

Universidade Federal do Pampa

Wolleson Kevin Kelm

# **Estado da Prática da Engenharia de Requisitos no Desenvolvimento de Aplicações Móveis**

Alegrete

2016



Wolleson Kevin Kelm

## **Estado da Prática da Engenharia de Requisitos no Desenvolvimento de Aplicações Móveis**

Projeto do Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software da Universidade Federal do Pampa como requisito parcial para a obtenção do título de Bacharel em Engenharia de Software.

Orientador: Profa. Dra. Aline Vieira de Mello

Alegrete

2016



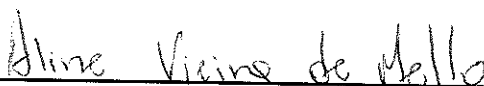
Wolleson Kevin Kelm

## Estado da Prática da Engenharia de Requisitos no Desenvolvimento de Aplicações Móveis


Projeto do Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Software da Universidade Federal do Pampa como requisito parcial para a obtenção do título de Bacharel em Engenharia de Software.

Trabalho de Conclusão de Curso defendido e aprovado em 29 de novembro de 2016

Banca examinadora:



Profa. Dra. Aline Vieira de Mello  
Orientador

  
Prof. Me. João Pablo Silva da Silva  
UNIPAMPA

  
Prof. Me. Jean Felipe P. Cheiran  
UNIPAMPA



*Dedico esse trabalho à minha família que sempre me apoiou nos momentos de dificuldade. Em especial, a minha querida colega Analice. Você estará para sempre em meu coração.*





# Agradecimentos

Agradeço primeiramente a Deus, por ter me dado conhecimento e sabedoria para alcançar meus objetivos.

Aos meus familiares que sempre estavam presentes nos momentos de dificuldade me apoiando com palavras de carinho e incentivo. Agradeço por estarem sempre torcendo pelo meu sucesso.

A minha orientadora, professora Aline Vieira de Mello pela dedicação e suporte no desenvolvimento deste trabalho.

Aos meus colegas e amigos Alex, Anne, Lukas, Cinthia e Luiz Paulo pela amizade e o companheirismo.



*“O mais competente não discute,  
domina a sua ciência e cala-se.” (Voltaire)*



# Resumo

O número de aplicações de software voltado para dispositivos móveis estão em frenética progressão, juntamente com os recursos e o poder de computação oferecidos por esse tipo de equipamento. Embora muitas técnicas “clássicas” da engenharia de software possam ser transferidas para esse domínio, essas aplicações têm de lidar com várias limitações inerentes ao ecossistema móvel que não estão presentes em aplicações desktop convencionais. Uma das áreas com maior importância, tanto na engenharia de software tradicional, quanto na aplicada a dispositivos móveis é a Engenharia de Requisitos. Nesse contexto, o objetivo geral do presente trabalho é identificar o estado da prática da engenharia de requisitos utilizada no mercado para a construção de aplicações móveis. Para atingir tal objetivo, uma pesquisa foi elaborada e aplicada a desenvolvedores de aplicações móveis disponibilizados no Google Play, obtendo 85 respostas de 26 países diferentes. Os resultados apontam que o tempo dedicado à Engenharia de Requisitos é menor do que o recomendado na literatura; menos da metade dos participantes da pesquisa realiza a atividade de inspeção de requisitos e poucos desenvolvedores fazem uso de abordagens para a modelagem de requisitos de software. Portanto, a Engenharia de Requisitos é negligenciada em vários aspectos por grande parte dos desenvolvedores de aplicações móveis. Dessa forma, a academia deveria dar maior ênfase ao estudo da Engenharia de Requisitos, destacando sua importância para o desenvolvimento de software com qualidade.

**Palavras-chave:** Engenharia de Requisitos, Aplicações Móveis, Estado da Prática.



# Abstract

The quantity of software applications related to mobile devices is in frantic progression, as well as the resources and the computer power offered by this kind of device. Though some “classic” techniques of software engineering can be transferred to this domain, these applications have to deal with a lot of limitations inherent to the mobile ecosystem that are not present in conventional desktop applications. One of the areas with the biggest importance, both in the traditional software engineering and on the one applied to mobile devices, is the Requirements Engineering. In this context, the general goal of this work is to identify the state of practice in requirements engineering used in the market for building mobile applications. To achieve this goal, a research was prepared and applied to mobile developers available on Google Play, obtaining 85 responses from 26 different countries. The results show that the time devoted to the Requirements Engineering is lower than recommended in the literature; less than half of survey participants performing the requirements inspection activity and few developers make use of approaches to modeling software requirements. Therefore, Requirements Engineering is neglected in many aspects by most mobile application developers. In this way, the academy should give greater emphasis to the study of Requirements Engineering, highlighting its importance for the development of quality software.

**Key-words:** Requirements Engineering. Mobile Applications. State of Practice.





# Lista de ilustrações

Figura 1 – Protocolo da revisão bibliográfica . . . . .	38
Figura 2 – Processo da pesquisa . . . . .	43
Figura 3 – Vantagens e desvantagens da pesquisa <i>online</i> . . . . .	46
Figura 4 – Países participantes da pesquisa . . . . .	54
Figura 5 – Categoria dos apps participantes da pesquisa . . . . .	55
Figura 6 – Nível de formalidade . . . . .	55
Figura 7 – Atividades de ER realizadas . . . . .	56
Figura 8 – Tempo dedicado à ER . . . . .	56
Figura 9 – Tamanho da equipe de desenvolvimento . . . . .	57
Figura 10 – O papel do analista de requisitos . . . . .	58
Figura 11 – Técnicas de elicitação de requisitos . . . . .	59
Figura 12 – Abordagens utilizadas na modelagem de requisitos . . . . .	60
Figura 13 – Percentual de apps que consideram requisitos não funcionais . . . . .	60
Figura 14 – Requisitos não funcionais considerados . . . . .	61
Figura 15 – Notação utilizada para especificar os requisitos . . . . .	61
Figura 16 – Percentual de apps que realizam inspeção de requisitos . . . . .	62
Figura 17 – Técnica de inspeção de requisitos utilizadas . . . . .	62
Figura 18 – Percentual de apps que utilizam ferramentas CASE . . . . .	63
Figura 19 – Ferramentas CASE utilizadas . . . . .	64
Figura 20 – Número de desenvolvedores x Nível de formalidade . . . . .	64
Figura 21 – Número de desenvolvedores x Tempo dedicado à ER . . . . .	65
Figura 22 – Nível de formalidade x Avaliação dos usuários . . . . .	66
Figura 23 – Tempo dedicado à ER x Avaliação dos usuários . . . . .	66



# Lista de tabelas

Tabela 1 – Síntese dos trabalhos selecionados . . . . .	39
Tabela 2 – Resultado da revisão bibliográfica . . . . .	41
Tabela 3 – Fontes das questões . . . . .	48
Tabela 4 – Formato das questões . . . . .	49
Tabela 5 – Relação dos apps que responderam ao teste piloto . . . . .	50
Tabela 6 – Classificação de correlação . . . . .	52



# Lista de abreviaturas

**apps** aplicações móveis (do inglês *applications*)

**SMTP** Protocolo de Transferência de Correio Simples (do inglês *Simple Mail Transfer Protocol*)

**SWEBOK** *Software Engineering Body of Knowledge* ou Guia do Conhecimento da Engenharia de Software

**UNIPAMPA** Universidade Federal do PAMPA



# Lista de siglas

**ER** Engenharia de Requisitos





# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>25</b>
<b>1.1</b>	<b>Objetivo</b>	<b>26</b>
<b>1.2</b>	<b>Organização do documento</b>	<b>26</b>
<b>2</b>	<b>REFERÊNCIAS TEÓRICAS</b>	<b>29</b>
<b>2.1</b>	<b>Softwares para dispositivos móveis</b>	<b>29</b>
<b>2.2</b>	<b>Engenharia de requisitos (ER)</b>	<b>31</b>
2.2.1	Processos de engenharia de requisitos	31
2.2.2	Elicitação de requisitos	32
2.2.3	Análise de requisitos	33
2.2.4	Especificação de requisitos	34
2.2.5	Validação de requisitos	35
<b>2.3</b>	<b>Survey</b>	<b>36</b>
<b>3</b>	<b>TRABALHOS RELACIONADOS</b>	<b>37</b>
<b>3.1</b>	<b>Metodologia</b>	<b>37</b>
<b>3.2</b>	<b>Análise dos Resultados</b>	<b>39</b>
3.2.1	Qual é o estado da prática em Engenharia de Requisitos no mercado?	39
3.2.2	Quais abordagens foram utilizadas para identificar o estado da prática da Engenharia de Requisitos ?	41
<b>3.3</b>	<b>Conclusões do capítulo</b>	<b>42</b>
<b>4</b>	<b>METODOLOGIA</b>	<b>43</b>
<b>5</b>	<b>DESENVOLVIMENTO</b>	<b>45</b>
<b>5.1</b>	<b>Planejamento e elaboração da pesquisa</b>	<b>45</b>
5.1.1	Objetivo da pesquisa	45
5.1.2	Público-alvo	45
5.1.3	Escolha do instrumento da pesquisa	46
5.1.4	Elaboração do pesquisa	47
<b>5.2</b>	<b>Teste piloto</b>	<b>49</b>
<b>5.3</b>	<b>Execução da pesquisa</b>	<b>51</b>
<b>5.4</b>	<b>Conclusão da pesquisa</b>	<b>51</b>
5.4.1	Análise dos dados	51
<b>6</b>	<b>RESULTADOS</b>	<b>53</b>
<b>6.1</b>	<b>Identificação</b>	<b>53</b>

6.2	Gestão e processo . . . . .	53
6.3	Elicitação de requisitos . . . . .	57
6.4	Análise de requisitos . . . . .	58
6.5	Especificação de requisitos . . . . .	59
6.6	Verificação de requisitos . . . . .	60
6.7	Ferramentas CASE . . . . .	63
6.8	Análise do coeficiente de correlação . . . . .	63
7	CONCLUSÕES . . . . .	67
	REFERÊNCIAS . . . . .	69
	APÊNDICES	73
	APÊNDICE A – QUESTIONÁRIO DA PESQUISA . . . . .	75

# 1 Introdução

O mercado de dispositivos móveis se encontra em constante e acelerado crescimento. Estima-se que no ano de 2015 foram adquiridos mais de meio bilhão (563 milhões) de dispositivos móveis em todo o mundo. Projeções futuras apontam que em 2020 o número de dispositivos móveis conectados à internet chegará a 1,5 per capita (CISCO, 2016), ou seja, mais de um dispositivo por pessoa. Por esse fato, dispositivos móveis, como telefones celulares e *tablets*, tornam-se um alvo promissor para os produtos e serviços fornecidos pela indústria de software.

Embora muitas técnicas “clássicas” da engenharia de software possam ser transferidas para o domínio dos aplicativos móveis, essas aplicações têm de lidar com várias limitações inerentes ao ecossistema móvel que não estão presentes em aplicações *desktop* convencionais (WASSERMAN, 2010). Essas restrições incluem as capacidades limitadas de dispositivos terminais, problemas de comunicação sem fio, problemas de mobilidade, a diversidade de padrões e plataformas operacionais, questões de segurança, e muitos outros (CORRAL, 2014). Além disso, o desenvolvimento deve ser ágil e de baixo custo para obter sucesso em um mercado altamente competitivo que envolve milhões de usuários e centenas de milhares de produtos. Todos esses aspectos colaboram para a construção de um cenário complexo e propício a falhas, representando um grande desafio para a engenharia de software (CORRAL; SILLITTI; SUCCI, 2015).

As pesquisas voltadas para as questões da engenharia de software para aplicações móveis são recente e vem sendo considerado um campo promissor para estudos (NAGAPPAN; SHIHAB, 2016). Uma das áreas com maior importância, tanto na engenharia de software tradicional, quanto na aplicada a dispositivos móveis é a Engenharia de Requisitos (ER), pois ela é uma disciplina significativa e necessária para a produção de qualquer produto de software que obtenha resultado exitoso (RAHMAN, 2014). É por meio da ER que são obtidas do cliente as características que o software deve possuir ao final do desenvolvimento.

A literatura apresenta a ER como sendo a responsável pelo desenvolvimento de abordagens, processos, técnicas, métodos, modelos e ferramentas para ajudar na concepção de software e sistemas, abrangendo as atividades de elicitação, análise, especificação, validação e gestão dos requisitos (PRESSMAN, 2009). Requisitos mal levantados foram reconhecidos como a principal causa de problemas de software: como falhas para atender às expectativas ou na degradação no ambiente controlado pelo software (RAHMAN, 2014).

Apesar da ER ser uma atividade essencial no processo de desenvolvimento de

sistemas, existem poucos estudos referentes a essa atividade aplicada na área de software para dispositivos móveis. Entretanto, é possível encontrar na literatura algumas propostas de práticas, processos e técnicas de ER para aplicações móveis. No entanto, a evolução constante do ambiente de execução móvel desafia várias das premissas destas propostas. Essa situação instiga os pesquisadores e profissionais a acompanhar o crescimento dessa área dinâmica, expandindo, aprofundando e modernizando o corpo de conhecimento em ER para aplicações móveis. Além disso, a literatura que acompanha o tema é limitada e não fornece elementos para avaliar a efetiva utilização e eficácia da ER no ambiente móvel (CORRAL; SILLITTI; SUCCI, 2015).

## 1.1 Objetivo

Dado o contexto, o objetivo geral do presente trabalho é identificar o estado da prática da ER utilizada no mercado para a construção de aplicativos móveis. Para atingir tal objetivo, elaboramos uma pesquisa para ser aplicada com desenvolvedores de aplicativos móveis disponibilizados no Google Play. Complementarmente, definiu-se os seguintes objetivos específicos:

- Comparar o que é recomendado na literatura com o que é aplicado na prática;
- Investigar o nível de maturidade do processo de ER no desenvolvimento de aplicativos móveis;
- Identificar as características da equipe de desenvolvimento de aplicativos móveis;
- Identificar fatos e aspectos que possam auxiliar e melhorar os processos de ER no desenvolvimento de apps;
- Identificar possíveis temas para estudos futuros.

## 1.2 Organização do documento

O restante do trabalho está organizado de acordo com a sequência de capítulos descritos a seguir:

- O [Capítulo 2](#) aborda a base teórica deste trabalho, apresentando os conceitos relacionados a software, engenharia de software, engenharia de requisitos, dentre outros.
- Os trabalhos relacionados são apresentados no [Capítulo 3](#), onde são destacadas as principais contribuições para o desenvolvimento do presente trabalho e, também, as diferenças existentes.

- 
- O [Capítulo 4](#) apresenta a metodologia usada para conduzir o presente trabalho, juntamente com a descrição de cada etapa deste processo.
  - O desenvolvimento deste trabalho é apresentado no [Capítulo 5](#), onde é detalhado como o instrumento de pesquisa foi elaborado, como o teste piloto foi conduzido e como os resultados foram analisados.
  - Os resultados são apresentados e analisados no [Capítulo 6](#), onde é feita uma comparação com o descrito na literatura e também com os resultados descritos nos trabalhos relacionados.
  - O [Capítulo 7](#) traz as considerações finais e os trabalhos futuros.



## 2 Referências teóricas

Neste Capítulo é abordada a base teórica deste trabalho, apresentando conceitos relacionados a software para dispositivos móveis, engenharia de requisitos e *survey*.

### 2.1 Softwares para dispositivos móveis

As aplicações móveis (do inglês *applications*) (**apps**) podem ser definidas como sendo as aplicações desenvolvidas para a atual geração de dispositivos móveis (*smartphones* e *tablets*). Essas aplicações são geralmente distribuídas através de uma plataforma específica em um mercado de aplicativos centralizado. Estudos recentes do mercado de apps centralizada na plataforma da Apple (iOS) e na plataforma do Google (Android) apontam que cada um deles possui mais de 1,5 milhões de aplicativos (**NAGAPPAN; SHIHAB, 2016**). Esses mercados de aplicativos são extremamente populares entre os desenvolvedores devido a sua flexibilidade e rendimento.

As plataformas de desenvolvimento são complexas. Por exemplo, Android e iOS contêm mais de 12 milhões de linhas de código cada um. No entanto, atualmente as ferramentas para o desenvolvimento de apps possibilitam que pessoas com pouco treinamento formal possam criar e vender seus aplicativos ao lado de outros desenvolvidos por grandes equipes. Apesar de alguns apps serem desenvolvidos por amadores, muitos engenheiros de software acreditam que os apps são o sistema de software mais desafiadores em construção na atualidade (**PRESSMAN; MAXIM, 2016**). Dentre as considerações técnicas a serem tratadas pelos apps listam-se:

- **A diversidade de plataformas de hardware e software.** Para que um app execute em um determinado dispositivo móvel, é necessário que este app tenha sido desenvolvido considerando as características de um determinado dispositivo e a versão do sistema operacional que executa sobre ele (**WASSERMAN, 2010**).
- **Muitas estruturas de desenvolvimento e linguagens de programação.** Atualmente, pode-se escrever apps em pelo menos três linguagens de programação diferentes (Java, Objective C e C#) para no mínimo cinco plataformas diferentes (Android, iOS, BlackBerry, Windows e Symbian). Poucos dispositivos móveis permitem que se desenvolva diretamente no aparelho. Os desenvolvedores fazem uso de emuladores executados em sistemas *desktop*, porém esses emuladores não refletem com perfeição as limitações e ambiente do dispositivo móvel em si (**PRESSMAN; MAXIM, 2016**).

- **Muitas lojas de aplicativos com diferentes regras e ferramentas.** Cada plataforma móvel tem sua própria loja de apps e seus próprios padrões de qualidade para aceitar um aplicativo. Assim o desenvolvimento de um app para várias plataformas tem de ocorrer separadamente e cada versão do app necessita de um especialistas em padrões (PRESSMAN; MAXIM, 2016).
- **A utilização de sensores.** Dispositivos móveis atuais incluem um acelerômetro que responde ao movimento do dispositivo, um *touch screen* que responde a numerosos gestos, um sistema de posicionamento global e vários protocolos de aplicações de rede (WASSERMAN, 2010).
- **A interface com o usuário.** Além de possuírem uma tela menor do que as dos computadores pessoais, dispositivos móveis oferecem um conjunto maior de possibilidades de interação, como, por exemplo, interações por voz, toque, gestos e monitoramento do olhos, cenários baseados no reconhecimento de contexto. Outro fator que interfere diretamente na interface do app é o estilo e a aparência que são impostas pela ferramenta de desenvolvimento de uma plataforma específica (PRESSMAN; MAXIM, 2016).
- **O gerenciamento de energia.** A vida da bateria é um das restrições mais limitantes nos dispositivos móveis. Leitura e escrita na memória, uso de conexões sem fio, uso de hardware especializado e velocidade do processador tem impacto direto na vida da bateria. Cabe ao desenvolvedor criar ou adaptar algoritmos visando preservar a carga da bateria (PRESSMAN; MAXIM, 2016).
- **A segurança.** As plataformas móveis permitem a instalação de novas aplicações *malware*<sup>1</sup> que podem afetar o funcionamento global do dispositivo, incluindo a transmissão oculta de dados locais a uma tal aplicação (WASSERMAN, 2010).

A Engenharia de Software para plataformas móveis é um campo relativamente recente e nos fornece uma agenda de pesquisa que inclui investigação teórica, modelos, algoritmos, aplicativos de pesquisa do sistema e middleware, convidando assim a comunidade da Engenharia de Software para abraçar a computação móvel como uma nova fronteira a ser conquistada (CORRAL; SILLITTI; SUCCI, 2015).

Uma das áreas com maior importância, tanto na engenharia de software tradicional, quanto na aplicada a dispositivos móveis é a Engenharia de Requisitos (ER), pois ela é uma disciplina significativa e necessária para a produção de qualquer produto de software que obtenha resultado exitoso (RAHMAN, 2014).

---

<sup>1</sup> O “malware”, termo do inglês “malicious software” (software nocivo ou software malicioso), é um software destinado a infiltrar-se em um sistema computacional alheio de forma ilícita, com o intuito de causar alguns danos, alterações ou roubo de informações (confidenciais ou não).



## 2.2 Engenharia de requisitos (ER)

A Engenharia Requisitos(ER) se preocupa com a elicitação, análise, especificação e validação de requisitos de software. É amplamente reconhecido dentro da indústria que os projetos de software são comprometidos quando estas atividades são mal executadas (BOURQUE; FAIRLEY et al., 2014).

Compreender os requisitos é uma das tarefas mais difíceis enfrentadas pelos engenheiros de software. Requisitos de um sistema são descrições do que o sistema deve fazer, os serviços que oferece e as restrições sobre seu funcionamento e sua implantação. Eles são comumente classificados em requisitos funcionais e não funcionais (SOMMERVILLE, 2011).

- **Requisitos funcionais** são declarações de serviços que o sistema deve fornecer, de como o sistema deve reagir as entradas específicas e de como o sistema deve se comportar em determinadas situações. Às vezes também pode fornecer especificações de o que o sistema não deve fazer em determinadas situações.
- **Requisitos não funcionais** são restrições aos serviços ou funções oferecidos pelo sistema. Incluem restrições de tempo, restrições no processo de desenvolvimento e restrições impostas pelas normas. Ao contrário das características individuais ou serviços do sistema, os requisitos não funcionais, muitas vezes, aplicam-se ao sistema como um todo e, frequentemente, são mais críticos que requisitos funcionais individuais.

A ER fornece os instrumentos apropriados para compreender o desejo do cliente, analisando as necessidades do mesmo, negociando soluções razoáveis e viáveis, validando as especificações e gerenciando as necessidades à medida do desenvolvimento do software (PRESSMAN, 2009).

### 2.2.1 Processos de engenharia de requisitos

Os processos de ER tem como objetivo gerar documentos de requisitos que especificam um sistema que satisfaz os requisitos dos stakeholders. No processo de ER podemos encontrar quatro atividades principais (SOMMERVILLE, 2011):

1. **Estudo de viabilidade** é a atividade que realiza uma estimativa acerca das necessidades já identificadas do usuário. O estudo deve considerar se o sistema proposto é rentável do ponto de vista de negócio e se ele pode ser desenvolvido no âmbito das atuais restrições orçamentais. Esse estudo deve ser rápido, barato e o resultado deve informar a decisão de avançar ou não, com uma análise mais detalhada.

2. **Elicitação e análise** de requisitos é o processo de derivação dos requisitos do sistema por meio da observação de software existente, discussões com usuários, análise de tarefas, entre outras etapas.
3. **Especificação de requisitos** é a atividade de traduzir as informações obtidas durante a análise em um documento que defina um conjunto de requisitos.
4. **Validação de requisitos** tem como objetivo verificar o realismo, consistência e completude dos requisitos. Durante esse processo, erros no documento de requisitos são inevitavelmente descobertos. Em seguida, o documento deve ser corrigido.

O processo de ER proposto em *Software Engineering Body of Knowledge* ou Guia do Conhecimento da Engenharia de Software (SWEBOK) (BOURQUE; FAIRLEY et al., 2014) segue uma abordagem semelhante a apresentada por Sommerville (2011), porém com a diferença que decompõem a atividade de elicitación e análise de requisitos em duas atividades: elicitación de requisitos e análise de requisitos. O objetivo de análise de requisitos, segundo o Bourque, Fairley et al. (2014), é detectar e resolver conflitos, descobrir os limites do software e como ele deve interagir com seu ambiente operacional e organizacional. Nas próximas seções são detalhadas as atividades de ER do processo proposto por SWEBOK (BOURQUE; FAIRLEY et al., 2014), o qual é adotado neste trabalho.

## 2.2.2 Elicitación de requisitos

Elicitación de requisitos é uma atividade que tem por objetivo reunir todas informações possíveis sobre o sistema requerido e os sistemas existentes e separar dessas informações os requisitos de usuário e de software (SOMMERVILLE, 2011). Existem três fontes principais de requisitos (KLAUS, 2012):

- **Stakeholders** são as pessoas ou organização que tem influência direta sobre os requisitos do sistema. Como exemplo podemos citar os usuários do sistema, desenvolvedores e clientes.
- **Documentos** muitas vezes podem conter informações importantes fornecendo requisitos. Documentação sobre normas e leis e relatório de falhas de sistemas legados podem ser considerados como exemplo.
- **Sistemas em operação** podem ser sistemas anteriores, legados ou até mesmo sistemas concorrentes. Tendo a oportunidade de testar o sistema, os *Stakeholders* podem formar ideias sobre o sistema atual e solicitar extensões, modificações com base nas suas impressões.

Também é possível obter requisitos de outras fontes como, por exemplo, o ambiente operacional onde software será executado; o ambiente organizacional, quando o software necessita suportar um processo de negócio; e a política interna da organização (BOURQUE; FAIRLEY et al., 2014).

Uma vez que as fontes do requisitos foram identificadas, o engenheiro de software pode começar a suscitar os requisitos. Essa é uma atividade muito difícil de ser realizada. O engenheiro precisa estar atento aos fatos, pois muitas vezes as fontes podem ter dificuldade em descrever sua tarefa ou omitir informações importantes (BOURQUE; FAIRLEY et al., 2014). Existem ferramentas que ajudam o engenheiro a contornar essas dificuldades, em seguida algumas são apresentadas:

- **Entrevistas** são provavelmente uma das técnicas mais tradicionais e são comumente usadas para elicitación de requisitos. Elas oferecem uma solução eficaz para coletar grandes quantidades de dados rapidamente (SOMMERVILLE, 2011).
- **Observação** é uma técnica onde o engenheiro acompanha o especialista ou os usuários do sistema no próprio local de trabalho, observando e documentando os processos e procedimentos executados (KLAUS, 2012).
- **Prototipagem** é uma técnica útil para construir a interface gráfica. Nessa técnica, uma versão inicial do produto é disponibilizada aos clientes que visualizam e, a partir disso, sugerem melhorias e novos requisitos (REHMAN; KHAN; RIAZ, 2013).
- **Brainstorming** é usado para gerar inúmeras ideias em um intervalo de tempo curto, onde um problema específico é discutido informalmente entre os participantes de diferentes grupos de interessados (REHMAN; KHAN; RIAZ, 2013).

### 2.2.3 Análise de requisitos

A análise de requisito tem como principal objetivo transformar as necessidades dos usuários em requisitos de software, identificar e resolver conflitos entre os requisitos, definir os limites do software e como ele deve interagir com o ambiente (BOURQUE; FAIRLEY et al., 2014).

Tradicionalmente a análise de requisitos é reduzida a apenas modelagem conceitual, a qual tem por objetivo contribuir para a compreensão do problema. Mas além do modelo conceitual deve-se realizar a classificação de requisitos, a busca por conflitos entre os requisitos e a negociação de possíveis conflitos entre as partes interessadas (BOURQUE; FAIRLEY et al., 2014).

Algumas técnicas podem apoiar a análise de requisitos. Analysis (2009) cita algumas delas:

- **Análise de regras de negócios.** Regras de negócios podem estar separadas de outros requisitos para implementação e gerenciamento em um mecanismo de regras de negócio ou algo similar.
- **Cenários e casos de uso.** Descreve os requisitos que suportam as metas individuais de cada ator ou a resposta para o evento disparador.
- **Histórias do usuário.** Descreve os objetivos das partes interessadas que a solução irá suportar.
- **Decomposição funcional.** Decompõe uma unidade organizacional, um escopo de produto ou algo semelhante em componentes menores. Cada parte pode ter os seus próprios conjuntos de requisitos.

## 2.2.4 Especificação de requisitos

A especificação de requisitos é a atividade de documentar os requisitos de usuário e de software. Os requisitos devem estar claros, inequívocos, completos, consistentes e ser de fácil compreensão (SOMMERVILLE, 2011). Documentos de requisitos são o meio normal para gravar e comunicar os requisitos. É uma boa prática separar os requisitos do usuário e os requisitos de software em documentos diferentes (BOURQUE; FAIRLEY et al., 2014).

Os requisitos de usuário devem descrever os requisitos de modo que sejam compreensíveis para os usuários do sistema que não tenham conhecimentos técnicos detalhados. Além disso, jargões de software, notações estruturadas ou notações formais não devem ser usados. O ideal é que os requisitos de usuário sejam descritos em linguagem natural, com tabelas simples e diagramas intuitivos. Já requisitos de sistema são versões expandidas dos requisitos de usuário. Eles acrescentam detalhes e explicam como os requisitos de usuário devem ser atendidos pelo sistema (SOMMERVILLE, 2011)

Existem várias maneiras diferentes de se especificar os requisitos, em seguida são listadas as formas mais comuns (SOMMERVILLE, 2011):

- **Linguagem natural.** Os requisitos são escritos em frases numeradas em linguagem natural. Cada frase deve expressar um requisito.
- **Linguagem natural estruturada.** Os requisitos são escritos em linguagem natural em um formulário padrão. Cada campo fornece informações sobre um aspecto do requisito.
- **Notações gráficas.** Os requisitos funcionais são comumente definidos em modelos gráficos suplementados por anotações de texto, diagramas de caso de uso e de sequência.

- **Especificações Formais.** Notações baseadas em conceitos matemáticos, como máquinas de estado finito ou conjuntos, reduzem a ambiguidade de um documento de requisitos. Em contrapartida, a maioria dos clientes não entende uma especificação formal, assim não podem verificar se as especificações representam o que eles querem.

### 2.2.5 Validação de requisitos

A validação de requisitos é o processo pelo qual se verifica se os requisitos definem o sistema que o cliente realmente quer (SOMMERVILLE, 2011). A validação está preocupada com a verificação dos documentos, buscando encontrar omissões, conflitos e ambiguidades, e desta forma assegurar que os requisitos sigam padrões de qualidade. Os requisitos devem ser escritos de uma maneira que deixa o menor espaço possível para erros de interpretação. (BOURQUE; FAIRLEY et al., 2014).

Outra razão para que a validação de requisitos seja realizada é o fato de que quanto mais cedo um defeito nos requisitos for corrigido, menores serão os custos associados a sua correção. Requisitos completos e sem defeitos formam a base para o desenvolvimento de um sistema bem sucedido (KLAUS, 2012).

A maneira mais eficiente de validação é através da utilização de análises formais dos documentos de requisitos. Um grupo de revisores é constituído para procurar erros, suposições equivocadas, a falta de clareza e o afastamento da prática padrão. Além da documentação, a qualidade dos modelos conceituais desenvolvidos durante a análise deve ser validada. Por exemplo, em modelos de objeto, é útil executar uma análise estática para verificar a existência de comunicação entre objetos (BOURQUE; FAIRLEY et al., 2014).

As técnicas de validação manuais, genericamente chamadas de revisões, são frequentemente utilizadas para a validação de requisitos. Algumas dessas técnicas são apresentadas a seguir:

- **Checklists** são úteis como uma técnica de controle da qualidade para a documentação dos requisitos. Eles podem incluir um conjunto padrão de elementos de qualidade que o analista de negócios (ou outros revisores) usam para validar os requisitos, ou ser desenvolvidos especificamente para capturar questões relacionadas ao projeto (ANALYSIS, 2009).
- **DBR (Defect-Based Reading)** é uma técnica que se concentra em tipos específicos de defeito em requisitos, onde os requisitos são expressos pela notação de máquinas de estado (SALGER; ENGELS; HOFMANN, 2009).

- **MBR (Metric-Based Reading)** é uma abordagem sistemática onde se faz uso de um conjunto de heurísticas para melhorar a eficácia e a eficiência da inspeção (BERNÁRDEZ; DURÁN; GENERO, 2004).
- **PBR (Perspective-Based Reading)** é uma técnica que se concentra em analisar os documentos de diferentes perspectivas dos usuários, minimizando a sobreposição entre as falhas encontradas pelos revisores (SALGER; ENGELS; HOFMANN, 2009).

## 2.3 Survey

O presente trabalho é classificado com um *Survey*. *Survey* podem ser descritos como a obtenção de dados ou informações sobre características, ações ou opiniões de determinado grupo de pessoas (FREITAS et al., 2000). Os *surveys* em geral podem ser (WOHLIN et al., 2012):

- **Descritivos** utilizados, por exemplo, para determinar a distribuição de atributos ou características;
- **Explanatórios** utilizados, por exemplo, para explicar o porquê dos desenvolvedores escolherem uma determinada técnica;
- **Explorativos** utilizados, por exemplo, em um estudo preliminar para uma investigação mais profunda.

Um *survey* é, tipicamente, feito através da aplicação de questionários ou entrevistas, visando coletar dados tanto qualitativos quanto quantitativos que respondam a uma (ou mais) questão de pesquisa. A coleta de dados é feita em uma amostra da população sob estudo, onde essa amostra deve ser representativa. As questões apresentadas no questionário ou entrevista devem ser claras e não ambíguas. Os resultados são analisados buscando-se extrair conclusões que possam ser generalizadas.

Uma das vantagens que os *surveys* oferecem é a possibilidade de levantar um grande número de variáveis a serem avaliadas. Porém é necessário obter o maior número de possibilidades de análise com o menor número de perguntas possíveis, porque pesquisas com muitas perguntas são entediadas para os participantes responderem, e a quantidade de dados pode, conseqüentemente, diminuir (WOHLIN et al., 2012).

## 3 Trabalhos relacionados

Este Capítulo apresenta a metodologia de busca dos trabalhos relacionados, assim como os trabalhos resultantes dessa busca. Para cada trabalho relacionado, é feita a análise de sua contribuição para o presente trabalho. Ao final do Capítulo, é destacado como o presente trabalho se diferencia dos demais.

### 3.1 Metodologia

Uma adaptação dos processos de revisão bibliográfica propostos por [Biolchini et al. \(2005\)](#) e [Conforto, Amaral e Silva \(2011\)](#) foi utilizada. Inicialmente, definiu-se o objetivo da pesquisa: encontrar trabalhos que pesquisavam a ER no mercado e descobrir que abordagens foram usadas para a coleta de dados. Em seguida elaborou-se duas questões para guiar o processo de busca:

1. “Qual é o estado da prática em Engenharia de Requisitos no mercado?”
2. “Quais abordagens foram utilizadas para identificar o estado da prática da Engenharia de Requisitos?”

A partir dessas questões foi gerada a seguinte *String* de busca: ( ( “Market Research” OR “Research In Industry” OR “Market Consultation” OR “ Survey industrial” OR “Survey Market” ) AND ( “requirements elicitation” OR “requirements engineering” OR “requirements specification” ) ).

Sobre o resultado da busca, foram adotados os seguintes critérios de inclusão e exclusão:

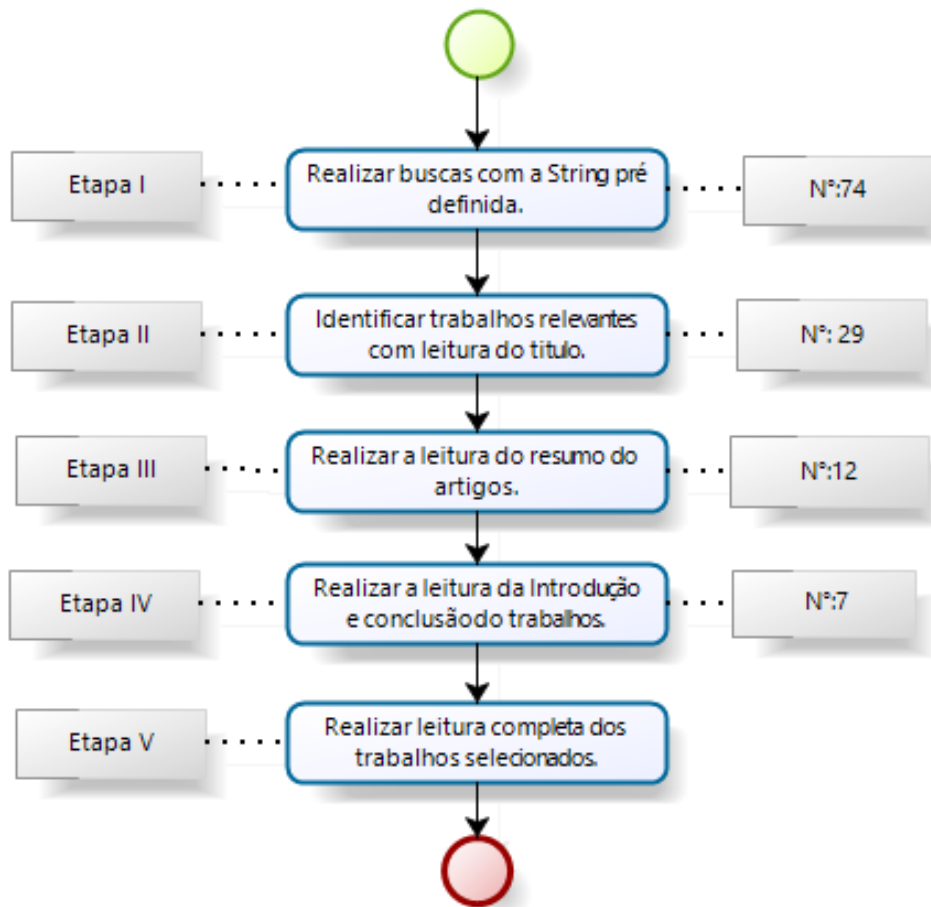
- **Inclusão:** livros, artigos, sites, teses de mestrado ou doutorado que descrevem pesquisas na área de ER. Trabalhos publicados a partir do ano de 2008 e disponíveis para download, seja de forma gratuita ou em bases de dados que acadêmicos da UNIPAMPA tenham acesso.
- **Exclusão:** Trabalhos que não tenham sido publicados em revistas, conferências, eventos ou não estejam associados a nenhuma universidade. Trabalhos que estão fora do domínio da engenharia de software.

A base de busca utilizada foi a Scopus. Essa base foi escolhida porque indexa outras bases de pesquisa como: IEEE (Institute of Electrical and Electronics Engineers),

Xplore Digital Library, ACM (Association Computing Machinery) Digital Library, Science Direct e Springer. Além de oferecer melhores resultados com busca por Strings do que outras bases de dados.

A busca foi realizada e os resultados passaram por cinco etapas de seleção, conforme estabelecido no protocolo da pesquisa e apresentado na [Figura 1](#).

Figura 1 – Protocolo da revisão bibliográfica



Na etapa I foi realizada a busca com as *strings*, sendo obtidos 74 trabalhos relacionados. Na etapa II do protocolo, foram lidos os títulos dos trabalhos e analisados quais estavam disponíveis por completo e se estavam coerentes com os objetivos da pesquisa. Ao final da etapa II, o número de trabalhos relacionados foi reduzido para 29. A etapa III consistiu em avaliar os trabalhos por meio da leitura dos resumos, resultando em 12 trabalhos relacionados. Em seguida na etapa IV, os trabalhos aprovados passaram por uma leitura da introdução e conclusão, onde, finalmente, foram selecionados os 7 trabalhos que foram lidos por completo na última etapa.



## 3.2 Análise dos Resultados

Nesta seção, são apresentados e analisados os resultados para as questões de pesquisa, parte motivadora desse trabalho.

### 3.2.1 Qual é o estado da prática em Engenharia de Requisitos no mercado?

A [Tabela 1](#) sintetiza as informações dos trabalhos selecionados, apresentando a referência, o objetivo e a principal descoberta de cada trabalho.

Tabela 1 – Síntese dos trabalhos selecionados

Referência	Objetivo da Pesquisa	Principal descoberta
( <a href="#">KHAN et al., 2015</a> )	Identificar a atual situação da verificação e validação de requisitos	Inspeções e revisões são as técnicas mais utilizadas
( <a href="#">KASSAB; NEILL; LAPLANTE, 2014</a> )	Trazer à tona o estado da arte da ER	Brainstorming é a técnica de elicitação de requisitos mais usada
( <a href="#">REGNELL et al., 2012</a> )	Ressaltar os desafios associados aos requisitos de qualidade (requisitos não funcionais)	A usabilidade e a segurança são considerados os requisitos com maior prioridade
( <a href="#">TALBOT, 2011</a> )	Apontar as práticas e os problemas da ER em pequenas e médias empresas de desenvolvimento de software	Os principais problemas de ER são o aumento do escopo e mudanças de requisitos
( <a href="#">OSSADA; MARTINS, 2010</a> )	Revelar o estado da ER para sistemas embarcados	Negligência na condução de processos da ER
( <a href="#">SOLEMON; SAHIBUDDIN; GHANI, 2009</a> )	Investigar o estado atual dos problemas da ER e práticas em empresas	Empresas com classificação CMMI alta nem sempre possuem práticas eficazes de ER
( <a href="#">ROUIBAH; AL-RAFEE, 2009</a> )	Identificar as principais técnicas e ferramentas de ER	Entrevistas e questionários são as técnicas mais usadas

Em sua pesquisa [Kassab, Neill e Laplante \(2014\)](#) identificaram que grande maioria dos entrevistados afirmam fazer uso de métodos ágeis. Esse fato pode estar relacionado com outra descoberta de seu trabalho, onde, ele aponta que 48% dos entrevistados afirmam não fazer uso de nenhuma abordagem ou método para a análise e modelagem de requisitos. Isso porque os métodos ágeis sugerem que não se deve gastar muito tempo com

a documentação de requisitos [Ágil \(2011\)](#), assim os desenvolvedores acabam ignorando qualquer tipo de metodologia para a análise e modelagem de requisitos.

Apesar da literatura afirmar que de 40% a 60% de todos os problemas encontrados em um projeto de software são causados por falhas no processo da ER e que a realização das atividades de validação e verificação de requisitos podem diminuir consideravelmente o número dessas falhas ([DIAS; ARAÚJO, 2008](#)). [Kassab, Neill e Laplante \(2014\)](#) em sua pesquisa constataram que mais de 50% das empresas não realizam as atividades de validação e verificação de requisitos. [Ossada e Martins \(2010\)](#) concluíram que existe uma certa negligência na condução de processos de ER no cenário industrial.

Segundo [Sommerville \(2011\)](#), embora as especificações formais possam reduzir a ambiguidade de um documento de requisitos, a maioria dos clientes não entende uma especificação formal, inviabilizando que os mesmos verifiquem se a especificação representa o que desejam. A análise de [Sommerville \(2011\)](#) condiz com os resultados encontrados na pesquisa de [Kassab, Neill e Laplante \(2014\)](#), onde ele aponta que menos de 1% dos entrevistados faz uso da linguagem formal, ressaltando a popularidade das especificações em linguagem natural.

[Sommerville \(2011\)](#) afirma que deixar de atender a um requisito não-funcional pode significar a inutilização de todo o sistema. Assim podemos concluir que os resultados da pesquisa de [Regnell et al. \(2012\)](#) são preocupantes, pois foi constatado que os requisitos não-funcionais muitas vezes são ignorados no desenvolvimento do projeto. [Regnell et al. \(2012\)](#) aponta que usabilidade e segurança são os requisitos não-funcionais com maior prioridade.

Segundo as pesquisa de [Talbot \(2011\)](#) e [Ossada e Martins \(2010\)](#), o tempo gasto com o processo de ER é de 17% e 23% do total do desenvolvimento, respectivamente, ambos inferiores ao tempo sugerido por [Berkun \(2005\)](#) que é de 33,3%. Isso demonstra que a ER é um tanto desvalorizada em questão do tempo dedicado para sua realização, mesmo com toda sua importância para o desenvolvimento de software.

[Solemon, Sahibuddin e Ghani \(2009\)](#) identificaram que as empresas de software estão atualmente enfrentando grandes desafios na obtenção de seus requisitos devido a fatores organizacionais e técnicos. Além disso, descobriram que as classificações de maturidade geralmente não se correlacionam ao melhor desempenho e não indicam que empresas com uma maturidade alta tenham práticas eficazes de ER.

Conforme as pesquisas de [Rouibah e Al-Rafee \(2009\)](#), [Ossada e Martins \(2010\)](#) e [Kassab, Neill e Laplante \(2014\)](#), questionários, entrevistas e brainstormings são as técnicas mais conhecidas e usadas na etapa de elicitação de requisitos. [Khan et al. \(2015\)](#) identificou que revisões e inspeções são as técnicas mais utilizadas para reduzir/minimizar erros nos requisitos, isso porque os especialistas acreditam que essas técnicas são fáceis

de implementar, tem menor custo e consomem menos tempo em comparação com outras estratégias. Esses resultados vão ao encontro do mencionado por [Zowghi e Coulin \(2005\)](#).

[Klaus \(2012\)](#) e [Knuth \(1984\)](#) afirmam que requisitos ambíguos, incompletos e incorretos são os principais problemas na ER. A pesquisa de [Talbot \(2011\)](#) coincide com as análises de [Klaus \(2012\)](#) e [Knuth \(1984\)](#), pois ela apontou a ambiguidade como sendo o maior problema relacionado a ER. Já a pesquisa de [Talbot \(2011\)](#) também indicou que requisitos incompletos e incorretos são uma das principais dificuldades enfrentadas pela ER, mas segundo sua pesquisa o maior problema ligado a ER está relacionado com o aumento do escopo e mudança dos requisitos.

### 3.2.2 Quais abordagens foram utilizadas para identificar o estado da prática da Engenharia de Requisitos ?

Na [Tabela 2](#) apresentamos a abrangência territorial de cada um dos trabalhos selecionados, a ferramenta de pesquisa utilizada e o número de respostas obtidas com o equivalente percentual de taxa de retorno, quando disponibilizado.

Tabela 2 – Resultado da revisão bibliográfica

Referência	Abrangência da pesquisa	Ferramenta de pesquisa	Número de Respostas
<a href="#">(KHAN et al., 2015)</a>	Paquistão	Questionário online	55 (ND)
<a href="#">(KASSAB; NEILL; LAPLANTE, 2014)</a>	23 países	Questionário online	250 (8%)
<a href="#">(REGNELL et al., 2012)</a>	Suécia	Entrevistas	11 (ND)
<a href="#">(TALBOT, 2011)</a>	Nova Zelândia	Questionário online	30 (8,8%)
<a href="#">(OSSADA; MARTINS, 2010)</a>	Brasil	Questionário online	53 (ND)
<a href="#">(SOLEMON; SAHIBUDDIN; GHANI, 2009)</a>	Malásia	Questionário online	64 (13%)
<a href="#">(ROUIBAH; AL-RAFEE, 2009)</a>	Kuwait	Questionário online	87 (49%)

ND = informação não disponível.

A maioria dos trabalhos selecionados se dedicaram a coletar os dados em apenas um país, com exceção do trabalho de [Kassab, Neill e Laplante \(2014\)](#), onde foi realizada uma pesquisa exploratória com profissionais da indústria de software e estudantes de TI de 23 países da América e Europa.

Sobre os instrumentos para obter informações do mercado, o questionário online é o instrumento mais usado, o que é justificável pela rapidez, menor custo e capacidade de atingir uma população específica, além de permitir ao respondente a possibilidade de responder da maneira que for mais conveniente e no tempo e local apropriado [Vieira, Castro e Junior \(2010\)](#). Apenas [Regnell et al. \(2012\)](#) optaram por fazer uso de entrevistas, pois elas permitem alcançar uma compreensão mais profunda sobre o assunto. Porém,

o uso de entrevistas traz algumas desvantagens como: maior tempo para a realização da pesquisa, a opinião do investigador pode influenciar nos resultados e a análise dos resultados é mais complexa.

A taxa de retorno das pesquisas - em que essa informação foi disponibilizada - em geral ficou abaixo de 15%, com exceção do trabalho de [Rouibah e Al-Rafee \(2009\)](#) que atingiu quase 50%. Uma das possíveis causas para essa baixa taxa de retorno é a extensão das pesquisas, pois pesquisas com um número grande de questões geralmente tendem a ter uma baixa taxa de retorno ou serem parcialmente respondidas. Segundo [Kassab, Neill e Laplante \(2014\)](#), a maioria dos participantes abortaram sua pesquisa antes da questão 23, alegando fadiga. Um fator que pode auxiliar no aumento da taxa de retorno é o uso de questões fechadas, onde o participante escolhe uma resposta a partir de um conjunto de opções pré-determinadas.

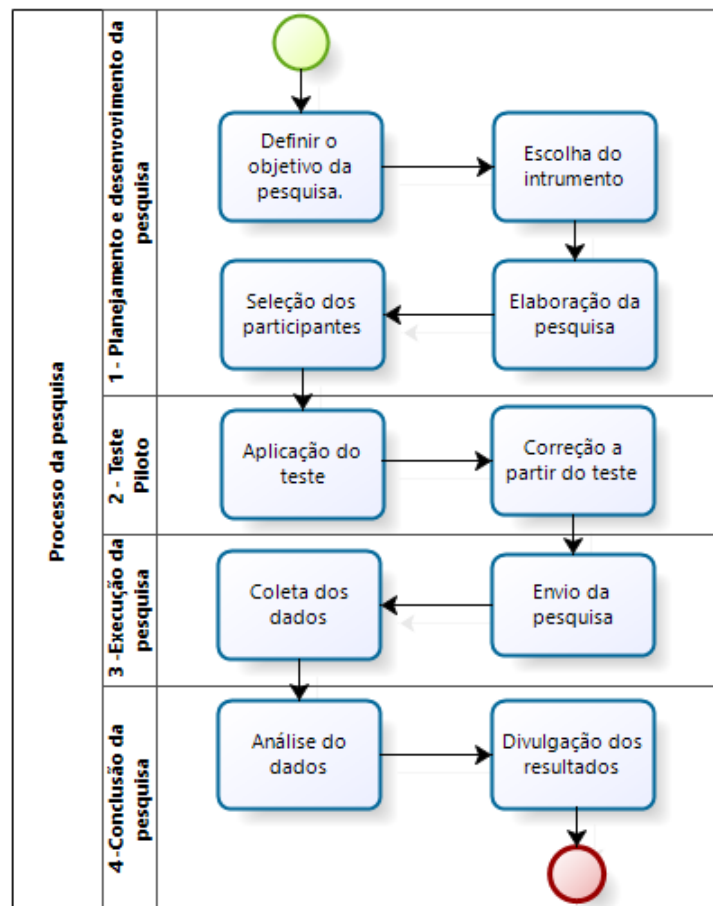
### 3.3 Conclusões do capítulo

Neste Capítulo foram apresentados e discutidos trabalhos que pesquisaram técnicas e práticas da ER adotadas no mercado. Essa revisão possibilitou comparar as práticas/métodos/técnicas indicadas na literatura com as que são de fato adotadas no mercado. Adicionalmente, esses trabalhos nos auxiliaram no desenvolvimento da metodologia e na elaboração do questionário de nossa pesquisa. Outra contribuição foi o relato de alguns problemas que podem ocorrer, nos auxiliando a pensar estratégias para evitá-los.

## 4 Metodologia

Para a realização da pesquisa foi desenvolvido um processo com base em [Kasunic \(2005\)](#). A [Figura 2](#) apresenta o processo desenvolvido, o qual é dividido em quatro etapas que são detalhadas a seguir.

Figura 2 – Processo da pesquisa



A primeira etapa do processo é nomeada Planejamento e Elaboração da Pesquisa e é composta por quatro atividades:

- 1. Definição do objetivo da pesquisa** onde é definida a área de aplicação da pesquisa, busca-se identificar o que já existe na área e, por fim, é apresentado qual é o objetivo específico da pesquisa.
- 2. Identificação do público-alvo** onde é especificado quem realmente vai responder a pesquisa, busca-se identificar o quão grande é o público-alvo e as formas de garantir que o resultados da pesquisa sejam relevantes.

3. **Escolha do instrumento** onde é definido qual vai ser o instrumento usado para a coleta de dados.
4. **Elaboração da pesquisa** onde são definidas as características ligadas ao *design* do questionário, como por exemplo, número de questões, idioma, formato das questões etc. As questões devem ser cuidadosamente formuladas para facilitar a análise e interpretação.

Na segunda etapa, Teste Piloto, são realizadas duas atividades:

1. **Aplicação do teste** onde é realizado o teste piloto para analisar a coerência das questões e identificar quais são os potenciais problemas e taxa de retorno.
2. **Correção a partir do teste** onde são realizadas as modificações no instrumento a fim de corrigir os problemas detectados no teste.

A terceira etapa do processo, Execução da Pesquisa, também é composta por duas atividades:

1. **Envio da pesquisa** tem como objetivo definir a forma como a pesquisa chegará até o público-alvo.
2. **Coleta dos dados** se concentra em coletar as respostas dos participantes da pesquisa.

Por fim, na última etapa, Conclusão da Pesquisa, é realizada a atividade de análise dos dados coletados e os resultados dessa análise são publicados.

## 5 Desenvolvimento

Neste Capítulo é descrito como as etapas do processo de pesquisa ([Figura 2](#)) foram desenvolvidas, ou seja, como a pesquisa foi planejada e elaborada, como o teste piloto foi conduzido, como a pesquisa foi executada e como os dados foram analisados. A atividade de divulgação dos resultados é apresentada no [Capítulo 6](#). As seções deste Capítulo seguem as atividades propostas no processo, ilustrado no [Capítulo 4](#).

### 5.1 Planejamento e elaboração da pesquisa

#### 5.1.1 Objetivo da pesquisa

As pesquisas sobre engenharia de software aplicada a aplicações móveis teve início recente e vem sendo considerado um campo promissor para estudos ([RAHMAN, 2014](#)). A ER é uma das áreas mais importantes tanto para o desenvolvimento de softwares tradicionais, quanto para aplicativos móveis, pois requisitos completos e sem defeitos formam a base para o desenvolvimento de um sistema bem sucedido. Baseado nesse cenário, chegamos ao objetivo dessa pesquisa: “identificar o estado da prática da ER utilizada no mercado para a construção de aplicativos móveis”. Espera-se que com os resultados seja possível apontar fatos e aspectos que possam auxiliar e melhorar os processos de ER no desenvolvimento de apps e, além disso, identificar possíveis temas para estudos futuros.

#### 5.1.2 Público-alvo

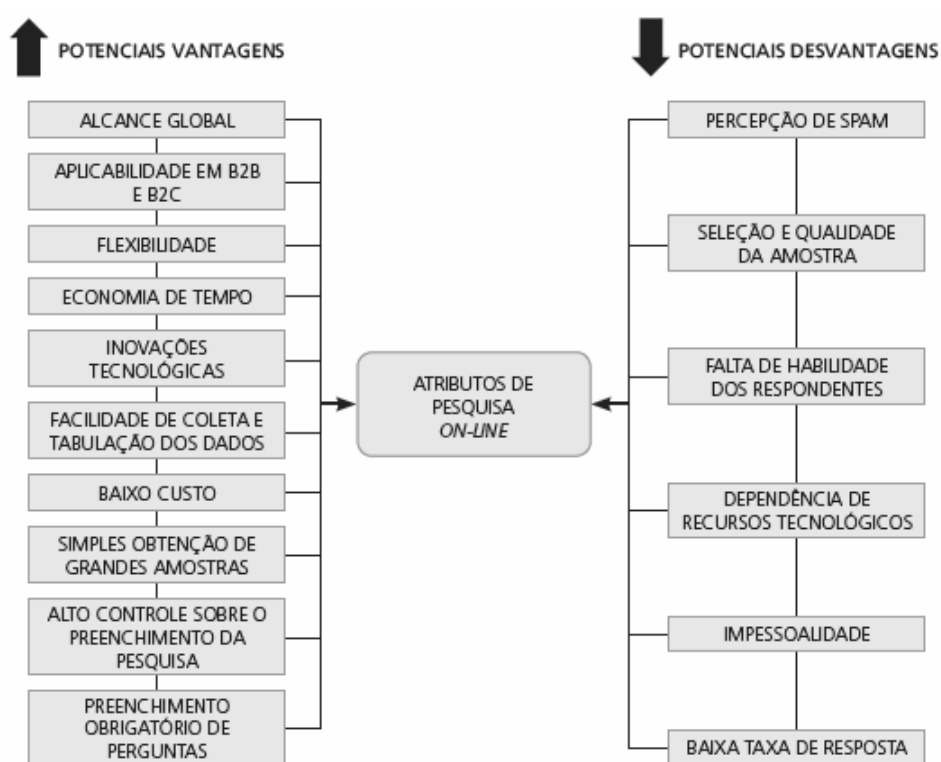
Considerando que o Android é o sistema operacional móvel mais utilizado no mundo ([NETMARKETSHARE, 2015](#)), o público-alvo dessa pesquisa são os desenvolvedores de aplicativos móveis disponibilizados no Google Play. O Google Play é o principal mercado centralizado de apps para Android e conta com uma base de mais de 2 milhões de apps divididos em 30 categorias ([STATISTICS, 2016](#)). Para cada app é disponibilizado o número de instalações, avaliação do usuário e contato com desenvolvedor. Essas informações são úteis para o envio da pesquisa, mas principalmente para a etapa de análise dos dados.

Segundo [Rumsey \(2009\)](#), em geral o tamanho da amostra deve ficar acima de 30 para que a teoria estatística funcione. Cabe destacar que 30 não é um número mágico e sim apenas uma regra geral.

### 5.1.3 Escolha do instrumento da pesquisa

Como método de coleta de dados para este trabalho foi decidido realizar uma pesquisa online através de um questionário. Esse método foi selecionado por ser o mais comumente utilizado em *Surveys*, ter baixo custo e possuir ferramentas que facilitam a análise dos resultados. Na [Figura 3](#) são apresentadas mais vantagens e algumas desvantagens das pesquisas online.

Figura 3 – Vantagens e desvantagens da pesquisa *online*



Fonte: (VIEIRA; CASTRO; JUNIOR, 2010)

Após uma avaliação de várias ferramentas de pesquisa online grátis como por exemplo Survey Monkey Basic, QuestionPro Livre, Survio, Evalandgo entre outras, o Google Forms foi escolhido. Ele oferece mais vantagens em relação às outras, como por exemplo, a possibilidade de criar um número indeterminado de questões e receber um número ilimitado de respostas. Google Forms também oferece a possibilidade de criar vários tipos diferentes de perguntas e permite o uso de plugins complementares para realização de diferentes formas de análise, tanto qualitativa como quantitativa. Os resultados podem ser exportados como um arquivo do Excel.



### 5.1.4 Elaboração do pesquisa

A fim de alcançar um maior número de respostas, o número de perguntas feitas foi limitada a 16, buscando sempre que possível fazer uso de perguntas fechadas. O design das questões também levou em conta a quantidade de explicação que era necessária para que os participantes a compreendessem. Devido ao fato da pesquisa buscar respostas no desenvolvimento de aplicativos móveis de diferentes nacionalidades, o idioma escolhido para o questionário foi o inglês.

O questionário foi dividido em 7 sessões diferentes. A primeira busca identificar informações do aplicativo como: nome, categoria e país de origem. A segunda sessão tem como objetivo obter dados relacionados a gestão, como: nível de formalidade do processo de ER, tempo gasto na ER, etc. O restante do questionário foi dividido seguindo o processo proposto no [SWEBOK Bourque, Fairley et al. \(2014\)](#), sendo: eleição de requisitos, análise de requisitos, especificação de requisitos, validação de requisitos e ferramentas para engenharia de requisitos.

A seguir apresentamos as questões que fazem parte da nossa pesquisa. O questionário completo encontra-se no [Apêndice A](#).

1. Qual é o nome do aplicativo?
2. Qual é o país onde a aplicação foi desenvolvida?
3. Em qual das categorias abaixo seu aplicativo pertence?
4. Na escala de 1 a 5, o quanto você considera que o processo de engenharia de requisitos é definido?  
\*Onde um (1) corresponde a Ad Hoc e cinco (5) corresponde a todos os passos mapeados e totalmente documentados.
5. Qual dessas atividades de engenharia de requisitos foram realizadas para este projeto?
6. Qual o percentual (%) do tempo de projeto foi gasto especificamente sobre a engenharia de requisitos?
7. Quantas pessoas estavam na equipe de desenvolvimento deste APP?
8. Sobre o papel do analista de requisitos?

\*Existe uma pessoa / grupo que trabalha especificamente com a engenharia de requisitos

\*Todos os membros da equipe de desenvolvimento trabalhando em requisitos de engenharia

9. Que técnicas de elicitação de requisitos foram usadas?
10. Quais das seguintes abordagens foram utilizadas na análise e modelagem de requisitos de software?
11. Requisitos não funcionais foram consideradas no desenvolvimento de aplicativos?
12. Se a sua resposta foi sim, quais dos seguintes requisitos não-funcionais foram considerados?
13. Em que tipo de notação foram especificados os requisitos ?
14. Você executou a inspeção de requisitos?
15. Se a sua resposta foi sim, quais das técnica foram usadas na inspeção de requisitos?
16. Você usou alguma ferramenta CASE na fase de Engenharia de Requisitos?
17. Se a sua resposta foi sim, quais das seguintes ferramentas foram utilizadas?

O questionário foi elaborado como base no [SWEBOK Bourque, Fairley et al. \(2014\)](#) e nas questões que foram aplicadas nas pesquisas dos trabalhos relacionados. A [Tabela 3](#) relaciona o número de cada questão de nossa pesquisa (colunas) com o autor do trabalho relacionado que foi usado como referência para sua elaboração (linhas). Algumas questões estavam presentes em mais de um trabalho, por isso existem colunas onde mais de uma linha é assinalada (X). As questões 1, 2 e 3 não estão incluídas na [Tabela 3](#), pois são questões com o objetivo de adquirir informações da app.

Tabela 3 – Fontes das questões

Fontes	4	5	6	7	8	9	10	11	12	13	14	15	16	17
( <a href="#">KHAN et al., 2015</a> )											X	X		
( <a href="#">KASSAB; NEILL; LAPLANTE, 2014</a> )			X			X	X			X			X	X
( <a href="#">REGNELL et al., 2012</a> )									X					
( <a href="#">TALBOT, 2011</a> )						X				X				
( <a href="#">OSSADA; MARTINS, 2010</a> )								X	X					
( <a href="#">ROUIBAH; AL-RAFEE, 2009</a> )						X	X						X	X

As questões 4, 5 e 7 não possuem nenhuma fonte marcada na tabela, pois são questões propostas por nosso trabalho. A questão 4 tem como objetivo apontar o nível de formalidade no processo de ER. A questão 5 tem a finalidade de identificar se algumas das atividades do processo de ER estão sendo negligenciadas. Já a questão 7 tem como propósito identificar como é tratado atualmente o papel do analista de requisitos.

Nem todos os trabalhos relacionados estão presentes na [Tabela 3](#), em razão de que sua contribuição para este trabalho se deu de outra forma, como já relatado no [Capítulo 3](#).

A [Tabela 4](#) apresenta o número de cada questão de nossa pesquisa e a forma como elas são apresentadas para o entrevistados. Dentre as 17 questões, somente a questão 1 é aberta, enquanto o restante das questões são fechadas, ou seja, o entrevistado escolhe uma resposta a partir de um número de opções apresentadas. Isto foi feito a fim de manter o tempo necessário para completar a pesquisa tão curto quanto possível, embora ainda proporcionando riqueza de dados suficiente para atingir o objetivo da pesquisa.

Tabela 4 – Formato das questões

Formato da Questão	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17
Aberta	x																
Múltipla escolha						x	x	x			x			x		x	
Caixa de seleção					x				x	x		x	x		x		x
Lista suspensa		x	x														
Grade de múltipla escolha				x													
Opção de outra									x	x		x	x		x		x

## 5.2 Teste piloto

Antes da execução do teste piloto, o questionário foi enviado a dois professores especialistas em Engenharia de Software da Universidade Federal do PAMPA ([UNIPAMPA](#)), com o objetivo de avaliar a coerências e clareza das questões. O *feedback* dos professores originou as questões utilizadas no teste piloto (apresentadas na [subseção 5.1.4](#)).

O teste piloto do questionário foi realizado com os desenvolvedores dos 100 aplicativos móveis mais pesquisados no Google Play. O convite para a participação da pesquisa foi enviado através do e-mail disponibilizado pelas empresas, juntamente com o link para o questionário online. O convite foi reforçado após 7 dias. No final de 21 dias, 6 empresas responderam ao questionário online, significando, portanto, 6% dos convites. A [Tabela 5](#) apresenta a relação das apps que responderam ao teste piloto, juntamente com a categoria a que ela pertence e seu país de origem.

A baixa taxa de retorno no teste piloto deve-se a alguns problemas que ocorreram no envio da pesquisa, são eles:

- E-mail inválido: 3% da amostra retornaram com o aviso de falha ao encontrar o

Tabela 5 – Relação dos apps que responderam ao teste piloto

App	Categoria	País
The Room	Entreterimento	Reino Unido
Wazzap Migrator	Comunicação	Itália
Sleep as Android	Estilo de Vida	República Checa
Cameringo	Fotografia	Espanha
DiskDigger	Ferramentas	Estados Unidos
Questões de Concurso Aprova	Educação	Brasil

destinatário;

- E-mail automático: 5% da amostra retornaram com e-mail automático informando que a única forma de entrar em contato era pelo site da empresa ou formulários;
- E-mail de suporte: 27% da amostra disponibilizou apenas um e-mail de suporte aos problemas de seus usuários, sendo assim, acreditamos ser pouco provável que nosso questionário chegue até o desenvolvedor.

O último problema tem relação com o fato da amostra para o teste piloto ser restrita aos 100 aplicativos mais pesquisados, o que envolve as aplicações como: whatsApp, Messenger, Facebook, que são desenvolvidos por empresas de grande porte.

Apesar da baixa taxa de respostas obtida, a realização do teste piloto possibilitou algumas análises. Uma delas foi a inconsistência das respostas individuais de alguns participantes. Na questão 5, quando foram questionados sobre quais das etapas do processo de ER realizavam, informaram que não realizavam uma determinada atividade, como por exemplo, especificação de requisitos. No entanto, na sessão que abordava essa determinada atividade, afirmaram fazer uso de técnicas ou ferramentas.

Com base nesse fato, duas hipóteses foram levantadas, a fim de entender o porquê isso ocorreu repetidamente mesmo numa amostra pequena. A primeira hipótese é sobre a clareza da pergunta e a segunda é sobre o conhecimento dos participantes sobre ER. A partir disso se decidiu colocar uma descrição na questão 5, explicando brevemente cada atividade com base no [SWEBOK \(BOURQUE; FAIRLEY et al., 2014\)](#).

A partir dos resultados de algumas das questões do teste piloto foi levantada a hipótese das equipes de desenvolvimento de apps serem muito pequenas ou até mesmo de não ser uma equipe e sim um único desenvolvedor. Essa hipótese é justificada pelo fato de existir um grande número de cursos de programação Android disponível e a facilidade que o Google Play oferece para se cadastrar um app. Essa hipótese resultou na inclusão de uma pergunta no questionário com a finalidade de obter o tamanho da equipe de desenvolvimento.

## 5.3 Execução da pesquisa

O contato com as empresas foi realizado pelo e-mail que as mesmas disponibilizam no Google Play. Os e-mails foram extraídos automaticamente com uso da versão gratuita do FOX E-mail Extrator. Para realizar a extração com o FOX E-mail Extrator, é necessário fornecer o *link* do site de onde se deseja extrair a quantidade de e-mails, juntamente com, no mínimo, uma palavra-chave de busca. Foram realizadas diversas extrações, onde as categorias do Google Play foram usadas como palavras-chave. Foram extraídos 200 e-mails para cada uma das 30 categorias, somando ao final da atividade de extração a quantidade de seis mil e-mails.

Os e-mails foram enviados de forma automática com uso do versão gratuita do SendBlaster, um software de envio de e-mail em massa. Os e-mails enviados às empresas continham o convite para a participação da pesquisa, juntamente com o link para o questionário online.

No decorrer do envio dos e-mails, surgiram alguns problemas relacionados as políticas de privacidade dos servidores Protocolo de Transferência de Correio Simples (do inglês *Simple Mail Transfer Protocol*) (SMTP). Constantemente o envio em massa de e-mails da nossa pesquisa era confundido como *Spam* pelos servidores SMTP, que bloqueavam o envio do corpo do e-mail para o destinatário. Esse problema foi contornado fazendo um rodízio de vários endereços de e-mails remetentes diferentes, porém esse problema acabou causando um atraso significativo na fase de coleta de dados.

A coleta de dados foi realizada no período de 01 de agosto a 31 de outubro de 2016. O Google Form, ferramenta utilizada no desenvolvimento de nosso questionário online, realiza a coleta de dados de forma automática mantendo os dados em seu servidor, com possibilidade de exportação.

## 5.4 Conclusão da pesquisa

### 5.4.1 Análise dos dados

O Excel foi a ferramenta utilizada para realizar a análise dos dados e gerar gráficos. O resultado de cada questão foi ilustrado por meio de um gráfico, sendo gráfico de pizza para as questões que permitem assinalar apenas uma alternativa e gráfico de barras para as questões que permitem assinalar mais de uma alternativa. Adicionalmente, gráficos de dispersão são utilizados para demonstrar a relação entre duas questões.

A fórmula de coeficiente de correlação linear foi utilizada para comprovar a possível

relação entre duas variáveis da pesquisa. A fórmula do coeficiente linear é dada por:

$$r = \frac{n \sum(x_i \cdot y_i) - (\sum x_i)(\sum y_i)}{\sqrt{n \sum x_i^2 - (\sum x_i)^2} \cdot \sqrt{n \sum y_i^2 - (\sum y_i)^2}} \quad (5.1)$$

O coeficiente de correlação varia de -1 a 1. O sinal indica direção positiva ou negativa do relacionamento e o valor sugere a força da relação entre as variáveis. Uma correlação perfeita (-1 ou 1) indica que o escore de uma variável pode ser determinado exatamente ao se saber o escore da outra. No outro oposto, uma correlação de valor zero indica que não há relação linear entre as variáveis (FILHO; JUNIOR, 2010).

Valores extremos (0 ou 1) dificilmente são encontrados na prática, assim os resultados são interpretados com o auxílio de uma classificação. Este trabalho adota a classificação apresentada na Tabela 6.

Tabela 6 – Classificação de correlação

<b>Coeficiente de correlação</b>	<b>Correlação</b>
r = 1	Perfeita Positiva
0,8 < r < 1	Forte Positiva
0,5 < r < 0,8	Moderada Positiva
0,1 < r < 0,5	Fraca Positiva
0,0 < r < 0,1	Ínfima Positiva
r = 0	Nula
0,0 < r < -0,1	Ínfima Negativa
- 0,1 < r < - 0,5	Fraca Negativa
- 0,5 < r < - 0,8	Moderada Negativa
- 0,8 < r < - 1	Forte Negative
r = -1	Perfeita Negativa

## 6 Resultados

Neste Capítulo são apresentados os resultados da pesquisa. Essa apresentação é dividida em sessões seguindo a mesma organização do questionário. A exceção da [seção 6.8](#) que mostra a relação entre duas questões da pesquisa.

A pesquisa obteve 85 respostas, portanto, conforme [Rumsey \(2009\)](#), o tamanho da amostra é considerado satisfatório. Estes 85 participantes responderam as oito questões das seções 1 (identificação) e 2 (gestão e processo) do questionário. As demais questões não eram obrigatórias, assim nem todos os participantes as responderam. Adicionalmente, existem questões que permitem a escolha de mais de uma alternativa e, por consequência, a soma das respostas em cada alternativa não é equivalente ao número de respondentes à questão.

### 6.1 Identificação

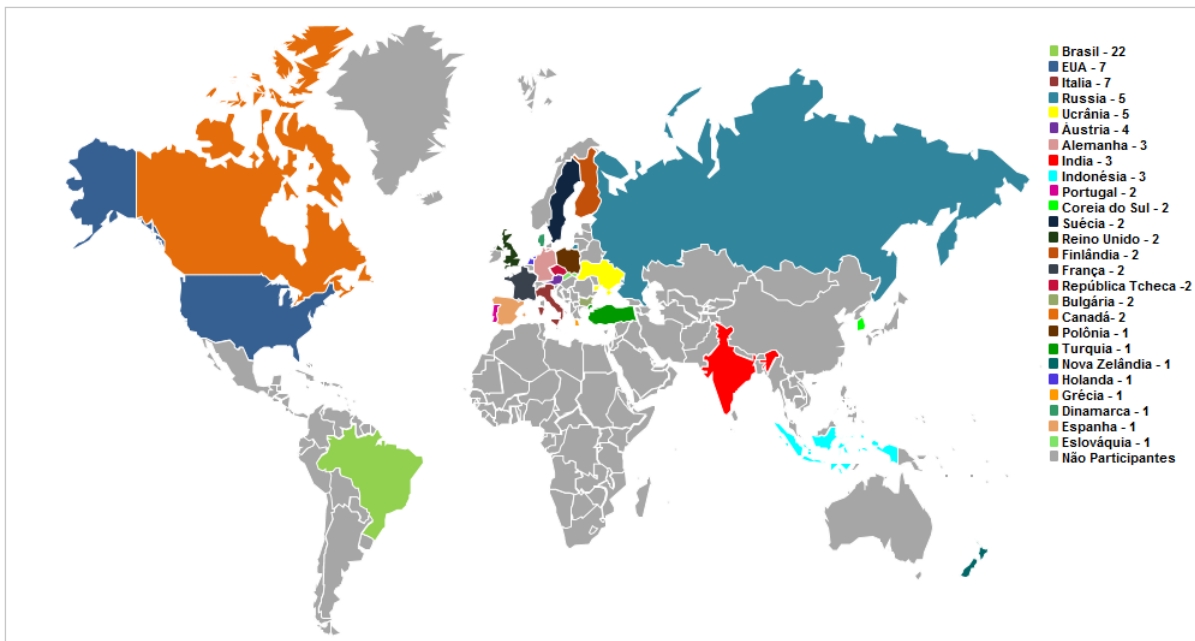
Os apps participantes da pesquisa foram classificados quanto ao país de origem e sua categoria. A [Figura 4](#) contém um mapa onde são indicados os países participantes da pesquisa, assim como o número de respostas de cada país. Conforme observado, desenvolvedores de 26 países diferentes responderam o questionário, demonstrando a abrangência da pesquisa. O Brasil foi o país com maior participação com 22 (25%) respostas, seguido pela Itália e os Estados Unidos, ambos com 7 (8%) respostas.

A [Figura 5](#) ilustra o gráfico de categorias de apps que participaram da pesquisa. A pesquisa obteve respostas de 20 das 30 categorias de apps disponíveis no Google Play, o que mostra a diversidade de apps que participaram da pesquisa. Jogos foi a categoria com maior representatividade com 12 (14%) respostas, seguida pela categoria Ferramentas com 9 (11%) respostas.

### 6.2 Gestão e processo

Com objetivo de identificar o nível de formalidade de processo de ER, os participantes responderam a questão em uma escala de 1 a 5, onde 1 corresponde a Ad Hoc e 5 corresponde a todas as etapas mapeadas e documentadas. De acordo com o gráfico mostrado na [Figura 6](#), apenas 7% dos respondentes afirma possuir nível de formalidade 5, ou seja, todos os processos mapeados e documentados. Em contrapartida, apenas 13% afirma realizar um processo Ad Hoc. Os 80% restantes indicaram valores 2 a 4 para nível de formalidade. Na [seção 6.8](#), a informação sobre o nível de formalidade adotado é usada

Figura 4 – Países participantes da pesquisa



para estabelecer uma relação com a avaliação dos usuários e, também, com o número de desenvolvedores da app.

A questão 5 tem como objetivo identificar as atividades de ER que são realizadas pelos desenvolvedores de apps. No gráfico apresentado na [Figura 7](#), podemos observar que a análise de requisitos é a atividade com maior percentual (74%), seguido pela especificação de requisitos (61%), elicitação de requisitos (38%) e, por último, validação (36%). Seguindo o processo proposto pelo SWEBOK, este resultado não possui coerência, pois não é possível analisar requisitos sem tê-los elicitados. Uma hipótese é que os participantes possuem pouco conhecimento teórico sobre a ER. Essa inconsistência já havia sido percebida durante a execução do teste piloto, o que conduziu a adição de uma descrição de cada atividade com o objetivo de alinhar conceitos.

A [Figura 8](#) apresenta os resultados para a questão 6, a qual aborda a quantidade do tempo total do projeto que foi dedicada à realização da ER. É possível observar que 4% não sabem especificar quanto foi gasto em atividades de ER; 33% dos respondentes afirmam gastar menos de 15% na ER; 29% dizem dedicar entre 15% e 30% para realização da ER; 19% destinam entre 30% a 45% do tempo para ER; e os demais (15%) gastam mais do que 45% do tempo em ER.

Segundo [Berkun \(2005\)](#) o tempo adequado para a realização das atividades de



Figura 5 – Categoria dos apps participantes da pesquisa

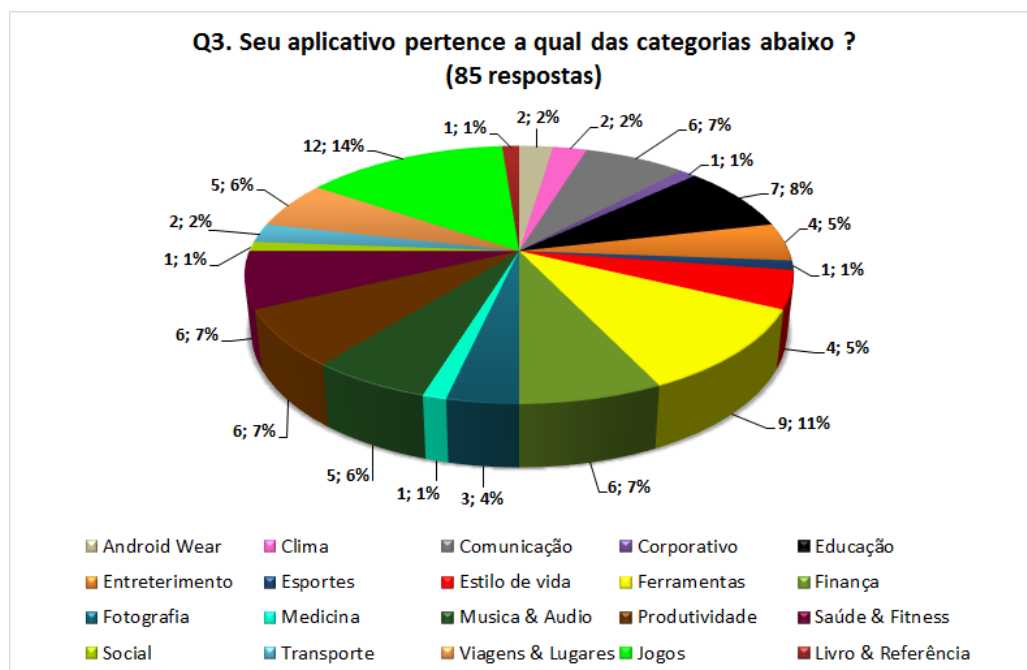
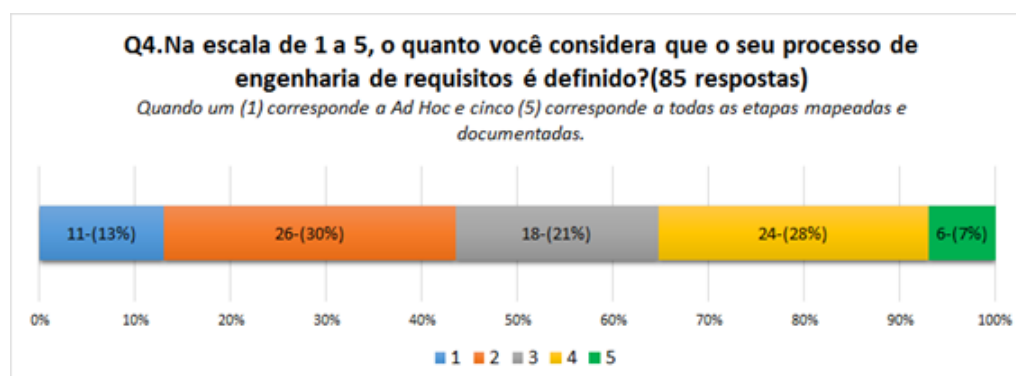


Figura 6 – Nível de formalidade



ER é de cerca de 33,3%. Assim é possível afirmar que a ER é um tanto desvalorizada em questão do tempo dedicado para sua realização, pois chegamos a 63,4% da amostra quando somando o percentual dos respondentes que afirmam dedicar até 30% do tempo com a ER.

Na questão 7 busca-se identificar o tamanho da equipe de desenvolvimento dos apps. Observando o gráfico da Figura 9, nota-se que a grande maioria dos apps (42%) é desenvolvido por apenas uma pessoa, 18% são desenvolvidos por duas pessoas, 16% por

Figura 7 – Atividades de ER realizadas

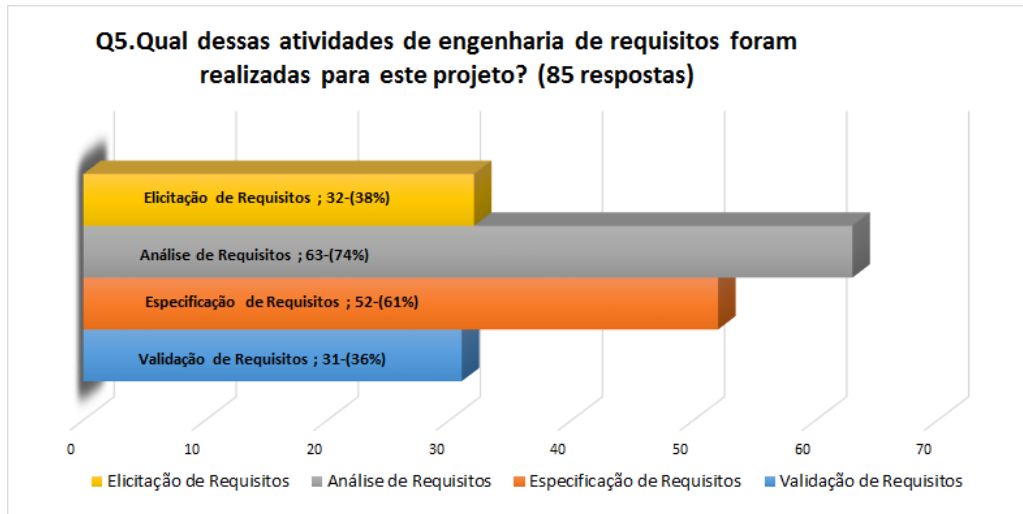
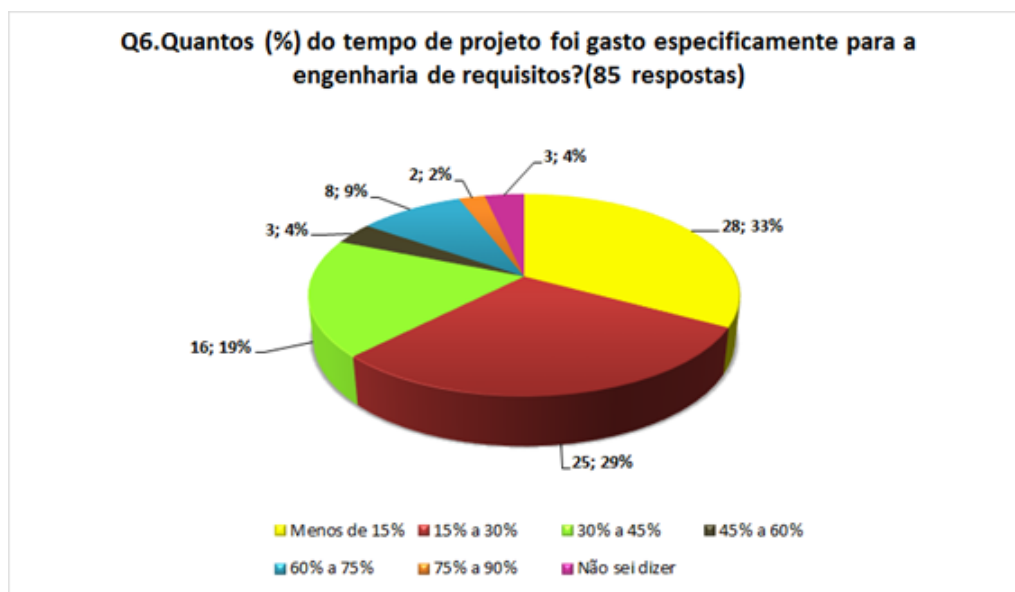


Figura 8 – Tempo dedicado à ER

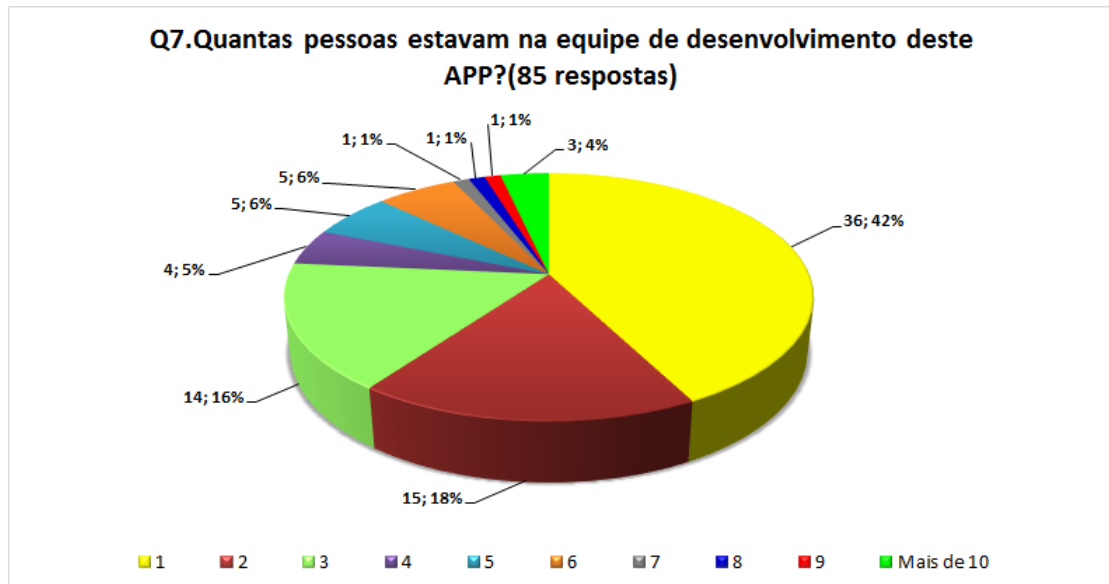


equipes compostas por 3 pessoas e apenas 4% são desenvolvidas por equipes compostas por mais de 10 pessoas.

Esses resultados indicam que a maioria das equipes de desenvolvimento de apps são menores do que indicado na literatura. Por exemplo, a metodologia Scrum aponta que ter menos de três membros na equipe de desenvolvimento reduz a interação e resulta em

menores ganhos de produtividade. Uma hipótese para o tamanho reduzido das equipes é que a maioria dos respondentes desta pesquisa sejam desenvolvedores independentes que buscam no Google Play uma forma de disponibilizar suas apps.

Figura 9 – Tamanho da equipe de desenvolvimento

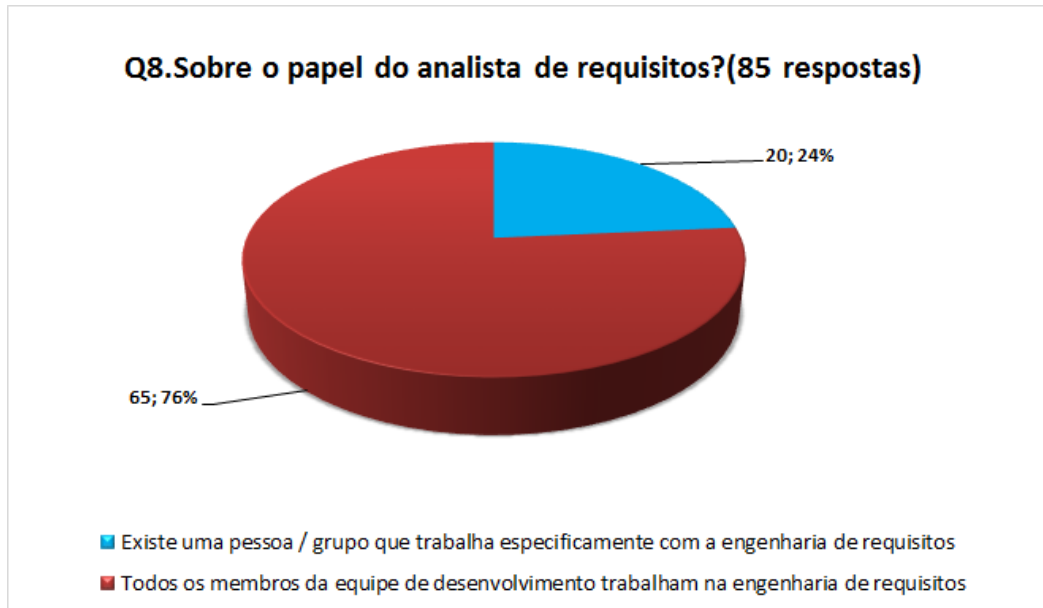


Na questão 8 busca-se saber quem realiza a ER no desenvolvimento de apps. Analisando os resultados apresentados no gráfico da Figura 10, observa-se que a grande maioria dos respondentes (76%) afirmam que toda a equipe é responsável pela ER, enquanto 24% afirmam que uma pessoa ou equipe exclusiva é responsável pelas atividades de ER. Tendo os resultados da questão anterior apontado que a grande maioria dos apps são desenvolvidos por equipes muito pequenas, é natural que nestes casos toda a equipe realize todas as atividades de desenvolvimento do app, inclusive a ER. Isso também pode indicar que a grande maioria dos respondentes fazem uso das metodologias ágeis.

### 6.3 Elicitação de requisitos

A questão 9 tem como objetivo identificar quais são as técnicas de elicitação de requisitos mais usadas. Os resultados, ilustrados na Figura 11, indicam que o *Brainstorming* (77%) é a técnica mais popular, seguida por prototipação (51%) e observação (47%). Comparando esses resultados com os obtidos pelo artigos da revisão bibliográfica, é possível observar que o *Brainstorming* se manteve como uma das técnicas mais usadas. Entretanto, as técnicas entrevista e questionário, citadas como as mais usadas pelos trabalhos relacionados, não são tão populares no desenvolvimento de apps.

Figura 10 – O papel do analista de requisitos



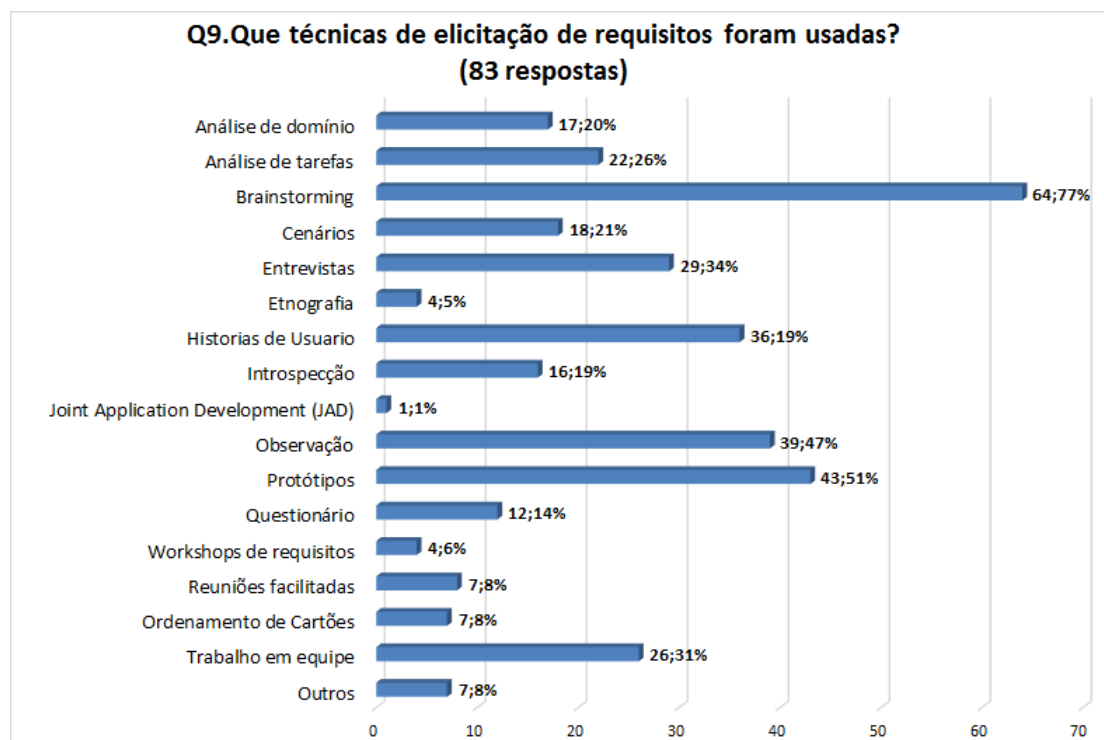
## 6.4 Análise de requisitos

Na [Figura 12](#) são apresentados os resultados para a questão 10, cujo objetivo é identificar quais as abordagens são utilizadas na modelagem de requisitos de software. Os resultados demonstram que a maioria dos respondentes (57%) não faz uso de nenhuma abordagem para a modelagem de requisitos de software. Quando uma abordagem é usada, a análise orientada a objetos é a mais citada (23%), seguida pela análise estruturada (19%) e pela modelagem de fluxo de dados (17%). O uso de outros tipos de abordagem são citados por 2,5% dos respondentes.

Como na pesquisa de [Kassab, Neill e Laplante \(2014\)](#), a maioria das respondentes não usa nenhuma abordagem para a modelagem de requisitos de software. Segundo [Kassab, Neill e Laplante \(2014\)](#) esse fato pode ser explicado pelo aumento do uso dos métodos ágeis, onde é enfatizado que devesse gastar menos tempo com documentação e mais com desenvolvimento, por isso muitos dos desenvolvedores acabam por não adotar nenhuma abordagem ou técnica para a modelagem de requisitos.

Nas questões 11 e 12 busca-se descobrir se os requisitos não funcionais são considerados nos desenvolvimentos dos apps e quais estão sendo considerados. Na [Figura 13](#) é ilustrado o gráfico com resultados da questão 11, onde 87% dos respondentes afirmam considerar os requisitos não funcionais no desenvolvimento de seus apps, enquanto apenas 13% dizem ignorá-los. Esse cenário é completamente diferente ao relatado nos trabalhos relacionados, onde os requisitos não funcionais não são valorizados. Isso pode ser expli-

Figura 11 – Técnicas de elicitación de requisitos



cado pelo fato de requisitos não funcionais possuírem maior importância em projetos de apps do que em outros tipos de software (CORRAL; SILLITTI; SUCCI, 2015).

Conforme pode ser observado na Figura 14, usabilidade (88%) e performance (84%) são os requisitos não funcionais que mais preocupam os desenvolvedores de apps, seguidos de consumo de memória (66%) e confiabilidade (56%).

## 6.5 Especificação de requisitos

A questão 13 visa identificar qual o tipo de notação mais usada para a especificação de requisitos. Os resultados apresentados na Figura 15 indicam que a linguagem natural (72%) é a mais usada, seguida pelas notações gráficas (41%), a especificação semiformal (10%), a especificação formal (10%) e, por último, outros tipos de especificações (9%).

Esses resultados estão de acordo com o relatado nos trabalhos relacionados, ou seja, a maioria dos desenvolvedores usa linguagem natural e a linguagem formal é pouco utilizada. Isso porque apesar da linguagem formal reduzir a quantidade de ambiguidades nos requisitos, o uso desta linguagem é muito custosa e a maioria dos clientes não a compreende. Fatores esses que dificultam a adoção de uma linguagem formal.

Figura 12 – Abordagens utilizadas na modelagem de requisitos

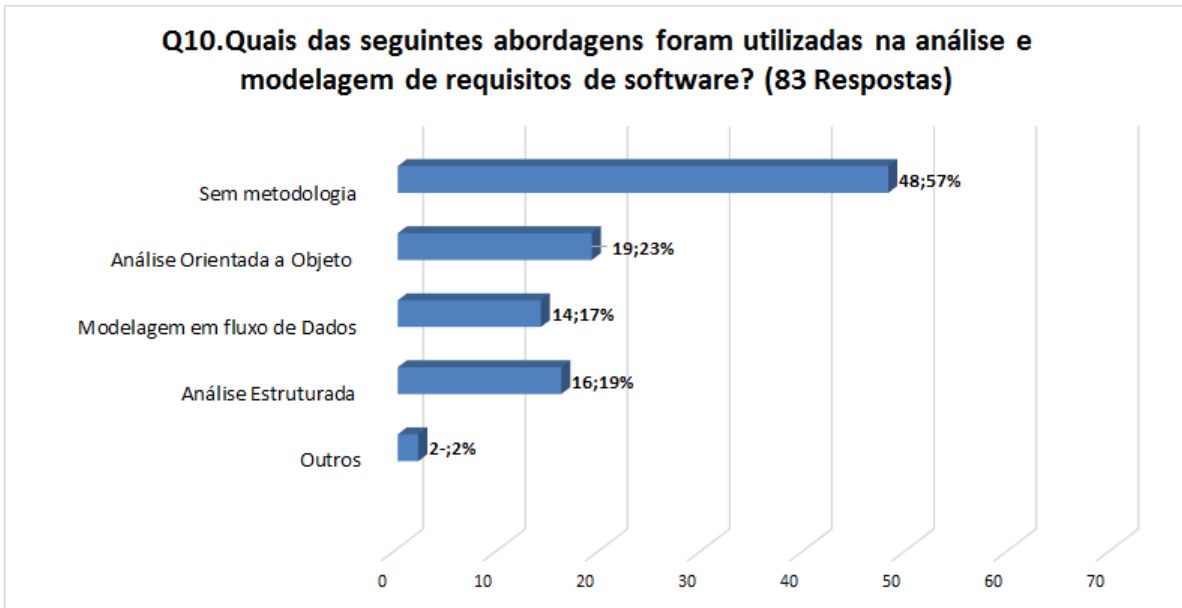
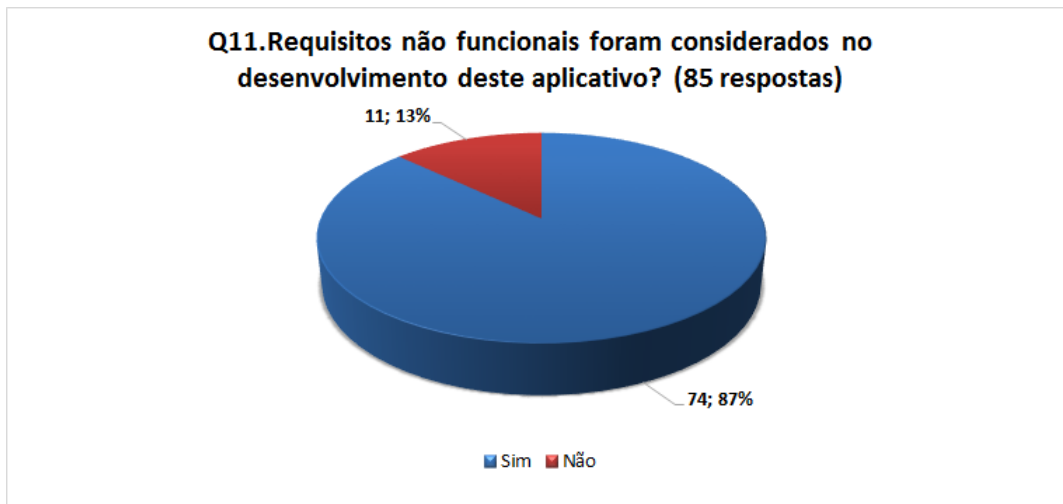


Figura 13 – Percentual de apps que consideram requisitos não funcionais



## 6.6 Verificação de requisitos

As questões 14 e 15 identificam o estado da prática da inspeção de requisitos. A Figura 16 apresenta o gráfico mostrando o percentual de participantes que realizam inspeção de requisitos. Como já indicado na questão 5 (Figura 7), onde apenas 36% realizava a atividades de validação, a inspeção de requisitos é negligenciada pela maioria

Figura 14 – Requisitos não funcionais considerados

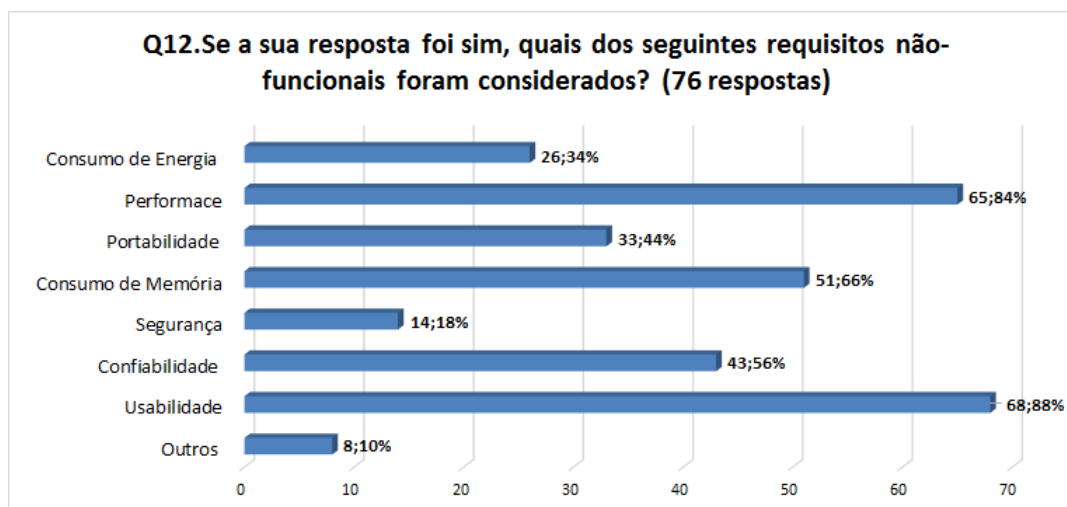
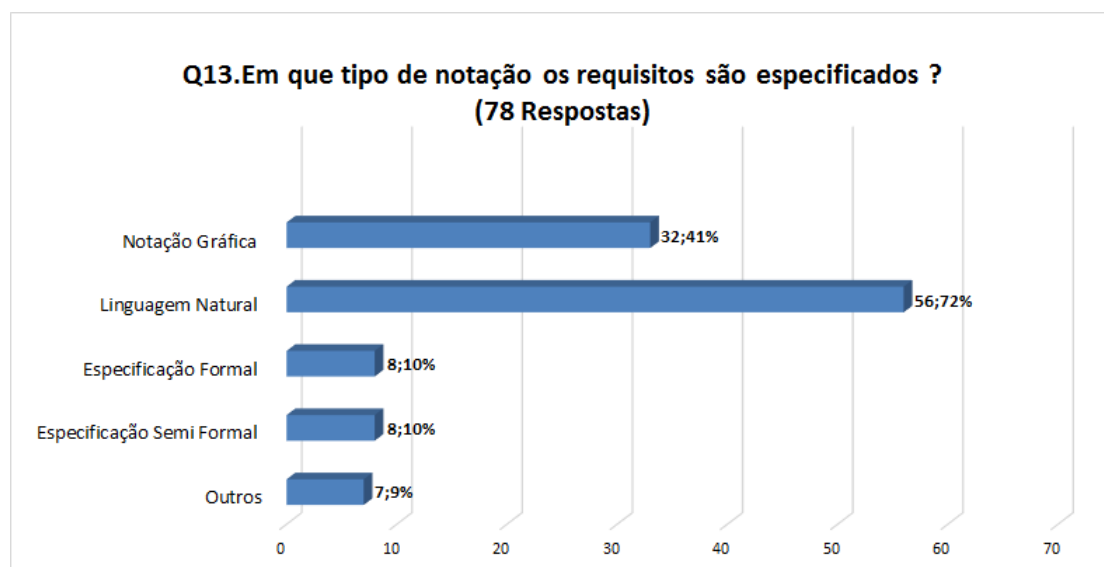


Figura 15 – Notação utilizada para especificar os requisitos



dos participantes (53%).

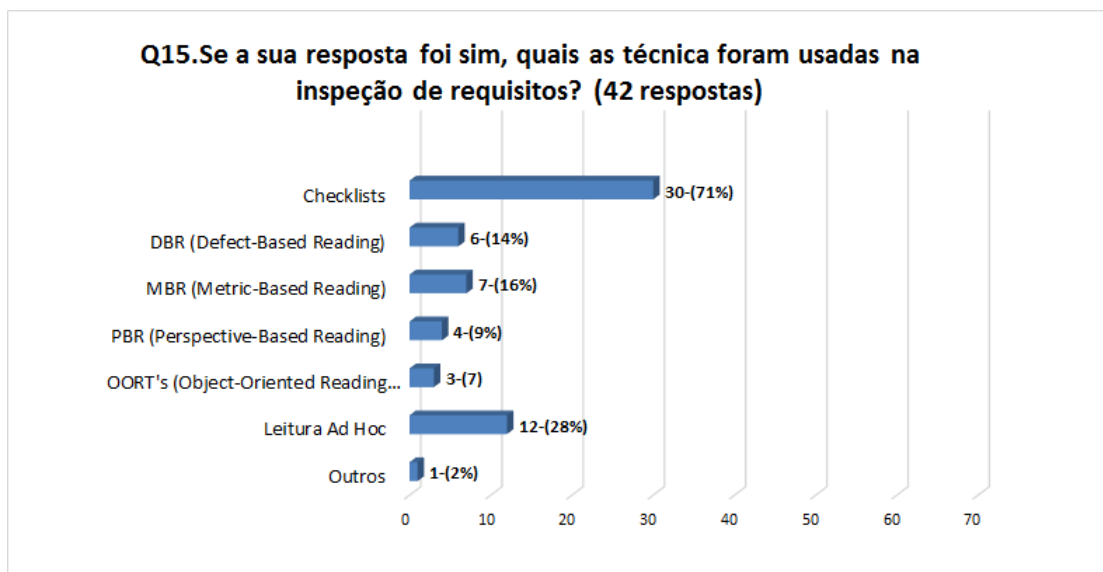
Os resultados desta pesquisa se assemelham muito com os encontrados na revisão bibliográfica, onde a pesquisa de [Kassab, Neill e Laplante \(2014\)](#) aponta que mais de 50% dos entrevistados de sua pesquisa não realiza nenhuma atividade de validação ou verificação de requisitos. Esse é um fato preocupante pois a qualidade dos requisitos tem influência direta sobre a qualidade do produto final.

Figura 16 – Percentual de apps que realizam inspeção de requisitos



A questão 15 buscou identificar quais são as técnicas de inspeção mais utilizadas. Conforme observado na [Figura 17](#), o *checklist* com 71% é a técnica mais utilizada seguida pela leitura Ad hoc com 28%.

Figura 17 – Técnica de inspeção de requisitos utilizadas



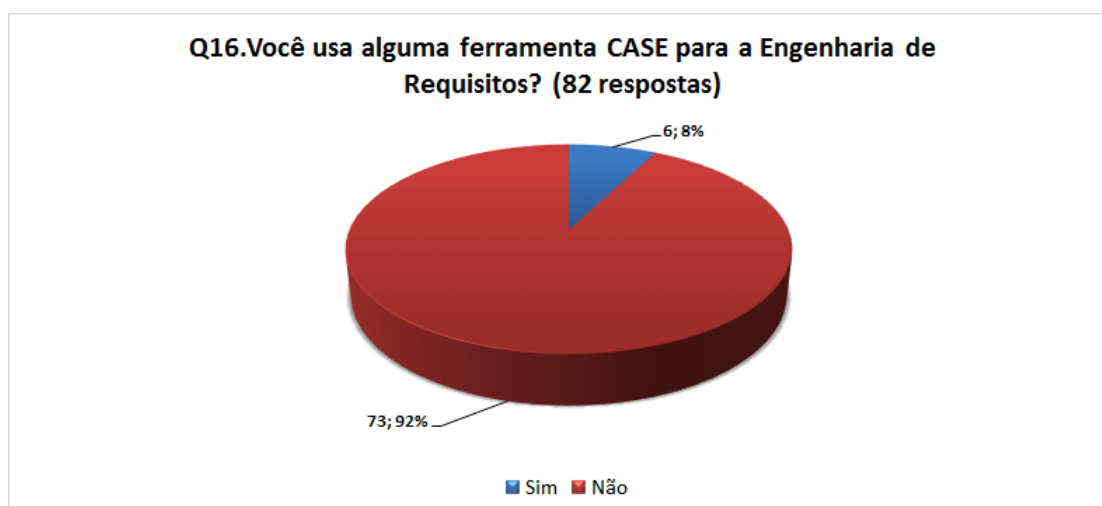


## 6.7 Ferramentas CASE

As questões 16 e 17 tem como objetivo identificar se são usadas ferramentas Case para auxiliar no processo de ER e também quais são utilizadas. Na [Figura 18](#) é apresentado o percentual de participantes que utilizam ferramentas CASE. Os resultados apontam que apenas 8% estão fazendo uso destas ferramentas.

Apesar da haver uma diversidade de ferramentas que facilitam e auxiliam inúmeras atividades da ER, não há uma grande utilização destas no meio dos desenvolvedores de apps. Uma das hipóteses para isso é as ferramentas CASE não suportarem as necessidades específicas dos desenvolvedores de apps. Outra hipótese é dos desenvolvedores desconhecerem as ferramentas CASE adequadas.

Figura 18 – Percentual de apps que utilizam ferramentas CASE

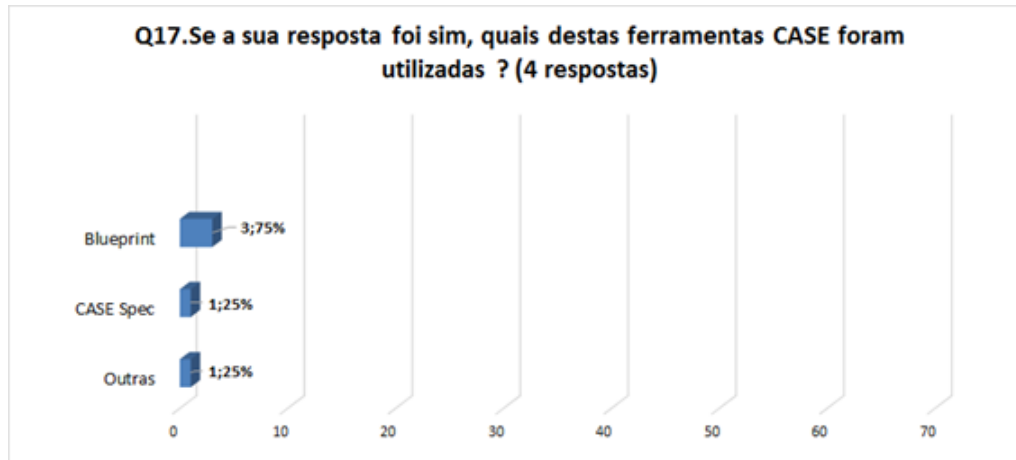


A [Figura 19](#) apresenta as ferramentas CASE utilizadas pelos quatro participantes que responderam a questão 17. Conforme pode ser observado, três participantes indicaram fazer uso da ferramenta Blueprint, um participante utiliza a ferramenta CASE Spec e um participante faz uso de outra ferramenta não listada na pesquisa (mencionada como ferramenta de desenvolvimento próprio).

## 6.8 Análise do coeficiente de correlação

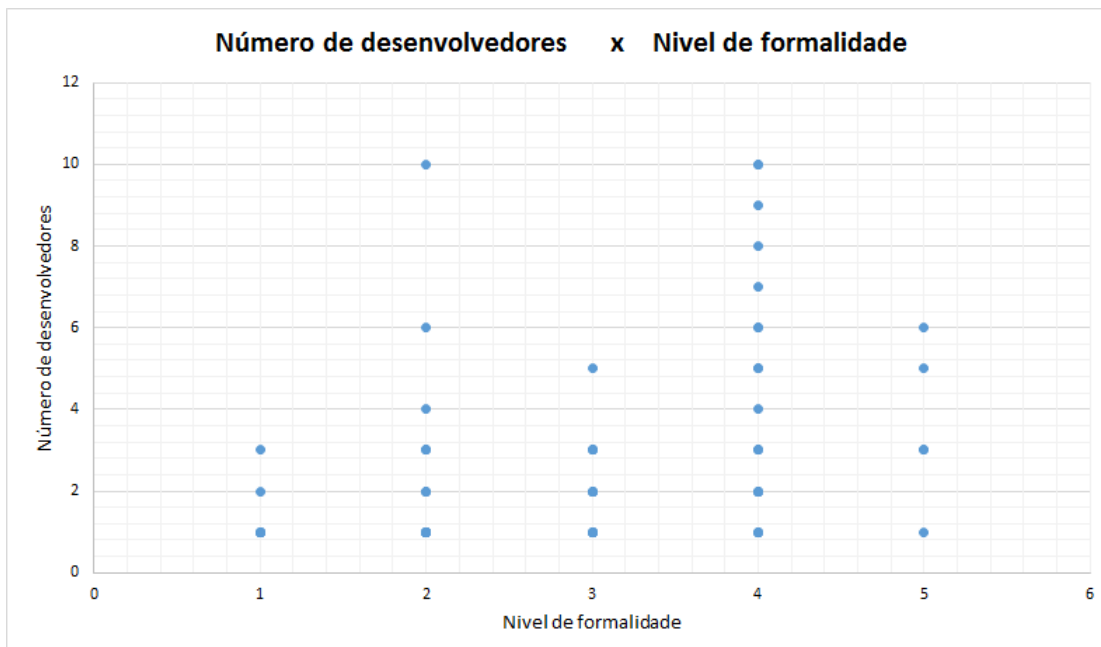
Na [Figura 20](#) é exibido um diagrama de dispersão a fim de buscar identificar a existência de um relação entre o número de desenvolvedores e o grau de formalidade de seus processos de ER. O cálculo do coeficiente de correlação linear entre essas duas variáveis é 0,38, indicando que há uma relação de coeficiente fraca positiva. Desta forma,

Figura 19 – Ferramentas CASE utilizadas



é possível concluir que o número de desenvolvedores não tem grande influência no nível de formalismo.

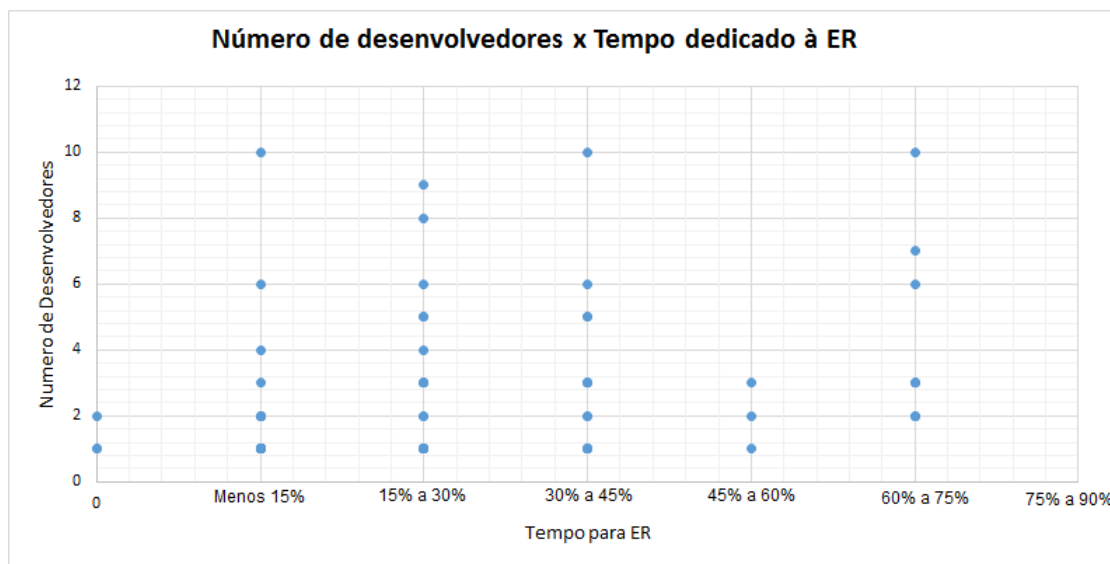
Figura 20 – Número de desenvolvedores x Nível de formalidade



O número de desenvolvedores também foi relacionado com o tempo dedicado para a ER. A [Figura 21](#) contém o gráfico de dispersão gerado por essas duas variáveis. O resultado do cálculo da relação de coeficiente foi 0,28 sendo classificada também como

uma relação de coeficiente fraca positiva. Isso significa que o número de desenvolvedores não interfere sobre o tempo dedicado a ER.

Figura 21 – Número de desenvolvedores x Tempo dedicado à ER



Supondo que quanto maior o nível de formalidade dos processos de ER maior seria a qualidade do aplicativo, um gráfico de dispersão foi gerado com essas duas variáveis para comprovar ou refutar essa suposição. A média das avaliações dos usuários, extraídas do Google Play, foi utilizada como parâmetro de qualidade do app. Conforme pode ser observado no gráfico da [Figura 22](#), existe uma relação ínfima positiva entre o nível de formalidade e a avaliação dos usuários. O cálculo da relação de coeficiente resultou em 0,09.

A avaliação do usuário também foi relacionada ao tempo dedicado para ER. Esperava-se que o tempo dedicado a ER impactasse na qualidade do app e, por consequência, na avaliação dos usuários. No entanto, como pode ser observado na [Figura 23](#), mais uma vez houve uma relação fraca positiva entre o tempo dedicado à ER e a avaliação do usuário. O cálculo do coeficiente de relação resultou em 0,15.

Figura 22 – Nível de formalidade x Avaliação dos usuários

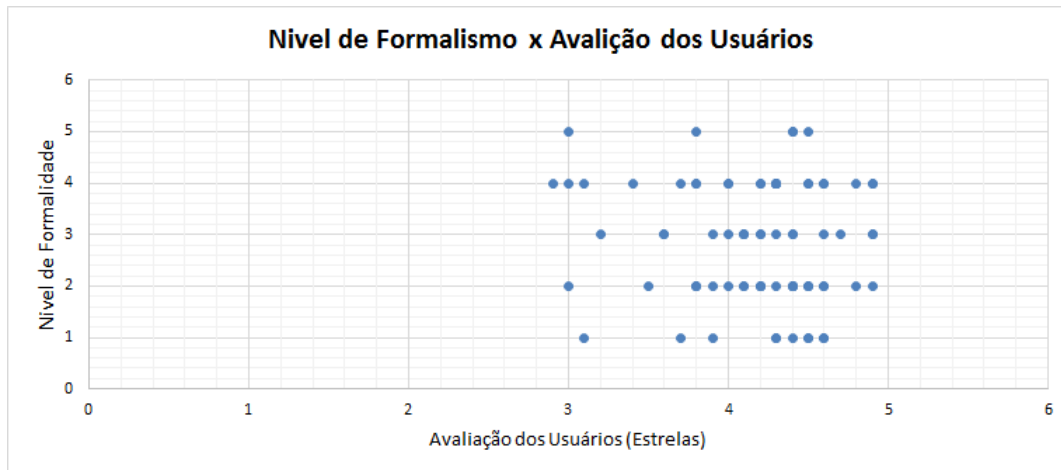
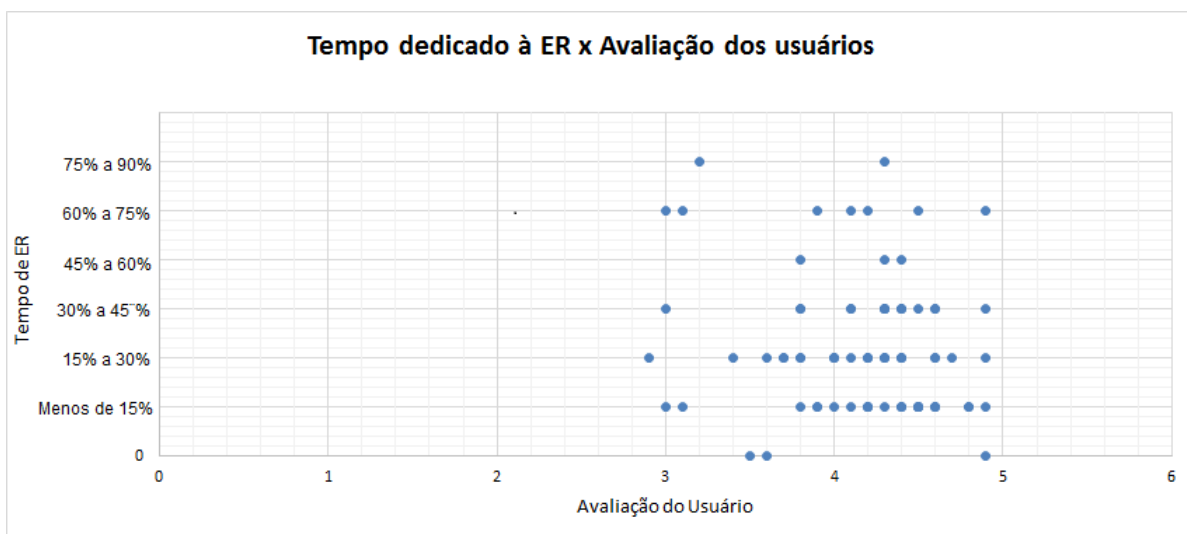


Figura 23 – Tempo dedicado à ER x Avaliação dos usuários



## 7 Conclusões

O objetivo geral do presente trabalho era identificar o estado da prática da ER utilizada no mercado para o desenvolvimento de aplicações móveis. Para atingir tal objetivo, uma pesquisa foi elaborada para ser aplicada com desenvolvedores de aplicações móveis disponibilizados no Google Play.

Alguns desafios foram encontrados durante a condução desta pesquisa, entre eles estão:

- Contatar os desenvolvedores, porque o e-mail de contato cadastrado no Google Play era inválido ou de suporte com retorno automático;
- Envio de e-mails em massa, onde o envio da pesquisa era confundido com *spam* pelo servidor SMTP;
- Baixa taxa de retorno, necessitando o envio de aproximadamente seis mil e-mails para obter 85 respostas.

Contudo, ao final da pesquisa, um número satisfatório de respostas foi obtido. A partir da análise dos dados coletados foi possível fazer algumas descobertas interessantes:

- A maioria da amostra dedica menos tempo do que recomendando pela literatura para a realização das atividades de ER;
- Pouquíssimos apps são desenvolvidas com o nível de formalidade 5, ou seja, com todas as etapas documentadas e mapeadas;
- A maioria dos apps são desenvolvidos por equipes pequenas. Em 45% dos casos é desenvolvida por apenas uma pessoa;
- Requisitos não funcionais são uma preocupação da maioria dos desenvolvedores de apps, sendo a usabilidade o requisitos não funcional mais prestigiado;
- A maioria dos respondentes (58%) não faz uso de nenhuma abordagem para a modelagem de requisitos de software;
- Mais da metade dos respondentes da amostra afirmam não realizar nenhum tipo de inspeção de requisitos (53%), e quando realizam o Checklist é a técnica mais usada;
- Ferramentas CASE são utilizadas por pouquíssimos desenvolvedores.

Ao final da análise dos resultados podemos concluir que o mercado deixa a desejar em alguns quesitos como, por exemplo: tempo dedicado para as atividades de ER; não adoção de uma abordagem para modelagem de requisitos; e não realização de inspeção de requisitos. Portanto, o processo de ER é negligenciado por grande parte dos desenvolvedores de aplicações móveis. Esse fato é preocupante já que a ER está ligada diretamente com a qualidade do software. Desta forma, acreditamos que a academia deveria dar maior ênfase ao estudo da ER, destacando sua importância para o desenvolvimento de software com qualidade.

Cita-se como possíveis trabalhos futuros, a comprovação ou refutação das hipóteses levantadas ao longo deste trabalho e que não foram contempladas nessa pesquisa, como:

- Os desenvolvedores de apps possuem pouco conhecimento formal da ER;
- A maioria das apps são criadas por desenvolvedores independentes, ou seja, sem vínculo com uma empresa formal;
- Os desenvolvedores de apps utilizam metodologias ágeis?
- As ferramentas CASE disponíveis no mercado não suportam as necessidades dos desenvolvedores de apps.

Adicionalmente, é possível expandir o escopo da pesquisa e buscar identificar a situação da engenharia de software como um todo no desenvolvimentos de apps.

# Referências

- ÁGIL, M. Manifesto para o desenvolvimento ágil de software. *Disponível em: <http://manifestoagil.com.br/>. Acessado em 15/09/2016*, v. 17, 2011. Citado na página 40.
- ANALYSIS, I. I. of B. *A Guide to the Business Analysis Body of Knowledge (BABOK Guide), Version 2.0*. [S.l.]: International Institute of Business Analysis, 2009. Citado 2 vezes nas páginas 33 e 35.
- BERKUN, S. *The art of project management*. [S.l.]: O'reilly, 2005. Citado 2 vezes nas páginas 40 e 54.
- BERNÁRDEZ, B.; DURÁN, A.; GENERO, M. Empirical evaluation and review of a metrics-based approach for use case verification. *Journal of Research and Practice in Information Technology*, Sydney, Australia: Australian Computer Society, c2000-, v. 36, n. 4, p. 247–258, 2004. Citado na página 36.
- BIOLCHINI, J. et al. Systematic review in software engineering. *System Engineering and Computer Science Department COPPE/UFRJ, Technical Report ES*, v. 679, n. 05, p. 45, 2005. Citado na página 37.
- BOURQUE, P.; FAIRLEY, R. E. et al. *Guide to the software engineering body of knowledge (SWEBOOK (R)): Version 3.0*. [S.l.]: IEEE Computer Society Press, 2014. Citado 8 vezes nas páginas 31, 32, 33, 34, 35, 47, 48 e 50.
- CISCO. *Global Mobile Data Traffic Forecast Update, 2015–2020 White Paper*. 2016. Disponível em: <<http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html>>. Citado na página 25.
- CONFORTO, E. C.; AMARAL, D. C.; SILVA, S. D. Roteiro para revisão bibliográfica sistemática: aplicação no desenvolvimento de produtos e gerenciamento de projetos. In: *8º Congresso Brasileiro de Gestão de Desenvolvimento de Produto. Anais do 8º CBGDP*. [S.l.: s.n.], 2011. Citado na página 37.
- CORRAL, L. *A Software Assurance Model for Mobile Applications*. Tese (Doutorado), 2014. Citado na página 25.
- CORRAL, L.; SILLITTI, A.; SUCCI, G. Software assurance practices for mobile applications. *Computing*, Springer, v. 97, n. 10, p. 1001–1022, 2015. Citado 4 vezes nas páginas 25, 26, 30 e 59.
- DIAS, K. B.; ARAÚJO, M. A. P. Boas práticas na engenharia de requisitos. *Engenharia de Software Magazine*, v. 27, n. 1, p. 14–20, 2008. Citado na página 40.
- FILHO, D. B. F.; JUNIOR, J. A. S. Desvendando os mistérios do coeficiente de correlação de pearson (r). *Revista Política Hoje*, v. 18, n. 1, 2010. Citado na página 52.
- FREITAS, H. et al. O método de pesquisa survey. *Revista de administração*, v. 35, n. 3, p. 105–112, 2000. Citado na página 36.

- KASSAB, M.; NEILL, C.; LAPLANTE, P. State of practice in requirements engineering: contemporary data. *Innovations in Systems and Software Engineering*, Springer, v. 10, n. 4, p. 235–241, 2014. Citado 7 vezes nas páginas 39, 40, 41, 42, 48, 58 e 61.
- KASUNIC, M. *Designing an effective survey*. [S.l.], 2005. Citado na página 43.
- KHAN, H. U. et al. An empirical study of software requirements verification and validation techniques along their mitigation strategies. *Asian Journal of Computer and Information Systems (ISSN: 2321–5658)*, v. 3, n. 03, 2015. Citado 4 vezes nas páginas 39, 40, 41 e 48.
- KLAUS, R. C. P. *Fundamentos da Engenharia de Requisitos*. [S.l.]: Rockynooock, 2012. Citado 4 vezes nas páginas 32, 33, 35 e 41.
- KNUTH, D. E. *The T<sub>E</sub>X Book*. 15th. ed. [S.l.]: Addison-Wesley, 1984. Citado na página 41.
- NAGAPPAN, M.; SHIHAB, E. Future trends in software engineering research for mobile apps. In: *23rd IEEE International Conference on Software Analysis, Evolution, and Reengineering SANER'16*. [S.l.: s.n.], 2016. Citado 2 vezes nas páginas 25 e 29.
- NETMARKETSHARE. *Mobile/Tablet Top Operating System Share Trend*. 2015. Disponível em: <<https://www.netmarketshare.com/operating-system-market-share.aspx?qprid=9&qpcustomb=1&qpct=4&qpsp=175&qpnp=12&qptimeframe=M>>. Citado na página 45.
- OSSADA, J. C.; MARTINS, L. E. G. Um estudo de campo sobre o estado da prática da elicitação de requisitos em sistemas embarcados. WER10 - Workshop em Engenharia de Requisitos, p. 53–62, 2010. Citado 4 vezes nas páginas 39, 40, 41 e 48.
- PRESSMAN, R. *Engenharia de Software*. McGraw Hill Brasil, 2009. ISBN 9788580550443. Disponível em: <<https://books.google.com.br/books?id=y0rH9wuXe68C>>. Citado 2 vezes nas páginas 25 e 31.
- PRESSMAN, R.; MAXIM, B. *Engenharia de Software-8ª Edição*. [S.l.]: McGraw Hill Brasil, 2016. Citado 2 vezes nas páginas 29 e 30.
- RAHMAN, A. A. Requirements engineering approach for real-time and embedded systems: a case study of android-based smart phone devices. In: ACM. *Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication*. [S.l.], 2014. p. 111. Citado 3 vezes nas páginas 25, 30 e 45.
- REGNELL, B. et al. Quality requirements in industrial practice—an extended interview study at eleven companies. the IEEE Computer Society, 2012. Citado 4 vezes nas páginas 39, 40, 41 e 48.
- REHMAN, T. ur; KHAN, M. N. A.; RIAZ, N. Analysis of requirement engineering processes, tools/techniques and methodologies. *International Journal of Information Technology and Computer Science (IJITCS)*, v. 5, n. 3, p. 40, 2013. Citado na página 33.



- ROUIBAH, K.; AL-RAFEE, S. Requirement engineering elicitation methods: A kuwaiti empirical study about familiarity, usage and perceived value. *Information management & computer security*, Emerald Group Publishing Limited, v. 17, n. 3, p. 192–217, 2009. Citado 5 vezes nas páginas 39, 40, 41, 42 e 48.
- RUMSEY, D. *Estatística para leigos*. [S.l.]: Alta Books Editora, 2009. Citado 2 vezes nas páginas 45 e 53.
- SALGER, F.; ENGELS, G.; HOFMANN, A. Inspection effectiveness for different quality attributes of software requirement specifications: An industrial case study. In: IEEE. *Software Quality, 2009. WOSQ'09. ICSE Workshop on*. [S.l.], 2009. p. 15–21. Citado 2 vezes nas páginas 35 e 36.
- SOLEMON, B.; SAHIBUDDIN, S.; GHANI, A. A. A. Requirements engineering problems and practices in software companies: An industrial survey. In: *Advances in Software Engineering*. [S.l.]: Springer, 2009. p. 70–77. Citado 3 vezes nas páginas 39, 40 e 41.
- SOMMERVILLE, I. *Engenharia de software*. [S.l.]: PEARSON EDUCATION - BR, 2011. Citado 6 vezes nas páginas 31, 32, 33, 34, 35 e 40.
- STATISTICS, T. P. *Number of available applications in the Google Play Store from December 2009 to February 2016*. 2016. Disponível em: <<http://www.statista.com/statistics/266210/number-of-available-applications-in-the-google-play-store/>>. Citado na página 45.
- TALBOT, A. *An investigation into requirements engineering current practice and capability in small and medium software development enterprises in New Zealand*. Tese (Doutorado) — Auckland University of Technology, 2011. Citado 4 vezes nas páginas 39, 40, 41 e 48.
- VIEIRA, H. C.; CASTRO, A. E. d.; JUNIOR, V. F. S. O uso de questionários via e-mail em pesquisas acadêmicas sob a ótica dos respondentes. *XIII SEMEAD Seminários em administração*, p. 01–13, 2010. Citado 2 vezes nas páginas 41 e 46.
- WASSERMAN, A. I. Software engineering issues for mobile application development. In: ACM. *Proceedings of the FSE/SDP workshop on Future of software engineering research*. [S.l.], 2010. p. 397–400. Citado 3 vezes nas páginas 25, 29 e 30.
- WOHLIN, C. et al. *Experimentation in software engineering*. [S.l.]: Springer Science & Business Media, 2012. Citado na página 36.
- ZOWGHI, D.; COULIN, C. Requirements elicitation: A survey of techniques, approaches, and tools. In: *Engineering and managing software requirements*. [S.l.]: Springer, 2005. p. 19–46. Citado na página 41.



# Apêndices



# APÊNDICE A – Questionário da pesquisa

# Research on Requirements Engineering

This form aims to identify the state of the practice in requirements engineering in the development of mobile applications. This research is part of the undergraduation conclusion work in Software Engineering at Federal University of Pampa (UNIPAMPA) - Brazil.

\*Obrigatório



## App Information

---

**1. What is the application name? \***

If your company has more than one app,  
choose one.

.....

**2. Which is the country where the application was developed? \****Marcar apenas uma oval.*

- Austria
- Australia
- Argentina
- Belgium
- Bolivia
- Bulgaria
- Brazil
- Canada
- China
- Costa Rica
- Czech Republic
- Denmark
- Dominican Republic
- Ecuador
- Egypt
- Spain
- Ethiopia
- Estonia
- France
- Finland
- Germany
- Greece
- India
- Italy
- Iran
- Japan
- Indonesia
- Israel
- Iraq
- Kazakhstan
- Kuwait
- Mexico
- Netherlands
- New Zealand
- Nigeria
- Norway
- North Korea
- Turkey
- Panama
- Paraguay
- Peru

- Poland
- Portugal
- Russia
- Saudi Arabia
- Slovak Republic
- South Africa
- South Korea
- Sudan
- Sweden
- Switzerland
- Thailand
- Taiwan
- Tunisia
- United Kingdom
- United Arab Emirates
- United States
- Ukraine
- Uruguay



**3. In which of the categories below your application belongs? \***

*Marcar apenas uma oval.*

- Android Wear
- Climate
- Shopping
- Communication
- Corporate
- Education
- Entertainment
- Sports
- Lifestyle
- Tools
- Finance
- Photography
- Humor
- Medicine
- Multimedia
- Music & Audio
- News & Magazines
- Customization
- Interactive Background
- Productivity
- Health and fitness
- Social
- Transport
- Travel & Local
- Widgets
- Games
- Family
- Book and Reference
- Comics

## Management and Process

---

**4. On the scale from 1 to 5, how much you consider that the requirements engineering process is defined? \***

Where one (1) corresponds to fully Ad Hoc and five (5) matches all mapped and documented steps.

*Marcar apenas uma oval por linha.*

	1	2	3	4	5
	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

**5. Which of these requirements engineering activities were carried out for this project? \***

Requirements elicitation is concerned with the origins of software requirements and how the software engineer can collect them. Objective of Requirements Analysis is detect and resolve conflicts; discover the bounds of the software and how it must interact with its organizational and operational environment; elaborate system requirements to derive software requirements. Software Requirements Specification typically refers to the production of a document that can be systematically reviewed, evaluated, and approved. Requirements Validation is to verify that a requirements document conforms to company standards and that it is understandable, consistent, and complete.

*Marque todas que se aplicam.*

- Requirements Elicitation
- Requirements Analysis
- Requirements Specification
- Requirements Validation

**6. How many % of the project time was spent specifically on the requirements engineering? \***

*Marcar apenas uma oval.*

- Less than 15%
- 15% to 30%
- 30% to 45%
- 45% to 60%
- 60% to 75%
- 75% to 90%
- I can not say

**7. How many people were in the development team of this APP? \***

*Marcar apenas uma oval.*

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- More than 10

**8. On the role of requirements analyst? \***

*Marcar apenas uma oval.*

- There is a person / group that works specifically with requirements engineering
- All members of the development team working on engineering requirements

## Requirements Elicitation

---

**9. Which techniques of requirement's elicitation were used?**

*Marque todas que se aplicam.*

- Brainstorming
- Card Sorting
- Domain Analysis
- Ethnography
- Facilitated meetings
- Interviews
- Introspection
- Group Work
- Joint Application Development (JAD)
- Laddering
- Observation
- Prototypes
- Questionnaire
- Requirements Workshops
- Scenarios
- Task Analysis
- User Stories
- Viewpoints
- Outro: .....

## Requirements Analysis

---

**10. Which one of the following approaches were used in the analysis and modeling of software requirements?**

*Marque todas que se aplicam.*

- No methodology
- Analysis Structured
- Data Flow Modeling
- Oriented Object Analysis
- SSADM
- Outro: .....

**11. Non-functional requirements were considered in the application development?**

*Marcas apenas uma oval.*

- Yes
- No

**12. If your answer was yes, which of the following non-functional requirements were considered?**

*Marque todas que se aplicam.*

- Energy consumption
- Performance
- Portability
- Memory Consumption
- Safety
- Reliability
- Usability
- Outro: .....

## Requirements Specification

---

**13. In which kind of notation was the requirements specification expressed?**

*Marque todas que se aplicam.*

- Graphical notations
- Natural language
- Specification formal
- Specification semiformal
- Outro: .....

## Requirements Validation

---

**14. Did you perform requirements inspection?**

*Marcar apenas uma oval.*

- Yes
- No

**15. If your answer was yes, which technique were used in the requirements inspection?**

*Marque todas que se aplicam.*

- Checklists
- DBR (Defect-Based Reading)
- MBR (Metric-Based Reading)
- PBR (Perspective-Based Reading)
- OORT's (Object-Oriented Reading Techniques)
- Reading Ad Hoc
- Outro:

## Software Requirements Tools

---

**16. Did you use any Case Tool in the Requirements Engineering phase?**

*Marcar apenas uma oval.*

Yes

No

**17. If your answer was yes, which one of the following tools were used?**

*Marque todas que se aplicam.*

- Acclaro DFSS
- Aligned Elements
- Blueprint
- Bright Green Projects
- Caliber RM
- Cameo Requirements+
- CASE Spec
- Cognition Cockpit
- GMARC
- inteGREAT
- Leap SE
- MacA&D / WinA&D
- MKS Integrity
- Polarion Requirements
- QFDcapture
- Rational Doors
- Reqtify & Requirement Central
- RTIME
- Outro:

---

Powered by

