazo, agregando alto valor ao projeto.



Figura 4.27 – Representação gráfica do fator Desenvolvedor solo

- **Amostra pouco representativa**

A primeira etapa da análise exploratória consistiu em verificar se a amostra selecionada estava apta para ser executada utilizando a técnica de Análise Fatorial.

Para tanto, foram extraídos indicadores provenientes da análise, mostrando que a matriz fatorial construída pode ser considerada.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin.</i>	0,769
Teste de esfericidade de <i>Bartlett</i>	Aproximação Q <sup>2</sup> -34,905
	df 190
	Sig. 0,000

Quadro 4.12 - Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem pouco representativa do critério Prazo

O Quadro 4.12 demonstra os testes de validação da análise fatorial. O índice KMO = 0,769, aproximadamente 0,8, indica que a análise possui uma recomendação ótima. O teste de *Bartlett* apresentou um valor de significância  $p < 0,0001$  e, portanto concluí-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados, pode-ser observar no Quadro 4.13 que a Análise Fatorial mostrou uma estrutura de cinco fatores que explicam 59,515%, aproximadamente 60% da variância de respostas do questionário, considerando autovalores

superiores ou iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).

Fator	Autovalores iniciais		
	Total	% da Variância	% Cumulativa
1	5,757	28,787	28,787
2	1,998	9,990	38,777
3	1,887	9,433	48,210
4	1,226	6,130	54,340
5	1,035	5,175	59,515

Quadro 4.13 – Porcentagem da variância explicada para a análise pouco representativa para o critério Prazo.

A Tabela 4.6 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.6 – Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério prazo

	F1	F2	F3	F4	F5
Equipes de funcionalidades	-0,119	-0,129	0,132	0,187	<b>0,574</b>
Várias equipes trabalhando em paralelo	0,056	0,145	0,109	0,074	0,471
Pequenos times em equipes grandes	0,137	0,293	-0,015	0,307	0,255
Uso de Cenários	0,139	0,037	0,144	<b>0,678</b>	0,280
Documento evocativo	0,102	0,032	-0,190	<b>0,764</b>	0,055
Arquitetura de soluções potenciais	0,188	0,139	0,399	0,073	0,115
Modelo de domínio	<b>0,613</b>	0,211	0,271	0,141	-0,033
Estudo do objetivo de negócio	0,313	0,485	0,255	0,183	0,036
Propriedade coletiva do código	0,124	0,071	<b>0,644</b>	0,035	0,172
Desenvolva depois teste	0,090	-0,035	-0,073	0,043	0,102
Teste primeiro depois desenvolva	0,259	0,116	0,114	0,075	-0,014
Automação de testes	-0,025	0,174	<b>0,548</b>	-0,096	0,012
Gráficos de andamento	0,043	0,473	0,170	-0,120	0,029
Jogos de planejamento	<b>0,748</b>	<b>0,567</b>	0,398	0,141	-0,071
Projeto Velocity	<b>0,657</b>	0,331	0,308	0,025	-0,061
Estimativa por funcionalidade	0,176	0,225	0,319	-0,090	0,182
Relatório de progresso	0,438	<b>0,553</b>	0,149	0,071	0,445
Lista de verificação	0,277	0,322	0,086	0,045	0,378
Técnico ágil presente	<b>0,604</b>	0,073	0,216	0,124	0,084
Análise em grupo com foco no cliente	<b>0,611</b>	0,284	-0,114	-0,006	-0,002

A partir do resultado encontrado na análise fatorial, foi possível a extração de cinco fatores:

**Fator 1 – Modelagem e estimativa de processos:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas classificadas como práticas de modelagem de projetos e técnicas de planejamento de prioridades na implementação de funcionalidades aliadas a práticas de monitoramento. As práticas que obtiveram maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com a etapa de modelagem contribui positivamente em relação ao critério prazo.

**Fator 2 – Planejamento monitorado:** este segundo fator apresenta pesos fatoriais elevados significativos em práticas classificadas como práticas de planejamento e monitoramento de etapas. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positivo dos respondentes para com a etapa de análise do progresso das funcionalidades planejadas, ou seja, se estão sendo desenvolvidas conforme o planejado sem muitas quebras de ciclo contribui positivamente em relação ao critério prazo.

**Fator 3 – Código coletivo e teste automatizado:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas com a codificação coletiva e o teste automatizado da funcionalidade: os desenvolvedores compartilham seu código e paream entre si para tirar dúvidas antes da validação do teste. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério prazo.

**Fator 4 – Cenários simplificados:** este quarto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à criação de cenários que descrevem com simplicidade as necessidades do cliente. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério custo.

**Fator 5 – Equipes de funcionalidades:** este quinto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas com características peculiares para formação de equipes: que estas estejam alinhadas às funcionalidades, ou seja, a criação de equipes e documentação específicas para entregar uma determinada funcionalidade. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério prazo.

#### 4.4.1.4 Escopo

- **Amostra representativa**

Com objetivo de verificar a validação da análise para o critério escopo, foram extraídos indicadores, mostrando que a matriz fatorial construída pode ser considerada.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin</i>		0,760
Teste de esfericidade <i>Bartlett</i>	Aproximação Q <sup>2</sup>	7,858
	DF	300
	Sig.	0,000

Quadro 4.14 – Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem representativa do critério Escopo

O Quadro 4.14 demonstra os testes de validação da análise fatorial. O índice KMO = 0,760, aproximadamente 0,8, indica que a análise possui uma recomendação ótima. O teste de *Bartlett* responde se as variáveis selecionadas são correlacionadas ou não. Neste caso, o teste de *Bartlett* apresenta um valor de significância  $p < 0,0001$  e, portanto conclui-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados no Quadro 4.15, a Análise Fatorial mostrou uma estrutura de oito fatores que explicam 60,8 % da variância de respostas do questionário, considerando autovalores superiores ou iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).



Fator	Autovalores iniciais		
	Total	% da Variância	Cumulativo %
1	6,421	25,683	25,683
2	2,143	8,573	34,255
3	1,834	7,336	41,592
4	1,354	5,416	47,007
5	1,273	5,094	52,101
6	1,119	4,474	56,575
7	1,079	4,318	60,893

Quadro 4.15 – Porcentagem da variância explicada para a análise representativa para o critério Escopo.

A Tabela 4.7 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.7 – Solução da análise fatorial da amostragem representativa de práticas ágeis sob o critério escopo

	F1	F2	F3	F4	F5	F6	F7
Equipes pequenas	0,121	0,035	<b>0,749</b>	0,096	0,087	0,097	0,012
Equipes multifuncionais	0,529	0,037	0,302	0,099	0,226	0,350	0,058
Programador solo	0,086	0,144	0,075	-0,172	-0,016	0,066	0,135
Representante do cliente	0,148	0,119	0,439	0,508	-0,029	0,002	0,204
Facilitador do time	0,137	0,259	<b>0,582</b>	0,137	0,171	0,224	0,483
Programador-Líder	0,149	-0,056	0,092	0,028	0,136	0,163	<b>0,833</b>
Reunião Diária	0,127	0,087	-0,072	-0,031	0,528	0,175	0,270
Reunião em pé	0,176	0,077	0,153	-0,042	<b>0,786</b>	0,041	0,016
Reunião de planejamento da iteração	0,068	-0,042	0,375	0,307	0,438	0,280	0,219
Retrospectiva da iteração	0,199	0,306	0,133	0,396	0,422	0,150	0,228
Lista de prioridades do produto	0,058	0,102	0,132	<b>0,761</b>	0,093	0,072	0,176
Cliente participativo	0,113	0,023	0,068	<b>0,605</b>	-0,071	0,256	0,288
Casos de uso	0,053	<b>0,752</b>	-0,100	0,082	0,142	0,023	0,125
Documento de visão	-0,060	<b>0,769</b>	0,023	0,251	0,143	0,031	0,050
User story	0,046	0,076	0,540	0,273	0,153	0,234	0,011
Diagramas UML	0,097	<b>0,673</b>	0,211	-0,174	-0,001	0,142	0,172
Refatoração	<b>0,555</b>	0,368	0,210	0,067	0,160	0,143	0,183
Programação em pares	0,270	0,163	0,042	0,217	0,234	0,085	0,304
Triagem de erros	0,300	<b>0,613</b>	0,222	0,030	-0,283	0,048	0,288
Testes funcionais	0,527	0,532	0,138	0,081	-0,037	0,311	0,175
Desenvolvimento por casos de teste	<b>0,871</b>	-0,006	-0,027	0,094	0,127	0,011	0,186
Testes unitários	<b>0,882</b>	0,119	0,090	0,043	0,127	0,017	0,053
Integração contínua	0,401	-0,020	0,158	0,273	0,434	0,281	0,022
Iterações fixas	0,235	0,343	-0,130	0,242	0,163	0,505	0,264
Kanban	0,238	0,139	0,376	-0,234	-0,076	0,504	0,114
Lançamento frequente de versões	0,043	0,049	0,101	0,156	0,138	<b>0,765</b>	0,044

A partir do resultado encontrado na análise fatorial, foi possível a extração de sete fatores:

**Fator 1 – Desenvolvimento baseado em casos de teste:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas ao desenvolvimento baseado em casos de teste, onde o desenvolvedor elabora um teste falho e programa códigos para que ele se torne uma funcionalidade sem erros, em seguida aprimora seu código por meio da refatoração. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas de refatoração, testes unitários e desenvolvimento por casos de teste. Essa percepção é observada sob a ótica do critério escopo, agregando alto valor ao projeto.

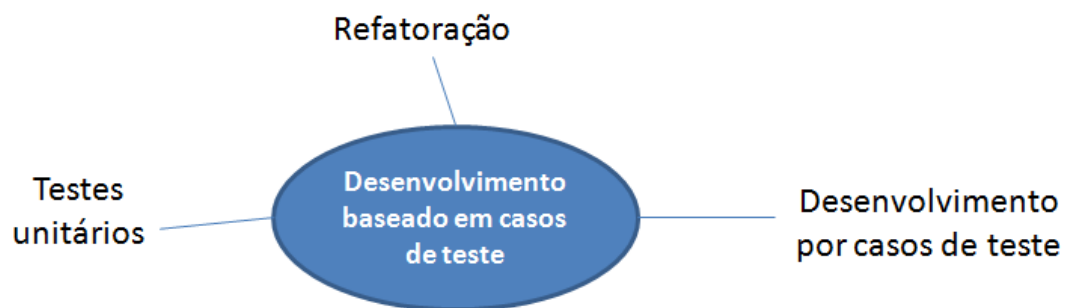


Figura 4.28 – Representação gráfica do fator Desenvolvimento solo

**Fator 2 – Diagramas de erros:** este segundo fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à elaboração de diagramas UML e casos de uso para a representação dos erros de aceitação, com objetivo de serem corrigidos. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas casos de uso, documento de visão, diagrama UML e triagem de erros. Essa percepção é observada sob a ótica do critério escopo, agregando alto valor ao projeto.

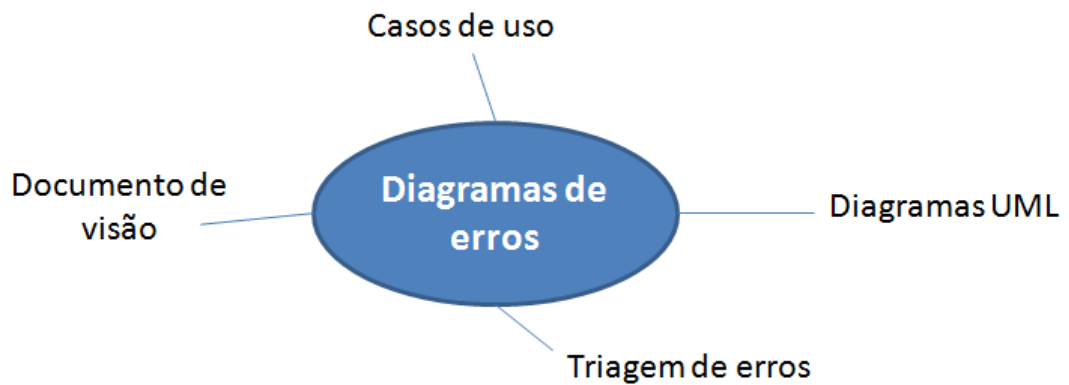


Figura 4.29 – Representação gráfica do fator Diagrama de erros

**Fator 3 – Equipes pequenas lideradas pelo facilitador:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à formação de equipes pequenas lideradas pelo facilitador do time. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas de equipes pequenas e o papel do facilitador do time. Essa percepção é observada sob a ótica do critério escopo, agregando alto valor ao projeto.

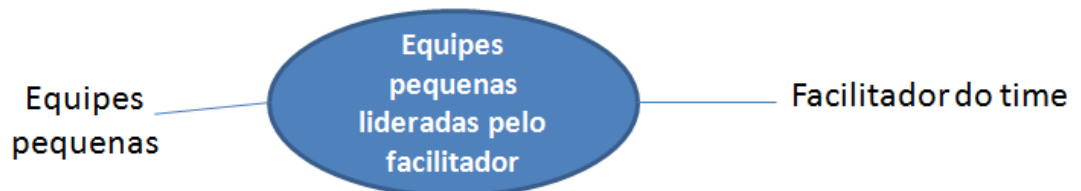


Figura 4.30 – Representação gráfica do fator Equipes pequenas lideradas pelo facilitador

**Fator 4 – Funcionalidades definidas pelo cliente:** este quarto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas à elaboração da lista de prioridades com a participação do cliente. As práticas que obtiveram maiores pesos fatoriais justificam a percepção de valor positiva dos respondentes para as práticas lista de prioridades do produto e cliente participativo. Essa percepção é observada sob a ótica do critério escopo, agregando alto valor ao projeto.

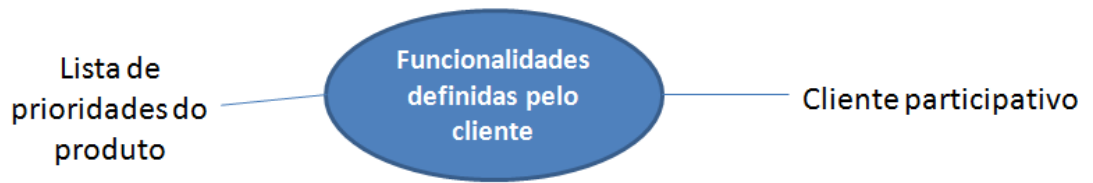


Figura 4.31 – Representação gráfica do fator Funcionalidades definidas pelo cliente

**Fator 5 – Reuniões em pé:** este quinto fator apresenta pesos fatoriais elevados significativos para a prática de reunião em pé. A prática com maior peso fatorial justifica a percepção de valor “Muito alta” dos respondentes para a prática de reunião em pé. Essa percepção é observada sob a ótica do critério escopo, agregando alto valor ao projeto.

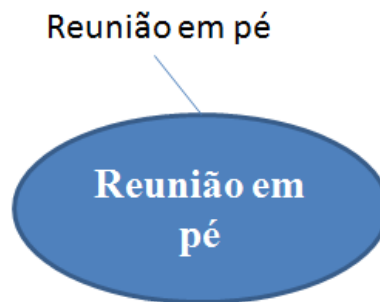


Figura 4.32 – Representação gráfica do fator Reunião em pé

**Fator 6 – Releases freqüentes:** este sexto fator apresenta pesos fatoriais elevados significativos para a prática de lançamento frequente de versões do software. A prática justifica a percepção de valor positiva dos respondentes, observada sob a ótica do critério escopo, agregando alto valor ao projeto.



Figura 4.33 – Representação gráfica do fator Releases frequentes

**Fator 7 – Programador-líder:** este sétimo fator apresenta pesos fatoriais elevados significativos para o papel de programador-líder. O fator justifica a percepção de valor positiva dos respondentes observada sob a ótica do critério escopo, agregando alto valor ao projeto.



Figura 4.34 – Representação gráfica do fator Programador-líder

- **Amostra pouco representativa**

A primeira etapa da análise exploratória consistiu em verificar se a amostra selecionada estava apta para ser executada utilizando a técnica de Análise Fatorial.

Para tanto, foram extraídos indicadores provenientes da análise, mostrando que a matriz fatorial construída pode ser considerada.

Medida de adequação amostral <i>Kaiser-Meyer-Olkin</i>	0,760
Teste de esfericidade de Bartlett	Aproximação Q <sup>2</sup>
	DF
	Sig
	-32,225
	190
	0,000

Quadro 4.16 – Coeficiente KMO da análise fatorial e teste de esfericidade de *Bartlett* para amostragem pouco representativa do critério Escopo

O Quadro 4.16 resume as informações de validação da análise fatorial. O índice KMO define a qualidade da amostra cujo valor é  $KMO = 0,760$ , aproximadamente 0,8, valida a análise com recomendação ótima. O teste de *Bartlett* responde se as variáveis selecionadas são correlacionadas ou não. Neste caso, o teste de *Bartlett* apresenta um valor de significância  $p < 0,0001$  e, portanto conclui-se que as variáveis estão correlacionadas significativamente.

Em relação ao número de fatores observados, pode-se observar no Quadro 4.17 que a Análise Fatorial mostrou uma estrutura de sete fatores que explicam 72,091%, aproximadamente 72% da variância de respostas do questionário, considerando autovalores superiores ou iguais a 1, como recomendado pelo critério de raiz latente (HAIR *et al.*, 2009; MAROCO, 2010).

Fator	Autovalores iniciais		
	Total	% da Variância	% Cumulativa
1	6,246	31,229	31,229
2	1,991	9,957	41,186
3	1,740	8,700	49,886
4	1,257	6,283	56,169
5	1,099	5,494	61,663
6	1,069	5,346	67,008
7	1,016	5,082	72,091

Quadro 4.17 – Porcentagem da variância explicada para a análise pouco representativa para o critério Escopo.

A Tabela 4.8 mostra a lista de práticas selecionadas e os fatores escolhidos como solução da análise.

Tabela 4.8 – Solução da análise fatorial da amostragem pouco representativa de práticas ágeis sob o critério escopo

	F1	F2	F3	F4	F5	F6	F7
Equipes de funcionalidades	-0,062	0,157	0,075	0,093	0,231	0,082	0,468
Várias equipes trabalhando em paralelo	-0,151	0,047	0,019	<b>0,749</b>	0,127	0,335	0,042
Pequenos times me equipes grandes	0,217	0,204	0,218	<b>0,797</b>	0,014	0,085	0,011
Uso de cenários	0,288	0,060	0,178	0,165	<b>0,766</b>	0,049	0,032
Documento evocativo	0,160	0,216	0,185	-0,032	<b>0,691</b>	0,017	0,224
Arquitetura de soluções potenciais	0,247	0,090	<b>0,798</b>	0,131	0,145	0,280	0,096
Modelo de domínio	0,166	0,380	<b>0,593</b>	0,078	0,164	0,172	0,110
Estudo do objetivo do negócio	0,103	0,222	0,182	0,067	0,004	<b>0,585</b>	0,056
Propriedade coletiva do código	<b>0,672</b>	0,099	0,084	0,007	0,165	0,251	0,054
Desenvolva depois teste	0,186	0,164	<b>0,594</b>	0,077	0,175	0,124	0,040
Teste primeiro depois desenvolva	<b>0,554</b>	0,285	0,122	0,086	0,115	0,093	0,197
Automação de testes	<b>0,592</b>	0,100	0,334	0,121	-0,044	0,223	0,069
Gráficos de andamento	0,274	-0,013	-0,042	0,118	0,007	0,413	0,195
Jogos de planejamento	<b>0,592</b>	0,225	0,211	0,173	-0,085	0,170	0,044
Projeto Velocity	0,436	0,282	0,144	0,125	-0,033	0,333	0,159
Estimativa por funcionalidade	0,179	0,054	-0,001	0,384	0,032	0,376	0,119
Relatório de progresso	0,303	<b>0,708</b>	0,162	0,175	0,240	0,245	0,048
Lista de verificação	0,207	<b>0,793</b>	0,187	0,069	0,124	0,185	0,064
Técnico ágil presente	0,493	<b>0,576</b>	0,262	0,166	0,046	0,072	0,104
Análise em grupo com foco no cliente	0,109	0,479	0,248	0,273	0,109	0,056	<b>0,643</b>

A partir do resultado encontrado na análise fatorial, foi possível a extração de sete fatores:

**Fator 1 – Planejamento coletivo:** este primeiro fator apresenta pesos fatoriais elevados significativos em práticas classificadas como práticas de planejamento de funcionalidades e execução de testes. O peso fatorial de maior valor para a prática Propriedade coletiva do código indica uma tendência no planejamento e implementação coletiva de funcionalidades, onde toda a equipe colabora na definição de prioridades, não apenas o cliente e seu representante. As práticas que obtiveram maiores pesos fatoriais hipoteticamente justificam a percepção de valor dos respondentes em relação ao critério

custo, ou seja, os respondentes percebem que essas práticas contribuem para a redução do escopo do projeto.

**Fator 2 – Verificação técnica:** este segundo fator apresenta pesos fatoriais elevados significativos em práticas classificadas como práticas de monitoramento por um técnico ágil. Os valores indicam uma tendência da necessidade de monitoramento do andamento do projeto coordenada por uma pessoa especialista em práticas ágeis, para que as habilidades da equipe cresçam e ganhem maturidade. As práticas que obtiveram maiores pesos fatoriais hipoteticamente justificam a percepção de valor dos respondentes em relação ao critério custo, ou seja, os respondentes percebem que essas práticas contribuem para a redução do escopo de trabalho, cujo domínio é falho em equipes com falta de experiência.

**Fator 3 – Modelagem de processos de teste:** este terceiro fator apresenta pesos fatoriais elevados significativos em práticas relacionadas com arquitetura de projetos: modelagem de soluções potenciais para os problemas de implementação e testando em seguida. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério escopo agrega alto valor ao projeto.

**Fator 4 – Equipes pequenas trabalhando paralelamente:** este quarto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas com características peculiares para formação de equipes trabalhando paralelamente, divididas em equipes menores para componentizar o trabalho. Os maiores pesos fatoriais hipoteticamente justificam a percepção de valor dos respondentes para com esta etapa em relação ao critério escopo agrega alto valor ao projeto.

**Fator 5 – Cenários simplificados:** este quinto fator apresenta pesos fatoriais elevados significativos em práticas relacionadas com características peculiares para arquitetura do projeto: que os requisitos do cliente sejam representados por cenários bem simplificados para cada funcionalidade requerida pelo cliente. Os maiores pesos fatoriais



hipoteticamente justificam a percepção de valor dos respondentes para com esta etapa em relação ao critério escopo agrega alto valor ao projeto.

**Fator 6 - Estudo do objetivo do negócio:** este sexto fator apresenta pesos fatoriais elevados significativos na prática de estudo do objetivo do negócio, onde antes mesmo de iniciar o projeto, a equipe avalia sua viabilidade e quais as melhores práticas a serem utilizadas. O maior peso fatorial hipoteticamente justifica a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério custo agrega alto valor ao projeto.

**Fator 7 - Análise em grupo com foco no cliente:** este sétimo fator apresenta pesos fatoriais elevados significativos na prática análise em grupo com foco no cliente, cuja característica é a revisão de todo ciclo do projeto, observando se as necessidades dos clientes foram atendidas. O maior peso fatorial hipoteticamente justifica a percepção de valor positiva dos respondentes para com esta etapa em relação ao critério custo agrega alto valor ao projeto.

#### **4.4.2 Análise de Cluster**

A avaliação da percepção de valor dos praticantes para com as práticas de metodologias ágeis constituiu-se na etapa chave para construção de fatores que justificam a correlação das variáveis estudadas, ou seja, a correlação entre práticas de metodologias ágeis em relação aos critérios de sucesso de projetos de software.

Visando contribuir com um melhor entendimento dos fatores encontrados, a segunda etapa da análise constituiu na verificação de semelhanças e características de percepção comuns entre um conjunto de categorias de respondentes, utilizando a técnica de *Análise de Clusters*.

No contexto deste trabalho, a *Análise de Cluster* é utilizada para identificar semelhanças na percepção de valor que os respondentes possuem sobre os fatores encontrados para cada critério analisado, sob o ponto de vista do cargo por ele ocupado na empresa.

Para todas as análises realizadas foi utilizado o método de *Ward* para agrupamento hierárquico dos sujeitos analisados, onde em cada um dos passos do algoritmo, os *Clusters* são formados para minimizar a variabilidade interna dentro dos *Clusters* usando a distância euclidiana quadrada como medida de similaridade.

Essa árvore hierárquica é chamada de Dendrograma e para cada fator analisado será gerada uma representação gráfica onde os respondentes que possuem percepção de valor semelhante em relação ao fator analisado são identificados através do cargo em que ocupa na empresa. Com o objetivo de obter grupos mais heterogêneos de percepção semelhantes, foram realizados cortes a uma distância  $d= 5$  em todos os casos analisados.

#### 4.4.2.1 Custo

Para a análise de cluster do critério custo, foram utilizados apenas os fatores gerados das práticas de maior representatividade amostral. A partir desta análise, foi possível a identificação da relação entre o perfil do usuário e os fatores encontrados.

**Fator 1 - Interação da equipe:** Segundo o Dendrograma representado na Figura 4.35, percebe-se três grandes grupos. O primeiro é composto por indivíduos dos cargos de gerente de projetos, analista de sistemas, *Scrum Coach*, analista de negócio e sócio; o segundo por gerentes de projeto, analista de processos, programador e pesquisador/desenvolvedor; o terceiro grupo é composto por analista de sistema, líder de projeto, programador, líder técnico, CIO e professor/pesquisador. Esse modelo de agrupamento sugere que há para cada um dos três grupos, os indivíduos que a ele pertencem possui uma percepção de valor semelhante para as práticas deste fator, ou seja, seguiram o mesmo padrão de avaliação, com dominância de percepção de valor dos gerentes de projeto nos três agrupamentos.

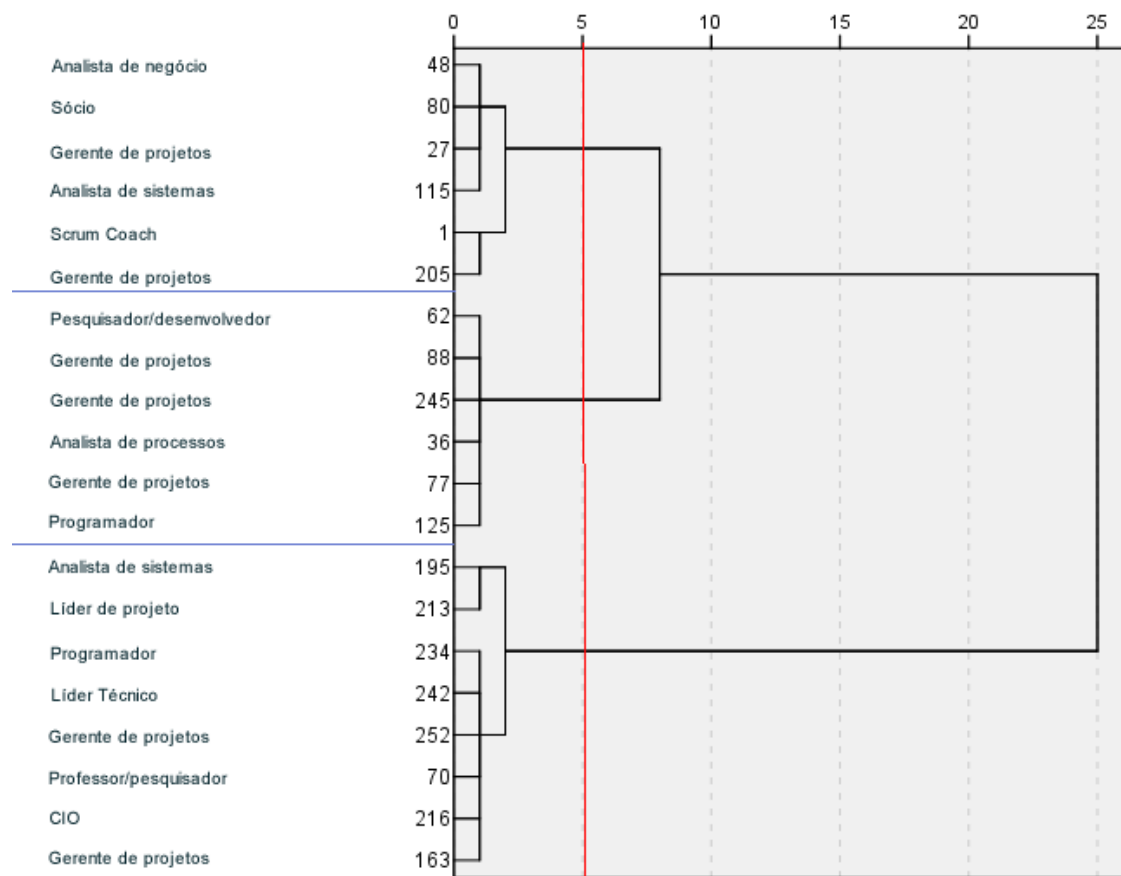


Figura 4.35 – Dendrograma da análise de cluster para o Fator 1 – Interação da equipe, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 2 - Liderança e custos:** Segundo o Dendrograma representado na Figura 4.36, percebe-se três grandes grupos. O primeiro é composto por indivíduos dos cargos de líder de projetos, analista de sistemas, diretor fiscal e de tecnologia e analista de negócio; o segundo por gerentes de projeto, líder técnico, analista de sistemas, líder de projeto, programador e pesquisador/desenvolvedor; o terceiro grupo é composto por analista de negócio, gerente de projetos, *Scrum Master*, sócio, CIO e pesquisador/desenvolvedor. Esse modelo de agrupamento sugere que há para cada um dos três grupos, os indivíduos que a ele pertencem possui uma percepção de valor semelhante para as práticas relacionadas à liderança e custos, ou seja, seguiram o mesmo padrão de avaliação, com dominância na percepção do líder de projeto.

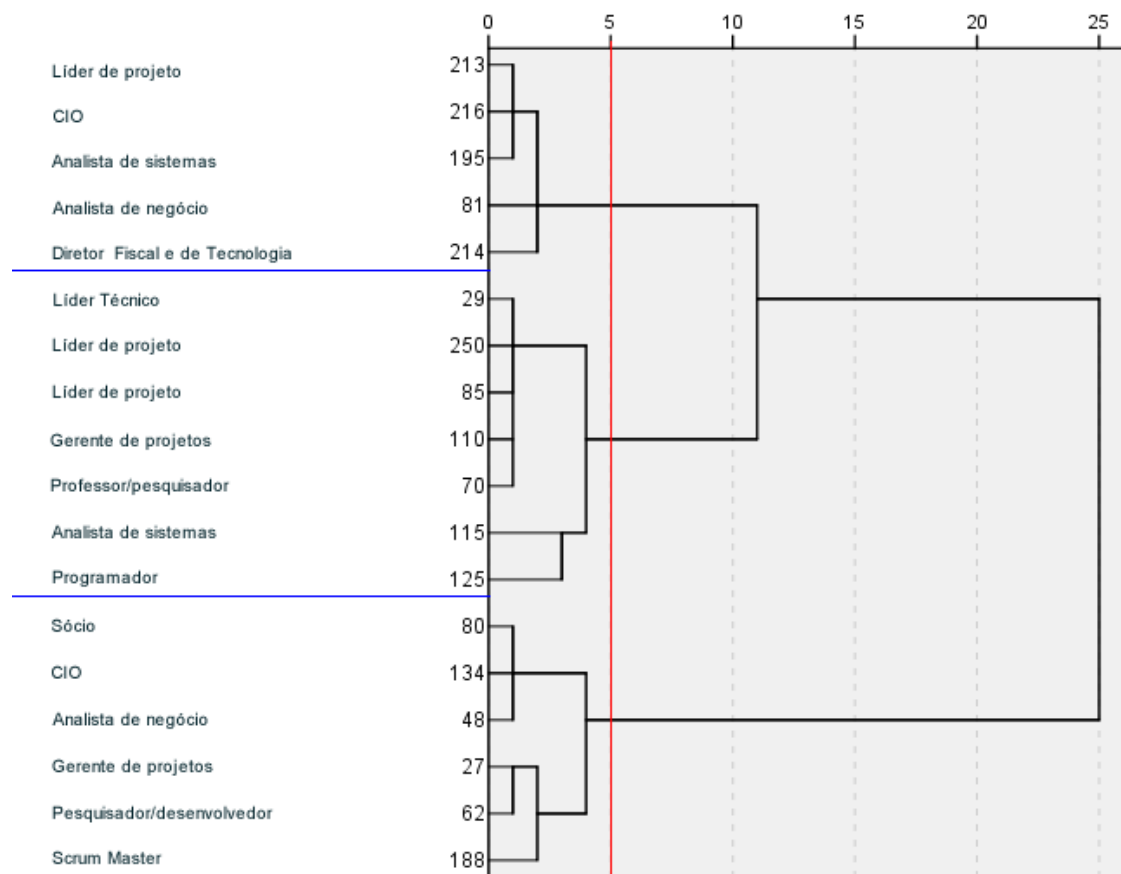


Figura 4.36– Dendrograma da análise de cluster para o Fator 2 – Liderança e custos, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 3 - Desenvolvimento:** Na Figura 4.37, o Dendrograma desenhado representa três grandes grupos. O primeiro é composto por indivíduos dos cargos de gerente de projetos, programadores, líder técnico e líder de projeto; o segundo por gerentes de projeto, analista de sistemas, programadores e arquiteto de software; o terceiro grupo é composto por analista de sistema, gerente de projeto, CIO e pesquisador/desenvolvedor. Esse modelo de agrupamento sugere que há para cada um dos três grupos, os indivíduos que a ele pertencem possui uma percepção de valor semelhante para as práticas relacionadas ao fator desenvolvimento, com dominância de percepção de programadores e gerentes de projeto.

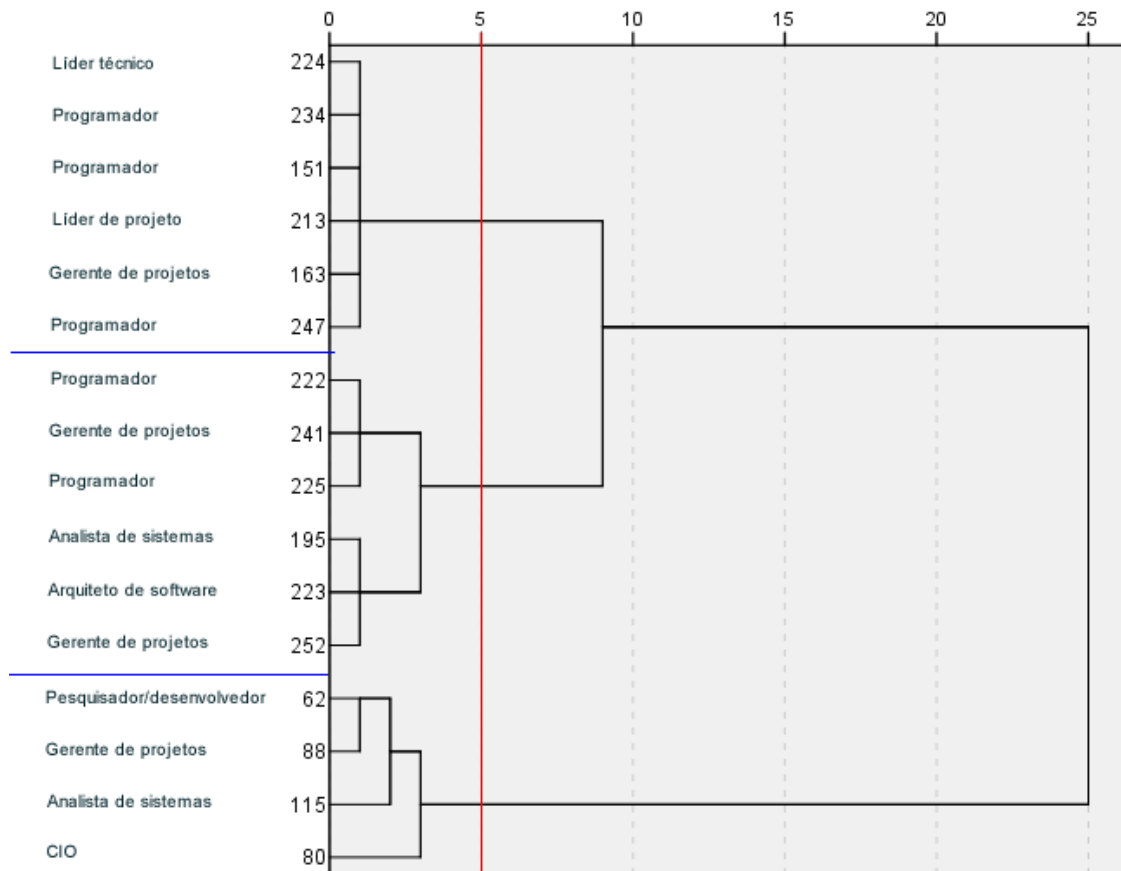


Figura 4.37 – Dendrograma da análise de cluster para o Fator 3 – Desenvolvimento, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 4 – Tempo de resposta (Just-in-time):** A partir do Dendrograma representado na Figura 4.38, percebe-se dois grandes grupos. O primeiro é composto por indivíduos dos cargos de gerência e análise, como gerente de projetos, analista de sistemas, programadores e analista de TI; o segundo grupo além de ser composto por indivíduos de gerência e análise como gerentes de projeto, analista de sistemas, analista de negócio, administrador de banco de dados, possui cargos de alta cúpula como CIO e CEO, além de um indivíduo pesquisador/desenvolvedor. Esse modelo de agrupamento sugere que há para cada um dos dois grupos, os indivíduos que a ele pertencem possui uma percepção de valor semelhante para as práticas do fator tempo de resposta, ou seja, seguiram o mesmo padrão de avaliação, com dominância na percepção dos indivíduos nos cargos de líder de projeto e gerente de projeto.

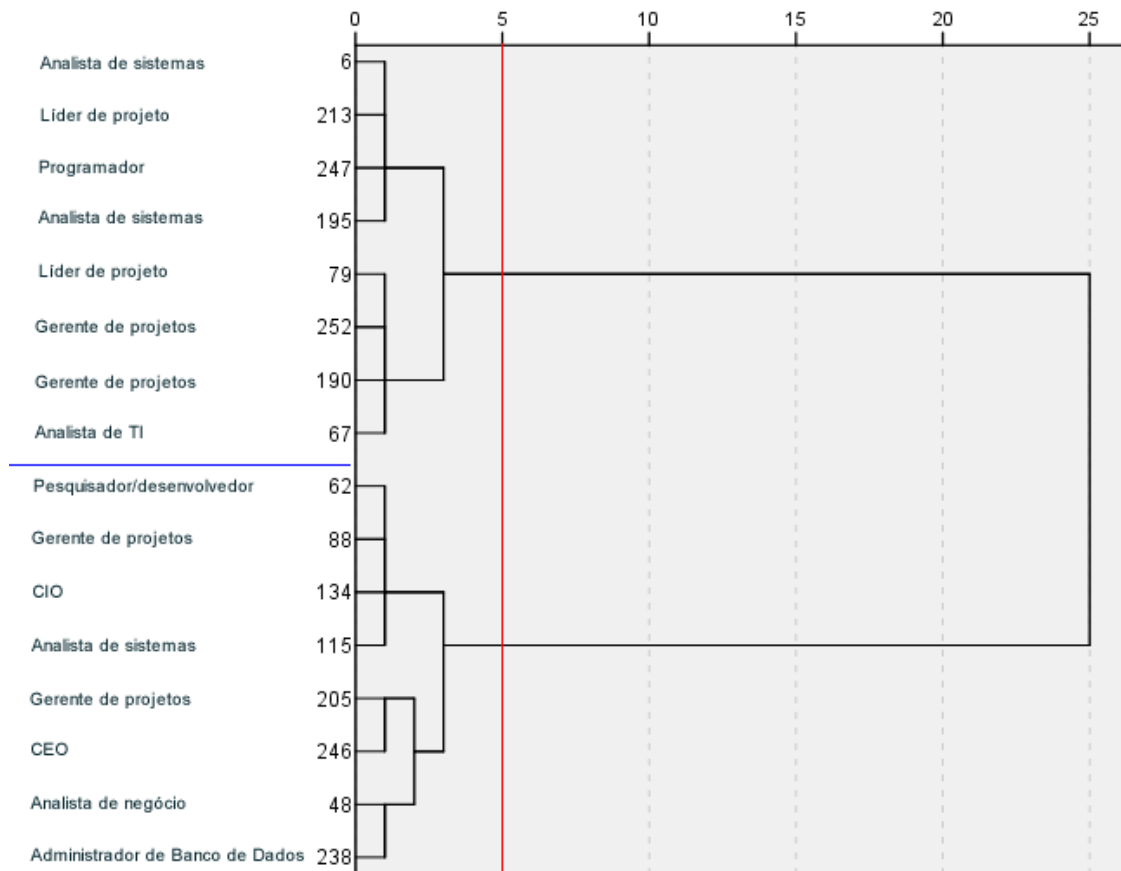


Figura 4.38 – Dendrograma da análise de cluster para o Fator 4 – Tempo de resposta, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 5 – Modelagem de processos:** Segundo o Dendrograma representado na Figura 4.39, percebe-se três grandes grupos. O primeiro é composto por indivíduos de cargos bem heterogêneos, como gerente de projetos, analista de negócio, Engenheiro de software, administrador de banco de dados, professor/pesquisador, CIO e sócio; o segundo por gerentes de projeto, analista de sistema, analista de TI, testador, programador, CIO, coordenador de processos e professor/pesquisador; o terceiro grupo é composto por analista de negócio, líder de projeto, CIO e CEO. Esse modelo de agrupamento sugere que há para cada um dos três grupos, os indivíduos que a ele pertencem possui uma percepção de valor semelhante para as práticas deste fator. Pode-se observar que há dominância de percepção nos três agrupamentos dos cargos programador, gerentes de projeto e líder de projeto.

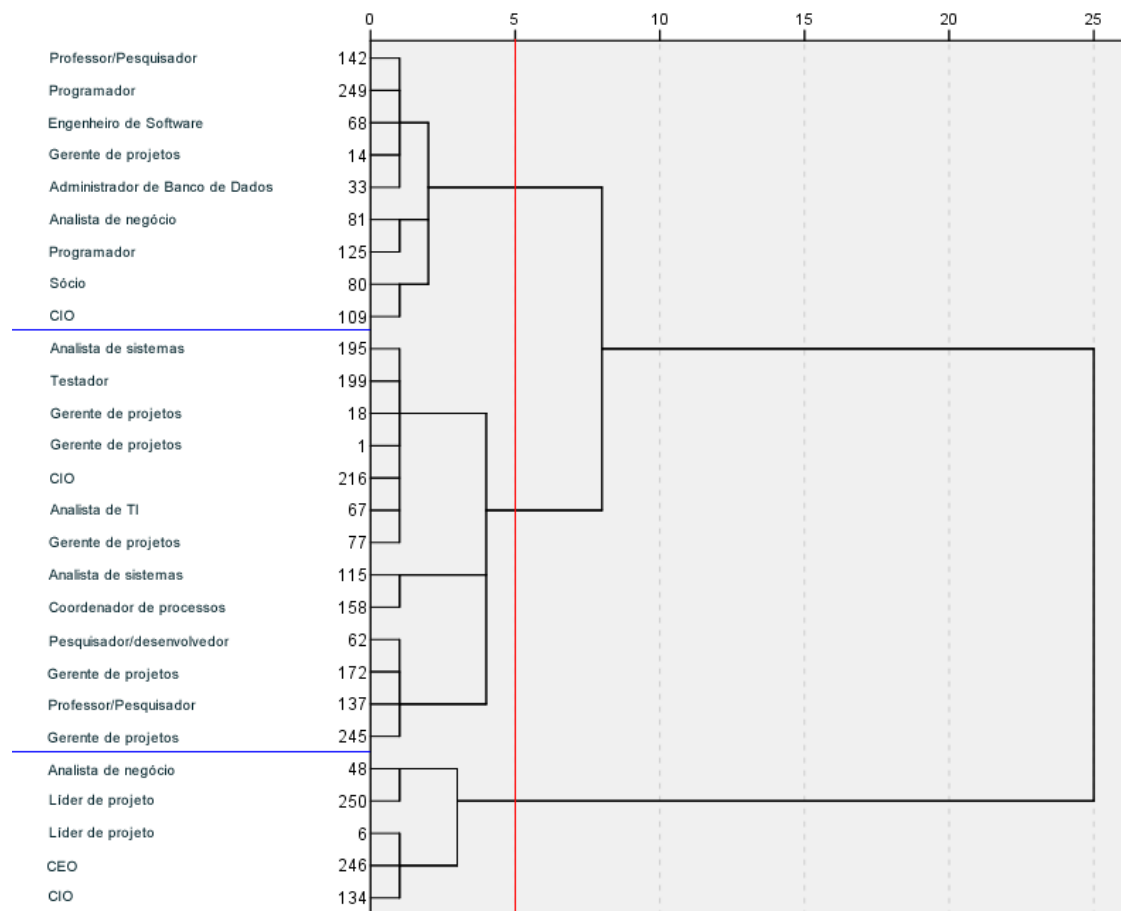


Figura 4.39 – Dendrograma da análise de cluster para o Fator 5 – Modelagem de processos, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 6 – Desenvolvimento solo:** Segundo o Dendrograma desenhado na Figura 4.40, percebe-se três grandes grupos. O primeiro é composto por indivíduos dos cargos de gerente de projetos, analista de sistemas, *Scrum Coach*, analista de negócio e sócio; o segundo por gerentes de projeto, analista de processos, programador e pesquisador/desenvolvedor; o terceiro grupo é composto por analista de sistema, líder de projeto, programador, líder técnico, CIO e professor/pesquisador. Esse modelo de agrupamento sugere que há para cada um dos três grupos, os indivíduos que a ele pertencem possui uma percepção de valor semelhante para as práticas deste fator, ou seja, seguiram o mesmo padrão de avaliação, com predominância na percepção dos gerentes de projeto e dos programadores.

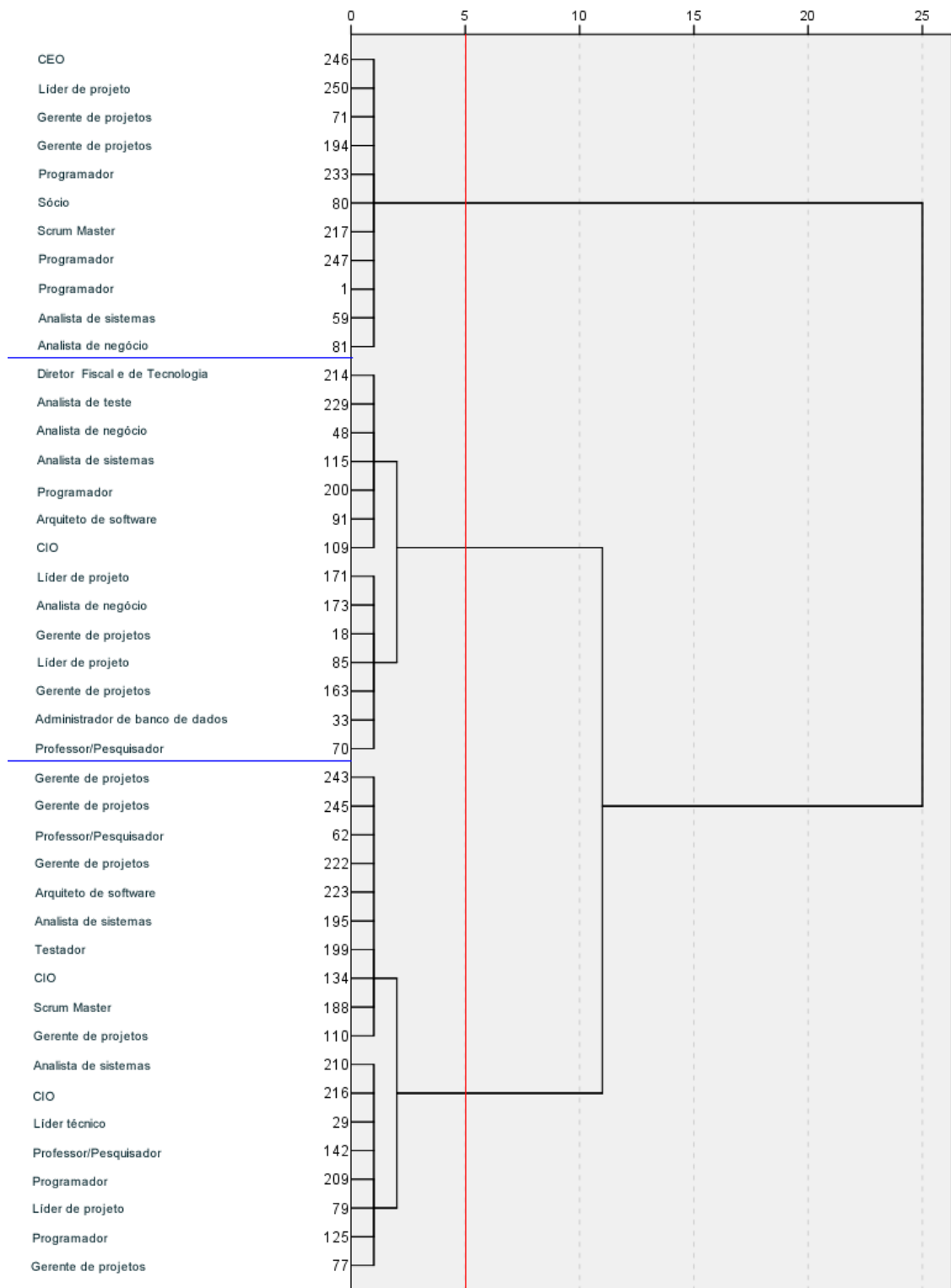


Figura 4.40 – Dendrograma da análise de cluster para o Fator 6 – Desenvolvimento solo, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.



### 4.4.2.2 Qualidade

Para a análise de cluster do critério qualidade, foram utilizados os fatores de maior representatividade amostral para a identificação da relação entre o perfil do usuário e os fatores encontrados.

**Fator 1 – Backlog de integração contínua:** Na Figura 4.41, de acordo com o Dendrograma gerado, para o primeiro grupo os cargos de nível administrativo, gerência, técnico e de alta gerência estão fortemente correlacionados neste fator. Isso significa que estes indivíduos possuem percepção semelhante uns aos outros. Já o outro grupo é formado pelos cargos de nível administrativo e de gerência. Em ambos os grupos, há uma predominância de percepção de valor do gerente de projetos em relação ao fator.

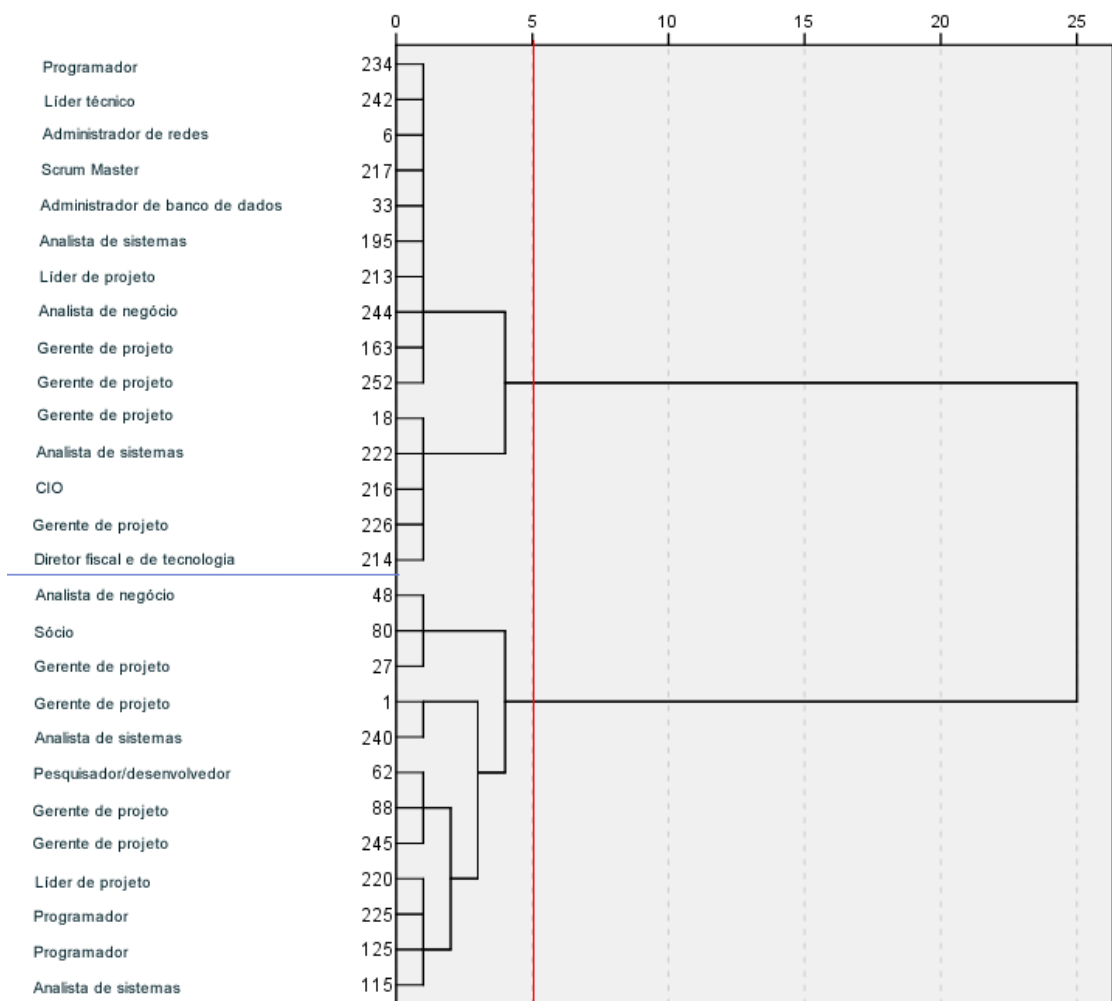


Figura 4.41 – Dendrograma da análise de cluster para o Fator 1 – Backlog de integração contínua, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 2 – Análise de requisitos:** Segundo o Dendrograma especificado na Figura 4.42, percebe-se quatro grupos com cargos bem heterogêneos. No primeiro grupo há predominância de percepção de indivíduos com o cargo de programador; o segundo grupo é composto por um gerente de projeto, CIO, CEO, analista de negócio e sócio, sem dominância de nenhum cargo específico; o terceiro grupo é composto por cargos de nível gerencial, com predominância de percepção do gerente de projetos; e o quarto grupo possui indivíduos de cargos de liderança e também programadores, com predominância de percepção do gerente de projetos.

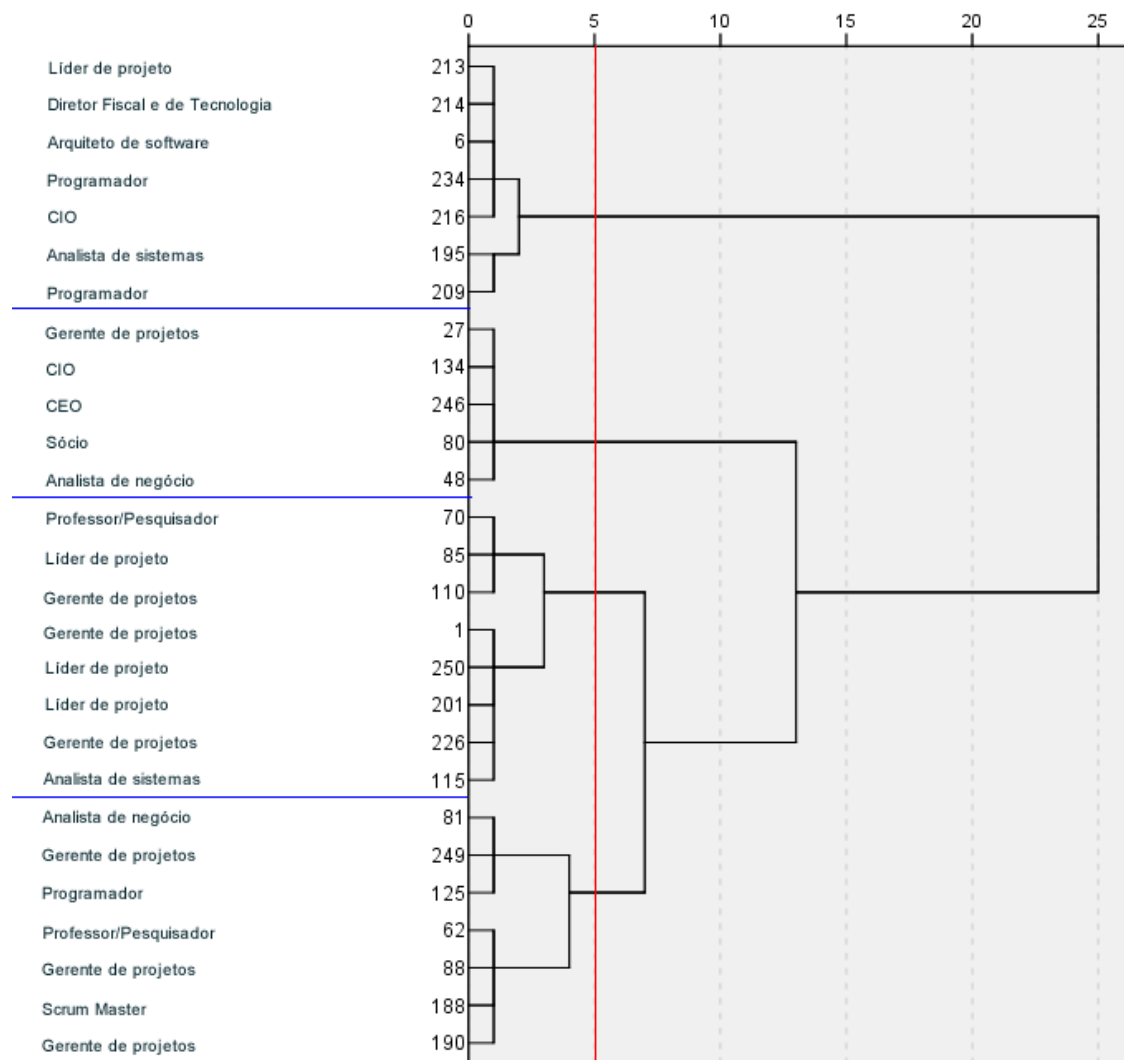


Figura 4.42 – Dendrograma da análise de cluster para o Fator 2 – Análise de requisitos, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

A semelhança de percepção de gerentes e programadores para esse fator pode ser explicada pela atividade de análise de requisitos que em muitas situações requer a participação dos dois cargos.

**Fator 3 – Modelagem do processo de teste:** De acordo com o Dendrograma gerado na Figura 4.43, percebe-se quatro grupos com cargos bem heterogêneos. O primeiro grupo é composto por um gerente de projeto, CIO, analista de negócio e analista de TI, sem dominância de nenhum cargo específico há predominância de percepção de indivíduos com o cargo de programador; o segundo grupo é composto por indivíduos de cargos de analista, com predominância de percepção do analista de sistemas; o terceiro grupo é composto por cargos de nível gerencial e de alta gerência, com predominância de percepção do líder de projeto, para modelagem de processos de teste.

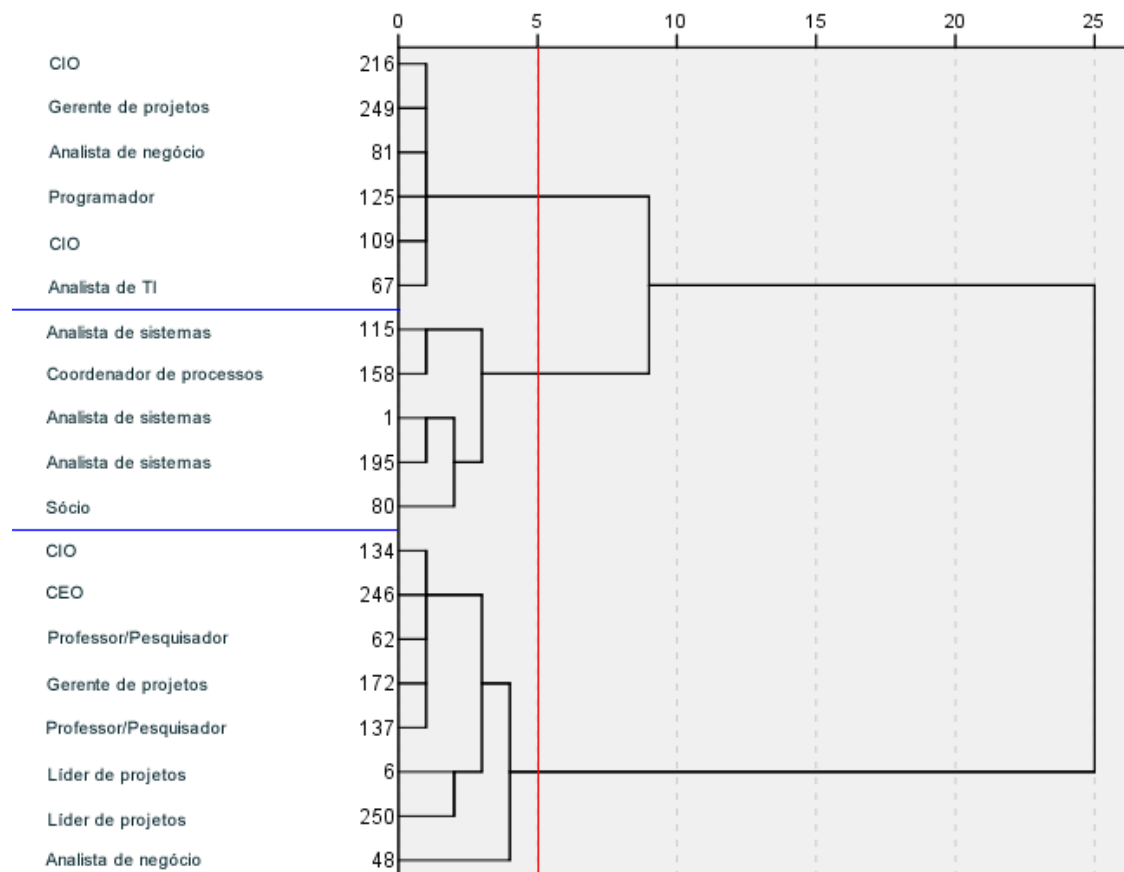


Figura 4.43 – Dendrograma da análise de cluster para o Fator 3 – Modelagem de processo de teste, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 4 – Prevenção de erros com casos de teste:** Segundo o Dendrograma representado na Figura 4.44, é possível a identificação de dois grupos com cargos bem heterogêneos. O primeiro grupo formado, que possui semelhança de percepção para cargos de nível de análise, gerencial, alta gerência e desenvolvimento, com predominância para a percepção do líder de projeto e do gerente de projeto; o segundo grupo é composto apenas por indivíduos de cargos de nível gerencial, com predominância de percepção do gerente de projetos e do líder de projeto, para o fator prevenção de erros com casos de teste.

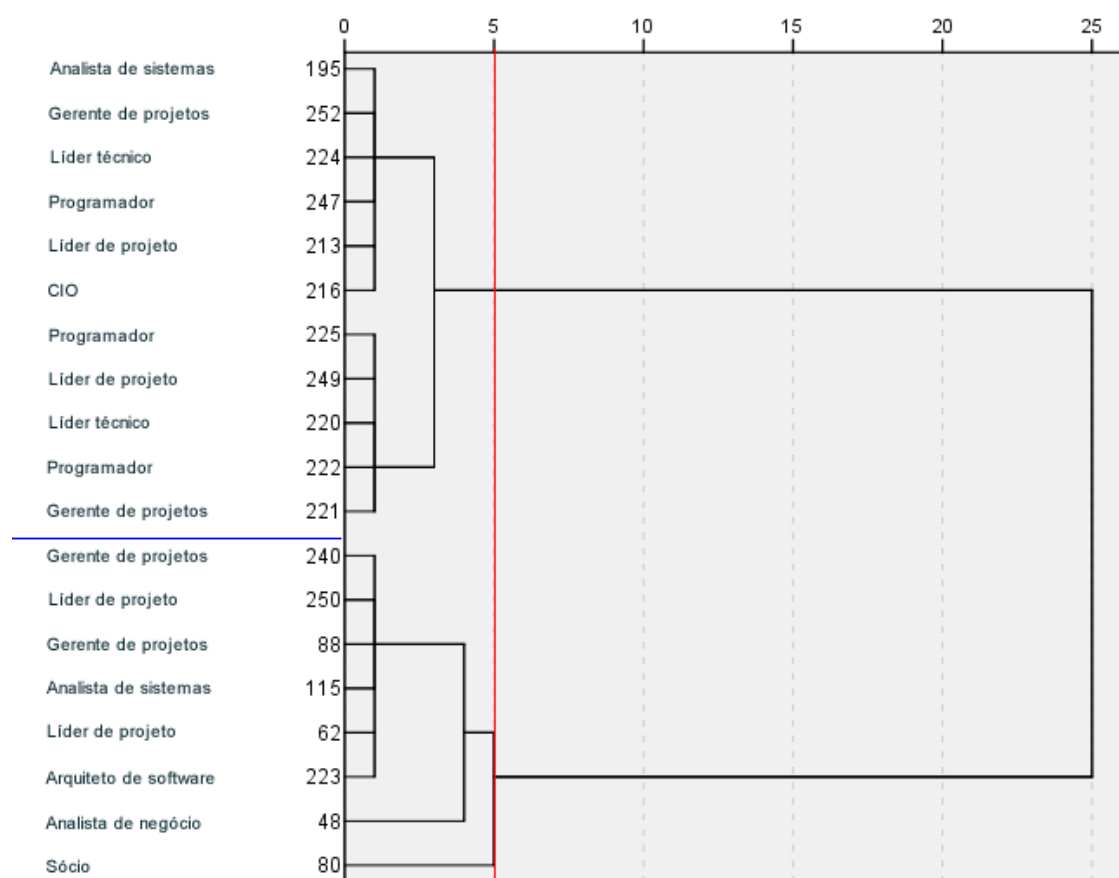


Figura 4.44 – Dendrograma da análise de cluster para o Fator 4 – Prevenção de erros com casos de teste, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 5 – Documento de visão:** O Dendrograma gerado na Figura 4.45, possibilita a identificação de três grupos com cargos bem heterogêneos. O primeiro grupo formado, que possui semelhança de percepção para cargos de nível de análise, gerencial, alta gerência, pesquisa e desenvolvimento, com predominância para a percepção do gerente de projeto; o segundo grupo é composto apenas por indivíduos de cargos de nível gerencial,

com predominância de percepção do gerente de projetos; e o terceiro grupo, formado por um indivíduo de cargo de alta gerência e indivíduos de cargos de nível técnico, com predominância de percepção de indivíduos do cargo programador, para o fator documento de visão.

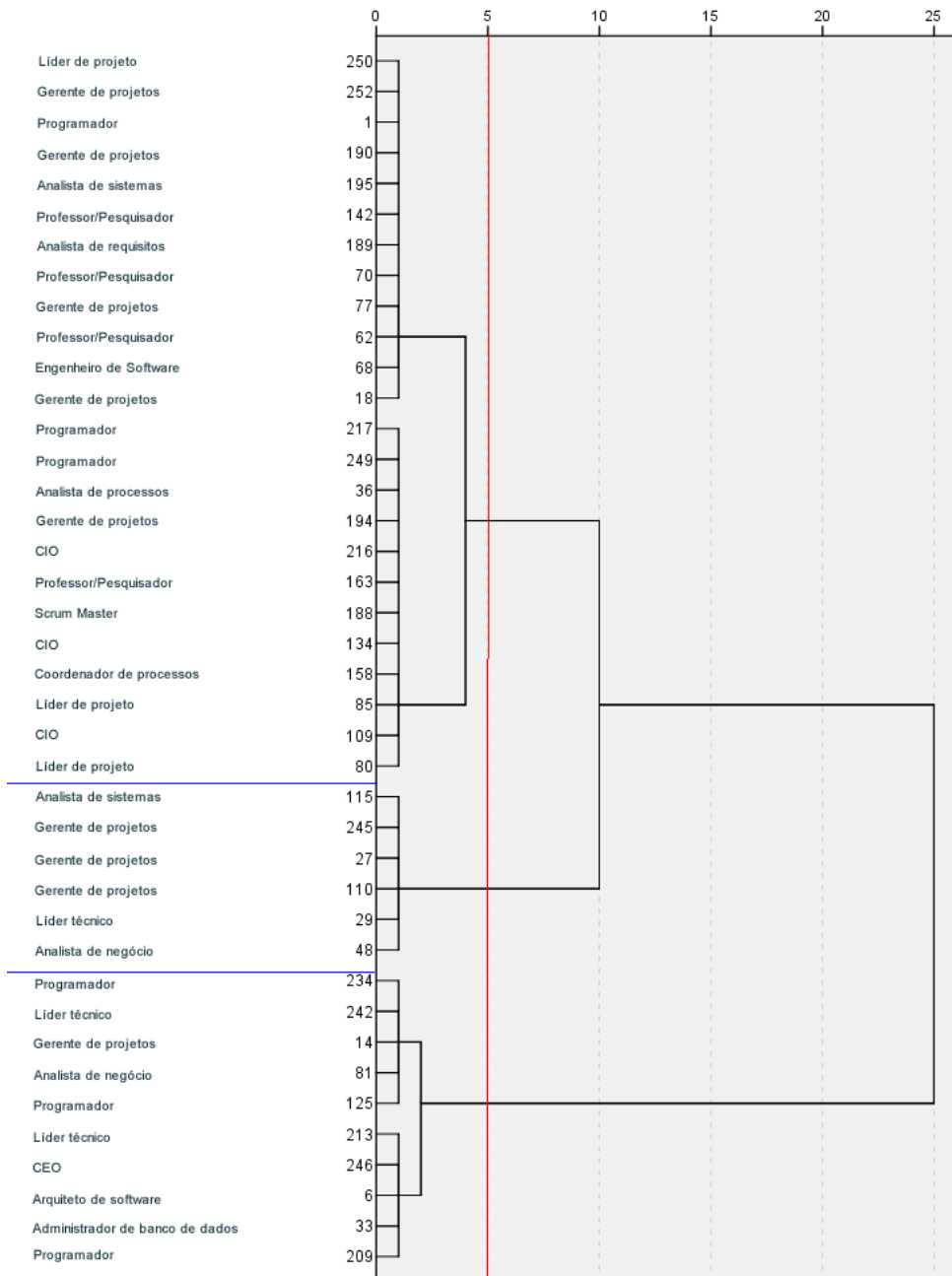


Figura 4.45 – Dendrograma da análise de cluster para o Fator 5 – Documento de visão, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 6 - Equipes de teste multifuncionais:** Segundo o Dendrograma gerado na Figura 4.46, é possível a identificação de três grupos. O primeiro grupo formado, que possui semelhança de percepção para cargos de nível desenvolvimento, gerencial e de alta gerência, com predominância para a percepção do gerente de projetos; o segundo grupo é composto apenas por indivíduos de cargos de nível gerencial, alta gerência, desenvolvimento e pesquisa com predominância de percepção do gerente de projetos; e o terceiro grupo, formado por um indivíduo de cargo de administração com a percepção predominante do analista de sistemas, para o fator equipes de teste multifuncionais.

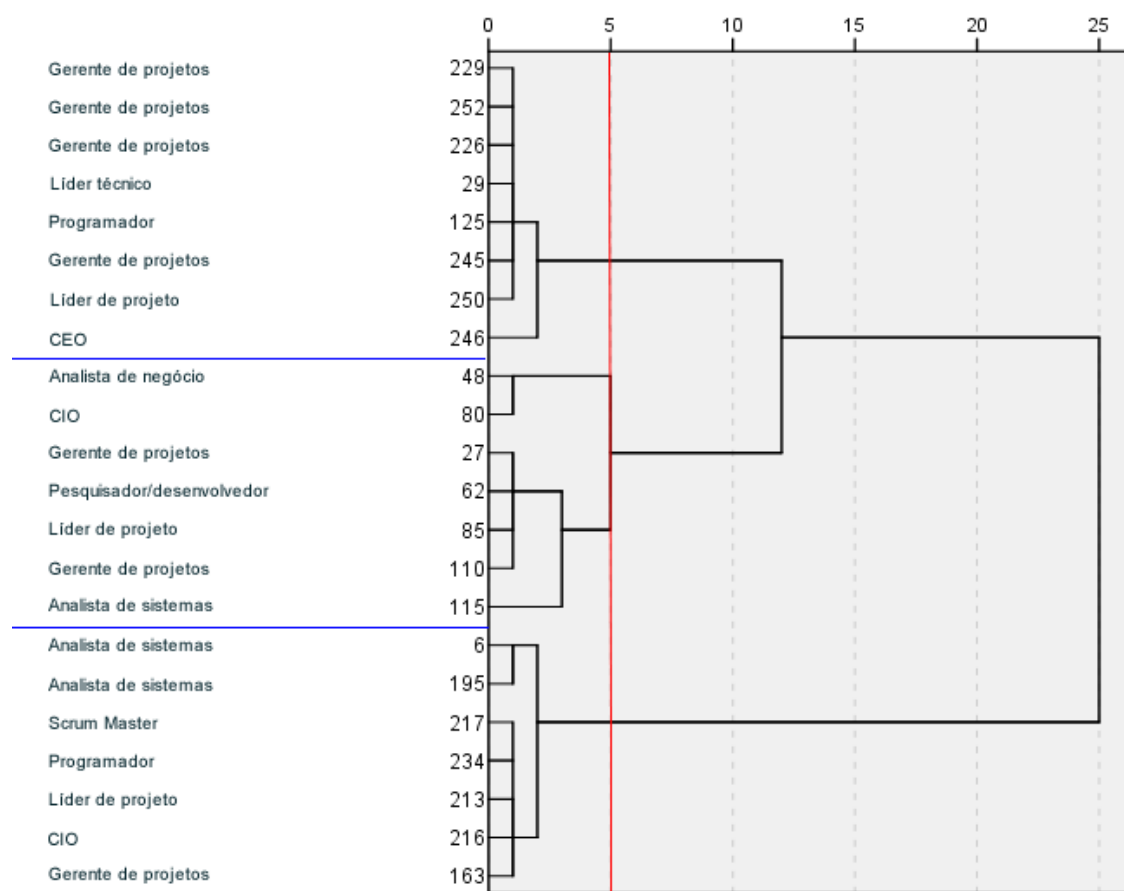


Figura 4.46 – Dendrograma da análise de cluster para o Fator 6 – Equipes de teste multifuncionais, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

### 4.4.2.3 Prazo

Para a análise de cluster do critério prazo, foram utilizados apenas os fatores gerados das práticas de maior representatividade amostral. A partir desta análise, foi possível a identificação da relação entre o perfil do usuário e os fatores encontrados.

**Fator 1 - Release com tempo pré-fixado:** A partir do Dendrograma gerado na Figura 4.47, percebe-se dois grandes grupos. O primeiro é composto por indivíduos dos cargos de nível gerencial, com predominância de percepção dos gerentes de projetos; o segundo é formado indivíduos de cargos de nível administrativos, com predominância de percepção dos indivíduos que são analistas de sistemas.

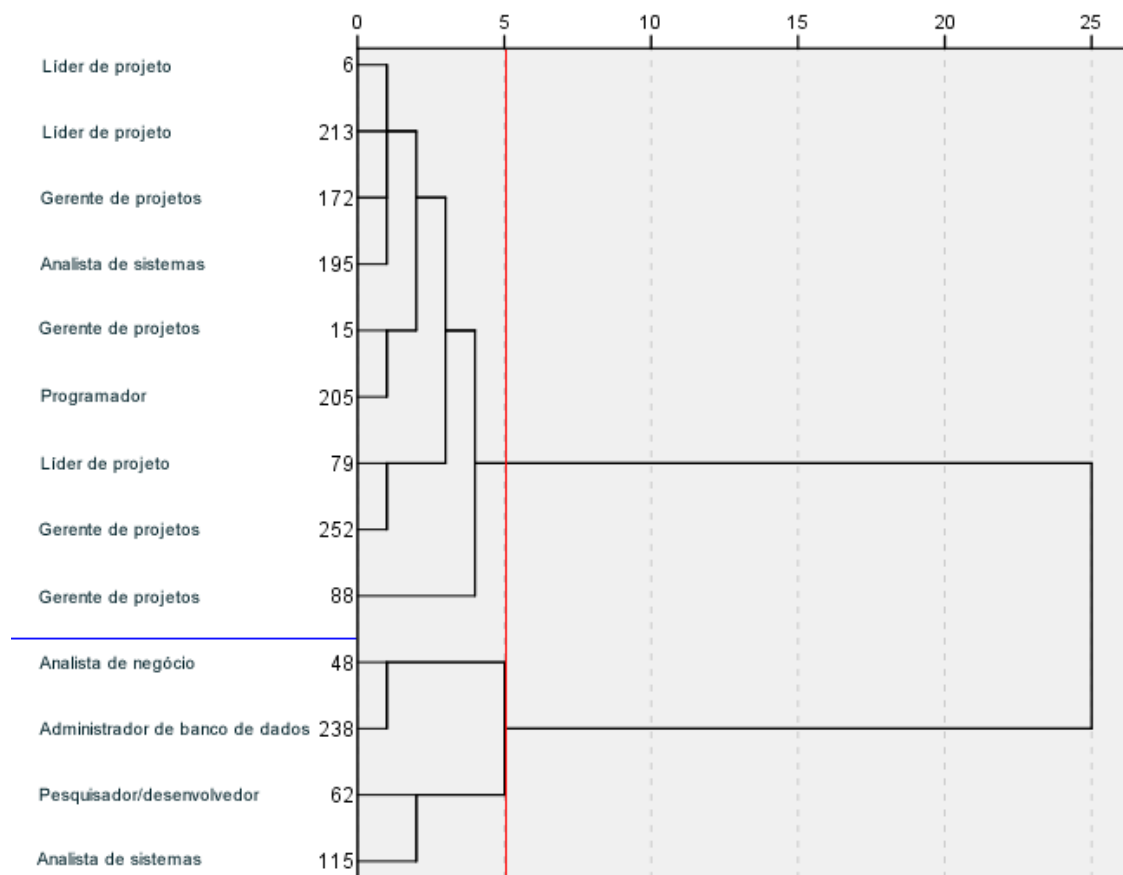


Figura 4.47 – Dendrograma da análise de cluster para o Fator 1 – Release com tempo pré-fixado, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 2 - Testes e liberação contínua:** Segundo o desenho do Dendrograma na Figura 4.48, percebe-se dois grandes grupos. O primeiro é composto por indivíduos dos cargos de nível gerencial, de alta gerência e de desenvolvimento, com predominância de percepção dos gerentes de projetos; o segundo é formado indivíduos de cargos de nível administrativos, de desenvolvimento e pesquisa, com predominância de percepção dos indivíduos que são programadores.

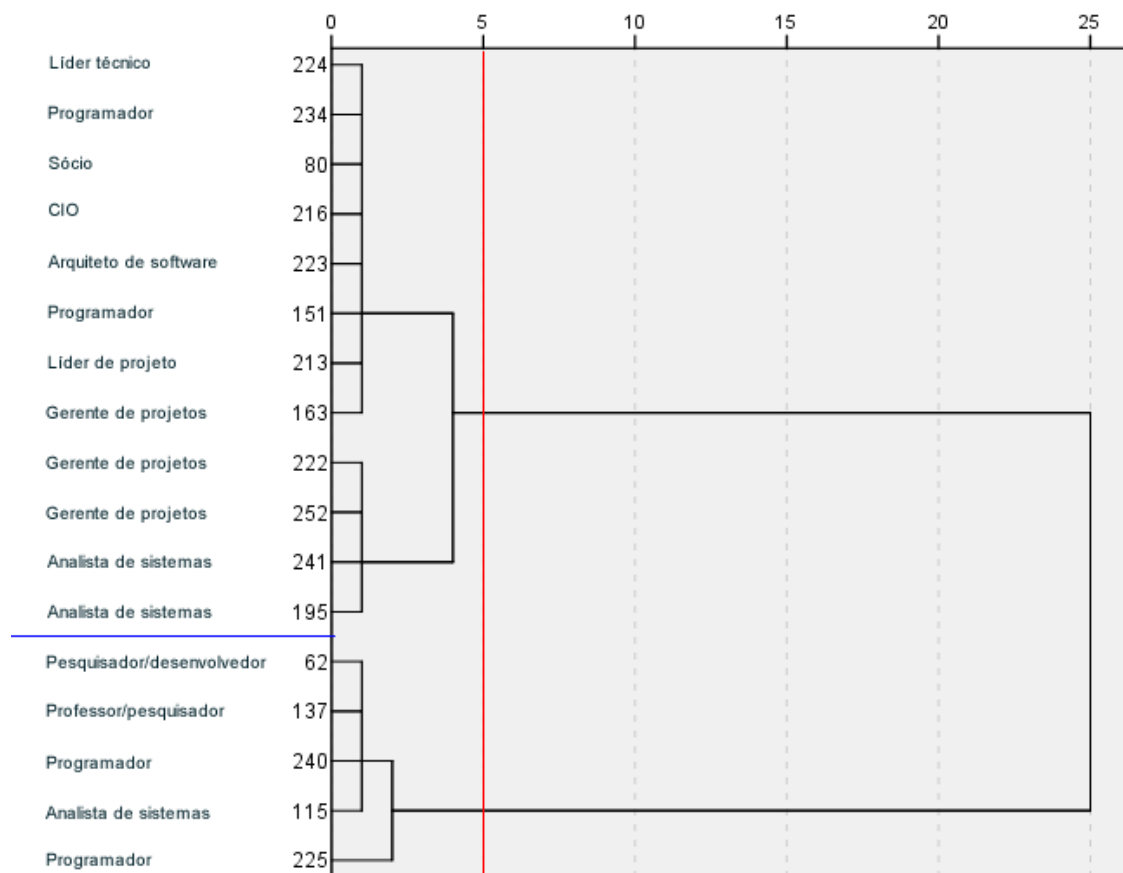


Figura 4.48 – Dendrograma da análise de cluster para o Fator 2 – Testes e liberação contínua, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 3 - Equipes de liderança:** De acordo com o Dendrograma desenhado na Figura 4.49, percebe-se dois grandes grupos. O primeiro é composto por indivíduos dos cargos de nível gerencial, de alta gerência e de desenvolvimento, com predominância de percepção dos gerentes de projetos; o segundo é formado indivíduos de cargos de nível



gerencial, alta gerência, administrativos, de desenvolvimento e pesquisa, não havendo predominância de percepção de nenhum cargo específico.

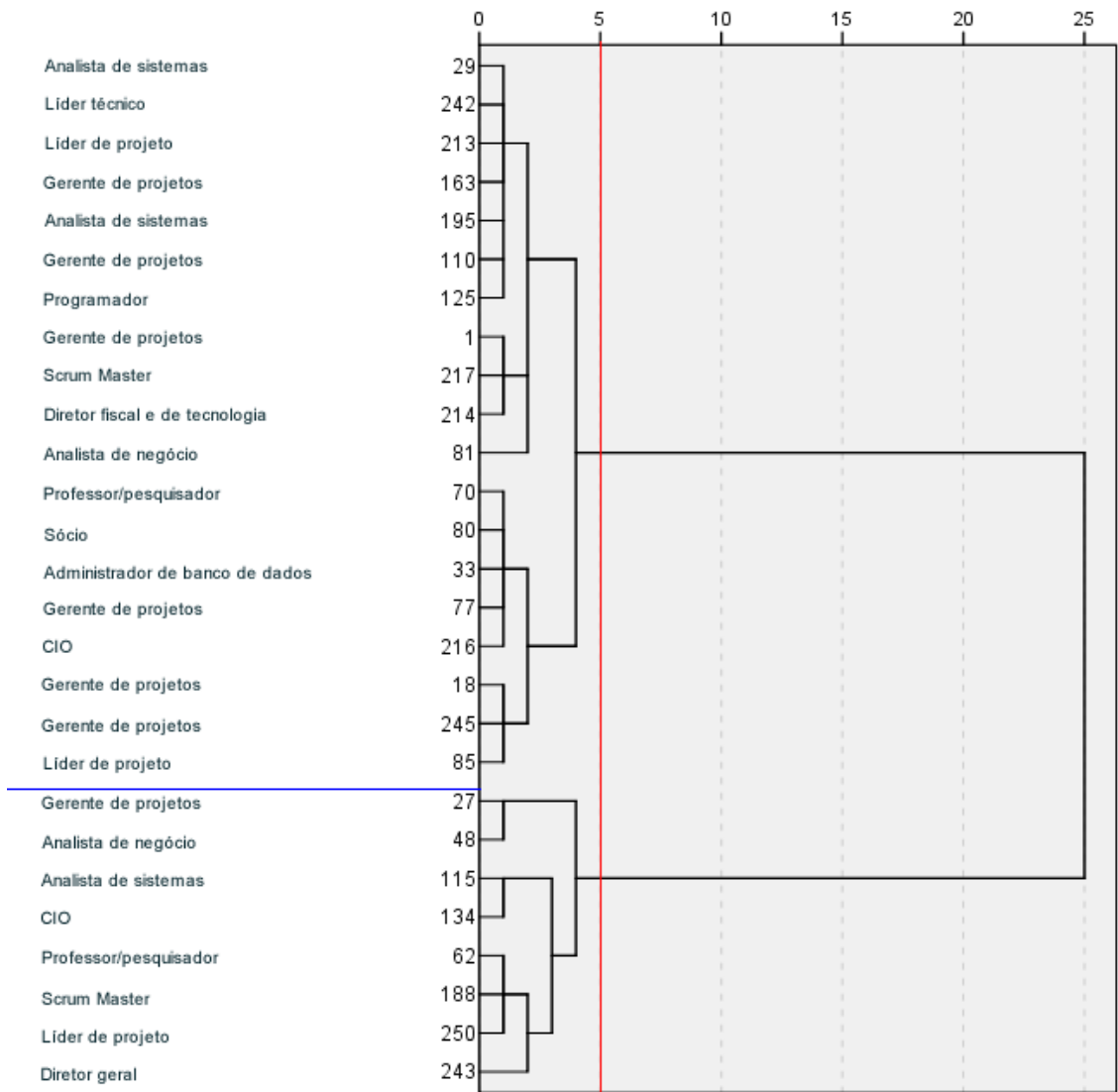


Figura 4.49 – Dendrograma da análise de cluster para o Fator 3 – Equipes de liderança, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 4 - Histórias de usuário:** Na Figura 4.50, percebe-se quatro grupos com cargos formados através do Dendrograma. No primeiro grupo há predominância de percepção de indivíduos com cargos de nível gerencial, administrativo e técnico, com predominância de percepção de valor do gerente de projetos; o segundo grupo é composto por indivíduos de cargos de nível gerencial, administrativo e pesquisa, sem predominância

de percepção do gerente de projetos; o terceiro grupo é composto por cargos de nível gerencial, com predominância de percepção do gerente de projetos; e o quarto grupo possui indivíduos de cargos de liderança e também programadores, com predominância de percepção do gerente de projetos, do CIO e dos programadores.

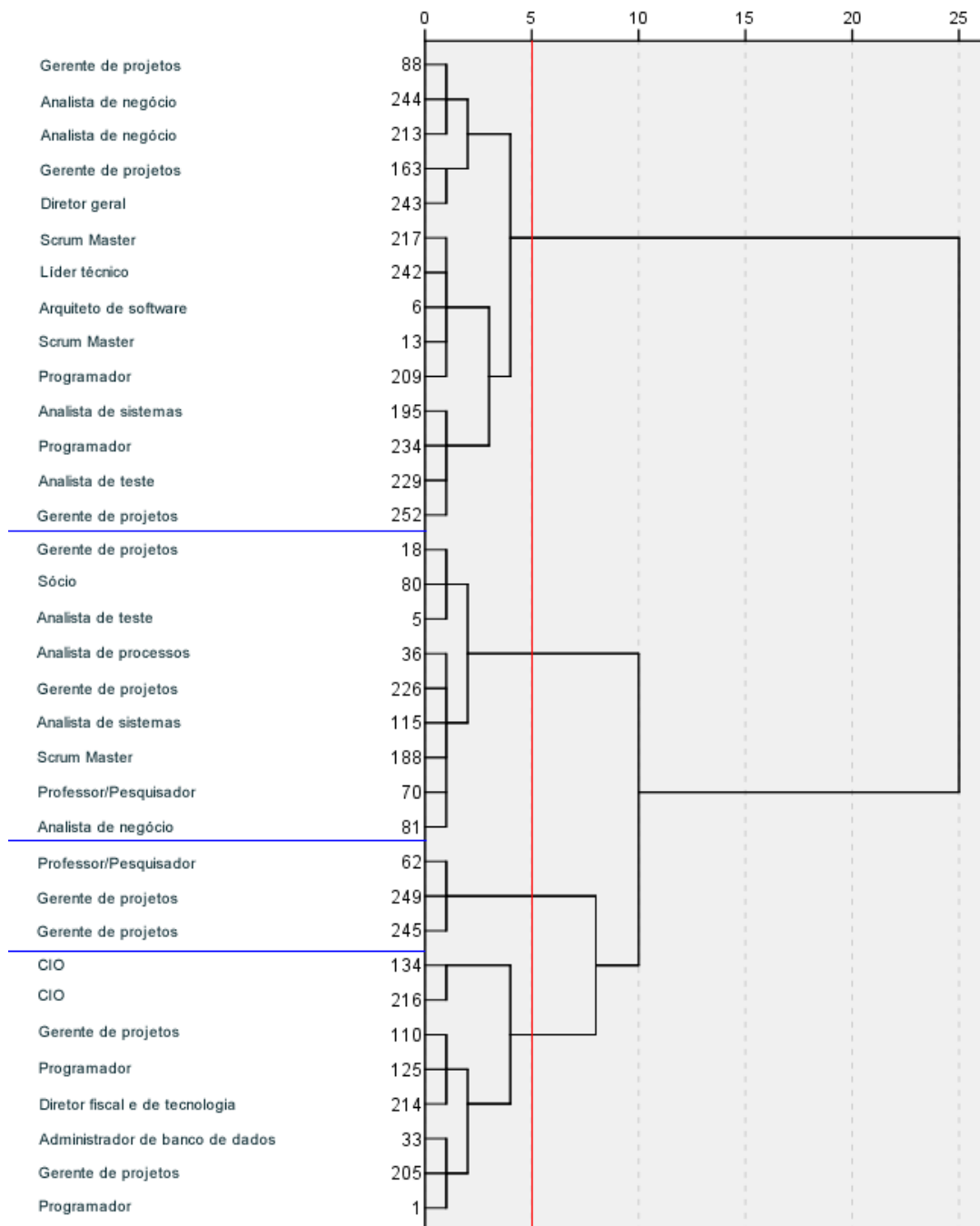


Figura 4.50 – Dendrograma da análise de cluster para o Fator 4 – Histórias de usuário, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 5 - Documentação representativa:** No Dendrograma gerado na Figura 4.51, percebe-se seis grupos com cargos bem heterogêneos. No primeiro grupo há predominância de percepção de indivíduos com cargos de alta gerência, com predominância de percepção de valor do CIO; o segundo grupo é composto por indivíduos de cargos de nível gerencial e de liderança, sem predominância de percepção do gerente de projetos; o terceiro grupo é composto por cargos de nível gerencial e de pesquisa, com predominância de percepção do gerente de projetos; e o quarto grupo possui indivíduos de cargos de gerência, com predominância de percepção do gerente de projetos; o quinto grupo representa cargos do nível de gerência e alta gerência com predominância da percepção dos gerentes de projeto; o último grupo contém indivíduos com cargos de liderança e técnicos, sem predominância de percepção.

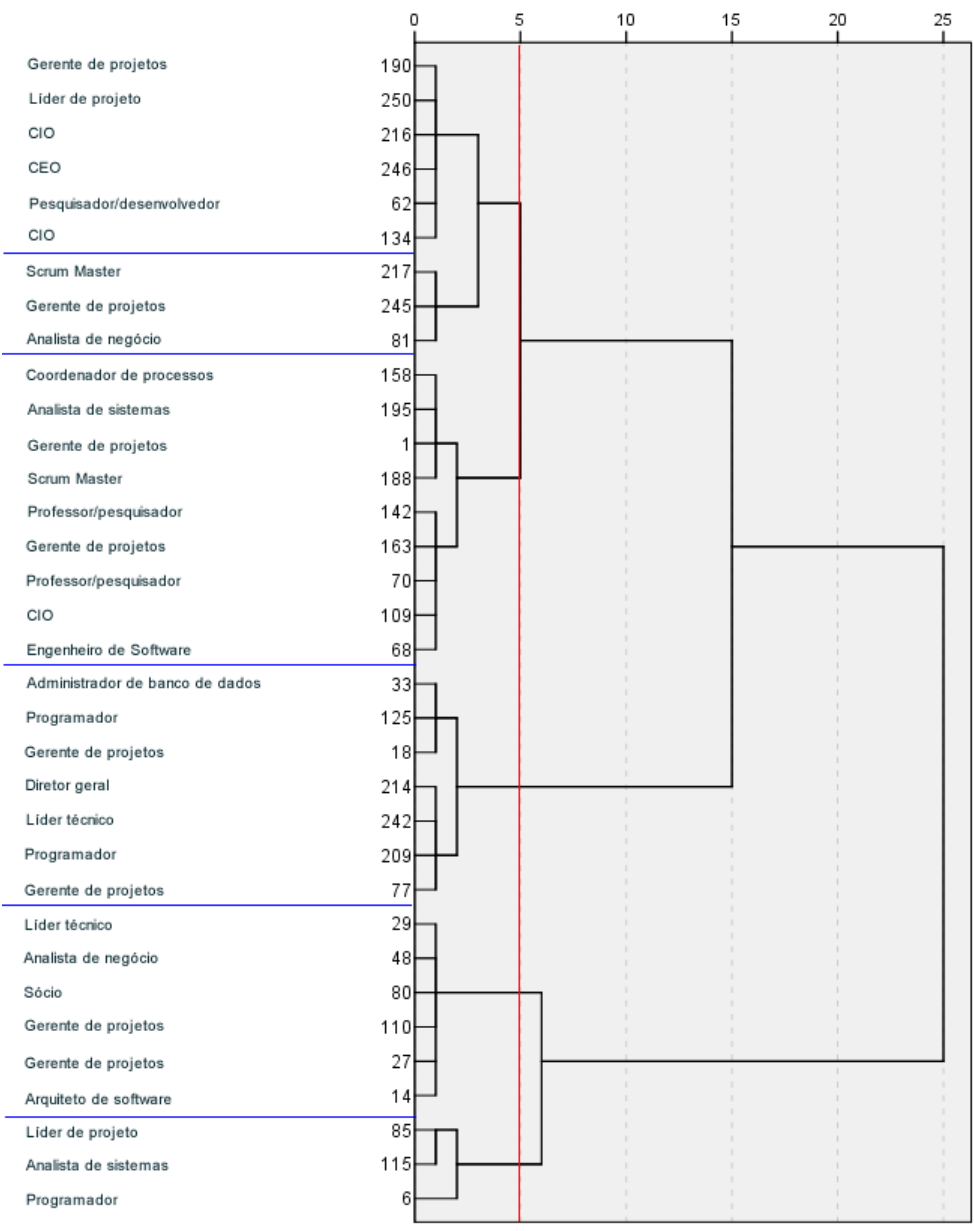


Figura 4.51 – Dendrograma da análise de cluster para o Fator 5 – Documentação representativa, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 6 - Diagramação:** Segundo o Dendrograma na Figura 4.52, percebe-se cinco grupos. No primeiro grupo contém indivíduos em cargos de gerência, técnicos e de pesquisa, sem predominância de percepção para nenhum cargo; o segundo grupo é composto por indivíduos de cargos de nível gerencial, com predominância de percepção do gerente de projetos; o terceiro grupo é composto por cargos de nível gerencial e de pesquisa, com predominância de percepção do gerente de projetos; o quarto grupo possui

indivíduos de cargos de nível administrativo e de gerência, sem predominância de percepção para nenhum cargo; e o quinto grupo representa cargos do nível de gerência e de desenvolvimento com predominância da percepção dos programadores.

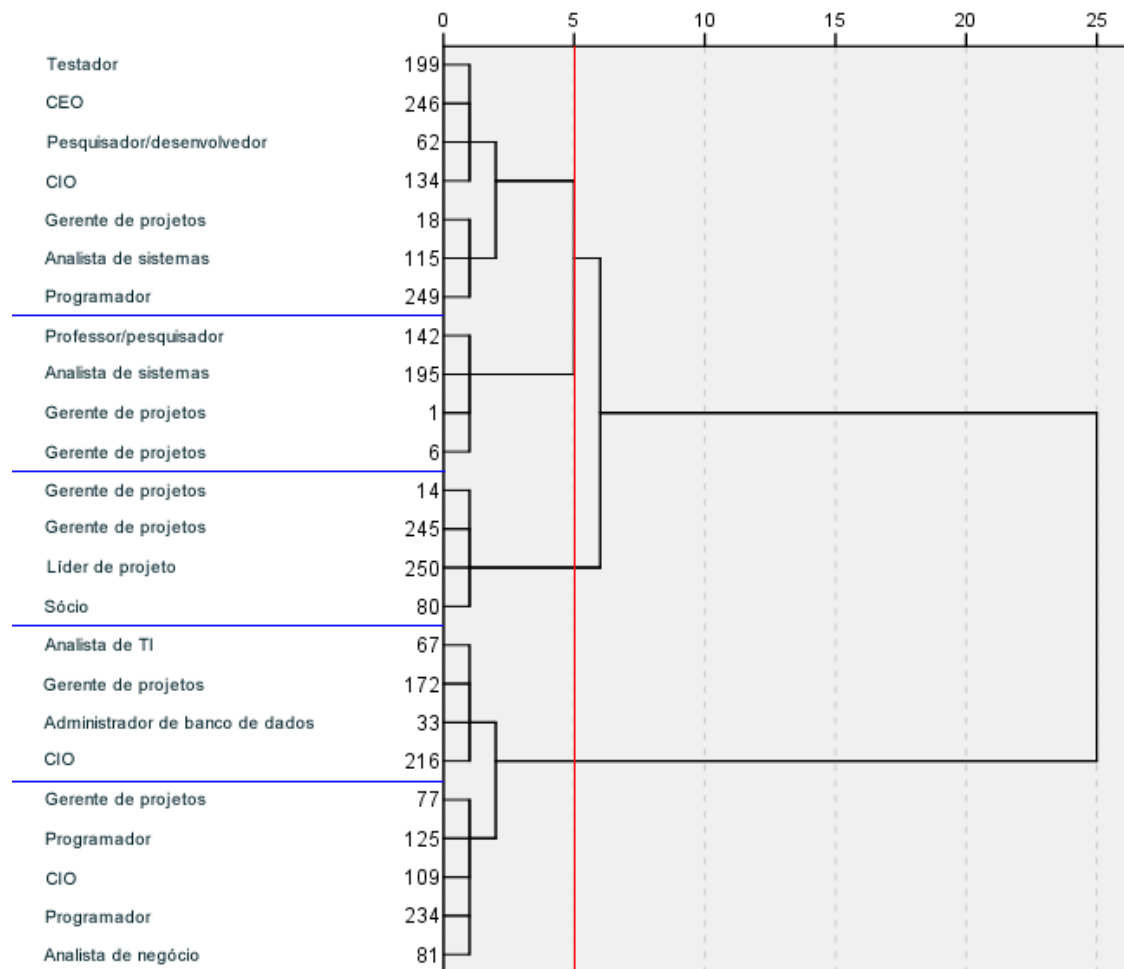


Figura 4.52 – Dendrograma da análise de cluster para o Fator 6 – Diagramação, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 7 - Desenvolvedor solo:** De acordo com o Dendrograma na Figura 4.53, com percebe-se quatro grupos. No primeiro grupo contém indivíduos em cargos de gerência, técnicos e de pesquisa, com predominância na percepção dos gerentes de projeto; o segundo grupo é composto por indivíduos de cargos de nível gerencial e de alta gerência, com predominância de percepção do gerente de projetos; o terceiro grupo é composto por cargos de nível técnico, gerencial e de alta gerência, com predominância de percepção dos

programadores; e o quarto grupo possui indivíduos de cargos de nível administrativo e de gerência, com predominância da percepção dos analistas de sistemas.

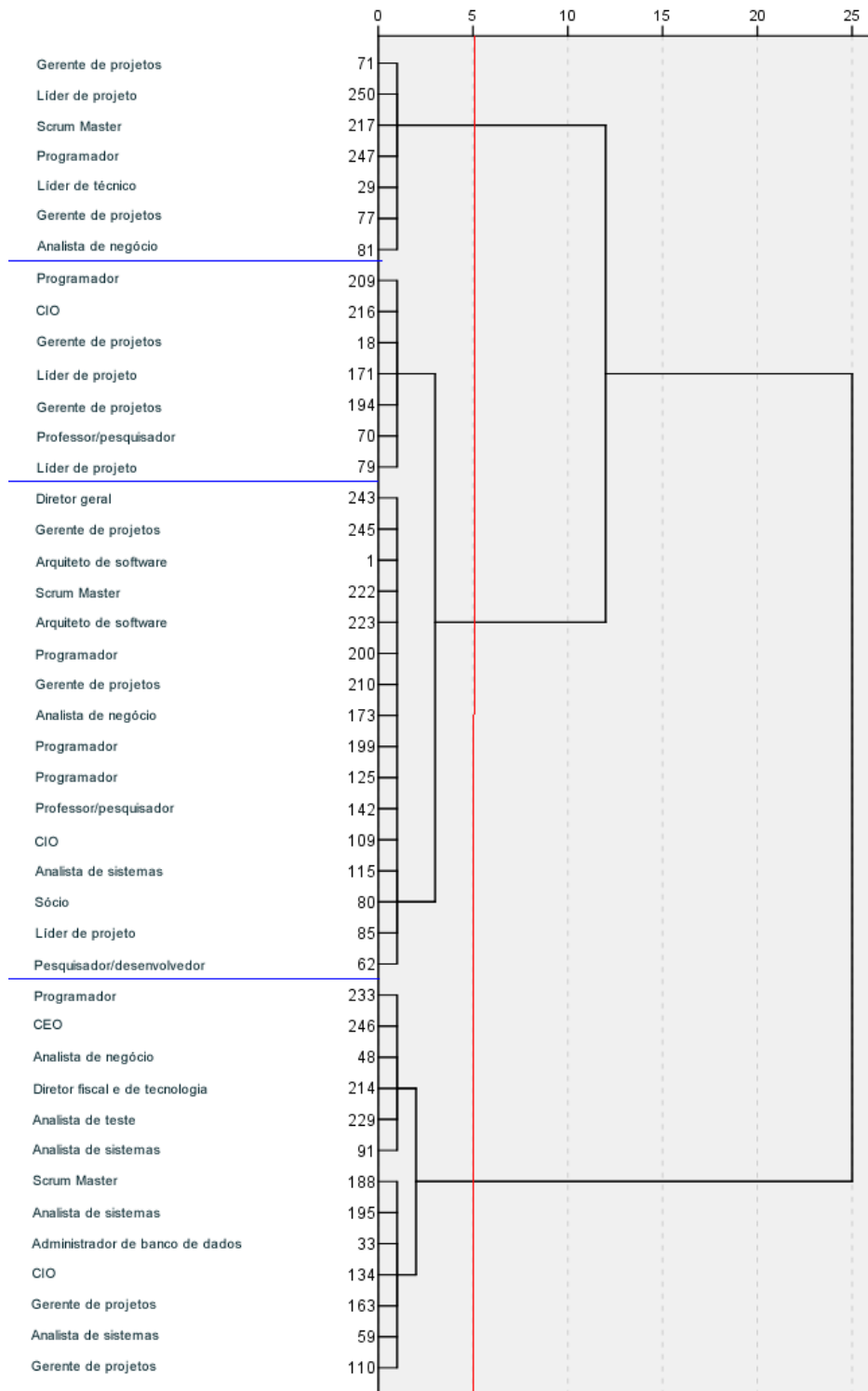


Figura 4.53 – Dendrograma da análise de cluster para o Fator 7 – Desenvolvedor solo, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

#### 4.4.2.4 Escopo

Para a análise de cluster do critério custo, foram utilizados apenas os fatores gerados das práticas de maior representatividade amostral. A partir desta análise, foi possível a identificação da relação entre o perfil do usuário e os fatores encontrados.

**Fator 1 - Desenvolvimento baseado em casos de teste:** No Dendrograma especificado na Figura 4.54, percebe-se dois grupos. No primeiro grupo contém indivíduos em cargos de nível administrativo, técnico e de gerência, com predominância na percepção dos gerentes de projeto; o segundo grupo é também composto por indivíduos de cargos de nível gerencial, técnico e administrativo, com predominância de percepção do programador.

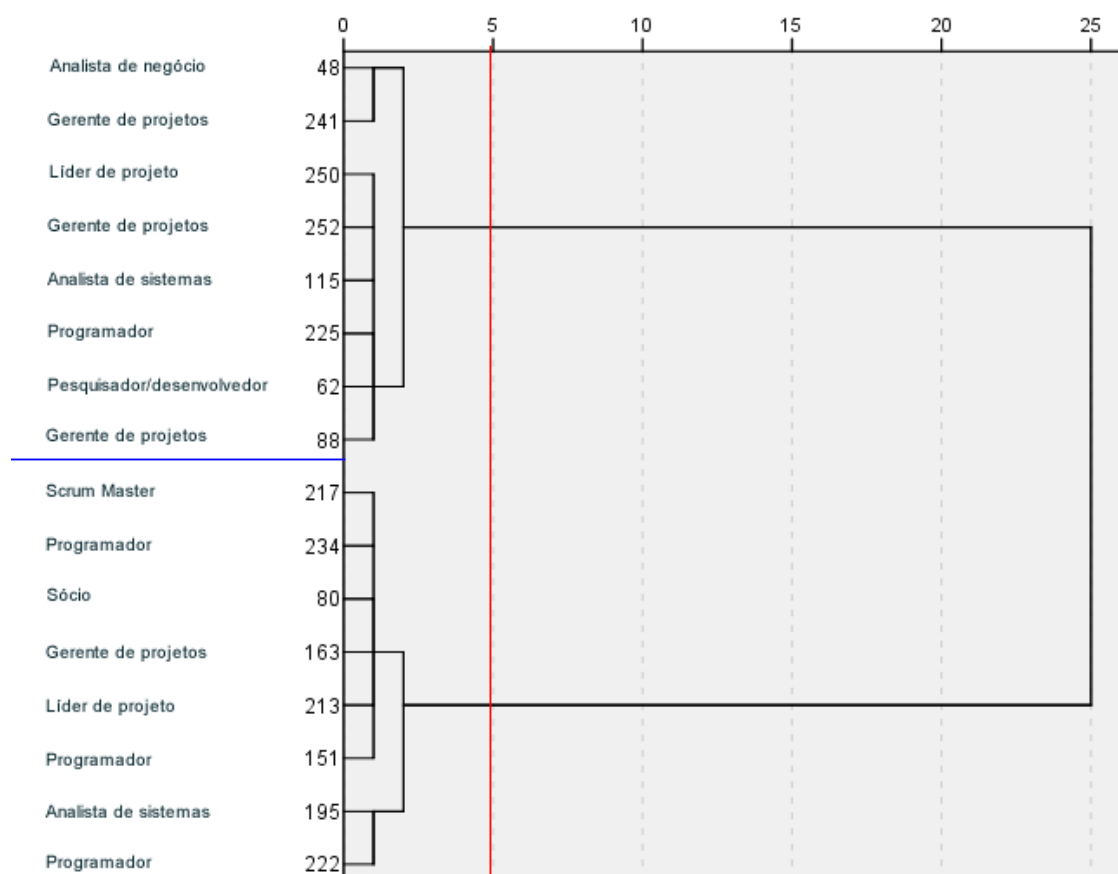


Figura 4.54 – Dendrograma da análise de cluster para o Fator 1 – Desenvolvimento baseado em casos de teste, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 2 - Diagrama de erros:** Na Figura 4.55, o Dendrograma gerado descreve dois grupos. No primeiro grupo contém indivíduos em cargos de nível administrativo, de gerência e de alta gerência, com predominância na percepção dos gerentes de projetos; o segundo grupo é também composto por indivíduos de cargos de nível gerencial, administrativo e de alta gerência, com predominância de percepção do CIO.

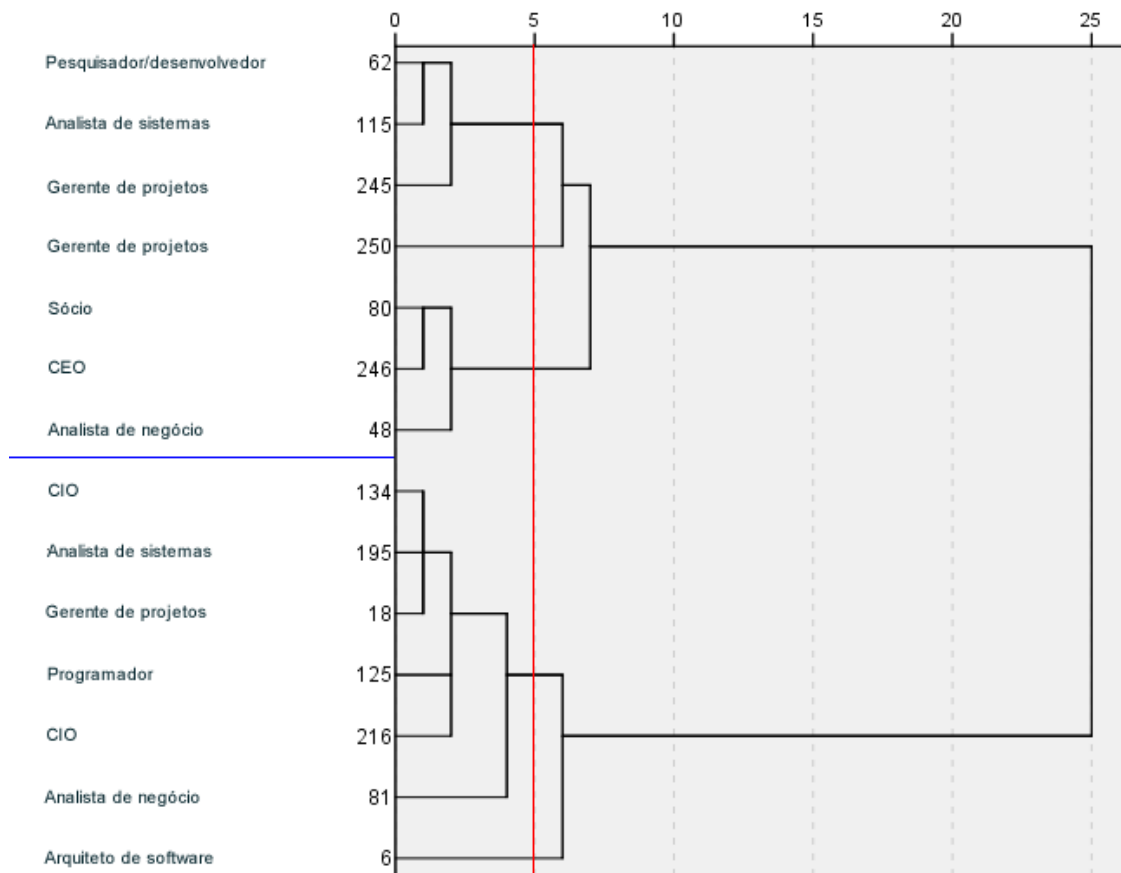


Figura 4.55 – Dendrograma da análise de cluster para o Fator 2 – Diagrama de erros, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 3 - Equipes pequenas lideradas pelo facilitador:** Para este fator o Dendrograma descrito na Figura 4.56, percebe-se três grupos. No primeiro grupo contém indivíduos em cargos de nível administrativo, técnico e de gerência, com predominância na percepção dos gerentes de projetos; o segundo grupo é também composto por indivíduos de cargos de nível gerencial, técnico e administrativo, com predominância de percepção do



programador; por fim, o terceiro grupo possui indivíduos de cargos de nível gerencial, técnico e administrativo, com predominância de percepção do gerente de projetos.

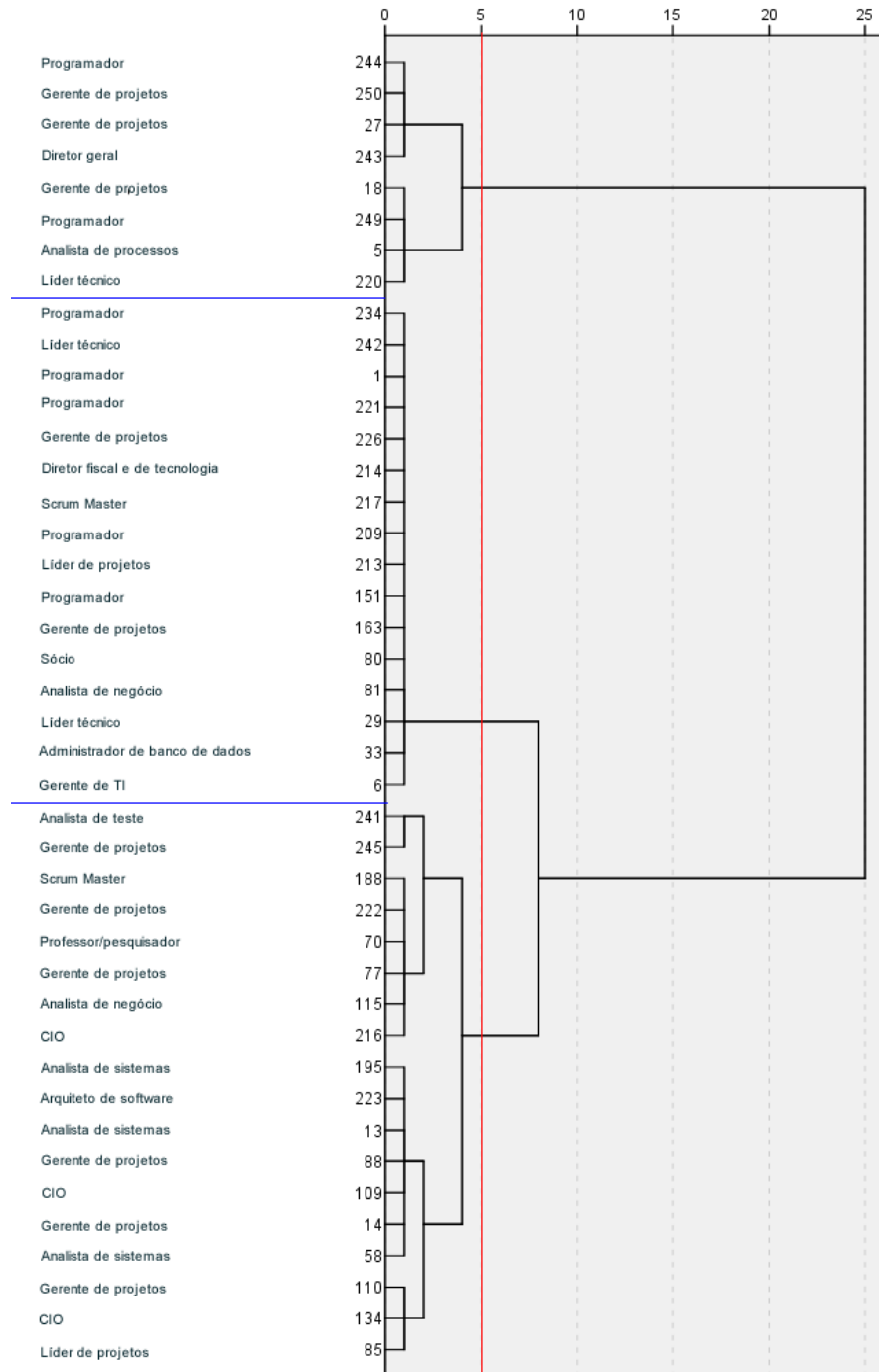


Figura 4.56 – Dendrograma da análise de cluster para o Fator 3 – Equipes pequenas lideradas pelo facilitador, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 4 - Funcionalidades definidas pelo cliente:** Segundo o Dendrograma descrito na Figura 4.57, percebe-se três grupos. No primeiro grupo contém indivíduos em cargos de nível administrativo, técnico e de gerência, com predominância na percepção dos analistas de sistemas; o segundo grupo é também composto por indivíduos de cargos de nível gerencial, técnico e administrativo, com predominância de percepção do gerente de projetos; por fim, o terceiro grupo é composto por indivíduos de cargos de nível gerencial, técnico e administrativo, com predominância de percepção do gerente de projetos e do programador.

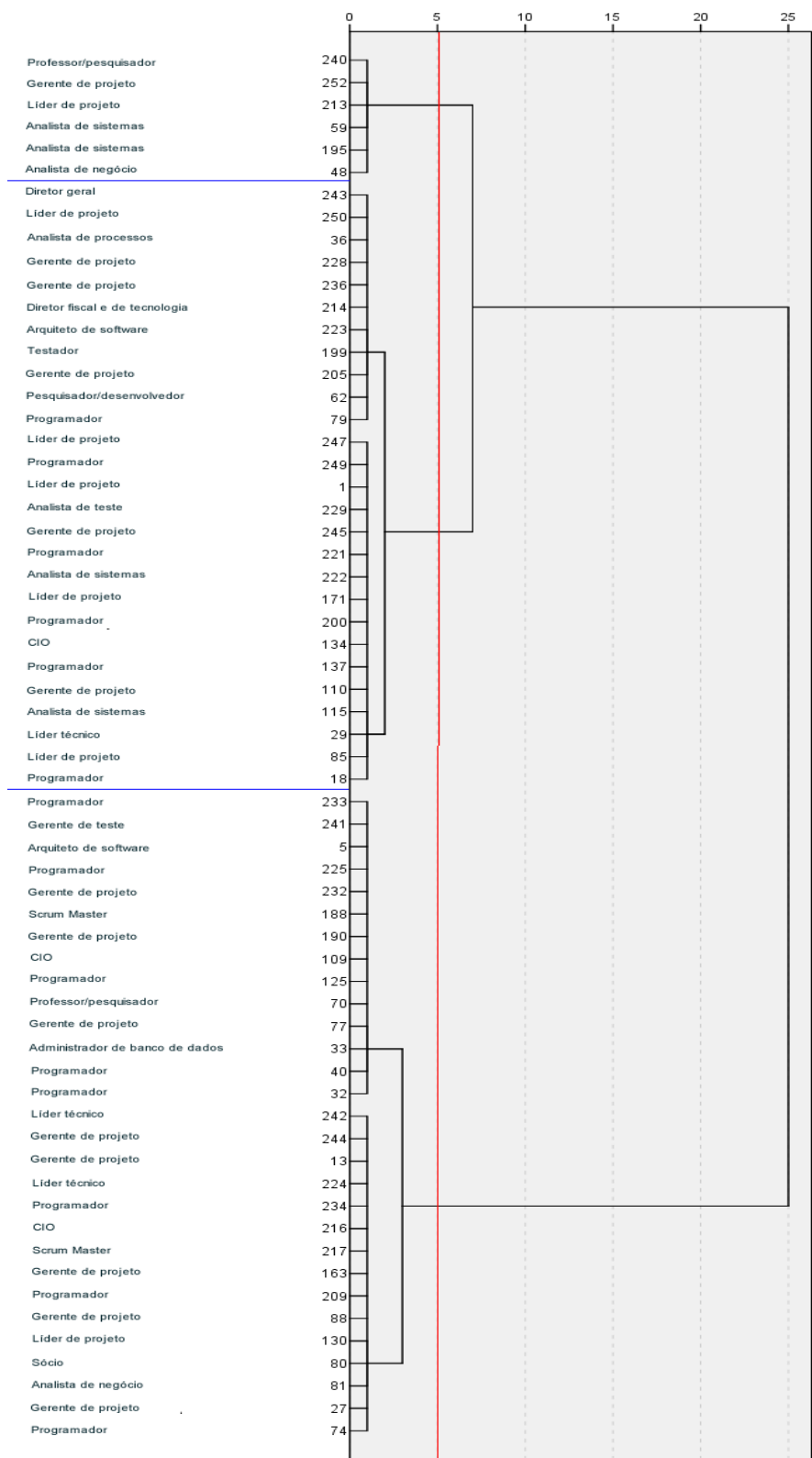


Figura 4.57 – Dendrograma da análise de cluster para o Fator 4 – Funcionalidades definidas pelo cliente, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 5 - Reuniões em pé:** De acordo com o Dendrograma da Figura 4.58, percebe-se três grupos. No primeiro grupo contém indivíduos em cargos de nível administrativo, técnico e de gerência, com predominância na percepção dos gerentes de projeto; o segundo grupo é também composto por indivíduos de cargos de nível gerencial, alta gerência, técnico e administrativo, com predominância de percepção do gerente de projetos, do líder de projetos e do analista de sistemas; por fim, o terceiro grupo é também composto por indivíduos de cargos de nível gerencial, técnico e administrativo, com predominância de percepção do gerente de projetos, arquitetos de software e dos analistas de sistemas.

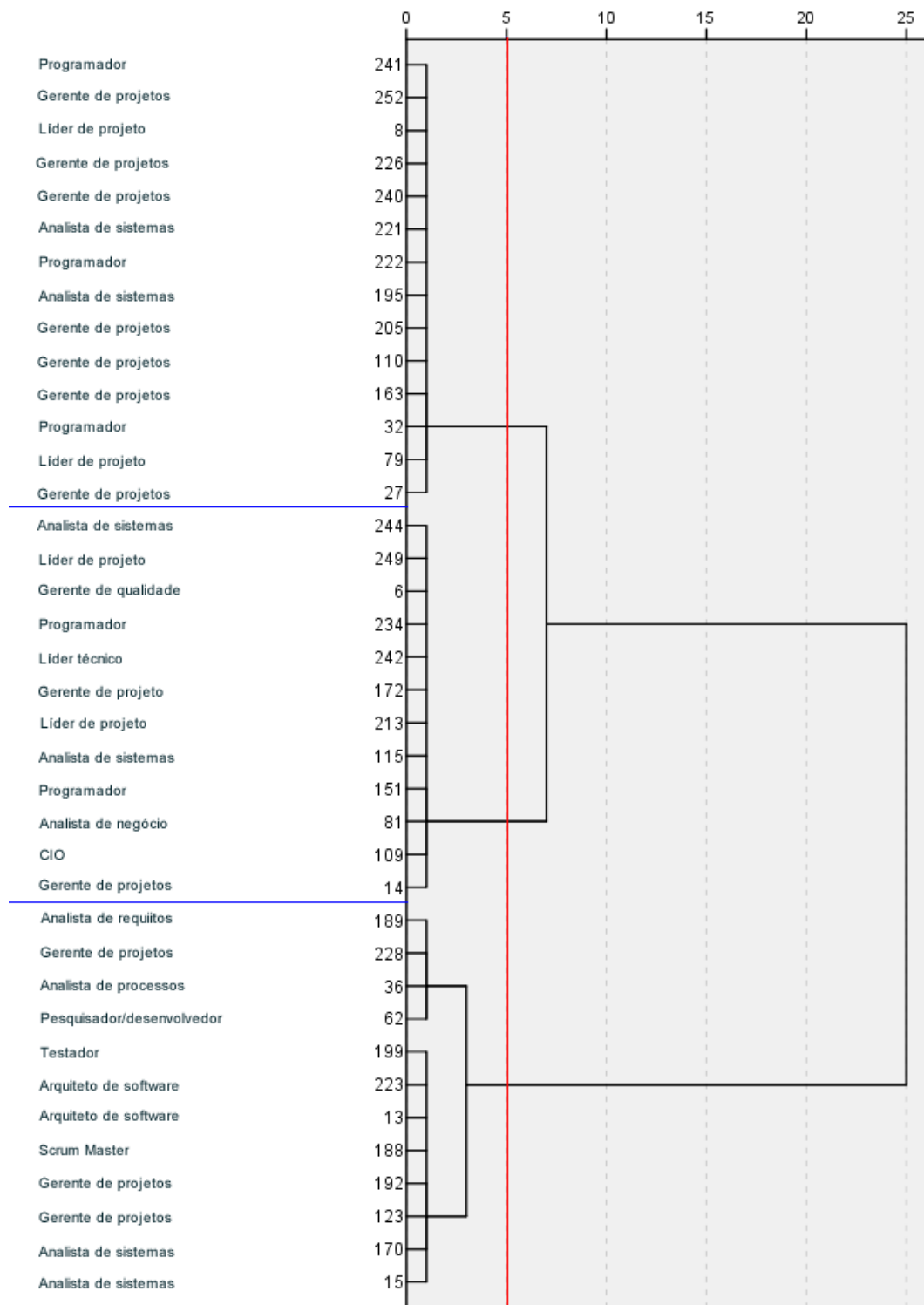


Figura 4.58 – Dendrograma da análise de cluster para o Fator 5 – Reuniões em pé, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 6 - Releases frequentes:** No Dendrograma desenhado na Figura 4.59, percebe-se três grupos. No primeiro grupo contém indivíduos em cargos de nível administrativo, técnico e de gerência, com predominância na percepção dos gerentes de projeto; o segundo grupo é também composto por indivíduos de cargos de nível gerencial, técnico e administrativo, com predominância de percepção do programador; por fim, o terceiro grupo é também composto por indivíduos de cargos de nível gerencial, alta gerência, técnico e administrativo, com predominância de percepção do gerente de projetos e do CIO.

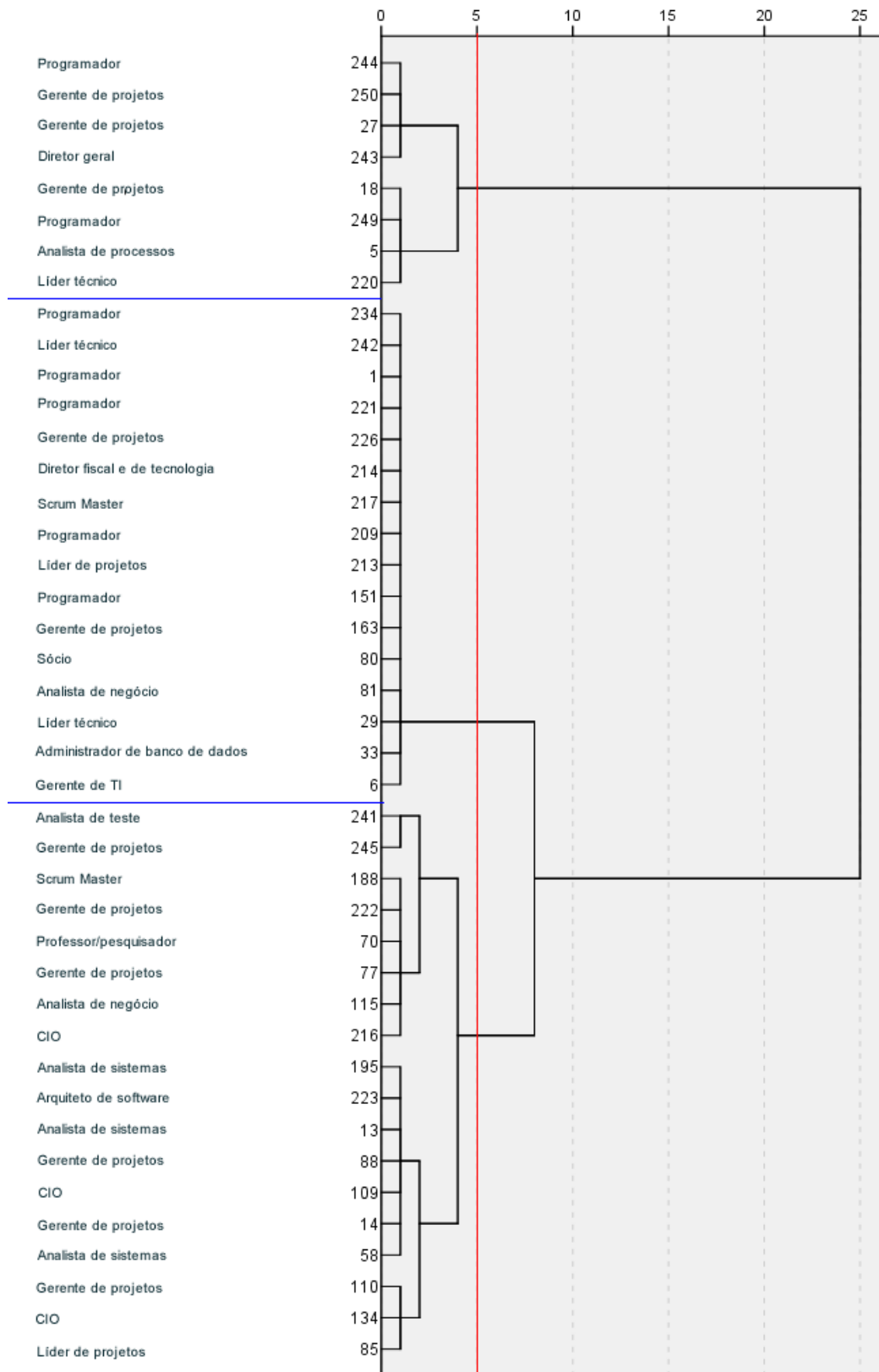


Figura 4.59 – Dendrograma da análise de cluster para o Fator 6 – Releases frequentes, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

**Fator 7 - Programador-líder:** Na Figura 4.60, o Dendrograma gerado mostra três grupos de percepção semelhante. No primeiro grupo contém indivíduos em cargos de nível administrativo, alta gerência, de gerência e de pesquisa sem predominância na percepção de valor; o segundo grupo é também composto por indivíduos de cargos de nível gerencial, alta gerência, técnico e administrativo, com predominância de percepção do gerente de projetos; por fim, o terceiro grupo é também composto por indivíduos de cargos de nível gerencial, alta gerência, técnico e administrativo, com predominância de percepção do gerente de projetos.



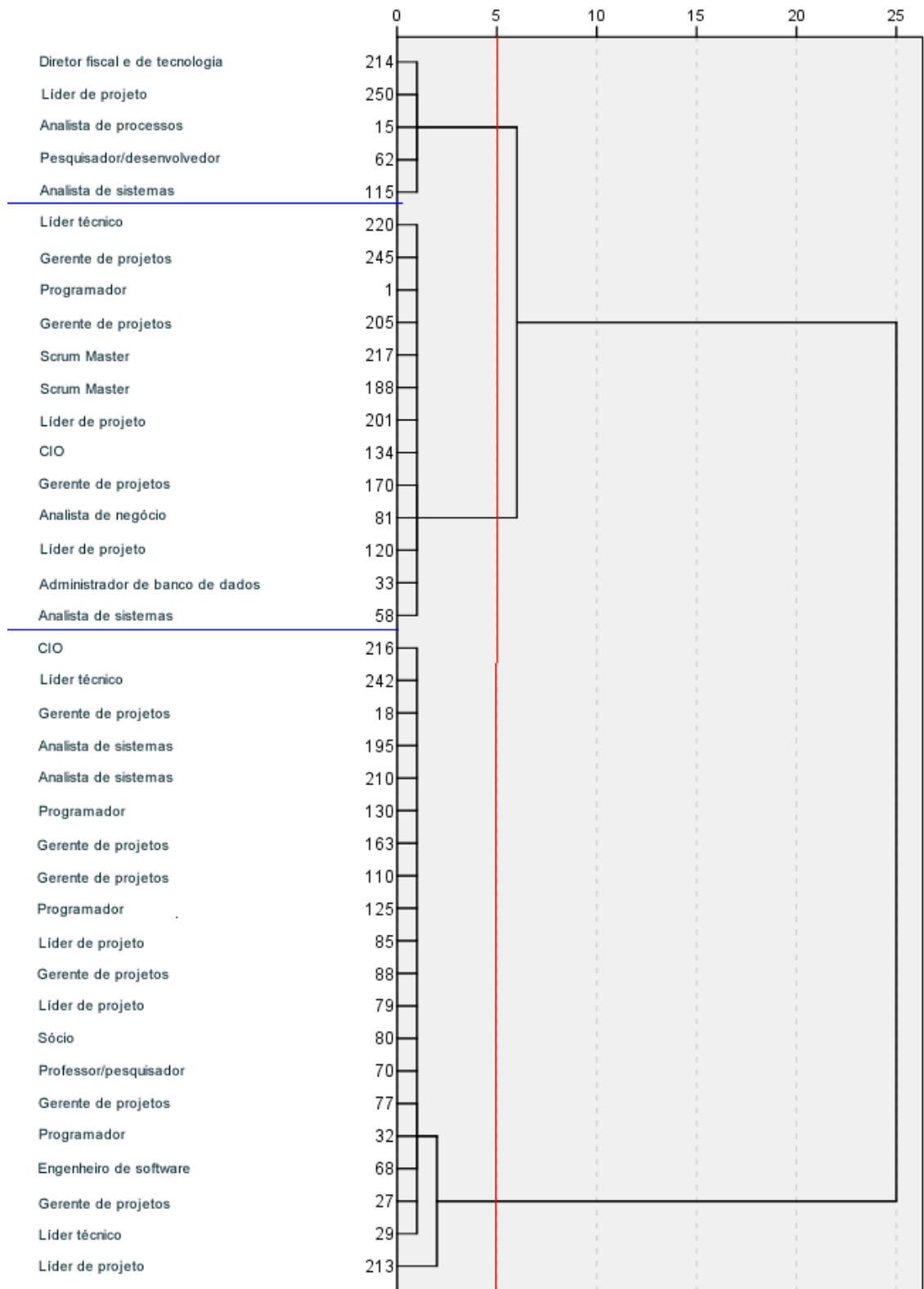


Figura 4.60 – Dendrograma da análise de cluster para o Fator 7 – Programador-líder, com o método de Ward usando a distância euclidiana quadrada como medida de semelhança.

### 4.4.3 Discussão

A estrutura relacional das práticas ágeis foi avaliada pela Análise Fatorial, que possui caráter exploratório, sendo assim, o objetivo da análise foi explorar as correlações entre as práticas ágeis e gerar fatores, dimensões ou possíveis cenários em que a utilização conjunta dessas práticas, nos fatores encontrados, com o intuito de obter informações mais relevantes para descrever o comportamento das variáveis estudadas, ou seja, o comportamento percebido pelos respondentes sobre as práticas ágeis.

Para o critério Custo, de maneira geral, podemos observar dois possíveis cenários: o cenário 1, advindo da amostragem representativa, pode-se observar a formação de equipes pequenas e multifuncionais, comumente classificadas de 2 a 7 integrantes, que utiliza histórias de usuário e lista de prioridades para definir requisitos, reuniões diárias e *Kanban* para monitoramento de tarefas, desenvolve códigos baseado em casos de teste que depois de testados e refatorados, são integrados de forma contínua.

Este cenário, cujo comportamento é discutido e explicado em fatores de redução de custos e, representam, na percepção dos respondentes que a utilização destas práticas, simplificadas nos fatores mencionados nos itens anteriores, pode ser determinante nos custos de um projeto.

Um segundo cenário observado é formado para explicar o fator Desenvolvedor Solo. Esse fator pode hipoteticamente representar uma valorização da aplicação das práticas ágeis em equipes formadas por apenas um desenvolvedor, responsável por todas as fases de execução do projeto.

Já para o critério Qualidade é possível desenhar outro possível cenário: formado pela utilização de práticas da análise representativa, onde equipes multifuncionais com um representante do cliente definem um modelo de negócio estruturado em diagramas e um documento de visão do cliente em relação ao negócio na reunião de planejamento do ciclo.

Uma vez realizados os detalhes do planejamento e arquitetura, o incremento é desenvolvido através dos casos de teste, seguidos dos testes unitários e funcionais. Após a correção dos erros encontrados na triagem de erros, os códigos são integrados de forma contínua.

Este cenário descreve um comportamento em que práticas de planejamento e arquitetura tem uma avaliação positiva, que pode ser simplificada, discutida e explicada em fatores de aprimoramento da qualidade.

Estes fatores representam, na percepção dos respondentes que a utilização destas práticas, simplificadas nos fatores mencionados nos itens anteriores, pode ser determinante na qualidade de um projeto.

Para o critério Prazo, utilizando a amostra representativa como padrão de análise, pode-se identificar dois cenários distintos: um cenário em que as práticas reunião de planejamento, a utilização de *user story* para representar os requisitos a serem trabalhados e diagramas UML para definir funcionalidades a serem implementadas, práticas que são monitoradas pelo programador-líder, que gerencia tecnicamente o desempenho da equipe. O desenvolvimento do incremento de software seria feito baseado em casos de teste e a evolução do projeto seria monitorada pelo *Kanban*, com integração contínua e lançamento frequente de versões em ciclos fixos.

Esse cenário tem seu comportamento descrito nos fatores que representam situações em que as práticas são implementadas pelos respondentes e promovem um retorno positivo em relação ao prazo. Para esse caso, práticas que correspondem a todas as etapas do ciclo de desenvolvimento desde a formação de equipes até o lançamento de versões representam maior valor em relação ao prazo.

O outro cenário observado corresponde ao fator Desenvolvedor Solo. Esse fator pode hipoteticamente representar uma valorização da aplicação das práticas ágeis em equipes formadas por apenas um desenvolvedor, responsável por todas as fases de execução do projeto, assim como observado no cenário do fator Custo.

Por fim, para o critério Escopo, é observado um cenário representativo que se remete a utilização de equipes pequenas, com um facilitador do time responsável por solucionar os problemas da equipe com o representante do cliente e que também atua como programador-líder.

Com a utilização de um documento de visão para representar os requisitos a serem trabalhados e diagramas UML para definir funcionalidades a serem implementadas, o projeto é implementado baseado em casos de teste, com monitoramento de metas feito

através da reunião em pé, e assim que testado e refatorado, um novo incremento de software é lançado ao final do ciclo.

Segundo a percepção dos respondentes, esse conjunto de práticas, combinadas e explicadas nos fatores, pode contribuir para o melhor redirecionamento e controle do escopo do projeto.

Um ponto importante a ser observado foi a presença da prática Desenvolvimento por casos de teste em fatores dos quatro critérios. Isso pode indicar que há uma tendência dos respondentes na utilização de práticas da metodologia TDD, mesmo que não adotem oficialmente como metodologia de desenvolvimento de software de apoio às outras já utilizadas.

A prática Programador solo também se mostrou relevante na análise fatorial, pois possuiu peso fatorial elevado em dois fatores: custo e prazo. Esses resultados indicam a possibilidade dos desenvolvedores estarem experimentando a utilização de práticas ágeis em projetos solo ou pessoais.

As práticas avaliadas como amostra pouco representativa podem representar outros dois possíveis cenários hipotéticos: essas práticas são pouco utilizadas pelos respondentes da pesquisa porque não são comuns ou restritas a situações específicas; ou essas práticas já foram muito utilizadas e estão tendendo a cair em desuso, por não atenderem de forma satisfatória as necessidades dos usuários.

Pela análise de agrupamentos pode-se perceber que em apenas um critério, a semelhança da percepção de valor dos respondentes teve discrepância em relação ao cargo ocupado.

Para os fatores do critério custo, apenas nos fatores Liderança e custo, Modelagem de processos e Desenvolvimento solo, os indivíduos dos cargos de CIO e programadores apresentaram percepções semelhantes.

O fato de que um indivíduo do cargo CIO não estar presente na etapa de desenvolvimento e não estar envolvido nas atividades da equipe pode apresentar uma barreira no entendimento de que estas práticas podem contribuir no controle de custos. Outro motivo pode ser a falta de resultados de uso destas práticas em sua organização.

Quanto à relação dos cargos e os fatores, em geral, podemos ver que a percepção que o usuário possui de um fator não é exclusiva ao tipo de cargo, como alta gerência, administração, gerência e desenvolvimento.

A percepção provavelmente deve variar conforme o conhecimento ou prévia utilização da prática em projetos passados.

O que se pode observar é que determinados fatores tiveram uma predominância na semelhança de percepção dos gerentes de projeto, que segundo o tamanho da amostra estudada, representa o cargo com maior número de respondentes. Portanto, existe uma maior possibilidade de que pessoas deste cargo avaliem percepções de valor.

Por fim, esse resultado é coerente com a proposta ágil, uma vez que a idéia é identificar práticas que retornem valor ao negócio e ao cliente, com toda a equipe da organização envolvida no cumprimento das metas e na filosofia ágeis.

A partir das análises realizadas tornou-se possível a proposta de um modelo para o aprimoramento do gerenciamento e desenvolvimento ágil, baseado na percepção do usuário.

## **4.5 Proposta de modelo de aprimoramento do gerenciamento e desenvolvimento ágil baseado na percepção de valor do usuário (*User-based Perception Model – UBPM*)**

Por meio dos resultados da execução da análise fatorial das práticas ágeis mapeadas durante a etapa de análise da amostra representativa, o presente trabalho propõe um modelo de aprimoramento do gerenciamento e desenvolvimento ágil baseado na percepção de valor do usuário (*User-based Perception Model – UBPM*).

Esse modelo contém um conjunto de conhecimentos explicitados em fatores de impacto em relação aos critérios de desempenho de projetos custo, qualidade, prazo e escopo e orientações para a aplicação em projetos de desenvolvimento de software.

Em suma, o *Agile UBPM* consiste em orientar os praticantes na adaptação de seu processo de desenvolvimento, aplicando fatores práticos direcionados ao atendimento das necessidades e deficiências enfrentadas no projeto.

Como metodologia base do modelo proposto, foi escolhida a metodologia *Scrum*, uma vez que os resultados mostraram que os respondentes da pesquisa a utilizam em sua maioria, o que evidência sua popularidade e, portanto, maior aceitação da proposta do modelo.

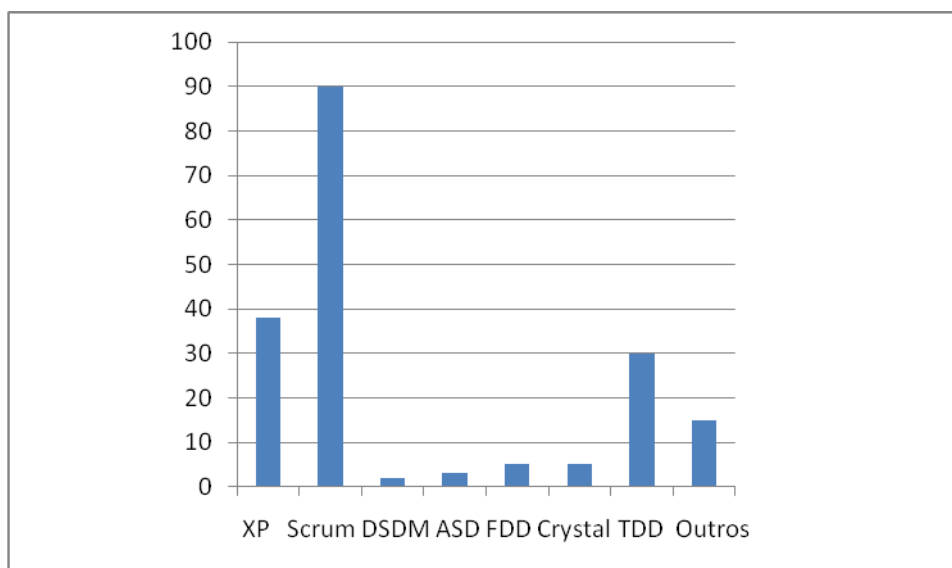


Figura 4.61 – Número de respondentes e as metodologias por eles utilizadas

## Agile UBPM for Scrum

O *Agile UBPM for Scrum* é um modelo adaptativo para a metodologia *Scrum*, com desenvolvimento iterativo e incremental. A proposta foi baseada nos resultados da análise fatorial com a amostra de práticas com número representativo de percepções de valor.

Com isso, o modelo possui quatro perspectivas de aplicação: perspectiva em custo, perspectiva em qualidade, perspectiva em prazo e perspectiva em escopo.

- **Perspectiva em custo:** representa o conjunto de práticas mais utilizadas no processo ágil e que contribuem para redução de custos no projeto. No modelo *Agile UBPM*, a perspectiva em custo é representada por dois fatores: Interação de equipe e Liderança em custo. A formação de equipes pequenas e de características multifuncionais, ou seja, com membros que

possuem habilidades variadas, pode influenciar positivamente no andamento de projeto e no poder criativo da equipe. Quando se tem papéis e liderança bem definidos, a equipe criativa e alto desempenho passam a ser uma estratégia tanto na redução de custos para estimação de tarefas, custos de planejamento e mudanças, que afetam diretamente no orçamento do projeto.

- **Perspectiva em qualidade:** agregam as práticas mais utilizadas pelos usuários no processo ágil, e, contribuem para manutenção da qualidade no projeto. No modelo *Agile UBPM*, a perspectiva em qualidade está relacionada ao fator Prevenção de erros em casos de teste. O desenvolvimento das funcionalidades estruturado em casos de teste, ou seja, criar testes antes de programar a funcionalidade propriamente dita, previne que o programador libere versões do software sem que as funcionalidades estejam devidamente codificadas e oferece oportunidades de melhorias e simplificação do mesmo. Assim, o código liberado é garantidamente funcional, adequado para que os testes de aceitação com o cliente sejam bem executados e os erros advindos destes testes melhor entendidos.
- **Perspectiva em prazo:** nesta perspectiva estão agrupadas as práticas mais utilizadas no processo ágil que contribuem para o atendimento dos requisitos do projeto dentro prazo. No *Agile UBPM*, a perspectiva em prazo é representada por dois fatores: Diagramação e Testes e liberação contínua. A elaboração de diagramas contribui na tradução das histórias de usuário (*user story*) em requisitos técnicos para que a equipe tenha conhecimento técnicas que serão utilizadas no processo de implementação. A prática de integração contínua de código no processo de desenvolvimento contribui em agregar todo o trabalho da equipe de forma, e, diante de um erro, o membro da equipe responsável por ele, seja capaz de corrigir de forma mais rápida e eficaz. Se os requisitos são entendidos e o processo de correção de erros é rápido e eficaz, a perspectiva prazo é atendida com sucesso.
- **Perspectiva em escopo:** representa o conjunto de práticas mais utilizadas no processo ágil e que contribuem para redução do escopo no projeto. No modelo *Agile UBPM*, a perspectiva em escopo é representada por três

fatores: Reuniões em pé, Funcionalidades definidas pelo cliente e Desenvolvimento por casos de teste. As funcionalidades a serem implementadas devem ser priorizadas segundo o interesse do cliente e a facilidade de implementá-las no projeto. Com a elaboração de uma lista de requisitos prioritários, desenvolvidos em ciclos curtos, consegue-se garantir que o projeto entregue é funcional e possui os principais requisitos do cliente implementados. Esses requisitos podem ser refatorados, para gerar um código mais legível e coeso. De maneira gradual, o escopo do projeto é reduzido e passa a se concentrar em tarefas essenciais para o sistema. A realização destas tarefas pode ser monitorada pelas Reuniões diárias, que ajudará a equipe entender suas dificuldades e o andamento do projeto.

A formação das perspectivas foi sugerida utilização de alguns dos fatores encontrados. O critério de seleção de práticas para o *Agile UBPM for Scrum* foi verificação de práticas que apareceram com maior frequência nos fatores formados e foram dispostos segundo a sua descrição e coerência da utilização das práticas em projetos de software e aderência com a metodologia *Scrum*.

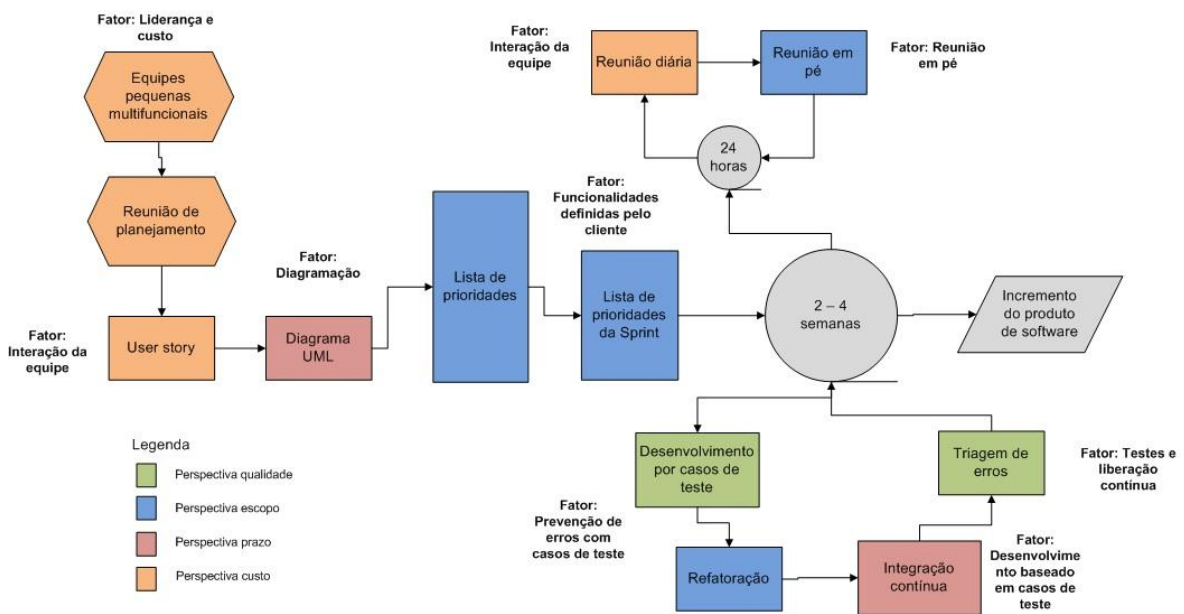


Figura 4.62 – Visão geral do ciclo de vida do Agile UBPM for Scrum



A Figura mostra o ciclo de vida do *Agile UBPM for Scrum*. O primeiro passo é definir uma equipe pequena (de 2 a 7 integrantes no máximo) e multifuncional, obedecendo o fator Liderança e custo que sugere a participação dos papéis de Representante do Cliente, um Facilitador do Time que pode também atuar como programador-líder e o próprio Cliente, participando da iteração.

Em seguida é feita a Reunião de Planejamento da do ciclo de trabalho ou da *Sprint*, onde são elaborados *user stories*, que são o detalhamento dos requisitos em forma de histórias contadas pelo cliente. Essas histórias são interpretadas para diagramas básicos de funcionalidades, que serão brevemente descritas e priorizadas na Lista de prioridades do projeto, segundo as necessidades do cliente.

Selecionadas as funcionalidades da *Sprint* e definido o período de ciclo (2 a 4 semanas), a equipe deve desenvolver baseando-se em casos de teste, com triagem de erros advindos dos testes de aceitação, refatoração de código e integração contínua, com monitoramento de integrações falhas e bem sucedidas.

Para controlar as atividades do projeto, a equipe deverá fazer reuniões diárias em pé, em um horário fixo até o final do projeto.

Ao final de cada ciclo ou *Sprint*, a equipe entrega um incremento de software funcional para o cliente.

Nesse sentido, o *Agile UBPM for Scrum* auxilia equipes ágeis a se organizarem na utilização de práticas que o próprio usuário reconhece com impacto positivo nos critérios de desempenho de projetos.

## 5 CONCLUSÃO

A agilidade se faz cada vez mais importante no atual cenário de desenvolvimento de software. E diante de um mercado competitivo, buscar soluções ágeis e adaptativas pode representar um caminho para entregar software de forma rápida, com qualidade, dentro do prazo, com escopo de trabalho reduzido e custos adequados.

O objetivo deste trabalho em questão foi investigar e analisar o uso de práticas das metodologias ágeis para o desenvolvimento de sistemas de software. Este objetivo foi alcançado uma vez que os resultados obtidos mostraram a percepção de valor dos praticantes para com os princípios e práticas ágeis utilizadas no mercado mundial.

A partir da execução do questionário e, posteriormente, a análise dos dados obtidos, verificou-se que as práticas ágeis podem ser agrupadas em fatores para o aprimoramento de processos de gerenciamento e desenvolvimento já existentes dentro do contexto organizacional, corroborando na elaboração do modelo de boas práticas baseadas na percepção do usuário para a metodologia *Scrum*, o *Agile UBPM for Scrum*.

O modelo proposto constitui-se em um resultado importante para a comunidade ágil, pois demonstram com base estatística que é possível correlacionar práticas das várias metodologias ágeis para o contexto de melhoramento de estratégias de desempenho em projetos de software.

Ao combinar práticas avaliadas pelo usuário como positivas para o sucesso nos critérios custo, qualidade, prazo e escopo, o trabalho mostrou que o foco na escolha de melhores práticas ágeis e não de uma metodologia específica, pode ser um passo de maturidade para organizações adotarem a cultura ágil com menos riscos e barreiras.

Entretanto, algumas práticas não puderam ser avaliadas em virtude da baixa quantidade de votos, o que de certa forma, reduz o nível de abrangência de práticas utilizadas. Esse comportamento pode ser justificado pelo número de respondentes ou pelo desconhecimento dos mesmos quanto à utilização destas práticas.

Para os pesquisadores em Engenharia de Software, esse trabalho contribui na obtenção de um mapeamento das práticas ágeis referenciando a pluralidade de métodos existentes no mercado e abrindo maiores possibilidades na pesquisa de percepção de valor do usuário para com práticas ágeis.

Para os praticantes de metodologias ágeis, esse trabalho pode auxiliar na identificação de quais práticas podem ser mais adequadas considerando as suas necessidades e estratégias organizacionais.

Na área de Estatística, o estudo colabora na certificação de que a pesquisa em Engenharia de Software pode ser verificada e validada, ou seja, que as informações obtidas no presente trabalho, podem ser certificadas e verdadeiramente utilizadas, testadas e aprimoradas por pesquisadores interessados no tema abordado.

Em síntese, o presente estudo contribui para área de Engenharia de Software, incentivando ainda mais pesquisas quantitativas e no intuito de que as organizações que utilizam metodologias ágeis, no desenvolvimento de produtos de software, possam cada vez mais direcionar melhor seus investimentos, recursos e esforços para atingir a excelência dos seus processos.

## 5.4 Trabalhos futuros

Com aplicação do questionário, outros tipos de dados foram coletados, mas que não faziam parte do escopo do trabalho. Assim, como trabalhos futuros sugerem-se:

- Análises de dados complementares obtidos com o questionário: avaliação dos benefícios e dificuldades de adoção de metodologias ágeis.
- Desenvolver análises estatísticas mais aprofundadas, como a utilização de métodos de análise confirmatória.
- Expansão do modelo Agile UBPM a partir dos resultados deste trabalho, para propor melhorias para outras metodologias estudadas como: XP, TDD, ASD, DSDM e *Crystal methodologies*.
- Verificação do modelo *Agile UBPM* e aprimoramento do mesmo.

## 6 REFERÊNCIAS BIBLIOGRÁFICAS

ABRAHAMSSON, P.; SALO, O.;RONKAINEN, J.; WARSTA, J. Agile software development methods: review and analysis, **VTT Technical report**, 2002.

ÅGERFALK,P., FITZGERALD, B. Flexible and distributed software processes: old petunias in new bowls? **Communications of the ACM** **49** (10) (2006), 27–34.

AGILE ALLIANCE, **The Twelve Principles of Agile Software**. Disponível em: <http://www.agilealliance.org/the-alliance/the-agile-manifesto/the-twelve-principles-of-agile-software/>. Acesso em: 22/05/2011.

AHMED, A., AHMAD, S., EHSAN, DR. N., MIRZA, E., SARWAR. S. Z. Agile Software Development: Impact on Productivity and Quality. In: **Proceedings of the 2010 IEEE ICMIT**, 2010.

APPOLINÁRIO, Fábio. **Dicionário de metodologia científica**: um guia para a produção do conhecimento científico. São Paulo: Atlas, 2004.

BECK, K. **Extreme Programming Explained**. Boston: Addison-Wesley Professional, 1999.

BECK, K;FOWLER,M. **Planning Extreme Programming**. Boston: Addison-Wesley Professional, 2000.

BECK et al. **Agile Manifesto**. <http://www.agilemanifesto.org/>. Disponível em: 2001. Acesso em: 25/05/10.

BIFFL, S., AURUM, A., BOEHM, B., ERDOGMUS, H.; GRÜNBACHER, P. **Value-Based Software Engineering**. Springer-Verlag, 388p., 2006.

CAO, L., MOHAN, K., XU, P., RAMESH, B. A framework for adapting agile development methodologies. **European Journal of Information Systems** (2009), v.18, pg. 332–343, 2009.

CHARETTE R. Fair Fight? Agile Versus Heavy Methodologies. In: **Agile Methodologies: the Great Debate**. Arlington: Cutter Consortium, 2002.

CHOW, T., CAO, DAC-BUU. A survey study of critical success factors in agile software projects. **The Journal of Systems and Software** v.81 (2008), pg. 961–971, 2008.

COCKBURN, A. **Agile software development**. Reading, MA: Addison-Wesley, 2002.

COCKBURN, A. **Crystal Clear: A Human-Powered Methodology for Small Teams**. Addison-Wesley Professional, pages 336, 2004.

DSDM Atern: the Agile Project Framework. **DSDM Atern** – Disponível em <http://www.dsdm.org/dsdm-atern>. Acesso em: 22/04/2011.

DYBÅ, T.; DINGSØYR, T. Empirical studies of agile software development: A systematic review. **Informatics. Software Technology**. 50, 9-10 (Aug. 2008), 833-859

EXTREME PROGRAMMING. **The Rules of Extreme Programming** – Disponível em <http://www.extremeprogramming.org/rules.html>. Acesso 22/04/2011.

FREEMAN, S. PRYCE, N. **Growing Object-Oriented Software, Guided by Tests**. Addison-Wesley Signature Series, 2010.

FUGGETTA, A. “Software Process: A Roadmap”. In: **22nd International Conference on the Future of Software Engineering, Limerick, Ireland: ACM -Association for Computing Machinery**, p. 25-34, 2000.

GESTÃO DE PROJETOS DEBLM. **Proposta de Gestão de Projetos para DEBLM** – Disponível em <http://code.google.com/p/gestao-de-projeto-deblm/wiki/DesenvolvimentoNoSERPROHistoricoEProcessosAgeis>. Acesso em: 22/04/2011.

GIL, Antonio Carlos. **Métodos e técnicas de pesquisa social**. 5. ed. São Paulo: Atlas, 1999.

HAIR, JOSEPH F. ET AL. **Análise multivariada de dados**. 6. ed. Porto Alegre: Bookman, 2009. x, 688 p.

HEPTAGON. **O Que É FDD?** – Disponível em <http://www.heptagon.com.br/fdd-oque>. Acesso em: 22/04/2011.

HIGHSMITH, J. **Agile Software Development Ecosystems**. Foreword by Tom DeMarco. Addison-Wesley Pearson Education, 2002.

ILIEVA, S.; IVANOV, P.; STEFANOVA, E. Analyses of an agile methodology implementation, In: **Proceedings 30th Euromicro Conference, IEEE Computer Society Press**, pp. 326–333, 2004.

JOHNSON, Richard A., WICHERN, Dean W. **Applied multivariate statistical analysis**. Prentice Hall. 800pp. Edition 6, 2007.

JUNG, C. F. **Metodologia para pesquisa & desenvolvimento aplicada a novas tecnologias, produtos e processos**. Rio de Janeiro: Axcel Books, xvi, 312 p., 2004.

KOSCIANSKI, A. SOARES, M.S. **Qualidade de Software: Aprenda as metodologias e técnicas mais modernas para o desenvolvimento de software**. 2ª Edição. 2007.

- KÖTLER, P. **Administração de marketing**. 10. ed. São Paulo: Prentice Hall, 2000 764 p.
- LEE, G., XIA, W. Toward Agile An Integrated Analysis Of Quantitative And Qualitative Field Data On Software Development Agility. **MIS Quarterly** Vol. 34 No. 1, pp. 87-114. March, 2010.
- MALHOTRA, Naresh K. **Pesquisa de Marketing: uma orientação aplicada**. Editora Bookman, Porto Alegre, 2001.
- MANN, C., MAURER, F. A case study on the impact of scrum on overtime and customer satisfaction. **Agile Development Conference**, p. 70-79. IEEE Computer Society. 2005.
- MARCONI, Marina de Andrade; LAKATOS, Eva Maria. **Fundamentos de metodologia científica**. 5. ed. São Paulo: Atlas, 2003.
- MAROCO, J. **Análise Estatística com a utilização do SPSS**. 3ª Ed. Silabo. Lisboa. 822 pp, 2007.
- MINGOTI, S. A.; **Análise de dados através de métodos de estatística multivariada: uma abordagem aplicada**, Editora UFMG, 2005.
- OLIVEIRA, F. E. M. de, **SPSS Básico para Análise de Dados**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2007.
- PALMER S. R., FELSING J. M., **A Practical Guide to Feature-Driven Development (The Coad Series) (1stEdition)**, Prentice Hall PTR, USA, 2002.
- PFLEEGER, S. L.; ATLEE, J. M. **Software engineering: theory and practice**. 3rd ed. Upper Saddle River, NJ: Pearson, 2006.
- PINEYWOODS TECH, **Our Methodology**. Disponível em: <http://pineywoodstech.com/Services/Methodology.aspx>. Acesso em: 22/04/2011.
- PRESSMAN, R. S. **Engenharia de software**. 6. ed. São Paulo: McGraw-Hill, 2006 xxxi, 720 p.
- SANTOS, M.A., GREGHI, J.G., BERMEJO, P.H.S. The Impact of Scrum in Software Development: A Case Study using SWOT Analysis. **INFOCOMP Journal of Computer Science**. Special, v.2, p.65—71, 2010.
- SCHWABER, K. **Agile Project Management with Scrum**. Microsoft Press, 2004;
- SCHWABER, K., BEEDLE, M. **Agile Software Development with Scrum**. Prentice Hall, 2002.

SCRUM ALLIANCE. **What is Scrum?** - Disponível em <http://www.scrumalliance.org/>. Acesso em: 24/05/10.

SILLITTI, A.; CESCHI, M.; RUSSO, B.; SUCCI, G. Managing uncertainty in requirements: a survey in documentation-driven and agile companies, In: **Proceedings of the 11th International Software Metrics Symposium (METRICS)**, 2005.

SILVA, Edna Lúcia; MENEZES, Estera Muszkat. **Metodologia da pesquisa e elaboração de dissertação**. Florianópolis: [S. n.], 2000.

SHORE, J., WARDEN, S. **The Art of Agile Development**. O'Reilly Media, 2007.

SOMMERVILLE, I. **Engenharia de software**. 8. ed. Rio de Janeiro: Pearson Addison Wesley, c2007.

STANDISH GROUP. **New Standish Group report shows more project failing and less successful projects**. [http://www.standishgroup.com/newsroom/chaos\\_2009.php](http://www.standishgroup.com/newsroom/chaos_2009.php). Disponível em 2009. Acesso em 01/06/2010.

STELLMAN, A.; GREENE, J. **Applied software project management**. O'Reilly Media, Inc. 308pp, 2005.

SVENSSON, H., HÖST, M. Views from an organization on how agile development affects its collaboration with a software development team, **Lecture Notes in Computer Science**, v. 3547, Springer Verlag, Berlin, 2005, pp. 487–501.

TAKEUCHI, H.; NONAKA, I. “The New New Product Development Game”, **Harvard Business Review**, 1986.

YIN, Robert K. **Estudo de caso: planejamento e métodos**. 4. ed. Porto Alegre: Bookman, 2010.

## 1. Apresentação geral da empresa

Questões com "\*" são de preenchimento obrigatório!

### 1. Perfil do respondente:

Nome:

Idade:

Nome da empresa:

### \* 2. Cargo em que ocupa dentro da empresa:

- |   |   |   |
|---|---|---|
| <input type="radio"/> <input type="radio"/> Administrador de banco de dados | <input type="radio"/> <input type="radio"/> Arquiteto de banco de dados | <input type="radio"/> <input type="radio"/> Gerente de projetos   |
| <input type="radio"/> <input type="radio"/> Administrador de redes          | <input type="radio"/> <input type="radio"/> Arquiteto de software       | <input type="radio"/> <input type="radio"/> Gerente de Teste      |
| <input type="radio"/> <input type="radio"/> Analista de negócio             | <input type="radio"/> <input type="radio"/> Auditor                     | <input type="radio"/> <input type="radio"/> Líder de projeto      |
| <input type="radio"/> <input type="radio"/> Analista de Requisitos          | <input type="radio"/> <input type="radio"/> CEO                         | <input type="radio"/> <input type="radio"/> Líder técnico         |
| <input type="radio"/> <input type="radio"/> Analista de sistema             | <input type="radio"/> <input type="radio"/> CIO                         | <input type="radio"/> <input type="radio"/> Professor/Pesquisador |
| <input type="radio"/> <input type="radio"/> Analista de suporte             | <input type="radio"/> <input type="radio"/> Cliente                     | <input type="radio"/> <input type="radio"/> Programador           |
| <input type="radio"/> <input type="radio"/> Analista de Teste               | <input type="radio"/> <input type="radio"/> Design de interfaces        | <input type="radio"/> <input type="radio"/> Testador (Tester)     |
| <input type="radio"/> <input type="radio"/> Analista de TI                  | <input type="radio"/> <input type="radio"/> Design de Teste             | <input type="radio"/> <input type="radio"/> Webdesign             |
| <input type="radio"/> <input type="radio"/> Outro (especifique):            |   |   |



# Análise de práticas de metodologias ágeis

## \* 3. Selecione uma ou mais opções que melhor descrevem os produtos desenvolvidos pela sua organização:

- |   |   |   |
|---|---|---|
| <input type="checkbox"/> <input type="checkbox"/> Aplicações móveis                             | <input type="checkbox"/> <input type="checkbox"/> Sistemas Embarcados   | <input type="checkbox"/> <input type="checkbox"/> Software de gestão administrativa<br>(Exemplo: controle de estoque, cadastros em geral, soluções financeiras)   |
| <input type="checkbox"/> <input type="checkbox"/> E-commerce                                    | <input type="checkbox"/> <input type="checkbox"/> Sistemas ERP  | <input type="checkbox"/> <input type="checkbox"/> Software de mídias sociais  |
| <input type="checkbox"/> <input type="checkbox"/> Jogos e entretenimento                        | <input type="checkbox"/> <input type="checkbox"/> Software crítico (Cujas características possuem riscos físicos, financeiros, pessoais inerentes: software para administração de produtos químicos, sistema bancário, missão, controle de pacientes médicos)         | <input type="checkbox"/> <input type="checkbox"/> Software de simulação e otimização  |
| <input type="checkbox"/> <input type="checkbox"/> Robótica                                      | <input type="checkbox"/> <input type="checkbox"/> Software de automação industrial<br>(Exemplo: Software para controle de variáveis naturais como temperatura pressão e vazão, Softwares com controladores lógicos, softwares para controle de resíduos industriais). | <input type="checkbox"/> <input type="checkbox"/> Software Educacional (Exemplo: Sistemas e-Learning, Objetos de aprendizagem, Sistema de Gestão da Aprendizagem) |
| <input type="checkbox"/> <input type="checkbox"/> Serviços de TI                                |   | <input type="checkbox"/> <input type="checkbox"/> Software Livre  |
| <input type="checkbox"/> <input type="checkbox"/> Sistemas de Hiperídia (Exemplo: mídias ricas) | <input type="checkbox"/> <input type="checkbox"/> Software de Geoprocessamento  | <input type="checkbox"/> <input type="checkbox"/> Soluções de Inteligência de Negócio   |
|   | <input type="checkbox"/> <input type="checkbox"/> Software de Gerenciamento de Rede<br>(Exemplo: Softwares de monitoramento de rede, controle de dispositivos, resolução de problemas de rede)  |   |
| <input type="checkbox"/> <input type="checkbox"/> Outro (especifique):                          |   |   |
| <input type="text"/>  |   |   |

## \* 4. Característica da empresa:

- |  |   |   |   |
|--|---|---|---|
| <input type="radio"/> <input type="radio"/> Micro (1-9 funcionários) | <input type="radio"/> <input type="radio"/> Pequena(10-49 funcionários) | <input type="radio"/> <input type="radio"/> Média (50-200 funcionários) | <input type="radio"/> <input type="radio"/> Grande(Acima de 200 funcionários) |
|--|---|---|---|

## \* 5. A empresa utiliza metodologias ágeis no seu desenvolvimento de software?

- |   |   |
|---|---|
| <input type="radio"/> <input type="radio"/> Sim | <input type="radio"/> <input type="radio"/> Não |
|---|---|

# Análise de práticas de metodologias ágeis

## 2. Análise de utilização de metodologias ágeis

### \* 6. Há quanto tempo adota metodologias ágeis em sua empresa?

Tempo Ano(s)  Mês(es)

### \* 7. Em qual estágio de adoção a empresa se encontra?

Inicial/ ad hoc   Parcialmente implementado   Definido sem medições formais   Definido com medições formais e melhoria contínua

### \* 8. Quais práticas ágeis sobre a organização de equipes e feedback são utilizadas? (Marque apenas as práticas que você utiliza. É obrigatório responder pelo menos uma prática).

Qual a sua percepção do valor de uso dessa prática em relação ao custo? Qual a sua percepção do valor de uso dessa prática em relação à qualidade? Qual a sua percepção do valor de uso dessa prática em relação ao prazo? Qual a sua percepção do valor de uso dessa prática em relação ao escopo?

	em relação ao custo?	em relação à qualidade?	em relação ao prazo?	em relação ao escopo?
Equipes Pequenas	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Equipe multifuncional	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Programador solo	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Representante do cliente	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Facilitador do time	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Programador-Líder	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Equipes de funcionalidades	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Várias equipes trabalhando em paralelo	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Pequenos times em equipes grandes	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reunião diária	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reuniões em pé (Stand Up meetings)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Reunião de planejamento da iteração	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Retrospectiva da iteração	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Lista de prioridades do produto	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Cliente participativo	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Casos de uso	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Uso de Cenários	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Documento de visão	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Documento evocativo	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
User story	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Outro (especifique):

# Análise de práticas de metodologias ágeis

## 3. Análise de utilização de metodologias ágeis - Parte 2

Nesta página serão avaliados a percepção de valor que o praticante possui sobre práticas das metodologias ágeis e o grau de utilização destas práticas ágeis na sua empresa. As práticas listadas no questionário são resultados de uma extensa revisão de literatura, realizada na primeira fase do projeto de pesquisa.

### \* 9. Quais práticas ágeis consideradas técnicas são utilizadas? (Marque apenas as práticas que você utiliza. É obrigatório responder pelo menos uma prática).

	Qual a sua percepção do valor de uso dessa prática em relação ao custo?	Qual a sua percepção do valor de uso dessa prática em relação à qualidade?	Qual a sua percepção do valor de uso dessa prática em relação ao prazo?	Qual a sua percepção do valor de uso dessa prática em relação ao escopo?
Arquitetura de soluções potenciais (Arquitetura Spike)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Modelo de domínio	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Estudo do objetivo de negócio	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Diagramas UML	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Refatoração	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Programação em pares	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Propriedade coletiva do código	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Build de 10 minutos	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Propriedade de classe individual (a classe possui um único desenvolvedor responsável)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Triagem de erros (Bug tracking)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Gestão da configuração	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Desenvolva depois teste (Test Last Development)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Teste primeiro depois desenvolva (Test-first development)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Testes funcionais (Functional Test)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Desenvolvimento baseado em casos de teste (TDD)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Testes unitários	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Automação de testes	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Refino de metodologia (da metodologia Crystal)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Integração contínua	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Outro (especifique):

# Análise de práticas de metodologias ágeis

\* **10. Quais as práticas de planejamento, controle e metas são utilizadas? (Marque apenas as práticas que você utiliza. É obrigatório responder pelo menos uma prática).**

	Qual a sua percepção do valor de uso dessa prática em relação ao custo?	Qual a sua percepção do valor de uso dessa prática em relação à qualidade?	Qual a sua percepção do valor de uso dessa prática em relação ao prazo?	Qual a sua percepção do valor de uso dessa prática em relação ao escopo?
Iterações fixas	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Gráficos de andamento	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Kanban (quadro de tarefas)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Jogos de planejamento	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Projeto Velocity	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Estimativa por funcionalidade	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Estimativa por missão	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Relatório de progresso	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Lista de verificação (Checklist)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Diagrama Parking Lot	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Lançamento frequente de versões do software	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Técnico Ágil presente (Agile Coach)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Revisão passo-a-passo (Walkthrough)	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Análise em grupo com foco na visão do cliente	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Workshop de análise	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Outro (especifique):

\* **11. Qual é o grau de satisfação do cliente quanto ao uso de metodologias ágeis?**

Muito baixo

Muito alto

\* **12. Qual é o grau de satisfação e entendimento da equipe para com o uso de metodologias ágeis?**

Muito baixo

Muito alto

\* **13. Qual ou quais metodologia(s) ágil(eis) a empresa adota?**

ASD
   Crystal
   DSDM
   FDD
   Scrum
   TDD
   XP

Methodologies

Outro (especifique):

## 4. Análise de utilização de metodologias ágeis - Parte 3

### 14. Quais dos itens abaixo você considera como dificuldades durante a adoção de uma metodologia ágil? (Marque três opções que você acredita serem mais relevantes)

- |  |  |
|--|--|
| <input type="checkbox"/> <input type="checkbox"/> Barreiras geográficas entre a equipe                   | <input type="checkbox"/> <input type="checkbox"/> Falta de reconhecimento de certificação da organização (como CMMI) |
| <input type="checkbox"/> <input type="checkbox"/> Dificuldade na seleção de uma metodologia adequada     | <input type="checkbox"/> <input type="checkbox"/> Falta de referências bibliográficas                                |
| <input type="checkbox"/> <input type="checkbox"/> Dificuldade no planejamento e gerenciamento do projeto | <input type="checkbox"/> <input type="checkbox"/> Pouca documentação   |
| <input type="checkbox"/> <input type="checkbox"/> Falta de estudos comprovando sua eficácia              | <input type="checkbox"/> <input type="checkbox"/> Requisitos contratuais com clientes                                |
| <input type="checkbox"/> <input type="checkbox"/> Falta de experiência do cliente                        | <input type="checkbox"/> <input type="checkbox"/> Requisitos de produto/serviço                                      |
| <input type="checkbox"/> <input type="checkbox"/> Falta de experiência do time                           |  |

### 15. Quais dos itens abaixo você considera potenciais benefícios na adoção de metodologias ágeis? (Marque três opções que você acredita serem mais relevantes)

- |   |   |
|---|---|
| <input type="checkbox"/> <input type="checkbox"/> Adaptabilidade  | <input type="checkbox"/> <input type="checkbox"/> Redução de custos                               |
| <input type="checkbox"/> <input type="checkbox"/> Aumento da comunicação entre a equipe                   | <input type="checkbox"/> <input type="checkbox"/> Redução de riscos                               |
| <input type="checkbox"/> <input type="checkbox"/> Aumento da qualidade do produto final                   | <input type="checkbox"/> <input type="checkbox"/> Redução do conflito de interesses com o cliente |
| <input type="checkbox"/> <input type="checkbox"/> Diminuição do escopo do projeto                         | <input type="checkbox"/> <input type="checkbox"/> Redução do tempo de entrega                     |
| <input type="checkbox"/> <input type="checkbox"/> Maior escalabilidade no gerenciamento e desenvolvimento | <input type="checkbox"/> <input type="checkbox"/> Resposta rápida a mudanças de requisitos        |

### 16. Você poderia nos indicar uma ou mais pessoas para responder este questionário?



## ANEXO B

Prezado (a) Senhor (a),

A Universidade Federal de Lavras (UFLA), por meio do Professor Paulo Henrique de Souza Bermejo, vem convidá-lo a participar do projeto de pesquisa intitulado **Análise das práticas de metodologias ágeis de desenvolvimento de software**, cujo trabalho de coleta de dados está sendo realizada pela aluna Mariana de Azevedo Santos.

Desenvolvido pelos Departamentos de Ciência da Computação e Ciências Exatas da UFLA sob a coordenação dos Professores Paulo Henrique de Souza Bermejo e Marcelo Silva de Oliveira, o projeto destina-se analisar o desempenho e a percepção de valor para com os métodos ágeis e suas práticas, segundo critérios de percepção muito alta/benéfica a percepção muito baixa/difícil, de forma a desenvolver um framework de boas práticas personalizadas com as características básicas dos segmentos empresariais estudados no projeto.

O projeto possui viabilidade e mérito científico reconhecidos pelo Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), além de contar com o financiamento deste órgão de fomento.

A participação das organizações convidadas dar-se-á mediante ao preenchimento de um questionário por parte de membros da empresa, de forma a construir um diagnóstico das práticas de metodologias ágeis da organização.

**Para participar de nossa pesquisa acesse o link abaixo:**

[https://www.surveymonkey.com/s.aspx?sm=Q3jKddKjmtHEkdjASOV\\_2bse4lMA6mu2\\_2fsar\\_2f2VbNPK1w\\_3d](https://www.surveymonkey.com/s.aspx?sm=Q3jKddKjmtHEkdjASOV_2bse4lMA6mu2_2fsar_2f2VbNPK1w_3d)

O questionário é destinado para praticantes e não praticantes de metodologias ágeis, e tem como público alvo diferentes perfis de profissionais que atuam no desenvolvimento, gestão, aquisição e uso de produtos de software. Adicionalmente, os respondentes desta pesquisa concorrerão ao sorteio do livro **“Agile Project Management: Creating Innovative Products (2nd Edition)”** <http://amzn.to/fJjaCL>. O sorteio ocorrerá no dia 30/03/2011. **Importante: para concorrer ao livro, o respondente deve informar nome ou email no questionário.**

Caso haja dúvidas sobre esse questionário ou sobre o projeto de pesquisa, entre em contato conosco através do email [mariana@bsi.ufla.br](mailto:mariana@bsi.ufla.br).

Desde já agradecemos a sua colaboração.

Atenciosamente,

Mariana de Azevedo Santos (Pesquisadora do CNPq / DCC - UFLA),  
Paulo Henrique de Souza Bermejo (Supervisor do projeto / DCC-UFLA),  
Marcelo Silva de Oliveira (Co-orientador do projeto / DEX - UFLA),  
Adriano Olimpio Tonelli (Co-orientador / DCC - UFLA).

Atenção: Se não desejar receber nossos emails, clique no link abaixo e você será removido automaticamente de nossa lista de mala direta.

<https://www.surveymonkey.com/optout.aspx>