



Aplicação de Pontos por Função em Projetos que Usam Métodos
Ágeis: Uma Análise Comparativa entre Abordagens Existentes

por

EDUARDO GARCIA WANDERLEY
Dissertação de Mestrado



Universidade Federal de Pernambuco
CIN - CENTRO DE INFORMÁTICA
PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

posgraduacao@cin.ufpe.br

www.cin.ufpe.br/~posgraduacao

RECIFE
2015

EDUARDO GARCIA WANDERLEY

Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis:
Uma Análise Comparativa entre Abordagens Existentes

Dissertação apresentada como requisito parcial para a obtenção do título de Mestre, pelo Programa de Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco.

ORIENTADOR: PROF. DR. ALEXANDRE MARCOS LINS DE VASCONCELOS

RECIFE
2015

Catálogo na fonte
Bibliotecária Joana D'Arc Leão Salvador CRB4-532

W245a Wanderley, Eduardo Garcia.
Aplicação de pontos por função em projetos que usam métodos ágeis: uma análise comparativa entre abordagens existentes / Eduardo Garcia Wanderley. – Recife: O Autor, 2015.
100 f.: fig., tab.

Orientador: Alexandre Marcos Lins de Vasconcelos.
Dissertação (Mestrado Profissional) – Universidade Federal de Pernambuco. CIN, Ciência da Computação, 2015.
Inclui referências e apêndices.

1. Engenharia de software. 2. Análise de pontos de função. I. Vasconcelos, Alexandre Marcos Lins de (Orientador). II. Título.

005.1 CDD (22. ed.) UFPE-MEI 2015-088

Dissertação de Mestrado apresentada por **Eduardo Garcia Wanderley** à Pós-Graduação em Ciência da Computação do Centro de Informática da Universidade Federal de Pernambuco, sob o título “**Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis: Uma Análise comparativa entre abordagens existentes**”, orientada pelo **Prof. Alexandre Marcos Lins de Vasconcelos** aprovada pela Banca Examinadora formada pelos professores:

Profª. Carla Taciana Lima Lourenço Silva Schuenemann
Centro de Informática / UFPE

Prof. Célio Andrade de Santana Júnior
Departamento de Ciência da Informação / UFPE

Prof. Alexandre Marcos Lins de Vasconcelos
Centro de Informática / UFPE

Visto e permitida a impressão.
Recife, 25 de fevereiro de 2015.

Profª. Edna Natividade da Silva Barros
Coordenador da Pós-Graduação em Ciência da Computação
Centro de Informática da Universidade Federal de Pernambuco

AGRADECIMENTOS

A minha família, que acreditou no meu projeto de vida e esteve ao meu lado em todos os momentos. A estes que, às vezes sem entender, apoiam, torcem, vibram, sorriem, choram, consolam.

Ao professor Alexandre Vasconcelos que me orientou, apoiou, discutiu, questionou, contribuiu, colaborou e me deu a oportunidade de desenvolver este trabalho.

Ao professor Bruno Tenório Ávila pela co-orientação, pelo desafio, pelo apoio e conhecimento compartilhado para construir esse trabalho, sem ele eu não teria conseguido terminar.

A minha esposa Amanda por todo o amor e carinho, te amo muito e a cada dia mais.

Aos meus filhos Giovanna e Eduardo, papai ama muito vocês.

Lista de Abreviaturas / Acrônimos

AIE - Arquivos de Interface Externa.

ALI - Arquivos Lógicos Internos.

APF – Análise de Pontos de Função.

AR - Arquivos Referenciados.

BFPUG – *Brazilian Function Point User Group*.

CA - Abordagem *Function Point Analysis and Cost Estimation in An Agile Development Environment*.

CE - Consulta Externa.

CIIn – Centro de Informática.

DMS - Diferença Mínima Significativa.

DR - Dados Referenciados.

EE - Entrada Externa.

ESBE – Engenharia de Software Baseada em Evidências.

EX - Abordagem *Extending Function Point Analysis*.

FM - Abordagem *Agile Estimation Using Functional Metrics*.

IFPUG - *International Function Point Users Group*.

MSL – Mapeamento Sistemático da Literatura.

NIG - Nível de Influência Geral.

PFA - Pontos de Função Ajustados.

PFNA - Pontos de Função Não Ajustados.

RL - Registros Lógicos.

SE - Saída Externa.

SQE - Soma de Quadrados dos Erros.

SQR - Soma de Quadrados da Regressão.

SQT – Soma de Quadrados Total.

TD – Tipos de Dados.

TI – Tecnologia da Informação.

UFPE – Universidade Federal de Pernambuco.

VAF – Valor do Fator de Ajuste.

VE - Valor Estimado.

XP – *Extreme Programming*.

Resumo

O desenvolvimento ágil tem se tornado cada vez mais comum no ambiente de desenvolvimento de software. Contudo estimativas de esforço em projetos de software que usam metodologias ágeis são feitas de maneira diferente das realizadas em projetos que usam metodologias tradicionais. Com base nesses problemas e com respeito ao grau de precisão de estimativa de esforço, essa dissertação apresenta uma análise comparativa entre as abordagens *Extending Function Point Analysis*, *Function Point Analysis and Cost Estimation in An Agile Development Environment* e *Agile Estimation Using Functional Metrics*. Através de um mapeamento sistemático, foram identificadas as formas de estimativas de esforço existentes na literatura e, a partir do mapeamento, foram escolhidas três abordagens que utilizam Pontos de Função como base para realizar estimativas. Por meio de uma pesquisa experimental e controlada com três cenários distintos, as abordagens foram avaliadas no intuito de testar sua aplicabilidade e analisar seus resultados. No mapeamento sistemático foram identificados 2572 estudos. Entre estes, 40 estudos primários que passaram por uma avaliação da qualidade foram incluídos para serem analisados em profundidade. Os resultados do experimento mostraram, que no contexto estudado, a abordagem *Function Point Analysis and Cost Estimation in An Agile Development Environment* não se adequou bem e teve os piores resultados em todos os cenários. Nos dois primeiros cenários, as abordagens *Extending Function Point Analysis* e *Agile Estimation Using Functional Metrics* se mostraram com resultados bem semelhantes e se mostraram adequadas. No último cenário a abordagem *Agile Estimation Using Functional Metrics* foi a que melhor adequou-se, sendo considerada assim a melhor abordagem para os cenários explorados pelo experimento.

Palavras-chave: Pontos de Função. Métricas de Software. Metodologias Ágeis.

Abstract

Agile development has become increasingly common in software development environment. However, effort estimation in software projects using agile methodologies are made differently to those undertaken in projects using traditional methods. Based on these problems and with respect to the degree of effort estimation accuracy, this work presents a comparative analysis between the approaches: Extending Function Point Analysis, Function Point Analysis and Cost Estimation in An Agile Development Environment, and Agile Estimation Using Functional Metrics. By performing a systematic mapping, it was identified forms of effort estimation existing in the literature and three approaches based of Function Points to perform estimation were chosen. By performing an experimental and controlled research using three different scenarios, the approaches were evaluated in order to test their applicability and analyze their results. In the systematic mapping it was identified 2572 studies. Among these, 40 primary studies that have undergone a quality assessment were included for a deep analysis. The experiment results showed that, in the context studied, the Function Point Analysis and Cost Estimation in An Agile Development Environment wasn't suitable and presented the worst results in all scenarios. The first two scenarios the approaches Extending Function Point Analysis and Agile Estimation Using Functional Metrics presented very similar results and were considered suitable. In the latter scenario, the approach Agile Estimation Using Functional Metrics presented the best results, being considered the best approach to the scenarios explored in the experiment.

Keywords: Function Points. Software Metrics. Agile Methodologies.

Índice de Figuras

Figura 1: Diagrama de processo da APF.	27
Figura 2: Etapas do Mapeamento Sistemático.....	41
Figura 3: Resultado do processo de busca e seleção nas fontes de pesquisa.	49
Figura 4: Métodos Ágeis investigados pelos estudos.....	63
Figura 5: Distribuição temporal dos estudos primários.....	63
Figura 6: Distribuição por fonte de dados dos estudos primários.....	64
Figura 7: Distribuição por métricas de esforço	65
Figura 8: Distribuição por métodos de pesquisa.	67
Figura 9: Distribuição por ambiente de estudo.	68
Figura 10: Distribuição por tipo de dados.....	69

Índice de Tabelas

Tabela 2-1: Complexidade funcional dos ALI e AIE.....	30
Tabela 2-2: Contribuição dos ALIs para a contagem.....	30
Tabela 2-3: Contribuição dos AIEs para a contagem.....	31
Tabela 2-4:Complexidade funcional para Entradas Externas (EE).	32
Tabela 2-5: Complexidade funcional para Saídas Externas (SE) e Consultas Externas (CE).....	32
Tabela 2-6:Contribuição dos EEs para a contagem.....	33
Tabela 2-7: Contribuição dos SEs para a contagem.....	33
Tabela 2-8: Contribuição dos CEs para a contagem.....	33
Tabela 2-9: Diferenças entre este trabalho e o de Cong e Cao (2013).	37
Tabela 2-10:Diferenças entre este trabalho e o de Banerjee, Narayanan e Mahadevan (2013).	38
Tabela 3-1: Score da avaliação da qualidade.	45
Tabela 3-2: Distribuição do score por nível de qualidade.....	46
Tabela 3-3: Ordem de execução dos analistas por projeto/abordagem.	56
Tabela 3-4: Tabela da análise de variância.....	60
Tabela 4-1: Métricas, frequências e Referências.....	65
Tabela 4-2: Dados do experimento com o estudo E1.	70
Tabela 4-3: Dados do experimento com o estudo E2.	70
Tabela 4-4: Dados do experimento com o estudo E3.	71
Tabela 4-5: Resíduos associados com cada observação do estudo E1.	71
Tabela 4-6: Resíduos associados com cada observação do estudo E2.	72
Tabela 4-7: Resíduos associados com cada observação do estudo E3.	72
Tabela 4-8: Dados da Análise de Variância do experimento com o estudo E1.....	73
Tabela 4-9: Dados da Análise de Variância do experimento com o estudo E2.....	73
Tabela 4-10: Dados da Análise de Variância do experimento com o estudo E3.....	73
Tabela 4-11: Valores das médias e diferenças entre elas do estudo E1.....	74
Tabela 4-12: Valores das médias e diferenças entre elas do estudo E2.....	75
Tabela 4-13: Valores das médias e diferenças entre elas do estudo E3.....	75

Sumário

1	Introdução	15
1.1	<i>Questões de Pesquisa</i>	19
1.2	<i>Objetivos</i>	19
1.3	<i>Visão Geral do Documento</i>	19
2	Referencial Teórico	21
2.1	<i>Métodos Ágeis</i>	21
2.2	<i>Estimativas de Software</i>	24
2.3	<i>Análise de Pontos de Função – APF</i>	25
2.4	<i>Trabalhos Relacionados</i>	36
2.5	<i>Considerações Finais do Capítulo</i>	38
3	Metodologia da Pesquisa	39
3.1	<i>Mapeamento Sistemático da Literatura</i>	40
3.1.1	<i>Procedimento de Busca e Seleção</i>	47
3.2	<i>Experimento Controlado</i>	50
3.2.1	<i>Planejamento do Experimento</i>	50
3.3	<i>Considerações Finais do Capítulo</i>	61
4	Resultados	62
4.1	<i>Síntese dos Resultados do Mapeamento Sistemático</i>	62
4.1.1	<i>Visão Geral do Estudo</i>	62
4.1.2	<i>Mapeamento das Evidências</i>	64
4.2	<i>Síntese dos Resultados do Experimento Controlado</i>	69
4.2.1	<i>Caracterização dos Estudos</i>	69
4.2.2	<i>Dados Coletados</i>	70
4.2.3	<i>Hipótese Nula</i>	73
4.2.4	<i>Teste de Tukey</i>	74
4.2.5	<i>Avaliação dos Resultados</i>	75
4.3	<i>Considerações Finais do Capítulo</i>	76
5	Conclusões e Considerações Finais	76
5.1	<i>Conclusões</i>	76
5.2	<i>Limitações e Ameaças à Validade</i>	77
5.2.1	<i>Mapeamento Sistemático da Literatura</i>	78
5.2.2	<i>Experimento Controlado</i>	78

5.3	<i>Dificuldades</i>	80
5.3.1	Mapeamento Sistemático da Literatura	80
5.3.2	Experimento Controlado	80
5.4	<i>Contribuições</i>	81
5.5	<i>Trabalhos Futuros</i>	81
5.6	<i>Considerações finais</i>	82
	Referências	83
	APÊNDICE A – Estudos incluídos no Mapeamento Sistemático	91
	APÊNDICE B – Formulário do Mapeamento Sistemático	95
	APÊNDICE C – Formulário Perfil Pesquisador	96
	APÊNDICE D – Formulários de Coletas Experimento	97
	APÊNDICE E – Roteiros de Execução	98
	APÊNDICE F – Limites Unilaterais da Distribuição F de Fisher-Snedecor ao Nível de 5% de Probabilidade	112
	APÊNDICE G – Tabela para o Teste de Tukey	113
	APÊNDICE H – Valores Reais e Valores Encontrados por Participante	115

1 Introdução

Os métodos ágeis são abordagens para desenvolvimento de software baseadas no desenvolvimento iterativo e incremental, no envolvimento direto do cliente no processo, na entrega rápida de recursos de maior valor de negócio e em respostas rápidas às mudanças (SOMMERVILLE, 2011). De acordo com Soares (2010), os métodos ágeis não rejeitam os processos e ferramentas, a documentação, a negociação de contratos ou o planejamento, mas simplesmente dizem que eles são de importância secundária.

Métodos ágeis têm sido apontados como uma alternativa às abordagens tradicionais para o desenvolvimento de software. Seu uso é necessário quando nos projetos há muitas mudanças, os requisitos são passíveis de alterações, o ambiente das organizações é dinâmico, as datas de entrega do software são curtas e o desenvolvimento rápido é fundamental. Por estes motivos, está havendo um crescimento no interesse e na aceitação dos processos ágeis (VERSIONONE, 2011) (MELO; SANTOS; CORBUCCI; KATAYAMA; GOLDMAN; KON, 2012). No Brasil, numa pesquisa feita por Melo *et al.* (2012) onde foram obtidas 466 respostas, 88,6% afirmaram utilizar uma método ágil.

Segundo Mens e Demeyer (2001), a demanda de empresas por produtos e serviços na área de tecnologia de informação vem crescendo consistentemente, contudo os orçamentos e cronogramas estão cada vez mais inflexíveis devido, por exemplo, à necessidade de redução de custos e cumprimentos de prazos cada vez mais curtos. Portanto, torna-se imprescindível o aumento da qualidade e da produtividade do desenvolvimento de software. Para este fim, usa-se métricas para: (i) servir como fonte de informação para o monitoramento da situação atual do processo de software; (ii) determinar a evolução do desenvolvimento; (iii) a identificação de desvios na execução do processo; (iv) identificar e priorizar as necessidades de melhorias no desenvolvimento do software; (v) fugir da intuição e subjetividade para algo mais preciso. A partir dessas informações é possível montar em um histórico dos projetos que podem ser utilizados para se estimar com maior precisão projetos futuros. Além disso, o gerente de projetos terá mais controle e pode tomar as ações que julgar necessárias, como por exemplo, alocar mais recursos. De fato, *visando suportar as iniciativas de melhoria no processo de software, a mensuração de software está se consolidando*

como uma prática importante entre as empresas (FRANÇA; VON STAA; LUCENA, 1998).

Um aspecto importante do projeto de desenvolvimento de software é o seu custo (DEMARCO, 1982). Em geral, o principal fator de custo de desenvolvimento de software é o da equipe de desenvolvimento (SHEPPERD; SCHOFIELD, 1997). Em geral, o custo da equipe é o derivado do somatório dos esforços individuais, onde esforço é definido como o tempo gasto para se realizar uma atividade. De acordo com Cao: (CAO, 2008),

A estimativa de esforço de desenvolvimento é fundamental para o planejamento, gestão e controle de um projeto de software. Subestimar fará pressão sobre o programa que resulta em maiores taxas de defeitos e custo de desenvolvimento (ABDEL-HAMID 1990). Por outro lado, superestimar irá causar desperdício de recursos e prejudicar a competitividade na licitação de um contrato.

Segundo Usman *et al.*, estimativa de esforço como outras atividades da engenharia de software é feita sob a “tutela” do processo de desenvolvimento. Há uma grande quantidade de pesquisa na área sob a forma de modelos, técnicas, métodos e ferramentas (USMAN; MENDES; WEIDT; BRITTO, 2014).

O tempo de desenvolvimento de uma funcionalidade de um software pode ser obtido a partir do tamanho dela. No entanto, há diversas formas de se medir o tamanho de um software (KAN, 2002). Uma dessas técnicas que merece destaque é a Análise de Pontos de Função (APF) (ALBRECHT, 1979) que foi padronizada pela norma ISO/IEC 20.926 (ISO 20926, 2002) como um método de medição funcional, aderente à norma ISO/IEC 14.143 (DEKKERS, 2003). Uma pesquisa do Quality Assurance Institute (PERRY, 1986), indicou que a APF é a métrica mais adequada para o estabelecimento de medições de qualidade e produtividade de projetos de software. Em 1993, APF tornou-se a métrica mais utilizada e estudada na Engenharia de Software (JONES, 1994).

As vantagens de se utilizar APF são: (i) método objetivo de encontrar a pontuação de um elemento; (ii) servir de base comparativa entre projetos diferentes e até mesmo equipes diferentes (SANTANA; GUSMÃO, 2009) (iii) alinhar a comunicação e a expectativa sobre o tamanho do sistema; (iv) O tamanho do sistema é medido pela visão do cliente (OEST, 2011). Por estas vantagens, percebe-se que o uso de APF está se consolidando (BFPUG, 2008). Dos 893 profissionais certificados em pontos de função

em todo o mundo, 371 são do Brasil, liderando esse número na frente da Coréia, Índia, Itália e EUA (AGUIAR, 2014; BFPUG, 2008).

No Brasil, diversas instituições públicas e privadas têm utilizado a APF devido aos diversos benefícios desta métrica, e por recomendações de órgãos reguladores para sua utilização nos contratos de serviços de desenvolvimento e manutenção de software em organizações governamentais, o que tornou a APF unânime no mercado de medição nacional (SISP, 2012). No âmbito da Administração Pública Brasileira, a Instrução Normativa número 4 (GOVERNO FEDERAL, 2008) no seu artigo 10 afirma que deve-se, de forma clara e objetiva, mensurar o custo financeiro estimado. Já no artigo 14 é afirmado que a fixação de procedimentos e de critérios de mensuração dos serviços prestados, abrangendo métricas, indicadores e valores deve ser realizado.

Portanto, empresas desenvolvedoras de software que utilizam métodos ágeis precisam colocar agora o planejamento e medições em primeiro plano. Todavia, num processo de desenvolvimento ágil, o software é desenvolvido de forma incremental em pequenas iterações e os feedbacks dos clientes são entradas importantes para as iterações subsequentes. Isso implica que as estimações e planos precisam ser feitos de forma progressiva (USMAN; MENDES; WEIDT; BRITTO, 2014). Além disso, dado o elevado número de iterações nos métodos ágeis, o planejamento e estimativa são realizados de forma diferente do tradicional desenvolvimento de software (COHN, 2005). Do ponto de vista da estimativa de software, nem todas as métricas e métodos de modelos de ciclos de vida convencionais podem ser utilizados em métodos ágeis sem adaptação (SCHMIETENDORF; KUNZ; DUMKE, 2008). Assim, técnicas particulares para métodos ágeis foram propostas e estão sendo utilizadas em projetos (FUQUA, 2003).

Nesta dissertação, um estudo foi conduzido baseado em um mapeamento sistemático da literatura para identificar quais técnicas de medição do esforço de software estão sendo utilizadas em projetos que adotam métodos ágeis, como resultado foram encontrados 40 estudos, destes 9 utilizam APF para derivar o esforço de desenvolvimento. Entretanto, nenhum desses artigos apresentou um estudo comparativo entre eles. Por isso, realizar uma análise comparativa entre as abordagens existentes torna-se importante para se observar a melhor forma de se medir esforço em projetos ágeis. Com o estudo comparativo, é possível: (i) validar as abordagens existentes por um pesquisador diferente dos que propuseram; (ii) compará-las entre si utilizando uma

metodologia científica unificada; (iii) verificar as abordagens em um ambiente potencialmente diferente do proposto. A partir do mapeamento, foram selecionadas abordagens que usam APF como base para a estimativa de esforço, em métodos ágeis e, para a avaliação das abordagens escolhidas, foi realizado um experimento controlado para avalia-las na prática.

1.1 Questões de Pesquisa

Este estudo objetiva responder as seguintes questões de pesquisa:

- Q1: Quais técnicas de medição do esforço de software reportadas na literatura estão sendo utilizadas em projetos que adotam métodos ágeis?
- Q2: Qual é a técnica de medição de esforço baseada em APF que melhor se adequa ao ambiente de uma empresa de desenvolvimento de software, que utiliza métodos ágeis, dentre as abordagens selecionadas na literatura?

1.2 Objetivos

O objetivo geral desse trabalho é realizar uma análise comparativa entre abordagens existentes que aplicam Pontos por Função, na medição de esforço, em projetos de desenvolvimento de software que utilizam métodos ágeis.

Os objetivos específicos que contribuirão para atingir o objetivo geral são:

- a) Identificar as técnicas de medição do esforço de software empregadas em projetos com métodos ágeis;
- b) Apontar a abordagem de medição de esforço baseada em APF mais adequada ao ambiente de uma empresa de desenvolvimento de software.

1.3 Visão Geral do Documento

Além deste capítulo introdutório, este trabalho está estruturado nos seguintes capítulos:

- Capítulo 2 – Referencial teórico: apresenta o referencial teórico utilizado neste trabalho;
- Capítulo 3 – Metodologia da pesquisa: apresenta em detalhes quais são os métodos de pesquisa utilizados para a condução deste trabalho;
- Capítulo 4 – Resultados: são apresentados os resultados obtidos no mapeamento sistemático da literatura e no experimento controlado;
- Capítulo 5 – Considerações finais: são apresentadas as limitações e ameaças à validade da pesquisa, assim como suas implicações, conclusões finais do

trabalho e recomendações para trabalhos futuros;

- Apêndices – Apresentam a lista dos estudos primários do mapeamento, modelos, resultados e formulários relevantes para a pesquisa.

2 Referencial Teórico

Neste capítulo é apresentada a fundamentação teórica para o melhor entendimento deste trabalho, de modo a contextualizar a pesquisa e permitir a correta interpretação dos resultados aqui apresentados. Os conceitos apresentados são resultados de um mapeamento da literatura conduzida de maneira não sistemática (ad hoc).

Nas próximas seções serão explorados os métodos ágeis, apresentando a motivação para o seu surgimento e suas principais características. Em seguida, serão discutidos os conceitos de estimativas/métricas de software e de APF, apresentando sua história e suas principais características.

2.1 Métodos Ágeis

A “indústria do *software*” sempre usou métodos cujos processos de desenvolvimento eram baseados em um conjunto de atividades predefinidas, descritas como processos prescritivos (AMBLER, 2004), onde muitas vezes, o trabalho começava com o levantamento completo de um conjunto de requisitos, seguido por um projeto de alto nível, de uma implementação, de uma validação e, por fim, de uma manutenção (SOMMERVILLE, 2011).

No fim da década de 90, começaram a surgir novos métodos sugerindo uma abordagem de desenvolvimento mais ágil onde os processos adotados tentam se adaptar às mudanças, apoiando a equipe de desenvolvimento no seu trabalho (AGILE MANIFESTO, 2001). Estes novos métodos surgiram como uma reação aos métodos tradicionais de desenvolvimento de sistemas, ganhando com o passar dos anos um número cada vez maior de adeptos.

Os métodos ágeis são abordagens para desenvolvimento de software baseadas no desenvolvimento iterativo e incremental, no envolvimento direto do cliente no processo, na entrega rápida de recursos de maior valor de negócio e em respostas rápidas às mudanças (SOMMERVILLE, 2011).

Através de um manifesto, publicado em fevereiro de 2001 por um grupo de dezessete especialistas em desenvolvimento de softwares foi criada uma definição oficial de Desenvolvimento Ágil de Software. Este grupo foi denominado Aliança Ágil

(AGILE MANIFESTO, 2001).

Após dias de debates, não se chegou ao consenso quanto a propor uma única metodologia, todavia foram identificados 4 valores e 12 princípios. Esses valores foram divulgados com a publicação do Manifesto Ágil (AGILE MANIFESTO, 2001). Desde então o termo Desenvolvimento Ágil passou a descrever abordagens de desenvolvimento que seguissem estes conceitos que implicam em valorizar:

- Indivíduos e interação entre eles são mais que processos e ferramentas;
- Software em funcionamento mais que documentação abrangente;
- Colaboração com o cliente mais que negociação de contratos;
- Responder a mudanças mais que seguir um plano.

O Manifesto Ágil não rejeita os processos e ferramentas, a documentação, a negociação de contratos ou o planejamento, mas simplesmente mostra que eles têm importância secundária quando comparado com os indivíduos e interações, com o software funcionando, com a colaboração com o cliente e com as respostas rápidas a mudanças e alterações (SOARES M. dos S., 2010).

Os doze princípios por trás do Manifesto Ágil (AGILE MANIFESTO, 2001) são:

1. Nossa maior prioridade é satisfazer o cliente através da entrega antecipada e contínua de software;
2. São sempre bem-vindas mudanças de requisitos, mesmo no fim do desenvolvimento. Processos ágeis se adequam a mudanças, para que o cliente possa tirar vantagens competitivas;
3. Entregar software funcionando com frequência, na escala de semanas até meses, com preferência em períodos mais curtos;
4. Pessoas relacionadas a negócios e desenvolvedores devem trabalhar em conjunto e diariamente, durante todo o curso do projeto;
5. Construir projetos em torno de indivíduos motivados. Dê-lhes o ambiente e apoio de que necessitam, e confie neles para fazer o trabalho;
6. O Método mais eficiente e eficaz de transmitir informações dentro de um time de desenvolvimento, é através de uma conversa cara a cara;
7. Software funcional é a principal medida de progresso;
8. Processos ágeis promovem um ambiente sustentável. Os patrocinadores, desenvolvedores e usuários, devem ser capazes de manter indefinidamente, passos constantes;
9. Contínua atenção à excelência técnica e bom design, aumenta a

agilidade;

10. Simplicidade: a arte de maximizar a quantidade de trabalho que não precisou ser feito;
11. As melhores arquiteturas, requisitos e designs emergem de times auto organizáveis;
12. A intervalos regulares, o time reflete sobre como se tornar mais eficaz, e então ajusta seu comportamento de acordo com as reflexões.

Métodos ágeis têm sido apontados como uma alternativa às abordagens tradicionais para o desenvolvimento de software. Os métodos tradicionais, conhecidos também como pesados ou orientados a planejamentos, devem ser aplicados apenas em situações em que os requisitos do sistema são estáveis e quando os requisitos futuros são previsíveis. Entretanto quando isso não é possível, como por exemplo, quando nos projetos há muitas mudanças, os requisitos são passíveis de alterações, o ambiente das organizações é dinâmico, as datas de entrega do software são curtas e o desenvolvimento rápido é fundamental, necessita-se então de métodos ágeis (BOEHM; TURNER, 2005 apud FUQUA, 2003).

Os métodos ágeis são abordagens modernas para criação de software com base na colaboração com o cliente, trabalho em equipe, desenvolvimento iterativo e incremental, e com respostas rápidas às mudanças. Tais métodos estão se tornando uma alternativa ao desenvolvimento de software tradicional, com propostas que dão ênfase a flexibilidade, comunicação informal e código funcionando, objetivando eliminar a dependência em realizar um extenso planejamento inicial e documentação dos requisitos do sistema (HIGHSMITH, 2004).

Os métodos ágeis enfatizam talentos e habilidades inerentes aos indivíduos, moldando o processo às pessoas e equipes específicas (COCKBURN; HIGHSMITH, 2001). Portanto, para um método se enquadrar na categoria dos métodos ágeis deve utilizar o desenvolvimento iterativo e incremental, valorizar a colaboração e comunicação entre cliente e toda a equipe, e ser adaptativo, com capacidade de responder às mudanças (ABRAHAMSSON; SALO; RONKAINEN; WARSTA, 2002).

Em projetos ágeis, os requisitos a serem desenvolvidos são expressos na forma de histórias feitas por usuários (LEFFINGWELL; BEHRENS 2009). Um dos métodos mais populares de dimensionamento destas histórias é uso de *Story Points*, unidade subjetiva de estimativa obtida por equipes ágeis. Neste método, uma equipe compara a

estória do usuário a uma ou mais estórias semelhantes e atribui à nova estória um tamanho em termos de "*Story Points*" ou "dia ideal de trabalho". O número de *Story Points* associados com uma estória representa o tamanho global da mesma. Não há uma fórmula criada para definir o tamanho de uma estória (SANTANA; LEONEO; GUSMÃO; VASCONCELOS, 2011). Cada equipe pode definir *Story Points* como quiser. O uso de *Story Points* (o número de estórias de usuários entregues em uma iteração) pode ser considerado como boa prática para a medição da velocidade de desenvolvimento de uma equipe usando um desenvolvimento ágil (COHN M. 2005)

Com base nos princípios e valores do Manifesto Ágil, diversos métodos têm sido propostos, tais como: *Adaptative Software Development*, *Crystal*, *Dynamic Systems Development Method*, *eXtreme Programming (XP)*, *Feature Driven Development*, *Lean Development e Scrum* (BOEHM, 2006).

2.2 Estimativas de Software

Com o avanço da Tecnologia da Informação (TI), as pessoas e as organizações necessitam cada vez mais de sistemas capazes de fornecer informações com qualidade (FERREIRA; NUNES, 2008). Para criar esses sistemas, as organizações de desenvolvimento de software precisam cada vez mais se apoiar a tríade prazo, escopo e qualidade. Para manter o equilíbrio deste tripé e concluir o projeto com sucesso, é necessário lançar mão de técnicas de planejamento e gestão dos projetos, processos e produtos de software. Nesse contexto, são criados mecanismos baseados em informações quantitativas e qualitativas para o acompanhamento do projeto.

Segundo Monteiro (2005) "A medição é fundamental para qualquer atividade de engenharia; a engenharia de software não é a exceção.". Muitas vezes, na engenharia de software, o planejamento de um projeto pode ter uma variação nas estimativas devido à falta de dados históricos, e à disponibilização de um reduzido número de técnicas maduras (PRESSMAN, 2002).

As estimativas de custo e esforço em projetos de software são, comumente, baseadas na predição do tamanho do sistema que será desenvolvido (ABRAHAO; INFRAN, 2008). Estimar o tamanho do software é uma das atividades mais críticas dentro do ciclo de vida do desenvolvimento, uma vez que é a partir desta estimativa tem-se uma ideia do tamanho e do tempo que precisará ser gasto do sistema que será

desenvolvido e assim, com esta projeção, o projeto será contratualmente regido.

A medição de software contribui para um melhor entendimento dos processos que fazem parte de um projeto, ajuda a estabelecer pontos importantes no projeto, contribui na análise do status do projeto de acordo com o que foi planejado e permite avaliar até que ponto o processo que está sendo executado encontra-se de acordo com os padrões especificados (FLORAC; CARLETON, 1999).

A literatura aponta diversas técnicas para estimativa de tamanho de projetos de software. Como por exemplo, Análise de Pontos de Função (APF), Análise de Pontos de Caso de Uso (PCU), COSMIC *Full Function Point* (Albrecht 1979 apud ASHMAN 2004 apud COSMIC, 2007).

2.3 Análise de Pontos de Função – APF

No início da década de 70, pesquisadores da IBM perceberam que um sistema seria mais bem **avaliado através das funções executadas pelos programas**, em oposição ao volume ou complexidade do código gerado (técnica bastante usada na época).

Em 1979, Albrecht, trabalhando para IBM, definiu a Análise de Pontos de Função (*Function Point Analysis*). Em 1984, ele refinou as definições sobre a APF no IBM CIS Guideline 313, *AD/M Productivity Measurement and Estimate Validation*. E, em 1986, um grupo de usuários da APF formou o *International Function Point User Group* (IFPUG). A partir desta data, o IFPUG se tornou o responsável pela definição das regras de contagem, pelo treinamento de usuários, pela certificação dos profissionais interessados na utilização desta técnica em ambiente de trabalho e pelo fornecimento de dados históricos de produtividade da indústria de desenvolvimento de software (ANDRADE, 2004; IFPUG, 2010). A APF tornou-se um padrão internacional através da norma ISO/IEC 20.9261 (ISO/IEC 20926, 2002) e é considerada uma das primeiras técnicas a medir o tamanho do software com alguma precisão.

No Brasil, a Instrução Normativa número 4, de 19 de maio de 2008, expedida pela Secretaria de Logística e Tecnologia da Informação, em vigor desde 02 de janeiro de 2009 (GOVERNO FEDERAL 2008), que dispõe sobre o processo de contratação de serviços de tecnologia da informação pela Administração Pública Direta, autárquica e

fundacional, praticamente extingue a aferição de custo e esforço por meio da métrica homem/hora de trabalho. Ainda neste sentido, o Tribunal de Contas da União tem se pronunciado constantemente apontando a técnica de Ponto de Função como sendo a unidade mais adequada para o entendimento do tamanho de qualquer sistema de informação (de ALBUQUERQUE 2012).

O *Brazilian Function Point User Group* (BFPUG) define ponto de função como uma medida funcional de tamanho de software, introduzida em 1979 por Alan Albrect da IBM (de ALBUQUERQUE 2012).

Essa técnica mede o tamanho funcional do software quantificando as funcionalidades do ponto de vista do usuário, com base apenas no projeto lógico e especificação funcional, sendo seus principais objetivos (ALBRETCH 1979):

- Medir a quantidade de funcionalidade que o usuário requer e recebe;
- Medir o tamanho do software e da manutenção de forma independente de tecnologia usada para o desenvolvimento;
- Prover uma unidade de medição normalizada entre projetos e organizações;
- Identificar um padrão de medida para a produtividade e qualidade;
- Fornecer ferramentas para auxílio nas estimativas de desenvolvimento de software;
- Possibilitar o uso por diferentes projetos, em diferentes empresas e ambientes variados, demonstrando consistência na comparação;
- Ser compreensível pelo pessoal não técnico;
- Ser utilizável em todo o ciclo de desenvolvimento do software;
- Ser um processo simples de mensuração;
- Ser consistente e conciso.

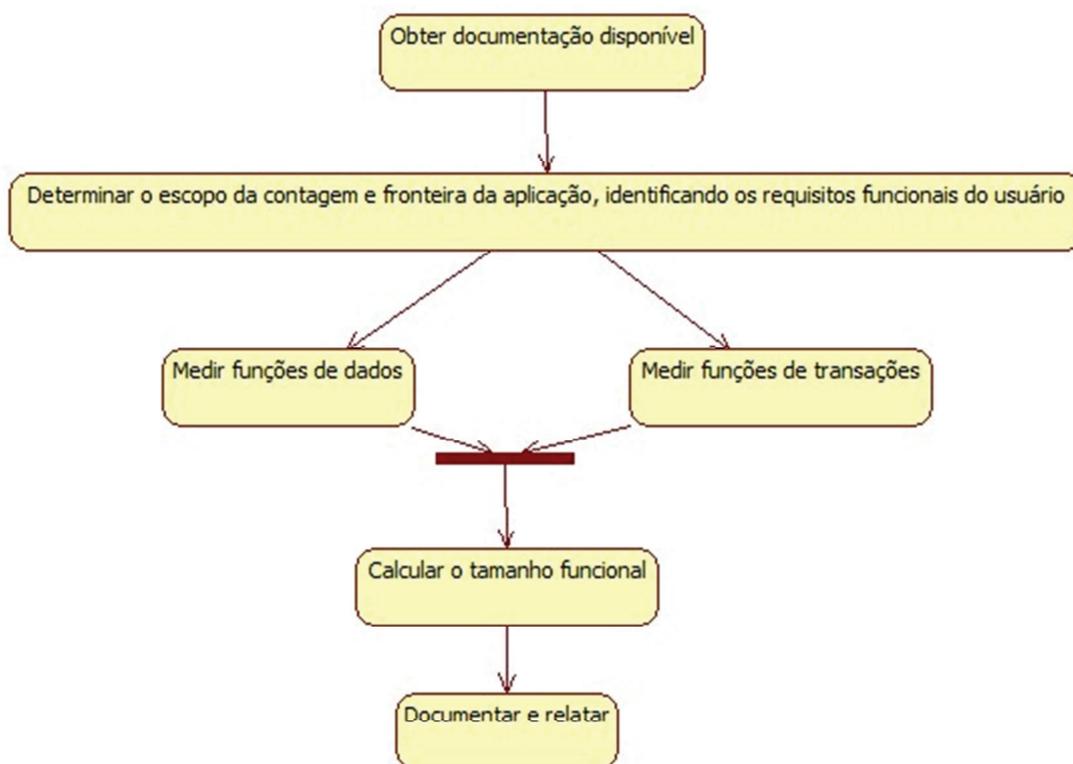
De acordo com a técnica APF, uma aplicação de software, vista sob a ótica do usuário, é um conjunto de funções ou atividades do negócio que o beneficiam na realização de suas tarefas (CASTRO 1998).

APF permite não só medir o tamanho do sistema em termos da funcionalidade fornecida ao usuário, mas também estimar seu tamanho em qualquer fase do seu ciclo de vida, mesmo que os requisitos ainda não tenham sido detalhados (VAZQUEZ; SIMÕES; ALBERT, 2010).

É importante destacar que pontos de função não medem diretamente esforço, produtividade ou custo. É exclusivamente uma medida de tamanho funcional do software. Este tamanho em conjunto com outras variáveis é que pode ser usado para derivar produtividade e estimar esforço e custo (VAZQUEZ; SIMÕES; ALBERT, 2010).

Com o crescimento do uso de pontos de função, tem havido uma maior aplicação e utilização da APF. Desde a sua criação em 1986 a *International Function Point Users Group-IFPUG* vem continuamente, aprimorando esse o método de dimensionamento de software. A versão da IFPUG sobre o método está publicada no *The Counting Practices Manual* e está na versão 4.3.1 (IFPUG, 2010). O diagrama de processo de medição é mostrado na Figura 1.

Figura 1: Diagrama de processo da APF.



Fonte: O Autor

Antes de iniciar a contagem é preciso definir o tipo de contagem, o escopo e a fronteira da aplicação. A determinação do tipo de contagem de pontos por função

representa a função do objeto da contagem que será usado para estimar o projeto, podendo esse projeto ser de desenvolvimento, de manutenção ou uma aplicação já finalizada (IFPUG, 2010).

Os três tipos de contagem são:

- Projeto de desenvolvimento: Tem como objetivo desenvolver e fornecer a primeira versão de uma aplicação (criação de uma aplicação). A contagem de pontos de função de um projeto de desenvolvimento mede as funções fornecidas ao usuário na primeira instalação da aplicação (quando o projeto é concluído);
- Projeto de melhoria: Tem como objetivo efetuar e entregar uma manutenção adaptativa de uma aplicação já existente. A contagem de um projeto de melhoria mede as modificações nas funções da aplicação existente, através da inclusão, alteração ou exclusão de funções fornecidas ao usuário na conclusão do projeto. Quando um projeto de melhoria é concluído e instalado, a contagem de pontos de função da aplicação deve ser atualizada, utilizando uma fórmula específica para refletir as alterações das respectivas funcionalidades;
- Aplicação: Está associada à aplicação instalada, conhecida também como uma contagem de pontos de função da *baseline*. Esta contagem fornece uma medida das funções que a aplicação oferece atualmente ao usuário. Ele é inicializado ao final da contagem do projeto de desenvolvimento, sendo atualizado no término de todo projeto de melhoria que altera as funcionalidades da aplicação.

Após a escolha do tipo da contagem, deve-se procurar a documentação disponível sobre o sistema e/ou projeto que será medido. A documentação ideal deve ser capaz de (VAZQUEZ; SIMÕES; ALBERT, 2010):

- Descrever a funcionalidade entregue pelo software; ou
- Descrever a funcionalidade que é impactada pelo projeto de software medido.

Uma documentação adequada pode incluir requisitos, modelos de dados/objetos, diagramas de classe, diagramas de fluxo de dados, casos de uso, descrições procedurais, *layouts* de relatórios, telas, manuais de usuário e outros artefatos do desenvolvimento de software (IFPUG, 2010).

A documentação de um projeto/aplicação disponibilizada para sua medição deve refletir a visão de seu usuário. Um usuário é qualquer pessoa ou coisa que se comunica

ou interage com o software a qualquer momento (IFPUG, 2010).

O passo seguinte é determinar o escopo da contagem e fronteira da aplicação, identificando os requisitos funcionais do usuário. O escopo define as funções que serão contadas de acordo com a visão do usuário, e a fronteira da aplicação é considerada um limite conceitual que separa o software medido da visão dos usuários. Neste contexto, “usuário” é considerado qualquer pessoa ou coisa que defina os requisitos funcionais do software e que se comunique ou interaja com o software, inclusive outro sistema (IFPUG, 2010 apud SILVA, 2000 apud ANDRADE, 2004).

O IFPUG define que o escopo da contagem (IFPUG, 2010):

- Define o (sub)conjunto do software que está sendo medido;
- É determinado pelo propósito para a realização da contagem de pontos de função;
- Identifica quais funções serão incluídas na medida de tamanho funcional assim como fornece respostas relevantes para o propósito da contagem;
- Pode incluir mais de uma aplicação.

O IFPUG especifica as seguintes regras para determinação da fronteira da aplicação (VAZQUEZ; SIMÕES; ALBERT, 2010):

- A fronteira entre aplicações deve ser baseada na separação das funções conforme estabelecido pelos processos de negócio, não em considerações tecnológicas;
- Sua determinação deve ser feita com base no ponto de vista do usuário. O foco deve estar no que ele pode entender e descrever;
- Em projetos de melhoria, a fronteira estabelecida no início do projeto deve estar de acordo com a fronteira já estabelecida para a aplicação sendo modificada.

Com o escopo e a fronteira definidos, o próximo passo é medir as funções de dados. As funções de dados representam a funcionalidade oferecida ao usuário para satisfazer requisitos de dados internos e externos à aplicação, ou seja, representam os requisitos de armazenamento de dados do usuário. Há dois tipos (classificação) de funções de dados:

- Arquivos Lógicos Internos (ALI): Grupos de dados logicamente relacionados ou

informações de controle, que são alterados na própria aplicação;

- Arquivos de Interface Externa (AIE): Grupos de dados logicamente relacionados, cujo processo de alteração é feito em outra aplicação;

Vale salientar que a intenção primária de um ALI é armazenar dados mantidos através de um ou mais processos elementares da aplicação que está sendo contada. Já a intenção primária de um AIE é armazenar dados referenciados através de um ou mais processos elementares dentro da fronteira da aplicação que está sendo contada. Isto significa que um AIE contado para uma aplicação deve ser um ALI em outra aplicação (IFPUG, 2010).

A complexidade de cada ALI ou AIE é baseada em dois conceitos:

- A quantidade de registros lógicos (RL): Um subgrupo de dados, **reconhecido pelo usuário, e logicamente relacionados dentro de ALI/AIE;**
- A quantidade de itens de dados referenciados (DR): **Campo único, reconhecido pelo usuário, não repetido, e que atravessa a fronteira da transação.**

A complexidade dos arquivos lógicos é determinada a partir das quantidades de seus tipos de dados e tipos de registros identificados, de acordo com a Tabela 2-1:

Tabela 2-1: Complexidade funcional dos ALI e AIE

Quantidade	Entre 1 e 19 DRs	Entre 20 e 50 DRs	Acima de 50 DRs
1 RL	Baixa	Baixa	Média
Entre 2 e 5 RLs	Baixa	Média	Alta
Acima de 5	Média	Alta	Alta

Fonte (IFPUG, 2010).

A contribuição dos arquivos lógicos, ou seja, o tamanho em pontos de função, deve ser calculado após a determinação de sua complexidade. Os valores em pontos de função para os ALIs e AIEs são mostrados nas Tabelas 2-2 e 2-3.

Tabela 2-2: Contribuição dos ALIs para a contagem.

Complexidade	Valor em Pontos de Função
Baixa	7
Média	10

Alta	15
------	----

Fonte (IFPUG, 2010).

Tabela 2-3: Contribuição dos AIEs para a contagem.

Complexidade	Valor em Pontos de Função
Baixa	5
Média	7
Alta	10

Fonte (IFPUG, 2010).

Paralelamente à medição das funções de dados deve ser feita a medição das funções de transações. Elas representam as funções de processamento dos dados fornecidos pela aplicação ao usuário. As funções de transação podem ser Entradas Externas (EE), Saídas Externas (SE); e Consultas Externas (CE) (IFPUG, 2010).

Antes de se classificar uma função transacional deve-se verificar se a mesma é um processo elementar e se não existe nenhuma outra transação com os mesmos tipos de dados, arquivos referenciados, e lógicas de processamento, que já tenha sido contada no projeto (VAZQUEZ; SIMÕES; ALBERT, 2010).

Um processo elementar é a menor unidade de atividade que satisfaz as seguintes regras para ser reconhecido como uma função de transação (VAZQUEZ; SIMÕES; ALBERT, 2010):

- Constituir uma transação completa;
- Ter significado para o negócio, ou seja, ser reconhecido pelo usuário e satisfazer um requisito funcional;
- Ser autocontido;
- Permitir o negócio da aplicação ser medido em um estado consistente

O processo elementar deve ser classificado de acordo com seu objetivo principal, em uma das funções de transações abaixo (VAZQUEZ; SIMÕES; ALBERT, 2010):

- Entrada Externa (EE): Processos elementares (transações) que processam dados ou informações de controle que entram pela fronteira da aplicação. Ex: incluir cliente; alterar conta; excluir pessoa;

- Saída Externa (SE): Processos elementares que enviam dados ou informações de controle para fora da fronteira da aplicação. Seu objetivo é apresentar informações ao usuário que foram recuperadas através de um processamento lógico e não apenas uma simples recuperação de dados. Uma SE deve envolver cálculos ou criação de dados derivados ou também, pode manter um ALI ou alterar o comportamento do sistema. Exemplo: relatório do total de vendas por período, tela de Login (com criptografia).
- Consulta Externa (CE): Semelhante a uma SE, é um processo elementar que envia dados (ou informações de controle) para fora da fronteira da aplicação. Todavia, seu objetivo é apresentar informação para o usuário, por meio apenas de uma recuperação das informações, sem a realização de nenhum cálculo nem a criação de dados derivados. Exemplo: consulta de clientes, lista de vendas.

A complexidade de cada das transações é baseada em dois dados:

- Quantidade de Arquivos Referenciados (AR): Onde arquivo referenciado é um ALI lido ou mantido, ou um AIE lido, pela função do tipo transação.
- Quantidade de Tipos de Dados (TD): Onde tipo de dado é um campo único, reconhecido pelo usuário, e que atravessa a fronteira da transação, sendo exibido para o usuário.

A complexidade das funções de transação é determinada a partir das quantidades de seus tipos de dados e arquivos referenciados, de acordo com a Tabela 2-4 para os EEs e de acordo com a Tabela 2-5 para os CEs e SEs:

Tabela 2-4: Complexidade funcional para Entradas Externas (EE).

	Até 5 TDs	Entre 5 e 15 TDs	Acima de 15 TDs
Menor que 2 ARs	Baixa	Baixa	Média
Entre 2 e 3 ARs	Baixa	Média	Alta
Acime de 3 ARs	Média	Alta	Alta

Fonte (IFPUG, 2010).

Tabela 2-5: Complexidade funcional para Sidas Externas (SE) e Consultas Externas (CE).

	Até 5 TDs	Entre 5 e 19 TDs	Acima de 19 TDs
Menor que 2 ARs	Baixa	Baixa	Média
Entre 2 e 3 ARs	Baixa	Média	Alta

Acime de 3 ARs	Média	Alta	Alta
-----------------------	-------	------	------

Fonte (IFPUG, 2010).

A contribuição das funções de transação, ou seja, o tamanho em pontos de função, deve ser calculada após a determinação de sua complexidade. Os valores em pontos de função para os EEs, SEs e CEs são mostrados respectivamente nas Tabelas 2-6, 2-7 e 2-8.

Tabela 2-6: Contribuição dos EEs para a contagem

Complexidade	Valor em Pontos de Função
Baixa	3
Média	4
Alta	6

Fonte (IFPUG, 2010).

Tabela 2-7: Contribuição dos SEs para a contagem.

Complexidade	Valor em Pontos de Função
Baixa	4
Média	5
Alta	7

Fonte (IFPUG, 2010).

Tabela 2-8: Contribuição dos CEs para a contagem.

Complexidade	Valor em Pontos de Função
Baixa	3
Média	4
Alta	6

Fonte (IFPUG, 2010)

Após o a medição das funções de dados e transações soma-se os dois valores e assim é obtido o tamanho funcional, também chamado de Pontos de Função Não Ajustados (PFNA), que representa o tamanho do software obtido pela quantificação dos requisitos funcionais do usuário. Os PFNA de uma aplicação medem os requisitos

solicitados e recebidos pelo usuário, sendo calculado a partir da soma das contribuições de todas as funções identificadas (IFPUG 2010, VAZQUEZ; SIMÕES; ALBERT, 2010).

A partir da aplicação do fator de ajuste sobre os pontos de função não ajustados os pontos de função ajustados (PFA) de um software são obtidos. O Valor do Fator de Ajuste (VAF) é baseado em quatorze características gerais do sistema, que classificam a funcionalidade geral da aplicação sendo contada. Cada característica possui descrições associadas que ajudam a determinar o nível de influência da característica. O nível de influência de cada característica possui uma escala que varia de zero (sem influência) até cinco (forte influência). O Nível de Influência Geral (NIG) é obtido pelo somatório do nível de influência de cada característica e o Valor do Fator de Ajuste é obtido pela expressão:

$$VAF = 0,65 + (NIG * 0,01) \quad (1)$$

O total de pontos por função da aplicação será encontrado mediante a multiplicação do número de pontos por função não ajustados pelo VAF:

$$PFA = PFNA * VAF \quad (2)$$

Enquanto as funções do tipo dado refletem requisitos específicos de armazenamento e as funções do tipo transação refletem requisitos específicos de processamento, as características gerais refletem funções que afetam a aplicação de maneira geral. As quatorze características gerais do sistema são (IFPUG, 2010):

- Comunicação de dados;
- Processamento distribuído;
- Atualização de dados online;
- Entrada de dados online;
- Volume de transações;
- Eficiência do usuário final;
- Complexidade do processamento;

- Facilidade de implantação;
- Multiplicidade de locais;
- Facilidade de mudanças;
- Facilidade operacional;
- Desempenho;
- Utilização do equipamento;
- Reutilização de código.

Vale salientar que ao final do ano de 2002, o uso do fator de ajuste tornou-se opcional, porque várias destas características gerais contemplam requisitos não funcionais. Entretanto, mesmo antes do uso do fator de ajuste tornar-se opcional, uma pesquisa apoiada pelo IFPUG demonstrou que vários usuários já não o utilizavam (VAZQUEZ; SIMÕES; ALBERT, 2010).

Devido à constatação de que algumas dessas características gerais já estão bastante desatualizadas e da grande margem na interpretação das mesmas foi criado um grupo de trabalho do comitê de práticas de contagem do IFPUG (VAZQUEZ; SIMÕES; ALBERT, 2010), com a responsabilidade de se dedicar ao tema de medição dos requisitos não funcionais, tendo como objetivo a criação de um outro método para medir esses requisitos independentemente da técnica de APF.

Com relação aos benefícios da aplicação da APF nas organizações é possível destacar (VAZQUEZ; SIMÕES; ALBERT, 2010):

- É uma ferramenta para determinar o tamanho de um pacote adquirido pela contagem de todas as funções incluídas;
- É um meio para estimar custo e recursos para o desenvolvimento e manutenção de software;
- Provê auxílio aos usuários na determinação dos benefícios de um pacote para sua organização, através da contagem das funções que especificamente correspondem aos seus requisitos;
- Complementa o gerenciamento dos requisitos ao auxiliar na verificação da solidez e completude dos requisitos especificados;

- Apoia o gerenciamento de escopo de projetos;
- Suporta a análise de produtividade e qualidade, seja diretamente ou em conjunto com outras métricas como esforço, defeitos e custo;
- É um fator de normalização para comparação de software ou para comparação da produtividade na utilização de diferentes técnicas;
- É uma ferramenta para fundamentar a negociação de contratos.

A técnica ainda apresenta dificuldades por possuir relativa subjetividade, considerando que cada usuário pode ter uma visão diferente do negócio. A complexidade do projeto está relacionada diretamente com volume de arquivos lógicos, registros lógicos e itens de dados identificados.

Dentre as principais razões para a utilização da APF como métrica estão as seguintes (AGUIAR, 2003 apud MACORATTI, 2005):

- O fato dos Pontos de Função serem mantidos por uma organização internacional sem fins lucrativos, o IFPUG, desde 1986;
- Os PF possuem suporte no Brasil através do *chapter* – BFPUG;
- Os PF serem padronizados pela ISO através da norma ISO/IEC 20296;
- Existir um grande acervo de informações sobre PF armazenadas em diversas organizações o que permite estudos e comparações;
- Os PF modelarem os requisitos em um nível de abstração mais alto e independente dos artefatos e poderem ser usados por organizações que usam qualquer forma de representação de requisitos;
- Os PF serem usados em contratos e licitações no Brasil em organizações governamentais e pelo mercado em geral.

2.4 Trabalhos Relacionados

Nesta seção são apresentados os trabalhos relacionados com esta pesquisa. Uma tabela comparativa mostrando as diferenças e similaridades entre este estudo e os trabalhos relacionados é apresentada no fim desta seção.

A Review of Effort Estimation Studies in Agile, Iterative and Incremental Software Development (CONG; CAO, 2013)

Cong e Cao (CONG; CAO, 2013) afirmam que existem poucos estudos empíricos sobre as estimativas de esforço em métodos e desenvolvimento iterativo incremental, e que pouco se sabe sobre como essas estimativas estão sendo conduzidas na prática e quais são os seus efeitos e benefícios.

Objetivando minimizar esta problemática, conduziram um mapeamento sistemático da literatura para avaliar, sintetizar e apresentar as evidências sobre o uso, as lacunas existentes e por fim, realizar recomendações para trabalhos futuros.

Com base nesse problema identificado, o mapeamento sistemático da literatura realizado por este trabalho correspondeu a uma **extensão** do estudo de Cong e Cao (CONG; CAO, 2013), retirando a parte dos estudos sobre processo de desenvolvimento não ágeis, procedendo à busca pelos estudos nos anos de 1999 a 2014, seguindo basicamente todas as suas etapas metodológicas, com apenas algumas diferenças que estão descritas na Tabela 2-9.

Tabela 2-9: Diferenças entre este trabalho e o de Cong e Cao (2013).

Categoria	Este Trabalho	Cong e Cao
Método	Mapeamento sistemático da literatura e experimento controlado	Mapeamento sistemático da literatura
Protocolo	1. Pesquisa pelos estudos publicados em inglês e português até 2013; 2. Inclusão de novos sinônimos na <i>string</i> de busca, e exclusão de algumas palavras repetidas.	Pesquisa pelos estudos publicados em inglês até 2012;
Fontes de busca	Cinco engenhos de busca automática (com o uso de uma ferramenta automática) e sete fontes de busca manual.	Nove engenhos de busca automática.
Nº de trabalhos revisados	40 trabalhos	32 trabalhos
Garantia da Qualidade	Foram incluídos critérios de qualidade definidos no trabalho.	Não houve inclusão de nenhum critério, bastando apenas ter sido publicado em algum periódico ou anais.

Fonte: O Autor

Estimating Agile Iterations by Extending Function Point Analysis (BANERJEE; NARAYANAN; MAHADEVAN, 2012)

Banerjee, Narayanan e Mahadevan (BANERJEE; NARAYANAN;

MAHADEVAN, 2012) apresentam uma proposta de estimativa de tamanho de interações em projetos ágeis estendendo a técnica de Pontos por Função com as *activity scope* de um projeto de software definidas por Caper Jones (JONES, 2008). A técnica foi aplicada em três projetos ágeis e foi observado que há uma correlação linear entre o esforço consumido e o tamanho estimado da iteração.

Este estudo foi escolhido como uma das três abordagens escolhidas para a realização do experimento controlado. Na Tabela 2-10 estão descritas as diferenças entre os estudos.

Tabela 2-10: Diferenças entre este trabalho e o de Banerjee, Narayanan e Mahadevan (2013).

Categoria	Este Trabalho	Banerjee et al
Método	Mapeamento sistemático da literatura e experimento controlado	Estudo de caso
Escopo do Estudo	Indústria e academia	Indústria
Objetivo	Analisar entre as abordagens existentes na literatura qual é a que melhor se adequa para o ambiente de uma empresa de desenvolvimento de software.	Propor uma nova abordagem para o uso de Pontos de Função e projetos ágeis.

Fonte: O Autor

2.5 Considerações Finais do Capítulo

Este capítulo apresentou os conceitos e fundamentos teóricos usados como base para esta pesquisa. Este referencial envolveu os conceitos sobre metodologias ágeis, estimativa de software, Análise de Pontos por Função e os trabalhos relacionados com esta pesquisa. Este referencial foi fruto da busca de conceitos nos principais estudos publicados em artigos, revistas científicas, livros e dissertações da área dessa pesquisa. No capítulo seguinte será exposta a metodologia de pesquisa deste trabalho.

3 Metodologia da Pesquisa

Neste capítulo são descritos em detalhes quais são os métodos de pesquisa utilizados para a condução deste trabalho, de modo a tornar os resultados mais confiáveis e possíveis de serem reproduzidos por outros pesquisadores.

Um método de pesquisa é o conjunto das atividades sistemáticas e racionais que, com maior segurança e economia, permite alcançar um objetivo, traçando um caminho a ser seguido, detectando erros e auxiliando as decisões do cientista (MARCONI; LAKATOS, 2008). Entretanto, Easterbrook et al. (EASTERBROOK; SHULL; SINGER; SJØBERG, 2008) recomendam que o posicionamento filosófico da pesquisa deve ser escolhido antes da definição do método.

O posicionamento filosófico escolhido para este trabalho foi o pragmático, pois o mesmo considera o conhecimento prático, e o uso de métodos de pesquisa mista. O posicionamento pragmático acredita que todo conhecimento é aproximado e incompleto, e seu valor depende dos métodos pelos quais foi obtido, sendo, portanto, a verdade relativa ao observador [MENAND, 1997 apud EASTERBROOK; SHULL; SINGER; SJØBERG, 2008].

A abordagem escolhida foi a indutiva, que se caracteriza por partir de um conjunto de dados particulares, suficientemente constatados (amostra), inferindo-se uma verdade geral ou universal, não contida nas partes examinadas (população) (MARCONI; LAKATOS, 2008). Portanto, o objetivo dos argumentos é levar a conclusões prováveis, cujo conteúdo é muito mais amplo do que o das premissas nas quais se basearam.

Com relação à natureza metodológica, a adotada no trabalho foi a natureza qualitativa. A pesquisa qualitativa trabalha com o universo de significados, motivos, aspirações, crenças, valores e atitudes, o que corresponde a um espaço mais profundo das relações, dos processos e dos fenômenos que não podem ser reduzidos à operacionalização de variáveis (MINAYO, 2001).

A abordagem qualitativa é apropriada para obter entendimento mais profundo e detalhado sobre as investigações, ambientes e comportamentos (MARCONI; LAKATOS, 2008) e entre as diversas características atribuídas à pesquisa qualitativa podemos destacar objetivação dos fenômenos, hierarquização das ações de descrever,

compreender, explicar, precisão das relações entre o global e o local em algum fenômeno, observação das diferenças entre o mundo social e o mundo natural, o respeito ao caráter interativo entre os objetivos buscados pelos pesquisadores, suas orientações teóricas e seus dados empíricos, a busca de resultados o mais realistas possível, e a oposição ao que defende um modelo único de pesquisa para todas as ciências (LAKATOS; MARCONI, 2003).

O escopo da pesquisa envolveu dois métodos de pesquisa: mapeamento sistemático da literatura (KITCHENHAM, 2007), como forma de analisar e interpretar um conjunto de dados obtidos na literatura existente sobre uma questão de investigação particular, área temática ou fenômeno de interesse, baseando-se em evidências e a partir dos estudos escolhidos na revisão; e um experimento controlado (JURISTO; MORENO, 2010), como forma de analisar as abordagens existentes e a partir desta análise avaliar qual a melhor forma de utilizar pontos de função em projetos que utilizam métodos ágeis para o desenvolvimento de software.

3.1 Mapeamento Sistemático da Literatura

Um mapeamento sistemático da literatura (MSL) tem o objetivo de identificar, avaliar e interpretar o que existe de relevante para uma particular questão de pesquisa, área ou fenômeno de interesse (KITCHENHAM, 2007). É uma abordagem disciplinada para a condução de um Mapeamento da literatura, possuindo, assim, maior credibilidade, pois esta deve ser conduzida seguindo rigorosamente os métodos definidos (PETTICREW; ROBERTS, 2006). Com isso teremos a redução do viés do MSL, uma vez que ele pode permitir sua repetição e ser auditável.

Assim, as maiores razões para utilização de um MSL são: resumir a evidência existente sobre um tratamento ou tecnologia, identificar eventuais lacunas na pesquisa atual, a fim de sugerir áreas para posterior investigação e fornecer um quadro, a fim de posicionar adequadamente novas atividades de investigação.

Conduzir um MSL é uma forma do autor demonstrar o conhecimento sobre um campo de estudo particular, incluindo o vocabulário, teorias, variáveis e fenômenos chave, e seus métodos e história. Conduzir um MSL também informa ao pesquisador, grupos de pesquisas e pesquisadores influentes na área de estudo (RANDOLPH, 2009).

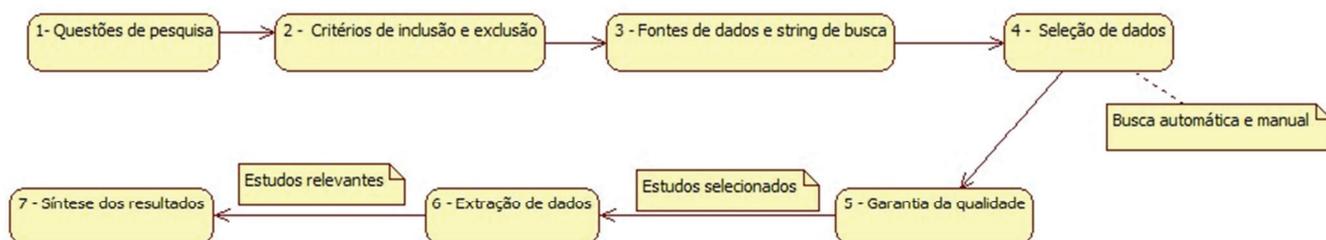
Este MSL é uma extensão do mapeamento conduzido por Cong e Cao (CONG;

CAO, 2013), retirando a parte do mapeamento onde eles avaliam métodos de desenvolvimento não ágeis, e tem o objetivo de identificar os estudos empíricos sobre técnicas e métricas de estimativas de esforço em métodos ágeis e consolidar as evidências destes estudos sobre os benefícios e as limitações destas técnicas no desenvolvimento de software.

Para que um mapeamento possa ser auditável, é sugerida a elaboração de um protocolo descrevendo todos os métodos e procedimentos de como será conduzida o mapeamento (KITCHENHAM, 2007).

Todas as etapas do protocolo da pesquisa estão representadas na Figura 4

Figura 2: Etapas do Mapeamento Sistemático



Fonte: O Autor

A seguir é apresentada uma descrição das seções do protocolo:

Questões de Pesquisa (*Research Questions – RQ*)

Com o objetivo de identificar resultados de pesquisas, práticas e aplicações sobre técnicas e métricas de estimativas de esforço em Métodos Ágeis, de modo a estender o mapeamento de Cong e Cao (CONG; CAO, 2013), este mapeamento possui duas questões de pesquisa:

RQ1: Quais métricas para medição do esforço de software estão sendo utilizadas em projetos que adotam métodos ágeis?

RQ2: Sob quais circunstâncias as métricas de para medição do esforço de software são aplicadas com sucesso?

Critérios de Inclusão e Exclusão (*Inclusion Criteria - IC e Exclusion Criteria - EC*)

Após a busca, todos os estudos retornados passaram por uma avaliação utilizando os critérios de inclusão e exclusão. Foram definidos quatro critérios de inclusão e nove de exclusão. Para que o estudo fosse aceito, ele teria que atender todos os critérios de inclusão e nenhum critério de exclusão. Caso algum critério de exclusão fosse atendido o estudo seria excluído imediatamente.

Os critérios de inclusão foram os seguintes:

- IC1. Estudos que tratem sobre medição do esforço de software em projetos de Software que utilizam métodos ágeis;
- IC2. Estudos da indústria ou da academia;
- IC3. Pesquisas qualitativas ou quantitativas;
- IC4. Estudos primários.

Os critérios de exclusão foram os seguintes:

- EC1. Estudos escritos em um idioma que não seja o Inglês ou Português;
- EC2. Estudos duplicados ou repetidos;
- EC3. Estudos que não tratem de métricas de software;
- EC4. Estudos incompletos, rascunhos, slides de apresentações ou resumos;
- EC5. Estudos terciários e meta-análises;
- EC6. Estudos acadêmicos que tratem do ensino de métodos ágeis;
- EC7. Estudos que não abordem pelo menos um método ágil;
- EC8. Artigos que não estão disponíveis gratuitamente para download nos ambientes institucionais do CIn/UFPE;
- EC9: Estudos que não respondam pelo menos a uma das questões de pesquisa.

Fontes de Dados e *String* de Busca

De modo a garantir uma busca exaustiva, e atingir a maior cobertura possível da literatura existente a respeito do tema, foram realizadas as buscas automáticas em cinco engines utilizando a *String* de busca nas principais bases de dados na área de investigação disponíveis na internet, e as manuais nas principais referências (conferências e periódicos) de engenharia de software e métodos ágeis.

Estas fontes de dados foram escolhidas com base na possibilidade da gratuidade dos downloads dos estudos por parte dos alunos do CIn/UFPE através dos ambientes institucionais do centro.

As fontes de dados foram as seguintes:

- *IEEE Xplore* (<http://www.ieeexplore.ieee.org>);
- *Scopus* (<http://www.scopus.com>);
- *ACM Digital Library* (<http://dl.acm.org>);
- *SpringerLink* (<http://springerlink.com>);
- *Science Direct* (<http://www.sciencedirect.com/>).

Para complementar a seleção dos artigos, também foram selecionados artigos de maneira manual utilizando três abordagens:

- Conferências e periódicos específicos das áreas, referentes aos últimos cinco anos;
- As referências citadas nos estudos selecionados;
- Os estudos de pesquisadores que são referências na área do estudo.

As conferências consultadas foram:

- *Agile Development Conference*;
- *XP Conference*.

Os periódicos consultados foram:

- *ACM Transactions on Software Engineering and Methodology*;
- *Empirical Software Engineering*;
- *IEEE Software*;
- *IEEE Transactions on Software Engineering*;
- *Journal of the ACM*;

A busca automática foi realizada a partir da definição de uma String de busca, derivada da questão principal da pesquisa.

A String de busca foi construída utilizando a combinação de palavras-chave da Q1 e seus sinônimos ou palavras derivadas, concatenados por meio dos operadores booleanos “OR” e “AND”. A String de busca derivada da combinação é apresentada a seguir:

((("software metrics" OR "software metric size" OR "software metric" OR "software metrics size" or "estimating software" OR "software measurement") AND ("agile" OR "agility") AND ("scrum" OR "extreme programming" OR "xp" OR "dynamic system development" OR "dsdm" OR "crystal methodologies" OR "crystal clear" OR "crystal orange" OR "crystal red" OR "crystal blue" OR "feature driven development" OR "fdd" OR "lean software development" OR "adaptive software development" OR "test driven development" OR "tdd"))))

Os termos utilizados para a String de busca foram escolhidos de forma a serem os mais abrangentes possíveis, de modo que a busca retornasse a maioria dos estudos sobre as técnicas e métricas de estimativas de esforço nos métodos ágeis.

Seleção de Dados

Os critérios de inclusão e exclusão, definidos no protocolo da revisão, foram aplicados após a consolidação dos estudos das buscas automática e manual. Também foram excluídos os artigos duplicados. A aplicação dos critérios se deu em dois momentos:

- Primeiramente o título e o abstract do estudo foram lidos.
- E no segundo, com base na leitura da Introdução, Conclusão e Metodologia, e em casos onde não ficou claro o entendimento sobre o estudo, a leitura completa do mesmo foi efetuada.

Os estudos foram divididos em três grupos, e avaliados por uma dupla de pesquisadores com objetivo de diminuir o viés da pesquisa. Cada pesquisador, individualmente, aplicou os critérios de inclusão e exclusão nos estudos que lhe foram escolhidos para avaliar. Ao final da avaliação de todos os estudos, os resultados de cada par foi comparado e para cada divergência a dupla avaliadora conversou entre si para entrar num acordo. Nos casos em que não houve acordo, um terceiro avaliador foi acionado para gerar o desempate. Nesses casos, deverá ser registrado o motivo do conflito, assim como das exclusões.

Para a consulta utilizando a String (busca automática) foi utilizada a ferramenta acadêmica chamada ReviewER (REVIEWER 2013). A grande vantagem do uso desta ferramenta se dá porque utilizando a interface da mesma é possível fazer a consulta em

todas fontes de dados de maneira única. Após a consulta, todos os estudos retornados pelas diferentes fontes ficam armazenados na ferramenta, possibilitando ao pesquisador o armazenamento dos endereços dos estudos e a leitura dos resumos. Na própria ferramenta também é possível o controle dos estudos que foram incluídos ou excluídos.

Garantia da Qualidade

A análise da qualidade foi definida com base numa adaptação do questionário definido por Dyba e Dingsøyr (DYBA, T, DINGSOYR., 2008). Os estudos foram divididos na metade para cada uma das duplas que aplicaram o questionário para cada artigo, sendo necessária a leitura completa desses estudos. As respostas do questionário foram tabuladas de tal forma que foi possível avaliar o grau de concordância/discordância das respostas entre os membros de cada dupla. Para ser considerado com um nível de qualidade aceitável, o nível de concordância foi de, no mínimo, 60%. Caso o estudo não atinja esse nível ele foi descartado. Estes critérios estão listados abaixo:

- Existe uma definição clara dos objetivos da pesquisa?
- Existe uma descrição adequada do contexto em que a pesquisa foi realizada?
- O projeto de pesquisa foi adequado para abordar os objetivos da pesquisa?
- Os dados foram recolhidos de forma que abordaram corretamente a questão de pesquisa?
- A análise dos dados foi suficientemente rigorosa?
- Existe uma declaração clara dos resultados?
- O estudo tem valor para pesquisa ou prática?
- Outro critério utilizado para avaliar a qualidade dos artigos foi a aderência da questão de pesquisa dos artigos com as questões de pesquisa descritas anteriormente.

A partir da avaliação, cada um dos critérios foi graduada em uma escala dicotômica ("sim "ou" não") e recebeu uma pontuação de acordo com a Tabela 3-1.

Tabela 3-1: Score da avaliação da qualidade.

Valor	Score
-------	-------

Sim	1
Não	0

Fonte: O Autor

De acordo com Dyba e Dingsøyr (DYBA; DINGSOYR, 2008), estes critérios incluem três questões importantes que estão relacionadas com a qualidade e foram consideradas para a nossa extensão. São eles:

- Rigor: Foi aplicada uma abordagem completa e apropriada aos métodos fundamentais no estudo de pesquisa?
- Credibilidade: As conclusões são bem apresentadas e significativas?
- Relevância: Quão úteis são os resultados para a indústria de software e a comunidade científica?

Os estudos avaliados foram classificados em quatro níveis de qualidade: baixa, média, alta e muito alta. A Tabela 3.2 foi feita como a distribuição da pontuação entre as faixas estabelecidas.

Tabela 3-2: Distribuição do score por nível de qualidade.

Baixa	Média	Alta	Muito Alta
Abaixo de 4	$5 \leq N \leq 8$	$8 \leq N \leq 12$	Acima de 12

Fonte: O Autor

Extração de Dados

No processo de extração de dados dos artigos, as seguintes informações foram catalogadas: título, autor, ano, editora, tipo de fonte (conferência ou revista), tipo de estudo (empírico/teórico), palavras chaves, localização geográfica, tipo de sujeito (aluno/profissional), a técnica utilizada para medição do tamanho do software (por exemplo, *Story Points*), metodologia ágil adotada (por exemplo XP, SCRUM) e trechos do artigo que respondam a questão de pesquisa.

Um único pesquisador fez a extração de dados e ela foi revisada pelo co-orientador da pesquisa, a partir da leitura de cada estudo, onde todas as informações citadas no parágrafo anterior foram coletadas. Qualquer outra informação que pudesse auxiliar nas respostas às questões de pesquisa também foi armazenada. Caso não respondessem à Questão Q1 os estudos foram excluídos.

Para a coleta dos dados, foi utilizado um formulário armazenado em uma planilha, para registrar as características de cada estudo e para registrar os resultados encontrados. O Anexo B ilustra esse formulário.

Síntese dos Resultados

Com relação à síntese dos dados, inicialmente foi feita uma análise temática procurando categorizar os artigos de acordo com a técnica utilizada para medição do esforço do software, metodologia ágil adotada (por exemplo, SCRUM, XP, FDD), limitações e benefícios da técnica utilizada, grau de maturidade do processo de desenvolvimento da empresa. Em seguida, foi feito um relacionamento dos resultados com as áreas temáticas identificadas.

Foram utilizadas planilhas eletrônicas do MS Excel™ para apoiar todo o processo de realização da revisão sistemática, especialmente, nas atividades de extração e síntese de dados. Além disso, também foi utilizada a ferramenta ReviewER (REVIEWER 2013), uma ferramenta para apoiar na busca dos artigos.

3.1.1 Procedimento de Busca e Seleção

Esta seção apresenta os procedimentos executados para a obtenção dos estudos para este mapeamento. É descrita a equipe envolvida nas etapas da pesquisa e são apresentados os passos conduzidos pelo mapeamento e os resultados dos mesmos.

Equipe Envolvida

Para a condução desta pesquisa, foram envolvidos quatro pesquisadores, sendo três estudantes de pós-graduação (um doutorando, e dois mestrandos) em Ciências da Computação, pelo Centro de Informática (CIn) da Universidade Federal de Pernambuco (UFPE) e um professor orientador. Entre os pesquisadores, apenas um estudante possui experiência prévia na condução de mapeamentos sistemáticos da literatura ou mapeamento sistemático, e o professor orientador auxiliou e orientou durante o processo.

Segundo Dybå e Dingsøyr (2008), A principal limitação de um mapeamento sistemático é o viés introduzido pelos pesquisadores na seleção dos estudos (DYBA; DINGSOYR, 2008). Para mitigar este viés, quase todas as etapas do mapeamento foram

conduzidas envolvendo pelo menos dois pesquisadores, com exceção das etapas de extração e síntese dos resultados, que foram conduzidas pelo autor desta dissertação.

Execução

O mapeamento sistemático foi conduzido de acordo com o protocolo apresentado no Capítulo 3. Os resultados de cada etapa da condução estão descritos abaixo.

ETAPA 1: Busca automática e manual

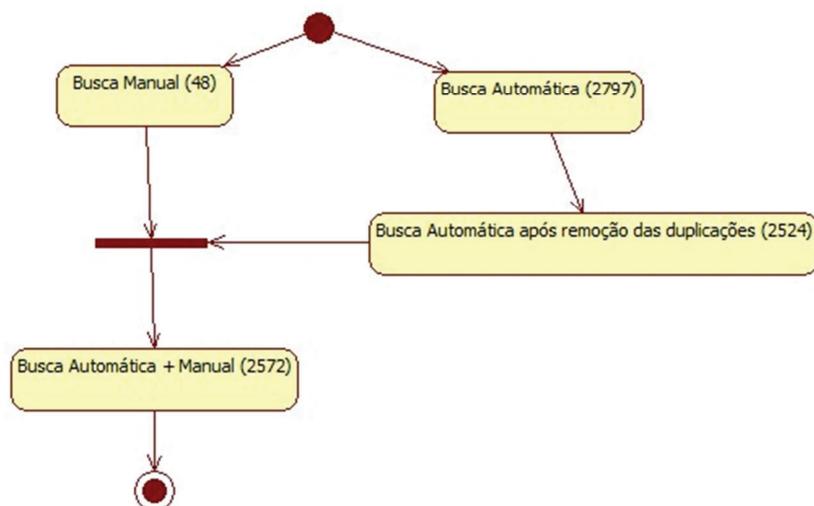
A partir da *string* de busca, da definição das fontes de pesquisa, da ferramenta Reviewer (REVIEWER, 2013) e dos demais critérios definidos no protocolo, a busca primária foi realizada no mês de dezembro de 2013, sendo conduzida por pares de pesquisadores.

A partir das buscas primárias, foram retornados 2845 estudos, dos quais 2797 foram provenientes da busca automática nos engenhos eletrônicos e 48 identificados pela busca manual. Na busca manual, vale ressaltar, que estão listados os estudos que já passaram por uma pré-seleção com base na leitura do título e abstract, os quais foram obtidos a partir da busca em periódicos e nos anais das conferências selecionadas nos anos indicados (2006 a 2013), bem como recomendados pelo orientador e pesquisadores da área ou selecionados a partir das referências contidas nos estudos recomendados.

Os estudos retornados e suas referências foram armazenados em planilhas do Microsoft Excel™ e compartilhados numa pasta no Googe Drive (<https://drive.google.com/drive/u/1/folders/0B7mWW-7Qc0gWdDhWWnZUQmxwV1U>) de modo a facilitar o acesso e o gerenciamento dos documentos.

A Figura 5 apresenta este resultado e mostra que houve alguns estudos duplicados que foram removidos.

Figura 3: Resultado do processo de busca e seleção nas fontes de pesquisa.



Fonte: O Autor

Os estudos retornados da busca automática foram quantitativamente distribuídos da seguinte forma:

- 559 no *IEEE Xplore*;
- 843 no *Scopus*;
- 828 no *ACM Digital Library*;
- 239 no *SpringerLink*;
- 376 no *Science Direct*;

ETAPA 2: Seleção dos estudos pelo título e abstract

Na busca manual, os estudos foram selecionados com base na leitura do título e *abstract*, sendo retornados 48 estudos na área. Os estudos da busca automática sem as duplicações (2572) juntamente com os estudos da busca manual foram divididos igualmente entre os pares de pesquisadores, que efetuaram a leitura do título e abstract. Apenas os estudos considerados fora da área de pesquisa foram excluídos. Assim, do total de estudos identificados, restaram 97 estudos potencialmente relevantes.

ETAPA 3: Seleção dos estudos pela introdução e conclusão

Os 97 potenciais estudos foram divididos e distribuídos para 03 grupos, cada um com dois pesquisadores, para, com base na leitura efetuada sobre a Introdução e Conclusão de cada estudo, serem aplicados os critérios de inclusão e exclusão (ver a Seção 3.1 do capítulo 3) de forma independente por pesquisador. Nas situações em que

a partir desta leitura não foi possível a aplicação dos critérios, o estudo foi lido integralmente. O resultado da análise individual da dupla foi comparado, utilizando o coeficiente Kappa (LANDIS; KOCH, 1977). O coeficiente Kappa da dupla foi de 0.84, mostrando um nível de concordância Excelente entre os pesquisadores, o que aumenta a confiabilidade dos resultados dessa fase. No final, 48 estudos foram considerados irrelevantes e 49 incluídos como potencialmente relevantes.

ETAPA 4: Avaliação da qualidade

Os estudos potencialmente relevantes que passaram pela etapa anterior foram avaliados criticamente quanto à sua qualidade, de acordo com as perguntas descritas na seção 3.1 (protocolo da pesquisa, tópico garantia de qualidade). Após a avaliação, restaram 40 potenciais estudos que foram analisados cuidadosamente para serem extraídos dados de modo a responder às questões de pesquisa.

ETAPA 5: Extração e síntese dos dados

A etapa de extração de dados foi realizada exclusivamente pelo autor desta dissertação.

Deste modo, os resultados da extração de dados são apresentados com base em 40 **estudos primários**, que foram obtidos com base na aplicação dos métodos da Teoria Fundamentada em Dados (*Grounded Theory*) e da meta-etnografia sobre a leitura dos estudos primários. Estes resultados estão descritos na Seção 4.1.3 (Mapeamento das Evidências). As referências dos estudos primários e suas características encontram-se disponíveis no **APÊNDICE A**.

3.2 Experimento Controlado

Na engenharia de software, experimentação refere-se à combinação de fatos com suposições, especulações e crenças abundantes na construção de software (JURISTO; MORENO, 2010). Oferece um modo sistemático, disciplinado, computável e controlado para avaliação da atividade humana. Experimentos são apropriados para confirmar teorias, confirmar o conhecimento convencional, explorar relacionamentos, avaliar predição de modelos, ou validar medidas. A possibilidade de repetição, juntamente com o controle total sobre o processo e as variáveis são a maior força do experimento.

3.2.1 Planejamento do Experimento

Este planejamento tem o objetivo de criar um experimento para avaliação sobre

qual das propostas encontradas no MSL é a melhor para se calcular estimativas de esforço, utilizando como unidade pontos de função/semana em projetos desenvolvidos com métodos ágeis.

Definição dos Objetivos

Esta seção descreve os objetivos a serem alcançados com a execução do experimento.

Objetivos

Avaliar qual proposta é a melhor para calcular estimativas de esforço, em projetos desenvolvidos utilizando métodos ágeis. É levado em consideração o grau de precisão e facilidade de utilização/aplicação da proposta. Um mesmo projeto é medido pelos três analistas, utilizando as três abordagens propostas baseadas em APF encontradas no MSL para fazer este tipo de medição. Após a medição, serão comparados os valores medidos com os valores reais, estes últimos obtidos de projetos finalizados anteriormente ao experimento e que possuíam a medição final do esforço despendido.

A organização dos objetivos do estudo segue a abordagem GQM (Goal Question Metric) (SOUSA, K. D. DE; OLIVEIRA, K. M. DE; ANQUETIL, N., 2005), cuja estrutura é apresentada a seguir.

- **Analisar** o processo de estimativa do esforço de desenvolvimento de abordagens que utilizam APF como base para a estimativa de esforço em projeto ágeis.
- **Com o propósito de** avaliar a utilização de diferentes abordagens de estimativas.
- **Com respeito à** precisão.
- **Do ponto de vista** da equipe de desenvolvimento.
- **No contexto de** analistas de sistemas.

Questões

Qual das abordagens é a mais precisa: Extending Function Point Analysis (BANERJEE; NARAYANAN; MAHADEVAN 2012), Function Point Analysis and Cost Estimation in An Agile Development Environment (ALEXANDER, 2009) ou Agile Estimation Using Functional Metrics (CAGLEY, 2009).

Métricas

Para se avaliar a precisão do experimento será levado em consideração os valores obtidos sobre o seguinte aspecto:

- Valor estimado (medido em pontos de função por semana).

Descrição do Experimento

Esta seção descreve quais são as hipóteses, as unidades e os objetos experimentais, os parâmetros, as variáveis de resposta e o tratamento deste estudo.

Definição das Hipóteses

Antes de apresentar as hipóteses, é necessário introduzir alguns símbolos utilizados para denotar as métricas a serem coletadas e analisadas

VE = Valor estimado

EX = Abordagem Extending Function Point Analysis

CA = Abordagem Function Point Analysis and Cost Estimation in An Agile Development Environment

FM = Abordagem Agile Estimation Using Functional Metrics

A métrica possui três variações, uma para cada abordagem analisada.

A principal hipótese do estudo é a hipótese nula, a qual implica que não há diferenças estatisticamente significativas entre as três abordagens. Portanto, o estudo tentará refutar essa hipótese. Há seis hipóteses nulas uma para cada métrica que o estudo analisa.

Hipótese Nula: O valor estimado, não é significativamente diferente utilizando as três diferentes abordagens.

H01: $VEEX \approx VECA$

H02: $VEEX \approx VEFM$

H03: $VECA \approx VEFM$

Hipótese alternativa 1: O valor estimado é mais preciso utilizando a abordagem Extending Function Point Analysis em comparação com as abordagens de Function Point Analysis and Cost Estimation in An Agile Development Environment e Agile Estimation Using Functional Metrics.

H01: $VEEX > VECA$

H02: VEEX > VEFM

Hipótese alternativa 2: O valor estimado é mais preciso utilizando a abordagem Function Point Analysis and Cost Estimation in An Agile Development Environment em comparação com as abordagens de Extending Function Point Analysis e Agile Estimation Using Functional Metrics.

H01: VECA > VEEX

H02: VECA > VEFM

Hipótese alternativa 3: O valor estimado é mais preciso utilizando a abordagem Agile Estimation Using Functional Metrics em comparação com as abordagens de Function Point Analysis and Cost Estimation in An Agile Development Environment e Extending Function Point Analysis.

H01: VEFM > VECA

H02: VEFM > VEEX

Unidade Experimental

Os objetos sobre os quais o experimento é executado são chamados de unidades experimentais ou objetos experimentais. Neste estudo foram selecionados três projetos desenvolvidos numa empresa pública de esfera municipal. Estes projetos foram escolhidos por possuírem todos os elementos (estórias completas e disponíveis, valores de tempo e esforço gasto coletados) necessários para a realização do experimento.

Sujeito Experimental

A pessoa que aplica os métodos ou técnicas na unidade experimental é chamada de sujeito experimental. Neste estudo os sujeitos foram oito analistas de sistemas que nunca tiveram contato com pontos de função nem com o processo de estimativa de esforço de desenvolvimento.

Parâmetros e Fatores

São as variáveis de entrada para o experimento controlado definindo os parâmetros (variáveis cujos valores serão mantidos como constantes ao longo do experimento) e fatores (variáveis que representam os tratamentos a serem recebidos pelos objetos experimentais).

Variáveis Independentes

- Abordagens de medição.

Variáveis Dependentes

A variável dependente também é conhecida como uma “variável de resposta”, “variável de regresso”, “variável de medida”, “variável de resultado”, “variável de experimento” e “variável de saída”.

Neste estudo, foi considerada a seguinte variável de resposta:

- Valor estimado;

Variáveis de Bloqueio

A experiência dos analistas e os projetos por serem diferentes exercem influência no resultado do experimento. Para que não haja interferência neste experimento, os oito analistas foram aleatoriamente arranjados.

3.2.1.1 Planejamento de Execução

As seções seguintes estruturam o planejamento e medidas adotadas para a execução do experimento com as respectivas justificativas. O intuito é esclarecer e documentar todas as decisões relacionadas à preparação e execução do experimento proposto.

A seção “Contexto” mostra o contexto onde o experimento foi realizado. A seção "Treinamentos" apresenta o planejamento dos treinamentos aplicados antes da execução do experimento. A seção "Perfil dos Participantes" cita as estratégias para identificação do perfil dos participantes. A seção "Desenho do Experimento" apresenta os detalhes relacionados à modelagem do experimento. A seção "Instrumentação" apresenta os detalhes relacionados às ferramentas e suas configurações para execução do experimento. A seção "Plano de Coleta" apresenta os elementos de coleta e a forma como os dados serão armazenados. A seção "Execução Piloto" apresenta informações sobre a execução piloto para validação dos instrumentos de coleta. A seção "Tratamento para Análise dos Dados" apresenta considerações de tratamentos antes do processamento dos dados.

Contexto

O experimento será executado no âmbito do programa de pós-graduação do Centro de Informática da Universidade Federal de Pernambuco. A equipe de participantes será constituída por analistas de sistemas da indústria do estado de Pernambuco. Todos os participantes possuem pelo menos cinco anos de experiência no mercado.

Treinamentos

Com a intenção de uniformizar e garantir um nível de conhecimento mínimo antes da execução do experimento, foram planejados alguns treinamentos a serem ministrados antes da execução do plano piloto do experimento. A ideia é buscar o nivelamento mínimo do conhecimento dos participantes em relação às ferramentas, os conceitos de Pontos de Função, as abordagens, e os procedimentos de coleta dos dados. Desta maneira, procura-se mitigar os efeitos que o primeiro contato dos participantes com os conceitos e as abordagens possa causar sobre o tempo total de execução das atividades.

O primeiro grupo de treinamento é relativo à técnica de Pontos de Função. Primeiramente, foram apresentados os conceitos e como se faz a contagem. Após a apresentação dos conceitos foram feitos exercícios práticos para assegurar que todos os participantes tenham compreendido bem como usar a técnica. O tempo total do treinamento foi de 4 horas.

Com o conhecimento de Pontos de Função adquiridos os participantes foram treinados nas três abordagens do experimento. Cada abordagem foi ensinada num dia distinto e para todos os participantes ao mesmo tempo. O tempo de treinamento de cada abordagem foi de 2 horas.

Perfil dos Participantes

Com o propósito de formar equipes coesas e uniformes, em que o efeito relacionado ao conhecimento e experiência possam ser mitigados, foi realizada uma etapa de caracterização dos participantes do estudo antes da formação das equipes. Esta caracterização se deu por meio da elaboração e aplicação de um questionário, onde se identificou o perfil de cada um dos integrantes. Dentre as características observadas podemos elencar:

- Nível de conhecimento do participante sobre Pontos de Função: O nível de

conhecimento do participante, em relação à utilização de Pontos de Função podem influenciar na performance da execução da atividade, independente da abordagem;

- Nível de conhecimento do participante em estimativas: O nível de conhecimento do participante, em relação a estimativas de software pode influenciar na performance da execução da atividade, independente da abordagem;
- Perfil do participante: As experiências dos analistas são diferentes e podem influenciar na execução do estudo;
- Nível de interpretação da língua inglesa: Falantes não nativos podem sentir dificuldades na compreensão e interpretação das informações o que pode vir a comprometer o desempenho na execução das atividades do estudo;

Desenho do Experimento

Sabe-se que o desenho proposto para a experimentação influencia diretamente na formação das equipes participantes. Foi escolhido o quadrado latino como modelo experimental, pois ele é o que melhor se adequa ao experimento. Segundo Juristo e Moreno (JURISTO; MORENO, 2010), o quadrado latino tem como característica que cada fator desejado que ocorre uma vez em cada linha e uma vez em cada coluna, isto é, que ocorre apenas uma vez para cada combinação possível de duas variáveis de bloqueio.

Para utilizarmos o quadrado latino (JURISTO; MORENO, 2010), precisaremos definir três equipes distintas, onde cada uma delas executará todas as abordagens em cada um dos projetos, conforme tabela 3-3. Na tabela 3-3, os valores da primeira coluna representam as equipes que executam o experimentos, e o restante indica a ordem de execução dos experimentos pelas equipes:

Tabela 3-3: Ordem de execução dos analistas por projeto/abordagem.

	<i>1º</i>	<i>2º</i>	<i>3º</i>
<i>A1</i>	FM	CA	EX
<i>A2</i>	EX	FM	CA
<i>A3</i>	CA	EX	FM

Fonte: O Autor

A partir do mapeamento sistemático feito no estudo, foram identificados nove estudos sobre estimativas de esforço com Pontos de Função. Destes nove estudos, foram escolhidos três: *Extending Function Point Analysis* (BANERJEE; NARAYANAN;

MAHADEVAN, 2012), *Function Point Analysis and Cost Estimation in An Agile Development Environment* (ALEXANDER, 2009) e *Agile Estimation Using Functional Metrics* (CAGLEY, 2009).

Dos seis estudos identificados e eliminados, quatro (SANTANA; GUSMÃO, 2009, SANTANA, LEONEO; GUSMÃO, VASCONCELOS, 2011; FUQUA, 2003 e FUQUA, 2004) falavam, entre outras coisas, sobre a correlação entre pontos por estória (*Story Points*) e pontos de função, sem fazer nenhum tipo de proposta de como utilizar Pontos de Função com métodos ágeis.

Outro estudo (HORVAT, 2011) falou das vantagens do uso e mostrou exemplos de como utilizar, todavia não demonstrou como os valores foram encontrados.

Já o estudo (KANG; CHOI; BAIK, 2010) utiliza o *Kalman Filtering Algorithm* (WELCH; BISHOP, 2001) para acompanhar diariamente o projeto e fazer ajustes nas estimativas o que inviabiliza o uso do mesmo no experimento.

Definição das Rodadas de Execução

Para a sistematização do processo de coleta dos dados alguns elementos foram definidos por sorteio:

- Abordagens/estudo E1, E2 e E3;
- Projetos P1, P2 e P3;
- Grupo de Analistas A1, A2 e A3.

Os analistas foram divididos em três grupo onde dois grupos continham três analistas e um grupo continha dois analistas. Os três grupos de analistas executaram as atividades propostas e coletaram os dados presentes no formulário de coleta, para cada um dos estudos e em ferramentas distintas. Desta forma temos nove execuções para cada uma dos projetos, conforme o desenho proposto:

Rodada 1: Grupo de Analistas A1 executa a abordagem/estudo do E1 nos projetos P1, P3, P2.

Grupo de Analistas A2 executa a abordagem/estudo do E3 nos projetos P1, P3, P2.

Grupo de Analistas A3 executa a abordagem/estudo do E2 nos projetos P1, P3, P2.

Rodada 2: Grupo de Analistas A1 executa a abordagem/estudo do E2 nos projetos P2, P1, P3.

Grupo de Analistas A2 executa a abordagem/estudo do E1 nos projetos P2, P1, P3.

Grupo de Analistas A3 executa a abordagem/estudo do E3 nos projetos P2, P1, P3.

Rodada 3: Grupo de Analistas A1 executa a abordagem/estudo do E3 nos projetos P3, P2, P1.

Grupo de Analistas A2 executa a abordagem/estudo do E2 nos projetos P3, P2, P1.

Grupo de Analistas A3 executa a abordagem/estudo do E1 nos projetos P3, P2, P1.

Instrumentação

Para tornar possível a execução do experimento e coleta dos dados foram utilizados os seguintes elementos de instrumentação:

- Softwares: Microsoft Excel e Adobe Acrobat.
- Formulário de Coleta: Foi confeccionado um formulário de coleta de dados para cada analista registrar os dados de execução e medição de cada uma das abordagens/estudos.
- Roteiros de Execução: Foram confeccionados três roteiros de execução, um para cada abordagem, orientando os analistas na coleta das informações.

Plano de Coleta

Os itens listados abaixo foram alvos do processo de coleta e foram ser registrados em cada um dos campos pertinentes no formulário de coleta:

- Quantidade de ALIs, AIEs, EEs, CEs e SEs encontrados.
- O valor calculado em Pontos de Função.
- O valor calculado do esforço do desenvolvimento em Pontos de Função/Semana.

Execução Piloto

Para validar os elementos da instrumentação e processamento dos dados do experimento, foi executada uma etapa piloto. Nesta etapa, um projeto fictício foi criado e um participante teste fez a execução do experimento, contabilizando os valores calculados em cada abordagem e o tempo gasto para fazer o experimento. Foram utilizados todos os elementos de instrumentação e a execução foi realizada na presença do pesquisador, que observou as dificuldades encontradas pelo participante teste. Através desta execução foi possível estimar o tempo que será gasto por cada participante para realizar o experimento e identificar indícios de falhas nos elementos da

instrumentação, que foram corrigidos para não comprometer a realização do experimento.

Tratamento para Análise dos Dados

Com o intento de não interferir no resultado da análise dos dados, as informações de identificação dos participantes, projetos e abordagens foram omitidas dos dados de coleta antes da análise dos dados.

3.2.1.2 Análise dos Dados

A análise deste estudo tem por objetivo comparar os dados coletados das execuções do experimento para verificar se a hipótese nula pode ser rejeitada (vide Seção 3.2.1 para mais detalhes sobre a hipótese nula). Esta análise avalia as abordagens utilizadas para estimar o esforço de desenvolvimento de software em projetos ágeis. A análise completa do experimento será mostrada no capítulo de resultados.

Conforme descrito na Seção 3.2.1 no trecho desenho do experimento, o experimento realizado teve apenas um fator, e a distribuição dos projetos e das abordagens entre os sujeitos será feita de maneira randomizada, através de sorteio, conforme detalhado na Seção Definição das Rodadas de Execução. Para rejeitar a hipótese nula foi utilizado o método estatístico ANOVA (*analise of variance*) (FISHER, 1925). Para avaliar a magnitude destas diferenças utilizou-se um teste de comparações múltiplas conhecido como o Teste de Tukey (TUKEY, 1977).

O método estatístico ANOVA (FISHER, 1925) diz que em um experimento, cada observação Y_{ij} pode ser decomposta conforme o modelo representado pela Equação a seguir:

$$Y_{ij} = \mu + \tau_i + \epsilon_{ij} \quad i = 1, \dots, I \text{ e } j = 1, \dots, J \quad (3)$$

Onde μ é o efeito constante (média geral), τ_i é o efeito do i -ésimo tratamento, ϵ_{ij} é o erro associado ao i -ésimo tratamento na j -ésima unidade experimental ou parcela.

A ANOVA, baseia-se na decomposição da variação total da variável resposta em partes que podem ser atribuídas aos tratamentos (variância entre) e ao erro experimental (variância dentro) (UFPR 2011). Essa variação pode ser medida por meio das somas de quadrados definidas para cada um dos componentes mostrado nas Equações 4,5 e 6:

$$\sum_{i=1}^n (Y_i - \bar{Y})^2 = SQT \text{ (é a Soma de Quadrados Total);} \quad (4)$$

$$\sum_{i=1}^n (\hat{Y}_i - \bar{Y})^2 = SQR \text{ (é a Soma de Quadrados da Regressão)} \quad (5)$$

$$\sum_{i=1}^n (Y_i - \hat{Y}_i)^2 = SQE \text{ (é a Soma de Quadrados dos Erros (dos Resíduos))} \quad (6)$$

Desta forma, escreve-se: $SQT = SQR + SQE$, em que a soma de quadrados total é decomposta em soma de quadrados da regressão e soma de quadrados dos erros. Essas somas de quadrados podem ser organizadas em uma tabela, denominada tabela da análise de variância (tabela 3-4).

Para testar a hipótese H_0 (Nula), utiliza-se o teste F apresentado na tabela da Análise de Variância (Tabela 3-4).

Tabela 3-4: Tabela da análise de variância.

<i>Causas da Variação</i>	<i>Graus de Liberdade</i>	<i>Soma dos Quadrados</i>	<i>Quadrados Médios</i>	<i>F Calculado</i>
Tratamentos	I-1	SQR	QMR=SQR/(I-1)	QMR/QME
Resíduo	I(J-1)	SQE	QME=SQE/(I(J-1))	
Total	IJ-1	SQT		

Fonte: O Autor

Se $F_{calculado}$ for maior que $F_{tabelado}$, então rejeitamos a hipótese de nulidade H_0 , ou seja, existem evidências de diferença significativa entre pelo menos um par de médias de tratamentos, ao nível α de significância escolhido. Caso contrário, não se rejeitamos a hipótese de nulidade H_0 , ou seja, não há evidências de diferença significativa entre tratamentos, ao nível α de significância escolhido (UFPR 2011).

Desta forma, estaremos analisando os dados coletados para tentar rejeitar a hipótese nula mostrando que as médias das amostras de cada técnica não são iguais, levando-se em conta um determinado nível de significância (α). Neste experimento,

tendo em vista a quantidade pequena de participantes a serem envolvidos, estaremos utilizando um nível de significância (α) de 0,05. Ou seja, estaremos adotando uma probabilidade de 95% de confiança nos resultados obtidos.

Após concluirmos que existe diferença significativa entre tratamentos, por meio do teste F, podemos estar interessados em avaliar a magnitude destas diferenças utilizando um teste de comparações múltiplas.

O Teste de Tukey permite testar qualquer contraste, sempre, entre duas médias de tratamentos, ou seja, não permite comparar grupos entre si. O teste baseia-se na Diferença Mínima Significativa (DMS) Δ (UFPR 2011). A estatística do teste é dada segundo a Equação 7 abaixo:

$$TSD = q_{\alpha}(k, N - k)\sqrt{QME/n} \quad (7)$$

Onde q é a amplitude total studentizada, tabelada (APENDICE G), QME é o quadrado médio do resíduo, e n é o número de repetições. O valor de q depende do número de tratamentos e do número de graus de liberdade do resíduo. Foi utilizado o nível de 5% de significância. Se o contraste for maior do que Δ , então as médias diferem ao nível α de significância (UFPR 2011).

Como o teste de Tukey é, de certa forma, independente do teste F, é possível que, mesmo sendo significativo o valor de $F_{\text{calculado}}$, não se encontrem diferenças significativas entre contrastes de médias (UFPR 2011).

3.3 Considerações Finais do Capítulo

Este capítulo apresentou os métodos de pesquisa utilizados para a condução deste trabalho, de modo a tornar os resultados mais confiáveis e possíveis de serem reproduzidos por outros pesquisadores. No capítulo seguinte serão apresentados os resultados encontrados nesta pesquisa.

4 Resultados

Este capítulo apresenta os resultados obtidos com as diversas etapas da pesquisa. Primeiramente, é detalhado o mapeamento sistemático, com seus resultados. Após isso o resultado do experimento é descrito e discutido.

4.1 Síntese dos Resultados do Mapeamento Sistemático

Esta seção apresenta os resultados obtidos com a condução do mapeamento sistemático da literatura, bem como a análise dos mesmos, permitindo responder à Questão **Q1** deste trabalho: “*Como a medição do esforço de software tem sido conduzida em projetos que adotam métodos ágeis para desenvolvimento de Software?*”. Os resultados deste estudo são apresentados levando-se em consideração duas partes distintas.

- **Mapeamento das evidências** – apresenta e descreve as evidências identificadas pelo mapeamento sistemático para cada questão de pesquisa (Q1, Q2);
- **Análise e discussão dos resultados** – apresenta uma análise dos principais resultados obtidos pela pesquisa.

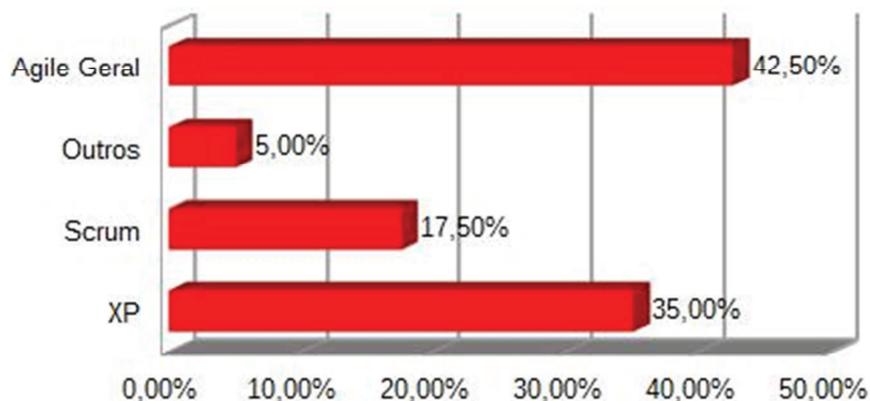
4.1.1 Visão Geral do Estudo

Nesta seção são apresentadas as informações gerais dos estudos que foram obtidas a partir da extração de dados.

Tipo de método ágil

Diversos tipos de métodos ágeis foram investigados pelos estudos, e o percentual encontrado de cada metodologia está apresentado no gráfico da Figura 4, onde se pode notar que a maioria dos estudos apresentou resultados de investigação utilizando o XP (35,00%) e o SCRUM (17,50%). Uma abordagem geral não explicitando especificamente nenhum método ágil teve uma representação considerável (42,50%) com relação aos demais métodos identificados.

Figura 4: Métodos Ágeis investigados pelos estudos

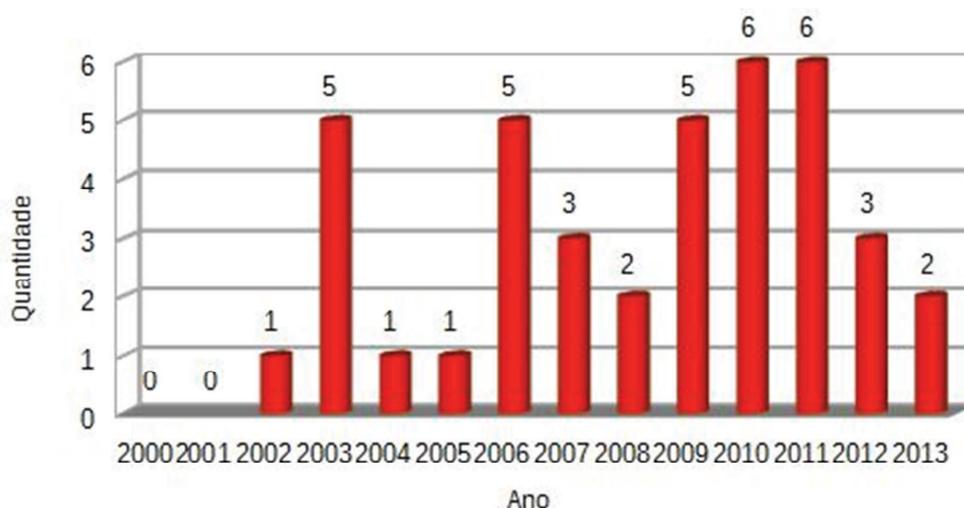


Fonte: O Autor

Distribuição temporal

A distribuição temporal dos estudos primários é apresentada no gráfico da Figura 5, onde é possível perceber, apesar de haver caso de diminuição, que os estudos estão distribuídos de forma mais ou menos uniforme durante os anos com o aumento dos números de estudos a partir do ano de 2006.

Figura 5: Distribuição temporal dos estudos primários



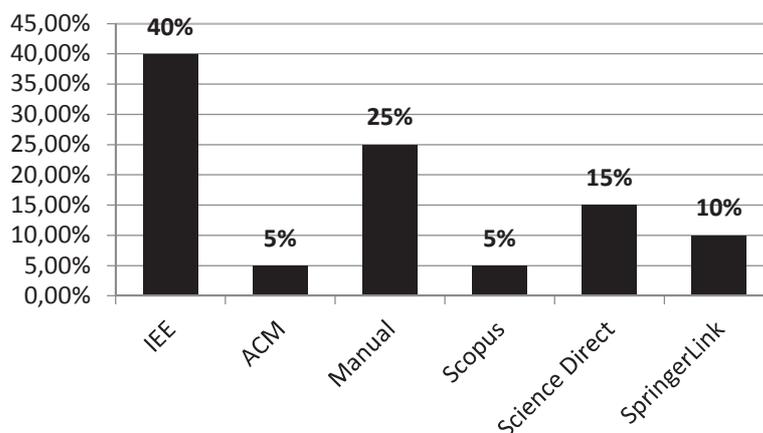
Fonte: O Autor

Distribuição por fonte de dados

A distribuição por fonte de dados é apresentada na Figura 6, onde é possível perceber que a maioria absoluta (65%) dos estudos foram encontrados na *IEEE Xplore*

(40%) e através da consulta manual (25%).

Figura 6: Distribuição por fonte de dados dos estudos primários



Fonte: O Autor

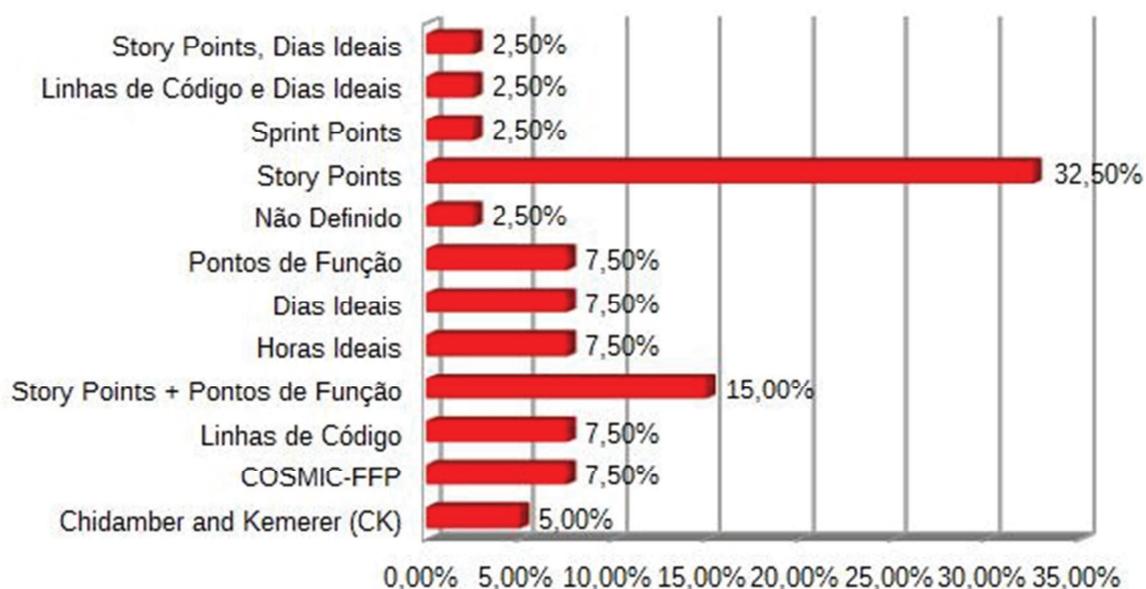
4.1.2 Mapeamento das Evidências

Nesta seção são apresentadas as evidências e os resultados para cada questão de pesquisa. As referências dos estudos primários e suas características encontram-se disponíveis no **APÊNDICE A**.

RQ1: Quais métricas para medição do esforço de software estão sendo utilizadas em projetos que adotam métodos ágeis?

Foram identificados 40 estudos que forneceram resposta para a Q1, cujos dados foram sintetizados e analisados. A tabela 4.1 apresenta as métricas encontradas, as frequências e as referências dos estudos relacionados. A distribuição percentual por métrica de esforço é apresentada na Figura 7, onde é possível verificar a predominância do uso de *Story Points* (50,00%) como unidade básica de dimensionamento, sendo que 32,50% dos estudos tratam apenas *Story Points*, 15% usam esta métrica em conjunto com pontos de função e 2,50% em conjunto com dias ideais (dias de trabalhos de 8 horas sem interrupções). Verifica-se também o grande número de estudos que usam de alguma forma pontos de função como métrica de esforço em métodos ágeis (22,50%). Vale a pena registrar que modelos tradicionais como por exemplo COCOMO II (COCOMO, 2004) não foram identificados em nenhum dos estudos.

Figura 7: Distribuição por métricas de esforço



Fonte: O Autor

Tabela 4-1: Métricas, frequências e Referências

<i>Métricas</i>	<i>Frequência</i>	<i>Referência</i>
<i>Story Points</i>	13	[P02][P04][P05][P09][P10][P11][P14][P15][P16][P18][P24][P28][P33]
<i>Story Points + Pontos de Função</i>	6	[P12][P29][P30][P31][P32][P40]
Chidamber and Kemerer (CK)	2	[P01][P07]
COSMIC-FFP	3	[P13][P20][P25]
Dias ideais	3	[P08][P23][P27]
Horas ideais	3	[P03][P17][P19]
Linhas de código	3	[P06][P22][P34]
Linhas de código + Dias ideais	1	[P21]
Não definido	1	[P36]
Pontos de função	3	[P35][P38][P39]
<i>Sprint Points</i>	1	[P26]
<i>Story Points, Dias ideais</i>	1	[P37]

Fonte: O Autor

As evidências contidas na Tabela 4-1 mostram que há uma grande quantidade de métricas existentes na literatura. Utilizando horas ideais o estudo P19, descreve um estudo empírico de estimativa de esforço no desenvolvimento de software ágil através da coleta dos esforços estimado e real de um projeto de 46 iterações. O estudo P29 fala sobre as métricas mais comuns nos métodos ágeis, falando sobre *Story Points* e dias ideais. A métrica linhas de código é utilizada no estudo P34, onde é criado um plug-in do *Eclipse* (ECLIPSE FOUNDATION, 2014) para auxiliar no desenvolvimento de software, principalmente na refatoração da medição de esforço. O estudo P07 utiliza CK e, através de vários modelos (incluindo modelos de regressão e redes neurais), cria uma nova abordagem para a predição de esforço do desenvolvimento de software.

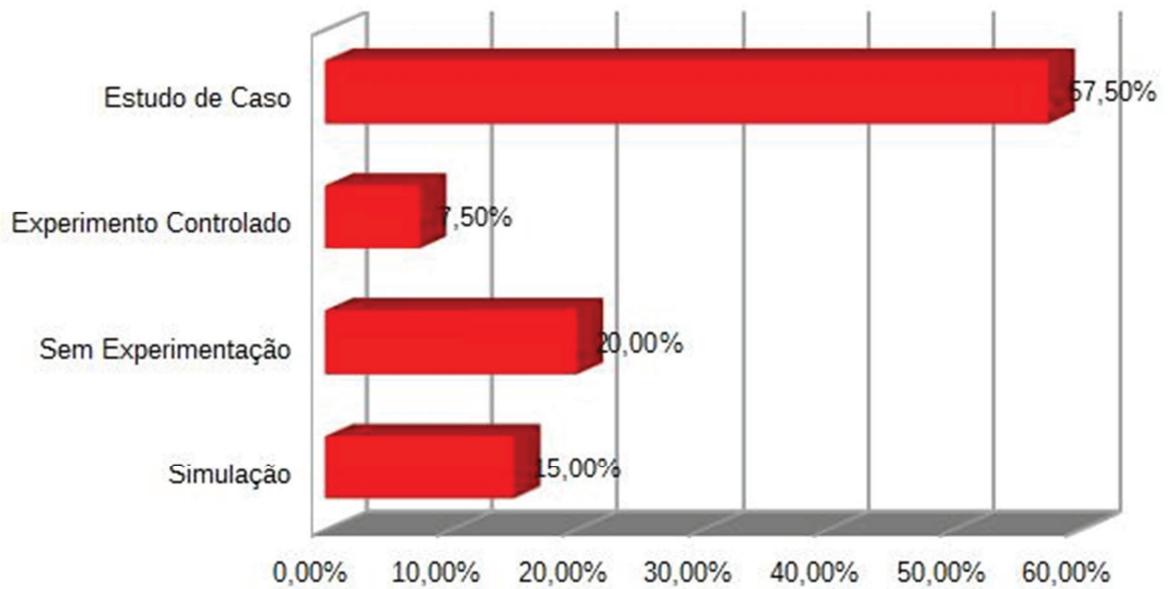
Apesar de haver uma grande quantidade de métricas existentes, pode-se afirmar que o uso de *Story Points* é a mais usada e está se tornando padrão na medição em projetos ágeis (COHN 2005). O estudo P05, utiliza *Story Points* como base para a medição, e a partir disso adapta o *Earned Value Management* (EVM) (PMBOK 2008), usando os valores definidos no Scrum. Já o estudo P33 apresenta uma linguagem de padrões, que descreve e documenta as principais técnicas e métodos utilizados na estimativa de software por equipes que adotam métodos ágeis.

RQ2: Sob quais circunstâncias as métricas para medição do esforço de software são aplicadas com sucesso?

Para serem aceitos, modelos de estimativas precisam ser validados empiricamente. A relevância das evidências empíricas é baseada no método de validação e no contexto do estudo (CONG; CAO, 2013).

Com relação à distribuição por métodos de pesquisa foram identificados cinco métodos de pesquisa nos estudos primários como mostra a Figura 8, com destaque para os estudos de caso (57,50%). Vale a pena destacar o alto número de estudos que não fizeram nenhum tipo de experimentação (20,00%).

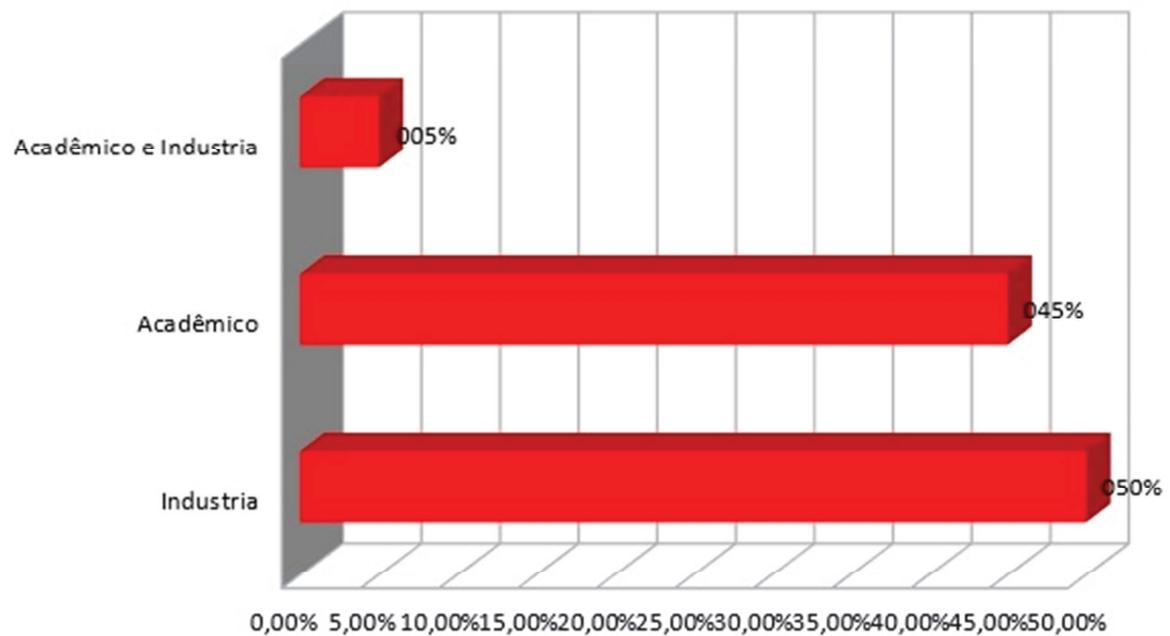
Figura 8: Distribuição por métodos de pesquisa.



Fonte: O Autor

Já na distribuição por ambiente de estudo, pode-se notar que os estudos, em sua maioria, possuem dados empíricos com pesquisas realizadas na indústria (50,00%), e em ambientes acadêmicos (45,00%). O restante foi distribuído em ambientes que misturam a academia e a indústria. A Figura 9 apresenta os valores por ambiente de estudo.

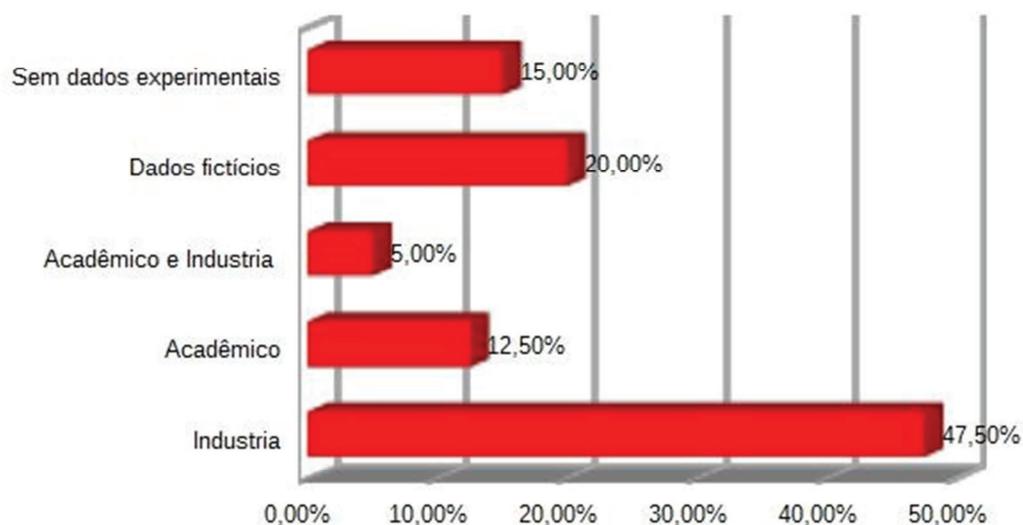
Figura 9: Distribuição por ambiente de estudo.



Fonte: O Autor

Na distribuição por tipo de dados, pode-se notar que os estudos, em sua maioria, possuem dados empíricos com pesquisas realizadas na indústria (47,50%), e em ambientes acadêmicos (12,50%). Vale também salientar a quantidade razoável de estudos que utilizaram dados fictícios (20,00%) e que não utilizaram dados experimentais (15,00%). O restante foi distribuído em ambientes que misturam a academia e a indústria. A Figura 10 apresenta os valores por tipo de dados.

Figura 10: Distribuição por tipo de dados.



Fonte: O Autor

As evidências listadas acima mostram que muitos modelos de estimativas foram validados via estudos de casos conduzidos em diferentes contextos como, por exemplo, em projetos profissionais/industriais e projetos acadêmicos. Entretanto muitos modelos ainda não foram validados empiricamente. Alguns modelos propostos ainda estão em estágios experimentais.

4.2 Síntese dos Resultados do Experimento Controlado

Esta seção apresenta os resultados obtidos com a condução do experimento controlado. A Seção 4.2.1 faz uma breve caracterização dos estudos. Na Seção 4.2.2 os dados coletados são apresentados. Na Seção 4.2.3 é demonstrado para todos os projetos que participaram do experimento que a hipótese nula foi rejeitada. Na Seção 4.2.4 a magnitude das diferenças entre as abordagens é discutida.

4.2.1 Caracterização dos Estudos

Primeiramente, antes de qualquer análise dos resultados é preciso apresentar as características de cada estudo. Essas características são:

- Estudo E1: Novo sistema simples, feito em uma única iteração.
- Estudo E2: Novo sistema com várias funcionalidades, desenvolvido em 3 iterações.
- Estudo E3: Nova funcionalidade para um sistema já existente, com

duração de 2 iterações.

No apêndice H podem ser encontrados os valores de tempo reais gastos em cada estudo.

4.2.2 Dados Coletados

Abaixo seguem os dados coletados dos participantes. As tabelas 4-2, 4-3 e 4-4, mostram os dados divididos nos três estudos. As primeiras linhas de cada tabela representam o módulo do valor do erro das estimativas em semanas. Esse valor foi obtido subtraindo o tempo estimado por cada participante pelo tempo real gasto para o desenvolvimento de cada estudo. A linha iniciada por média, mostrar a média de erro da abordagem. A linha média geral indica o valor médio do erro entre todas as abordagens. A linha efeito mostra o efeito de cada abordagem na variável de resposta. O efeito é calculado subtraindo a média de erro de cada abordagem pela média geral do experimento. Efeito positivo quer dizer que a abordagem aumentou a média geral, e o efeito negativo que a abordagem diminuiu a média geral. Vale registrar que o participante 8 não entregou os dados relativos ao estudo E2. Devido a isso o estudo E2 ficou com apenas sete experimentos válidos. No apêndice H podem ser encontrados os valores brutos encontrados por cada participante.

Tabela 4-2: Dados do experimento com o estudo E1.

<i>Estudo E1</i>	<i>Abordagem CA</i>	<i>Abordagem FM</i>	<i>Abordagem EX</i>
Participante 1	0,6375	0,1500	0,1500
Participante 2	0,6375	0,2250	0,2250
Participante 3	0,6375	0,0750	0,0750
Participante 5	0,6375	0,0375	0,0375
Participante 6	0,6375	0,4500	0,4500
Participante 7	0,6375	0,4875	0,4875
Participante 8	0,6375	0,1500	0,1500
Média geral	0,3469		
Efeito	0,2906	-0,1453	-0,1453

Fonte: O Autor

Tabela 4-3: Dados do experimento com o estudo E2.

<i>Estudo E2</i>	<i>Abordagem CA</i>	<i>Abordagem FM</i>	<i>Abordagem EX</i>
------------------	---------------------	---------------------	---------------------

Participante 1	2,5875	0,1500	0,0309
Participante 2	2,5875	0,1125	0,2441
Participante 3	2,5875	0,6750	0,8333
Participante 4	1,2750	0,1875	0,3226
Participante 5	2,5875	0,4875	0,3844
Participante 6	1,2750	0,1125	0,0084
Participante 7	1,2750	0,6000	0,5025
Participante 8	N/A	N/A	N/A
Média	2,0250	0,3321	0,3325
Média geral	0,8965		
Efeito	1,1285	-0,5643	-0,5642

Fonte: O Autor

Tabela 4-4: Dados do experimento com o estudo E3.

Estudo E3	Abordagem CA	Abordagem FM	Abordagem EX
Participante 1	1,5000	0,0750	0,3630
Participante 4	1,5000	0,2250	0,5020
Participante 5	1,5000	0,0750	0,4328
Participante 6	1,5000	0,4125	0,5896
Participante 7	1,5000	0,4125	0,5896
Participante 8	1,5000	0,0375	0,3806
Média	1,5000	0,2625	0,5010
Média geral	0,7545		
Efeito	0,7455	-0,4920	-0,2535

Fonte: O Autor

As tabelas 4-5, 4-6 e 4-7 representam os erros (resíduos) de cada observação com a relação à média de erros de cada abordagem. Esses valores representam a diferença entre o valor obtido de cada participante e a média de cada abordagem.

Tabela 4-5: Resíduos associados com cada observação do estudo E1.

Estudo P1	Abordagem CA	Abordagem FM	Abordagem EX
------------------	---------------------	---------------------	---------------------

Participante 1	0,0000	-0,0516	-0,0516
Participante 2	0,0000	0,0234	0,0234
Participante 3	0,0000	-0,1266	-0,1266
Participante 4	0,0000	-0,1641	-0,1641
Participante 5	0,0000	-0,1641	-0,1641
Participante 6	0,0000	0,2484	0,2484
Participante 7	0,0000	0,2859	0,2859
Participante 8	0,0000	-0,0516	-0,0516

Fonte: O Autor

Tabela 4-6: Resíduos associados com cada observação do estudo E2.

Estudo P2	Abordagem CA	Abordagem FM	Abordagem EX
Participante 1	0,5625	-0,1821	-0,3014
Participante 2	0,5625	-0,2196	-0,0882
Participante 3	0,5625	0,34286	0,5010
Participante 4	-0,7500	-0,1446	-0,0097
Participante 5	0,5625	0,1554	0,0521
Participante 6	-0,7500	-0,2196	-0,3239
Participante 7	-0,7500	0,2678	0,1702
Participante 8	N/A	N/A	N/A

Fonte: O Autor

Tabela 4-7: Resíduos associados com cada observação do estudo E3.

Estudo P3	Abordagem CA	Abordagem FM	Abordagem EX
Participante 1	0,0000	-0,1875	-0,1380
Participante 2	0,0000	-0,1875	-0,1380
Participante 3	0,0000	0,5250	0,2865
Participante 4	0,0000	-0,0375	0,0010
Participante 5	0,0000	-0,1875	-0,0682
Participante 6	0,0000	0,1500	0,0886
Participante 7	0,0000	0,1500	0,0886
Participante 8	0,0000	-0,2250	-0,1204

Fonte: O Autor

4.2.3 Hipótese Nula

Como cada estudo possui características próprias, não foi possível agrupar os resultados em um único experimento, por isso cada um deles foi tratado como um experimento separado e devido a isso a hipótese nula foi rejeitada em cada experimento.

As tabelas 4.2, 4.3 e 4.4 mostram os dados com os estudos E1, E2 e E3 respectivamente. Conforme descrito no capítulo 3, foi utilizado o método estatístico ANOVA (FISHER, 1925) que diz que devemos comparar o valor $F_{\text{calculado}}$ com o valor F_{tabelado} , e se $F_{\text{calculado}} > F_{\text{tabelado}}$ podemos rejeitar a hipótese nula. Como os projetos P1 e P3 possuem os mesmos graus de liberdade de tratamentos e de resíduo (2 e 21 respectivamente) o valor F_{tabelado} para os experimentos foi o mesmo. Consultando a tabela no apêndice F com os valores $F_{(2,21)}$ vemos que o valor é de 3,467. Como o projeto P2 teve menos dados o temos que encontrar o valor de $F_{(2,18)}$ e o valor encontrado foi de 3,555. Como podemos ver nas tabelas 4-8, 4-9 e 4-10 o valor calculado de F foi sempre maior que o valor tabelado, e por isso podemos rejeitar a hipótese nula em todos os experimentos realizados.

Tabela 4-8: Dados da Análise de Variância do experimento com o estudo E1.

<i>Causas da variação</i>	<i>Graus de liberdade</i>	<i>Soma dos quadrados</i>	<i>Quadrados médios</i>	<i>F calculado</i>
Tratamentos	2	1,0136	0,5068	
Resíduo	21	0,4384	0,0209	
Total	23			24,2755

Fonte: O Autor

Tabela 4-9: Dados da Análise de Variância do experimento com o estudo E2.

<i>Causas da variação</i>	<i>Graus de liberdade</i>	<i>Soma dos quadrados</i>	<i>Quadrados médios</i>	<i>F calculado</i>
Tratamentos	2	13,3722	6,6861	
Resíduo	18	3,8035	0,2113	
Total	20			31,6423

Fonte: O Autor

Tabela 4-10: Dados da Análise de Variância do experimento com o estudo E3.

<i>Causas da</i>	<i>Graus de</i>	<i>Soma dos</i>	<i>Quadrados</i>	<i>F calculado</i>

<i>variação</i>	<i>liberdade</i>	<i>quadrados</i>	<i>médios</i>	
Tratamentos	2	6,8967	3,4484	
Resíduo	21	0,6331	0,0301	
Total	23			114,37

Fonte: O Autor

4.2.4 Teste de Tukey

Para cada projeto experimentado foi calculado a diferença mínima significativa (DMS). Tukey (TUKEY, 1977) diz que se a diferença entre as médias for superior a DMS podemos rejeitar a hipótese de igualdade entre as médias dos níveis. As tabelas 4-11, 4-12 e 4-13 mostram esses valores. Nos estudos E1 e E2 conforme mostrado pelas tabelas 4-11 e 4-12 respectivamente, verificamos que a DMS de ambas (0,2 no estudo E1 e 0,63 no estudo E2) é superior às diferenças entre as médias de cada abordagem, por isso não podemos rejeitar a hipótese de igualdade entre as médias. No estudo E3 apresentado na tabela 4-13 podemos verificar que a DMS para esse experimento foi de 0,22 e as médias das Abordagens *Extending Function Point Analysis (EX)*, *Function Point Analysis and Cost Estimation in An Agile Development Environment (CA)* e *Agile Estimation Using Functional Metrics (FM)* foram respectivamente 1,5000, 0,2625 e 0,5010. Como as diferenças entre todas as abordagens foram superiores a DMS, pode-se rejeitar a hipótese de igualdade entre as médias com nível de 95% de significância. Como as médias representam os erros nas estimativas pode-se concluir que no estudo E3 a abordagem *Function Point Analysis and Cost Estimation in An Agile Development Environment (CA)* foi a que melhor se comportou. A seção 3.2.1 contém mais detalhes sobre o Teste de Tukey.

Tabela 4-11: Valores das médias e diferenças entre elas do estudo E1.

<i>Estudo E1</i>	<i>Média</i>	<i>Diferença com CA</i>	<i>Diferença com FM</i>	<i>Diferença com EX</i>
<i>DMS</i>	0,2000	N/A	N/A	N/A
<i>Abordagem CA</i>	0,6375	N/A	0,4359	0,4359
<i>Abordagem FM</i>	0,2016	-0,4359	N/A	0,0000
<i>Abordagem EX</i>	0,2016	-0,4359	0,0000	N/A

Fonte: O Autor.

Tabela 4-12: Valores das médias e diferenças entre elas do estudo E2.

<i>Estudo E2</i>	<i>Média</i>	<i>Diferencia com CA</i>	<i>Diferencia com FM</i>	<i>Diferencia com EX</i>
<i>DMS</i>	0,6300	N/A	N/A	N/A
<i>Abordagem CA</i>	2,0250	N/A	1,6929	1,6925
<i>Abordagem FM</i>	0,3321	-1,6929	N/A	-0,0004
<i>Abordagem EX</i>	0,3325	-1,6925	0,0004	N/A

Fonte: O Autor.

Tabela 4-13: Valores das médias e diferenças entre elas do estudo E3.

<i>Estudo E3</i>	<i>Média</i>	<i>Diferença com CA</i>	<i>Diferença com FM</i>	<i>Diferença com EX</i>
<i>DMS</i>	0,2200	N/A	N/A	N/A
<i>Abordagem CA</i>	1,5000	N/A	1,2375	0,9990
<i>Abordagem FM</i>	0,2625	-1,2375	N/A	-0,2385
<i>Abordagem EX</i>	0,5010	-0,9990	0,2385	N/A

Fonte: O Autor

4.2.5 Avaliação dos Resultados

Como mostrado na seção 4.2.3, a hipótese nula foi facilmente rejeitada, o que mostra que as abordagens são diferentes entre si.

Em todos os estudos a abordagem *Function Point Analysis and Cost Estimation in An Agile Development Environment* – CA (ALEXANDER, 2009) apresentou números muito diferentes das outras abordagens e um nível de erro elevado. Isso se dá porque a abordagem só leva em conta a contagem dos ALIs, e a partir dessa contagem, calcula o tempo que é gasto para se desenvolver a funcionalidade (atribui o valor de 35 pontos de função para cada ALI encontrado e diz que cada ponto de função leva 2,5 horas para ser desenvolvido). O experimento mostrou que determinar um número fixo de pontos de função e horas para qualquer tipo de ALI em qualquer tipo de contexto mostrou-se equivocado. Como a forma de cálculo das abordagens: *Extending Function Point Analysis* – EX (BANERJEE; NARAYANAN; MAHADEVAN 2012) e *Agile Estimation Using Functional Metrics* – FM (CAGLEY, 2009) para projetos com uma única iteração é semelhante, o resultado do estudo E1 foi o mesmo para as duas

abordagens. Como a média de erro foi pequena é possível afirmar que qualquer uma das abordagens se mostra aceitável para ser utilizada em projetos com características semelhantes ao estudo E1. No estudo E2, as abordagens *Extending Function Point Analysis* – EX e *Agile Estimation Using Functional Metrics* – FM também apresentaram resultados parecidos e uma média de erro reduzida e por isso pode-se concluir que em projetos com características semelhantes ao estudo E2, ambas são abordagens aceitáveis.

No estudo E3, a abordagem *Agile Estimation Using Functional Metrics* – FM foi, consideravelmente, a melhor abordagem para medir projetos com características semelhantes ao estudo. Isso se deu em grande parte por causa de um problema da abordagem *Extending Function Point Analysis* – EX. Ela possui uma regra que afirma que em novas histórias desenvolvidas dentro de uma interação, apenas 53,52% das atividades podem ser desenvolvidas numa mesma interação, ou seja, só é possível concluir até 53,52% do desenvolvimento da nova história. Isso faz com que iterações que possuam apenas uma única história tenham uma produtividade baixa, já que ficam limitadas à percentagem citada, que é o caso do estudo E3.

4.3 Considerações Finais do Capítulo

Este capítulo apresentou os principais resultados da pesquisa. No mapeamento sistemático foram identificados 40 estudos primários e a partir deles foram respondidas as questões de pesquisa. No experimento controlado a partir da técnica ANOVA foi rejeitada a hipótese nula e utilizando o teste de Tukey a magnitude das diferenças entre as abordagens foi discutida. No capítulo seguinte serão apresentadas as considerações finais do estudo juntamente com as ameaças à validade, dificuldades e trabalhos futuros.

5 Conclusões e Considerações Finais

5.1 Conclusões

Em Engenharia de Software e no Gerenciamento de Projetos, um método é um conjunto estruturado de práticas que pode ser repetível durante o processo de produção de software. Atualmente, eles são agrupados em 2 grandes grupos: tradicionais e os métodos ágeis.

O contexto tecnológico atual vem trabalhando com frequência com os métodos

ágeis. Como discutido na fundamentação teórica, eles se sobressaem quando nos projetos há muitas mudanças, os requisitos são passíveis de alterações, o ambiente das organizações é dinâmico, as datas de entrega do software são curtas e o desenvolvimento rápido é fundamental. Esse mesmo contexto e o próprio processo de desenvolvimento demandam a estimativa do esforço de desenvolvimento software, pois ela é fundamental para o planejamento, gestão e controle de um projeto de software. Contudo, variados métodos, com fundamentos, elementos de ações particulares, possibilitam essa ação (problema de pesquisa). A partir do mapeamento sistemático da literatura que encontrou 40 estudos foram encontrados 11 técnicas relevantes.

Dentre eles, essa pesquisa escolheu o uso de APF como base para o processo de esforço. Unindo APF com métodos ágeis temos uma ótima ajuda para estimar esforço e duração do projeto, também nos ajuda a melhorar o processo de desenvolvimento através de indicadores que permitem a comparação entre os resultados obtidos pelas equipes.

Após a escolha de APF com base, essa pesquisa questionou quais técnicas são aplicáveis e dentre elas qual a mais eficazes e adequadas no contexto de métodos ágeis. A partir do mapeamento sistemático feito, foram escolhidas três: *Extending Function Point Analysis*, *Function Point Analysis and Cost Estimation in An Agile Development Environment* e *Agile Estimation Using Functional Metrics*.

A partir dos resultados do experimento controlado, percebeu-se que a abordagem *Function Point Analysis and Cost Estimation in An Agile Development Environment - CA* é uma abordagem muito específica para o ambiente do estudo do artigo no qual a abordagem foi descrita, pois ela determina valores fixos de horas para cada ALI encontrado independente das funcionalidades que vão ser desenvolvidas, e por isso não se comportou bem quando foi utilizado em um ambiente diferente do proposto pelo autor. Percebeu-se também que apesar das abordagens *Extending Function Point Analysis - EX* e *Agile Estimation Using Functional Metrics - FM* possuem semelhanças e encontrarem valores próximos, a abordagem FM apresentou uma média de erro menor, principalmente com o estudo E3 e por isso foi considerada a mais adequada para o estudo.

5.2 Limitações e Ameaças à Validade

Esta seção tem o propósito de discutir a validade dos resultados obtidos com o

estudo e a possibilidade de expandi-los para uma população maior. Ou seja, replicá-lo em maiores proporções.

5.2.1 Mapeamento Sistemático da Literatura

Segundo (SJØBERG; HANNAY; HANSEN; KAMPENES; KARAHASANOVIC; LIBORG; REKDAL, 2005 apud KITCHENHAM, 2010), as principais limitações e ameaças à validade num mapeamento sistemático da literatura são:

1. Imprecisão na extração dos dados;
2. Seleção de estudos num período de tempo específico;
3. Viés de seleção dos estudos;
4. Credibilidade;
5. Transferência dos resultados.

Para mitigar essas limitações e ameaças, adotou-se a seguinte estratégia:

1. Foi descrito o protocolo e suas fases, incluindo extração dos dados e o processo de síntese;
2. Em cada fase do mapeamento, realizou-se a seleção com dois ou mais pesquisadores e os conflitos foram resolvidos em reunião entre os pesquisadores;
3. Selecionaram-se todos os estudos publicados entre os anos de 2000 e 2013;
4. Para evitar viés na seleção dos estudos, guias para auxiliar foram preparados e reuniões foram conduzidas em cada etapa da pesquisa. Cada etapa foi conduzida por mais de um pesquisador. A extração de dados foi a única etapa que foi conduzida por um único pesquisador, e auxiliada pelos orientadores do trabalho;
5. Foi realizada a busca automática em cinco engines de busca. Para aumentar a cobertura do mapeamento e reduzir a possibilidade de algum estudo potencialmente relevante não ter sido retornado na condução da busca automática, foi realizada a busca manual nos periódicos e nos anais dos principais periódicos e conferências de Engenharia de Software e de Metodologias Ágeis.

5.2.2 Experimento Controlado

Esta seção discute as ameaças à validade e à confiabilidade empírica do experimento controlado. Com relação às ameaças a validade existem quatro tipos de validade:

- Validade interna: discute a relação entre os dados coletados e as variáveis dependentes. Se o dado coletado veio realmente dos tratamentos ou foram influenciados por um fator descontrolado.
- Validade de conclusão: avalia a correteza da conclusão tomada com o experimento, se realmente o teste estatístico foi aplicado corretamente e as medidas foram confiáveis.
- Validade de construção: discute a relação causa efeito do experimento. Garante que a causa se refere ao tratamento e os dados coletados refletem o efeito do experimento.
- Validade externa: discute se o experimento pode generalizar os resultados para serem aplicados na indústria.

Validade interna

Neste experimento foi utilizado como sujeitos analistas de sistemas que ao longo do desenvolvimento podem sofrer influência de fatores que alterem o resultado do experimento. Com relação ao comprometimento dos envolvidos, eles podem ficar desestimulados pelo tempo de trabalho, mas como forma de contornar esta situação a foi acordado com a gerência dos mesmo que eles poderiam ganhar horas de folga pela ajuda no projeto. O cansaço de estimar pode ser outro fator, para isto o prazo de entrega das estimativas foi aumentado em 40% para não afetar a pressão de entrega das estimativas.

Validade de Conclusão

Esta validade se refere à correteza da aplicação do teste estatístico nos resultados obtidos durante o experimento, para segurança será acompanhado por um profissional de estatística. Neste estudo foi usado o teste de Tukey (TUKEY, 1977) para comparar os dados resultantes dos tratamentos.

Validade de Construção

O treinamento aplicado aos desenvolvedores nas métricas de estimativa pode não ser compreendido suficientemente por todos os envolvidos e assim pode afetar os

resultados deste experimento.

Validade externa

Neste estudo foram testadas três métricas de estimativa de medição para o esforço de um software desenvolvido utilizando uma metodologia ágil. Como forma de garantir validade externa, pode-se repetir o experimento em outros grupos com características distintas (por exemplo, alunos de pós-graduação), de tal forma que se tenha uma maior garantia que os resultados podem ser generalizados.

Confiabilidade Empírica

Os procedimentos para execução da pesquisa foram documentados de maneira detalhada, procurando fornecer o máximo de informação de tal forma que seja possível que outros pesquisadores repitam os experimentos, contribuindo assim para uma maior confiabilidade empírica.

5.3 Dificuldades

Esta seção tem o propósito de discutir as dificuldades encontradas para a realização da pesquisa.

5.3.1 Mapeamento Sistemático da Literatura

A grande dificuldade na realização do mapeamento sistemático da literatura foi o acesso completo ao material dos estudos. Apesar do Centro de Informática (CIn) da UFPE possuir permissão de acesso a artigos de diversos engines de buscas, muito dos arquivos foram provenientes dos engines de buscas que o CIn não tinha acesso. Para minimizar esta dificuldade, entrou-se em contato com os autores explicando o estudo e pedindo que os mesmos disponibilizassem o trabalho. Alguns autores, como A. M. Fuqua, e C. A. Santana disponibilizaram seus estudos, todavia muitos autores simplesmente não responderam ao e-mail com o pedido.

5.3.2 Experimento Controlado

A principal dificuldade encontrada na realização do experimento foi a de encontrar analistas dispostos a participar. Dos trinta e nove analistas mapeados com possíveis experimentadores apenas dezoito se mostraram dispostos a fazerem o

experimento, todavia menos de 50% iniciou o experimento e destes que iniciaram apenas oito concluíram com êxito. Outro problema foi a escolha dos projetos, uma vez que os projetos precisavam estar finalizados, terem sido feitos utilizando algum método ágil, o tempo de desenvolvimento devidamente coletado e as estórias disponíveis. Dos quinze projetos finalizados que foram feitos com um método ágil, apenas sete possuíam todas as estórias disponíveis e apenas três tinham coletado o tempo de desenvolvimento.

5.4 Contribuições

A seguir são apresentadas as contribuições e os resultados com o desenvolvimento deste trabalho:

- Contribuir com uma coleção de estudos sobre métricas de esforço em métodos ágeis;
- Contribuir na escolha de um método de estimativa de esforço que utiliza técnica de APF em empresas que utilizam métodos ágeis no desenvolvimento de software;
- Permitir que novas pesquisas sejam realizadas pela comunidade científica e pela indústria, a partir dos resultados e lacunas na área desta pesquisa.

5.5 Trabalhos Futuros

O baixo número de estudos que abordaram a temática nos últimos anos e a importância prática dos objetos de estudo dessa pesquisa evidenciam há necessidade de mais pesquisas nesta área, tanto na academia quanto na indústria. Assim, são recomendados os seguintes trabalhos futuros:

- Desenvolver estudos de caso sobre a temática envolvendo empresas de desenvolvimento de software brasileiras;
- Desenvolver estudos de pesquisa-ação para testar se os resultados do experimento que foram encontrados se confirmam com outras empresas no cenário brasileiro;
- Desenvolver estudos que ampliem a amostra para analisar o impacto sob a perspectiva de outros stakeholders, como por exemplo, gerentes de projetos;

5.6 Considerações finais

Por fim, a estrutura lógica e os resultados deste estudo nos dá alguma resposta para essa pesquisa no campo do mercado e no âmbito acadêmico. No geral, as abordagens estudadas não possuem grandes diferenças entre si. Em todas elas, o método de contagem é bem parecido e de fácil execução, todavia as abordagens foram pouco testadas e validades em ambientes distintos aos que foram propostas, e por isso, precisam ser melhor estudadas para serem consideradas um padrão de uso. Os resultados da pesquisa mostram salienta que o conhecimento do negócio é fundamental para a maior precisão da contagem, quanto mais conhecimento na área do projeto contado maior foi o grau de acerto. Considerando a indústria de *software* brasileira, o conhecimento do negócio no geral é bastante satisfatório e por isso, considero adequado o uso de alguma abordagem para o contexto em que as empresas brasileiras estão inseridas.

Referências

ABDEL-HAMID T. K.; **Investigating the cost/schedule trade-off in software development**, IEEE Software (7:1) 1990, pp 97-105, 1990.

ABRAHAMSSON P. et al; **Agile software development methods Review and analysis**. Finlândia: VIT Publications, 2002.

ABRAHAO S.; INSFRAN E.; **A Metamodeling Approach to Estimate Software Size from Requirements Specifications. Software Engineering and Advanced Applications**. XXXIV, pp. 465-475. Washington (DC): IEEE Computer Society, 2008.

ACTION; Disponível em <http://www.portalaction.com.br/content/14-an%C3%A1lise-de-vari%C3%A2ncia>, 2014.

AGILE MANIFESTO; **Manifesto for Agile Software Development**, 17 February 2001. Disponível em: <http://www.agilemanifesto.org/>. Acessado em 14 abril 2014, 2001.

AGUIAR M.; **Pontos de Função ou Pontos por Caso de Uso? Como Estimar Projetos Orientados a Objetos**. Disponível em: http://www.bfpug.com.br/Artigos/UCP/Aguiar-Pontos_de_Funcao_ou_Pontos_por_Caso_de_Uso.pdf. Acessado em 31 Outubro 2013, 2003.

AGUIAR M.; IFPUG Director of International Affairs, private communication, January 2014, 2014.

ALBRECHT A. J.; **Measuring Application Development Productivity. In Proc. IBM Applications Development Symposium**. GUIDE Int and Share Inc., IBM Corp., Monterey, pp. 83, 1979.

ALEXANDER A. J.; **Case Study: Function Point Analysis and Cost Estimation in An Agile Development Environment**. Disponível em <http://www.devdaily.com/fpa/fpa-louisville-agile/>, 2011.

de ALBUQUERQUE F. J. R.; **A Técnica de análise de ponto de função como uma ferramenta de gestão estratégica da organização**, 2012.

ANDRADE E. L. P.; **Pontos de Casos de Uso e Pontos de Função na Gestão de Estimativa de Tamanho de Projetos de Software Orientados a Objeto**. Proposta de Tese de Mestrado defendida em 2004 na Universidade Católica de Brasília, Brasília, Brasil, 2004.

AMBLER S.; **Modelagem ágil: práticas eficazes para a Programação Extrema e o Processo Unificado**. Porto Alegre: Bookman, 2004.

ASHMAN R.; **Project Estimation: A Simple Use-Case-Based Model**, IT Pro, Vol. 6, Issue 4, July/August 2004, pp. 40-44, 2004.

ATI; **Agência Estadual de Tecnologia da Informação (ATI) do Estado de Pernambuco**, Disponível em: www.ati.pe.gov.br. Acessado em 02 de junho de 2014,2006.

BANDEIRA-DE-MELLO, R., CUNHA, C.; **Grounded Theory**. In: Godoi, C. K., Bandeira-de-Mello, R., Silva, A. B. d. (eds), Pesquisa Qualitativa em Estudos Organizacionais: Paradigmas, Estratégias e Métodos, São Paulo, 2006.

BANERJEE A. U.; NARAYANAN B. K.; MAHADEVAN P C.; **Estimating Agile Iterations by Extending Function Point Analysis**. In: WORLDCOMP'12, 2012.

BECK K.; **Extreme Programming Explained: Embrace Change**. [S.l.]: Addison-Wesley,1999.

BECK K.; **Extreme Programming Explained: Embrace Change**. 2. ed. [S.l.]: Addison-Wesley, 2004.

BFPUG; **Brazilian Function Point Users Group, Número de CFPS por País**. Disponível em: www.bfpug.com.br, Acesso em: 29 Janeiro 2014, 2008.

BOEHM B.; **A View of 20th and 21st Century Software Engineering**. 28th International Conference on Software Engineering,2006.

BOEHM B. W.; MADACHY R.; STEECE B.; **Software Cost Estimation with Cocomo II with Cdrom**. Prentice Hall PTR, 2000.

BOEHM B.; TURNER R.; **Management Challenges to Implementing Agile Processes in Traditional Development Organizations**. IEEE Software, Vol. 22(5), September–October. pp. 30-39, 2005.

BRITTO R.; USMAN M.; MENDES E.; **Effort Estimation in Agile Global Software Development Context**. In: Agile Methods. Large-Scale Development, Refactoring, Testing, and Estimation. Springer International Publishing, 2014. p. 182-192, 2014.

CAGLEY T.; **Agile Estimation Using Functional Metrics**. The IFPUG Guide to IT and Software Measurement IFPUG CRC Press, 2009.

CAMPOS V.; **Desenvolvimento Ágil e Pontos de Função: Gerenciando o projeto de maneira ágil e alinhado ao IFPUG**. Acesso em 05 de janeiro de 2015. Disponível em <http://pt.slideshare.net/vcampos/desenvolvimento-gil-e-pontos-de-funo-gerenciando-o-projeto-de-maneira-gil-e-alinhado-ao-ifpug-148981958>, 2012.

CAO L.; **Estimating agile software project effort: An empirical study**. In Proceedings of 14th Americas Conference on IS (Toronto, Canada, 2008).1907-1916, 2008.

CASTRO J. F.; **Introdução à Medição de Software através de Ponto de Função**. XII Simpósio Brasileiro de Engenharia de Software. Maringá, PR – Brasil, 1998.

CHOW T.; CAO D.-B.; **A survey study of critical success factors in agile software projects**. The Journal of Systems and Software, New York, NY, USA, 2008.

COCOMO; **Cocomo II Model Definition Manual**. <http://sunset.usc.edu/research/COCOMOII>, 2004.

COHN M.; **Agile Estimation and Planning**: Addison-Wesley, 2005.

COHN M.; **An Overview of Scrum**. Acesso em 22 de abril de 2014, disponível em Mountain Goat Software: <http://www.mountaingoatsoftware.com/presentations/30-anoverview-of-scrum-january>, 2002.

CONG D. N.; CAO D. T. ; **A Review of Effort Estimation Studies in Agile, Iterative and Incremental Software Development**. Can Tho University, Vietnam. 2013 IEEE RIVF Intt. Conf. on Computing & Communication Technologies - Research, Innovation, and Vision for the Future (RIVF), 2013.

COCKBURN A.; HIGHSMITH J.; **Agile Software Development: The Business of Innovation**, IEEE Computer, 2001.

COSMIC; **Common Software Measurement International Consortium, O Método COSMIC para Medição de Tamanho Funcional**: Visão Geral do Método, 2007.

CRUZES D. S.; DYBA, T.; **Recommended Steps for Thematic Synthesis in Software Engineering**. ESEM, 2011.

DEMARCO T.; **Controlling Software Projects: Management, Measurement and Estimation** Yourdon Press, New York, NY, 1982.

DEKKERS C.; **Measuring the logical or function a Size of Software Projects and Software Application**. Spotlight Software, ISO Bulletin May 2003.

DYBA T; DINGSOYR. T.; **Empirical studies of agile software development: A systematic review**. Information and Software Technology, Butterworth-Heinemann Newton, MA, USA, 2008.

EASTERBROOK S. et al; **Selecting empirical methods for software engineering research**. In Guide to advanced empirical software engineering (pp. 285-311). Springer London, 2008.

ECLIPSE FOUNDATION; Disponível em <http://www.eclipse.org>, 2014.

FERREIRA S. B. L.; NUNES R. R.; **e-Usabilidade**. São Paulo: LTC, 2008.

FLORAC W. A.; CARLETON, A. D.; **Measuring the Software Process: Statistical Process Control for Software Process Improvement**. Addison-Wesley, MA, 1999.

FISHER R. A.; **Statistical methods for research workers**. Edinburgh: Oliver & Boyd, 1925.

FRANÇA L. P. A.; VON STAA A.; LUCENA C. J. P.; **Medição de Software para Pequenas Empresas: uma Solução Baseada na Web**. Rio de Janeiro: PUC-RJ, 1998.

FUQUA A. M.; **Using Function Points in XP Considerations**. In: Extreme Programming and Agile Processes in Software Engineering, LNCS vol. 2675/2003, pp.1013, 2003.

FUQUA A. M.; **Using Function Points in Extreme Programming**. Disponível em [http://groups.yahoo.com/group/extremeprogramming/files/XP and FPA.pdf](http://groups.yahoo.com/group/extremeprogramming/files/XP%20and%20FPA.pdf), 2004.

GOVERNO FEDERAL; Disponível em <http://www.governoeletronico.gov.br/anexos/instrucao-normativa-n-04>. Acessado em 29 de abril de 2014, 2008.

HAUGEN N. C.; **An Empirical Study of Using Planning Poker for User Story Estimation** In Agile Conference, 2006, pp.23-31, 2006.

HIGHSMITH J.; **Agile Project Management – Creating Innovative Products**, AddisonWesley. International Journal of Social Research Methodology, vol. 8, nº 1, p. 19 – 32, 2005.

HORVAT D.; **Function Point Analysis and Agile Methodology**. Disponível em: <http://www.qpmg.com>. Acessado em 15 de abril de 2013, 2011

IFPUG; **Function Point Counting Pratices**. Manual: Release 4.3.1. International Function Point Users Group, Disponível em www.ifpug.org. Acessado em 6 de novembro de 2013, 2010.

ISO/IEC 20926; Disponível em http://www.iso.org/iso/catalogue_detail.htm?csnumber=35582. Acessado em 18 de janeiro de 2014, 2002.

ISO/IEC 14143; Disponível em http://www.iso.org/iso/catalogue/catalogue_tc/catalogue_detail.htm?csnumber=31917 . Acessado em 25 de janeiro de 2014, 2002.

JONES C.; **Function points**. Computer, v. 27, n. 8, p. 66-67, Aug. 1994.

JONES C.; **Applied software measurement: global analysis of productivity and quality**. New York: Mcgraw-hill (Vol. 3), 2008.

JURISTO, N.; MORENO, A.M.; **Basics of software engineering experimentation**. Springer Publishing Company, Incorporated, 2010.

KANG S.; CHOI O.; BAIK J.; **Model-Based Dynamic Cost Estimation and Tracking Method for Agile Software Development**, IEEE/ACIS 9th International Conference on Computer and Information Science, pp. 743-748, 2010.

KAN S.; **Metrics and models in software quality engineering**. Addison-Wesley Longman Publishing Co., Inc., 2002.

KITCHENHAM B.; **Guidelines for performing Systematic Literature Reviews in Software Engineering**. Software Engineering Group, School of Computer Sciences and

Mathematics, Keele University, and Department of Computer Science, University of Durham, 2007.

KITCHENHAM B.; **What's up with software metrics?—A preliminary mapping study**. Journal of systems and software, 83(1), 37-51, 2010.

KITCHENHAM B.; PFLEEGER S.; FENTON, N.; **Towards a Framework for Software Measurement Validation**. IEEE Transactions on Software Engineering. 21 (12), pp. 929- 943, 1995.

LAKATOS E. M. de A.; MARCONI M. de A.; Fundamentos da metodologia científica. São Paulo: Atlas, 2003.

LANDIS J. R.; KOCH G. G.; **The measurement of observer agreement for categorical data**. *Biometrics*, v. 33, n. 1, p. 159-174, 1977.

LEFFINGWELL D.; BEHRENS P.; **A User Story Primer**. Leffingwell, LLC, 2009.

LEI 8666; Disponível em http://www.planalto.gov.br/ccivil_03/leis/18666cons.htm Acessado em 26 de fevereiro de 2014, 1993.

LEISHMAN C; **The 3 Ps of Estimation**, Disponível em <http://chrisleishman.com/2008/07/3-ps-of-estimation.html>. Acessado em 25 de janeiro de 2015, 2008.

LIBARDI P.; BARBOSA V.; **Métodos Ágeis**. Limeira 2010.

MACORATTI J. C.; **Análise de Pontos por Função – O Processo de contagem, 2005**. Disponível em: http://www.macoratti.net/apf_pcta.htm. Acessado em 10 de julho de 2013, 2005.

MARÇAL A. S. C et al; **Estendendo o SCRUM segundo as Áreas de Processos de Gerenciamento de Projetos do CMMI**. CLEI Electronic Journal, 2007.

MARCONI, M. A.; LAKATOS, E. M.; **Metodologia Científica**. 6a. ed. São Paulo: Atlas, 2008.

MELO C. et al; **Métodos ágeis no Brasil: estado da prática em times e organizações**. Relatório Técnico RT-MAC-2012-03. Departamento de Ciência da Computação. IME-USP, 2012.

VERSIONONE; 6th Annual State of Agile Development Survey Results. The State of Agile Development Survey Results, 2011. Disponível em: http://www.versionone.com/state_of_agile_development_survey/11/. Acesso em: 07 janeiro de 2014, 2011.

MENAND L.; **Pragmatism: A Reader**. New York: Vintage, 1997.

MENS T.; DEMEYER S.; **Future Trends in Software Evolution Metrics**. IWPSE '01 Proceedings of the 4th International Workshop on Principles of Software Evolution Pages 83-86, 2001;

MINAYO, M. C. S.; O Desafio do Conhecimento. Pesquisa qualitativa em saúde. São

Paulo: HUCITEC, 2007.

MONTEIRO T. C.; **Pontos de Caso de Uso Técnicos (TUCP): Uma Extensão da UCP**. Dissertação de Mestrado, Universidade de Fortaleza (UNIFOR), Mestrado em Informática Aplicada, Fortaleza, 2005.

MOLØKKEN K.; JØRGENSEN M; **A review of software surveys on software effort estimation**, International Symposium on Empirical Software Engineering, Rome, Italy, 2003.

NOBLIT G. W.; HARE, R. D.; **Meta-Ethnography: Synthesizing Qualitative Studies**. Newbury Park, CA: Sage Publications, 1988.

OEST C.; **Quando a Análise de Pontos de Função se Torna um Método Ágil?** In: 2ª Conferência Brasileira de Medição e Análise de Software, São Paulo, Brasil 2011.

PERKINS T.; PETERSON R.; SMITH L.; **Back to the Basics: Measurement and Metrics**. USA: Department of Defense and USA Army, 2003. Disponível em: <www.stsc.af.mil/crosstalk> Acessado em 26 de maio de 2014, 2003.

PERRY W. E.; **The best measures for measuring data processing quality and productivity**. Quality Assurance Institute Technical Report, 1986.

PETTICREW M.; ROBERTS H.; H. **Systematic Reviews in the Social Sciences: a practical guide**. Oxford: Blackwell Publishing, 2006.

PMBOK; 4th edition, ISBN-10: 1933890517 & ISBN-13: 978-1933890517. Project Management Institute, New Town Square, Pennsylvania 2008.

PRESSMAN R. S.; **Engenharia de Software** (5 ed.). Rio de Janeiro: McGraw-Hill, 2002.

RANDOLPH J. J.; **A Guide to Writing the Dissertation Literature Review**, Walden University, 2009.

REVIEWER; Disponível em: <https://sites.google.com/site/eseportal/tools/reviewer>. Acessado em 18 de dezembro de 2013.

SAVOLAINEN J.; KUUSELA J.; **Volatility analysis framework for product lines, Proceedings of the 2001 Symposium on Software Reusability: Putting Software Reuse in Context**, Toronto, Ontario, Canada, 2001.

SANTANA C.; GUSMÃO C.; **Uso de Análise de Pontos de Funções em Ambientes Ágeis**. In: Engenharia de Software Magazine, p. 33 – 40, 20 dez. 2009.

SANTANA C. et al; **Using Function Points in Agile Projects**. In: International Conference on Agile Software Development, 2011.

SBQS; **XIII Simpósio Brasileiro de Qualidade de Software**, Blumenau, Santa Catarina, Brasil 2014.

SCHWABER K.; **SCRUM Development Process**. OOPSLA'95 Workshop on Business Object Design and Implementation. Austin, Texas, USA: Springer-Verlag, 1995.

SCHWABER K.; **Agile Project Management with Scrum**. [S.l.]: Microsoft Press, 2004.

SCHMIETENDORF A.; KUNZ M.; DUMKE R.; **Effort estimation for Agile Software Development Projects**. In: 5 th Software Measurement European Forum. 2008.

SERENA; **An Introduction to Agile Software Development**. Disponível em: <<http://www.serena.com/>>. Acesso em: 18 abril 2014, 2007.

SHEPPERD M.; SCHOFIELD C.; **Estimating software project effort using analogies**, IEEE Transactions on Software Engineering (23:11) 1997, pp 736-743, 1997.

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

SILVA I. J. de M.; **Delphi 5 – Análise de Pontos por Função**. Rio de Janeiro, Book Express, 2000.

SISP; **Roteiro de Métricas de Software do SISP**: Versão 2.0. Ministério do Planejamento, Orçamento e Gestão: Secretaria de Logística e Tecnologia da Informação, Brasília, 2012.

SJOBERG D. I. K. et al; **A survey of controlled experiments in software engineering**. **Software Engineering**, IEEE Transactions on, 31(9), 733-753, 2005.

SOARES M. dos S.; **Comparação entre Metodologias Ágeis e Tradicionais para o Desenvolvimento de Software**. Unipac-Universidade Presidente Antônio Carlos, Faculdade de Tecnologia e Ciências de Conselheiro Lafaiete, 2010.

SOMMERVILLE, I.; **Engenharia de Software**. Editora Addison-Wesley, 2011.

SOUSA K. D. DE; OLIVEIRA K. M. DE; ANQUETIL N.; **Uso do GQM para avaliar implantação de processo de manutenção de software**. Universidade Católica de Brasília, 2005.

STRAUSS A.; CORBIN J.; **Pesquisa qualitativa: técnicas e procedimentos para o desenvolvimento de teoria fundamentada**. Tradução de Luciane de Oliveira da Rocha. 2a. ed. Porto Alegre: Artmed, 2008.

TAKEUCHI H.; NONAKA, I.; **The New Product Development Game**. Harvard Business Review, v. 64(1), 1986.

TELES V. M.; **Um Estudo de Caso da Adoção das Práticas e Valores do Extreme Programming**. Universidade Federal do Rio de Janeiro. Rio de Janeiro, RJ, Brasil,

2005.

TRAVASSOS G. H.; GUROV D.; DO AMARAL E. A. G.; **Introdução à Engenharia de Software Experimental**, 2002.

TUKEY J. W.; **Exploratory Data Analysis**. Addison-Wesley, 1977.

UFPR; Disponível em <http://www.est.ufpr.br/ce003/material/apostilace003.pdf>. Acessado em 02 de setembro de 2014, 2011.

USMAN M. et al; **Effort estimation in Agile Software Development: A systematic literature review**. In Proceedings of the 10th International Conference on Predictive Models in Software Engineering (pp. 82-91). ACM, 2014.

VAZQUEZ, C. E.; SIMÕES G. S.; ALBERT R. M.; **Análise de Pontos de Função**. 10a edição, São Paulo, Érica, 2010.

WANDERLEY E.; VASCONCELOS A. M. L.; ÁVILA B.; **Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis**. In XIII Simpósio Brasileiro de Qualidade de Software. Blumenau,SC,Brasil, 2014.

WELCH G.; BISHOP G.; **An Introduction to the Kalman Filter**, SIGGRAPH, 2001.

APÊNDICE A – Estudos incluídos no Mapeamento Sistemático

[P01] Alshayeb, M. et al.: An empirical validation of object-oriented metrics in two different iterative software processes. *IEEE Transactions on Software Eng.*, Vol. 29, Issue 11, 2003, pp. 1043 – 1049.

[P02] S. Kuppuswami, K. Vivekanandan, P. Rodrigues, “A System Dynamics Simulation Model to Find the Effects of XP on Cost of Change Curve”, In the *Lecture Notes in Computer Science*, Vol. 2675/2003, 2003.

[P03] S. Roock, H. Wolf, “Agile Project Controlling”, In the *Lecture Notes in Computer Science*, Vol. 3092/2004, 2004, pp. 202-209.

[P04] A. Cabri, M. Griffiths, “Earned Value and Agile Reporting”, *AGILE'06*, 2006, pp. 17-22.

[P05] T. Sulaiman, B. Barton, and T. Blackburn, “AgileEVM – Earned Value Management in Scrum Projects”, *Proceedings of AGILE 2006 Conference*, 2006, pp. 7-16.

[P06] R. Madachy, B. Boehm, J. Lane, “Spiral lifecycle increment modeling for new hybrid processes”, *Proceedings of the International Software Process Workshop and International Workshop on Software Process Simulation and Modeling (SPW/ProSim 2006)*, China, LNCS, vol. 3966, Springer-Verlag, 2006, pp. 167-177.

[P07] P. Abrahamsson, R. Moser, W. Pedrycz, A. Sillitti, G. Succi, “Effort Prediction in Iterative Software Development Processes – Incremental Versus Global Prediction Models”, In *Proceedings of the First International Symposium on Empirical Software Engineering and Measurement (ESEM '07)*. IEEE Computer Society, Washington, DC, USA, 2007, pp. 344-353.

[P08] R. Lavanya, S. Chandrasekaran, V. Kanchana, ‘Multi-criteria approach for agile software cost estimation model’, In: *Proceedings of the International Conference on Global Manufacturing and Innovation in Engineering, GMICIT*, 2006.

[P09] P. Hearty, N. Fenton, D. Marquez, M. Neil, “Predicting Project Velocity in XP Using a Learning Dynamic Bayesian Network Model”, *IEEE Transactions on Software Eng.*, Vol. 35, Issue 1, pp. 124 – 137, Jan.-Feb. 2009.

[P10] S. Bhalerao, M. Ingle, “Agile Estimation using CAEA: A Comparative Study of Agile Projects”, *IPCSIT Vol.2*, IACSIT Press, Singapore, 2011.

[P11] E. Miranda, P. Bourque , A. Abran, “Sizing User Stories Using Paired Comparisons”, *Information and Software Technology*, Vol. 51, Issue 9, Sept. 2009, pp. 1327-1337.

[P12] S. Kang, O. Choi, J. Baik, “Model-Based Dynamic Cost Estimation and Tracking Method for Agile Software Development”, IEEE/ACIS 9th International Conference on Computer and Information Science, pp. 743-748, 2010.

[P13] I. Hussain, L. Kosseim, O. Ormandjieva, “Towards Approximating COSMIC Functional Size from User Requirements in Agile Development Processes Using Text Mining”, In the Lecture Notes in Computer Science, Vol. 6177/2010, 2010, pp. 80-91.

[P14] E. Miranda, P. Bourque, “Agile monitoring using the line of balance”, Journal of Systems and Software, Vol. 83, Issue 7, 2010, pp. 1205-1215.

[P15] M. Abouelela, L. Benedicenti, “Bayesian Network Based XP Process Modelling”, International Journal of Software Engineering & Applications (IJSEA), Vol.1, No.3, July 2010.

[P16] L. Cao, B. Ramesh, T. Abdel-Hamid, “Modeling dynamics in agile software development”, ACM Transactions on Management, Information Systems, Vol. 1, No. 1, Article 5, Dec. 2010.

[P17] G.B. Alleman, M. Henderson, R. Seggelke, “Making agile development work in a government contracting environment-measuring velocity with earned value”, Proceedings of the Agile Development Conference (ADC’03), 2003, pp. 114 – 119.

[P18] Y. Yong, B. Zhou, “Extreme Programming Effect through System Dynamics Modeling”, International Conference on Computational Intelligence and Software Engineering, 2009, pp. 1 – 4.

[P19] L. Cao, “Estimating Agile Software Project Effort: An Empirical Study, y. In Proceedings of Americas Conference on Information Systems (AMCIS), 2008.

[P20] E. Berardi, L. Santillo, “COSMIC-based Project Management in Agile Software Development and Mapping onto related CMMI-DEV Process Areas”, International Workshop on Software Measurement – IWSM 2010, Stuttgart, Germany, Nov. 2010.

[P21] N.C. Haugen, “An empirical study of using planning poker for user story estimation”, In: Agile Conference, 2006.

[P22] K. Petersen, C. Wohlin, “Measuring the flow in lean software development”, Softw: Pract. Exper., 41: 975–996. doi: 10.1002/spe.975, 2011.

[P23] M. J. Rees, ”A Feasible User Story Tool for Agile Software Development?”, Software Engineering Conference, 2002. Ninth Asia-Pacific, pp. 22 – 30.

[P24] H. Sharp, H. Robinson, “Collaboration and co-ordination in mature eXtreme programming teams”, International Journal of Human-Computer Studies, v. 66, n. 7, p. 506-518, 2008

[P25] I. Hussain, L. Kosseim, O. Ormandjieva, “Approximation of COSMIC functional size to support early effort estimation in Agile”, Data & Knowledge Engineering, v. 85, p. 2-14, 2013.

- [P26] R. Popli, N. Chauhan, “A Sprint-Point Based Estimation Technique In Scrum”, In: Information Systems and Computer Networks (ISCON), 2013 International Conference on. IEEE, 2013. p. 98-103.
- [P27] R. Turner, D. Ingold, J. A. Lane, R. Madachy, D. Anderson “Effectiveness of kanban approaches in systems engineering within rapid response environments”, In: Procedia Computer Science, v. 8, p. 309-314, 2012.
- [P28] Zhong, S., Liping, C., Tian-en, C. Agile planning and development methods. In Computer Research and Development (ICCRD), 2011 3rd International Conference on (Vol. 1, pp. 488-491). IEEE.
- [P29] C. Santana, C. Gusmão, “Uso de Análise de Pontos de Funções em Ambientes Ágeis”, In Engenharia de Software Magazine, p. 33 – 40, 20 dez. 2009.
- [P30] A. M. Fuqua, “Using Function Points in XP Considerations”, In: Extreme Programming and Agile Processes in Software Engineering. Springer Berlin Heidelberg, 2003. p. 340-342.
- [P31] C. A. Santana, F. S. Leoneo, C. M. G. Gusmão, A. M. L. Vasconcelos, Using Function Points in Agile Projects, In: Agile Processes in Software Engineering and Extreme Programming. Springer Berlin Heidelberg, 2011. p. 176-191.
- [P32] A. U. Banerjee, B. K. Narayanan, and C. Mahadevan P, “Estimating Agile Iterations by Extending Function Point Analysis”, In: WORLDCOMP’12
- [P33] M. R. R. Braga, C. I. M. Bezerra, J. M. Monteiro, R. Andrade, “Uma Linguagem de Padrões para Estimativa de Software em Métodos Ágeis”, In: MiniPLoP Brasil 2011
- [P34] M. Kunz, R. R. Dumke, N. Zenker, “Software Metrics for Agile Software Development”, In: In: Software Engineering, 2008. ASWEC 2008. 19th Australian Conference on. IEEE, 2008. p. 673-678.
- [P35] D. Horvat, “Function Point Analysis and Agile Methodology”, Disponível em: <http://www.qpmg.com/>
- [P36] V. Mahnic, I. Vrana, “Using stakeholder-driven process performance measurement for monitoring the performance of a Scrum-based software development process”, In: Elektrotehniski vestnik, v. 74, n. 5, p. 241-247, 2007.
- [P37] L. Buglione, A. Abran, “Improving Estimations in Agile Projects: Issues and Avenues”, In: Proceedings of the 4th Software Measurement European Forum (SMEF 2007), Rome (Italy). 2007. p. 265-274.
- [P38] T. Cagley, “Agile Estimation Using Functional Metrics” The IFPUG Guide to IT and Software Measurement IFPUG CRC Press
- [P39] A. J. Alexander, “Case Study: Function Point Analysis and Cost Estimation in An Agile Development Environment” Disponível em <http://www.devdaily.com/fpa/fpa->

louisville-agile/

[P40] A. M. Fuqua: "Using Function Points in Extreme Programming". Disponível em [http://groups.yahoo.com/group/extremeprogramming/files/XP and FPA.pdf](http://groups.yahoo.com/group/extremeprogramming/files/XP%20and%20FPA.pdf)

APÊNDICE B – Formulário do Mapeamento Sistemático

D	E	F	G	H	I	J	K
Artigo	Ano	Local	Métodos de Desenvolvimento	Métricas de Tamanho	Uso de Dados Históricos	Validação	Contexto
M. J. Rees, "A Feasible User Story Tool for Agile Software Development?"	2002	IEE	XP	Dias Ideais	Sem Dados	Sem Experimentação	Sem dados experimentais
H. Sharp, H. Robinson, "Collaboration and co-ordination in mature eXtreme programming teams"	2007	Science Direct	XP	Story Points	Uso do Histórico das empresas	Estudo de Caso	Industria
I. Hussain, L. Kosseim, O. Ormandjieva, "Approximation of COSMIC functional size to support early effort estimation in Agile"	2013	Science Direct	Agile Geral	COSMIC-FFP	Uso do Histórico das empresas	Experimento Controlado	Industria
R. Popli, N. Chauhan, "A Sprint-Point Based Estimation Technique In Scrum"	2013	IEE	Scrum	Sprint Points	Dados fictícios	Experimento Controlado	Acadêmico
R. Turnera, D. Ingoldeb, J. A. Laneb, R. Madachyc, D. Anderson, "Effectiveness of kanban approaches in systems engineering within rapid response environments"	2012	Science Direct	Agile Geral	Dias Ideais	Dados fictícios	Sem Experimentação	Acadêmico
S. Z. C. Liping, C. Tian-e, "Agile Planning and Development Methods"	2011	IEE	XP, Scrum e Crystal		Sem Dados	Sem Experimentação	Sem dados experimentais
C. Santana, C. Gusmão, "Uso de Análise de Pontos de Funções em				Story Points e Pontos de		Sem	Sem dados

Gráficos **Artigos Revisão**

NTO

100%

APÊNDICE C – Formulário Perfil Pesquisador

[Edit this form](#)

Perfil Pesquisador

*** Required**

Qual é o seu nível de experiência com desenvolvimento de software? *

- Até 1 ano
- Entre 1 e 5 anos
- Entre 5 e 10 anos
- Mais de 10 anos

Qual é o seu nível de experiência com estimativas de software? *

- Nenhuma
- Até 1 ano
- Entre 1 e 5 anos
- Mais de 5 anos

Qual é o seu nível de experiência com métodos ágeis? *

- Nenhuma
- Até 1 ano
- Entre 1 e 5 anos
- Mais de 5 anos

Qual é o seu nível de experiência com o uso de story points? *

- Nenhuma
- Até 1 ano
- Entre 1 e 5 anos
- Mais de 5 anos

Qual é o seu nível de experiência com pontos de função? *

- Nenhuma
- Até 1 ano
- Entre 1 e 5 anos
- Mais de 5 anos

Never submit passwords through Google Forms.

Powered by  Google Drive

This content is neither created nor endorsed by Google.
[Report Abuse](#) - [Terms of Service](#) - [Additional Terms](#)

APÊNDICE D – Formulários de Coletas Experimento

Engenharia de Software Experimental		 	
Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis			
Uma Avaliação de Três Abordagens			
Formulário de Coleta de Dados da Abordagem: Estimating Agile Iterations by Extending Function Point Analysis			
Dados de Execução:	[DE-1]	Projeto Selecionado:	
	[DE-2]	Data de Realização:	
Dados de Leitura das estórias:	[C1]	Horário de Início:	[C2] Horário de Fim:
	[C3]	Observações:	
Dados de Calculo do Esforço:	[C4]	Horário de Início:	[C5] Horário de Fim:
Dados de Quantitativo de Resultados:	[C6]	ALIs encontrados:	0
	[C7]	AIes encontrados:	0
	[C8]	EES encontrados:	
	[C9]	SEs encontrados:	
	[C10]	CEs encontrados:	
	[C11]	Pontos de Função Calculados:	
	[C12]	Observações:	
A_U_Banerjee T_Cagley A_J_Alexander Comparação +			
Engenharia de Software Experimental		 	
Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis			
Uma Avaliação de Três Abordagens			
Formulário de Coleta de Dados da Abordagem: Agile Estimation Using Functional Metrics			
Dados de Execução:	[DE-1]	Projeto Selecionado:	
	[DE-2]	Data de Realização:	
Dados de Leitura das estórias:	[C1]	Horário de Início:	[C2] Horário de Fim:
	[C3]	Observações:	
Dados de Calculo do Esforço:	[C4]	Horário de Início:	[C5] Horário de Fim:
Dados de Quantitativo de Resultados:	[C6]	ALIs encontrados:	0
	[C7]	AIes encontrados:	0
	[C8]	EES encontrados:	
	[C9]	SEs encontrados:	
	[C10]	CEs encontrados:	
	[C11]	Pontos de Função Calculados:	
	[C12]	Observações:	
A_U_Banerjee T_Cagley A_J_Alexander Comparação +			
Engenharia de Software Experimental		 	
Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis			
Uma Avaliação de Três Abordagens			
Formulário de Coleta de Dados da Abordagem: Case Study: Function Point Analysis and Cost Estimation in An Agile Development Environment			
Dados de Execução:	[DE-1]	Projeto Selecionado:	
	[DE-2]	Data de Realização:	
Dados de Leitura das estórias:	[C1]	Horário de Início:	[C2] Horário de Fim:
	[C3]	Observações:	
Dados de Calculo do Esforço:	[C4]	Horário de Início:	[C5] Horário de Fim:
Dados de Quantitativo de Resultados:	[C6]	ALIs encontrados:	0
	[C7]	AIes encontrados:	0
	[C8]	EES encontrados:	
	[C9]	SEs encontrados:	
	[C10]	CEs encontrados:	
	[C11]	Pontos de Função Calculados:	
	[C12]	Observações:	
A_U_Banerjee T_Cagley A_J_Alexander Comparação +			

APÊNDICE E – Roteiros de Execução



Script para execução do experimento com a abordagem proposta por A. U. Banerjee – Estimating Agile Iterations by Extending Function Point Analysis

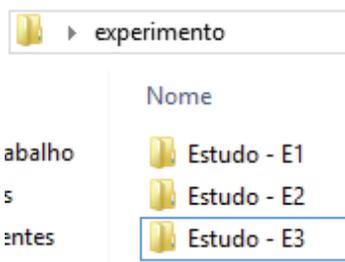
Passo P1 – Identificação do diretório com os artefatos a serem utilizados no experimento.

P1.1 – Abra o diretório “experimento” que contém todo o material necessário a ser utilizado na execução do experimento.

P1.2 – Mantenha o diretório aberto.

Passo P2 – Identificação dos artefatos do estudo sistemático a serem utilizados

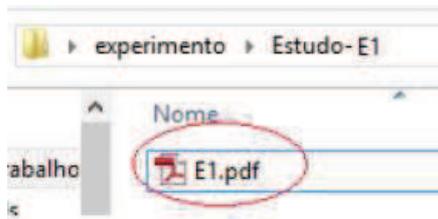
P2.1 – Observe que na raiz do diretório “experimento”, para cada um dos estudos a serem utilizados, existe um diretório correspondente. Os dados do estudo “E1” encontram-se na pasta “Estudo-E1”, os dados do estudo “E2” estão na pasta “Estudo-E2” e os dados do estudo “E3” estão na pasta “Estudo-E3”.



P2.2 – Acesse a pasta correspondente ao estudo que você foi designado a executar com a abordagem proposta por A. U. Banerjee.

P2.3 – Note que existe um arquivo correspondente ao nome do estudo no formato pdf. Exemplo: “E1.pdf”. Este arquivo contém informações do estudo referentes às histórias a serem

medidas.



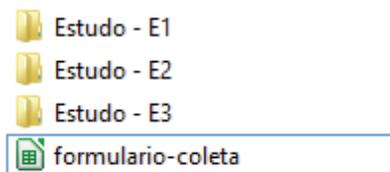
P2.4 – Abra o arquivo “E1.pdf” clicando duas vezes sobre o mesmo.

P2.5 – Mantenha o arquivo aberto para posterior consulta.

P2.6 – Retorne ao diretório “experimento”.

Passo P3 – Identificação e preenchimento dos dados de identificação no formulário de coleta

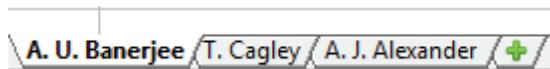
P3.1 – Note que na raiz do diretório “experimento” existe o arquivo “formulario-coleta.xls”. Este é o formulário onde todos os dados coletados durante a experimentação deverão ser registrados.



P3.2 – Clique duas vezes sobre o arquivo “formulario-coleta.xls” para abri-lo.

P3.3 – Selecione o formulário clicando sobre a planilha “experimento”.

P3.4 – Selecione o formulário de avaliação da ferramenta Start, clicando sobre a planilha “A. U. Banerjee”.



P3.5 – Será exibido o formulário de avaliação da abordagem de A. U. Banerjee.

Engenharia de Software Experimental			
Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis			
Uma Avaliação de Três Abordagens			
Formulário de Coleta de Dados da Abordagem: Estimating Agile Iterations by Extending Function Point Analysis			
Dados de Execução:	[DE-1]	Projeto Selecionado:	
	[DE-2]	Data de Realização:	
Dados de Leitura das estórias:	[C1]	Horário de Início:	[C2] Horário de Fim:
	[C3]	Observações:	
Dados de Cálculo do Esforço:	[C4]	Horário de Início:	[C5] Horário de Fim:
	[C6]	ALIs encontrados:	0
Dados de Quantitativo de Resultados:	[C7]	ALEs encontrados:	0
	[C8]	EES encontrados:	[C9] SES encontrados:
	[C10]	CEs encontrados:	[C11] Pontos de Função Calculados:
	[C12]	Observações:	

P3.6 – Proceda com o preenchimento das seguintes informações:

a) Campo DE-1 – Projeto Selecionado: Preencha com o estudo que foi designado para sua equipe;

b) Campo DE-2 – Data de Realização: Preencha com a data em que está coletando os dados;

P3.7 – Salve o arquivo “formulario-coleta.xls” e mantenha-o aberto para registrar os dados que serão coletados nos passos seguintes.

Passo P4 – Leitura, Identificação e Entendimento das estórias do projeto

P4.1 – Observe a hora atual do sistema operacional.

P4.2 – Retorne à planilha “formulario-coleta.xls” e registre a hora de início da execução no campo [C1] na seção “Dados de Leitura das estórias” da planilha “A. U. Banerjee”.

[C1]	Horário de Início:	00:00:00
------	--------------------	----------

P4.3 – Salve o arquivo “formulario-coleta.xls”.

P4.4 – Leia as estórias contidas no arquivo do estudo relacionado (No nosso exemplo o E1.pdf).

P4.5 – Para cada estórias lida verifique quais são as funções de dados e conte cada um deles.

Obs: As funções de dados representam a funcionalidade oferecida ao usuário para satisfazer requisitos de dados internos e externos à aplicação, ou seja, representam os requisitos de armazenamento de dados do usuário. Há dois tipos (classificação) de funções de dados:

Arquivos Lógicos Internos (ALI): Grupos de dados logicamente relacionados ou informações de controle, que são alterados na própria aplicação;

Arquivos de Interface Externa (AIE): Grupos de dados logicamente relacionados, cujo processo de alteração é feito em outra aplicação;

P4.6 – Para cada estórias lida verifique quais são as funções de transações e conte cada um deles.

Obs 2: Estas funções representam as funções de processamento dos dados fornecidos pela aplicação ao usuário. Há três tipos (classificação) de funções de transações:

Entrada Externa (EE): Processos elementares (transações) que processam dados ou informações de controle que entram pela fronteira da aplicação. Ex: Incluir Cliente; Alterar Conta; Excluir Pessoa;

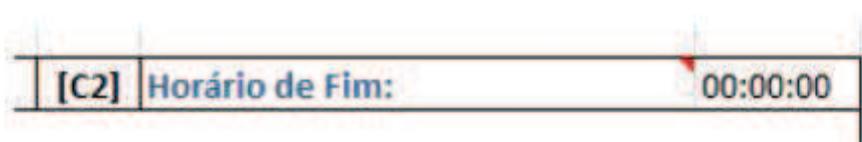
Saída Externa (SE): Processos elementares que enviam dados ou informações de controle para fora da fronteira da aplicação. Seu objetivo é apresentar informações ao usuário que foram recuperadas através de um processamento lógico e não apenas uma simples recuperação de dados. Uma SE deve envolver cálculos ou criação de dados derivados ou também, pode manter um ALI ou alterar o comportamento do sistema. Ex: Relatório do total de vendas por período, tela de Login (com criptografia).

Consulta Externa (CE): Semelhante a uma SE, é um processo elementar que envia dados (ou informações de controle) para fora da fronteira da aplicação. Todavia, seu objetivo é apresentar informação para o usuário, por meio apenas de uma recuperação das informações, sem a realização de nenhum cálculo nem a criação de dados derivados. Ex: Consulta de Clientes, Lista de Vendas.

Passo P5 – Registro do Fim da Leitura, Identificação e Entendimento das estórias do projeto

P5.1 – Observe a hora atual do sistema operacional.

P5.2 – Retorne à planilha “formulário-coleta.xls” e registre a hora de fim da execução no campo [[C2] na seção “Dados de Leitura das estórias” da planilha “A. U. Banerjee”.



[C2]	Horário de Fim:	00:00:00
------	-----------------	----------

P5.3 – Salve o arquivo “formulário-coleta.xls”.

Passo P6 – Cálculo do Esforço

P6.1 – Observe a hora atual do sistema operacional.

P6.2 – Retorne à planilha “formulário-coleta.xls” e registre a hora de início da execução no campo [C4] na seção “Dados de Cálculo do Esforço” da planilha “A. U. Banerjee”.

[C4]	Horário de Início:	00:00:00
------	--------------------	----------

P6.3 – Salve o arquivo “formulário-coleta.xls”.

P6.4 – Para cada função de dados e transações coletadas, calcule sua a complexidade.

P6.5 – Verifique o valor em pontos de função de dados e transações coletadas.

OBS3: Para ajudar o cálculo de sua complexidade e do valor em pontos de função, o arquivo PF.pdf sever como auxílio.

Passo P7 – Registro do Fim do Cálculo com a Abordagem de A. U. Banerjee

P7.1 – Observe a hora atual do sistema operacional.

P7.2 – Retorne à planilha “formulário-coleta.xls” e registre a hora de fim da execução no campo [[C5] na seção “Dados de Cálculo do Esforço:” da planilha “A. U. Banerjee”.

[C5]	Horário de Fim:	00:00:00
------	-----------------	----------

P7.3 – Registre a quantidade de ALIs, AIEs, EEs, SEs, CEs coletados nos campos [C6], [C7], [C8], [C9], [C10] respectivamente e coloque o valor de pontos de função calculado no campo [c11] na seção “Dados de Quantitativo de Resultados” da planilha “A. U. Banerjee”.

Dados de Quantitativo de Resultados:	[C6]	ALIs encontrados:	0	[C8]	EEs encontrados:		[C10]	CEs encontrados:	
	[C7]	AIEs encontrados:	0	[C9]	SEs encontrados:		[C11]	Pontos de Função Calculados:	
	[C12]	Observações:							

P7.4 – Salve o arquivo “formulário-coleta.xls”.



Script para execução do experimento com a abordagem proposta por Alvin J. Alexander – Case Study: Function Point Analysis and Cost Estimation in An Agile Development Environment

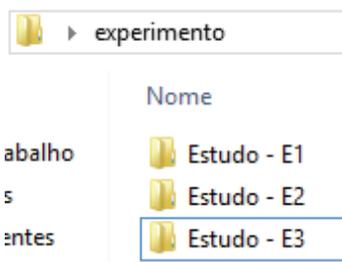
Passo P1 – Identificação do diretório com os artefatos a serem utilizados no experimento.

P1.1 – Abra o diretório “experimento” que contem todo o material necessário a ser utilizado na execução do experimento.

P1.2 – Mantenha o diretório aberto.

Passo P2 – Identificação dos artefatos do estudo sistemático a serem utilizados

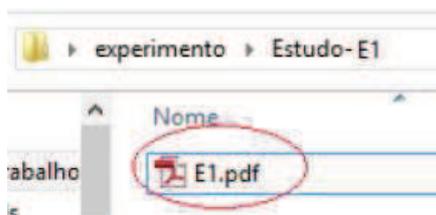
P2.1 – Observe que na raiz do diretório “experimento”, para cada um dos estudos a serem utilizados, existe um diretório correspondente. Os dados do estudo “E1” encontram-se na pasta “Estudo-E1”, os dados do estudo “E2” estão na pasta “Estudo-E2” e os dados do estudo “E3” estão na pasta “Estudo-E3”.



P2. 2 – Acesse a pasta correspondente ao estudo que você foi designado a executar com a abordagem proposta por Alvin J. Alexander.

P2.3 – Note que existe um arquivo correspondente ao nome do estudo no formato pdf.

Exemplo: “E1.pdf”. Este arquivo contém informações do estudo referentes às estórias a serem medidas.



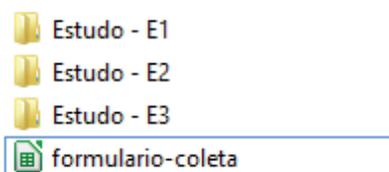
P2.4 – Abra o arquivo “E1.pdf” clicando duas vezes sobre o mesmo.

P2.5 – Mantenha o arquivo aberto para posterior consulta.

P2.6 – Retorne ao diretório “experimento”.

Passo P3 – Identificação e preenchimento dos dados de identificação no formulário de coleta

P3.1 – Note que na raiz do diretório “experimento” existe o arquivo “formulario-coleta.xls”. Este é o formulário onde todos os dados coletados durante a experimentação deverão ser registrados.



P3.2 – Clique duas vezes sobre o arquivo “formulario-coleta.xls” para abri-lo.

P3.3 – Selecione o formulário clicando sobre a planilha “experimento”.

P3.4 – Selecione o formulário de avaliação da ferramenta Start, clicando sobre a planilha “A. J. Alexander”.



P3.5 – Será exibido o formulário de avaliação da abordagem de A. J. Alexander.

B	C	D	E	F	G	H	I	J	K
Engenharia de Software Experimental									
Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis Uma Avaliação de Três Abordagens									
Formulário de Coleta de Dados da Abordagem: Case Study: Function Point Analysis and Cost Estimation in An Agile Development Environment									
Dados de Execução:		[DE-1] Projeto Selecionado:							
		[DE-2] Data de Realização:							
Dados de Leitura das estórias:		[C1] Horário de Início:	[C2] Horário de Fim:						
		[C3] Observações:							
Dados de Calculo do Esforço:		[C4] Horário de Início:	[C5] Horário de Fim:						
		[C6] AUs encontrados:	0	[C8] EEs encontrados:	[C10] CEs encontrados:				
Dados de Quantitativo de Resultados:		[C7] AIEs encontrados:	0	[C9] SEs encontrados:	[C11] Pontos de Função Calculados:				
		[C12] Observações:							

P3.6 – Proceda com o preenchimento das seguintes informações:

a) Campo DE-1 – Projeto Selecionado: Preencha com o estudo que foi designado para sua equipe;

b) Campo DE-2 – Data de Realização: Preencha com a data em que está coletando os dados;

P3.7 – Salve o arquivo “formulario-coleta.xls” e mantenha-o aberto para registrar os dados que serão coletados nos passos seguintes.

Passo P4 – Leitura, Identificação e Entendimento das estórias do projeto

P4.1 – Observe a hora atual do sistema operacional.

P4.2 – Retorne à planilha “formulario-coleta.xls” e registre a hora de início da execução no campo [C1] na seção “Dados de Leitura das estórias” da planilha “A. J. Alexander”.

[C1]	Horário de Início:	00:00:00
------	--------------------	----------

P4.3 – Salve o arquivo “formulario-coleta.xls”.

P4.4 – Leia as estórias contidas no arquivo do estudo relacionado (No nosso exemplo o E1.pdf).

P4.5 – Para cada estórias lida verifique quais são as funções de dados e conte cada ALI.

Obs: As funções de dados representam a funcionalidade oferecida ao usuário para satisfazer requisitos de dados internos e externos à aplicação, ou seja, representam os requisitos de armazenamento de dados do usuário. Há dois tipos (classificação) de funções de dados:

Arquivos Lógicos Internos (ALI): Grupos de dados logicamente relacionados ou informações de controle, que são alterados na própria aplicação;

Arquivos de Interface Externa (AIE): Grupos de dados logicamente relacionados, cujo processo de alteração é feito em outra aplicação;

Passo P5 – Registro do Fim da Leitura, Identificação e Entendimento das estórias do projeto

P5.1 – Observe a hora atual do sistema operacional.

P5.2 – Retorne à planilha “formulario-coleta.xls” e registre a hora de fim da execução no campo [[C2] na seção “Dados de Leitura das estórias” da planilha “A. J. Alexander”.

[C2]	Horário de Fim:	00:00:00
------	-----------------	----------

P5.3 – Salve o arquivo “formulario-coleta.xls”.

Passo P6 – Cálculo do Esforço

P6.1 – Observe a hora atual do sistema operacional.

P6.2 – Retorne à planilha “formulario-coleta.xls” e registre a hora de início da execução no campo [C4] na seção “Dados de Cálculo do Esforço” da planilha “A. J. Alexander”.

[C4]	Horário de Início:	00:00:00
------	--------------------	----------

P6.3 – Salve o arquivo “formulário-coleta.xls”.

P6.4 – Multiplique o número de ALIs encontrados por 35.

Passo P7 – Registro do Fim do Cálculo com a Abordagem de Alvin J. Alexander

P7.1 – Observe a hora atual do sistema operacional.

P7.2 – Retorne à planilha “formulário-coleta.xls” e registre a hora de fim da execução no campo [[C5] na seção “Dados de Cálculo do Esforço:” da planilha “A. J. Alexander”.

[C5]	Horário de Fim:	00:00:00
------	-----------------	----------

P7.3 – Registre a quantidade de ALIs coletados no campo [C6], coloque o valor N/A nos campos [C7], [C8], [C9], [C10] e coloque o valor de pontos de função calculado no campo [C11] na seção “Dados de Quantitativo de Resultados” da planilha “A. J. Alexander”.

Dados de Quantitativo de Resultados:	[C6]	ALIs encontrados:	0	[C8]	EEs encontrados:		[C10]	CEs encontrados:	
	[C7]	ALIs encontrados:	0	[C9]	SEs encontrados:		[C11]	Pontos de Função Calculados:	
	[C12]	Observações:							

P7.4 – Salve o arquivo “formulário-coleta.xls”.



Script para execução do experimento com a abordagem proposta por T. Cagley – Agile Estimation Using Functional Metrics

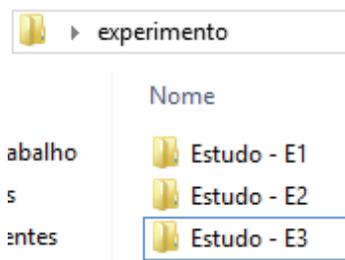
Passo P1 – Identificação do diretório com os artefatos a serem utilizados no experimento.

P1.1 – Abra o diretório “experimento” que contem todo o material necessário a ser utilizado na execução do experimento.

P1.2 – Mantenha o diretório aberto.

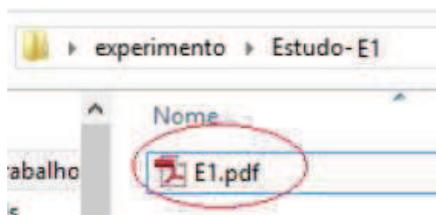
Passo P2 – Identificação dos artefatos do estudo sistemático a serem utilizados

P2.1 – Observe que na raiz do diretório “experimento”, para cada um dos estudos a serem utilizados, existe um diretório correspondente. Os dados do estudo “E1” encontram-se na pasta “Estudo-E1”, os dados do estudo “E2” estão na pasta “Estudo-E2” e os dados do estudo “E3” estão na pasta “Estudo-E3”.



P2.2 – Acesse a pasta correspondente ao estudo que você foi designado a executar com a abordagem proposta por T. Cagley.

P2.3 – Note que existe um arquivo correspondente ao nome do estudo no formato pdf. Exemplo: “E1.pdf”. Este arquivo contém informações do estudo referentes às estórias a serem medidas.



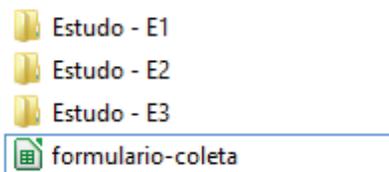
P2.4 – Abra o arquivo “E1.pdf” clicando duas vezes sobre o mesmo.

P2.5 – Mantenha o arquivo aberto para posterior consulta.

P2.6 – Retorne ao diretório “experimento”.

Passo P3 – Identificação e preenchimento dos dados de identificação no formulário de coleta

P3.1 – Note que na raiz do diretório “experimento” existe o arquivo “formulario-coleta.xls”. Este é o formulário onde todos os dados coletados durante a experimentação deverão ser registrados.



P3.2 – Clique duas vezes sobre o arquivo “formulario-coleta.xls” para abri-lo.

P3.3 – Selecione o formulário clicando sobre a planilha “experimento”.

P3.4 – Selecione o formulário de avaliação da ferramenta Start, clicando sobre a planilha “T. Cagley”.



P3.5 – Será exibido o formulário de avaliação da abordagem de T. Cagley.

Engenharia de Software Experimental			
Aplicação de Pontos por Função em Projetos que Usam Métodos Ágeis			
Uma Avaliação de Três Abordagens			
Formulário de Coleta de Dados da Abordagem: Agile Estimation Using Functional Metrics			
Dados de Execução:	[DE-1] Projeto Selecionado:		
	[DE-2] Data de Realização:		
Dados de Leitura das estórias:	[C1] Horário de Início:	[C2] Horário de Fim:	
	[C3] Observações:		
Dados de Calculo do Esforço:	[C4] Horário de Início:	[C5] Horário de Fim:	
Dados de Quantitativo de Resultados:	[C6] AIs encontrados: 0	[C8] EEs encontrados:	[C10] CEs encontrados:
	[C7] AIEs encontrados: 0	[C9] SEs encontrados:	[C11] Pontos de Função Calculados:
	[C12] Observações:		

P3.6 – Proceda com o preenchimento das seguintes informações:

a) Campo DE-1 – Projeto Selecionado: Preencha com o estudo que foi designado para

sua equipe;

b) Campo DE-2 – Data de Realização: Preencha com a data em que está coletando os dados;

P3.7 – Salve o arquivo “formulario-coleta.xls” e mantenha-o aberto para registrar os dados que serão coletados nos passos seguintes.

Passo P4 – Leitura, Identificação e Entendimento das estórias do projeto

P4.1 – Observe a hora atual do sistema operacional.

P4.2 – Retorne à planilha “formulario-coleta.xls” e registre a hora de início da execução no campo [C1] na seção “Dados de Leitura das estórias” da planilha “T. Cagley”.



P4.3 – Salve o arquivo “formulario-coleta.xls”.

P4.4 – Leia as estórias contidas no arquivo do estudo relacionado (No nosso exemplo o E1.pdf).

P4.5 – Para cada estórias lida verifique quais são as funções de dados e conte cada um deles.

Obs: As funções de dados representam a funcionalidade oferecida ao usuário para satisfazer requisitos de dados internos e externos à aplicação, ou seja, representam os requisitos de armazenamento de dados do usuário. Há dois tipos (classificação) de funções de dados:

Arquivos Lógicos Internos (ALI): Grupos de dados logicamente relacionados ou informações de controle, que são alterados na própria aplicação;

Arquivos de Interface Externa (AIE): Grupos de dados logicamente relacionados, cujo processo de alteração é feito em outra aplicação;

P4.6 – Para cada estórias lida verifique quais são as funções de transações e conte cada um deles.

Obs 2: Estas funções representam as funções de processamento dos dados fornecidos pela aplicação ao usuário. Há três tipos (classificação) de funções de transações:

Entrada Externa (EE): Processos elementares (transações) que processam dados ou informações de controle que entram pela fronteira da aplicação. Ex: Incluir Cliente; Alterar Conta; Excluir Pessoa;

Saída Externa (SE): Processos elementares que enviam dados ou informações de controle para fora da fronteira da aplicação. Seu objetivo é apresentar informações ao usuário que foram recuperadas através de um processamento lógico e não apenas uma simples recuperação de dados. Uma SE deve envolver cálculos ou criação de dados derivados ou também, pode manter um ALI ou alterar o comportamento do sistema. Ex: Relatório do total de vendas por período, tela de Login (com criptografia).

Consulta Externa (CE): Semelhante a uma SE, é um processo elementar que envia dados (ou informações de controle) para fora da fronteira da aplicação. Todavia, seu objetivo é apresentar informação para o usuário, por meio apenas de uma recuperação das informações, sem a realização de nenhum cálculo nem a criação de dados derivados. Ex: Consulta de Clientes, Lista de Vendas.

Passo P5 – Registro do Fim da Leitura, Identificação e Entendimento das estórias do projeto

P5.1 – Observe a hora atual do sistema operacional.

P5.2 – Retorne à planilha “formulário-coleta.xls” e registre a hora de fim da execução no campo [[C2] na seção “Dados de Leitura das estórias” da planilha “T. Cagley”.

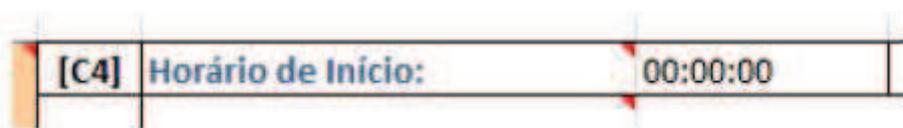


P5.3 – Salve o arquivo “formulário-coleta.xls”.

Passo P6 – Cálculo do Esforço

P6.1 – Observe a hora atual do sistema operacional.

P6.2 – Retorne à planilha “formulário-coleta.xls” e registre a hora de início da execução no campo [C4] na seção “Dados de Cálculo do Esforço” da planilha “T. Cagley”.



P6.3 – Salve o arquivo “formulário-coleta.xls”.

P6.4 – Para cada função de dados e transações coletadas, calcule sua a complexidade.

P6.5 – Verifique o valor em pontos de função de dados e transações coletadas.

OBS3: Para ajudar o cálculo de sua complexidade e do valor em pontos de função, o arquivo PF.pdf sever como auxílio.

Passo P7 – Registro do Fim do Cálculo com a Abordagem de T. Cagley

P7.1 – Observe a hora atual do sistema operacional.

P7.2 – Retorne à planilha “formulário-coleta.xls” e registre a hora de fim da execução no campo [[C5] na seção “Dados de Cálculo do Esforço:” da planilha “T. Cagley”.



P7.3 – Registre a quantidade de ALIs, AIEs, EEs, SEs, CEs coletados nos campos [C6], [C7], [C8], [C9], [C10] respectivamente e coloque o valor de pontos de função calculado no campo [c11] na seção “Dados de Quantitativo de Resultados” da planilha “T. Cagley”.

Dados de Quantitativo de Resultados:	[C6]	AIJs encontrados:	0	[C8]	EJs encontrados:		[C10]	CEs encontrados:	
	[C7]	AIEs encontrados:	0	[C9]	SEs encontrados:		[C11]	Pontos de Função Calculados:	
	[C12]	Observações:							

P7.4 – Salve o arquivo “formulário-coleta.xls”.

APÊNDICE F – Limites Unilaterais da Distribuição F de Fisher-Snedecor ao Nível de 5% de Probabilidade.

GL V2	V1																			
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	20	40	60	120	240
1	161.4	199.5	215.7	224.6	230.2	234.0	236.8	238.9	240.5	241.9	243.0	243.9	244.7	245.4	245.9	248.0	251.1	252.2	253.3	253.8
2	18.513	19.000	19.164	19.247	19.296	19.329	19.353	19.371	19.385	19.396	19.405	19.412	19.419	19.424	19.429	19.446	19.471	19.479	19.487	19.492
3	10.128	9.552	9.277	9.117	9.013	8.941	8.887	8.845	8.812	8.785	8.763	8.745	8.729	8.715	8.703	8.660	8.594	8.572	8.549	8.538
4	7.709	6.944	6.591	6.388	6.256	6.163	6.094	6.041	5.999	5.964	5.936	5.912	5.891	5.873	5.858	5.803	5.717	5.688	5.658	5.643
5	6.608	5.786	5.409	5.192	5.050	4.950	4.876	4.818	4.772	4.735	4.704	4.678	4.655	4.636	4.619	4.558	4.464	4.431	4.398	4.382
6	5.987	5.143	4.757	4.534	4.387	4.284	4.207	4.147	4.099	4.060	4.027	4.000	3.976	3.956	3.938	3.874	3.774	3.740	3.705	3.687
7	5.591	4.737	4.347	4.120	3.972	3.866	3.787	3.726	3.677	3.637	3.603	3.575	3.550	3.529	3.511	3.445	3.340	3.304	3.267	3.249
8	5.318	4.459	4.066	3.838	3.688	3.581	3.500	3.438	3.388	3.347	3.313	3.284	3.259	3.237	3.218	3.150	3.043	3.005	2.967	2.947
9	5.117	4.256	3.863	3.633	3.482	3.374	3.293	3.230	3.179	3.137	3.102	3.073	3.048	3.025	3.006	2.936	2.826	2.787	2.748	2.727
10	4.965	4.103	3.708	3.478	3.326	3.217	3.135	3.072	3.020	2.978	2.943	2.913	2.887	2.865	2.845	2.774	2.661	2.621	2.580	2.559
11	4.844	3.982	3.587	3.357	3.204	3.095	3.012	2.948	2.896	2.854	2.818	2.788	2.761	2.739	2.719	2.646	2.531	2.490	2.448	2.426
12	4.747	3.885	3.490	3.259	3.106	2.996	2.913	2.849	2.796	2.753	2.717	2.687	2.660	2.637	2.617	2.544	2.426	2.384	2.341	2.319
13	4.667	3.806	3.411	3.179	3.025	2.915	2.832	2.767	2.714	2.671	2.635	2.604	2.577	2.554	2.533	2.459	2.339	2.297	2.252	2.230
14	4.600	3.739	3.344	3.112	2.958	2.848	2.764	2.699	2.646	2.602	2.565	2.534	2.507	2.484	2.463	2.388	2.266	2.223	2.178	2.155
15	4.543	3.682	3.287	3.056	2.901	2.790	2.707	2.641	2.588	2.544	2.507	2.475	2.448	2.424	2.403	2.328	2.204	2.160	2.114	2.090
16	4.494	3.634	3.239	3.007	2.852	2.741	2.657	2.591	2.538	2.494	2.456	2.425	2.397	2.373	2.352	2.276	2.151	2.106	2.059	2.035
17	4.451	3.592	3.197	2.965	2.810	2.699	2.614	2.548	2.494	2.450	2.413	2.381	2.353	2.329	2.308	2.230	2.104	2.058	2.011	1.986
18	4.414	3.555	3.160	2.928	2.773	2.661	2.577	2.510	2.456	2.412	2.374	2.342	2.314	2.290	2.269	2.191	2.063	2.017	1.968	1.943
19	4.381	3.522	3.127	2.895	2.740	2.628	2.544	2.477	2.423	2.378	2.340	2.308	2.280	2.256	2.234	2.155	2.026	1.980	1.930	1.905
20	4.351	3.493	3.098	2.866	2.711	2.599	2.514	2.447	2.393	2.348	2.310	2.278	2.250	2.225	2.203	2.124	1.994	1.946	1.896	1.870
21	4.325	3.467	3.072	2.840	2.685	2.573	2.488	2.420	2.366	2.321	2.283	2.250	2.222	2.197	2.176	2.096	1.965	1.916	1.866	1.839
22	4.301	3.443	3.049	2.817	2.661	2.549	2.464	2.397	2.342	2.297	2.259	2.226	2.198	2.173	2.151	2.071	1.938	1.889	1.838	1.811
23	4.279	3.422	3.028	2.796	2.640	2.528	2.442	2.375	2.320	2.275	2.236	2.204	2.175	2.150	2.128	2.048	1.914	1.865	1.813	1.785
24	4.260	3.403	3.009	2.776	2.620	2.508	2.422	2.355	2.300	2.255	2.216	2.183	2.155	2.130	2.108	2.027	1.892	1.842	1.790	1.762
25	4.242	3.385	2.991	2.759	2.603	2.490	2.405	2.337	2.282	2.236	2.198	2.165	2.136	2.111	2.089	2.007	1.872	1.822	1.768	1.740
26	4.225	3.369	2.975	2.743	2.587	2.474	2.388	2.321	2.265	2.220	2.181	2.148	2.119	2.094	2.072	1.990	1.853	1.803	1.749	1.720
27	4.210	3.354	2.960	2.728	2.572	2.459	2.373	2.305	2.250	2.204	2.166	2.132	2.103	2.078	2.056	1.974	1.836	1.785	1.731	1.702
28	4.196	3.340	2.947	2.714	2.558	2.445	2.359	2.291	2.236	2.190	2.151	2.118	2.089	2.064	2.041	1.959	1.820	1.769	1.714	1.685
29	4.183	3.328	2.934	2.701	2.545	2.432	2.346	2.278	2.223	2.177	2.138	2.104	2.075	2.050	2.027	1.945	1.806	1.754	1.698	1.669
30	4.171	3.316	2.922	2.690	2.534	2.421	2.334	2.266	2.211	2.165	2.126	2.092	2.063	2.037	2.015	1.932	1.792	1.740	1.683	1.654
40	4.085	3.232	2.839	2.606	2.449	2.336	2.249	2.180	2.124	2.077	2.038	2.003	1.974	1.948	1.924	1.839	1.693	1.637	1.577	1.544
50	4.034	3.183	2.790	2.557	2.400	2.286	2.199	2.130	2.073	2.026	1.986	1.952	1.921	1.895	1.871	1.784	1.634	1.576	1.511	1.476
60	4.001	3.150	2.758	2.525	2.368	2.254	2.167	2.097	2.040	1.993	1.952	1.917	1.887	1.860	1.836	1.748	1.594	1.534	1.467	1.430
80	3.960	3.111	2.719	2.486	2.329	2.214	2.126	2.056	1.999	1.951	1.910	1.875	1.845	1.817	1.793	1.703	1.545	1.482	1.411	1.370
100	3.936	3.087	2.696	2.463	2.305	2.191	2.103	2.032	1.975	1.927	1.886	1.850	1.819	1.792	1.768	1.676	1.515	1.450	1.376	1.333
120	3.920	3.072	2.680	2.447	2.290	2.175	2.087	2.016	1.959	1.910	1.869	1.834	1.803	1.775	1.750	1.659	1.495	1.429	1.352	1.307
240	3.881	3.033	2.642	2.409	2.252	2.136	2.048	1.977	1.919	1.870	1.829	1.793	1.761	1.733	1.708	1.614	1.445	1.375	1.290	1.237

APÊNDICE G – Tabela para o Teste de Tukey

<i>GL</i> (<i>N</i> - <i>k</i>)	α	<i>k</i> níveis								
		2	3	4	5	6	7	8	9	10
5	0,05	3,64	4,6	5,22	5,67	6,03	6,33	6,58	6,8	6,99
	0,01	5,7	6,98	7,8	8,42	8,91	9,32	9,67	9,97	10,24
6	0,05	3,46	4,34	4,9	5,3	5,63	5,9	6,12	6,32	6,49
	0,01	5,24	6,33	7,03	7,56	7,97	8,32	8,61	8,87	9,1
7	0,05	3,34	4,16	4,68	5,06	5,36	5,61	5,82	6	6,16
	0,01	4,95	5,92	6,54	7,01	7,37	7,68	7,94	8,17	8,37
8	0,05	3,26	4,04	4,53	4,89	5,17	5,4	5,6	5,77	5,92
	0,01	4,75	5,64	6,2	6,62	6,96	7,24	7,47	7,68	7,86
9	0,05	3,2	3,95	4,41	4,76	5,02	5,24	5,43	5,59	5,74
	0,01	4,6	5,43	5,96	6,35	6,66	6,91	7,13	7,33	7,49
10	0,05	3,15	3,88	4,33	4,65	4,91	5,12	5,3	5,46	5,6
	0,01	4,48	5,27	5,77	6,14	6,43	6,67	6,87	7,05	7,21
11	0,05	3,11	3,82	4,26	4,57	4,82	5,03	5,2	5,35	5,49
	0,01	4,39	5,15	5,62	5,97	6,25	6,48	6,67	6,84	6,99
12	0,05	3,08	3,77	4,2	4,51	4,75	4,95	5,12	5,27	5,39
	0,01	4,32	5,05	5,5	5,84	6,1	6,32	6,51	6,67	6,81
13	0,05	3,06	3,73	4,15	4,45	4,69	4,88	5,05	5,19	5,32
	0,01	4,26	4,96	5,4	5,73	5,98	6,19	6,37	6,53	6,67
14	0,05	3,03	3,7	4,11	4,41	4,64	4,83	4,99	5,13	5,25
	0,01	4,21	4,89	5,32	5,63	5,88	6,08	6,26	6,41	6,54
15	0,05	3,01	3,67	4,08	4,37	4,59	4,78	4,94	5,08	5,2
	0,01	4,17	4,84	5,25	5,56	5,8	5,99	6,16	6,31	6,44
16	0,05	3	3,65	4,05	4,33	4,56	4,74	4,9	5,03	5,15
	0,01	4,13	4,79	5,19	5,49	5,72	5,92	6,08	6,22	6,35
17	0,05	2,98	3,63	4,02	4,3	4,52	4,7	4,86	4,99	5,11
	0,01	4,1	4,74	5,14	5,43	5,66	5,85	6,01	6,15	6,27
18	0,05	2,97	3,61	4	4,28	4,49	4,67	4,82	4,96	5,07
	0,01	4,07	4,7	5,09	5,38	5,6	5,79	5,94	6,08	6,2
19	0,05	2,96	3,59	3,98	4,25	4,47	4,65	4,79	4,92	5,04
	0,01	4,05	4,67	5,05	5,33	5,55	5,73	5,89	6,02	6,14
20	0,05	2,95	3,58	3,96	4,23	4,45	4,62	4,77	4,9	5,01
	0,01	4,02	4,64	5,02	5,29	5,51	5,69	5,84	5,97	6,09
24	0,05	2,92	3,53	3,9	4,17	4,37	4,54	4,68	4,81	4,92
	0,01	3,96	4,55	4,91	5,17	5,37	5,54	5,69	5,81	5,92
30	0,05	2,89	3,49	3,85	4,1	4,3	4,46	4,6	4,72	4,82
	0,01	3,89	4,45	4,8	5,05	5,24	5,4	5,54	5,65	5,76
40	0,05	2,86	3,44	3,79	4,04	4,23	4,39	4,52	4,63	4,73
	0,01	3,82	4,37	4,7	4,93	5,11	5,26	5,39	5,5	5,6
60	0,05	2,83	3,4	3,74	3,98	4,16	4,31	4,44	4,55	4,65
	0,01	3,76	4,28	4,59	4,82	4,99	5,13	5,25	5,36	5,45
120	0,05	2,8	3,36	3,68	3,92	4,1	4,24	4,36	4,47	4,56

	0,01	3,7	4,2	4,5	4,71	4,87	5,01	5,12	5,21	5,3
∞	0,05	2,77	3,31	3,63	3,86	4,03	4,17	4,29	4,39	4,47
	0,01	3,64	4,12	4,4	4,6	4,76	4,88	4,99	5,08	5,16

APÊNDICE H – Valores Reais e Valores Encontrados por Participante.

Estudo E1		
Estimativa	Pontos de Função	Semanas
Real	18	0,6750

Estudo E2		
Estimativa	Pontos de Função	Semanas
Real	71	2,6625

Estudo E3		
Estimativa	Pontos de Função	Semanas
Real	30	1,1250

Estudo E1						
Participante	Abordagem CA		Abordagem FM		Abordagem EX	
	PF	Semanas	PF	Semanas	PF	Semanas
Participante 1	35	1,3125	22	0,8250	22	0,8250
Participante 2	35	1,3125	12	0,4500	12	0,4500
Participante 3	35	1,3125	20	0,7500	20	0,7500
Participante 4	35	1,3125	19	0,7125	19	0,7125
Participante 5	35	1,3125	17	0,6375	17	0,6375
Participante 6	35	1,3125	30	1,1250	30	1,1250
Participante 7	35	1,3125	31	1,1625	31	1,1625
Participante 8	35	1,3125	22	0,8250	22	0,8250

Estudo E2						
Participante	Abordagem CA		Abordagem FM		Abordagem EX	
	PF	Semanas	PF	Semanas	PF	Semanas
Participante 1	140	5,2500	67	2,5125	67	2,6316
Participante 2	140	5,2500	74	2,7750	74	2,9066
Participante 3	140	5,2500	89	3,3375	89	3,4958
Participante 4	105	3,9375	76	2,8500	76	2,9851
Participante 5	140	5,2500	58	2,1750	58	2,2781
Participante 6	105	3,9375	68	2,5500	68	2,6709
Participante 7	105	3,9375	55	2,0625	55	2,1600
Participante 8	N/A	N/A	N/A	N/A	N/A	N/A

Estudo E3						
Participante	Abordagem CA		Abordagem FM		Abordagem EX	
	PF	Semanas	PF	Semanas	PF	Semanas
Participante 1	70	2,625	28	1,0500	28	1,4880
Participante 2	70	2,625	28	1,0500	28	1,4880
Participante 3	70	2,625	51	1,9125	51	1,9125
Participante 4	70	2,625	36	1,3500	36	1,6270
Participante 5	70	2,625	32	1,2000	32	1,5578
Participante 6	70	2,625	41	1,5375	41	1,7146
Participante 7	70	2,625	41	1,5375	41	1,7146
Participante 8	70	2,625	29	1,0875	29	1,5056