



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



UMA TAXONOMIA DE PREDITORES DE ESFORÇO PARA PROJETOS DE APLICATIVOS MÓVEIS

ERVILI TARSILA BRITO DE SOUZA

Manaus, dezembro de 2018

ERVILI TARSILA BRITO DE SOUZA

**UMA TAXONOMIA DE PREDITORES DE ESFORÇO PARA PROJETOS DE
APLICATIVOS MÓVEIS**

Dissertação submetida ao corpo docente do Programa de Pós-Graduação em Informática da Universidade Federal do Amazonas (PPGI-UFAM). Área de concentração: Estimativa de Projetos de Software.

Orientadora: Prof^ª. Tayana Uchôa Conte, D.Sc.
Colaboração: Prof^ª Natasha Malveira Costa Valentim, D.Sc.

Manaus, dezembro de 2018

ERVILI TARSILA BRITO DE SOUZA

**UMA TAXONOMIA DE PREDITORES DE ESFORÇO PARA PROJETOS DE
APLICATIVOS MÓVEIS**

Dissertação submetida ao corpo docente do Programa de Pós-Graduação em Informática da Universidade Federal do Amazonas (PPGI-UFAM). Área de concentração: Estimativa de Projetos de Software.

Banca Examinadora:

Prof^ª. Tayana Uchôa Conte, D.Sc. (Orientadora)

Prof. José Reginaldo H. Carvalho, D.Sc.

Prof. Rafael Prikladnicki., D.Sc.

Ficha Catalográfica

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

S729u Souza, Ervili Tarsila Brito de
Uma taxonomia de preditores de esforço para projetos de aplicativos móveis / Ervili Tarsila Brito de Souza. 2018
118 f.: il. color; 31 cm.

Orientador: Tayana Uchôa Conte
Dissertação (Mestrado em Informática) - Universidade Federal do Amazonas.

1. Taxonomia. 2. Classificação de Conhecimento. 3. Preditores de Esforço. 4. Aplicativos Móveis. 5. Estimativa de Projeto. I. Conte, Tayana Uchôa II. Universidade Federal do Amazonas III. Título



PODER EXECUTIVO
MINISTÉRIO DA EDUCAÇÃO
INSTITUTO DE COMPUTAÇÃO

PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA



UFAM

FOLHA DE APROVAÇÃO

"Uma Taxonomia de Preditores de Esforço para Projetos de Aplicativos Móveis"

ERVILI TARSILA BRITO DE SOUZA

Dissertação de Mestrado defendida e aprovada pela banca examinadora constituída pelos

Professores:

Profa. Tayana Uchôa Conte - PRESIDENTE

Prof. José Reginaldo Hughes Carvalho - MEMBRO INTERNO

Prof. Rafael Prikladnicki - MEMBRO EXTERNO

Manaus, 17 de Dezembro de 2018

“Permanência, perseverança e persistência, apesar de todos os obstáculos: é o que distingue a alma forte da fraca” – Thomas Carlyle.

AGRADECIMENTOS

Agradeço a Deus, pois sem a permissão dele nada seria possível. Aos meus pais, por toda a ajuda e apoio em todos os momentos. Ao meu amor, por todo carinho, compreensão e apoio nessa jornada. À professora Tayana Conte, por ter me orientado e me ajudado durante todo este trabalho. À professora Natasha pela colaboração, foi fundamental nesta pesquisa. Aos professores Reginaldo Carvalho e Rafael Prikladnicki por participarem da minha banca e pelas contribuições. À Fapeam pelo apoio financeiro ao longo do mestrado. Aos meus colegas do grupo USES pela cooperação, compartilhamento de conhecimento e troca de experiências vividas. Agradeço também a todos os participantes dos estudos apresentados nesta dissertação. Muito obrigada!

RESUMO

UMA TAXONOMIA DE FATORES PREDITORES DE ESFORÇO PARA PROJETOS DE APLICATIVOS MÓVEIS

Orientadora: Tayana Uchôa Conte, D.Sc.

Colaboradora: Natasha Malveira Costa Valentim, D.Sc.

Os aplicativos móveis têm características diferentes dos demais sistemas de informação, características como consciência de localização, tipos de conectividade, dentre outras. Essas características influenciam na estimativa de esforço, especialmente em alguns aspectos como preditores de esforço que são fatores associados com o esforço no sentido de que eles têm um efeito sobre o montante total do esforço necessário para desenvolver um projeto. Existem diferentes preditores, porém não há uma classificação destes que facilite o compartilhamento de conhecimento entre profissionais e pesquisadores, apoie o processo de tomada de decisão, auxilie a seleção de preditores de esforço para estimar projetos de aplicativos móveis e ajude a construir uma base para gestão de conhecimento de lições aprendidas na estimativa de projeto. No contexto da estimativa de esforço de aplicativos móveis, o principal objetivo desta pesquisa é organizar e classificar o corpo de conhecimento existente sobre os preditores de esforço de aplicativos móveis, elaborando e usando uma taxonomia que visa apoiar tanto a pesquisa quanto a prática da estimativa de esforço. Para elaborar a taxonomia, foi usado um método de projeto de taxonomia proposto recentemente e foram executados estudos experimentais com objetivo de caracterizar esses preditores de esforço. O primeiro estudo consistiu em uma pesquisa qualitativa com 8 profissionais envolvidos na estimativa de esforço de 5 diferentes empresas, resultando em 54 fatores. O segundo estudo consistiu em um mapeamento sistemático da literatura que identificou 66 fatores e 11 modelos de estimativa para projetos de aplicativos móveis. Por meio dos resultados obtidos dos estudos, foi proposta uma taxonomia de preditores de esforço (métricas de tamanho e fatores de custo) que possui uma hierarquia de três níveis e classificados em 12 categorias, além disso, foi usado uma base de conhecimento sobre preditores de esforço de 108 preditores organizados nas 12 categorias da taxonomia. A taxonomia proposta pode ser benéfica das seguintes maneiras: i) auxiliar a seleção de fatores preditores de esforço para estimar projetos de aplicativos móveis; ii) ajudar a identificar literatura de interesse; iii) auxiliar a construção de uma base para gestão do conhecimento de lições aprendidas na estimativa de projeto.

Palavras-chave: Estimativa de Esforço, Estimativa de Tamanho, Aplicativos Móveis, Fatores Preditores de Esforço, Taxonomia, Classificação de Conhecimento.

ABSTRACT

A TAXONOMY OF EFFORT PREDICTORS FACTORS FOR MOBILE APPLICATION PROJECTS

Advisors: Tayana Uchôa Conte, D.Sc.

Mobile applications have different characteristics in relation to other information systems, such as location awareness, types of connectivity, among others. These characteristics influence the estimation of effort, especially in some aspects like stress predictors that are factors associated with effort in the sense that they have an effect on the total amount of effort required to develop a project. There are different predictors, but there is no such classification that facilitates the sharing of knowledge among professionals and researchers, supports the decision-making process, assists the selection of effort predictors to estimate mobile application projects, and helps build a basis for managing knowledge of lessons learned in project estimation. In the context of mobile application effort estimation, the main goal of this research is to organize and classify the existing body of knowledge about mobile application effort predictors, designing and using a taxonomy that supports both research and the practice of effort estimating. To elaborate the taxonomy, a recently proposed taxonomy design method was used and experimental studies were carried out to characterize these effort predictors. The first study consisted of a qualitative research with 8 professionals involved in the effort estimation of 5 different companies, resulting in 54 factors. The second study consisted of a systematic literature mapping that identified 66 factors from the studies, a taxonomy of effort predictors (size metrics and cost factors) was proposed, with a hierarchy of three levels and classified in 12 categories, in addition, a knowledge base of 108 effort predictors was used and the predictors were organized in the 12 categories of taxonomy. The proposed taxonomy may be beneficial in the following ways: i) assist in the selection of effort predictors to estimate mobile application projects; ii) help identify literature of interest; iii) help build a base for knowledge management of lessons learned in project estimation.

Keywords: Effort estimation, Size Estimation, Mobile Applications, Effort Predictive Factors, Taxonomy, Knowledge Classification.

LISTA DE FIGURAS

Figura 1.1: Metodologia de pesquisa para a definição e avaliação da Taxonomia de Preditores de Esforço para Projetos de Aplicativos Móveis	17
Figura 2.1: Preditores de esforço.....	21
Figura 2.2: Técnica algorítmica para estimativa de esforço.....	25
Figura 2.3: Exemplo de uma árvore de regressão para estimativa de esforço Web.....	27
Figura 2.4: Usando CART para estimativa de esforço	27
Figura 3.1: Exemplo de codificação aberta - Citação e Código.....	36
Figura 3.2: Relações entre as Métricas de Tamanho	38
Figura 3.3: Relações entre os Fatores de Custo	38
Figura 3.4: Variação do Processo de Estimativa de Esforço nas Empresas.....	48
Figura 4.1: Processo de Seleção de Publicação.....	55
Figura 4.2: Distribuição das publicações por ano e por veículo de publicação.	58
Figura 4.3: Fatores Preditores de Esforço: Como estão organizados.....	67
Figura 5.1: Categorias da Taxonomia de Britto <i>et al.</i> (2017)	79
Figura 5.2: Taxonomia de Preditores de Esforço de Projetos de Aplicativos Móveis.....	79
Figura 5.3: Percentual de tipos de aplicativos estimados.....	95
Figura 5.4: Avaliação das Métricas de Length.....	97
Figura 5.5: Avaliação das Métricas de Complexidade.....	97
Figura 5.6: Avaliação das Métricas de Funcionalidade	98
Figura 5.7: Avaliação de Preditores relacionados ao Cliente.....	99
Figura 5.8: Avaliação de Preditores relacionados à Empresa	100
Figura 5.9: Avaliação de Preditores relacionados à Equipe de Desenvolvimento.....	101
Figura 5.10: Avaliação de Preditores relacionados ao Projeto.....	104
Figura 5.11: Avaliação de Preditores relacionados ao Produto	105
Figura 5.12: Avaliação de Preditores relacionados à Tecnologia	106

LISTA DE TABELAS

Tabela 1.1: Método de projeto da taxonomia.....	18
Tabela 2.1: Funcionalidades do sistema (agrupamento de requisitos).....	28
Tabela 2.2: Principais Métodos de Estimativas.....	30
Tabela 3.1: Experiências dos Entrevistados.....	33
Tabela 3.2: Questões usadas nas entrevistas.....	35
Tabela 3.3: Preditores de Esforço em Projetos de Aplicativos Móveis.....	39
Tabela 4.1: Questões de investigação estruturadas pelos critérios do PIO.....	52
Tabela 4.2: Critérios de Seleção.....	54
Tabela 4.3: Interpretação dos resultados do Kappa segundo Landis e Koch (1977).	56
Tabela 4.4: Lista das publicações relevantes extraídas.....	56
Tabela 4.5: Formulário de Extração dos Dados.....	57
Tabela 4.6: Lista de Modelos Extraídos.....	61
Tabela 4.7: Lista de Modelos Avaliados Experimentalmente.....	67
Tabela 4.8: Lista de Fatores Preditores de Esforço.....	68
Tabela 5.1: Resultados do Controle de Terminologia.....	78
Tabela 5.2: Fatores relacionados a Métricas de Tamanho.....	82
Tabela 5.3: Fatores relacionados ao Cliente.....	84
Tabela 5.4: Fatores relacionados à Empresa.....	85
Tabela 5.5: Fatores relacionados à Equipe de Desenvolvimento.....	86
Tabela 5.6: Fatores relacionados ao Produto.....	87
Tabela 5.7: Fatores relacionados ao Projetos de Aplicativos Móveis.....	90
Tabela 5.8: Fatores relacionados à Tecnologia.....	92
Tabela 5.9: Objetivo do estudo preliminar de acordo com o paradigma GQM.	93
Tabela 5.10: Características dos Participantes do Estudo de Avaliação.....	94

SUMÁRIO

CAPÍTULO 1 – INTRODUÇÃO	14
1.1 CONTEXTO	14
1.2 DEFINIÇÃO DO PROBLEMA	15
1.3 OBJETIVOS.....	16
1.4 METODOLOGIA DE PESQUISA	16
1.5 ORGANIZAÇÃO DO TEXTO	19
CAPÍTULO 2 – ESTIMATIVA DE PROJETOS DE SOFTWARE	20
2.1 INTRODUÇÃO.....	20
2.2 CONCEITOS DE ESTIMATIVA	20
2.3 PROCESSO DE ESTIMATIVA GENÉRICO	21
2.4 CATEGORIAS DE ESTIMATIVA DE ESFORÇO	22
2.4.1 Baseada por Especialistas	22
2.4.2 Técnicas algorítmicas.....	23
2.4.3 Técnicas de Inteligência Artificial	25
2.5 PRINCIPAIS MÉTODOS DE ESTIMATIVA TRADICIONAIS EXISTENTES NA LITERATURA	29
2.6 CONSIDERAÇÕES SOBRE O CAPÍTULO.....	31
CAPÍTULO 3 – ESTUDO QUALITATIVO SOBRE ESTIMATIVA DE ESFORÇO NO DESENVOLVIMENTO DE APLICATIVOS MÓVEIS... 32	32
3.1 INTRODUÇÃO.....	32
3.2 COLETA DE DADOS.....	32
3.3 ANÁLISE DE DADOS	34
3.4 AMEAÇAS À VALIDADE	37
3.5 RESULTADOS	38
3.5.1 Preditores de Esforço	44
3.5.2 Variações do Processo de Estimativa de Esforço	48
3.6 CONSIDERAÇÕES SOBRE O CAPÍTULO	49
CAPÍTULO 4 – MAPEAMENTO SISTEMÁTICO SOBRE ESTIMATIVAS PARA PROJETOS DE APLICATIVOS MÓVEIS..... 51	51
4.1 INTRODUÇÃO.....	51
4.2 PROTOCOLO DO MAPEAMENTO SISTEMÁTICO	52
4.2.1 Questões de Pesquisa	52
4.2.2 Termos de Pesquisa.....	53
4.2.3 Fontes de Busca	53
4.2.4 Critérios de Seleção de Estudos	54
4.2.5 Processo de Seleção de Estudos.....	54
4.2.6 Procedimento de Extração dos Dados dos Artigos Selecionados	57
4.3 RESULTADOS DO MAPEAMENTO SISTEMÁTICO	58

4.3.1	Visão Geral dos Resultados	58
4.3.2	Que modelos e métodos têm sido utilizados para a estimativa de tamanho e esforço de aplicativos móveis?	58
4.3.3	Como foram avaliados os modelos e métodos de estimativa?	66
4.3.4	Que fatores têm sido investigados como preditores de esforço para Aplicativos Móveis?	67
4.4	DISCUSSÃO DOS RESULTADOS	74
4.5	CONSIDERAÇÕES SOBRE O CAPÍTULO	75
CAPÍTULO 5 – UMA TAXONOMIA DE PREDITORES DE ESFORÇO PARA PROJETOS DE APLICATIVOS MÓVEIS.....		76
5.1	INTRODUÇÃO.....	76
5.2	PLANEJAMENTO DA TAXONOMIA	76
5.3	IDENTIFICAÇÃO E EXTRAÇÃO	77
5.4	DESIGN E CONSTRUÇÃO	78
5.5	CLASSIFICAÇÃO DA BASE DE CONHECIMENTO DE PREDITORES DE ESFORÇO DE PROJETOS DE APLICATIVOS MÓVEIS	81
5.5.1	Métricas de tamanho	81
5.5.2	Fatores de custo.....	83
5.6	AValiação DA TAXONOMIA	92
5.6.1	Demonstração de ortogonalidade.....	92
5.6.2	Demonstração de utilidade:.....	93
5.7	DISCUSSÃO.....	106
5.7.1	Contribuições para a academia.....	106
5.7.2	Contribuições para a indústria.....	107
5.8	CONSIDERAÇÕES SOBRE O CAPÍTULO	108
CAPÍTULO 6 – CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS.....		109
6.1	CONCLUSÕES.....	109
6.2	CONTRIBUIÇÕES	111
6.3	PERSPECTIVAS FUTURAS	112
REFERÊNCIAS		113
APÊNDICE A – ACORDO DE CONFIDENCIALIDADE		117
APÊNDICE B – LEGENDA DAS REFERÊNCIAS DAS ANÁLISES DOS DADOS		118

CAPÍTULO 1 – INTRODUÇÃO

Este capítulo apresenta a introdução a esta dissertação de mestrado. Além de contextualizar esta pesquisa, são apresentados: o problema, os objetivos, a metodologia e a organização do texto.

1.1 CONTEXTO

A Statista (2018) estima que existam mais de 2,16 bilhões de clientes de aplicativos móveis em todo o mundo, prevendo um crescimento de 2,5 bilhões no uso desta tecnologia em 2019. Com a evolução dos dispositivos móveis, os usuários foram motivados a incorporar os aplicativos aos seus dispositivos para ajudá-los a realizar suas atividades diárias, como comunicação, negócios, esportes, educação e notícias.

Um aplicativo móvel é definido como um aplicativo de software desenvolvido especificamente para a geração atual de dispositivos móveis, como smartphones e tablets (Nagappan, 2016). De acordo com Dombroviak e Ramnath (2007), as aplicações móveis têm características diferentes dos demais sistemas de informação, tais como: centralidade (coleta de dados ou execução de ações em resposta ao ambiente), consciência de localização absoluta (usam tipicamente métodos como GPS ou triangulação de sinal), consciência de aproximação (sensibilidade de aproximação de determinado objeto e espaço), consciência de espaço (o dispositivo está dentro da área de limite definido de lugares, exemplos: edifícios, prédios e etc.) e consciência de transição (o grau em que o comportamento de uma aplicação depende do conhecimento de transições entre espaços).

Além dessas, De Souza e Aquino (2014) complementam com as seguintes características: consumo de bateria, tipos de conectividade, fator de desempenho, dentre outros. Segundo Nitze *et al.* (2014), essas características são fatores influentes que modificam a estimativa de esforço. Devido a isto, a estimativa de esforço deve ser adaptada para este novo contexto de aplicativos móveis. A estimativa de esforço é o processo usado para prever o trabalho necessário para desenvolver uma determinada aplicação e é fundamental para gerenciar recursos do projeto de uma forma eficiente (Mendes, 2008). A estimativa de esforço de desenvolvimento de projeto envolve a seleção de uma técnica adequada de estimativa, bem como a identificação e estimativa de preditores de esforço de aplicativos móveis, isto é, fatores associados com o esforço no sentido de que eles têm um efeito sobre o montante total

do esforço necessário para desenvolver um projeto. Estes fatores podem ser constituídos de métricas de tamanho e fatores de custo (Britto *et al.*, 2017).

1.2 DEFINIÇÃO DO PROBLEMA

Segundo De Souza e Conte (2017), nos últimos 13 anos, foram propostas várias métricas de tamanho e fatores de custo de projetos de aplicativos móveis motivadas principalmente para serem utilizadas na estimativa de esforço. Apesar da sua importância para os profissionais que desenvolvem aplicativos móveis e aqueles que investigam a estimativa de esforço, não foi encontrada dentro do escopo desta pesquisa uma referência única até à data que mapeie e classifique esses fatores apontados pelos pesquisadores e praticantes da indústria de aplicativos móveis. Portanto, este trabalho busca responder a seguinte questão de pesquisa: Como organizar o conhecimento sobre preditores de esforço de projetos de aplicativos móveis?

Com esta questão de pesquisa, surge a necessidade de projetar uma taxonomia para classificar preditores de esforço para projetos de aplicativos móveis. Uma taxonomia é um esquema de classificação que foi inicialmente projetada por Carl Linnaeus para classificar os organismos (Linnaeus, 1758). Pesquisadores usaram essa abordagem para organizar o conhecimento em outros campos, como Ciência da Computação (Glass e Vessey, 1995) e Psicologia (Moffitt, 1993). Originalmente, ela foi projetada para classificar o conhecimento de forma hierárquica, embora existam diferentes estruturas para construir taxonomias como hierarquia, árvore e baseada em facetas (Kwasnik, 1999). A classificação do conhecimento pode ter implicações positivas tanto para a academia como para a indústria da seguinte forma (Brito *et. al.*, 2017):

- Facilita o compartilhamento de conhecimento.
- Ajuda a identificar lacunas em uma área de conhecimento particular.
- Fornece uma melhor compreensão das inter-relações entre os fatores associados a uma área de conhecimento particular.
- Apoia a estimativa de projeto, no sentido de tomada de decisão (quais recursos usar, número de telas e dentre outras coisas).

O valor dessa pesquisa é auxiliar os profissionais que estimam aplicativos móveis a organizarem os preditores de esforço de forma a apoiar durante o processo de estimativa, pois eles terão uma taxonomia que servirá como container para receber histórico de preditores de outros projetos de aplicativos móveis.

1.3 OBJETIVOS

O principal objetivo deste mestrado foi *elaborar uma taxonomia para organizar o conhecimento sobre preditores de esforço de projetos de aplicativos móveis (métricas de tamanho e fatores de custo)*. Para alcançar este objetivo geral, buscou-se decompô-lo nos seguintes objetivos específicos apresentados a seguir:

- Mapeamento de uma base de conhecimento sobre preditores de esforço de projetos de aplicativos móveis na literatura científica.
- Mapeamento de uma base de conhecimento sobre preditores de esforço de projetos de aplicativos móveis na indústria de software com praticantes e especialistas de estimativa.
- Projeto de uma taxonomia de preditores de esforço para projetos de aplicativos móveis baseada em evidências dos preditores encontrados na literatura científica e na indústria de software.
- Validação da taxonomia de preditores de esforço, seguindo a metodologia na presente pesquisa.

1.4 METODOLOGIA DE PESQUISA

Nesta pesquisa de Mestrado foram realizados diferentes estudos, com o propósito de alcançar os objetivos definidos na Seção 1.3. A presente metodologia foi dividida em duas fases: a Fase 1 (Exploratória) busca de evidências na literatura e na indústria sobre preditores de esforço. A Fase 2 (Projeto da Taxonomia) buscou projetar uma taxonomia de preditores de esforço identificados na literatura e na indústria.

Nesta Fase 2, foi utilizado o método proposto por Usman *et al.* (2017) para projetar uma taxonomia, pois o método é uma revisão do método proposto por Bayona-Oré *et al.* (2014) que foi o primeiro método de construção de uma taxonomia (Usman *et al.*, 2017) e também o método de Usman *et al.* (2017) foi utilizado na construção de uma taxonomia de preditores de esforço para projetos web (Britto *et al.*, 2017). A seguir são detalhadas as atividades que foram realizadas em cada fase da metodologia representada na Figura 1.1.

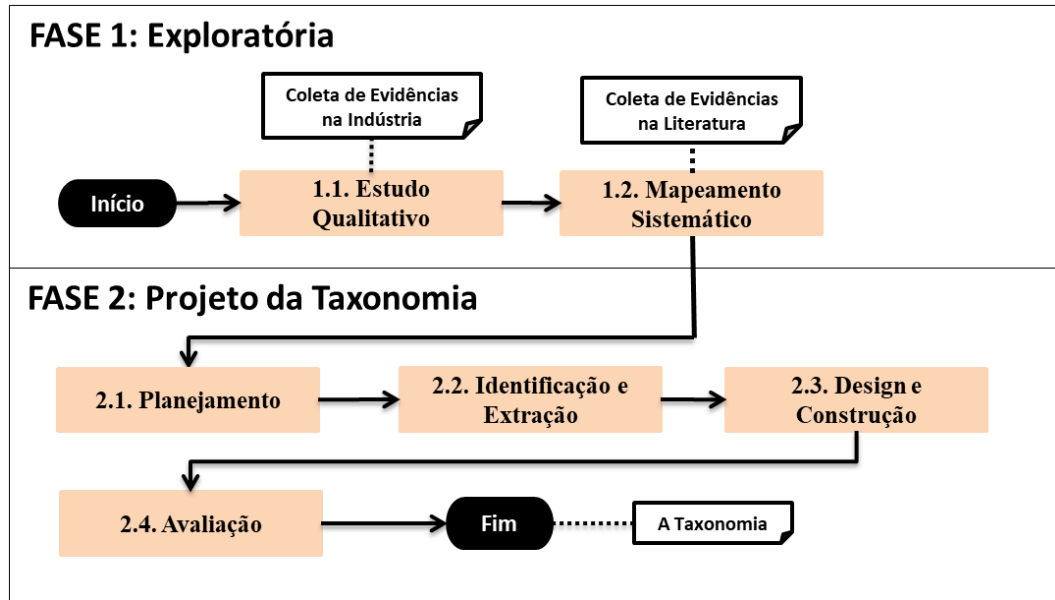


Figura 1.1: Metodologia de pesquisa para a definição e avaliação da Taxonomia de Preditores de Esforço para Projetos de Aplicativos Móveis

Fase 1 - Exploratória: Buscou-se na literatura e na indústria evidências sobre preditores de esforço de aplicativos móveis. Nesse sentido, as atividades desta primeira fase foram:

- 1.1. **Estudo Qualitativo:** Realização de um estudo qualitativo com especialistas de estimativa de projetos de aplicativos móveis na indústria para a identificação de modelos de estimativa e preditores de esforço. Este estudo foi realizado antes do mapeamento sistemático para evitar o viés de aprendizado dos pesquisadores.
- 1.2. **Mapeamento Sistemático da Literatura (MSL):** Realização de estudos secundários para identificação, avaliação e interpretação de todos os resultados relevantes sobre os modelos de estimativa e preditores de esforço de aplicativos móveis.

Fase 2 - Projeto da Taxonomia: Para projetar a taxonomia, utilizou-se o método proposto por Usman *et al.* (2017). Este método possui 13 atividades (Tabela 1.1) que foram agrupadas em 4 fases como segue abaixo:

- 2.1. **Planejamento:** Esta fase está relacionada à definição de aspectos iniciais da taxonomia a serem projetadas, como o objetivo da taxonomia, o conhecimento de engenharia de software (SE) (Bourque e Farley, 2014) associado com o assunto selecionado para ser classificado, os métodos de coleta de dados (Wohlin, 2014), tipo de estrutura de classificação (hierarquia, árvore, paradigma ou baseada em

facetar (Kwasnik, 1999)) e tipo de procedimento de classificação (qualitativo ou quantitativo (Wheaton, 1968)).

2.2. **Identificação e Extração:** Esta fase está relacionada à extração da informação e controle de terminologia que descreve os termos extraídos.

2.3. **Design e Construção:** As dimensões e categorias são identificadas e descritas, bem como as relações entre elas, levando a um esquema de classificação. Diretrizes para uso e evolução da taxonomia também foram definidos nesta fase.

2.4. **Avaliação:** Na última fase deste processo, a taxonomia é validada. Uma taxonomia pode ser validada de três maneiras (Britto *et al.*, 2017):

- ✓ **Demonstração de ortogonalidade:** Uma abordagem ascendente para identificar as categorias da taxonomia apresentada: Primeiro, identificar todos os preditores de esforço relatados nos estudos. Segundo, analisar os preditores de esforço com o objetivo de identificar as semelhanças e diferenças entre os preditores, este processo levará a identificar categorias ortogonais.
- ✓ **Benchmarking:** Comparar a taxonomia proposta com as demais taxonomias existentes sobre preditores de esforço de aplicativos móveis.
- ✓ **Demonstração de utilidade:** Para demonstrar a utilidade da taxonomia proposta serão classificados os fatores de custo e as métricas de tamanho extraídos do mapeamento sistemático e estudo qualitativo, e todos os preditores de esforço serão validados em relação à frequência e pretensão de uso, do ponto de vista de gerentes de projeto, desenvolvedores e analistas, no contexto da indústria que desenvolve aplicativos móveis.

Tabela 1.1: Método de projeto da taxonomia

Fase	Id	Atividade
Planejamento	B1	Definir área de conhecimento da Engenharia de Software
	B2	Descrever os objetivos da Taxonomia
	B3	Selecionar e descrever o assunto a ser classificado
	B4	Selecionar a estrutura do tipo de classificação
	B5	Selecionar o procedimento do tipo de classificação
	B6	Identificar as fontes de dados
Identificação e Extração	B7	Extrair todos os termos
	B8	Executar o controle de terminologia
Design e Construção	B9	Identificar e descrever as dimensões do projeto da taxonomia
	B10	Identificar e descrever as categorias de cada dimensão

	B11	Identificar e descrever os relacionamentos
	B12	Definir diretrizes para usar e atualizar a taxonomia
Avaliação	B13	Avaliar a taxonomia

1.5 ORGANIZAÇÃO DO TEXTO

Este capítulo introdutório apresentou a contextualização, a definição do problema, os objetivos e a metodologia de pesquisa a que se refere esta dissertação. O conteúdo será detalhado ao longo dos próximos capítulos. Este trabalho está organizado segundo a estrutura descrita abaixo:

Capítulo 2 – Estimativa de Projetos de Software: Descreve os conceitos relacionados à estimativa, processos de estimativas genéricos, categoria de estimativa de esforço e principais métodos tradicionais existentes na literatura.

Capítulo 3 – Estudo Qualitativo sobre Estimativa de Esforço no Desenvolvimento de Aplicativos Móveis: Identifica e analisa fatores que afetam a estimativa de esforço e o modo de como é realizada a estimativa de esforço nas empresas que desenvolvem projetos de aplicativos móveis.

Capítulo 4 – Mapeamento Sistemático sobre Estimativas para Projetos de Aplicativos Móveis: Apresenta o mapeamento sistemático realizado cujo objetivo consistiu em identificar modelos de estimativa de projetos de aplicativos móveis e preditores de esforço.

Capítulo 5 – Uma Taxonomia de Preditores de Esforço de Projetos de Aplicativos Móveis: Apresenta a taxonomia, detalhando as etapas do processo metodológico de construção e de avaliação.

Capítulo 6 – Considerações finais e perspectivas futuras: este capítulo contém as conclusões e contribuições deste trabalho, além de indicar a continuação desta pesquisa através de trabalhos futuros.

CAPÍTULO 2 – ESTIMATIVA DE PROJETOS DE SOFTWARE

Este capítulo apresenta os conceitos relacionados a estimativa de projetos de software, processos de estimativa genéricos, categorias de estimativa e principais métodos existentes na literatura.

2.1 INTRODUÇÃO

De acordo com Khatibi (2012), estimar o custo e o esforço de um projeto de software é uma das atividades mais importantes no gerenciamento de projetos de software. Isso ocorre porque um planejamento, monitoramento e controle do projeto não são viáveis se as estimativas do custo no desenvolvimento de software são altamente imprecisas. Segundo Whigham (2015), neste contexto uma variedade de métodos cada vez mais complexos tem sido considerada nos últimos 30 anos para a predição de esforço, muitas vezes com resultados mistos e contraditórios. Além disso, todos o referencial teórico foi baseado no livro de Mendes (2014).

2.2 CONCEITOS DE ESTIMATIVA

Segundo Mendes (2014), estimar em Engenharia de Software consiste em prever prazo, custo, recurso e esforço necessários para desenvolver um conjunto de tarefas do ciclo de vida de um projeto de software. Além disso, a estimativa pode ser dividida em (Mendes, 2014):

- **Estimativa de Tamanho**

Grandeza física medida através dos requisitos, análise e projeto ou código do software com base nas suas funções e complexidade do problema.

- **Estimativa de Esforço**

Trabalho necessário para desenvolvimento do projeto que pode ser obtido a partir da estimativa de tamanho.

- **Estimativa de Prazo**

Tempo necessário para desenvolvimento do projeto que pode ser obtido a partir da estimativa de esforço e a quantidade de recursos envolvidos no projeto

- **Preditores de Esforço**

Fatores que podem ser constituídos de métricas de tamanho e fatores de custo. Métricas de tamanho são métricas que ajudam a medir o tamanho do projeto

como, por exemplo, a quantidade de elementos da interface gráfica. Já fatores de custo são associados com o esforço no sentido de que eles têm um efeito sobre a quantidade total do esforço necessário para desenvolver um projeto como, por exemplo, o número total de membros da equipe que participarão do desenvolvimento do aplicativo móvel. A Figura 2.1 ilustra o conceito de preditores de esforço que podem ser métricas de tamanho e/ou fatores de custo.

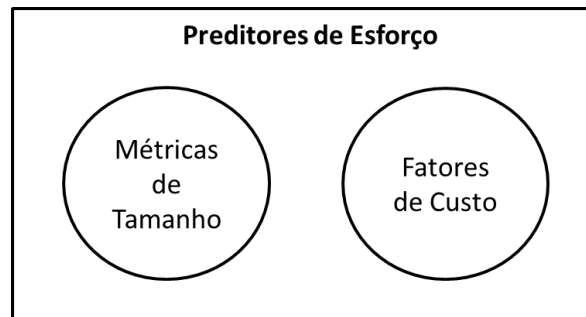


Figura 2.1: Preditores de esforço

2.3 PROCESSO DE ESTIMATIVA GENÉRICO

Mendes (2014) explica o processo geral de estimativas de projetos de software em quatro passos mostrados na Figura 2.2. Para entender o processo, será usado a metáfora da caixa preta, onde as características do projeto (variáveis independentes) são a entrada e o esforço (variável dependente) é a saída que desejamos prever. Por exemplo, uma determinada empresa que desenvolve aplicativos da Web pode achar que para prever o esforço necessário para implementar um novo aplicativo da Web, será exigido as seguintes entradas:

- Número estimado de novas páginas da Web
- O número de funções e recursos (por exemplo, carrinho de compras) a ser oferecido por o novo aplicativo da Web
- Número total de desenvolvedores que ajudarão a desenvolver o novo aplicativo da Web
- Número médio de anos de experiência dos desenvolvedores com as ferramentas que serão usadas para desenvolver a aplicação.
- A escolha da principal linguagem de programação usada

Dessas variáveis, o número estimado de novas páginas da Web e o número de funções ou recursos a serem oferecidos pelo novo aplicativo da Web são variáveis de tamanho (medidas). Os outros três - o número total de desenvolvedores que ajudarão a desenvolver o novo aplicativo Web, o número médio de anos de experiência dos desenvolvedores com as

ferramentas de desenvolvimento utilizadas e a principal linguagem de programação utilizada - não são usado para dimensionar o problema a ser resolvido (aplicativo da Web), em vez disso, eles são preditores que influenciam a quantidade de esforço necessário para desenvolver um aplicativo da Web e nesse sentido mais abstrato, ainda está muito relacionado ao esforço. Portanto, eles também são considerados entradas e conjuntamente denominados fatores de custo.

Uma tarefa a ser estimada pode ser tão simples quanto desenvolver uma única função (por exemplo, criar um formulário da Web com 10 campos) ou tão complexo quanto desenvolver um grande e-commerce. Independentemente do tipo de aplicação, em geral a entrada consistente (variável independente) que se acredita ter a maior influência no esforço é o tamanho (isto é, o número total de páginas da Web), com os fatores de custo desempenhando também um papel influente.

Na maioria dos casos, mesmo quando a estimativa de esforço é baseada em experiências passadas, ou dados sobre projetos concluídos podem ser usados para ajudar a estimar o esforço para novos projetos ainda para começar.

2.4 CATEGORIAS DE ESTIMATIVA DE ESFORÇO

As técnicas de estimativa têm sido propostas há mais de 30 anos e Mendes (2008) divide as técnicas de estimativa em 3 tipos de categorias de estimativa: estimativa baseada por especialistas, técnicas algorítmicas e técnicas de inteligência artificial. A seguir serão explicadas essas categorias.

2.4.1 Baseada por Especialistas

Processo de estimativa baseada na experiência, no desenvolvimento ou gestão de projetos anteriores semelhantes. A realização de estimativas de esforço precisas é diretamente proporcional à competência e experiência dos indivíduos envolvidos no projeto, como por exemplo, gerente de projeto e desenvolvedor.

As estimativas podem ser sugeridas por um gerente de projeto, ou por um grupo de pessoas que englobam gerentes de projetos e desenvolvedores, geralmente por meio de uma sessão de brainstorming. Mendes (2008) sugere que no contexto de desenvolvimento, as estimativas de esforço podem ser:

- **Estimativa Bottom-up**

Levam em conta todas as partes do nível mais baixo de um aplicativo e as tarefas funcionais necessárias para desenvolver esta aplicação. Cada tarefa atribuída com estimativas de esforço é repetidamente combinada em estimativas de nível superior até que finalmente obtém uma estimativa que é considerada como a soma de todas as partes de estimativa de nível inferior.

- **Estimativa Top-down**

Levam em conta todas as partes do nível mais alto de um aplicativo, inicialmente é sugerido uma estimativa total para as tarefas relativas ao conjunto total. Segundo Vliet (2000), existem três tipos de estimativas diferentes baseadas por especialistas: uma estimativa otimista (o), uma estimativa realista (r), e uma estimativa pessimista (p). Com base numa distribuição beta, o esforço E estimado é então calculada como (Equação 1):

$$E = (o + 4r + p)/6 \quad (1)$$

Mendes (2008) mostra um modelo de estimativa de esforço baseada por especialistas. Cada passo é explicado a seguir:

- **Passo 1)** Um especialista ou grupo de desenvolvedores implicitamente olham o tamanho e fatores de custo estimado relacionados com um novo projeto para o qual esforço necessita ser estimado.
- **Passo 2)** Com base nos dados obtidos no passo 1, eles recuperam os dados / conhecimento em projetos anteriores para os quais é conhecido o esforço real.
- **Etapa 3)** Com base nos dados do Passos 1 e 2, eles subjetivamente estimam o esforço para o novo projeto.

2.4.2 Técnicas algorítmicas

Segundo Mendes (2008), as técnicas algorítmicas objetivam construir modelos que representam precisamente a relação entre esforço e uma ou mais características do projeto através do uso de modelos algorítmicos. A relação entre o tamanho e esforço é muitas vezes traduzidos para uma equação mostrado pela Equação 2 a seguir, onde a e b são constantes, S é o tamanho estimado de uma aplicação, e E é o esforço estimado necessário para desenvolver uma aplicação de tamanho S .

$$E = a S^b \quad (2)$$

$$\left\{ \begin{array}{l} b < 1, \quad \text{projetos de maior dimensão que usam menos esforço} \\ b = 1, \quad \text{a relação é linear} \\ b > 1, \text{projetos de maior dimensão utilizam mais esforço} \end{array} \right\}$$

No entanto, o valor pode ser ajustado tendo em conta fatores de custo através da Equação 2.

$$E = aS^b \text{ Custos} \quad (3)$$

Existem modelos algorítmicos exatos e populares, dentre eles, o COCOMO que foi o primeiro modelo construtivo de custo proposto por Boehm em 1981. O COCOMO foi destinado a ser um modelo algorítmico genérico que poderia ser aplicado por qualquer organização para estimar o esforço em três estágios diferentes do ciclo de vida do desenvolvimento de projetos de software:

- No início do ciclo de vida do desenvolvimento, quando os requisitos não foram ainda totalmente especificados (*COCOMO Basic*);
- Depois que o detalhamento dos requisitos forem especificados (*COCOMO Intermediate*);
- Quando o design do aplicativo for finalizado (*COCOMO Advanced*);

Mendes (2008) mostra um modelo (Figura 2.3) de representação de como é feita a estimativa de esforço por modelos de algoritmos. Cada passo será explicado a seguir:

- **Passo 1).** Os dados anteriores são utilizados para gerar um modelo algorítmico.
- **Passo 2).** Um modelo algorítmico é construído a partir dos dados anteriores obtidos no Passo 1.
- **Passo 3).** O modelo criado no passo 2, em seguida, recebe como entrada, os valores para o tamanho estimado e fatores de custo em relação ao novo projeto para o qual o esforço é para ser estimado.
- **Passo 4).** O modelo gera um esforço estimado.

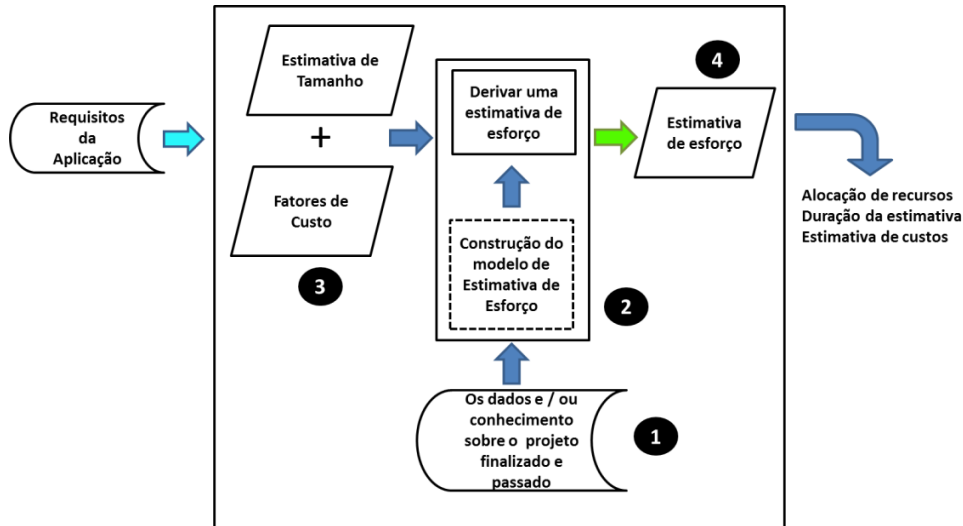


Figura 2.2: Técnica algorítmica para estimativa de esforço

Fonte: Mendes (2008).

2.4.3 Técnicas de Inteligência Artificial

Nos últimos 20 anos, as técnicas de inteligência artificial têm sido usadas como um complemento ou como uma alternativa para as duas categorias anteriores (estimativa de esforço baseada por especialista e por técnicas algorítmicas).

Segundo Mendes (2008) essas técnicas de inteligência artificial possuem quatro subcategorias bastante difundidas para estimativa de esforço em projetos de software que são Lógica Fuzzy, Classificação e Árvore de Regressão (CART), Redes Neurais e Raciocínio baseado em casos (CBR).

- **Lógica Fuzzy**

A lógica Fuzzy é a forma de lógica multivalorada na qual os valores lógicos das variáveis podem ser qualquer número real entre 0 e 1. Onde “0” correspondente ao valor falso e “1” correspondente ao valor verdadeiro.

Segundo Borsoi *et al.* (2011), a utilização de conjuntos fuzzy para lidar com conceitos inexatos foi inicialmente proposta por Zadeh em 1965, motivado pelo fato de que muitas classes de objetos existentes no mundo físico não apresentam critérios de pertinência definidos com precisão. A teoria dos conjuntos Fuzzy propõe uma generalização da função característica, originando uma função de pertinência, que determina com que grau um objeto x pertence a um conjunto A no universo em questão (Borsoi *et al.*, 2011).

Portanto, a lógica Fuzzy é utilizada para ajudar na estimativa de desenvolvimento de projeto de software tendo como base os requisitos do sistema, que representam as funcionalidades requeridas ou definidas pelo usuário.

- **Classificação e Árvore de Regressão (CART)**

Segundo Mendes (2008), árvore de regressão usa variáveis independentes (preditoras) para construir árvores binárias, em que cada nó folha representa uma categoria à qual pertence uma estimativa ou um valor para uma estimativa.

Sempre que são categóricas preditoras (por exemplo, Sim / Não) a árvore CART é chamado de árvore de classificação e sempre que são preditoras numéricas a árvore CART é chamado de árvore de regressão.

Na Figura 2.4 mostra um exemplo de uma árvore de regressão onde as variáveis independentes são: NW (Novo Projeto Web), NWP (Novas Páginas da Web), NIM (Novas Imagens) e NFN (Novas Funcionalidades/Funções). Utilizando a árvore de regressão da Figura 2.4 para supor um novo projeto que possuirá $NWP = 25$, $NIM = 15$ e $NFN = 3$, logo o esforço necessário será de 45 horas por pessoa ao realizar o caminhamento na árvore da sua raiz até as folhas.

Mendes (2008) mostra um modelo (Figura 2.4) de representação de como é feita a estimativa de esforço utilizando CART. Cada passo será explicado a seguir:

- **Passo 1)** Dados anteriores é utilizado para gerar um modelo CART.
- **Passo 2)** Um modelo CART é construído com base nos dados passados obtidos na Etapa
- **Passo 3)** O modelo criado na passo 2 recebe, como entrada, valores / categorias para o tamanho estimado e fatores de custo em relação ao novo projeto para o qual o esforço é estimado.
- **Passo 4)** O modelo gera um valor / categoria de esforço estimado.

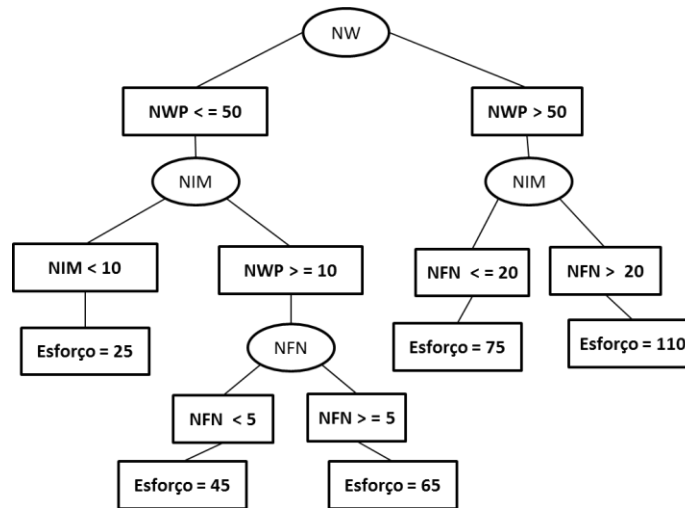


Figura 2.3: Exemplo de uma árvore de regressão para estimativa de esforço Web

Fonte: Mendes (2008).

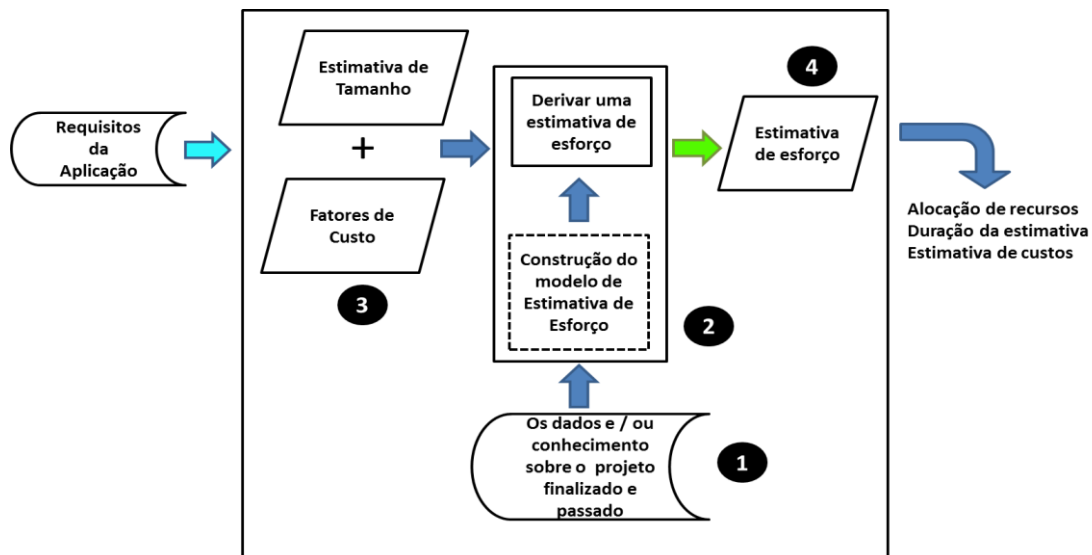


Figura 2.4: Usando CART para estimativa de esforço

Fonte: Mendes (2008).

- **Redes Neurais**

Segundo Borsoi *et al.* (2011), o uso de redes neurais decorre de elas empregarem técnicas de aproximação de funções por regressão não linear, aproximando-se da forma como um especialista realiza estimativas.

Isso deve-se ao fato de que os fatores como o prazo não têm aumento linear proporcional ao número de requisitos de entrada. As redes neurais utilizam como entrada os requisitos do sistema a ser desenvolvido e o tempo padrão para implementar cada tipo de requisito. Esse tempo é definido pelas redes treinadas.

Borsoi *et al.* (2011) afirmam que o treinamento está baseado em padrões de tempo informados por especialistas.

As entradas para as redes utilizam uma tabela de fatores definidores de prazo (Tabela 2.1). Os fatores que correspondem a esses agrupamentos e as respectivas quantidades associadas estão em Borsoi *et al.* (2011).

Tabela 2.1: Funcionalidades do sistema (agrupamento de requisitos)

Grupos	Fatores definidores de prazo – Quantidade
Manutenção de dados	1. Simples (inclusão, exclusão, alteração, consulta simples em banco de dados): a) até 5 campos; b) 6 a 15 campos; c) + de 15 campos.
	2. Complexa (com validação, referências cruzadas, campos calculados, buscas inclusive com filtros): a) até 5 campos; b) 6 a 15 campos; c) + de 15 campos.
Geração de relatórios	1. Simples (listagem de cadastros): a) 1 tabela; b) 2 a 5 tabelas; c) + de 5 tabelas.
	2. Complexo (campos calculados, <i>joins</i> , <i>unions</i> , <i>subselects</i> , filtros, agrupamentos, ordenação, referência cruzada, gráficos): a) 1 tabela; b) 2 a 5 tabelas; c) + de 5 tabelas.
	3. Com geração de arquivos externos para exportação de dados: a) 1 tabela; b) 2 a 5 tabelas; c) + de 5 tabelas.
Interação com periféricos	1. Acesso a sistemas externos (validação em banco de dados de outro sistema, interface/protocolo de comunicação): 1 tabela; b) 2 a 5 tabelas; c) + de 5 tabelas.
	2. Interação com dispositivos como leitores de biometria, código de barras, obtenção de dados de dispositivos (sensores): 1 tabela; b) 2 a 5 tabelas; c) + de 5 tabelas.
	3. Envio de comandos para periféricos (sensores, atuadores): 1 tabela; b) 2 a 5 tabelas; c) + de 5 tabelas.
Processamento (transações)	1. Validação de acesso (<i>login</i>)
	2. Cálculos com ou sem consulta a banco de dados, conversão de dados: a) 1 cálculo; b) 2 a 5 cálculos; c) + de 5 cálculos
	3. Cálculos para atender a legislação: Conversão de dados: a) 1 cálculo; b) 2 a 5 cálculos; c) + de 5 cálculos.
	4. Processamento interno complexo, lógico e matemático extensivo: a) 1 processamento; b) 2 a 5 processamentos; c) + de 5 processamentos.
	5. Segurança como SSL ou código para criptografia.
	6. Requisitos não funcionais relevantes: restrições de desempenho, restrições de memória, portabilidade, confiabilidade das operações realizadas, integração com sistemas existentes: a) 1 requisito; b) 2 a 4 requisitos; c) 5 a 8 requisitos; d) + de 8 requisitos.
	7. Bancos de dados distintos com peculiaridades em sentenças SQL para cada banco: a) 1 tabela; b) 2 a 5 tabelas; c) + de 5 tabelas.

Fonte: Borsoi *et al.*, (2010) *apud* Borsoi *et al.* (2011)

- **Raciocínio baseado em casos (CBR)**

Segundo Mendes (2008), CBR usa a suposição de que "problemas semelhantes provê soluções semelhantes". Ele fornece estimativas, comparando as características do projeto atual a ser estimado, contra uma biblioteca de informações históricas de projetos concluídos com esforço conhecido (caso base).

Mendes (2008) propõe uma sequência de passos utilizando o CBR, a fim de obter uma estimativa de esforço, os grupos de processos são iguais ao CART da Figura 2.4. A seguir será descrito os passos para realização da estimativa:

- ✓ **Passo 1).** Os preditores estimados de tamanho e custo relativos a um novo projeto são usados como entrada para recuperar projetos semelhantes a partir do caso base, por que é conhecido o esforço real.
- ✓ **Passo 2).** Utilizando os dados do passo 1 uma ferramenta CBR adequada recupera projetos semelhantes para projetar e classifica esses projetos similares em ordem crescente de semelhança, ou seja, de "mais semelhante" para "menos semelhante".
- ✓ **Passo 3).** Uma ferramenta adequada CBR calcula esforço estimado para o projeto.

As sequências de passos descritos são semelhantes aos empregados na obtenção do esforço estimado usando estimativas baseadas por especialistas. Ambos requerem que as características de um novo projeto devem ser conhecidas, a fim de recuperar projetos anteriores finalizados semelhantes. Uma vez que informações do esforço de projetos semelhantes são recuperados, o esforço do novo projeto poderá ser estimado.

2.5 PRINCIPAIS MÉTODOS DE ESTIMATIVA TRADICIONAIS EXISTENTES NA LITERATURA

De Souza e de Aquino (2014) realizaram uma revisão na literatura para identificar os principais métodos de estimativa tradicionais que abordam as características do sistema em geral. Os métodos identificados na pesquisa podem ser vistos na Tabela 2.2.

A Tabela 2.2 apresenta, em ordem cronológica os principais métodos de estimativa, mostrando o ano de criação, o nome do método e o autor do mesmo. Para cada um dos métodos identificados foi elaborado um resumo da descrição e das características, conforme pode ser visto a seguir:

Function Point Analysis (FPA) é um método de medição de tamanho em pontos de função, baseado no que o usuário percebe, levando em consideração a funcionalidade implementada. Este método é independente de tecnologia e foi projetado para estimar sistemas de informação empresariais. Possui como características: estimar a funcionalidade solicitada pelo usuário, calculando seu tamanho e custo; estimar projetos de desenvolvimento e manutenção de software, independente da tecnologia utilizada; estimar a funcionalidade recebida pelo usuário, após o desenvolvimento, de forma a verificar se seu tamanho e custo estão de acordo com estimado (Albrecht, 1999).

Tabela 2.2: Principais Métodos de Estimativas

Ano	Método	Autor
1979	Function Point Analysis (FPA)	Albrecht (Oligny <i>et al.</i> , 1999)
1988	Mark II FPA	Charles Symons (Symons, 2001)
1990	Netherlands Software Metrics Users Association (NESMA) FPA	The Netherlands Software Metrics Users Association (Engelhart, 2001)
1999	Common Software Measurement International Consortium (COSMIC) FFP	Common Software Measurement International Consortium (COSMIC) (Cosmic, 2003)
2004	Finnish Software Metrics Association FSM	The Finnish Software Metrics Association (FISMA) (Forselius, 2004)

Fonte: De Souza e de Aquino (2014)

Mark II FPA é um método certificado pelo ISSO como padrão internacional, projetado para estimar sistemas de informação empresariais. Foi criado para tentar sanar as deficiências identificadas no método FPA. Possui como características: estimar o processamento de informações, no qual considera um software como um conjunto de operações lógicas e pondera o tamanho funcional do mesmo, contando os tipos de entrada de dados, tipos de entidade de dados referenciados e tipos de saída de elementos de dados para cada transação lógica (Symons, 2001).

Netherlands Software Metrics Users Association (NESMA) é um método certificado pela ISO como padrão internacional, projetado para contar os pontos de função do software, semelhante ao método FPA. Possui como características: Entrada Externa, Saída Externa, Consulta Externa, Arquivo Lógico Interno e Arquivo de interface Externa. A diferença é que NESMA FPA dá orientações mais concretas, devido a seus três tipos de contagem: detalhada, estimada e indicativa.

Common Software Measurement International Consortium (COSMIC) FFP foi projetado baseado na FPA, porém direcionado para aplicações em tempo real e softwares multi-plataforma. Possui como características: estimar o esforço de desenvolvimento, evolução de qualidade de software, comparação de sistemas especificados em diferentes linguagens, em termos de produtividade e manutenção de custos, levando em consideração conceitos como, requisitos funcionais dos usuários, usuário do software, camadas e fronteiras.

Finnish Software Metrics Association FSM ou simplesmente FISMA é um método de medição de tamanho de software. Possui como características: é orientada a serviços ao invés de processos, ou seja, todos os serviços são identificados para o cálculo do tamanho funcional do software (Forselius, 2004).

Em uma primeira análise percebe-se que os principais métodos existentes não foram projetados para considerar os requisitos de aplicações móveis. Na verdade, a própria criação da maioria destes antecede o surgimento dos dispositivos móveis, como conhecemos hoje. Este fato sugere que o uso destes métodos para estimar o esforço do desenvolvimento de projetos envolvendo sistemas ou aplicações para dispositivos móveis ocorreria uma possível falha em quantificar a complexidade de certas características e conseqüentemente não produziriam estimativas adequadas.

2.6 CONSIDERAÇÕES SOBRE O CAPÍTULO

Este capítulo teve como objetivo apresentar conceitos de estimativa de projetos de software, processo de estimativa genérico, categorias de estimativa de esforço e principais métodos existentes na literatura. Além disso, o próximo capítulo mostra um estudo qualitativo sobre quais são os fatores preditores de esforço que os especialistas e praticantes apontam hoje em dia na indústria.

CAPÍTULO 3 – ESTUDO QUALITATIVO SOBRE ESTIMATIVA DE ESFORÇO NO DESENVOLVIMENTO DE APLICATIVOS MÓVEIS

Este capítulo apresenta um estudo qualitativo sobre quais fatores contribuem para uma melhor estimativa de esforço com base na opinião de profissionais diretamente envolvidos na estimativa de esforço em projetos de aplicativos móveis. Este capítulo detalha o método utilizado, análise dos dados, ameaças à validade, resultados encontrados e conclusão deste estudo.

3.1 INTRODUÇÃO

Este capítulo apresenta um estudo qualitativo com o objetivo de compreender quais os fatores que contribuem para uma melhor estimativa de esforço com base na opinião de profissionais diretamente envolvidos na estimativa de esforço em projetos de aplicativos móveis. Este estudo foi realizado antes do mapeamento sistemático para evitar o viés de aprendizado das pesquisadoras. Porém, após o término do Mapeamento Sistemático foi revisada a etapa axial do qualitativo com intuito de reorganizar o agrupamento dos fatores seguindo os conceitos das subcategorias levantadas no mapeamento sistemático. Como por exemplo, a subcategoria de tecnologia do qualitativo era uma subcategoria do projeto, e para seguir os conceitos das categorias do mapeamento sistemático, a tecnologia passou a ser categoria, então foi reordenado as categorias e fatores.

Os dados utilizados nessa pesquisa foram coletados por meio de entrevistas semiestruturadas com oito profissionais de cinco diferentes empresas de desenvolvimento de aplicativos móveis em Manaus, Amazonas (Brasil). Além disso, foram empregados procedimentos baseados no método de Grounded Theory (GT) propostos por Strauss e Corbin (1998) para auxiliar na análise dos dados. As subseções a seguir detalham as atividades relativas à coleta de dados, análise dos dados, ameaças a validade, resultados e considerações finais.

3.2 COLETA DE DADOS

Participaram desta coleta de dados, oito especialistas que trabalham em cinco diferentes empresas que desenvolvem aplicativos móveis em Manaus. A Tabela 3.1 mostra o tempo de experiência dos especialistas em estimar projetos de aplicativos móveis.

Tabela 3.1: Experiências dos Entrevistados

Quantidade de Especialista	Anos de Experiência
P1	1 anos
P6	3 anos
P2, P3, P4, P5	5 anos
P7, P8	10 anos

As entrevistas semiestruturadas foram realizadas com o objetivo de despertar o conhecimento tácito dos especialistas, como eles estimam o esforço para novos projetos de aplicativos móveis tendo como foco os fatores preditores de esforço que são importantes, como esses fatores estão inter-relacionados e também na perspectiva de haver uma influência positiva ou negativa sobre um fator da estimativa de esforço. Essas entrevistas foram realizadas no período de novembro a dezembro de 2015.

Os passos seguintes descrevem como foi realizada a coleta de dados:

- Definição de objetivos e seleção de técnicas a serem utilizadas como parte desta coleta;
- Seleção de profissionais que têm experiência em estimar o esforço para projetos de aplicativos móveis;
- Realização de entrevistas com especialistas através do uso de perguntas semiestruturadas e abertas. Todas as entrevistas foram gravadas com a permissão do entrevistado;
- Após cada entrevista, foram realizadas as seguintes atividades:
 - Transcrição das entrevistas (conteúdo gravado);
 - Codificação dos dados a partir dos transcritos, que compreende: (a) a identificação de códigos relevantes no contexto de estimativa de esforço de Aplicativos Móveis, (b) a análise das relações entre os códigos, e (c) a identificação de categorias para agrupar os códigos encontrados anteriormente;
- Análise dos dados na qual foram listadas as categorias identificadas que podem influenciar a estimativa de esforço de projetos de aplicativos móveis;
- Síntese das categorias em uma lista de fatores preditores de esforço, detalhes de cada categoria a partir dos dados recolhidos;
- Seleção de trechos representativos das entrevistas (citações ou *quotations*) para cada categoria;
- Interpretação dos dados - análise de cada fator de influência;

Antes de realizar as entrevistas, enviaram-se e-mails convidando os entrevistados a participarem dessa pesquisa e também revelando que as entrevistas seriam gravadas para fins de pesquisa. Cada entrevista variou entre 15 a 46 minutos. Esta diferença de duração está relacionada com as características individuais de cada participante (por exemplo, levando mais tempo para responder perguntas e fazer detalhamentos).

A fim de atender às necessidades éticas, foi elaborado um Termo de Consentimento Livre e Esclarecido (TCLE) para notificá-los sobre o tema, os procedimentos de investigação, confidencialidade e pedir o consentimento dos especialistas entrevistados. Este documento consta no APÊNDICE A.

Durante as entrevistas, os especialistas descreveram suas experiências para a resolução de problemas e como estimavam o esforço para projetos de aplicativos móveis. Questões semiestruturadas e abertas foram utilizadas para permitir uma análise detalhada da investigação sobre o contexto em que os entrevistados foram imersos.

A Tabela 3.2 mostra as perguntas realizadas nas entrevistas. Essas perguntas foram uma adaptação do roteiro de Matos *et al.* (2013), pois eles investigaram como é realizada a estimativa de esforço em projetos Web na indústria. Tais questões também poderiam levar a mais perguntas dependendo das respostas dos entrevistados. Todas as entrevistas foram realizadas de forma a incentivar os participantes a falarem livremente no intuito de responder às perguntas.

Cada entrevista teve a sua transcrição feita integralmente e através das mesmas, foi analisado cuidadosamente o conhecimento dos entrevistados. Deve-se enfatizar que todas as observações dos dados foram obtidas exclusivamente das entrevistas e foram omitidos os nomes de empresas citadas durante a transcrição das entrevistas, a fim de preservar a identidade dos participantes e das empresas. A análise qualitativa foi feita com base em procedimentos de GT, explicados na subseção seguinte.

3.3 ANÁLISE DE DADOS

A análise qualitativa, realizada neste trabalho, é baseada em procedimentos do método Grounded Theory (GT) proposto por Strauss e Corbin (1998), que usa um conjunto de dados para fazer procedimentos sistemáticos que envolvem coleta e análise para gerar, preparar e validar teorias substantivas sobre os fenômenos essencialmente sociais (Glaser e Strauss, 1967).

Tabela 3.2: Questões usadas nas entrevistas

Questões
1) Com base em quê você estima o esforço? (Exemplos: experiência dos envolvidos no projeto, escopo da aplicação). Existe alguma métrica de estimativa que é adotada pela empresa?
2) Quais as informações que você sempre pede a um cliente, a fim de compreender os Requisitos/ as Funcionalidades do aplicativo móvel?
3) Quais os fatores que você considera quando estima o esforço para projetos de aplicativos móveis? (Exemplos: Interface gráfica, Constante Interrupções, Limitação de energia).
4) Quais etapas você segue durante um processo de estimativa de esforço?
5) Em relação aos clientes, há algum conjunto de fatores relacionados que podem afetar uma estimativa de esforço? Quais são esses fatores? (Exemplos como Cliente não sabe o que quer, Clientes que entendem as regras de negócio do projeto, Indisponibilidade do cliente).
6) Em relação à empresa, há algum conjunto de fatores relacionados que pode afetar uma estimativa de esforço? Quais são esses fatores?
7) Em relação às pessoas envolvidas no projeto, existe algum fator, ou conjunto de fatores, relacionados, que sejam importantes de considerar durante a estimativa de esforço? Que fator/fatores é/são esse(s)? (Exemplo, experiência, dificuldade técnica do time, colaboração do time).
8) Em relação aos projetos passados, há algum conjunto de fatores que podem afetar uma estimativa de esforço? Quais são esses fatores?
9) Você pergunta ao cliente sobre as restrições / riscos que podem afetar a estimativa de esforço? Como você usa esses riscos na estimativa de esforço?
10) Todas as aplicações móveis que você desenvolve tem um cliente associado? Quando não tem um cliente “contratante”, a forma de estimar muda? O que você considera nesse caso?
11) Uma vez preparada à estimativa de esforço, ela é revisitada durante a duração do projeto? Se afirmativo, então que informações são consideradas para que a estimativa seja atualizada? Quem fornece essas informações?

Grounded Theory é baseado na ideia de codificação que é o processo de análise dos dados (Strauss e Corbin, 1998). Conceitos (códigos) e categorias são identificados durante a fase de codificação. O conceito é um fenômeno que tem o interesse do pesquisador; abstrai um evento, um objeto, uma ação ou interação que tem um significado para o pesquisador (Strauss e Corbin, 1998). As categorias são aglomeradas de conceitos unidos em um maior grau de abstração.

O processo de codificação pode ser dividido em três etapas: aberta, axial e seletiva. A codificação aberta envolve a separação, análise, comparação, conceituação e a categorização dos dados. Os códigos criados podem ser classificados como: (a) de codificação de primeira ordem, que é diretamente associada às citações (referidos como open coding); e (b) códigos abstratos ou teóricos que são associados a outros códigos sem necessariamente estar ligado a

alguma citação. Finalmente, durante a codificação aberta, as categorias são criadas para agrupar os códigos e reduzir o número de unidades que o pesquisador vai trabalhar.

Após a identificação das categorias conceituais pela codificação aberta, passou-se para a codificação axial que examina as relações entre as categorias. As relações entre os códigos foram definidas pela autora desta pesquisa. De acordo com Strauss e Corbin (1998), estas relações formam o que os autores chamam de um paradigma: as condições causais, novas condições, consequências e estratégias de ação / interações.

Por fim, a codificação seletiva realiza todos os refinamentos do processo, identificando o núcleo central com a qual todas as categorias estão relacionadas. Segundo Matos *et al.* (2013), a categoria central deve ser capaz de integrar todas as outras categorias e para expressar a essência do processo social. Não foi eleita uma categoria central nesta pesquisa, porque segundo Strauss e Corbin (1998), uma regra da Grounded Theory é a circularidade entre as fases de coleta e análise até a saturação teórica for atingida. Portanto, decidiu-se adiar a fase de codificação seletiva, por esta razão, logo, não se pode alegar que foi aplicado o método GT completo, mas somente alguns procedimentos específicos.

Logo após a transcrição de cada entrevista, começou-se o processo de codificação. Enquanto eram analisados os dados contidos na transcrição da entrevista, os códigos associados com partes textuais foram criados, conforme mostrado na Figura 3.1. Os códigos relacionados com as citações em cada entrevista transcrita foram revistos com mais outra pesquisadora que verificou esses códigos e categorias, a fim para auditar o processo de codificação. Isso foi feito, a fim de mitigar o viés, eventualmente causado pela participação de uma única pesquisadora no processo de codificação.

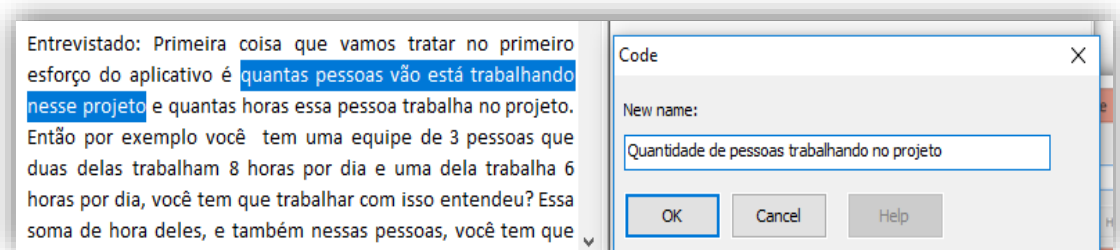


Figura 3.1: Exemplo de codificação aberta - Citação e Código

Depois de realizar a codificação aberta, iniciou-se a fase da codificação axial, na qual se criaram os códigos de relacionamento. Foram identificados dois principais grupos de fatores de estimativa de esforço para projetos de aplicativos móveis: Métricas de Tamanho e Fatores de Custo. As métricas de tamanho não possuem subcategorias, quanto que os Fatores

de Custo foram organizados em subcategorias como Projetos de Aplicativos Móveis, Produto, Empresa, Equipe de desenvolvimento, Clientes e Tecnologia. Cada subcategoria representa diferentes dimensões que têm impacto na estimativa de esforço de aplicativos móveis.

A Figura 3.2 apresenta o grupo de códigos relacionados a métricas de tamanho do projeto de aplicativos móveis. Esses códigos apontam fatores preditores de esforço, tais como Documento de User Interface (IU), Documento User Experience (UX), Quantidade de requisitos, Quantidade de dados, Quantidade de Telas.

A Figura 3.3 mostra o grupo de categorias classificadas como fatores de custo que estão associados com o esforço necessário para desenvolver um aplicativo móvel. Como dito anteriormente, esta categoria possui as seguintes categorias: Projetos de Aplicativos Móveis, Produto, Empresa, Equipe de desenvolvimento, Clientes e Tecnologia. Todos os códigos deste estudo estão no relatório técnico (De Souza e Conte, 2018).

A seção 3.5, mostra os resultados de todas as categorias de fatores preditores de esforço em projetos de aplicativos móveis e as variações do processo de estimativa de esforço nas empresas que desenvolvem aplicativos móveis. Mas, esses resultados estão sujeitos a ameaças a validade descritas na seção a seguir.

3.4 AMEAÇAS À VALIDADE

Neste estudo houve algumas ameaças que podem afetar a validade deste estudo. A principal ameaça é a generalização dos resultados para todos os profissionais que estimam projetos de aplicativos móveis. Estudos qualitativos não podem usar nada semelhante a um argumento estatístico para a generalização dos resultados (Oliveira *et al.*, 2016). Mitigou-se esse problema ao entrevistar 8 profissionais que trabalham com estimativa de projetos de aplicativos móveis em 5 diferentes empresas representativas da região norte do Brasil.

Outra ameaça são as percepções dos entrevistados que podem ser tendenciosas em relação a suas próprias crenças. Essas crenças podem causar algumas distorções ao interpretar a realidade. Para reduzir essa ameaça, os profissionais entrevistados foram aqueles que têm entre 1 à 10 anos de experiência com estimativas de projetos de aplicativos móveis.

Outra ameaça a validade dos resultados é a subjetividade da classificação dos dados, uma vez que a análise foi realizada de forma qualitativa pela primeira autora. Para reduzir essa ameaça, o processo de análise dos dados coletados foi realizado junto com outra pesquisadora que analisou todos os resultados intermediários (coleta e análise dos dados).

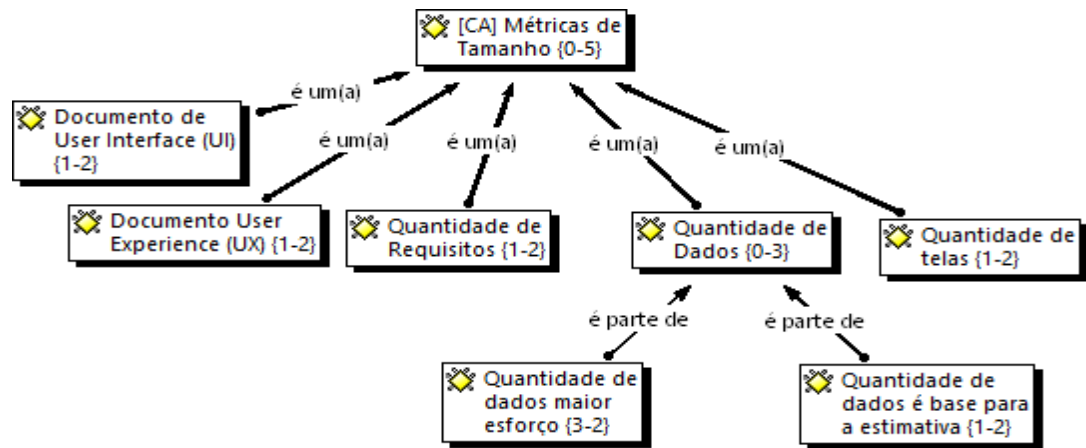


Figura 3.2: Relações entre as Métricas de Tamanho

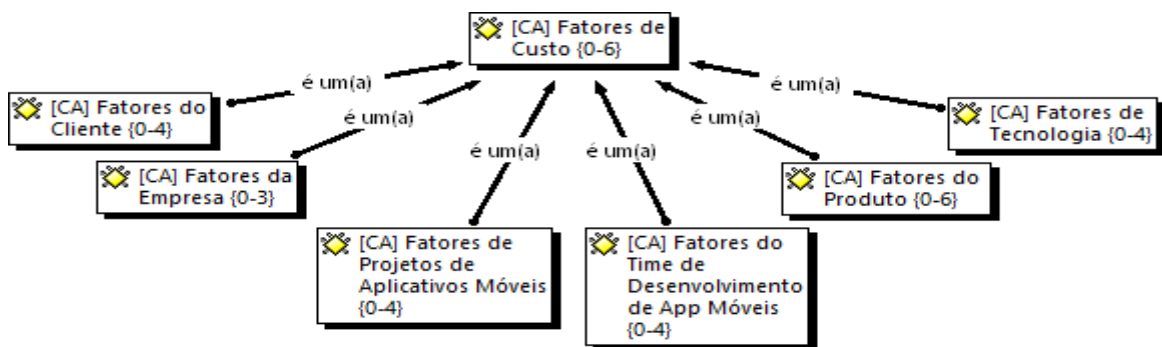


Figura 3.3: Relações entre os Fatores de Custo

3.5 RESULTADOS

Os resultados da pesquisa qualitativa resultaram na identificação de 54 fatores que afetam a estimativa e variações do processo de estimativa de esforço nas empresas que desenvolvem aplicativos móveis.

Os preditores de esforço são resultados dos códigos identificados e descritos no relatório técnico em De Souza e Conte (2018). Os fatores foram organizados em dois grupos: Métricas de Tamanho e Fatores de Custo. Os Fatores de Custo foram organizados em Categorias: Projetos de Aplicativos Móveis, Produto, Empresa, Tecnologia, Equipe de desenvolvimento e Clientes. Todos os fatores identificados são exibidos na Tabela 3.3.

Uma descrição detalhada dos fatores e suas relações são apresentadas nas subseções a seguir. A fim de fornecer uma descrição detalhada foram mostradas algumas citações dos entrevistados que ajudaram ao longo do processo a identificar esses fatores. A Seção 3.5.2 mostra a variação do processo de estimativa de esforço nas empresas que foram representadas pelos entrevistados.

Tabela 3.3: Preditores de Esforço em Projetos de Aplicativos Móveis

Categorias	Subcategoria	Fator	Descrição	Referência
Métricas de Tamanho	-	Quantidade de Dados	Quantidade de Dados que é necessário para manter o Aplicativo.	P1, P3, P4
		Quantidade de Requisitos	Quantidade de requisitos do projeto.	P7
		User Stories	Se utiliza ou não User Store para estimar o projeto.	P2, P6
		Telas	Indica quantidade ou avalia cada esboço de tela como fácil, médio ou complexo.	P1, P3
		Nível de Documentação do Projeto	Não tem, tem pouco detalhado, tem mais detalhado.	P2, P8
		Documento de User Interface (UI)	Se utiliza ou não Documento de User Interface (UI) para estimar o projeto.	P8
		Documento de User Experience (UX)	Se utiliza ou não Documento de User Experience (UX) para estimar o projeto.	P8
Fatores de Custo	Produto: Dispositivo Móvel	Tipo de Conectividade do Dispositivo Móvel	Os tipos de conectividade como 3G e Wifi interfere no preparo da infraestrutura.	P1
		Sistema Operacional do Dispositivo Móvel	Leva-se em consideração a Versão do sistema operacional e o sistema operacional: Android e IOS.	P1, P6, P7, P8
		Tamanho das Telas do Dispositivo Móvel	Quais de dispositivos móveis são utilizados que podem variar o tamanho de telas que o aplicativo será executado.	P2, P6
		Limitação de Energia do Dispositivo Móvel	Dependendo dos tipos de aplicativo há uma preocupação na limitação da energia do dispositivo. Consiste na existência dessa preocupação ou não.	P1, P3, P4
		Interrupção de chamadas	Consiste na existência dessa preocupação na interrupção	P2, P3

Categorias	Subcategoria	Fator	Descrição	Referência
			de chamadas ou não.	
	Produto: Aplicativo Móvel	Modo de Apresentação do Aplicativo	Se há um tratamento ou não diferenciado para a tela que está na horizontal ou na vertical do dispositivo móvel.	P1, P6, P7, P8
		Targets (Quantidade de Dispositivos que a aplicação precisa executar)	Consiste na quantidade de dispositivos que a aplicação será executada.	P2, P6, P7
		Nível de Complexidade da Interface Gráfica	Tem aplicativos que demandam animações, efeitos diferentes e tem preocupações com a usabilidade, com isso a complexidade da interface gráfica pode variar de baixa, média ou alta.	P4, P5
		Integração com as redes sociais	Se o aplicativo tem integração com redes sociais ou não.	P3, P4, P5
		Requisição para a Internet	Se for uma aplicação cliente-servidor, a própria requisição que espera um retorno, pode ser sucesso ou pode ser falha tem que tomar o cuidado de estimar os caminhos alternativos, quando é uma aplicação cliente servidor e tem que pensar nas respostas que o servidor deu e quais serão as possíveis respostas que o servidor vai dar pra você tratar.	P5
		Tipo de Comunicação com o Servidor	Essa comunicação com o servidor pode ser Json ou XML.	P4, P5
		Dependência com outro Sistema	Se existe uma dependência com outro sistema do cliente ou outra API ou não existe dependência.	P5
	Cliente	Nível de Competência do Cliente	Envolve conhecer do negócio e saber passar as	P1, P2, P4,

Categorias	Subcategoria	Fator	Descrição	Referência
			informações.	P6, P8
		Nível de Comunicação com o Cliente	Envolve a comunicação dos responsáveis pelo projeto com o cliente que pode variar de baixo, médio ou alta comunicação.	P1, P2, P3, P6
		Nível de Participação do Cliente	Se o cliente valida as especificações do projeto, se o cliente tem ciência dos riscos do projeto, se o cliente participa da reunião de concepção do produto, esses são exemplos de participação. A participação pode variar de baixa, média ou alta.	P1, P2, P3, P4, P5, P6, P7, P8
		Autonomia do Cliente na Tomada de Decisão	Se o cliente tem autonomia para tomar uma decisão no projeto ou não.	P7
	Empresa	Nível de Disponibilidade da Internet da Empresa	Consiste no nível de estabilidade/disponibilidade de acesso à internet da empresa que pode variar de nunca estável, as vezes estável e estável.	P3, P7
		Nível de Disponibilidade do Dispositivos para Teste	Se tem ou não disponibilidade dos dispositivos para teste do aplicativo.	P7
		Nível de Infraestrutura Ferramental da Empresa	Disponibilidade de salas para trabalho e ferramental que pode variar de sempre disponível, raramente, às vezes e nunca.	P8
		Disponibilidade da Infraestrutura para falar com o Cliente	Se existe ou não uma infraestrutura para falar com o cliente	P6, P8
	Equipe de Desenvolvimento	Nível de Experiência com a Tecnologia	O nível de experiência com a tecnologia que o projeto precisa pode variar de nenhuma, baixa, média ou alta.	P1, P2, P3, P4, P5, P6, P7, P8

Categorias	Subcategoria	Fator	Descrição	Referência
		Conhecimento do Problema	Se a equipe tem conhecimento do problema que do cliente ou não.	P2
		Nível de Experiência no Desenvolvimento em Projetos Anteriores	O nível de experiência no desenvolvimento em projetos anteriores pode variar de nenhuma, baixa, média ou alta.	P1, P2, P3, P5, P7
		Nível de Comunicação entre a Equipe	O nível de comunicação pode variar de baixo, médio ou alto.	P2
		Nível de Conhecimento da Competência da Equipe	Quando se conhece a competência da equipe que está participando do projeto. Esse conhecimento da competência da equipe pode variar de baixo, médio ou alto.	P1, P2
		Nível de Engajamento da Equipe	Envolve o comportamento e foco dos envolvidos no projeto que pode variar em baixo, médio ou alto nível de engajamento com o projeto.	P2, P4, P6
		Quantidade de Horas Dedicadas de cada Pessoa do projeto	Consiste na quantidade de horas dedicadas de cada pessoa da equipe do projeto.	P4
		Quantidade de Pessoas Trabalhando no Projeto	Consiste na quantidade de pessoas trabalhando no projeto.	P2, P4, P6
		Curva de Aprendizagem da Equipe	Consiste no tempo necessário para a equipe aprender algum recurso, contexto do projeto ou tecnologia necessária para o projeto.	P1, P6
	Projeto: Projeto de Aplicativos Móveis	Nível de Complexidade do Contexto do Aplicativo	Se o aplicativo precisa que seja estudado o contexto que o aplicativo será inserido. Exemplo: bolsa de valores e bancos. Esse nível de complexidade pode variar de baixo, médio ou alto.	P2

Categorias	Subcategoria	Fator	Descrição	Referência
		Nível de Confidencialidade do Projeto	Se o projeto exige uma confidencialidade que pode variar de baixa, média ou alta.	P4
		Mudanças	Se existe um controle das mudanças que são realizadas no projeto.	P6
		Riscos	Se o projeto possui clareza dos riscos existentes ou não.	P2
		Migração de Projeto	Se o projeto está preparado para uma possível migração de ferramenta ou não.	P1
		Tamanho do Escopo	Varia de acordo com as métricas de tamanho.	P2, P6, P7
		Tipo do Escopo	Se o escopo é aberto (não definidos os requisitos) ou fechado (todos os requisitos são definidos).	P6
	Projeto: Fatores de Desenvolvimento	Flexibilidade da Metodologia de Desenvolvimento	Se existe Flexibilidade da Metodologia de Desenvolvimento ou não.	P4
		Dependência entre Tarefas	Se existe alguma tarefa dependente de outra tarefa ou não.	P5
		APIs do Cliente	Se é utilizado alguma API do Cliente ou não.	P4
		Nível de Estabilidade dos Requisitos	Consiste em determinar o grau em que os requisitos estão estáveis. Pode variar de baixa, média ou alta estabilidade.	P1, P7
		Nível de Definição dos Requisitos	O grau de definição dos requisitos pode variar de baixo (requisitos não estão definidos), médio (uma parte dos requisitos estão definidos) ou alto (todos os requisitos estão definidos).	P5
		Variação de formas de realizar a mesma funcionalidade	Se existe uma variação de formas de realizar a mesma	P4

Categories	Subcategoria	Fator	Descrição	Referência
			funcionalidade ou não. Exemplo: a forma de pagamento que pode ser por boleto, cartão de crédito ou paypal.	
		Premissa do Projeto	Se é realizado o levantamento das premissas do projeto ou não.	P8
	Projeto: Modo de Como Fazer a Estimativa	Identificação dos MVPs	Se é identificado o Produto Mínimo Viável para o cliente ou não.	P7, P8
		Identificação de Personas	Se é identificadas as Personas ou não.	P7
		Identificação da Visão do Produto	Se é identificada a visão do produto ou não.	P7
		Técnicas de Concepção do Produto	Se são utilizadas técnicas de concepção do produto ou não.	P7

3.5.1 Preditores de Esforço

3.5.1.1 Métricas de Tamanho

Foram identificados 7 fatores relacionados com as métricas de tamanho dos projetos de aplicativos móveis. A seguir estão descritos alguns desses fatores:

- **Documento de User Interface (UI), Documento User Experience (UX):** são documentos referente a interface do usuário muito relevantes. Veja a citação do entrevistado 6.
“a gente precisa primeiro ver a questão da UX, ver a coisa UI, ai montar a arquitetura, que regras de negócio vai ter, ai cada atividade dessa a gente vai colocando as horinhas lá pra saber no final.” – Entrevistado 6.
- **Quantidade de Requisitos, Quantidade de Dados, Telas, Nível de Documentação do Projeto, User Stories** são fatores preditores de esforço, pois são usados como indicadores de esforço na hora da estimativa.
“o que mais pesava era a quantidade de dados que o cliente iria querer que tivesse no aplicativo” – Entrevistado 1.

3.5.1.2 Projetos de Aplicativos Móveis

Foram encontrados 18 preditores e organizados em 3 subcategorias relacionados com projetos de aplicativos móveis. A seguir estão descritos as subcategorias e seus respectivos fatores:

- **Projeto de Aplicativos Móveis:** é um dos primeiros preditores a ser levantados na hora de estimar, pois o **Tamanho do escopo**, o **Tipo do Escopo** que pode ser de **Tempo Aberto** (primeiro faz a concepção e depois fecha o orçamento baseada na concepção, ou seja, negociações são feitas durante o projeto) ou **Tempo Fechado** (neste caso, o tempo já está definido e a negociação consiste em o que entregar no prazo estabelecido).
“Qual é o escopo da aplicação? Tem que conhecer todos os detalhes dela, para enfim dizer eu vou precisar de x tempo pra desenvolver por causa disso, disso e disso” – Entrevistado 2.
- **Fatores do Desenvolvimento:** ocorre depois do início do projeto e são preditores relacionados ao desenvolvimento do projeto como por exemplo o **Nível de Definição dos Requisitos** (se o entendimento do problema e o contexto que será inserido o aplicativo está claro).
“Mas um dos principais é quando o requisito não está bem definido, então a gente pensa a gente estima uma coisa e na verdade é outra coisa. Então vai causar um conflito aí no que vai ser realizado e o tempo.” – Entrevistado 5.
- **Modo de Fazer a Estimativa:** preditores relacionados a como é realizada a estimativa. Essas estimativas são feitas em grupo, por atividades e por horas. Por exemplo, a **Identificação da Visão do Produto** é um preditor muito importante, pois dependendo da necessidade do cliente pode demandar mais esforço para gerar um MVP.
“O que é o sistema e o quê que de valor ele quer , o quê que vai agregar de valor pra ele no desenvolvimento do sistema entendeu?! ... Cliente vai escolher qual é a funcionalizada que realmente vai trazer valor para o sistema.” – Entrevistado 6.

3.5.1.3 Produto

Foram identificados 12 fatores do produto de projetos de aplicativos móveis e foram organizados em dois grupos: Aplicativos móveis e Dispositivos Móveis. A seguir detalha esses dois grupos :

- **Limitação de energia do dispositivo móvel:** Caso for um aplicativo de jogo consumirá mais energia e aí é importante tentar minimizar o gasto excessivo de energia do dispositivo móvel:
“Caso dessa questão dessa questão de limitação de energia não é muito a preocupação fazendo um aplicativo normal não, mas é mais quando você desenvolve algum tipo de jogo alguma coisa assim” – Entrevistado 4.
- **Targets:** é a quantidade de dispositivos que a aplicação precisa rodar.
“Tem que saber quais são os targets, quais são os dispositivos em que o aplicativo tem que rodar, isso é vital ... tu tens lá um indicador que diz quantos por centos, as dimensões e os tamanhos de celulares que estão em uso hoje no mercado.” – Entrevistado 8.
- **Sistema Operacional:** esse fator impacta muito a estimativa, pois para cada sistema operacional é uma equipe e experiência do usuário diferente.
“Se for um aplicativo que tem que rodar em IOS e Android aí eu tenho que ter dois times. Provavelmente um especialista em IOS e outro em Android. Pra poder desenvolver o aplicativo com a usabilidade de cada um desses dispositivos.” – Entrevistado 7.

3.5.1.4 Tecnologia

Foram identificados 2 fatores que estão relacionados a tecnologia dos dispositivos móveis. A seguir estão descritos esses fatores:

- **Tipo de Tecnologia:** Consiste se a equipe tem experiência com a tecnologia ou é uma tecnologia nova. **Tecnologia usada no projeto** ou **Nova Tecnologia de Desenvolvimento** impactam na estimativa, pois ao mudar a tecnologia terá o processo imigratório do projeto, envolverá a curva de aprendizado do time e dentre outros fatores.
“Quando tiver uma mudança de ferramenta, as coisas se eu ainda não soubesse mexer direito e iria demandar um pouco de tempo a aprender a mexer com a nova ferramenta e importar os projetos pra nova ferramenta” – Entrevistado 1.

3.5.1.5 Empresa

Foram encontrados 7 fatores relacionados com o ambiente de desenvolvimento de aplicativos móveis, empresa que desenvolve aplicativos móveis. A seguir está descrito um dos fatores:

- **Fatores de Infraestrutura** – muitos dos entrevistados disseram que esses fatores afetam a estimativa de esforço de forma direta e indiretamente, mas um entrevistado disse que a Infraestrutura da Empresa não afeta a estimativa (veja a citação do entrevistado 2). Exemplos de fatores são **infraestrutura de internet** que depende de internet para efetuar o trabalho para preparar o ambiente de desenvolvimento.

“Questão da infraestrutura acho que não chega a afetar não, não seria um fator que poderia influenciar muito não” – Entrevistado 2.

“O cliente quer uma tecnologia que precisa baixar um SDK que é 5GB... pra poder começar a pensar e analisar e estimar ...isso leva um tempo se tivesse uma estrutura estável, ai sim se torna mais rápido.” – Entrevistado 6.

3.5.1.6 Equipe de Desenvolvimento de Aplicativos Móveis

Foram encontrados 15 fatores e organizados em 2 subcategorias relacionados com a equipe que desenvolve aplicativos móveis. Esses fatores influenciam a estimativa de esforço, a seguir está descrito um dos fatores:

- **Experiência do Time:** é algo que interfere demais na hora da estimativa, pois saber quanto tempo se vai gastar para fazer o projeto e como fazer o projeto é imprescindível.

“O cara que trabalha em Android já tem 3 anos trabalhando com Android, se eu pegar um novo sistema com a estimativa dele, vai ser rápida em relação se for pegar por exemplo IOS, então conta bastante” – Entrevistado 7.

3.5.1.7 Clientes

Foram encontrados 5 fatores relacionados com os Clientes de aplicativos móveis. Esses fatores influenciam a estimativa de esforço de forma a facilitar e também a dificultar esse processo, a seguir está descrito um dos fatores:

- **Nível de Participação do Cliente:** O cliente estando presente no dia a dia do desenvolvimento e tirando as dúvidas da equipe e vendo o projeto sendo feito, maiores são as chances de esse projeto dar certo segundo o entrevistado 6. *“Que quanto mais o cliente fica junto com a gente, participando, entendendo, vendo sendo desenvolvido passo a passo ficar pronto, mais chance de sucesso e mais chance de satisfação que a cliente tem”* – Entrevistado 6.

3.5.2 Variações do Processo de Estimativa de Esforço

Foram encontradas variações do processo de estimativa de esforço nas empresas entrevistadas. Os processos variam entre o mais específico (baseado nos aspectos em projetos de aplicativos móveis como, por exemplo, estimativas baseadas nos documentos UX e UI) até o mais geral (baseado em técnicas mais tradicionais de estimativa de esforço). A Figura 3.4 mostra que as empresas não têm uma uniformidade no processo de estimativa, pois elas variam desde fazer um processo muito simplificado onde não tem nenhum requisito (somente entendimento do problema) até um processo mais voltado para o mercado de aplicativos utilizando UX (User eXperience), porque é o que precisa para o aplicativo móvel.

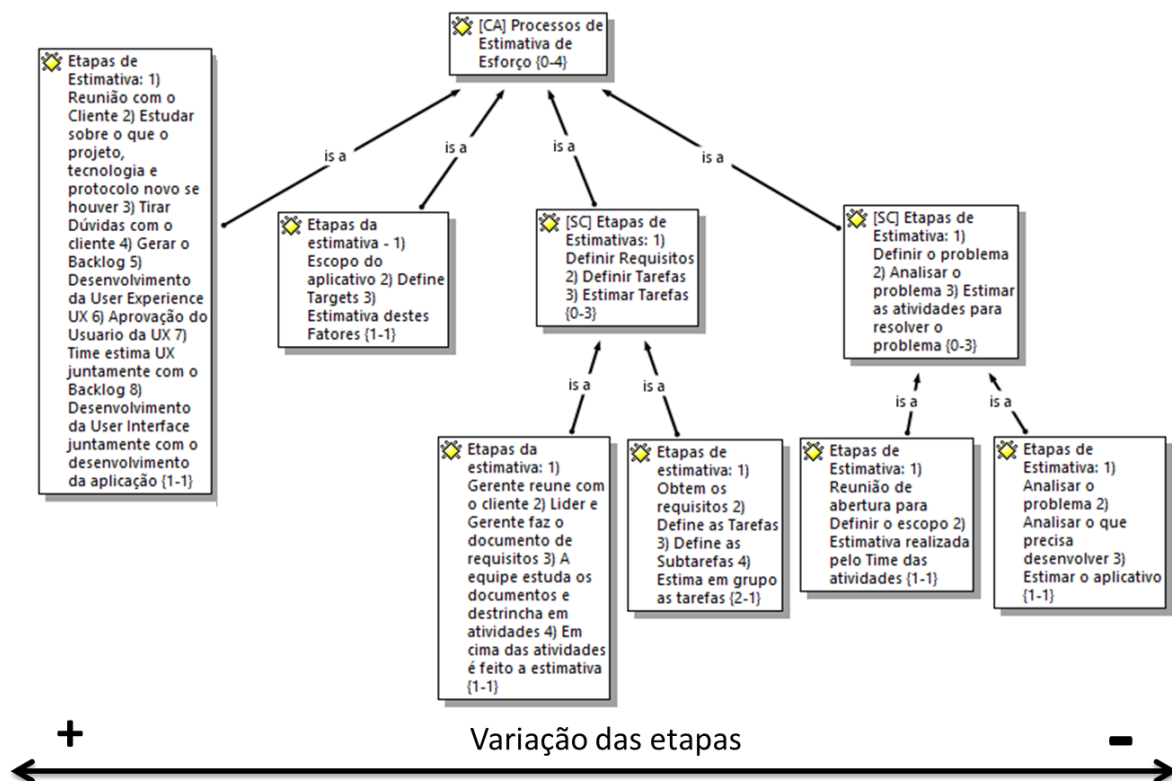


Figura 3.4: Variação do Processo de Estimativa de Esforço nas Empresas

Logo, há uma ausência de técnicas de estimativa ou de técnicas de estimativa específicas para a estimativa de esforço de projetos de aplicativos móveis. O uso de técnicas paramétricas é muito frequente na literatura como Pontos de Função (Albrecht, 1979) e Pontos de Função Cosmic (Cosmic, 2015), porém, nenhuma das empresas entrevistadas usa esses processos tradicionais. Então isso é uma dicotomia entre o mercado e academia, pois não usa nenhuma estimativa paramétrica e nem todas as empresas usam fatores específicos de aplicativos móveis, os mais especialistas foram os que apontaram realmente dependendo do Target (Quantidade de dispositivos que uma aplicação tem que rodar) terá mais trabalho ou menos trabalho (Veja a citação do Entrevistado 8). Outro entrevistado observou que a UX e UI são fatores preditores, pois é isso que acertam com o cliente (Veja a citação do Entrevistado 7). Logo, há uma variação entre o entrevistado mais especialista e o entrevistado menos especialista.

“Bom, de cara você tem que saber quais são os Targets, quais são os dispositivos em que o aplicativo tem que rodar, isso é vital, em cima disso você tenta pegar a maior abrangência de dispositivos...ai você vai ter que testar em todos esses dispositivos e é isso que dar trabalho. O pessoal faz um aplicativo e ai coloca no outro celular ou então uma quebra ai fica na horizontal, ai quebra ai fica uma coisa...Então tem que funcionar na vertical e na horizontal.” – Entrevistado 8.

“A gente precisa, primeiro ver a questão da UX, ver a coisa UI, ai montar a arquitetura, que regras de negócio vai ter, ai cada atividade dessa a gente vai colocando as horinhas lá pra saber no final.” – Entrevistado 7.

3.6 CONSIDERAÇÕES SOBRE O CAPÍTULO

Foram identificados através de uma pesquisa qualitativa, fatores que afetam a estimativa de esforço para projetos de aplicativos móveis e variações do processo de estimativa de esforço nas empresas que desenvolvem aplicativos móveis. Esses fatores foram levantados através do conhecimento obtido a partir da estimativa de esforço de oito especialistas de cinco diferentes empresas que desenvolvem aplicativos móveis. Foram utilizadas entrevistas semiestruturadas e procedimentos baseados em Grounded Theory ao longo da etapa de coleta e análise dos dados das entrevistas.

Os fatores identificados foram agrupados em dois principais grupos de fatores de estimativa de esforço para projetos de aplicativos móveis: Métricas de Tamanho e Fatores de Custo. As métricas de tamanho não possuem subcategorias, já os Fatores de Custo foram

organizados em subcategorias como Projetos de Aplicativos Móveis, Produto, Empresa, Equipe de desenvolvimento, Clientes e Tecnologia. Dependendo das características de um projeto de aplicativo móvel, é possível que algumas categorias se tornem mais relevantes do que outras ao estimar o esforço.

Além disso, alguns dos fatores aqui identificados diferem daqueles encontrados por Nitze *et al.* (2014), Abdullah *et al.* (2013), Abdullah *et al.* (2014), Heeringen e Gorb (2014), De Souza e Aquino (2014), D'Avanzo *et al.* (2015), Ferruci *et al.* (2015)(a), Ferruci *et al.* (2015) (b), Francese *et al.* (2015), Dombroviak e Ramnath (2007), Huijgens *et al.* (2014), Pocatilu e Ventrici (2009) (a), Pocatilu e Ventrici (2009)(b). Isto sugere que a utilização de técnicas e diferentes tipos de pesquisa podem levar a uma melhor compreensão e complementação daqueles aqui identificados.

Utilizando os princípios de Grounded Theory, não foi atingida a saturação teórica deste fenômeno, porque seriam necessários novos ciclos de experiências, para assim identificar a categoria do núcleo. Logo, como investigação futura, propõe-se a realização de novas entrevistas dentro de novos experimentos. Tais experiências devem ser executadas em diferentes contextos (cidades ou países) para reunir mais informações.

Além disso, o próximo capítulo apresenta o mapeamento sistemático na literatura sobre como é realizada a estimativa de projetos de aplicativos móveis, quais são os fatores preditores de esforço e modelos de estimativa específicos para aplicativos móveis.

CAPÍTULO 4 – MAPEAMENTO SISTEMÁTICO SOBRE ESTIMATIVAS PARA PROJETOS DE APLICATIVOS MÓVEIS

Este capítulo apresenta a condução e os resultados de um mapeamento sistemático cujo objetivo consistiu em identificar modelos de estimativa de projetos de aplicativos móveis e fatores preditores de esforço.

4.1 INTRODUÇÃO

Este mapeamento sistemático da literatura foi realizado entre junho a dezembro de 2016, na época a proposta de pesquisa ainda não estava definida, por isso que o tema de estimativa de projetos de aplicativos móveis foi investigado com maior abrangência para caracterizar as soluções existentes, identificando suas potencialidades e lacunas, para que possa agregar conhecimento à área e não simplesmente propor soluções similares às que já existem. Com este objetivo, foi conduzido um mapeamento sistemático da literatura que é um tipo de revisão sistemática. O mapeamento é um método usado para caracterizar e sumarizar informações existentes sobre questões de pesquisa (Kitchenham e Charters, 2007).

Considerando a importância que o desenvolvimento de aplicativos móveis desempenha na indústria de hoje e suas diferentes características comparada aos demais sistemas de software, um mapeamento sistemático da literatura é importante para estabelecer o estado atual de como é realizada a estimativa de esforço de aplicativos móveis. O mapeamento sistemático é orientado a identificar, avaliar e interpretar todas as pesquisas disponíveis relevantes sobre questões de pesquisa (Kitchenham e Charters, 2007).

Antes de iniciar o mapeamento sistemático, procurou-se outro trabalho similar já publicado e encontrou-se De Souza e Aquino (2014) que realizaram uma revisão sistemática da literatura sobre as características específicas das aplicações móveis como limitação de energia, largura de banda da rede e dentre outras. Do conjunto de 234 artigos identificados, 40 foram aceitos por meio dos critérios definidos por eles e 29 características foram extraídas. Depois, os autores realizaram um refinamento dessas características por meio de um *survey* realizado nas universidades e empresas que desenvolvem aplicativos móveis, este *survey* teve a finalidade de ratificar as características previamente levantadas e provar sua respectiva influência sobre o desenvolvimento móvel, resultando em 13 características. Porém, o foco deste mapeamento sistemático é investigar como é realizada a estimativa de tamanho e esforço de aplicativos móveis e não características específicas de aplicativos móveis. A seção

a seguir apresenta o protocolo do mapeamento sistemático da literatura realizado pelas autoras desta pesquisa.

Além desta seção introdutória, as seções seguintes deste capítulo estão organizadas da seguinte forma: A Seção 4.2 detalha o protocolo do mapeamento sistemático e o método de pesquisa utilizado. A Seção 4.3 apresenta os resultados do mapeamento para cada questão de pesquisa. A Seção 4.4 discute os resultados encontrados sobre estimativa de tamanho, esforço e fatores preditores de esforço de aplicativos móveis. Por fim, a Seção 4.5 apresenta as considerações finais.

4.2 PROTOCOLO DO MAPEAMENTO SISTEMÁTICO

Os mapeamentos sistemáticos requerem maior rigor na sua realização visto que eles fazem uso de uma metodologia rigorosa e passível de auditoria e repetição. Logo, seu resultado tende a ser mais confiável. Nas próximas seções, o protocolo para a condução do mapeamento sistemático é apresentado, baseado nas diretrizes para condução de revisões sistemáticas proposto por Kitchenham e Charters (2007).

4.2.1 Questões de Pesquisa

Uma abordagem para a formulação das questões de pesquisa é utilizar os critérios especificados pelo PICOC (Petticrew e Roberts, 2006) que investiga as estruturas das perguntas de acordo com cinco atributos: população, intervenção, comparação, resultados e contexto. No entanto, o foco deste mapeamento da literatura é caracterizar intervenções, logo os atributos de comparação não serão utilizados e, portanto, apenas atributos como a população, intervenção e resultado (PIO) serão considerados. A Tabela 4.1 mostra os detalhes.

Tabela 4.1: Questões de investigação estruturadas pelos critérios do PIO

População	Projetos de Aplicativos Móveis
Intervenção	Métodos / técnicas/ modelos utilizados para a estimativa de tamanho e/ou esforço para aplicativos móveis, fatores preditores considerados.
Resultado	Métodos / técnicas/ modelos utilizados para aplicativos móveis e fatores preditores.

A questão de pesquisa foi formulada para que pudesse abranger todo o escopo de estimativa de projetos de aplicativos móveis em Engenharia de Software. Neste contexto, a

questão de pesquisa deste mapeamento sistemático é a seguinte: “*Como é realizada a estimativa de projetos de aplicativos móveis?*”. A partir dessa questão de pesquisa foram definidas três subquestões:

- *Que modelos e métodos têm sido utilizados para a estimativa de tamanho e esforço de aplicativos móveis?*
- *Como foram avaliados os modelos e métodos de estimativa?*
- *Que fatores têm sido investigados como preditores de esforço para Aplicativos Móveis?*

4.2.2 Termos de Pesquisa

Os termos de pesquisa foram reunidos em uma string de busca que foi utilizada no processo de pesquisa. Os termos estão escritos em inglês por ser adotado pela grande maioria das conferências e periódicos internacionais relacionados ao tema de pesquisa. A seguir está a *string* de busca, sendo que foi utilizado um recurso oferecido pelas fontes de busca que delimita a área de pesquisa Ciência da Computação.

(“Mobile apps” OR “Mobile Applications” OR “Mobile Computing” OR “Mobile System” OR “Mobile Software”) AND (“Process” OR “Technique” OR “Model” OR “Method” OR “Approach” OR “Features” OR “Characteristics” OR “Factors”) AND (“Effort Estimat” OR “Effort Prediction” OR “Effort Measurement” OR “Size Estimat*” OR “Functional Size Measurement”)*

Para avaliar a qualidade e abrangência da *string* de busca, foi realizada uma pesquisa exploratória na qual foram definidos 4 artigos de controle, que são Nitze *et al.* (2014), De Souza e Aquino (2014), Heeringen e Gorb (2014) e Ferrucci *et al.* (2015b). Após a execução da *string* de busca nas bibliotecas digitais, verificou-se que os artigos de controle estavam entre as publicações retornadas.

4.2.3 Fontes de Busca

A busca foi realizada nas bibliotecas digitais Scopus, Engineering Village e ACM, pois as três bibliotecas digitais possuem um bom funcionamento e abrangência de suas máquinas de busca, evidenciada em alguns trabalhos, como o de Nitze *et al.* (2014), Francese *et al.* (2015) e Pocatilu e Vettrici (2009). Kitchenham e Charters (2007) afirmam que a Scopus é a maior

base de dados de indexação de resumos e citações. A ACM DL também indexa algumas publicações da Springer Link, Science Direct. Trindade *et al.* (2008) afirmam que Engineering Village agrega informações de diversos bancos de dados bibliográficos em Ciência da Computação (Compendex e Referex), abrangendo importantes periódicos e conferências da IEEE, ACM, Springer e Elsevier.

4.2.4 Critérios de Seleção de Estudos

A fim de selecionar os estudos que melhor respondem as questões de pesquisa, Kitchenham e Charters (2007) sugerem a definição de critérios de inclusão e exclusão para os estudos que são retornados pela *string* de busca. Para este mapeamento, foram definidos os critérios descritos na Tabela 4.2. Não foi adicionado critério de exclusão de artigos que não fossem relacionados a artigos primários com o objetivo de abranger estudos secundários e terciários, pois nenhuma revisão ou mapeamento da literatura específico sobre estimativa de aplicativos móveis foi retornado nas buscas.

Tabela 4.2: Critérios de Seleção

#	Código	Descrição
Critérios de Inclusão	CI01	O artigo analisa a estimativa de esforço e/ou tamanho dentro do domínio do desenvolvimento de Aplicativos Móveis.
	CI02	O artigo caracteriza os fatores preditores de esforço no desenvolvimento de Aplicativos Móveis.
Critérios de Exclusão	CE01	O artigo não analisa a estimativa de esforço e/ou tamanho dentro do domínio do desenvolvimento de Aplicativos Móveis.
	CE02	O artigo não realiza a estimativa nas fases iniciais do projeto.
	CE03	O artigo estima o tamanho do espaço de armazenamento que o aplicativo irá ocupar.
	CE04	Artigos duplicados.
	CE05	Falta de disponibilidade do artigo para download.
	CE06	Artigos que não estão escritos em Inglês ou em Português.

4.2.5 Processo de Seleção de Estudos

O mapeamento sistemático foi conduzido por duas pesquisadoras e o processo de seleção de estudos foi sistematizado seguindo três etapas: (1) execução da busca; (2) primeiro filtro e (3) segundo filtro.

Durante a execução de busca, foram realizados quatro refinamentos na *string* de busca para obter melhores resultados. Não houve restrição no período de tempo, porém a busca do último refinamento da *string* ocorreu em novembro de 2016. Utilizou-se a *string* de busca nas

fontes selecionadas e armazenou-se o conjunto de referências recuperadas na ferramenta Start¹.

No primeiro filtro, foram analisados os artigos retornados na busca por meio dos critérios de inclusão e exclusão, através da leitura dos títulos e abstracts dos artigos. No segundo filtro, foi realizada a leitura completa dos artigos incluídos no primeiro filtro e analisados novamente por meio dos critérios de inclusão e exclusão. Foram excluídos os artigos cujo conteúdo não atendia aos critérios de seleção, justificando a decisão. Após a verificação dos resultados da seleção do segundo filtro, foi realizada a extração dos dados dos artigos incluídos. Os resultados quantitativos obtidos em cada etapa do procedimento de seleção dos artigos são resumidos na Figura 4.1

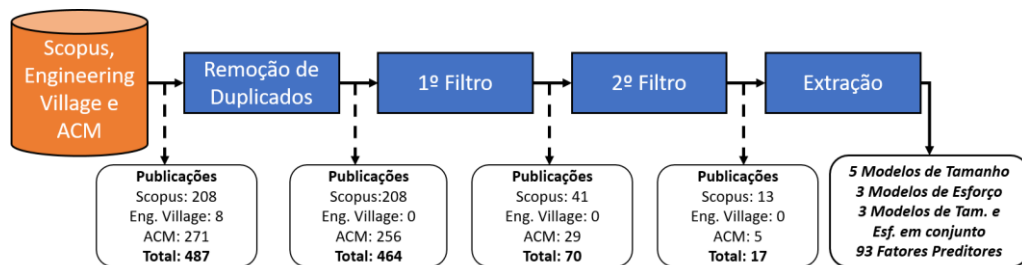


Figura 4.1: Processo de Seleção de Publicação

Durante a execução da busca, foram retornados 487 artigos nas fontes de busca. Em seguida, foram removidos os artigos duplicados, o primeiro filtro foi realizado por uma pesquisadora que analisou cada artigo, supervisionado pela outra pesquisadora. Foi conduzido um teste na biblioteca Engineering Village, porém 100% das publicações foram duplicadas em relação aos resultados da Scopus. A lista de todas as publicações está na Tabela 4.4

Com intuito de garantir a confiabilidade do processo de seleção, uma amostra de 50 artigos, selecionados aleatoriamente, foi utilizada para analisar o grau de concordância entre as duas pesquisadoras durante a seleção dos artigos. O acordo entre estas duas pesquisadoras foi avaliado utilizando-se o método estatístico Teste Kappa (Cohen, 1960). A interpretação dos resultados do Kappa segundo Landis e Koch (1977) está na Tabela 4.3. O resultado obtido foi um nível de concordância igual a 0,729, indicando uma concordância significativa entre as pesquisadoras (Landis e Koch, 1977).

¹ http://lapes.dc.ufscar.br/tools/start_tool

Tabela 4.3: Interpretação dos resultados do Kappa segundo Landis e Koch (1977).

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

Tabela 4.4: Lista das publicações relevantes extraídas

#	Fonte	Título	Autores	Ano
1	Scopus	Investigating Functional and Code Size Measures for Mobile Applications	Ferrucci, F. and Gravino, C. and Salza, P. and Sarro, F.	2015
2	Scopus	On the Use of Requirements Measures to Predict Software Project and Product Measures in the Context of Android Mobile Apps: A Preliminary Study	Francese, R. and Gravino, C. and Risi, M. and Scanniello, G. and Tortora, G.	2015
3	Scopus	COSMIC functional measurement of mobile applications and code size estimation	D'Avanzo, L. and Ferrucci, F. and Gravino, C. and Salza, P.	2015
4	Scopus	Investigating functional and code size measures for mobile applications: A replicated study	Ferrucci, F. and Gravino, C. and Salza, P. and Sarro, F.	2015
5	Scopus	Mobile application estimate the design phase	De Souza, L.S. and de Aquino, G.S., Jr.	2014
6	Scopus	An analogy-based effort estimation approach for mobile application development projects	Nitze, A. and Schmietendorf, A. and Dumke, R.	2014
7	Scopus	Mobile game size estimation: COSMIC FSM rules, UML mapping model and Unity3D game engine	Abdullah, N.A.S. and Rusli, N.I.A. and Ibrahim, M.F.	2014
8	Scopus	Measure the functional size of a mobile app: Using the cosmic functional size measurement method	Heeringen, H.V. and Gorp, E.V.	2014
9	Scopus	Mobile application development: How to estimate the effort?	De Souza, L.S. and De Aquino, G.S.	2014
10	Scopus	The applicability of present estimation models to the context of mobile applications	De Souza, L.S. and De Aquino Jr., G.S.	2014
11	Scopus	MEstiAM: Estimation model for mobile applications	De Souza, L.S. and De Aquino Jr., G.S.	2014
12	Scopus	Mobile applications: The paradox of software estimation	De Souza, L.S. and De Aquino, G.S., Jr.	2014
13	Scopus	A case study in COSMIC functional size measurement: Angry bird mobile application	Abdullah, N.A.S. and Rusli, N.I.A. and Ibrahim, M.F.	2013
14	ACM	M-applications Development Using High Performance Project Management Techniques	Pocatilu, Paul and Vetrici, Marius	2009
15	ACM	Schedule Risk Management for Business M-applications Development Projects	Pocatilu, Paul and Vetrici, Marius	2009
16	ACM	How to Build a Good Practice Software Project Portfolio?	Huijgens, Hennie and Solingen, Rini van and Deursen, Arie van	2014

#	Fonte	Título	Autores	Ano
17	ACM	A Taxonomy of Mobile and Pervasive Applications	Dombroviak, Krista M. and Ramnath, Rajiv	2007
18	ACM	A Case Study on Naked Objects in Agile Software Development	Keränen, Heikki and Abrahamsson, Pekka	2005

4.2.6 Procedimento de Extração dos Dados dos Artigos Selecionados

Depois do término do segundo filtro, os artigos selecionados foram submetidos ao processo de extração dos dados. A extração foi realizada de forma sistemática por meio de um formulário definido para registrar as informações necessárias relativas a cada artigo. O formulário de extração permite o registro de todas as informações necessárias para responder às questões de pesquisa. O modelo do formulário de extração encontra-se na Tabela 4.5 e os formulários preenchidos com os dados dos artigos selecionados encontram-se no relatório técnico (De Souza e Conte, 2018).

Tabela 4.5: Formulário de Extração dos Dados

Título do Artigo (Autores, Conferência e Ano de Publicação)	
Item	Valor
Resumo	
Estimativa	() Tamanho () Esforço
É um	() Modelo () Método () Técnica () Outros
Em que estágio é feita a estimativa?	
A estimativa tem apoio de alguma ferramenta? Qual?	() Obrigatório () opcional () Não Qual?
Usa algum método/técnica como base? Qual? E em que estágio da estimativa?	
Como é realizada a estimativa?	
Quais métricas foram usadas para avaliar a estimativa de tamanho e/ou esforço?	
Que fatores têm sido investigados como preditores de esforço para Aplicativos Móveis?	
Em que estágio esses preditores de esforço estão reunidos?	
Qual o tipo de estudo experimental realizado?	() Não Possui () Estudo de Caso () Experimento Controlado () Survey () Pesquisa-Ação
Qual análise dos dados foi realizada?	() Não possui () Quantitativo () Qualitativo Resumo:
Novas contribuições e ideias	
Referências Interessantes	

4.3 RESULTADOS DO MAPEAMENTO SISTEMÁTICO

4.3.1 Visão Geral dos Resultados

No primeiro momento, os resultados do mapeamento sistemático foram analisados quantitativamente em relação aos veículos e ao ano de publicação. Em relação aos veículos de publicação, 11% dos artigos foram publicados em revistas ou periódicos, enquanto 89% foram publicados em conferências. No que se refere ao período de publicação, a Figura 4.2 mostra as publicações por ano.

Desde 2005, a estimativa de projeto de aplicativos móveis vem sendo relatada nas publicações científicas. Em 2014 houve um aumento no número de publicações, esse fato ocorreu devido a 5 publicações dos autores De Souza e Aquino sobre o mesmo modelo de estimativa de tamanho e esforço em conjunto, variando apenas o projeto que foi utilizado para avaliar o modelo, sendo assim as 5 publicações poderia ter se resumido em apenas uma publicação. Uma possível explicação por não ter estudos em 2016 é que as publicações e trabalhos relevantes ainda não tinham sido indexados nas fontes de busca.

Das publicações analisadas para os projetos de aplicativos móveis, 28% descrevem modelos de estimativa de tamanho, 6% abordam modelos de estimativa de esforço, 17% analisam fatores preditores de esforço, 11% abordam modelos de estimativa de tamanho e esforço no mesmo trabalho e 39% falavam de ambos (estimativa de tamanho, esforço e fatores preditores de esforço para projetos de aplicativos móveis).

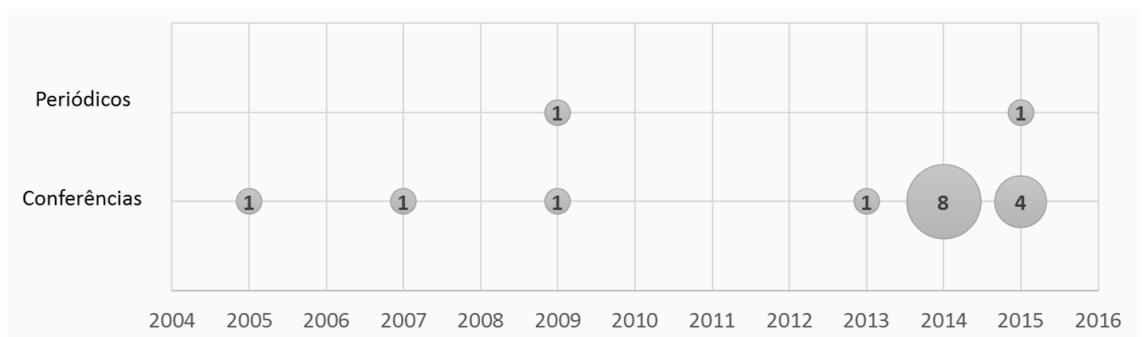


Figura 4.2: Distribuição das publicações por ano e por veículo de publicação.

4.3.2 Que modelos e métodos têm sido utilizados para a estimativa de tamanho e esforço de aplicativos móveis?

Foram identificados 11 modelos de estimativa de projetos de aplicativos móveis. Pode-se observar que modelos de estimativa de tamanho foram os mais citados com 45%, enquanto

que os modelos de estimativa de esforço com 27% e os modelos que estimam tamanho e esforço em conjunto correspondem 27% também. Os resultados dessa questão de pesquisa foram subdivididos em modelos de estimativa de tamanho, esforço e ambos.

- *Estimativa de Tamanho* – o modelo é focado em estimar o tamanho funcional do aplicativo em pontos de função (Albrecht, 1979) ou pontos de função cosmic (Cosmic, 2015). Os modelos são encontrados em Abdullah *et al.* (2013), Abdullah *et al.* (2014), Heeringen e Gorb (2014), Ferrucci *et al.* (2015a) e D’Avanzo *et al.* (2015).
- *Estimativa de Esforço* – o modelo é focado em estimar o esforço que será necessário para desenvolver o aplicativo móvel em horas/ pontos de função cosmic. Os modelos são encontrados em Pocatilu e Ventrici (2009a), Pocatilu e Ventrici (2009b) e Keränen e Abrahamsson (2005).
- *Ambos* – o modelo é focado em estimar conjuntamente o tamanho e esforço para desenvolver o aplicativo móvel. Os modelos são encontrados em Nitze *et al.* (2014), De Souza e Aquino (2015) e Francese *et al.* (2015).

O resumo de cada modelo encontra-se na Tabela 4.6 que mostra os modelos de estimativa de projetos de aplicativos móveis que são detalhados nas seções 4.3.2.1., 4.3.2.2 e 4.3.2.3. A Tabela 4.6 possui uma coluna chamada “Combinação” que corresponde à legenda da classificação dos modelos apontados: A- Modelos de Estimativa de Tamanho; B - Modelos de Estimativa de Esforço; C - Fatores Preditores de Esforço. Além disso, algumas características dos modelos julgadas relevantes foram analisadas a seguir:

- *Classificação de Estimativa* - os modelos encontrados foram classificados em três categorias de estimativa definidos por Mendes (2014). As categorias são modelos classificados como técnicas algorítmicas, baseado por especialista e híbridos (como por exemplo, Pocatilu e Ventrici (2009b) utiliza um modelo que é classificado em técnicas algorítmicas e baseado por especialistas). Cerca de 64% modelos são classificados como técnicas algorítmicas, 27% híbridos e apenas 9% baseado por especialistas.
- *Entrada de Dados* – os modelos apontados possuem entrada de dados específicas para estimar projetos. Cerca de 18% dos modelos utilizam diagramas da UML, 64% usam documentos de requisitos, apenas 9% usam *template* de *mockups* e 9% usam *uses*

stores. Logo a maioria utiliza documento de requisitos para estimar esse tipo de projeto.

- *Métrica de Tamanho* – os modelos encontrados que estimam somente o tamanho e esforço conjuntamente contabilizam 9 modelos e definiram a métrica de tamanho em pontos de função cosmic (COSMIC, 2015), pontos de função (Albrecht, 1979), medição RAD (documento de análise de requisitos) e medição SC (código-fonte) (Francese *et al.*, 2015). Cerca de 78% utilizam pontos de função cosmic, 11% pontos de função, e 11% medição RAD & medição SC.
- *Métrica de Esforço* – os modelos que estimam somente o esforço conjuntamente com o tamanho contabilizam 6 modelos e utilizam as seguintes nomenclaturas para as métricas de esforço: cerca de 33% utiliza horas, 17% pessoa-hora, 33% homens-ano e 17% não informam a métrica que utiliza.
- *Ponderam Fatores* – os modelos utilizam fatores preditores de esforço para estimar projetos de aplicativos móveis. Esses fatores foram classificados em Métricas de Tamanho e Fatores de Custo que são detalhados na seção 4.4. Cerca de 73% dos modelos ponderam fatores preditores de esforço específicos para aplicativos móveis que podem ser encontrados em Abdullah *et al.* (2013), Abdullah *et al.* (2014), Heeringen e Gorb (2014), Nitze *et al.* (2014), De Souza e Aquino (2015), Pocatilu e Ventrici (2009a), Pocatilu e Ventrici (2009b) e Francese *et al.* (2015). Os outros 27% dos modelos não ponderam fatores.

Tabela 4.6: Lista de Modelos Extraídos

Referência	Combinação	Classificação da Estimativa	Modelo	Métrica de Tamanho	Métrica de Esforço	Pondera Fatores?	Dados de Entrada
Nitze <i>et al.</i> (2014)	A,B,C	Híbrida (Técnicas Algorítmicas e Especialistas)	Técnicas Algorítmicas: mapeamento de Templates de Mockups para Pontos de Função Cosmic; e Baseada por Especialistas: Ponderação de Fatores e Feedback de especialistas	Pontos de Função Cosmic (CFP)	Não informado pelos autores	Sim (7 Fatores)	Templates de Mockups
Abdullah <i>et al.</i> (2013)	A,C	Técnicas Algorítmicas	Mapeamento de Diagramas UML para Pontos de Função Cosmic	Pontos de Função Cosmic (CFP)	-	Sim (5 Fatores)	Diagramas UML
Abdullah <i>et al.</i> (2014)	A,C	Técnicas Algorítmicas	Mapeamento de Diagramas UML para Pontos de Função Cosmic	Pontos de Função Cosmic (CFP)	-	Sim (28 Fatores)	Diagramas UML
Heeringen e Gorb (2014)	A,C	Técnicas Algorítmicas	Identificação de tipos de processo funcional e definição de cada parâmetro envolvido nos processos	Pontos de Função Cosmic (CFP)	-	Sim (10 Fatores)	Documento de Requisitos
De Souza e Aquino (2014) (a,b,c,d,f)	A,B,C	Técnicas Algorítmicas	Utiliza FISMA para calcular o Tamanho e usa uma equação matemática para calcular o esforço	Pontos de Função.	Horas	Sim (7 Fatores)	Documento de Requisitos
D'Avanzo <i>et al.</i> (2015)	A	Técnicas Algorítmicas	Utiliza o COSMIC e Regressão Linear	Pontos de Função Cosmic (CFP) e CodeSize (kB)	-	Não	Documento de Requisitos
Ferrucci <i>et al.</i> (2015) (a)	A	Técnicas Algorítmicas	Mapeamento do movimento de dados do Documento de requisitos conforme D'Avanzo <i>et al.</i> (2015)	Pontos de Função Cosmic (CFP)	-	Não	Documento de Requisitos

Referência	Combinação	Classificação da Estimativa	Modelo	Métrica de Tamanho	Métrica de Esforço	Pondera Fatores?	Dados de Entrada
Ferrucci <i>et al.</i> (2015) (b)	A	Técnicas Algorítmicas	Comparação dos modelos de D'Avanzo <i>et al</i> (2015) e Heeringen e Gorb (2014)	Pontos de Função Cosmic (CFP)	-	Não	Documento de Requisitos
Francese <i>et al.</i> (2015)	A,B,C	Técnicas Algorítmicas	Modelo de regressão linear	Medição RAD (Documento de Análise de Requisitos) e Medição SC (código-fonte)	Pessoa/Hora	Sim (13 Fatores)	Documento de Requisitos
Pocatilu e Ventrici (2009) (b)	B,C	Híbrida (Técnicas Algorítmicas e Especialistas)	Especialista: Top-Down e Bottom-Up. Técnicas Algorítmicas: Métricas de desempenho e Método Estatístico de Simulação Monte Carlo	-	Homens-anos	Sim (28 Fatores)	Documento de Requisitos do projeto e anteriores similares com estimativas passadas
Pocatilu e Ventrici (2009) (a)	B,C	Híbrida (Técnicas Algorítmicas e Especialistas)	Especialista: Top-Down e Bottom-Up. Técnicas Algorítmicas: Métricas de desempenho	-	Homens-anos	Sim (12 Fatores)	Documento de Requisitos
Keränen e Abrahamsson (2005)	B	Baseada por Especialistas	Especialistas consultam cada Uses Stores e pontuam em horas.	-	Horas	Não	User Stores

4.3.2.1 Modelos de Estimativa de Tamanho

Este tipo de modelo é focado em estimar o tamanho funcional do aplicativo em pontos de função (Albrecht, 1979) ou pontos de função cosmic (Cosmic, 2015). Os modelos são encontrados em Abdullah *et al.* (2013), Abdullah *et al.* (2014), Heeringen e Gorb (2014), Ferrucci *et al.* (2015a) e D'Avanzo *et al.* (2015).

Abdullah *et al.* (2013) propuseram um modelo que utiliza medição funcional por meio de pontos de função COSMIC com base em UML para estimar a aplicação móvel do jogo. A ideia-chave é usar representações em UML para capturar as informações necessárias para a medição e estimativa. Um estudo de caso foi realizado com o jogo móvel Angry Birds², usado para demonstrar como UML pode realmente melhorar a prática de medição COSMIC mapeando contexto UML com as regras FSM COSMIC (Medição de Tamanho Funcional Cosmic) (Cosmic, 2003). As representações da UML envolvem o uso de diagrama de caso de uso, os elementos do modelo de comportamento, diagramas de componentes, diagramas de objetos e diagramas de sequência para representar requisitos funcionais e requisitos de fluxo de trabalho de base. A vantagem é que a saída é em pontos de função cosmic, facilitando a comparação com outros métodos. A desvantagem é que utiliza regressão linear, logo é necessário ter histórico de estimativa para definir o modelo.

Do mesmo modo, Abdullah *et al.* (2014) adaptaram a Medição de Tamanho Funcional COSMIC (FSM COSMIC) como uma alternativa para estimar o tamanho do aplicativo de jogo e usa Unity3D para representar a arquitetura do jogo, ou seja, realiza a estimativa de tamanho de acordo com o Diagrama de Componentes da UML, uma vez que Unity3D é capaz de controlar as características e componentes para o desenvolvimento de aplicativos de jogo para dispositivos móveis. A vantagem desse modelo é a facilidade da modelagem de jogos Unity3D para Diagramas de Componentes da UML. A desvantagem consiste em estimar por meio de diagramas de componentes, pois o nível de complexidade que se dar dos passos a serem seguidos é alta no mapeamento para pontos de função COSMIC. A limitação do modelo consiste na restrição de domínio para jogos desenvolvidos em Unity3D.

D'Avanzo *et al.* (2015) aplicaram método de medição COSMIC para calcular o tamanho funcional das aplicações móveis. Fizeram um experimento com 8 aplicações Android que permite avaliar se no domínio considerado, o tamanho funcional COSMIC pode

² <https://play.google.com/store/apps/details?id=com.rovio.angrybirds&hl=pt>

ser usado para obter previsões de tamanho de código (em Kb). O estudo foi baseado em 8 aplicações e uma regressão linear foi utilizada para construir os modelos de previsão. Os resultados da validação foram obtidos por meio da aplicação de uma validação cruzada "leave-one-out", estimativas precisas foram obtidas levando-se em consideração limiares amplamente utilizados no contexto da estimativa do esforço. A vantagem é que usa pontos de função COSMIC (Cosmic, 2015). A desvantagem é a elaboração para projetos novos, pois para realizar a estimativa por regressão linear é necessário ter um histórico de estimativas de projetos que já foram desenvolvidos.

Do mesmo modo, Heeringen e Gorb (2014) propõem um modelo aproximado usado para dimensionar aplicações móveis, por meio de identificação de cada tipo de processo funcional e a quantificação dos parâmetros envolvidos para cada tipo de processo funcional. A vantagem é que a estimativa é realizada através do processo funcional dos requisitos que categoriza o tipo de requisito tornando assim mais fácil o processo de estimativa e os fatores preditores são levados em consideração no momento da estimativa. A desvantagem é que não possui experimento que avalie o modelo proposto.

4.3.2.2 Modelos de Estimativa de Esforço

Este tipo de modelo é focado em estimar o esforço que será necessário para desenvolver o aplicativo móvel em horas/ pontos de função cosmic. Os modelos são encontrados em Pocatilu e Ventrici (2009a), Pocatilu e Ventrici (2009b) e Keränen e Abrahamsson (2005).

Pocatilu e Ventrici (2009a) analisam as especificidades dos projetos de desenvolvimento de m-aplicativos (projetos realizados para aplicativos móveis). Eles utilizam uma abordagem híbrida (técnicas algorítmicas e baseado em especialista). As técnicas baseadas por especialista foram duas "Top-Down" e "Bottom-Up". Já a técnica algorítmica usou um conjunto de métricas de desempenho para a avaliação da qualidade do projeto. A vantagem consiste na utilização da parte da estimativa de métodos difundidos na literatura: Top-down e Bottom-Up, pois é mais rápido a aceitação das pessoas. As desvantagens consistem no modelo não ter sido avaliado através de nenhum experimento e a grande quantidade de equações para calcular as métricas de desempenho.

Do mesmo modo, Pocatilu e Ventrici (2009b) realizaram um trabalho muito semelhante ao anterior, porém o diferencial é que utilizaram o método estatístico de Simulação Monte Carlo. A vantagem é a mesma do parágrafo anterior que consiste em utilizar parte da estimativa de métodos difundidos na literatura: *Top-down* e *Bottom-Up*. A

desvantagem é que possui uma alta complexidade em calcular a matriz de Monte Carlo que requer uma função de distribuição da probabilidade de duração para cada tarefa.

Keränen e Abrahamsson (2005) aplicaram *Naked Objects* (um padrão arquitetônico que expõe objetos de negócios) (Pawson, 2002) e desenvolvimento ágil de software em conjunto. Este artigo relata um estudo de caso para o desenvolvimento de um aplicativo móvel usando o Framework de Naked Objects. Os resultados empíricos revelaram que o framework ainda não está maduro suficiente para aplicações que requerem operações de banco de dados intensas, por outro lado a equipe de desenvolvimento foi capaz de criar uma interface de usuário operacional apenas em cinco horas, o que demonstra a aplicabilidade do Framework de Naked Object em configurações práticas.

4.3.2.3 Modelos de Estimativa de Tamanho e Esforço em Conjunto

Este tipo de modelo é focado em estimar conjuntamente o tamanho e esforço para desenvolver o aplicativo móvel. Estes tipos de modelos foram encontrados em Nitze *et al.* (2014), De Souza e Aquino (2014) e Francese *et al.* (2015).

Nitze *et al.* (2014) propõem um modelo que possui três passos: Primeiro, mede o tamanho do app – técnica consiste em criar templates de mockups e cada template é associado a pontos de função Cosmic (Cosmic, 2003). Segundo, pondera fatores e por último, coletam o feedback dos interessados no projeto para coletar o esforço real do projeto para poder calcular a acurácia da estimativa e, portanto, integrar no processo. A vantagem é a facilidade em estimar por meio de templates de mockups e por ser baseado em analogia, ou seja, a equipe recorre as estimativas de projetos anteriores semelhantes para estimar o novo projeto. A desvantagem consiste se não houver histórico de estimativa não terá como utilizar o modelo proposto.

De Souza e Aquino (2014) propõem um modelo de estimativa paramétrica, a estimativa é calculada através de uma fórmula: Estimativa de esforço (horas) = tamanho (pontos de função) * reutilização * taxa de entrega (hora/ pontos de função) * o status do projeto; as variáveis de reutilização, taxa de entrega e status do projeto não são explicadas no artigo. A vantagem do modelo é que possui fatores preditores específicos para aplicativos móveis. A desvantagem é que a estimativa de tamanho é realizada por meio de uma ferramenta paga que implementa o modelo FISMA (The Finnish Software Metrics Association) (Forselius, 2004).

Francese *et al.* (2015) também utiliza um modelo paramétrico que possui três passos: Primeiro, os modelos de estimativa são construídos a partir de informações recolhidas nos

documentos de especificação de requisitos (número de atores, o número de casos de uso e número de classes). Segundo, os autores empregaram uma regressão linear para construir esses modelos. Por último, os autores avaliam a precisão das previsões obtidas pela aplicação dos modelos, compararam as suas estimativas com os obtidos considerando os modelos construídos em medidas de software (por exemplo, número de classes, número de arquivos, e número de linha de código). A vantagem é que extrair informações dos requisitos é uma tarefa relativamente fácil. A desvantagem é que para construir uma regressão linear é necessário ter projetos anteriores desenvolvidos, mas caso um projeto possua uma tecnologia nova, equipe de desenvolvimento nova e não houver histórico de desenvolvimento, não será possível construir essa regressão linear.

4.3.3 Como foram avaliados os modelos e métodos de estimativa?

De 11 modelos de estimativa de projetos de aplicativos móveis identificados, apenas 7 foram avaliados experimentalmente, correspondendo cerca de 64% dos modelos encontrados. Apenas um estudo comparativo, realizado por Ferrucci *et al.* (2015b), foi encontrado no mapeamento. Eles compararam dois modelos de estimativa de tamanho proposto por D'Avanzo *et al.* (2015) e Heeringen e Gorb (2014). Os resultados mostraram que o tamanho funcional COSMIC (Cosmic, 2015) avaliado com o modelo Heeringen e Gorb (2014) que a precisão da predição não satisfaz os critérios de avaliação e se mostrou ligeiramente pior do que o obtido no estudo original baseado na abordagem proposta por D'Avanzo *et al.* (2015). A Tabela 4.7 mostra como foram avaliados experimentalmente os modelos. Os principais resultados estão destacados a seguir:

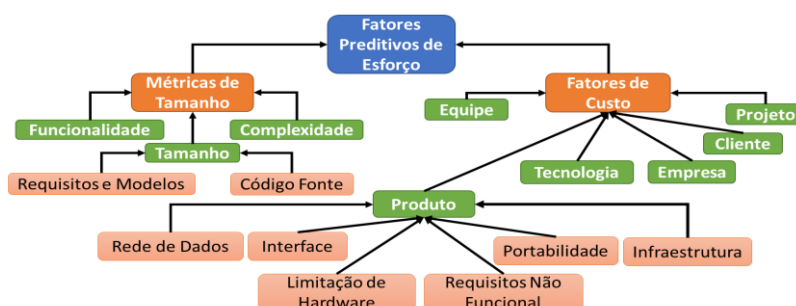
- *Metodologia Experimental* – os modelos foram avaliados da seguinte forma: cerca de 57% usaram estudo de caso e 43% usaram experimento controlado.
- *Ambiente* – cerca de 29% dos modelos foram avaliados na academia e 71% foram avaliados na indústria.
- *Tipo de Análise de Dados* – os modelos foram avaliados pelos seguintes tipos de análise de dados: Cerca de 86% quantitativamente e 14% qualitativamente.
- *Amostra* – foi contado o número de aplicativos que foram usados para o experimento que variou entre 1 a 23 aplicativos.

Tabela 4.7. Lista de Modelos Avaliados Experimentalmente

Referência	Metodologia	Ambiente	Tipo de Análise de Dados	Amostra
Abdullah <i>et al.</i> (2013)	Estudo de Caso	Indústria	Quantitativo	1
D'Avanzo <i>et al.</i> (2015)	Experimento Controlado	Indústria	Quantitativo	8
Ferrucci <i>et al.</i> (2015) (a)	Estudo de Caso	Indústria	Quantitativo	13
Ferrucci <i>et al.</i> (2015) (b)	Experimento Controlado	Indústria	Quantitativo	13
Francesse <i>et al.</i> (2015)	Estudo de Caso	Academia	Quantitativo	23
Keränen e Abrahamsson (2005)	Estudo de Caso	Indústria	Quantitativo e Qualitativo	1
De Souza e Aquino (2014 a,b,c,d,f)	Experimento Controlado	Academia	Quantitativo	2

4.3.4 Que fatores têm sido investigados como preditores de esforço para Aplicativos Móveis?

Entre os estudos selecionados, foram encontradas evidências relevantes de fatores preditores de esforço de projeto de aplicativos móveis. Dos 18 estudos retornados, apenas 10 apontaram fatores preditores de esforço, correspondendo a cerca de 59%. Foram extraídos 66 fatores que foram organizados em uma hierarquia de 4 níveis (ver Figura 4.3), adaptação da hierarquia da taxonomia proposta por Britto *et al.* (2017), cada nível é formado por categorias.

**Figura 4.3:** Fatores Preditores de Esforço: Como estão organizados

As categorias do terceiro e quarto níveis hierárquico são definidas da seguinte forma:

- **Métricas de tamanho**

- *Tamanho* - Esta categoria inclui métricas que medem diretamente o comprimento de aplicações Web com base no tamanho / comprimento dos seus elementos de composição.
- *Funcionalidade* - Esta categoria engloba métricas que medem indiretamente o tamanho do aplicativo móvel, baseadas em suas características e funções.

- *Complexidade* - Esta categoria abrange métricas que medem indiretamente o tamanho do aplicativo móvel, baseadas na dificuldade associada com os seus elementos de composição.
- **Fatores de custo**
 - *Cliente* - esta categoria abrange fatores de custo relacionados ao cliente que exige o desenvolvimento de um aplicativo móvel.
 - *Empresa* - esta categoria engloba os fatores de custos associados à empresa contratada por um cliente para desenvolver um aplicativo o aplicativo móvel.
 - *Produto* - esta categoria inclui os fatores de custo relacionados a requisitos e restrições associada a um aplicativo móvel.
 - *Projeto* - esta categoria abrange os fatores de custo associados à configuração de um aplicativo móvel.
 - *Equipe* - esta categoria abrange os fatores de custo que estão associados com a equipe de desenvolvimento responsável pela execução de um projeto de aplicativo móvel.
 - *Tecnologia* - Esta categoria abrange os fatores de custos associados às tecnologias (Linguagem de programação, ferramentas, plataformas) exigidos em um projeto de aplicativo móvel.

Os fatores preditores de esforço estão na Tabela 4.8 deste capítulo, as referências EP1 a EP15 encontram-se no Apêndice B.

Tabela 4.8: Lista de Fatores Preditores de Esforço

Métricas	Categoria	Fator	Descrição	Referência
Métricas de Tamanho	Tamanho			
	Requisitos e Modelos	Número de Requisitos Funcionais	Número de requisitos do projeto	EP1
		Número de Atores do Caso de Uso	Número total de Atores do caso de uso do projeto, pode ser descritivo ou diagrama	EP1
		Número de Casos de Uso	Número total de Casos de Uso pode ser descritivo ou diagrama	EP1
		Número de Classes	Número total de Classes do projeto, pode ser descritivo ou diagrama	EP1
		Número de Diagramas de Sequência	Número total de Diagramas de Sequência do projeto.	EP1
	Código Fonte	Número de Classes	Número total de Classes contidas no código fonte do aplicativo móvel	EP1
		Número de Arquivos	Número total de Arquivos do projeto	EP1
		Número de Métodos (incluindo os herdados)	Número total de métodos implementados contidos no código fonte do aplicativo móvel	EP1
		Número de todas as linhas	Número total de linhas do código fonte do aplicativo móvel	EP1
Número de linhas contendo código fonte		Número total de linhas contendo o código fonte do aplicativo móvel	EP1	

Métricas	Categoria	Fator	Descrição	Referência
		Número de linhas contendo comentários	Número total de linhas contendo comentários do código fonte do aplicativo móvel	EP1
		Número de declarações	Número total de declarações contidas no código fonte do aplicativo móvel	EP1
		Tamanho do código (kB)	Tamanho em kB do aplicativo móvel	EP11
		Número de arquivos XMI sobre elementos gráficos de um aplicativo para dispositivos móveis.	Número de arquivos XMI sobre elementos gráficos (por exemplo, caixa de texto e assim por diante) de um aplicativo para dispositivos móveis	EP11
	Complexidade	Complexidade Ciclométrica	Medida de complexidade, onde é considerado os caminhos independentes que o código do projeto pode tomar.	EP1, EP5
	Funcionalidade	COSMIC	Pontos de Função calculados usando o método COSMIC	EP3, EP4, EP11, EP12, EP13, EP14, EP15
		IFPUG	Pontos de função calculados usando o método IFPUG	EP2
Fatores de Custo	Equipe de Desenvolvimento	Nível de Comunicação entre a equipe	Grau de comunicação entre os membros da equipe que pode variar de baixo, médio ou alto.	EP5, EP6
		Nível de Experiência da equipe em Desenvolvimento	Grau de experiência em programação e desenvolvimento de banco de dados que pode variar de baixo, médio ou alto.	EP1, EP5
		Experiência da equipe no contexto da aplicação	Se tem experiência ou não no contexto da aplicação, por exemplo conhecimento nas áreas e aplicações empresariais.	EP1, EP5
		Definição de papéis das equipes	Se os papéis dos membros da equipe estão definidos ou não.	EP5
	Empresa	Nível de certificação da empresa	Se a empresa tem ou não.	EP6
		Nível de Falhas de software e hardware	A frequência de falhas de software e hardware que pode variar de nunca, raramente e sempre.	EP6
		Envolvimento dos concorrentes	Se existe envolvimento dos concorrentes ou não.	EP6
		Nível de Competências de fornecedores	Competência está relacionado a cumprir os prazos e qualidade da entrega da matéria prima. O grau de competência pode variar de baixo, médio ou alto.	EP6
	Cliente	Nível de Competências dos usuários e clientes	Competência do Cliente pode ser baixa, média ou alta.	EP6
		Nível de Expectativas	Grau da expectativa do cliente pode	EP6

Métricas	Categoria	Fator	Descrição	Referência
	Projeto	dos usuários e clientes	variável de baixo, médio ou alto.	
		Nível de Preparação da gestão de mudança	O grau de preparação para eventuais mudanças que poderão ocorrer durante o projeto é de extrema relevância. Esse grau pode variar de preparação máxima para mudanças. Preparação razoável para mudanças. Preparação mínima para mudanças.	EP5
		Nível de Definição dos Objetivos das partes interessadas	Se os objetivos estão definidos ou não.	EP5
		Processo de validação e verificação e implementação	Se existem um processo de verificação e validação e implementação ou não.	EP5
		Teste em ambiente real	Se o teste é realizado nos dispositivos dos usuários finais ou não.	EP5
		Mensurar o processo	Se é mensurada cada etapa do processo do projeto ou não.	EP5
		Entendimento dos custos de modificações	Se é calculado os custos das modificações antes de fazê-las ou não.	EP5
		Tipos de sistemas de back-end (CMS, web site, ERP, CRM ...)	O aplicativo tem um sistema de back-end a complexidade é baixa, o aplicativo tem dois sistemas de back-end a complexidade é média e três ou mais a complexidade é alta.	EP3, EP4
		Bugs existentes no software	Consiste na frequência de bugs no software que pode variar de nunca, raramente, as vezes e sempre.	EP5
		Uso de APIs nativas (Maior Esforço)	O uso de APIs nativas do dispositivo móvel para gravar aplicativos requer maior esforço e o não uso requer baixo esforço.	EP5
Produto				
Rede de dados	Largura de banda da rede	◦ (-) O aplicativo deve exigir a largura de banda máxima. ◦ (+/-) O aplicativo deve exigir largura de banda razoável. ◦ (+) A aplicação exigirá uma largura de banda mínima.	EP2, EP4, EP5	
	Comutação de Wifi	O aplicativo deve trabalhar em offline e online. Caso for offline será menos complexo que online pois dependerá menos de comunicação com servidores e dentre outros fatores.	EP2, EP4	
	Tipos de Conectividade na rede	◦ (-) O aplicativo deve ter a máxima disposição para usar conexões como 3G, Wi-Fi, Wireless, Bluetooth, Infravermelho e outros. ◦ (+/-) O aplicativo deve ter predisposição razoável para usar conexões como 3G,	EP2, EP4	

Métricas	Categoria	Fator	Descrição	Referência
			Wi-Fi e Wireless. ° (+) O aplicativo deve ter apenas uma predisposição para usar conexões, que podem ser: 3G, Wi-fi, Wireless, Bluetooth, Infravermelho ou outros.	
		Sincronização do banco de dados	Nível de complexidade para sincronizar com o banco de dados que pode variar de baixa para aplicações locais, média para aplicações que salva local e depois envia para o servidor ou alta que conecta com servidor para poder sincronizar com o banco de dados.	EP5
	Interface	Variação de Entrada da Interface Gráfica	A Entrada de Interface pode ter: O aplicativo deve ter interfaces de entrada para tela de toque, voz, vídeo, teclado e outros ou o aplicativo deve ter interfaces de entrada padrão para teclado ou o aplicativo deve ter qualquer um dos tipos de interfaces, tais como: tela sensível ao toque, voz, vídeo, teclado ou outros.	EP2
		Qualidade visual	Ajuste de componentes de interface do usuário para reprodução de um projeto corporativo, logotipo, marca corporativa, navegação, animações. Complexidade baixa se for poucos componentes, complexidade média se for razoável o número de componentes de interface e complexidade alta se for muitos componentes para ajustar.	EP3
		Modos de apresentação	Modo retrato e paisagem. Referente se a tela é somente retrato ou paisagem sem mudanças (menor complexidade) ou se o aplicativo suporta as modo retrato e paisagem (maior complexidade). Ex.: Calendário do Google, modo retrato é um formato de tela e modo paisagem modifica totalmente a interface e a disposição dos componentes.	EP3, EP4
		Tamanho das Telas do Dispositivo Móvel	Refere-se as limitações da aplicação devido ao tamanho da tela que pode de acordo com o dispositivo móvel utilizado: A aplicação tem limitações porque será usada principalmente por usuários de telefones celulares ou o aplicativo tem uma limitação razoável porque será usado tanto pelo Telefone celular e usuários de tablets ou o aplicativo tem pouca limitação devido	EP2, EP3

Métricas	Categoria	Fator	Descrição	Referência
			ao tamanho da tela porque será usado principalmente pelos usuários de tablets.	
	Limitação de Hardware	Baixo Consumo de Bateria	O consumo de bateria pode variar do aplicativo deve estar preocupado com a otimização de recursos para um menor consumo de bateria, a otimização de recursos para menor consumo de bateria pode ou não existir ou a otimização de recursos para um menor consumo de bateria não deve ser levada em consideração.	EP2, EP4
		Interrupções de Chamada	Se existe tratamento para interrupções de chamada ou não.	EP4
		Fator de Desempenho	O desempenho consiste na otimização de recursos para uma melhor eficiência e tempo de resposta que pode variar: Estar preocupado com a otimização, pode ou não existir ou deve ser levada em consideração.	EP2, EP4
	Portabilidade de	Portabilidade entre sistemas operacionais	Se existe ou não uma portabilidade entre os sistemas operacionais.	EP3, EP5
		Compatibilidade entre emuladores e dispositivos móveis	Se foi ou será realizado o teste em emuladores e dispositivos móveis.	EP6
	Tecnologia	Realidade aumentada	Se o aplicativo usa esse tipo de tecnologia ou não.	EP3
		Sistema de Rastreamento	Ajuda a encontrar um dispositivo perdido. Se a aplicação possui ou não.	EP3
	Requisitos Não-Funcionais	Nível de Colaboração	O grau de relacionamento em que vários agentes participam em uma única instância do aplicativo. Exemplo: número de dispositivos usados na cerca virtual que delimitam um rebanho de gados. De acordo com o número de colaborações a escala varia de: o aplicativo deve exigir a colaboração máxima ou o aplicativo deve exigir colaboração razoável ou a aplicação exigirá uma colaboração mínima.	EP8
		Tempo de vida da instância da aplicação	O período de tempo em que uma instância da aplicação é executada. De acordo com o número do tempo de vida o aplicativo deve exigir um tempo de vida máxima ou o aplicativo deve exigir um tempo de vida razoável ou a aplicação exigirá um tempo de vida mínima.	EP8
		Consciência de Localização Absoluta	A localização é uma posição em um sistema de coordenadas geográficas.	EP8

Métricas	Categoria	Fator	Descrição	Referência
			Aplicações que exigem conhecimento da localização absoluta usam tipicamente métodos como GPS ou triangulação de sinal. Se a aplicação possui ou não.	
		Consciência de Espaço	Edifícios, shoppings e etc - Considera se o dispositivo está dentro da área de limite definido de lugares. Se a aplicação possui ou não.	EP8
		Restrição de Tempo de Resposta	O aplicativo aceita atraso na saída ou requer tempo real. De acordo com Restrição de Tempo de Resposta o aplicativo deve exigir um tempo de resposta com tempo real com precisão máxima. Consiste na existência dessa preocupação na restrição do tempo de resposta ou não.	EP8
		Consciência de Proximidade	Consciência de proximidade (sensibilidade de aproximação). De acordo com a Consciência de proximidade o aplicativo deve exigir uma sensibilidade de aproximação com precisão ou não.	EP8
		Consciência de Eventos	Conhecimento de eventos no ambiente da aplicação. Eventos de nível superior podem ser extraídos da ocorrência de outros eventos. Se a aplicação possui ou não.	EP8
		Objetivo do Aplicativo	(Entretenimento, Operacional ou outros)	EP8
		Consciência de Transição	O grau em que o comportamento de uma aplicação depende do conhecimento de transições entre espaços. Se a aplicação possui ou não.	EP4, EP8
		Consciência de Objetos	Consciência de outros objetos no sistema e seu estado. Exemplo: Um livro eletrônico de laboratório de química permite que os químicos gravem automaticamente e não instrutivamente experiências para análises posteriores. Se a aplicação possui ou não.	EP8
		Publicidade (anúncios)	Se a aplicação possui anúncios ou não.	EP3
		Consciência operacional	O comportamento de uma aplicação depende da conscientização da história do sistema, como o contexto de um fluxo de trabalho. Se a aplicação possui ou não.	EP8

Métricas	Categoria	Fator	Descrição	Referência
		Centricidade	Coleta de dados ou execução de ações em resposta ao ambiente. Se a aplicação possui ou não.	EP8
		Nível de Segurança	A segurança da aplicação pode variar de baixa, média ou alta.	EP4
		Nível de Qualidade dos produtos	A qualidade do produto pode variar de baixa, média e alta qualidade.	EP6

4.4 DISCUSSÃO DOS RESULTADOS

Este mapeamento sistemático visou identificar modelos de estimativa e fatores preditores de esforço no contexto do desenvolvimento de aplicativos móveis. Os resultados mostraram que existem vários estudos propondo modelos e fatores preditores. No entanto, ainda existem algumas lacunas que podem ser exploradas por estudos futuros:

- Não houve consenso sobre quais fatores preditores de esforço utilizar no processo de estimativa, ou seja, apenas alguns fatores são comuns em vários estudos. Logo, uma investigação mais detalhada sobre os preditores de esforço é justificada;
- Apenas 1 estudo apresentou uma comparação entre dois modelos (Ferrucci *et al.*, 2015b), o que dificulta a definição do modelo mais adequado para o contexto de aplicativos móveis. Outro fator limitante é a indisponibilidade dos modelos em um número considerável de estudos, o que dificulta a realização desse tipo de comparação;
- Quando se trata dos modelos de estimativa mais utilizados, os resultados mostraram que há uma tendência dos pesquisadores em proporem modelos do tipo técnicas algorítmicas para estimar projetos de aplicativos móveis.
- Alguns modelos foram avaliados experimentalmente, de 11 modelos encontrados, apenas 7 foram avaliados. Dos que foram avaliados, apenas 1 foi analisado qualitativamente. Segundo Usman *et al.* (2014), a não medição da precisão da previsão dos modelos pode implicar no não interesse dos pesquisadores e profissionais no uso dos modelos, uma vez que não conseguiram fornecer evidência de quão adequadas são os modelos de estimativa de esforço.
- Poucos modelos estimam o tamanho e esforço conjuntamente. De 11 modelos encontrados, apenas 3 foram propostos. Modelos que estimam o tamanho e esforço em conjunto torna mais simples a estimativa.

- Atualmente, não há diretrizes para os profissionais decidirem qual modelo de estimativa ou fatores preditores de esforço escolherem em um determinado projeto de aplicativo móvel.

Apesar da maioria dos modelos de estimativa utilizarem fatores preditores de esforço, ainda há espaço para mais pesquisas nesta área, a fim de melhorar a qualidade dos modelos de estimativa.

4.5 CONSIDERAÇÕES SOBRE O CAPÍTULO

Neste mapeamento sistemático, foram analisadas as publicações referentes aos modelos de estimativa de tamanho, esforço e fatores preditores de esforço de projetos de aplicativos móveis. A partir de um conjunto inicial de 487 publicações, foram selecionadas 18 publicações, resultando em 6 modelos para estimar tamanho, 3 modelos de esforço, 3 modelos para estimar ambos em conjunto e 66 fatores preditores de esforço.

Embora vários estudos tenham sido conduzidos com relação à estimativa de tamanho e esforço para projetos de aplicativos móveis, os resultados revelaram que não há um modelo contendo todos os fatores citados na literatura que ajude os profissionais a selecionarem preditores de esforço.

CAPÍTULO 5 – UMA TAXONOMIA DE PREDITORES DE ESFORÇO PARA PROJETOS DE APLICATIVOS MÓVEIS

Este capítulo apresenta a taxonomia proposta, classificação da base de conhecimento de preditores de esforço, juntamente com os resultados das tarefas das fases de planejamento, identificação e extração, design e construção e avaliação.

5.1 INTRODUÇÃO

Tendo em vista o surgimento de diferentes preditores encontrados na literatura e na indústria, não foi identificada uma classificação para esses preditores existentes. Com isso, foi proposta uma taxonomia de preditores de esforço que classifica e organiza os preditores consolidados na literatura e os apontados pelos profissionais na indústria para estimar projetos de aplicativos móveis. Para projetar a taxonomia, utilizou-se o método proposto por Usman *et al.* (2017), pois o método é uma revisão do método proposto por Bayona-Oré *et al.* (2014) que foi o primeiro método de construção de uma taxonomia (Usman *et al.*, 2017) e também o método de Usman foi utilizado na construção de uma taxonomia de preditores de esforço para projetos web (Britto *et al.*, 2017). O método de Usman *et al.* (2017) possui 13 atividades que foram agrupadas em 4 fases que serão detalhadas nas seções seguintes.

5.2 PLANEJAMENTO DA TAXONOMIA

Nesta fase, foram realizadas 6 atividades (B1, B2, B3, B4, B5 e B6). A atividade B1 consiste na definição da área de conhecimento associada à taxonomia, no caso da taxonomia proposta é a área de gerenciamento de projetos. A atividade B2 consiste em descrever os objetivos da taxonomia, que é definir um conjunto de categorias que permite classificar os preditores de esforço para projetos de aplicativos móveis (métricas de tamanho e fatores de custo) relatados na literatura e na indústria. A atividade B3 consiste na seleção e descrição do assunto a ser classificado e o assunto da taxonomia projetada são preditores de esforço para projetos de aplicativos móveis.

A atividade B4 consiste em selecionar a estrutura do tipo de classificação e para projetar a taxonomia aqui apresentada, escolheu-se a hierarquia (Kwasnik, 1999). Esta escolha foi feita porque existe um estado da arte maduro de estimativa de esforço para projetos de aplicativos móveis que permite projetar uma taxonomia hierárquica com categorias bem definidas (Britto *et al.*, 2017). A Estrutura hierárquica relaciona categorias e

subcategorias em uma relação pai-filho (Kwasnik, 1999), que é também o caso da taxonomia proposta (por exemplo, as linhas de código são uma métrica de tamanho que é um preditor de esforço para projetos de aplicativos móveis).

A atividade B5 consiste em selecionar o procedimento do tipo de classificação e para ajudar nesse processo foi realizado um estudo qualitativo com especialistas em estimativa de esforço para projetos de aplicativos móveis na indústria. Esse conhecimento ajudou a identificar a melhor estrutura para projetar a taxonomia proposta, abordagem apropriada para assegurar a exclusividade entre categorias em uma taxonomia hierárquica (Wheaton, 1968). A última atividade é a B6 que consiste em identificar as fontes de dados que foram coletadas por meio de um mapeamento sistemático da literatura (De Souza e Conte, 2017) e estudo qualitativo (De Souza e Conte, 2018) com profissionais na indústria que estimam aplicativos móveis.

5.3 IDENTIFICAÇÃO E EXTRAÇÃO

Nesta fase, foram realizadas 2 atividades (B7 e B8). A atividade B7 consiste na extração de todos os termos vindos das fontes de dados e como resultado dessa atividade extraíram-se 120 preditores (24 métricas de tamanho e 96 fatores de custo) (Tabela 3.3 e Tabela 4.8). A atividade B8 que é o controle de terminologia e para realizar essa atividade, utilizou-se o conhecimento de estimativa de esforço de aplicativos móveis das autoras e as seguintes ações para melhorar a terminologia associada aos preditores de esforço extraídos:

1. Remoção de duplicados.
2. Sempre que identificado um caso em que um preditor foi representado por termos diferentes em diferentes estudos, uma unificação terminológica foi utilizada. Para isso, foi realizado um processo inspirado em Delphi [20], pelo qual as autoras escolheram de forma independente um termo para nomear o preditor em questão; as deliberações foram feitas para chegar a um consenso; este processo foi repetido até o consenso.
3. A terminologia associada a muitos preditores foi ligeiramente alterada para melhorar a legibilidade (exemplo: os termos eram APIs do Cliente e Uso de APIs nativas (Maior Esforço), os termos foram unificados resultando no termo Nível de Utilização de APIs); em muitos casos, foi quase impossível entender o propósito de um preditor sem um texto abrangente para descrevê-lo.

Como resultado das ações enumeradas acima, o número de preditores foi reduzido de 120 (24 métricas de tamanho e 96 fatores de custo) para 108 (23 métricas de tamanho e 85 fatores de custo). Os termos que foram unificados são apresentados na Tabela 5.1, resultado do controle terminológico e os termos finais são detalhados na Seção 5.5.

Tabela 5.1: Resultados do Controle de Terminologia

Termo Final	Termos Originais
Número de Requisitos	Número de Requisitos Funcionais, Quantidade de Requisitos
Quantidade de elementos da interface gráfica	Quantidade de elementos da interface gráfica, Qualidade visual.
Nível de Experiência no Desenvolvimento em Projetos Anteriores	Experiência no desenvolvimento em projetos anteriores, Nível de Experiência no Desenvolvimento.
Nível de Competências dos usuários e clientes	Nível de Competências dos usuários e clientes, Nível de Competência do Cliente.
Experiência da equipe no contexto da aplicação	Experiência da equipe no contexto da aplicação, Conhecimento do Problema.
Nível de Utilização de APIs	APIs do Cliente, Uso de APIs nativas (Maior Esforço).
Tipo de Conectividade do Dispositivo Móvel	Tipo de Conectividade do Dispositivo Móvel, Tipos de Conectividade na rede.
Modo de Apresentação do Aplicativo	Modo de Apresentação do Aplicativo, Modos de apresentação.
Limitação de Energia do Dispositivo Móvel	Limitação de Energia do Dispositivo Móvel, Baixo Consumo de Bateria.

5.4 DESIGN E CONSTRUÇÃO

A partir dos termos extraídos através das atividades B7 e B8, foram realizadas 4 atividades (B9, B10, B11 e B12) cujo objetivo era identificar, descrever as dimensões e categorias da taxonomia.

A atividade B9 consiste na identificação e descrição das dimensões do design da taxonomia, foi identificada apenas uma dimensão, denominada preditores de esforço de projetos de aplicativos móveis, esta dimensão é a categoria pai de todas as subcategorias. Isso foi usado para classificar o corpo de conhecimento em métricas de tamanho e fatores de custo (Seção 5.5).

A atividade B10 consiste em identificar e descrever as categorias de cada dimensão e para identificar as principais categorias. A taxonomia possui 11 categorias herdadas da taxonomia de Britto *et al.* (2017) (Figura 5.1), pois além das categorias estarem bem definidas e validadas, foram consideradas relevantes e semelhantes às que foram identificadas durante os estudos do qualitativo com profissionais que estimam projetos de aplicativos móveis na indústria (De Souza e Conte, 2017) e um mapeamento sistemático da literatura sobre estimativa de projetos de aplicativos móveis descritos nos Capítulo 3 e Capítulo 4.

A atividade B11 consiste em identificar e descrever os relacionamentos das categorias. Como mencionado anteriormente, foi selecionada a hierarquia como o tipo de estrutura de classificação, pois tem uma influência direta sobre o tipo de relação entre as categorias (Kwasnik, 1999). Portanto, utilizou-se a relação “pai-filho” na taxonomia para relacionar categorias e subcategorias, o que resultou na taxonomia de preditores de esforço de projetos de aplicativos móveis exibida na Figura 5.2.

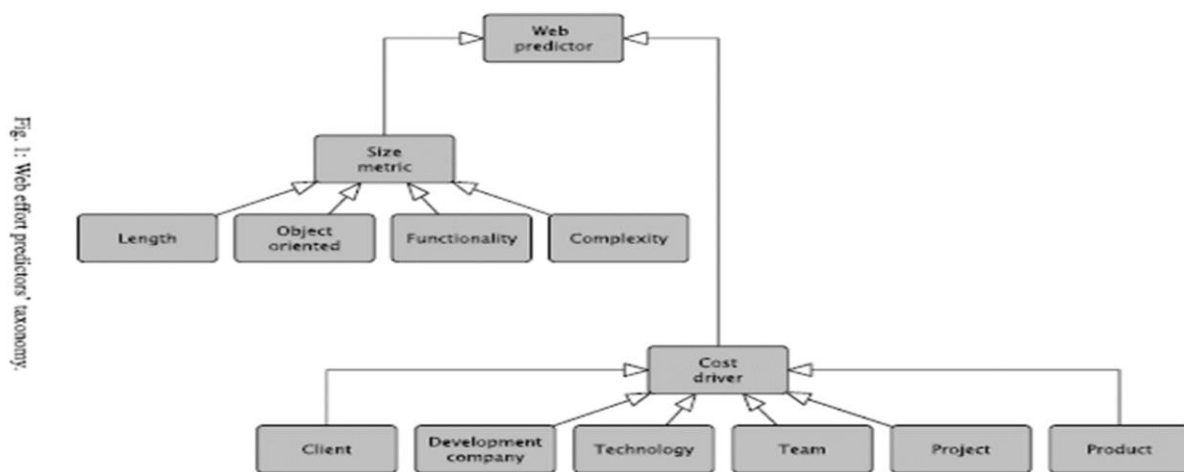


Fig. 1: Web effort predictors' taxonomy.

Figura 5.1: Categorias da Taxonomia de Britto *et al.* (2017)

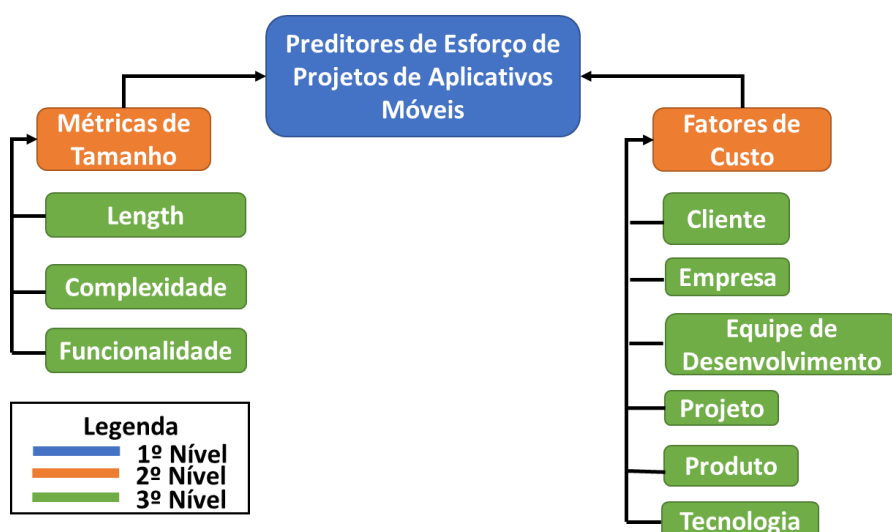


Figura 5.2: Taxonomia de Preditores de Esforço de Projetos de Aplicativos Móveis

A taxonomia tem 3 níveis hierárquicos: O primeiro nível contém a raiz da taxonomia, que é o assunto a ser classificado (preditores de esforço de projetos de aplicativos móveis). O

segundo nível tem duas categorias: Métricas de Tamanho e Fatores de Custo. As categorias do segundo e terceiro nível hierárquico são definidas da seguinte forma:

- **Métricas de tamanho:** São variáveis independentes baseadas nos elementos de composição (telas, número de requisitos funcionais e outros), funcionalidade e complexidade do aplicativo.
 - *Length* - Esta categoria inclui métricas que medem diretamente o tamanho de aplicativos móveis com base na contabilização de seus elementos de composição (número de atores do caso de uso, número de classes do código fonte e outros).
 - *Funcionalidade* - Esta categoria engloba métricas que são medidas a partir de um método ou modelo baseadas nas características e funções do aplicativo móvel (COSMIC, Análise por Pontos de Função e outros).
 - *Complexidade* - Esta categoria abrange métricas que indiretamente medem o tamanho de aplicativos móveis baseados na dificuldade associada aos seus elementos de composição (complexidade ciclomática e outros).
- **Fatores de custo:** São variáveis independentes que impactam o esforço necessário para desenvolver um projeto, mas não caracterizam o tamanho do aplicativo (número de desenvolvedores, ferramentas e outros).
 - *Cliente* - Esta categoria engloba fatores de custo associados ao cliente que solicita o desenvolvimento de um aplicativo móvel.
 - *Empresa* - Esta categoria engloba os fatores de custo associados à empresa que desenvolve aplicativos móveis.
 - *Produto* - Esta categoria engloba fatores de custo associados às características do produto e as restrições por serem executados pelos dispositivos móveis.
 - *Projeto* - Esta categoria engloba os fatores de custo associados à configuração (planejamento, acompanhamento do projeto, processo e outros) de um projeto de aplicativo móvel.
 - *Equipe de Desenvolvimento* - Esta categoria engloba fatores de custo que estão associados com a equipe de desenvolvimento responsável pelo desenvolvimento de um aplicativo móvel.
 - *Tecnologia* - Esta categoria engloba fatores de custos associados às tecnologias (Linguagem de programação, ferramentas, plataformas) exigidos em um projeto de aplicativo móvel.

Quanto a completude dos níveis, o segundo nível hierárquico está completo de acordo com pesquisas em engenharia de software (De Souza e Conte, 2017) (Britto *et al.*, 2017): métricas de tamanho e fatores de custo. Acredita-se que o terceiro nível hierárquico é o único que pode ser atualizado para evoluir a taxonomia. Além disso, caso um novo preditor de esforço precise ser classificado, o pesquisador deverá consultar a descrição das categorias para compreender e aplicar a taxonomia (resultado de B12 que é a definição de diretrizes para usar e atualizar a taxonomia).

5.5 CLASSIFICAÇÃO DA BASE DE CONHECIMENTO DE PREDITORES DE ESFORÇO DE PROJETOS DE APLICATIVOS MÓVEIS

Nesta subseção, uma base de conhecimento de 108 preditores de esforço de projetos de aplicativos móveis foi organizada, onde os preditores de esforço foram classificados nas 12 categorias da taxonomia. A base de conhecimento foi construída por meio do mapeamento sistemático da literatura e estudo qualitativo. Importante ressaltar que a taxonomia proposta vai além de ser específica só para aplicativos móveis, pois se aplica para projetos em geral, é na interpretação de como considerar o preditor na estimativa que vem a diferenciação para aplicativos móveis e em demais projetos. A seguir apresenta a classificação dos preditores detalhados em Tabela 5.2, Tabela 5.3, Tabela 5.4, Tabela 5.5, Tabela 5.6, Tabela 5.7, Tabela 5.8. Observe que a coluna "F" apresenta as frequências de cada preditor em todas as tabelas. As legendas das referências da literatura EP1 a EP15 estão no Apêndice B e as legendas P1 a P8 são os participantes do estudo qualitativo do Capítulo 3.

5.5.1 Métricas de tamanho

Esta categoria possui 3 subcategorias de métricas relacionadas ao tamanho do projeto: Length, Complexidade e Funcionalidade, essas categorias são aplicáveis para projetos em geral, é na interpretação de como considerar a métrica que vem a diferenciação para aplicativos móveis. A Tabela 5.2 contém as subcategorias e suas respectivas 23 métricas que representam cerca de 22% da taxonomia. A seguir contém exemplos de métricas que foram identificadas apenas no MSL (que possui referências que começam com EP), qualitativo (que possui referências que começam com P) e ambos estudos (EP e P) respectivamente.

- *Número de Atores do Caso de Uso:* Número total de Atores do caso de uso do projeto pode ser descritivo ou diagrama (EP1).

- *Quantidade de dados*: Quantidade de Dados que o aplicativo contém, por exemplo, apps de comércio e e-commerce possuem os dados dos produtos e dos clientes (P1, P3, P4).
- *Quantidade de elementos da interface gráfica*: Número de diferentes tipos de elementos que a interface gráfica contém, por exemplo, têm aplicativos que demandam logotipo, marca corporativa, navegação, animações, efeitos diferentes e tem preocupações com a usabilidade (EP3, P4).

Tabela 5.2: Fatores relacionados a Métricas de Tamanho

Subcategoria	Preditor	Descrição	Referências	F
Length	Número de Requisitos Funcionais	Número de requisitos funcionais do projeto.	EP1, P7	2
	Número de Atores do Caso de Uso	Número total de Atores do caso de uso do projeto pode ser descritivo ou diagrama.	EP1	1
	Número de Casos de Uso	Número total de Casos de Uso pode ser descritivo ou diagrama.	EP1	1
	Número de Classes do documento de requisitos	Número total de Classes do projeto, pode ser descritivo ou diagrama.	EP1	1
	Número de Diagramas de Sequência	Número total de Diagramas de Sequência do projeto.	EP1	1
	Quantidade de Dados	Quantidade de Dados que o aplicativo contém, por exemplo, apps de comércio e e-commerce possuem os dados dos produtos e dos clientes.	P1, P3, P4	3
	Número de Classes do Código Fonte	Número total de Classes contidas no código fonte do aplicativo móvel	EP1	1
	Número de Arquivos do Código Fonte	Número total de Arquivos do projeto	EP1	1
	Número de Métodos (incluindo os herdados)	Número total de métodos implementados contidos no código fonte do aplicativo móvel	EP1	1
	Número total de linhas	Número total de linhas do código fonte do aplicativo móvel	EP1	1
	Número de linhas contendo código fonte	Número total de linhas contendo somente o código fonte do aplicativo móvel	EP1	1
	Número de linhas contendo comentários	Número total de linhas contendo somente comentários do código fonte do aplicativo móvel	EP1	1
	Número de declarações	Número total de declarações contidas no código fonte do aplicativo móvel	EP1	1

	Tamanho do código (kB)	Tamanho em kB do aplicativo móvel	EP11	1
	Número de arquivos XMI sobre elementos gráficos de um aplicativo para dispositivos móveis	Número de arquivos XMI sobre elementos gráficos (por exemplo, caixa de texto e assim por diante) de um aplicativo para dispositivos móveis.	EP1	1
	Quantidade de elementos da interface gráfica	Número de diferentes tipos de elementos que a interface gráfica contém, por exemplo, têm aplicativos que demandam logotipo, marca corporativa, navegação, animações, efeitos diferentes e tem preocupações com a usabilidade.	EP3, P4	2
Complexidade	Complexidade Ciclomática	Medida de complexidade, onde é considerado os caminhos independentes que o código do projeto pode tomar.	EP1	1
	Nível de Complexidade do Contexto do Aplicativo	Se o aplicativo precisa que seja estudado o contexto que o aplicativo será inserido. Exemplo: bolsa de valores e bancos. Esse nível de complexidade pode variar de baixo, médio ou alto.	P2	1
Funcionalidade	COSMIC	Pontos de função calculados usando o método COSMIC	EP3, EP4, EP11, EP12, EP13, EP14, EP15	7
	IFPUG	Pontos de função calculados usando o método IFPUG	EP2	1
	FISMA	Pontos de função calculados usando o método FISMA	EP02	1
	COSMIC with UML model mapping	Pontos de função calculados usando o método proposto por Abdullah <i>et al.</i> 2014.	EP14, EP15	2
	Tempo total para fazer as telas do aplicativo	Indica quantidade de tempo gasta para cada tela do aplicativo (por exemplo, tempo para fazer ícones e gifs e outros).	P1, P3	2

5.5.2 Fatores de custo

Esta categoria possui 6 subcategorias de preditores relacionados ao custo de projeto de aplicativo móvel: Cliente, Empresa, Equipe de Desenvolvimento, Produto, Projeto e Tecnologia, essas categorias são aplicáveis para projetos em geral, é na interpretação de como considerar a métrica que vem a diferenciação para aplicativos móveis, apenas a categoria de produto é específica para aplicativos móveis, as demais podem ser aplicadas para projetos em geral. A seguir são detalhadas as subcategorias e seus respectivos preditores relacionados a cada subcategoria que possuem um total de 85 preditores, representando cerca de 78% da taxonomia.

- **Cliente**

Esta subcategoria abrange os fatores de custo relacionados com o cliente que solicita o desenvolvimento de um aplicativo móvel. A Tabela 5.3 contém esses 5 fatores no total, a seguir contém exemplos de fatores que foram identificados apenas no MSL, qualitativo e ambos estudos respectivamente.

- *Nível de Participação do Cliente:* Se o cliente valida as especificações do projeto, se o cliente tem ciência dos riscos do projeto, se o cliente participa da reunião de concepção do produto, esses são exemplos de participação. A participação pode variar de baixa (quando o cliente só participa das reuniões de concepção), média (quando o cliente participa das reuniões e valida as especificações) ou alta (quando o cliente está presente durante todo o desenvolvimento do projeto) (P1, P2, P3, P4, P5, P6, P7, P8).
- *Nível de Competências dos usuários e clientes:* Envolve conhecer do negócio e saber passar as informações. A competência do Cliente pode ser baixa (quando o cliente passa as informações erradas), média (quando o cliente sabe das informações, porém não conhece a parte da implementação do projeto) ou alta (quando o cliente tem conhecimento do problema e conhece a parte da implementação do projeto) (EP6, P1, P2, P4, P6, P8).

Tabela 5.3: Fatores relacionados ao Cliente

Preditor	Descrição	Legenda	F
Nível de Competências dos usuários e clientes	Envolve conhecer do negócio e saber passar as informações. A competência do Cliente pode ser baixa (quando o cliente passa as informações erradas), média (quando o cliente sabe das informações, porém não conhece a parte da implementação do projeto) ou alta (quando o cliente tem conhecimento do problema e conhece a parte da implementação do projeto).	EP6, P1, P2, P4, P6, P8	6
Expectativas dos usuários e clientes	Situação de quem espera a ocorrência de algo, ou sua probabilidade de ocorrência, em determinado momento. Se o cliente tem expectativa sobre algum aspecto do projeto ou não.	EP5	1
Autonomia do cliente na tomada de decisão	Se o cliente tem autonomia para tomar uma decisão no projeto ou não.	P7	1
Nível de Comunicação com o Cliente	Envolve a comunicação dos responsáveis pelo projeto com o cliente que pode variar de raramente consegue comunicação, quase sempre tem comunicação ou sempre tem comunicação com o cliente.	P1, P2, P3, P6	4

Nível de Participação do Cliente	Se o cliente valida as especificações do projeto, se o cliente tem ciência dos riscos do projeto, se o cliente participa da reunião de concepção do produto, esses são exemplos de participação. A participação pode variar de baixa (quando o cliente só participa das reuniões de concepção), média (quando o cliente participa das reuniões e valida as especificações) ou alta (quando o cliente está presente durante todo o desenvolvimento do projeto).	P1, P2, P3, P4, P5, P6, P7, P8	8
----------------------------------	--	--------------------------------	---

- **Empresa**

Esta subcategoria abrange os fatores de custo relacionados com a empresa responsável pela execução de um projeto de aplicativo móvel. A Tabela 5.4 contém 7 fatores, a seguir contém exemplos de fatores que foram identificados apenas no MSL e estudo qualitativo respectivamente.

- *Competência de Fornecedores:* Competência está relacionada a cumprir os prazos e qualidade da entrega da matéria prima. Consiste se são competentes ou não (EP6).
- *Disponibilidade da Infraestrutura para falar com o Cliente:* Se existe ou não uma infraestrutura para falar com o cliente (P6, P8).

Tabela 5.4: Fatores relacionados à Empresa

Preditor	Descrição	Legenda	F
Nível de Infraestrutura Ferramental da Empresa	Disponibilidade de salas para trabalho e ferramental que pode variar de sempre disponível, raramente e nunca.	P8	1
Nível de Certificação da empresa	Se a empresa tem ou não certificações.	EP6	1
Competência de Fornecedores	Competência está relacionada a cumprir os prazos e qualidade da entrega da matéria prima. Consiste se são competentes ou não.	EP6	1
Envolvimento dos concorrentes	Se existe envolvimento dos concorrentes ou não.	EP6	1
Nível de Disponibilidade da Internet da Empresa	Consiste no nível de estabilidade/disponibilidade de acesso à internet da empresa que pode variar de nunca estável, às vezes estável e estável.	P3, P7	2
Disponibilidade dos Dispositivos para Teste	Se tem ou não disponibilidade dos dispositivos para teste do aplicativo.	P7	1
Disponibilidade da Infraestrutura para falar com o Cliente	Se existe ou não uma infraestrutura para falar com o cliente.	P6, P8	2

- **Equipe de Desenvolvimento**

Esta subcategoria abrange os fatores de custo relacionados com a equipe de desenvolvimento responsável pela execução de um projeto de aplicativo móvel. A Tabela 5.5

contém 10 fatores, a seguir alguns exemplos de fatores identificados apenas no MSL, qualitativo e ambos estudos respectivamente:

- *Definição de papéis das equipes:* Se os papéis (desenvolvedor, tester e outros) dos membros da equipe estão definidos ou não (EP5).
- *Nível de Experiência com a Tecnologia:* O nível de experiência com a tecnologia que o projeto precisa pode variar de nenhuma, baixa (fez alguns exercícios de aprendizado), média (fez alguns projetos com a tecnologia) ou alta (fez muitos projetos com a tecnologia) (P1, P2, P3, P4, P5, P6, P7, P8).
- *Nível de Comunicação da equipe:* Grau de comunicação entre os membros da equipe que pode variar de baixo (quando se comunicam apenas em algumas reuniões), médio (quando se comunicam em todas as reuniões) ou alto (quando a comunicação é frequente) (EP5, EP6, P2).

Tabela 5.5: Fatores relacionados à Equipe de Desenvolvimento

Preditor	Descrição	Referências	F
Nível de Experiência com a Tecnologia	O nível de experiência com a tecnologia que o projeto precisa pode variar de nenhuma, baixa (fez alguns exercícios de aprendizado), média (fez alguns projetos com a tecnologia) ou alta (fez muitos projetos com a tecnologia).	P1, P2, P3, P4, P5, P6, P7, P8	8
Experiência da equipe no contexto da aplicação	Tem-se experiência ou não no contexto da aplicação, por exemplo, conhecimento nas áreas e aplicações empresariais.	EP1, EP5, P2	3
Nível de Comunicação entre a equipe	Grau de comunicação entre os membros da equipe que pode variar de baixo (quando se comunicam apenas em algumas reuniões), médio (quando se comunicam em todas as reuniões) ou alto (quando a comunicação é frequente).	EP5, EP6, P2	3
Nível de Experiência no Desenvolvimento em Projetos Anteriores	O nível de experiência no desenvolvimento em projetos anteriores pode variar de nenhuma (quando nunca trabalharam com um projeto), baixa (quando trabalharam em um projeto), média (quando trabalharam em alguns projetos) ou alta (quando trabalharam em muitos projetos).	EP1, EP5, P1, P2, P3, P5, P7	7
Quantidade de horas dedicada de cada pessoa do projeto	Consiste na quantidade de horas que cada pessoa trabalha no projeto.	P4	1
Definição de papéis das equipes	Se os papéis (desenvolvedor, tester e outros) dos membros da equipe estão definidos ou não.	EP5	1

Nível de Conhecimento da Competência da equipe	Quando se conhece a competência da equipe que está participando do projeto. Esse conhecimento da competência da equipe pode varia de nenhuma (quando não se trabalhou com os membros da equipe no desenvolvimento de um projeto), baixa (trabalharam em apenas um projeto), média (trabalharam em alguns projetos) ou alta (trabalharam em muitos projetos).	P1, P2	2
Quantidade de Pessoas Trabalhando no Projeto	Consiste na quantidade de pessoas trabalhando no projeto.	P2, P4, P6	3
Curva de Aprendizagem da Equipe	Consiste no tempo necessário em horas para a equipe aprender algum recurso, contexto do projeto ou tecnologia necessária para o projeto.	P1, P6	2
Nível de Engajamento da equipe	Engajamento da equipe com o projeto pode variar de baixo (quando a equipe é dispersa), médio (quando a equipe se disponibiliza para reunir) ou alto (quando a equipe leva o projeto a frente e a equipe passa a confiança que está envolvido em solucionar um problema) .	P2, P4, P6	3

- **Produto**

Esta subcategoria engloba fatores de custo associados às características do produto e as restrições por serem executados pelos dispositivos móveis. A Tabela 5.6 contém esses 35 fatores no total e a seguir contém exemplos de fatores identificados apenas no MSL, qualitativo e ambos estudos respectivamente.

- *Portabilidade entre sistemas operacionais:* Capacidade do aplicativo ser compilado ou executado em diferentes sistemas operacionais. Consiste na existência ou não da portabilidade entre os sistemas operacionais (EP3, EP5).
- *Integração com as redes sociais:* Algumas funcionalidades do aplicativo usarem os recursos das redes sociais como por exemplo login, compartilhamento de conteúdo, convidar amigos e etc. Este fator é medido no uso ou no não uso das redes sociais (P3, P4, P5).
- *Modo de Apresentação do Aplicativo:* Se há um tratamento ou não diferenciado para a tela que está na horizontal ou na vertical do dispositivo móvel (EP3, EP4, P1, P6, P7, P8).

Os fatores sinalizados com “*” têm a mesma ponderação proposto por FISMA (Forselius, 2004), mas apenas variando de - a +: ° (+) = [1,05] Boa situação, melhor do que em circunstâncias do caso médio; ° (+ / -) = [1,0] normal de Situação; ° (-) = [0,95] situação ruim, as circunstâncias piores do que no caso da média.

Tabela 5.6: Fatores relacionados ao Produto

Preditor	Descrição	Legenda	F
----------	-----------	---------	---

Largura de banda da rede*	(-) A aplicação deve exigir uma largura de banda máxima. (+/-) A aplicação deve exigir uma largura de banda razoável. (+) A aplicação necessita de uma largura de banda mínima.	EP2, EP4, EP5	3
Comutação de Acesso a Rede*	(-) A aplicação deve trabalhar <i>offline</i> e possuir um sincronismo. (+/-) A aplicação deve trabalhar <i>offline</i> e não é necessário possuir um sincronismo. (+) A aplicação não deve trabalhar <i>offline</i> .	EP2, EP4	2
Tipo de Conectividade do Dispositivo Móvel*	(-) A aplicação deve possuir o máximo de predisposição para uso de conexões como: 3G, Wifi, Wireless, Bluetooth, Infravermelho e outros. (+/-) A aplicação deve possuir razoável predisposição para uso de conexões como: 3G, Wi-fi e Wireless. (+) A aplicação deve possuir apenas uma predisposição para uso de conexões, que pode ser: 3G, Wi-fi, Wireless, Bluetooth, Infravermelho ou outros.	EP2, EP4, P1	3
Requisição para a Internet	Se for uma aplicação cliente-servidor tem que estimar os caminhos alternativos, caso a requisição retornar com falha ou sucesso. Utiliza-se ou não esse recurso, se utilizar, estimar os caminhos alternativos também.	P5	1
Formato de Comunicação com o Servidor	É um formato de padrão aberto que utiliza texto legível a humanos para transmitir objetos de dados consistindo de pares atributo-valor. Esse formato de comunicação com o servidor pode ser Json ou XML.	P4, P5	2
Dependências com outros sistemas	Se existe uma dependência com outro sistema do cliente ou outra API ou não existe dependência.	P5	1
Sincronização com banco de dados	Se a aplicação utiliza bando de dados ou não.	EP6	1
Variação de Entrada da Interface Gráfica*	(-) A aplicação deve ter interfaces de entrada para tela sensível ao toque, voz, vídeo, teclado e outros. (+/-) A deve ter interfaces de entrada padrão para teclado. (+) A deve ter qualquer um dos tipos de interfaces, tais como: tela de toque, voz, vídeo, teclado ou outros.	EP2	1
Modo de Apresentação do Aplicativo	Se há um tratamento ou não diferenciado para a tela que está na horizontal ou na vertical do dispositivo móvel.	EP3, EP4, P1, P6, P7, P8	6
Tamanho das Telas do Dispositivo Móvel*	(-) A aplicação possui limitação, devido ao tamanho da tela, pois a mesma será utilizada na sua maioria por usuários de aparelhos celulares. (+/-) A aplicação possui razoável limitação, devido ao tamanho da tela, pois a mesma será utilizada tanto por usuários de aparelhos celulares quanto de tablets. (+) A aplicação possui pouca limitação, devido ao tamanho da tela, pois a mesma será utilizada por em sua maioria por usuários de tablets.	EP2, EP3, P2, P6	4
Limitação de Energia do Dispositivo Móvel*	Dependendo dos tipos de aplicativo (jogos, por exemplo) há uma preocupação na otimização do uso dos recursos por causa da limitação de energia do dispositivo. Essa preocupação pode variar: (-) O aplicativo deve estar preocupado com a otimização de recursos para um menor consumo de bateria. ◦ (+/-) A otimização de recursos para menor consumo de bateria pode ou não existir. ◦ (+) A otimização de recursos para um menor consumo de bateria não deve ser levada em consideração.	EP2, EP4, P1, P3, P4	4
Interrupções de Chamada	Quando uma aplicação é interrompida ou pausada para executar uma chamada telefônica. Consiste na existência dessa preocupação na interrupção de chamadas ou não.	EP4, P2, P3	3

Fator de Desempenho*	<ul style="list-style-type: none"> ◦ (-) A aplicação deve se preocupar com a otimização de recursos por uma melhor eficiência e tempo de resposta. ◦ (+/-) A otimização de recursos para uma melhor eficiência e tempo de resposta pode ou não existir. ◦ (+) A otimização de recursos para melhorar a eficiência e o tempo de resposta não deve ser tomado em consideração. 	EP2, EP4	2
Sistema Operacional do Dispositivo Móvel	Leva-se em consideração ou não a versão do sistema operacional e o sistema operacional: Android e IOS.	P1, P6, P7, P8	4
Portabilidade entre sistemas operacionais	Capacidade do aplicativo ser compilado ou executado em diferentes sistemas operacionais. Consiste na existência ou não da portabilidade entre os sistemas operacionais.	EP3, EP5	2
Compatibilidade entre emuladores e dispositivos móveis	Se foi/será realizado os testes em emuladores e dispositivos móveis ou não.	EP6	1
Targets	Quantidade de dispositivos que a aplicação será executada.	EP3, EP6, P2, P6, P7	5
Integração com as redes sociais	Algumas funcionalidades do aplicativo usarem os recursos das redes sociais como por exemplo login, compartilhamento de conteúdo, convidar amigos e etc. Este fator é medido no uso ou no não uso das redes sociais.	P3, P4, P5	3
Publicidade (anúncios)	Divulgação de alguma empresa, serviço ou produto. Consiste se a aplicação possui anúncios ou não.	EP3	1
Tempo de vida da instância da aplicação	O período de tempo em que uma instância da aplicação é executada. Um aplicativo pode ser executado de alguns segundos a vários dias ou anos. A longevidade de uma instância da aplicação impacta requisitos de potência e que grau de persistência é necessário a infra-estrutura.	EP8	1
Objetivo do Aplicativo	O objetivo é o tipo de benefício a ser alcançado do aplicativo que pode ser simplesmente entretenimento ou pode ser o aumento do acesso à informação ou desempenho. O objetivo pode afetar, entre outros, a segurança, confiabilidade e qualidade da arquitetura.	EP8	1
Colaboração	O grau de relacionamento em que vários agentes participam em uma única instância do aplicativo. Exemplo: número de dispositivos usados na cerca virtual que delimitam um rebanho de gados.	EP8	1
Consciência de Localização Absoluta	A localização absoluta é uma posição em um sistema de coordenadas geográficas. Aplicações que exigem conhecimento da localização absoluta usam tipicamente métodos como GPS ou triangulação de sinal. Consiste se a aplicação utiliza esse recurso ou não.	EP8	1
Consciência de Espaço	Considera se o dispositivo está dentro da área de limite definido de lugares (Edifícios, shoppings e etc). Consiste se a aplicação utiliza esse recurso ou não.	EP8	1
Restrição de Tempo de Resposta	Se a aplicação tem em tempo real ou é tolerante ao atraso. As aplicações podem aceitar algum atraso na saída ou requerem uma resposta em tempo real. Tempo real geralmente indicam a necessidade de maior largura de banda confiável do que as aplicações tolerantes ao atraso.	EP8	1
Consciência de Proximidade	Sensibilidade de aproximação de um determinado objeto ou lugar. Sensibilização de proximidade em geral indica a necessidade de métodos como sensores. Consiste se a aplicação utiliza esse recurso ou não.	EP8	1

Consciência de Eventos	Conhecimento de eventos no ambiente da aplicação que requer capacidades de sistema. Eventos de nível superior podem ser extraídos da ocorrência de outros eventos. Consiste se a aplicação utiliza ou não esse recurso.	EP8	1
Consciência de Transição	O grau em que o comportamento de uma aplicação depende do conhecimento de transições entre espaços. Consiste se a aplicação utiliza ou não esse recurso.	EP8	1
Consciência de Objetos	Consciência de outros objetos no sistema e seus estados. Exemplo: Um livro eletrônico de laboratório de química permite que os químicos gravem automaticamente e não instrutivamente experiências para análises posteriores. Se a aplicação possui ou não.	EP8	1
Realidade aumentada	A integração das informações virtuais e visualizações do mundo real em tempo real. Se o aplicativo usa esse tipo de tecnologia ou não.	EP3	1
Consciência operacional	O comportamento de uma aplicação depende da conscientização da história do sistema, como o contexto de um fluxo de trabalho. Se a aplicação possui ou não.	EP8	1
Centricidade	Coleta de dados ou execução de ações em resposta ao ambiente. Se a aplicação possui ou não.	EP8	1
Sistema de Rastreamento	Ajuda a encontrar um dispositivo perdido. Se a aplicação possui ou não.	EP3	1
Segurança	Proteção de um conjunto de informações de um indivíduo ou uma organização. Se a aplicação possui ou não.	EP4	1
Qualidade dos produtos	Satisfaz as necessidades do usuário e cliente, atendendo os requisitos do projeto. Se a aplicação possui ou não.	EP6	1

- **Projeto de Aplicativos Móveis**

Esta subcategoria abrange os fatores de custo associados à configuração de um aplicativo móvel. A Tabela 5.7 contém esses 25 fatores e a seguir contém exemplos de fatores identificados apenas do MSL e estudo qualitativo respectivamente.

- *Preparação da gestão de mudança:* Se existem ou não uma preparação para eventuais mudanças que poderão ocorrer durante o projeto é de extrema relevância (EP5).
- *Nível de Documentação do Projeto:* Documentação do projeto é um registro de todas as fases do projeto, quais decisões foram tomadas, como e porquê. Consistem em: não tem documentação, tem pouco detalhado, tem mais detalhado (P2, P8).

Tabela 5.7: Fatores relacionados ao Projetos de Aplicativos Móveis

Preditor	Descrição	Legenda	F
Preparação da gestão de mudança	Se existem ou não uma preparação para eventuais mudanças que poderão ocorrer durante o projeto é de extrema relevância.	EP5	1
Nível de Documentação do Projeto	Documentação do projeto é um registro de todas as fases do projeto, quais decisões foram tomadas, como e porquê. Consistem em: não tem documentação, tem pouco detalhado, tem mais detalhado.	P2, P8	2

Documento de User Interface (UI)	Especificação formal que mostra as decisões de design do aplicativo. Consiste em utilizar ou não Documento de User Interface (UI) para estimar o projeto.	P8	1
Documento de User Experience (UX)	Especificação formal que se concentrando em aspectos hedônicos, como emoções e estímulos do usuário, enquanto interagem com um aplicativo. Consiste em utilizar ou não Documento de User Experience (UX) para estimar o projeto.	P8	1
User Stories	Descrição concisa de uma necessidade do usuário do aplicativo sob o ponto de vista desse usuário. Consiste em utilizar ou não User Stories para estimar o projeto.	P2, P6	2
Confidencialidade do Projeto	Se o projeto exige uma confidencialidade das informações do projeto ou não.	P4	1
Mudanças	Se existe ou não um controle das mudanças que são realizadas no projeto.	P4	1
Riscos	Se o projeto possui clareza dos riscos existentes ou não.	P2	1
Migração de Projeto	Se o projeto está preparado para uma possível migração ou não.	P1	1
Tipo do Escopo	Se o escopo é aberto (não definidos os requisitos) ou fechado (todos os requisitos são definidos).	P6	1
Nível de Definição dos Requisitos	O grau de definição dos requisitos pode varia de baixo (requisitos não estão definidos), médio (uma parte dos requisitos estão definidos) ou alto (todos os requisitos estão definidos).	P5	1
Estabilidade dos Requisitos	Consiste em determinar se os requisitos estão estáveis ou não.	P1, P7	2
Variação de formas de realizar a mesma funcionalidade	Se existe uma variação de formas de realizar a mesma funcionalidade ou não. Exemplo: a forma de pagamento que pode ser por boleto, cartão de crédito ou paypal.	P4	1
Premissa do Projeto	Premissas são fatores associados ao escopo do projeto que, para fins de planejamento, são assumidos como verdadeiros, reais ou certos, sem a necessidade de prova ou demonstração. Se é/será realizado o levantamento das premissas do projeto ou não.	P8	1
Definição dos Objetivos das partes interessadas	Se os objetivos das partes interessadas estão definidos ou não.	EP5	1
Flexibilidade da Metodologia de Desenvolvimento	Se existe Flexibilidade da Metodologia de Desenvolvimento ou não.	P4	1
Dependência entre Tarefas	Se existe alguma tarefa dependente de outra tarefa ou não.	P5	1
Identificação dos MVPs	Versão de um produto que pode ser lançada com uma quantidade mínima de esforço e desenvolvimento. Se é/será identificado o Produto Mínimo Viável do projeto ou não.	P7, P8	2
Identificação de Personas	Personas são personagens fictícios que são usados para representar usuários em termos de seus objetivos e características pessoais. Se é identificada as Personas ou não.	P7	1
Identificação da Visão do Produto	Visão do produto é a razão de existir do produto, deve direcionar todas as decisões em relação a esse produto. Se é identificada a visão do produto ou não.	P7	1
Técnicas de Concepção do Produto	Se são utilizadas técnicas de concepção do produto ou não.	P7	1
Processo de validação e verificação e implementação	Se existem um processo de verificação e validação e implementação ou não.	EP5	1
Teste em ambiente real	Se o teste é realizado nos dispositivos dos usuários finais ou não.	EP5	1

Mensurar o processo	Se é mensurada cada etapa do processo do projeto ou não.	EP5	1
Entendimento dos custos de modificações	Se é calculado os custos das modificações antes de fazê-las ou não.	EP5	1

- **Tecnologia**

Esta subcategoria abrange os fatores de custos associados às tecnologias (Linguagem de programação, ferramentas, plataformas) exigidos em um projeto de aplicativo móvel. A Tabela 5.8 contém esses 3 fatores no total.

Tabela 5.8: Fatores relacionados à Tecnologia

Preditor	Descrição	Legenda	F
Nível de Falhas de software e hardware	A frequência de falhas de software e hardware que pode variar de nunca, raramente e sempre.	EP5, EP6	2
Utilização de APIs	API é um conjunto de rotinas e padrões estabelecidos por um software para a utilização das suas funcionalidades por aplicativos que não pretendem envolver-se em detalhes da implementação do software, mas apenas usar seus serviços. Se é utilizado alguma API, (Cliente ou API nativa) ou não.	EP5, P4	2
Tipos de sistemas de back-end	Sistemas de back-end são responsáveis pela implementação da regra de negócio, por exemplo CMS, web site, ERP, CRM. O aplicativo tem um sistema de back-end a complexidade é baixa, o aplicativo tem dois sistemas de back-end a complexidade é média e três ou mais a complexidade é alta.	EP3, EP4	2

5.6 AVALIAÇÃO DA TAXONOMIA

Segundo Usman *et al.* (2017), a taxonomia pode ser validada de três formas: por demonstração de ortogonalidade, benchmarking e demonstração de utilidade. A taxonomia proposta nesta dissertação foi validada de duas formas por demonstração de ortogonalidade e demonstração de utilidade. Benchmarking não foi utilizado por não existir uma taxonomia de preditores de esforço de aplicativos móveis na literatura. A seguir é descrito como foi realizada a avaliação da taxonomia.

5.6.1 Demonstração de ortogonalidade

Foi utilizada uma abordagem *bottom-up* para identificar as categorias da taxonomia apresentada:

- I. Todos os preditores de esforço de aplicativos móveis foram identificados por estudos primários selecionados (Capítulo 4) e estudo qualitativo com profissionais que estimam aplicativos móveis (Capítulo 3)
- II. Os preditores de esforço foram analisados para identificar diferenças e semelhanças entre eles (atividade B8). Esse processo levou a identificação de categorias ortogonais, conforme mostrado na Figura 5.2.

5.6.2 Demonstração de utilidade:

Foi realizado um estudo cujo objetivo foi definido de acordo com o paradigma Goal-Question-Metric (GQM) (Basili e Rombach, 1988), conforme a Tabela 5.9.

Tabela 5.9: Objetivo do estudo preliminar de acordo com o paradigma GQM.

Analisar	Taxonomia de preditores de esforço de aplicativos móveis.
Com o propósito de	Avaliar.
Em relação a	Frequência de uso. Pretensão de uso.
Do ponto de vista de	Gerentes de projeto, desenvolvedores e analistas.
No contexto de	Na indústria que desenvolve aplicativos móveis.

Com base no objetivo do estudo, surgiu a seguinte questão de pesquisa “*Qual a frequência do uso e pretensão de uso de preditores de esforço para estimar aplicativos móveis?*” As métricas associadas à questão são *frequência dos participantes pretenderem e usarem cada preditor de esforço da taxonomia*. Esta abordagem de definição orientou a definição dos artefatos de coleta de dados necessários no estudo.

Em março de 2018, participaram do estudo 7 profissionais da indústria que trabalham ou trabalharam com estimativa de aplicativos móveis em 5 diferentes empresas de Manaus (AM). Primeiramente foi feita a caracterização de cada participante por meio de um formulário de caracterização que tinha como objetivo identificar a experiência dos participantes com relação à: (a) tempo de experiência com desenvolvimento de aplicativos móveis, (b) tempo de experiência com estimativa de aplicativos móveis, (c) número de aplicativos estimados e (d) tipos de aplicativos móveis que já estimaram. Os dados de caracterização foram analisados e a Tabela 5.10 mostra as características gerais dos participantes.

Quanto ao percentual de tipos de aplicativos que os participantes estimaram, 100% estimaram aplicativos do tipo compras e 71,4% corporativo e 57,1% estimaram jogos, entretenimento e educação, e apenas 28,6% estimaram outros como internet das coisas, segurança, plataforma de pagamento, manufatura, disponibilizar serviços (cursos e emprego). Os artefatos utilizados no estudo são descritos abaixo:

*Termo de Consentimento Livre e Esclarecido (TCLE)*³: os participantes assinaram eletronicamente um Termo de Consentimento Livre e Esclarecido (TCLE), onde concordavam em disponibilizar seus resultados para análise.

*Questionário de avaliação*⁴: Para coletar os dados sobre uso e pretensão de uso de cada preditor no momento da estimativa, foi elaborado um questionário detalhando cada preditor da taxonomia. Cada preditor é validado por meio de uma escala que mede o grau de uso e pretensão de uso: (a) uso e pretendo continuar usando, (b) uso, mas não sei se pretendo continuar usando, (c) não uso, mas pretendo usar e (d) não uso e não pretendo usar.

Para realizar o estudo, foi enviado um e-mail para os participantes, convidando-os a participarem do estudo, 100% aceitaram e em seguida foram enviados o formulário de caracterização e o formulário³ de avaliação juntamente com o TCLE. Após a execução do estudo, foi realizada a análise dos dados coletados. Todas as métricas de tamanho e fatores de custo foram analisadas de acordo com o uso e a pretensão de uso através de quatro afirmativas: (a) uso e pretendo continuar usando, (b) uso, mas não sei se pretendo continuar usando, (c) não uso, mas pretendo usar e (d) não uso e não pretendo usar.

A Figura 5.3 mostra o percentual de tipos de aplicativos que os participantes estimaram, 100% estimaram aplicativos do tipo compras e 71,4% corporativo e 57,1% estimaram jogos, entretenimento e educação e apenas 28,6% estimaram outros como internet das coisas, segurança, plataforma de pagamento, manufatura, disponibilizar serviços (cursos e emprego).

Tabela 5.10: Características dos Participantes do Estudo de Avaliação

Participantes	Cargo:	Experiência com desenvolvimento de aplicativos móveis (ano):	Experiência com estimativa aplicativos móveis (ano):	Nº de aplicativos estimados:	Tipo de aplicativos móveis que já estimou:
S1	Product Owner	6	6	32	Jogos, Entretenimento, Compras, Corporativo, Internet das Coisas, Segurança, Plataforma de Pagamento, Manufatura.
S2	Empresário	10	10	20	Jogos, Entretenimento, Compras, Corporativo
S3	Desenvolvedor e Líder Técnico	10	9	100	Jogos, Entretenimento, Compras, Corporativo, Educação

³ <https://goo.gl/forms/rev5ou1kCiHbcPn13>

⁴ <https://goo.gl/forms/HHPcMmNEM4cCRMqe2>

S4	Desenvolvedor de software	7	4	20	Jogos, Compras, Corporativo, Educação
S5	Analista de Sistemas	6	6	5	Compras, Educação, para disponibilizar serviços, como cursos, emprego
S6	Analista de Negócio	2	2	3	Compras, Corporativo
S7	Desenvolvedor	4	4	7	Entretenimento, Compras, Educação

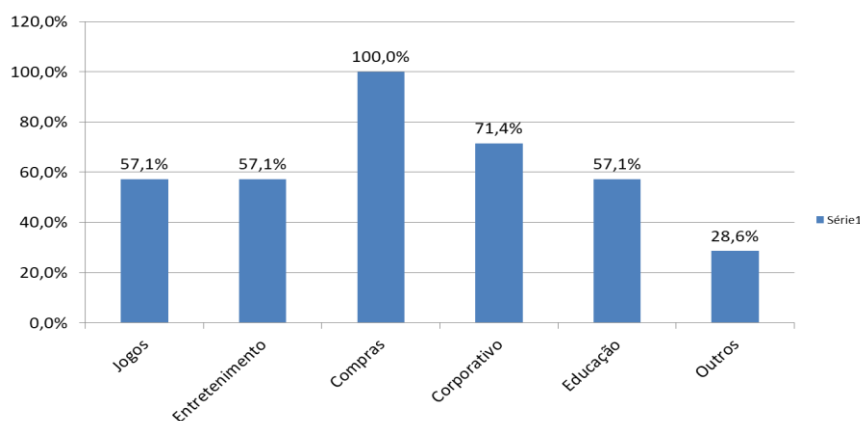


Figura 5.3: Percentual de tipos de aplicativos estimados

- **Métricas de Tamanho**

A Figura 5.4 apresenta a frequência do uso e pretensão do uso para cada métrica de *length*. Observou-se que 100% dos participantes apontaram *user stories* como sendo uma métrica que usam e que pretendem continuar usando durante a estimativa de aplicativos móveis, enquanto que 29%, 57%, 86% e 71% dos participantes não usam e não pretendem usar quantidade de elementos da interface gráfica, número de arquivos XMI e todas as métricas relacionadas ao código fonte e alguns de documentação.

No entanto, 14% dos participantes não usam, mas pretendem usar quantidade de elementos da interface gráfica, número de linhas contendo comentários, número de linhas contendo código fonte, número total de linhas e número de diagramas de sequência. Os resultados indicam que 58% das métricas de *length* não são usadas e nem pretendem ser usadas para estimar projetos de aplicativos móveis. Além disso, importante ressaltar que 86% das métricas de tamanho vieram do mapeamento sistemático publicados entre 2005 a 2015, implicando em uma defasagem das métricas de tamanho nos dias atuais e deixando indícios de dicotomia entre os dados publicados na literatura e os usados por especialistas em estimativa na indústria de software.

A

Figura 5.5 apresenta a frequência do uso e pretensão do uso para cada métrica de complexidade. Observou-se que 57% e 29% participantes usam e pretendem continuar usando nível de complexidade do contexto do aplicativo e complexidade ciclomática durante a estimativa de aplicativos móveis, enquanto que 29% e 71% dos participantes não usam e não pretendem usar métricas de complexidade.

Os resultados indicam que o nível de complexidade do contexto do aplicativo (métrica coletada pelo estudo qualitativo) é usado por 57% e dos participantes que ainda não usaram 14% pretendem usar. Porém, 71% dos participantes afirmam que não usam e não pretendem usar complexidade ciclomática (métrica coletada pelo MSL), reforçando indícios de que as métricas de complexidade vindas da literatura já estão se tornando arcaicas no contexto de estimativa de aplicativos móveis.

A Figura 5.6 apresenta a frequência do uso e pretensão do uso para cada métrica de funcionalidade. Observou-se que 100% participantes não usam e que pretendem usar, COSMIC, FISMA e variação do método COSMIC como métrica de funcionalidade durante a estimativa de aplicativos móveis (esses 3 métodos de desenvolvimento de projeto são tradicionais para projetos grandes que possui uma complexidade bem maior, analisando o contexto de aplicativos móveis o escopo é bem menor e os requisitos são bem mais voláteis então é adotado metodologias mais flexíveis como o Scrum para projetos de dispositivos móveis), enquanto que 71% e 14% dos participantes usam e pretendem continuar usando o tempo total para fazer as telas do aplicativo e “*t-shirt sizes*” (métrica apontada pelos participantes da avaliação e adicionada na taxonomia) e análise por pontos de função respectivamente. Os resultados indicam que 67% das métricas de funcionalidade não são usadas e nem pretendem ser usadas para estimar projetos de aplicativos móveis, 67% das métricas foram coletas por meio do MSL, reforçando que há dicotomia entre os dados publicados na literatura e os usados por especialistas em estimativa na indústria de software.

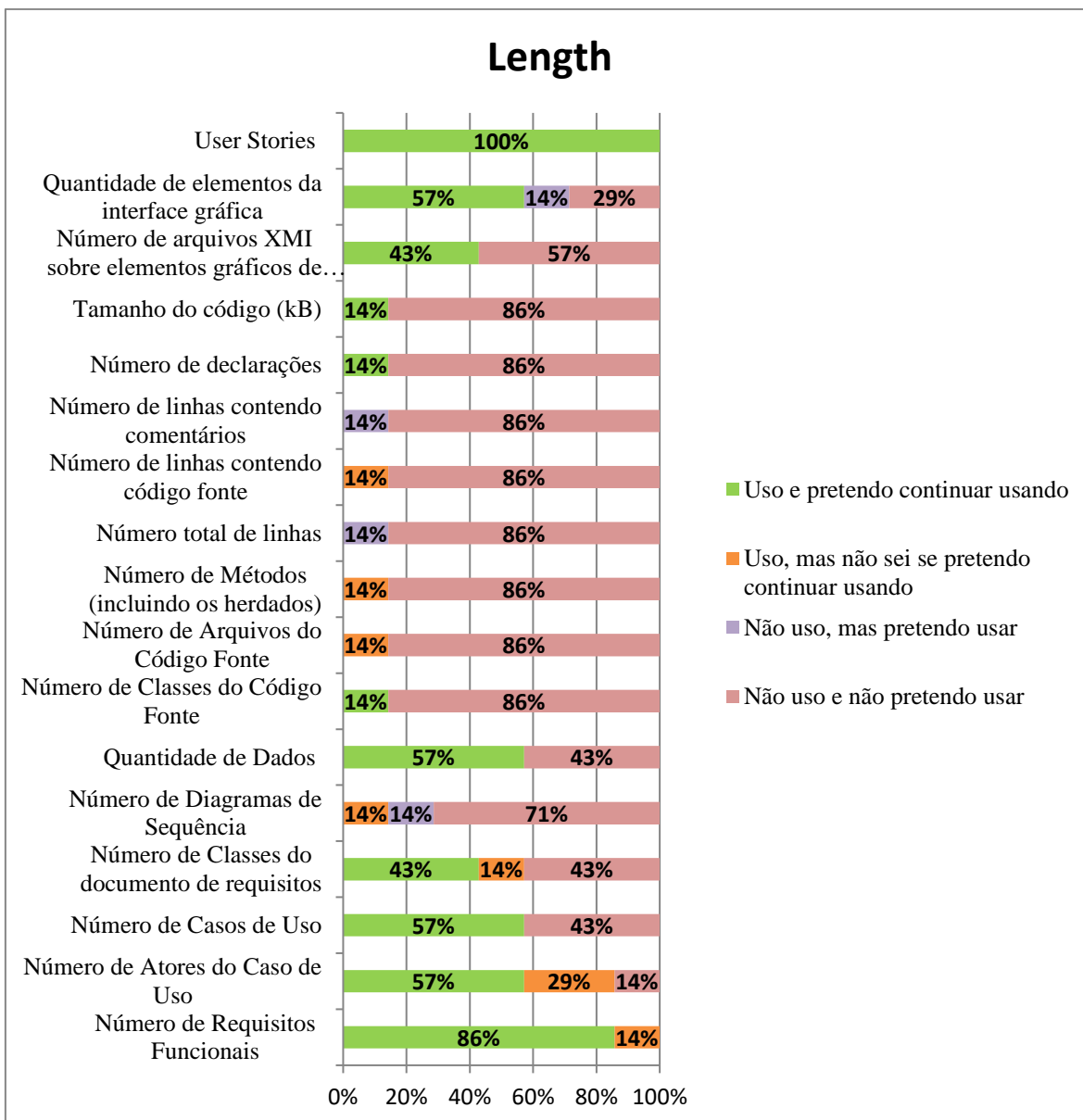


Figura 5.4: Avaliação das Métricas de Length

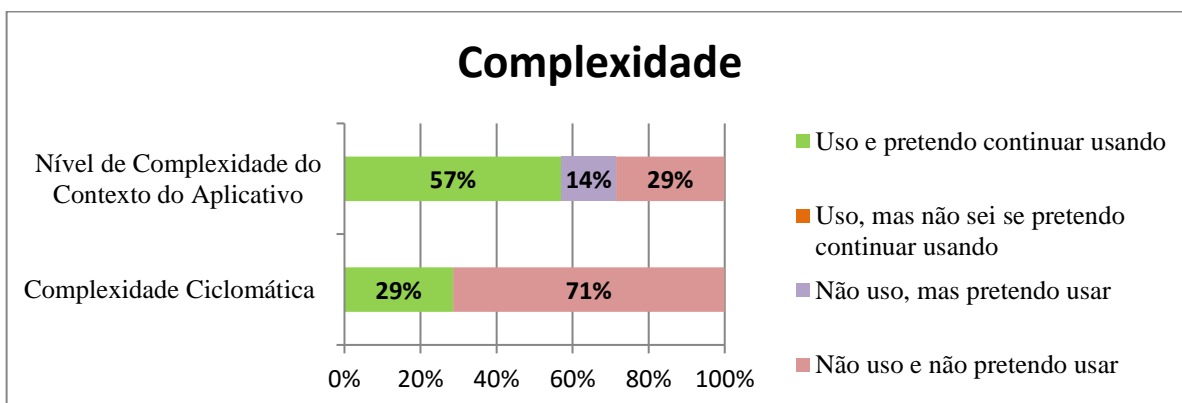


Figura 5.5: Avaliação das Métricas de Complexidade

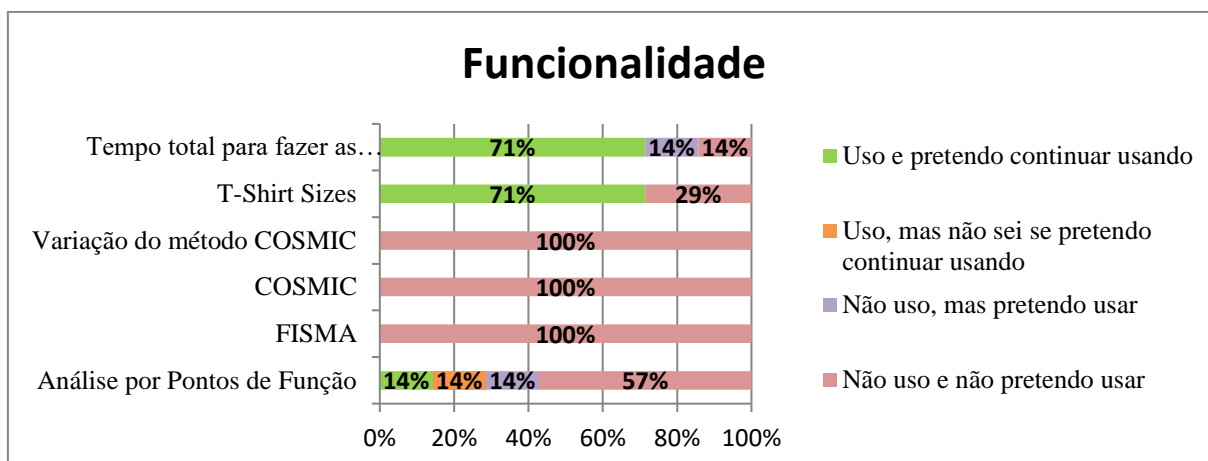


Figura 5.6: Avaliação das Métricas de Funcionalidade

- **Fatores de Custo**

A Figura 5.7 apresenta a frequência do uso e pretensão do uso para cada fator de custo relacionado ao cliente. Observou-se que 60% dos preditores relacionados ao cliente são usados e pretende-se continuar usando por mais de 50% dos participantes, os preditores são nível de participação do cliente (S1, S4, S5, S6), nível de comunicação com o cliente (S1, S2, S4, S5, S6), expectativas dos usuários e clientes (S1, S2, S3, S4, S5) durante a estimativa de aplicativos móveis, enquanto que 40% dos preditores relacionados ao cliente não são usados e não pretendem ser usados por menos que 50% dos participantes, os preditores são autonomia do cliente na tomada de decisão (S2, S3, S6) (esse preditor teve mais de 57% de rejeição dos participantes, isso se deve ao fato de que nem todo o cliente é usuário final do aplicativo móvel e sim os funcionários ou clientes do mesmo, então o fornecimento dos requisitos e modelagem do problema é fornecida pelos usuários finais do aplicativo, por isso que esse fator não é muito utilizado para estimar) e nível de competências dos usuários e clientes (S1, S2, S4), porém 14% dos participantes (S5) não usam, mas pretendem usar esses 2 preditores citados anteriormente. Os resultados indicam que os preditores relacionados ao cliente são usados e que pretendem continuar usando sim no momento da estimativa de aplicativos móveis.

A Figura 5.8 apresenta a frequência do uso e pretensão do uso para cada fator de custo relacionado à empresa. Observou-se que 14% dos preditores relacionados à empresa são usados e pretende-se continuar usando por mais de 50% dos participantes, o preditor disponibilidade dos dispositivos para teste (S1, S3, S4, S5, S6), 57% dos preditores não são

usados e nem pretende-se usar por mais de 50% dos participantes, os preditores são disponibilidade de infraestrutura para falar com o cliente (S2, S5, S6, S7), nível de disponibilidade da internet da empresa (S2, S3, S6, S7), acordos relacionados a prática de contratação de funcionário entre os concorrentes (S1, S2, S3, S7), nível de certificação da empresa (S1, S2, S5, S6, S7) durante a estimativa de aplicativos móveis.

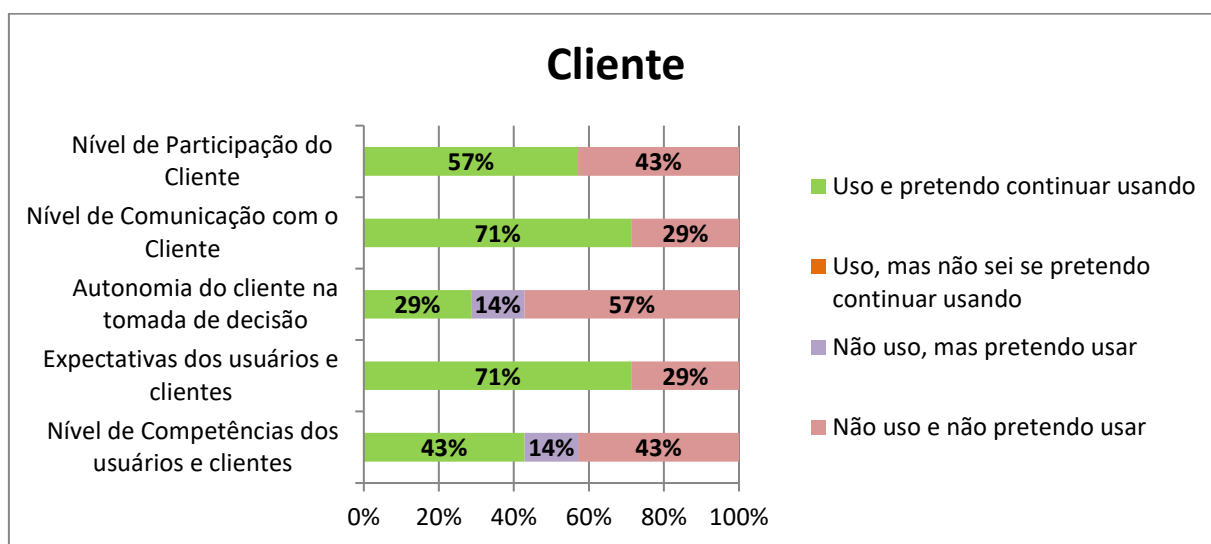


Figura 5.7: Avaliação de Preditores relacionados ao Cliente

No entanto, 57% dos preditores não são usados, mas 14% dos participantes pretendem usar, os preditores são acordos relacionados à prática de contratação de funcionário entre os concorrentes (S5), competência de fornecedores (S5), nível de certificação da empresa e nível de infraestrutura ferramental da empresa (S5), e 14% dos participantes afirmaram que usam, mas não sabem se pretendem continuar usando nível de infraestrutura ferramental da empresa (S4).

Os resultados indicam que grande parte dos participantes não utilizam preditores relacionados à empresa para estimar projetos de aplicativos móveis, isso pode ocorrer devido à flexibilidade de ambiente de trabalho onde se produz o software que pode ser em qualquer lugar, desde que tenha um notebook, dispositivo móvel e internet.

A Figura 5.9 apresenta a frequência do uso e pretensão do uso para cada fator de custo relacionado à equipe de desenvolvimento. Observou-se que 100% dos participantes usam e pretendem continuar usando quantidade de pessoas trabalhando no projeto, experiência da equipe no contexto da aplicação e nível de experiência com a tecnologia, enquanto que 90% dos preditores são usados e pretende-se continuar usando-os por mais de 57% dos participantes. Além disso, 50% dos preditores não são usados e não se pretende usar por

menos que 43% dos participantes, os preditores são nível de engajamento da equipe (S2 e S6), curva de aprendizagem da equipe (S2, S3, S6), definição dos papéis das equipes (S6), quantidade de horas dedicadas de cada pessoa do projeto (S7) e nível de comunicação entre a equipe (S2, S6 e S7).

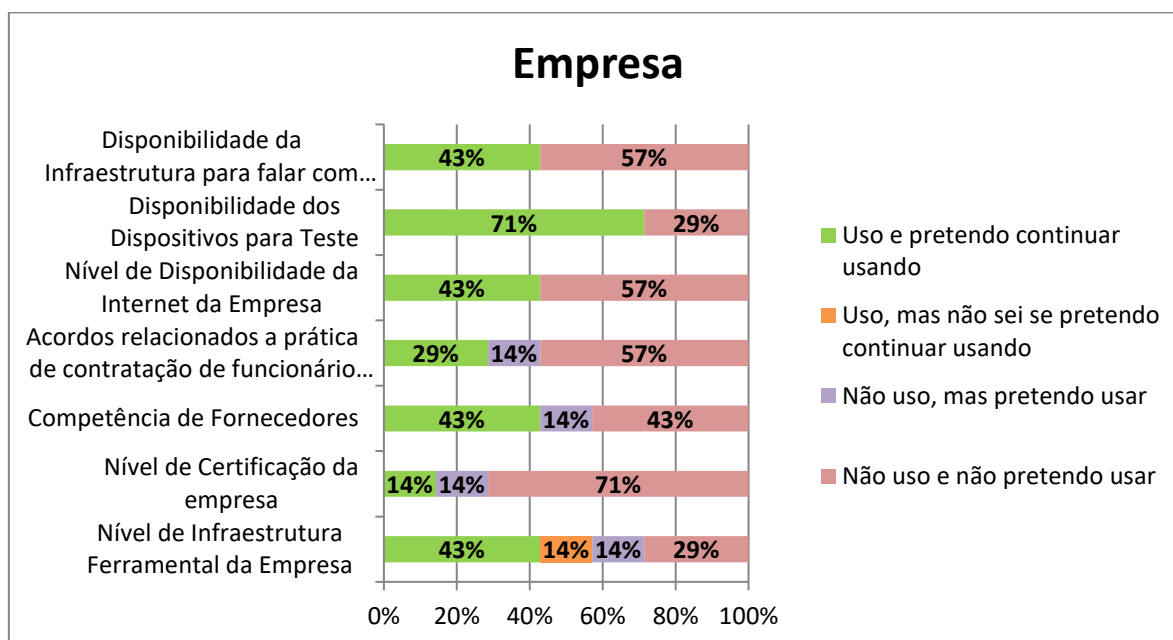


Figura 5.8: Avaliação de Preditores relacionados à Empresa

No entanto, 14% dos participantes não usam, mas pretendem usar nível de conhecimento da competência da equipe (S7), nível de experiência no desenvolvimento de projetos anteriores (S7) e nível de comunicação da equipe (S3), e 14% dos participantes afirmaram que usam, mas não sabem se pretendem continuar usando definição de papéis das equipes (S7). Os resultados indicam que 86% parte dos participantes utilizam preditores relacionado à equipe para estimar projetos de aplicativos móveis, não só aplicativos móveis, mas também outros tipos de projetos.

A Figura 5.10 apresenta a frequência do uso e pretensão do uso para cada fator de custo relacionado ao projeto. Observou-se que 100% dos participantes usam e pretendem continuar usando teste em ambiente real e nível de documentação do projeto, enquanto que 77% dos preditores são usados e pretende-se continuar usando-os por mais de 57% dos participantes. Enquanto, 4% dos preditores não são usados e não se pretende usar por mais que 50% dos participantes, o preditor é flexibilidade da metodologia de desenvolvimento (S2 e S3, S5, S6).

Mais de 40% dos preditores relacionados ao projeto não são usados, mas 14 à 43% dos participantes pretendem usá-los, os preditores são mensurar o processo (S3, S4, S6), processo de avaliação e verificação e implementação (S5), técnicas de concepção do produto (S5), identificação de personas (S3, S5), identificação dos MVPs (S5), definição dos objetivos pelas partes interessadas (S5), premissa do projeto (S5), documento de user experience (UX) (S5) e preparação de gestão de mudança (S7).

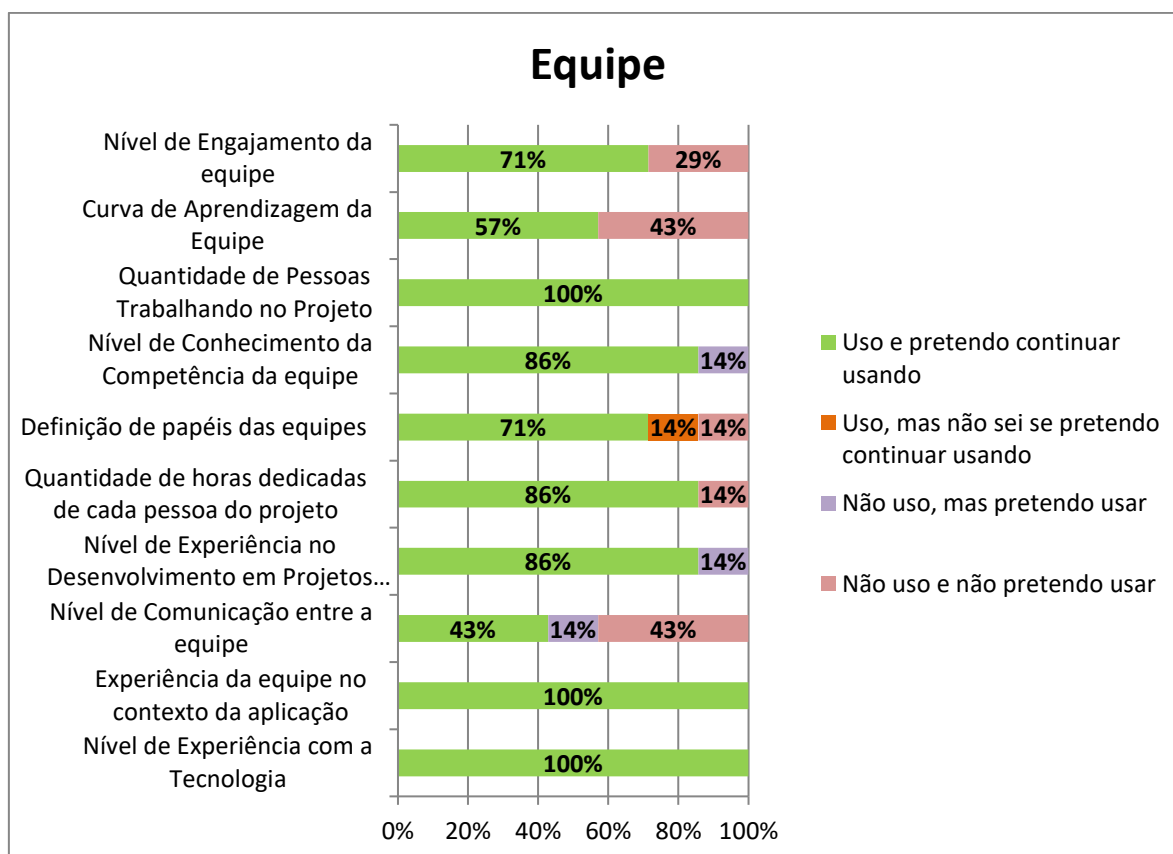


Figura 5.9: Avaliação de Preditores relacionados à Equipe de Desenvolvimento

Além disso, 40% dos preditores relacionados ao projeto são usados, mas não se sabe se pretende continuar usando por 14% a 29% dos participantes, os preditores são mensurar o processo (S5), processo de validação e verificação e implementação (S6), identificação dos MVPs (S3), flexibilidade da metodologia de desenvolvimento (S4), tipo de escopo (S4), migração do projeto (S4, S7), riscos (S5), documento de *user experience* (S4) e preparação da gestão de mudanças (S4). Os resultados indicam que grande parte dos participantes utilizam preditores relacionado ao projeto para estimar projetos de aplicativos móveis, vale ressaltar que esses preditores relacionados ao projeto pode ser aplicado a projetos em geral, pois é na interpretação de como considerar a métrica que vem a diferenciação para aplicativos móveis,

apenas a categoria de produto é específica para aplicativos móveis, as demais podem ser aplicadas para projetos em geral.

A Figura 5.11 apresenta a frequência do uso e pretensão do uso para cada fator de custo relacionado ao produto. Observou-se que 100% dos participantes usam e pretendem continuar usando 4 preditores que são qualidade dos produtos, integração com as redes sociais, requisição para a internet e tipo de conectividade do dispositivo móvel, enquanto que mais da metade do número de participantes que representam 57%, e 71% não usam e não pretendem usar sistema de rastreamento (S2, S3, S4 e S5), compatibilidade entre emuladores e dispositivos móveis (S2, S3, S4 e S6) e interrupções de chamadas (S2, S3, S4, S5 e S6).

No entanto, 14% e 29% dos participantes não usam, mas pretendem usar objetivo do aplicativo, tempo de vida da instância da aplicação, publicidade, portabilidade entre sistemas operacionais, comutação de acesso a rede, variação de entrada da interface gráfica e largura de banda da rede, e 14% e 29% dos participantes afirmaram que usam, mas não sabem se pretendem continuar usando centralidade, consciência operacional, realidade aumentada, consciência de proximidade, consciência de espaço consciência de localização absoluta, tempo de vida da instância da aplicação e outros mostrados na Figura 5.11. Os resultados indicam que grande parte dos participantes utilizam fatores relacionado ao produto para estimar projetos de aplicativos móveis e que apenas 4 preditores são usados por menos que 50% dos participantes e esses preditores como sistema de rastreamento, compatibilidade entre emuladores e interrupção de chamada, centralidade não são mais comuns a preocupação com esses preditores, pois a tecnologia e o hardware avançaram nos últimos anos, tornando esses preditores menos usados com o tempo.

A Figura 5.12 apresenta a frequência do uso e pretensão do uso para cada fator de custo relacionado à tecnologia. Observou-se que 100% dos participantes usam e pretendem continuar usando tipos de sistemas de back-end e utilização de APIs, e 86% também usam e pretendem continuar usando nível de falhas de software e hardware, enquanto que 14% dos participantes não usam e não pretendem usar nível de falhas de software e hardware (S3).

Os resultados indicam que grande parte dos participantes utilizam e pretendem continuar usando preditores relacionados à tecnologia no momento de estimar projetos de aplicativos móveis e que apenas o participante S3 não usa e não pretende usar nível de falhas, o S3 que estimou 100 aplicativos nos contextos de jogos, entretenimento, compras,

corporativo, educação, porém os participantes S1, S2, S4, S5, S6 e S7 usam e pretendem continuar usando todos os preditores relacionados a tecnologia.

E para finalizar, o participante S1 relata durante o estudo de avaliação que a taxonomia é uma boa ferramenta de gestão de conhecimento para apoiar lições aprendidas para melhorar as estimativas futuras, como por exemplo: “Isso aqui demorou mais tempo que a gente estimou ou foi muito mais rápido do que a gente estimou. Porque? O que a gente não pensou na hora de estimar? Ai a gente pode pegar essa taxonomia e ver o fator que a gente deixou passar, então, vamos botar isso aqui como entrada nas próximas vezes.” Portanto, existem evidências de que a taxonomia pode apoiar a estimativa de projetos de aplicativos móveis.

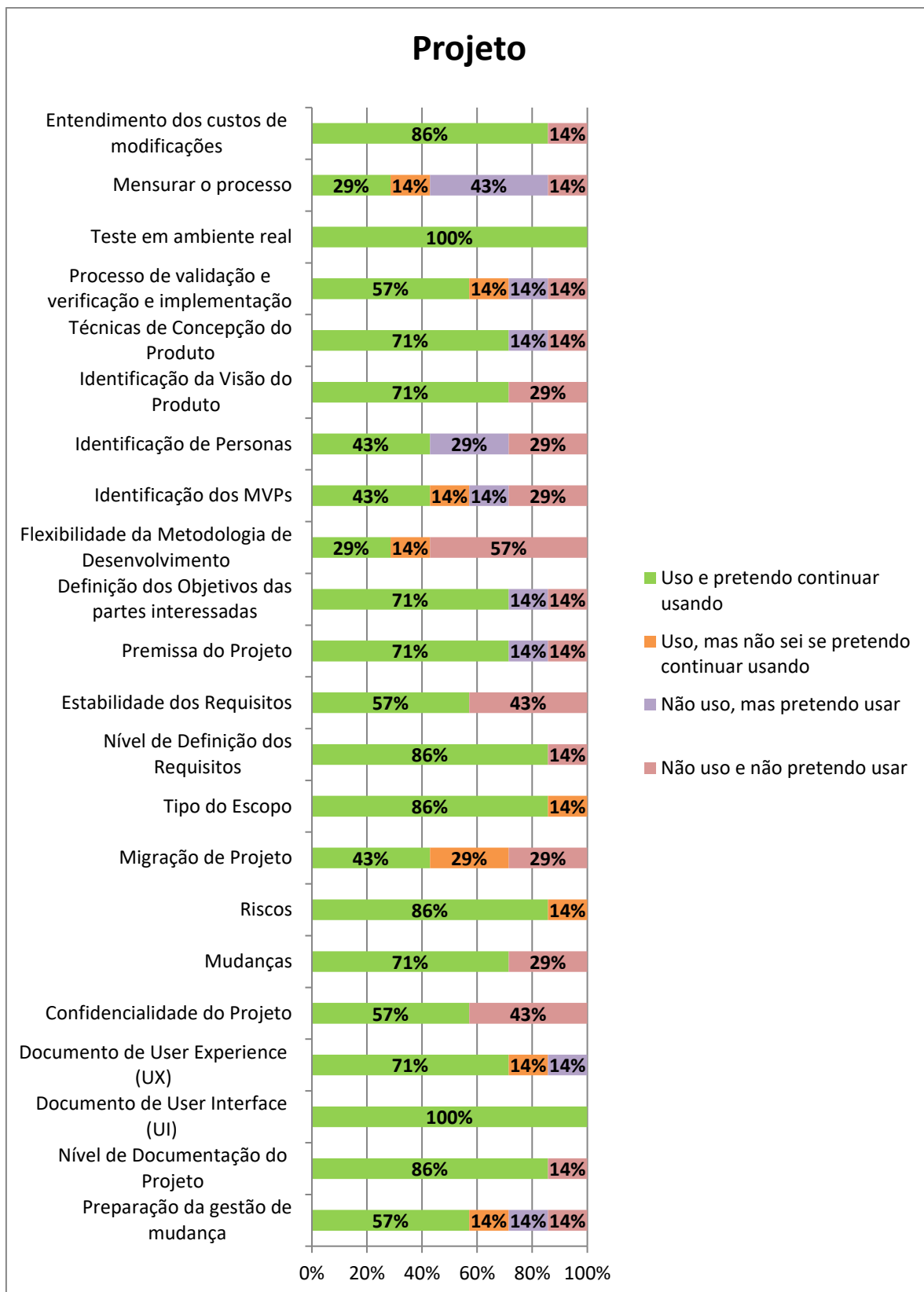


Figura 5.10: Avaliação de Preditores relacionados ao Projeto

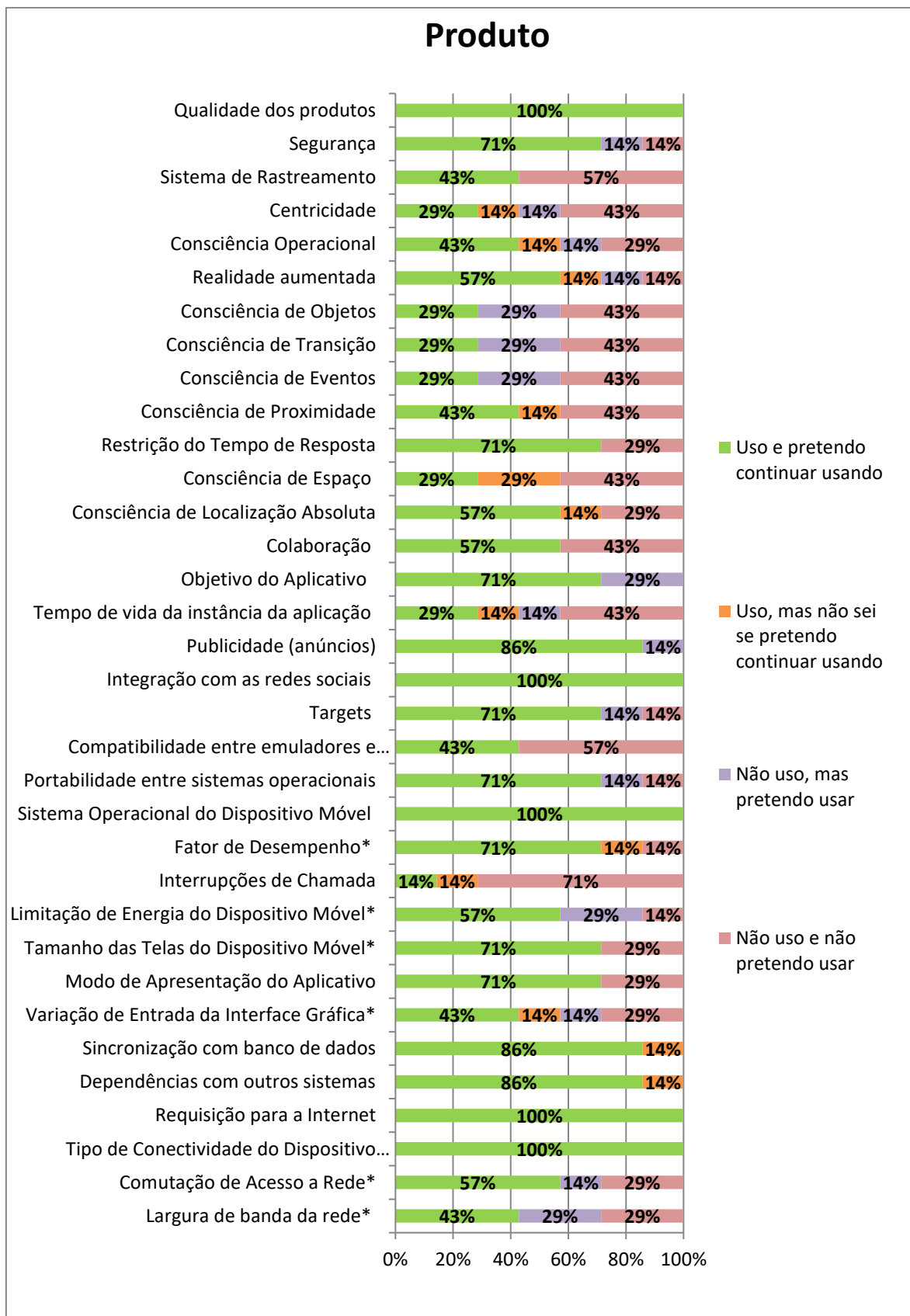


Figura 5.11: Avaliação de Preditores relacionados ao Produto

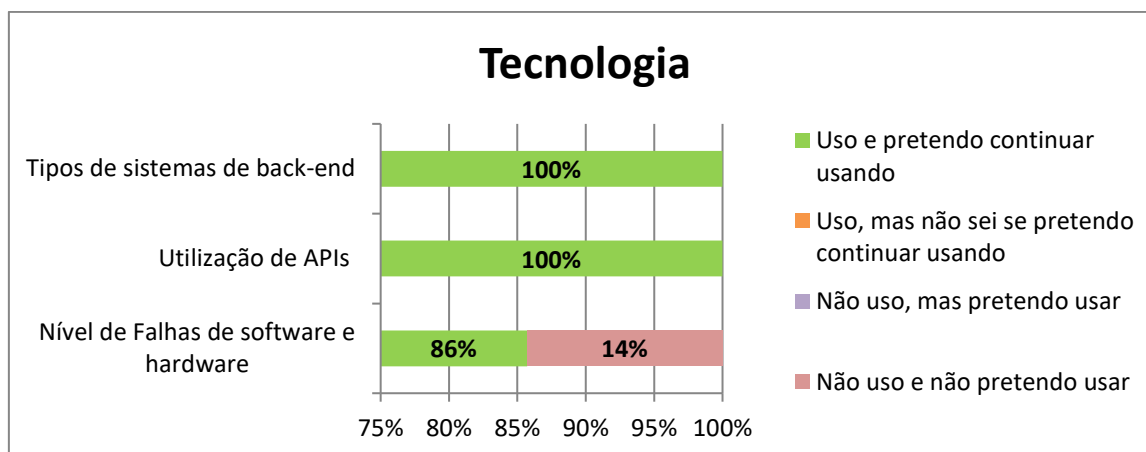


Figura 5.12: Avaliação de Preditores relacionados à Tecnologia

5.7 DISCUSSÃO

A taxonomia de preditores de esforço de aplicativos móveis e a organização do corpo de conhecimento têm contribuições tanto para a academia quanto para indústria, que serão discutidas a seguir.

5.7.1 Contribuições para a academia

Os pesquisadores podem usar a taxonomia e o corpo organizado de conhecimento (o resultado da aplicação da taxonomia para classificar os 108 preditores de esforço identificados na literatura e na indústria) para selecionar os preditores a serem considerados ao investigar novas abordagens de estimativa de esforço, pois todos os estudos aqui apresentados possuem evidências empíricas e os preditores mais frequentes podem levar à investigação de modelos de estimativa mais acurados.

Os resultados deste trabalho também indicam possíveis lacunas que podem ser abordadas através de novas pesquisas, tais como diretrizes para os profissionais que trabalham com projetos de aplicativos móveis selecionarem o modelo e preditores de esforço mais adequados para os projetos deles.

Outro ponto a ser destacado é a evolução ao longo do tempo da taxonomia de preditores que deve acontecer, pois, acredita-se que enquanto novos trabalhos relacionados forem produzidos, talvez haja necessidade de incorporar novas categorias na taxonomia apresentada neste artigo, para que o corpo de conhecimento também seja atualizado, atendendo as novas categorias e estudos.

5.7.2 Contribuições para a indústria

A taxonomia e o corpo organizado de conhecimento podem ajudar os profissionais que estimam aplicativos móveis de três maneiras diferentes. Primeiro, como pesquisadores, profissionais podem identificar literatura de interesse consultando os estudos identificados no mapeamento sistemático da literatura (veja o Capítulo 4). Segundo, a taxonomia pode ser usada como “*container*” que armazena organizadamente uma base de conhecimento dos preditores usados para estimar projetos, ficando mais fácil um futuro compartilhamento de conhecimento. Terceiro, pode ajudar os profissionais a selecionarem preditores para estimar o esforço de projetos de aplicativos móveis. Para exemplificar como os profissionais podem usar os resultados, a seguir é apresentado um exemplo:

Uma equipe precisa estimar o esforço necessário para desenvolver um aplicativo que auxilia as aulas de programação para uma instituição de ensino superior. A equipe é experiente, pois vem desenvolvendo aplicativos móveis há muitos anos e usando diferentes técnicas de estimativa baseadas por especialistas para estimar o esforço de desenvolvimento. Apesar do fato da equipe ser muito experiente em desenvolvimento e estimativa de aplicativos móveis, esta é a primeira vez que a equipe desenvolve um aplicação no domínio da educação.

No exemplo acima, a equipe não pode confiar apenas em sua própria experiência, já que nunca desenvolveram um aplicativo móvel no domínio da educação. Portanto, a equipe poderia usar o corpo organizado de conhecimento para selecionar as métricas de tamanho e os fatores de custos a serem usados no processo de estimativa. Para fazer isso, a equipe pode selecionar os preditores de duas maneiras diferentes:

- Selecionar os preditores usados com mais frequência, considerando um limite predefinido. A equipe, no exemplo mencionado, poderia considerar todos os preditores identificados em 18 estudos e apontados por 8 profissionais que estimam na indústria, o que resultaria, por exemplo, nas seguintes métricas (conforme a Seção 5.5.1): Número de Requisitos Funcionais, Número de Atores do Caso de Uso, Número de Casos de Uso, Número de Classes do documento de requisitos, Número de linhas contendo código fonte, Tamanho do código (kB), Número de arquivos XMI sobre elementos gráficos de um aplicativo para dispositivos móveis, Quantidade de elementos da interface gráfica.
- Selecionar os preditores relevantes com base em informações contextuais. A equipe, no exemplo mencionado, falou com o cliente e os representantes da empresa, porém perceberam que há um baixo nível de Comunicação com o Cliente e o pessoal da

empresa não parece estar disposto a colaborar com eles. Assim, a equipe suspeita que esses dois aspectos provavelmente impactarão o esforço associado no desenvolvimento do aplicativo móvel. Assim, os preditores de nível de Comunicação com o Cliente e Experiência da equipe no contexto da aplicação são selecionados, pois a equipe está trabalhando pela primeira vez no contexto de educação e há pouca interação com o cliente que usará o aplicativo.

A taxonomia e o corpo de conhecimento aqui apresentados abrangem aspectos gerais de desenvolvimento de aplicativos móveis e podem facilitar o compartilhamento de conhecimento e apoiar a tomada de decisão no momento da estimativa. Isto pode acontecer mesmo em aplicações que se baseiam em projetos já existentes.

5.8 CONSIDERAÇÕES SOBRE O CAPÍTULO

Este capítulo apresentou o planejamento, identificação e extração, design e construção, classificação de preditores de esforço e avaliação da taxonomia proposta que possui o objetivo de organizar o conhecimento de preditores de esforço para projetos de aplicativos móveis, os dados de entrada da taxonomia são os preditores de esforço descritos no Capítulo 3 e Capítulo 4. O capítulo a seguir descreve as conclusões, contribuições e perspectivas futuras.

CAPÍTULO 6 – CONSIDERAÇÕES FINAIS E PERSPECTIVAS FUTURAS

Este capítulo apresenta as conclusões desta dissertação, resumindo sua motivação, propostas e apresentando as suas contribuições. As perspectivas futuras fornecem a direção para que seja dada continuidade ao trabalho relacionado à taxonomia de preditores de esforço de projetos de aplicativos móveis.

6.1 CONCLUSÕES

Esta dissertação de mestrado apresentou uma pesquisa sobre como organizar o corpo de conhecimento de preditores de esforço para projetos de aplicativos móveis. Com este propósito, foi definida uma taxonomia de preditores de esforço para projetos de aplicativos móveis que auxilia profissionais no processo de tomada de decisão e compartilhamento de conhecimento.

Para a definição e avaliação da taxonomia, foi utilizada uma metodologia dividida em duas fases: a Fase 1 (Exploratória) buscou evidências na literatura e na indústria sobre preditores de esforço de aplicativos móveis. A Fase 2 (Projeto da Taxonomia) buscou projetar uma taxonomia de preditores de esforço identificados na literatura e na indústria, esta fase foi utilizado a metodologia proposta por Usman *et al.* (2017) para projetar a taxonomia.

A primeira fase, dois estudos foram realizados. O primeiro consistiu em um estudo qualitativo resultando em fatores que afetam a estimativa de esforço e variações do processo de estimativa de esforço nas empresas que desenvolvem aplicativos móveis. Esses fatores foram levantados por meio do conhecimento obtido a partir da estimativa de esforço de oito especialistas de cinco diferentes empresas que desenvolvem aplicativos móveis. Foram utilizadas entrevistas semiestruturadas e procedimentos baseados em *Grounded Theory* (Strauss e Corbin, 2015) ao longo da etapa de coleta e análise dos dados das entrevistas.

O segundo estudo consistiu em um mapeamento sistemático sobre os modelos de estimativa de tamanho, esforço e preditores de esforço de projetos de aplicativos móveis. Embora vários estudos tenham sido conduzidos com relação à estimativa de tamanho e esforço para aplicativos móveis, os resultados revelaram que ainda há necessidade de mais pesquisas nesta área. Entre as referências identificadas, por exemplo, não há uma organização contendo todos fatores citados na literatura que ajude os pesquisadores e profissionais a

selecionarem preditores de esforço para projetos de aplicativos móveis. Os preditores foram classificados em categorias, agrupados em Métricas de Tamanho e Fatores de Custo.

Em seguida, foi realizado um estudo para avaliar os preditores do corpo organizado de conhecimento resultante da taxonomia proposta. O estudo de avaliação por demonstração de ortogonalidade foi utilizada uma abordagem ascendente para identificar as categorias da taxonomia apresentada. Primeiro, foram identificados todos os preditores de esforço de projetos de aplicativos móveis relatados nos estudos. Segundo, foram analisados os preditores de esforço com o objetivo de identificar as semelhanças e diferenças entre os preditores, este processo levou a identificar categorias ortogonais.

O estudo de avaliação por demonstração de utilidade teve como objetivo medir o uso e a pretensão da continuidade do uso de preditores de esforço no processo de estimativa de projetos de aplicativos móveis. Participaram sete profissionais que estimam ou já estimaram projetos de aplicativos móveis em cinco diferentes empresas referencia da região norte do Brasil.

Os resultados mostraram que metade dos participantes do estudo não usam e não pretendem usar métricas de tamanho, pois cerca de 22% dessas métricas vieram do estudo qualitativo (Capítulo 3) e 78% vieram do MSL (Capítulo 4). Portanto, há indícios de dicotomia entre os dados publicados na literatura e os usados por especialistas em estimativa na indústria de software. Já os fatores de custo mais que a metade dos participantes usam e pretendem continuarem usando esses preditores para estimar projetos de aplicativos móveis, apenas a categoria de empresa que menos que a metade dos participantes não usam e não pretendem usar, isso pode ocorrer devido à flexibilidade de ambiente de trabalho onde se produz o software que pode ser em qualquer lugar, desde que tenha um notebook, dispositivo móvel e internet.

Quanto as ameaças à validade dos dados dos estudos, acredita-se que as principais limitações desta pesquisa estão relacionadas aos resultados do MSL e do conhecimento dos especialistas em estimativa. A principal ameaça é a generalização dos resultados para todos os profissionais que estimam projetos de aplicativos móveis. Mitigou-se esse problema ao entrevistar 8 profissionais e validada por mais 7 profissionais que trabalham com estimativa de projetos de aplicativos móveis em 5 diferentes empresas representativas da região norte do Brasil e também foram escolhidas três meta-bibliotecas digitais que indexam outras bibliotecas digitais de diferentes áreas de conhecimento, incluindo a área de estimativa de projetos de softwares para realizar o mapeamento sistemático.

Outra ameaça à validade dos resultados é a possibilidade de que a primeira autora possa ter introduzido o viés durante o protocolo de revisão e execução dos estudos. Para reduzir essa ameaça, a execução de todos os métodos de estudo foi realizado e revisado por outros pesquisadores experientes.

Portanto, existem evidências de que a taxonomia pode apoiar a estimativa de projetos de aplicativos móveis servindo como container para armazenar uma base de conhecimento de lições aprendidas no projeto, ajudando a melhorar as estimativas futuras. Importante ressaltar que as categorias desta taxonomia poderão ser usadas por outros tipos de projeto que não seja dispositivos móveis para organizar a base de conhecimento de preditores de esforço do projeto da empresa, além disso, o que mudará será a instância dos preditores de esforço da base de conhecimento.

6.2 CONTRIBUIÇÕES

Contribuiu-se para a área de Estimativa de Projetos de Aplicativos Móveis, Gerência de Projetos e Engenharia de Software, fornecendo uma taxonomia que visa a organização dos preditores de esforço consolidados na literatura e os apontados pelos profissionais na indústria. Além do mais, apoiar a estimativa de esforço de aplicativos móveis ajudando os profissionais a selecionarem preditores de esforço e auxiliando o compartilhamento de conhecimento.

As contribuições obtidas com esta pesquisa até o momento foram:

- Identificação de modelos de estimativa específicos para projetos de aplicativos móveis existentes na literatura e também a não existência de padrões no processo de estimativa nas 5 empresas entrevistadas no estudo qualitativo descrito no Capítulo 4.
- Disseminação dos resultados obtidos nesta pesquisa até o presente momento deste trabalho de mestrado. A seguir são descritos os detalhes:
 - O artigo “*Estimativa de Projetos de Aplicativos Móveis: Um Mapeamento Sistemático da Literatura*” (De Souza e Conte, 2017) foi publicado no 16º Simpósio Brasileiro de Qualidade de Software (SBQS 2017). Nesse artigo, foi apresentado um mapeamento sistemático da literatura com o objetivo de caracterizar fatores preditores de esforço e

modelos de estimativa voltados para projetos de aplicativos móveis conforme foi apresentado no Capítulo 3.

- O artigo “*Uma Taxonomia de Preditores de Esforço para Projetos de Aplicativos Móveis*” (De Souza et al., 2018) está sendo escrito para o journal *Institution of Engineering and Technology Software* (IET *Software*). Nesse artigo, está sendo apresentada a taxonomia de preditores para projetos de aplicativos móveis com o objetivo de organizar o corpo de conhecimento de preditores de esforço para projetos de aplicativos móveis conforme foi detalhada no Capítulo 5.
- Definição de uma taxonomia de preditores de esforço para projeto de aplicativos moveis.

6.3 PERSPECTIVAS FUTURAS

Considerando que qualquer conhecimento ou esquema de classificação é um esforço comunitário que evolui ao longo do tempo, espera-se que a taxonomia aqui apresentada evolua através de novas coletas de dados de conhecimento de especialistas em diferentes regiões do mundo.

A taxonomia pode ser ampliada de maneiras diferentes, por exemplo, pode ser estendida para explicar preditores que também afetam a qualidade, design e manutenção de projeto de aplicativos móveis, explicar outros elementos relacionados à estimativa de esforço, como técnicas de estimativa de esforço e de recursos.

REFERÊNCIAS

- Abdullah, N. A. S., Rusli, N. I. A., & Ibrahim, M. F. (2014). Mobile game size estimation: COSMIC FSM rules, uml mapping model and unity3d game engine. In *Open Systems (ICOS), 2014 IEEE Conference on* pp. 42-47.
- Albrecht, A. J. (1979). Measuring application development productivity. In *Proc. of the Joint SHARE/GUIDE/IBM Application Development Symposium*, pp. 83-92.
- Basili, V. R., & Rombach, H. D. (1988). The TAME project: Towards improvement-oriented software environments. *IEEE Transactions on software engineering*, vol. 14, n. 6, pp. 758-773.
- Bayona-Oré, S., Calvo-Manzano, J. A., Cuevas, G., & San-Feliu, T. (2014). Critical success factors taxonomy for software process deployment. *Software Quality Journal*, vol. 22, n.1, pp. 21-48.
- Boehm, B. W. (1981). *Software engineering economics. (Vol. 197)*. Englewood Cliffs (NJ): Prentice-hall.
- Borsoi, B. T., Colazzos, K. S., Ascari, R. E., Toscan, L. F., Bossola, L. H., Bolo, M. M., & Arsego, M. M. (2011). *Redes Neurais Aplicadas na Estimativa de Prazo de Projeto de Software. Encontro Paranaense de Computação (IV EPAC)*.
- Bourque, P., & Fairley, R. E. (2014). "Guide to the software engineering body of knowledge (SWEBOOK (R)): Version 3.0. IEEE Computer Society Press", <https://www.computer.org/web/swebok/v3>
- Britto, R., Usman, M., Mendes, E. (2017). A Taxonomy of Web Effort Predictors. In *Journal of Web Engineering (JWE)*, vol. 18, pp. 541-570.
- C.O.S.M.I.C. Consortium et al. (2007). *The cosmic functional size measurement method-version 3.0 measurement manual (the cosmic implementation guide for iso/iec 19761: 2003)*.
- COSMIC. (2015). "The COSMIC Functional Size Measurement Method Version 4.0.1 Measurement Manual", <http://cosmic-sizing.org/publications/measurement-manual-401/>, Abril.
- D'avanzo, L., Ferrucci, F., Gravino, C., & Salza, P. (2015). Cosmic functional measurement of mobile applications and code size estimation. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, pp. 1631-1636.
- De Souza, E. T. B., Conte. T. (2017). Estimativa de Projetos de Aplicativos Móveis: Um Mapeamento Sistemático. In *XVI Brazilian Symposium on Software Quality*, pp 206-220.
- De Souza, E. T. B., Valentim, N., Conte. T. (2018). Relatório Técnico sobre uma Taxonomia de Preditores de Esforço para Projetos de Aplicativos Móveis. Em *Relatório Técnico USES TR-USES-2018-014*, <http://uses.icomp.ufam.edu.br/relatorios-tecnicos/>
- De Souza, L. S., & de Aquino Jr, G. S. (2014). Meffortmob: A effort size measurement for mobile application development. *International Journal of Software Engineering & Applications*, vol. 5, n. 4, p. 63.

- De Souza, L. S., & de Aquino Jr, G. S. (2014). Mobile Applications: The Paradox of Software Estimation. In SEKE, p. 59-62.
- De Souza, L. S., & de Aquino, G. S. (2014). The applicability of present estimation models to the context of mobile applications. In Evaluation of Novel Approaches to Software Engineering (ENASE), 2014 International Conference on IEEE, pp. 1-6.
- De Souza, L. S., & de Aquino, G. S. (2014). MEstiAM: Estimation model for mobile applications. In Information Systems and Technologies (CISTI), 2014 9th Iberian Conference on IEEE, pp. 1-7.
- De Souza, L. S., & de Aquino, G. S. (2014). Mobile Application Development: How to Estimate the Effort?. In International Conference on Computational Science and Its Application, pp. 63-72.
- Dombroviak, K. M., & Ramnath, R. (2007). A taxonomy of mobile and pervasive applications. In Proceedings of the 2007 ACM symposium on Applied computing, pp. 1609-1615.
- Engelhart, J., & Langbroek, P. (2001). Function point analysis (FPA) for software enhancement. Nesma.
- Ferrucci, F., Gravino, C., Salza, P., & Sarro, F. (2015). Investigating functional and code size measures for mobile applications. In 2015 41st Euromicro Conference on Software Engineering and Advanced Applications, pp. 365-368.
- Forselius, P. (2004). Finnish software measurement association (FiSMA), FSM working group: FiSMA functional size measurement method v. 1.1.
- Francese, R., Gravino, C., Risi, M., Scanniello, G., & Tortora, G. (2015). On the use of requirements measures to predict software project and product measures in the context of android mobile apps: a preliminary study. In 2015 41st Euromicro Conference on Software Engineering and Advanced Applications, pp. 357-364.
- Glaser, B. G., & Strauss, A. L. (1967). The discovery of grounded theory: strategies for qualitative research Aldine Publishing Company. New York.
- Glass, R. L., & Vessey, I. (1995). Contemporary application-domain taxonomies. IEEE Software, vol. 12, n. 4, pp. 63-76.
- Keränen, H., & Abrahamsson, P. (2005). A case study on naked objects in agile software development. In International Conference on Extreme Programming and Agile Processes in Software Engineering, pp. 189-197.
- Khatibi, E., & Ibrahim, R. (2012). "Efficient Indicators to Evaluate the Status of Software Development Effort Estimation inside the Organizations". In: International Journal of Managing Information Technology, vol.4, n° 3, pp. 23-32.
- Kitchenham, B., e Charters, S. (2007). Guidelines for performing Systematic Literature Reviews in Software Engineering. Staffordshire, UK.
- Kwasnik, B. H. (1999). The role of classification in knowledge representation and discovery. Library trends, vol. 48, n° 1, pp. 22.
- Matos, O., Fortaleza, L., Conte, T., & Mendes, E. (2013). Realising web effort estimation: a qualitative investigation. In Proceedings of the 17th International Conference on Evaluation and Assessment in Software Engineering, pp. 12-23.
- Mendes, E. (2008). Cost Estimation Techniques for Web Projects. IGI Global.

- Mendes, E. (2014). *Practitioner's Knowledge Representation: A Pathway to Improve Software Effort Estimation*. Springer Science & Business, pp. 27.
- Moffitt, T. E. (2017). Adolescence-limited and life-course-persistent antisocial behavior: A developmental taxonomy. In *Biosocial Theories of Crime*, pp. 69-96.
- Nagappan, M., & Shihab, E. (2016). Future trends in software engineering research for mobile apps. In *Software Analysis, Evolution, and Reengineering (SANER), 2016 IEEE 23rd International Conference*, vol. 5, pp. 21-32.
- Nitze, A., Schmietendorf, A., & Dumke, R. (2014). An analogy-based effort estimation approach for mobile application development projects. In *Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2014 Joint Conference of the International Workshop*, pp. 99-103.
- Oligny, S., Desharnais, J. M., & Abran, A. (1999). A method for measuring the functional size of embedded software. In *3rd International Conference on Industrial Automation*, pp. 7-9.
- Oliveira, E., Conte, T., Cristo, M., & Mendes, E. (2016). Software Project Managers' Perceptions of Productivity Factors: Findings from a Qualitative Study. In *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, pp. 15.
- Pawson, R., *Naked Objects*. PhD thesis, Trinity College, Dublin, 2004.
- Pocatilu, P., & Vetrici, M. (2009). M-applications development using high performance project management techniques. In N. E. Mastorakis, A. Croitoru, V. E. Balas, E. Son, & V. Mladenov (Eds.), *WSEAS International Conference. Proceedings. Recent Advances in Computer Engineering (No. 10)*.
- Smite, D., Wohlin, C., Galviņa, Z., & Prikladnicki, R. (2014). An empirically based terminology and taxonomy for global software engineering. *Empirical Software Engineering*, vol. 19, n. 1, pp. 105-153.
- Statista. (2018). "Number of smartphone users worldwide from 2014 to 2020 (in billions)", Disponível em: <<https://www.statista.com/statistics/330695/number-of-smartphone-users-worldwide/>>. Acessado em 01 de Agosto de 2018.
- Strauss, A.; Corbin, J. (1998). *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory*. Thousand Oaks: SAGE Publications, 1998.
- Symons, C. (2001). Come back function point analysis (modernized)—all is forgiven!. In *Proc. of the 4th European Conference on Software Measurement and ICT Control, FESMA-DASMA*, pp. 413-426.
- Thangaratinam, S., Redman, C. W. (2007). The delphi technique. In *the obstetrician & gynaecologist*, vol. 7, n. 2, pp. 120-125.
- Usman, M., Britto, R., Börstler, J., & Mendes, E. (2017). Taxonomies in software engineering: A Systematic mapping study and a revised taxonomy development method. *Information and Software Technology*, vol. 85, pp. 43-59.
- Van Heeringen, H., & Van Gorp, E. (2014). Measure the functional size of a mobile app: Using the cosmic functional size measurement method. In *Software Measurement and the International Conference on Software Process and Product Measurement (IWSM-MENSURA), 2014 Joint Conference of the International Workshop*, pp. 11-16.

- Vliet, H. V. (2000). *Software engineering: Principles and practice* (2nd ed.). New York: John Wiley & Sons.
- Wheaton, G. R. (1968). *Development of a taxonomy of human performance: A review of classificatory systems relating to tasks and performance* (No. AIR-726-12/68-TR-1). AMERICAN INSTITUTES FOR RESEARCH PITTSBURGH PA.
- Whigham, P. A., Owen, C. A., & Macdonell, S. G. (2015). A baseline model for software effort estimation. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 24, n.3, pp. 20.
- Wohlin, C., & Aurum, A. (2015). Towards a decision-making structure for selecting a research design in empirical software engineering. *Empirical Software Engineering*, vol. 20, n. 6, pp. 1427-1455.

APÊNDICE A – ACORDO DE CONFIDENCIALIDADE

Acordo de Confidencialidade sobre Dados para Pesquisa

Este acordo estabelece um entendimento mútuo, entre a equipe de pesquisadores e os colaboradores da empresa, de que os seguintes pontos serão observados no tratamento de toda informação recolhida durante a pesquisa e após a mesma, de maneira permanente.

- 1- Os membros da equipe de pesquisadores, individual e coletivamente, concordam em tratar como confidencial toda informação coletada da pesquisa. Esta informação não será divulgada de maneira que possa ser identificada pessoas ou projetos como fontes de informação, ou mesmo a própria empresa, caso ela não concorde.
- 2- Os resultados da pesquisa serão apresentados somente de forma sumária, de maneira que nenhuma pessoa ou projeto possa ser identificado.
- 3- Cada participante da pesquisa concorda em não discutir, comunicar ou compartilhar informação que tenha obtido durante as entrevistas com nenhuma outra pessoa que não pertença à equipe de pesquisadores.

As assinaturas abaixo expressam a concordância quanto ao cumprimento dos termos deste acordo, por prazo indeterminado.

Local: Universidade Federal do Amazonas - UFAM

Data: __/__/____

Membros da Equipe de Pesquisadores:

Professora Coordenadora: Profa. Tayana Conte / UFAM

Pesquisadora: Ervili Tarsila Brito de Souza/ UFAM

Colaboradores da Organização:

Representante da Empresa: Nome do Representante.

Profa. Tayana Conte
Professora Coordenadora

Ervili Tarsila
Pesquisadora

Nome do Representante
Representante da Empresa

APÊNDICE B – LEGENDA DAS REFERÊNCIAS DAS ANÁLISES DOS DADOS

Código	Referências
EP01	Francese et al. (2015)
EP02	De Souza e Aquino (2014) (A)(B)(C)(D)(F)
EP03	Nitze et al. (2014)
EP04	Heeringen e Gorb (2014)
EP05	Pocatilu e Ventrici (2009) (A)
EP06	Pocatilu e Ventrici (2009) (B)
EP07	Huijgens et al. (2014)
EP08	Dombroviak e Ramnath (2007)
EP09	Keränen e Abrahamsson (2005)
EP11	D'Avanzo et al. (2015)
EP12	Ferrucci et al. (2015) (A)
EP13	Ferrucci et al. (2015) (B)
EP14	Abdullah et al. (2013)
EP15	Abdullah et al. (2014)