



**Universidade Federal do Ceará  
Centro de Ciências  
Departamento de Computação  
Mestrado e Doutorado em Ciência da Computação**

**Dissertação de Mestrado**

**Avaliação da Confiança no Funcionamento de Sistemas  
Ubíquos usando Medidas de Qualidade**

**Andressa Bezerra Ferreira**

Fortaleza – Ceará  
2016

**Andressa Bezerra Ferreira**

**Avaliação da Confiança no Funcionamento de Sistemas  
Ubíquos usando Medidas de Qualidade**

Dissertação de Mestrado submetida à  
Coordenação do Programa de Pós-  
graduação em Ciência da Computação  
(MDCC) da Universidade Federal do  
Ceará (UFC) como requisito parcial para a  
obtenção do grau de Mestre em Ciência da  
Computação.

Orientadora: Rossana Maria de Castro  
Andrade, PhD.

Co-orientador: Reinaldo Bezerra Braga,  
Dr.

Dados Internacionais de Catalogação na Publicação  
Universidade Federal do Ceará  
Biblioteca Universitária

Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

---

F439a Ferreira, Andressa Bezerra.

Avaliação da Confiança no Funcionamento de Sistemas Ubíquos usando Medidas de Qualidade / Andressa Bezerra Ferreira. – 2016.

143 f. : il. color.

Dissertação (mestrado) – Universidade Federal do Ceará, Centro de Ciências, Programa de Pós-Graduação em Ciência da Computação, Fortaleza, 2016.

Orientação: Profa. Dra. Rossana Maria de Castro Andrade.

Coorientação: Prof. Dr. Reinaldo Bezerra Braga.

1. Qualidade de Software. 2. Medidas de Qualidade. 3. Confiança no Funcionamento. 4. Confiança. I. Título.

CDD 005

---

**ANDRESSA BEZERRA FERREIRA**

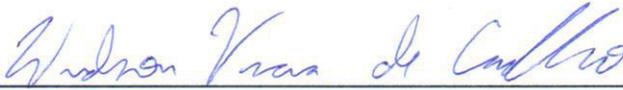
**Avaliação da Confiança no Funcionamento de Sistemas Ubíquos usando Medidas de Qualidade**

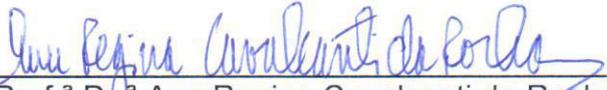
Dissertação submetida à Coordenação do Curso de Pós-Graduação em Ciência da Computação, da Universidade Federal do Ceará, como requisito para a obtenção do grau de Mestre em Ciência da Computação.

**BANCA EXAMINADORA**

  
Prof.<sup>a</sup> Dr.<sup>a</sup> Rossana Maria de Castro Andrade  
(Presidente)  
Universidade Federal do Ceará – UFC

  
Prof. Dr. Reinaldo Bezerra Braga  
(Coorientador)  
Instituto Federal de Ciência, Educação e  
Tecnologia do Ceará – IFCE

  
Prof. Dr. Windson Viana de Carvalho  
Universidade Federal do Ceará – UFC

  
Prof.<sup>a</sup> Dr.<sup>a</sup> Ana Regina Cavalcanti da Rocha  
Universidade Federal do Rio de Janeiro – UFRJ

  
Prof. Dr. Antônio Mauro Barbosa de Oliveira  
Instituto Federal de Ciência, Educação e  
Tecnologia do Ceará – IFCE

Fortaleza, 16 de dezembro de 2015.

Ao Deus da minha estranha calma.

# Agradecimentos

*A Deus,*

A Ti Jesus, Meu Senhor e Meu Salvador!

Pela fidelidade e amor extremo.

*A minha família,*

Pelo amor, pelo incentivo, pela dedicação e por todos os sacrifícios que fizeram por mim.

*A minha orientadora Rossana,*

Por acreditar em mim. Por me dar a oportunidade de realizar este trabalho e insistir, revisar, cortar, corrigir, melhorar, acrescentar, sugerir e me incentivar a cuidar de cada parágrafo como se fosse o último, de cada capítulo como se fosse o único. Obrigada!

*Ao meu coorientador Reinaldo,*

Por buscar sempre enriquecer este trabalho com suas sugestões e críticas brilhantes e sinceras.

*Aos demais professores e profissionais envolvidos,*

Pelo generoso auxílio em compartilhar comigo os seus conhecimentos.

*Aos meus amigos,*

“Por diante da vastidão do tempo e imensidão do universo, me darem a honra de dividir um mesmo planeta, tempo e época com eles”.

*Ao MDCC, GREat, CTQS e ARiDa,*

Pela acolhida e por me darem quatro novas famílias.

Que sorte a minha ter vocês!

*A CAPES, FUNCAP e FCPC,*

Pelo apoio financeiro ao longo do mestrado.

“Talvez seja porque meus maiores desafios tenham vindo de ameaças externas, mas eu não acredito no banimento dos laços. Não posso imaginar a minha vida sem Mara, Leia, ou mesmo Han.”

Luke Skywalker

# Resumo

Sistemas ubíquos são, por princípio, autônomos e capazes de perceber o ambiente, as atividades e necessidades do usuário, sem que seja necessário determiná-las explicitamente. Consequentemente, a aceitação e utilização desse tipo de sistema requerem confiança no funcionamento por parte do usuário. Sendo assim, é importante avaliar e garantir ao usuário final de um sistema ubíquo, a confiança no seu funcionamento. Uma das possíveis técnicas para a avaliação de características de qualidade, como por exemplo, a confiança no funcionamento, de um *software* é a realização de medições. Entretanto, por englobar diversos atributos de qualidade, avaliar a confiança no funcionamento de um sistema de *software* já é uma tarefa difícil e, devido às peculiaridades inerentes aos sistemas ubíquos (e.g., sensibilidade ao contexto, mobilidade e heterogeneidade), essa avaliação se torna ainda mais desafiadora do que em sistemas convencionais. Sendo assim, este trabalho de mestrado tem como objetivo investigar e, quando necessário, adaptar e propor medidas de qualidade para avaliação da confiança no funcionamento em sistemas ubíquos, com foco em aplicações executando em dispositivos móveis (e.g., *smartphones*, *tablets*, e *smart devices*). Três estudos de casos são utilizados neste trabalho para avaliar as medidas definidas: um guia de visitas móveis, um sistema para controle de volume de chamadas do celular, e um sistema de detecção e alerta de quedas. Como principal contribuição espera-se auxiliar o analista de qualidade na identificação de quais atributos estão impactando a confiança do usuário no funcionamento do sistema ubíquo, impulsionando, consequentemente, melhorias que possibilitem o aumento da aceitação e utilização desse tipo de sistema.

**Palavras-chave:** Sistemas Ubíquos, Qualidade de *Software*, Medidas de Qualidade, Confiança no Funcionamento, Confiança.

# Abstract

Ubiquitous systems are, in principle, autonomous and capable of perceiving environment, activities and user needs, without having to explicitly determine them. Consequently, in order to use this type of systems, a user needs to trust them. One of the possible techniques to evaluate the user's trust in the system's functionalities is using measurements. However, since several quality attributes are involved to assure this characteristic, this is a hard task. Moreover, due to other specific characteristics of ubiquitous systems, such as context-awareness, mobility, and heterogeneity, their quality evaluation becomes more challenging than in traditional systems. With that in mind, this master thesis' objective is to investigate and, when necessary, adapt and propose software quality measures to evaluate the user's trust in the functionality of ubiquitous systems (i.e., their dependability) that run in mobile devices (e.g., smartphones, tablets, and smart devices). Three case studies are used in this work to evaluate the proposed measures: a mobile visit guide, a cellphone's call volume control system, and a system to alert and detect falls. The main contribution of this work is to help software quality analysts in the identification of which attributes are impacting the trust of the user in the functionality of a ubiquitous system and, consequently, suggesting improvements to the developers that allow an increase in the acceptance and use of this kind of systems.

**Key-words:** Ubiquitous Systems, Software quality, Quality Measures, Dependability, Trust.

# Lista de Figuras

Figura 1: Metodologia utilizada nesta dissertação .....	17
Figura 2: GREat Tour - Guia de visitas móvel do laboratório GREat.....	19
Figura 3: GREat Mute - Aplicação em desenvolvimento no laboratório GREat.....	23
Figura 4: Partes componentes da norma SQuaRE .....	25
Figura 5: A metodologia GQM retirada de (ROCHA et al 2012) .....	28
Figura 6: A confiança e seus atributos adaptada de (SANTOS 2014) .....	31
Figura 7: A satisfação e seus atributos adaptada de (ISO 25010) .....	32
Figura 8: A confiança no funcionamento e seus atributos .....	33
Figura 9: Processo de identificação dos trabalhos relacionados .....	34
Figura 10: Planejamento do GQM.....	44
Figura 11: Medidas de Confiabilidade.....	66
Figura 12: Medidas de Proteção e Confidencialidade.....	69
Figura 13: Medidas de Disponibilidade.....	74
Figura 14: Medidas de Integridade.....	78
Figura 15: Medidas de Manutenibilidade .....	79
Figura 18: Planejamento para os estudos de caso .....	86
Figura 19: GREatMute Versão 2.0 .....	89
Figura 20: Representação do funcionamento do sistema fAlert (PIVA et al 2014).....	101
Figura 21: Interface gráfica do fAlert.....	103
Figura 22: Dispositivos espalhados pelo corpo do usuário .....	105
Figura 23: Casos de queda.....	105
Figura 24: Resultados obtidos para a medida precisão.....	108
Figura 25: Resultados obtidos para a medida sensibilidade.....	108
Figura 26: Resultados obtidos para a medida disponibilidade.....	109
Figura 27: Resultados obtidos para a medida especificidade.....	109
Figura 28: Resultados obtidos para a medida grau de desempenho.....	109
Figura 29: Resultados obtidos para a medida acurácia .....	110
Figura 30: Telas do GREat Tour .....	115
Figura 31: Informações providas pelo GREat Tour.....	116

# Lista de Tabelas

Tabela 1: Critérios Qualitativos para Definição de Medidas .....	26
Tabela 1 (Continuação): Critérios Qualitativos para Definição de Medidas.....	27
Tabela 2: Template para a definição do objetivo de medição (Basili et al 1994).....	29
Tabela 3: Palavras-chave utilizadas para a busca de trabalhos .....	35
Tabela 4: Referência e Descrição dos Trabalhos Relacionados .....	36
Tabela 5: Comparação dos Trabalhos Correlatos .....	38
Tabela 6: Questões e Desafios .....	41
Tabela 7: <i>Abstract Sheet</i> .....	46
Tabela 8: <i>Abstract Sheet</i> – Disponibilidade .....	47
Tabela 9: <i>Abstract Sheet</i> – Integridade .....	48
Tabela 10: <i>Abstract Sheet</i> – Manutenibilidade.....	49
Tabela 11: <i>Abstract Sheet</i> – Confidencialidade.....	50
Tabela 12: <i>Abstract Sheet</i> – Proteção.....	51
Tabela 13: <i>Abstract Sheet</i> – Confiabilidade.....	52
Tabela 13 (Continuação): <i>Abstract Sheet</i> – Confiabilidade .....	53
Tabela 14: Sobre os Resultados das Funções de Medições .....	56
Tabela 15: As 11 Medidas para a questão Q1.1.....	57
Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1.....	58
Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1.....	59
Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1.....	60
Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1.....	61
Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1.....	62
Tabela 16: As 3 Medidas para a questão Q1.2.....	67
Tabela 16 (Continuação): As 3 Medidas para a questão Q1.2.....	68
Tabela 17: As 6 Medidas para a questão Q1.3.....	70
Tabela 17 (Continuação): As 6 Medidas para a questão Q1.3.....	71
Tabela 17 (Continuação): As 6 Medidas para a questão Q1.3.....	72
Tabela 18: As 2 Medidas para a questão Q1.4.....	75
Tabela 19: As 3 Medidas para a questão Q1.5.....	76
Tabela 19 (Continuação): As 3 Medidas para a questão Q1.5.....	77
Tabela 20: Perfil dos usuários envolvidos nos estudos de caso .....	87
Tabela 21: Dispositivos usados nos estudos de caso.....	88
Tabela 22: Resultados - Disponibilidade – GREat Mute v1.0 .....	90
Tabela 22 (Continuação): Resultados - Disponibilidade – GREat Mute v1.0.....	91
Tabela 23: Resultados de Integridade – GREat Mute v1.0 .....	91

Tabela 24: Resultados – Manutenibilidade – GREat Mute v1.0.....	92
Tabela 25: Resultados - Confiabilidade - GREat Mute v1.0 .....	92
Tabela 26: Resultados - Proteção - GREat Mute v1.0 .....	93
Tabela 27: Resultados - Disponibilidade - GREat Mute v2.0 .....	94
Tabela 28: Resultados - Integridade - GREat Mute v2.0.....	95
Tabela 29: Resultados - Manutenibilidade - GREat Mute v2.0.....	95
Tabela 30: Resultados - Confiabilidade - GREat Mute v2.0 .....	96
Tabela 31: Resultados - Proteção - GREat Mute v2.0 .....	97
Tabela 32: Resultados - Disponibilidade - GREat Mute v3.0 .....	98
Tabela 33: Resultados - Integridade – GREat Mute v3.0 .....	98
Tabela 34: Resultados - Manutenibilidade - GREat Mute v3.0.....	99
Tabela 35: Resultados - Confiabilidade - GREat Mute v3.0 .....	100
Tabela 36: Resultados - Proteção - GREat Mute v3.0 .....	100
Tabela 37: Configuração técnica do aparelho utilizado [PIVA et al 2014].....	102
Tabela 38: Informações providas pelos Sensores Utilizados para Detecção .....	104
Tabela 39: Quantidade de testes por categoria .....	106
Tabela 40: Medidas utilizadas para avaliar o fAlert.....	107
Tabela 41: Resultados - Disponibilidade - fAlert .....	111
Tabela 42: Resultados - Integridade - fAlert .....	112
Tabela 43: Resultados - Manutenibilidade - fAlert.....	112
Tabela 44: Resultados - Confiabilidade - fAlert.....	113
Tabela 45: Resultados - Proteção - fAlert.....	114
Tabela 46: Resultados - Disponibilidade - GREat Tour .....	117
Tabela 47: Resultados - Integridade - GREat Tour .....	117
Tabela 48: Resultados - Manutenibilidade - GREat Tour .....	118
Tabela 49: Resultados - Confiabilidade - GREat Tour .....	118
Tabela 50: Resultados - Proteção - GREat Tour .....	119
Tabela 51: Validação dos resultados - Confiabilidade - GREatMute.....	120
Tabela 52: Validação dos resultados - Confiabilidade - fAlert .....	121
Tabela 53: Validação dos resultados - Confiabilidade - GREat Tour.....	122

# Lista de Abreviaturas

IHC	Interação Humano-Computador
ISO	International Organisation for Standardisation
GPS	Global Positioning System
GREat	Grupo de Redes de Computadores, Engenharia de Software, e Sistemas
QRCode	Quick Responde Code
GQM	Goal-Question-Metric
PoC	Proof of Concept
TCP/IP	Transmission Control Protocol/ Internet Protocol

# Sumário

<b>1 INTRODUÇÃO .....</b>	<b>14</b>
1.1 CONTEXTUALIZAÇÃO E MOTIVAÇÃO .....	14
1.2 OBJETIVO E METAS .....	16
1.3 METODOLOGIA UTILIZADA .....	17
1.4 ORGANIZAÇÃO DA DISSERTAÇÃO .....	18
<b>2 REFERENCIAL TEÓRICO E TRABALHOS CORRELATOS.....</b>	<b>19</b>
2.1 SISTEMAS UBÍQUOS .....	19
2.2 QUALIDADE DE SOFTWARE .....	24
2.3 CONFIANÇA NO FUNCIONAMENTO DE SISTEMAS DE SOFTWARE .....	30
2.4 TRABALHOS CORRELATOS.....	34
2.4.1 Metodologia de Busca.....	34
2.4.2 Análise dos Trabalhos Correlatos e Desafios em aberto.....	37
2.5 CONSIDERAÇÕES FINAIS .....	42
<b>3 AS MEDIDAS PARA CONFIANÇA NO FUNCIONAMENTO .....</b>	<b>43</b>
3.1 PLANEJAMENTO .....	43
3.2 DEFINIÇÃO .....	44
3.2.1 OBJETIVO DE MEDIÇÃO .....	45
3.2.2 ENTREVISTAS.....	46
3.2.3 DERIVANDO AS QUESTÕES .....	53
3.2.4 MEDIDAS .....	54
3.3 COLETA DAS MEDIDAS.....	79
3.4 INTERPRETAÇÃO DOS RESULTADOS .....	80
3.5 DISCUSSÃO SOBRE AS MEDIDAS .....	80
3.6 CONSIDERAÇÕES FINAIS .....	83
<b>4 AVALIAÇÃO DAS MEDIDAS.....</b>	<b>85</b>
4.2 GREAT MUTE.....	89
4.2.1 GREAT MUTE VERSÃO 1.0 .....	90
4.2.2 GREAT MUTE VERSÃO 2.0.....	93
4.2.3 GREAT MUTE VERSÃO 3.0 .....	97
4.3 fALERT.....	101
4.3.1 Mecanismo para a Detecção de Quedas.....	103
4.2.4 Testes Realizados.....	104
4.2.4 Interpretação dos Resultados.....	108
4.4 GREAT TOUR .....	114
4.5 VALIDAÇÃO DOS RESULTADOS.....	119
4.5.1 GREat Mute .....	120
4.5.2 fAlert .....	121
4.5.3 GREat Tour.....	122
4.6 CONSIDERAÇÕES FINAIS .....	123
<b>5 CONCLUSÃO.....</b>	<b>124</b>
5.1 RESULTADOS ALCANÇADOS .....	124
5.2 LIMITAÇÕES.....	127
5.3 TRABALHOS FUTUROS .....	127
<b>REFERÊNCIAS BIBLIOGRÁFICAS.....</b>	<b>129</b>

# 1 INTRODUÇÃO

Esta dissertação de mestrado tem como objetivo identificar, definir e aplicar um conjunto de medidas de qualidade de *software* para avaliação da *Confiança no Funcionamento* em sistemas ubíquos (e.g., *smartphones*, *tablets*, e *smart devices* em geral). Este capítulo introduz, na Seção 1.1, o contexto no qual este trabalho de dissertação está inserido e a motivação para o seu desenvolvimento, na Seção 1.2 são descritos os objetivos desta dissertação, na Seção 1.3 é apresentada a metodologia utilizada nesta pesquisa e, por fim, a Seção 1.4 apresenta a estrutura organizacional deste trabalho.

## 1.1 Contextualização e Motivação

Os sistemas ubíquos possibilitam tornar real o conceito de computação em todo lugar e a qualquer momento, fazendo com que o uso dos recursos computacionais e a comunicação sejam transparentes para o usuário (YAU et al. 2002). Para tanto, é necessário que este tipo de sistema possua a habilidade de perceber o ambiente, as atividades e necessidades do usuário, sem que seja necessário determiná-los explicitamente (WEISER 1991) (SANTOS 2014) (SPÍNOLA 2007).

A autonomia desses sistemas pode ser proporcionada por meio do uso da sensibilidade ao contexto. Com o uso de informações contextuais, o usuário não precisa necessariamente realizar uma operação explícita (e.g., clicar em um botão) para que o sistema entenda isso como uma entrada. Muitas vezes, ações cotidianas (e.g., entrar em um ambiente) podem ser suficientes para que o sistema reaja provendo serviços (e.g., ligar o ar-condicionado de maneira automática) (DEY e ABOWD 2001)(LIMA 2011)(DEBASHIS e AMITAVA 2003).

Um exemplo de aplicação ubíqua que faz uso da sensibilidade ao contexto é um sistema de monitoramento de quedas em pessoas que necessitam de cuidados especiais, encontrado em (PIVA et al 2014). Utilizando dados obtidos por meio dos sensores do dispositivo móvel do usuário (e.g., localização, posição do dispositivo, variação de movimento), que podem servir para representar seu atual contexto, é possível detectar e alertar quedas.

Diante desse cenário, fazer com que aplicações percebam o contexto no qual o usuário está inserido pode ser fundamental para que a tomada de decisão seja a mais benéfica e adequada possível. Entretanto, a aceitação e, em consequência, a utilização em

escala desse tipo de sistema necessitam, entre outras características, de confiança no seu funcionamento<sup>1</sup> (*dependability*) por parte do usuário. Neste caso, é necessário levar em consideração características de qualidade, tais como: disponibilidade (*availability*), confiabilidade (*reliability*), proteção (*safety*), confidencialidade (*confidentiality*), integridade (*integrity*) e manutenibilidade (*maintenability*). Essas características, segundo (AVIZIENIS et al 2004), podem impactar ou caracterizar a confiança no funcionamento, apresentada em mais detalhes no restante deste trabalho.

Uma das técnicas para a avaliação de características de qualidade é a realização de medições (SCHOLTZ e CONSOLVO 2004) e (SANTOS 2013). Uma medição consiste no processo contínuo de definição, coleta e análise de dados sobre o processo de desenvolvimento de software e seus produtos, fornecendo informações significativas com o objetivo de melhorá-los (ISO 25000). O resultado da medição é chamado de medida<sup>2</sup>. O conjunto de propriedades ou atributos de um produto de software, pelas quais a qualidade pode ser descrita e avaliada é chamado de característica.

Apesar da confiança no funcionamento de um sistema ubíquo ser uma questão importante, por englobar diversos atributos de qualidade, garanti-la é uma tarefa difícil. Segundo (ROCHA et al 2010), os sistemas ubíquos impõem obstáculos ao uso das metodologias existentes na engenharia de software pois, devido as suas características específicas (e.g., mobilidade, heterogeneidade), não é possível projetá-los inicialmente como um todo, de maneira coerente pois, alguns comportamentos do sistema podem ser conhecidos apenas no momento do uso.

Pesquisadores como (SCHOLTZ e CONSOLVO 2004)(SANTOS 2014) têm reportado dificuldades para utilizar as formas de avaliação de qualidade existentes (e.g., modelos e medidas de qualidade) nesses sistemas, buscando como alternativa, adaptá-las,

---

<sup>1</sup>*Nesta pesquisa de mestrado, o conceito Dependability é traduzido como Confiança no Funcionamento, baseado em discussões de pesquisadores da área de Engenharia de Software que tentaram chegar a um consenso sobre a tradução correta do termo para o português [http://www.cs.kent.ac.uk/people/staff/rdl/CoF/]*

<sup>2</sup>*De acordo com a [ISO 25010], uma medida de qualidade é definida como uma variável a qual é atribuído um valor como resultado da medição [ISO 25010]. De acordo com a norma, as medidas quantitativas são também chamadas de métricas. Porém, neste trabalho, é utilizada apenas a nomenclatura medida.*

levando em consideração as diferenças (e.g., sensibilidade ao contexto, mobilidade e heterogeneidade) entre sistemas convencionais e ubíquos.

Logo, identificar e apresentar medidas que permitam a avaliação da confiança no funcionamento de sistemas ubíquos, em particular, aquelas aplicações que executam dispositivos móveis, ainda é um desafio de pesquisa em aberto.

## 1.2 Objetivo e Metas

Considerando o que foi apresentado na seção anterior, este trabalho de dissertação de mestrado tem como objetivo identificar e, quando necessário, propor medidas de qualidade de *software* para avaliação da confiança no funcionamento (i.e., *dependability*) em sistemas ubíquos.

Para atingir esse objetivo, as seguintes metas foram definidas:

- (i) Buscar na literatura medidas para avaliação da confiança no funcionamento (i.e., *dependability*) e seus atributos em sistemas tradicionais e ubíquos;
- (ii) Identificar se as medidas utilizadas em sistemas tradicionais podem ser aplicadas em sistemas ubíquos e, caso não possam, propor uma possível adaptação ou a definição novas medidas;
- (iii) Avaliar as medidas apresentadas utilizando estudos de caso.

É importante ressaltar que o objetivo deste trabalho está inserido em um projeto de pesquisa chamado Maximum<sup>3</sup> intitulado “Uma Abordagem baseada em Medições para Avaliação da Qualidade da Interação Humano-Computador (IHC) em Sistemas Ubíquos”. Os resultados deste trabalho se agregam ao trabalho de (SANTOS 2014), que propôs um conjunto de características e medidas para a avaliação da qualidade da IHC em sistemas ubíquos.

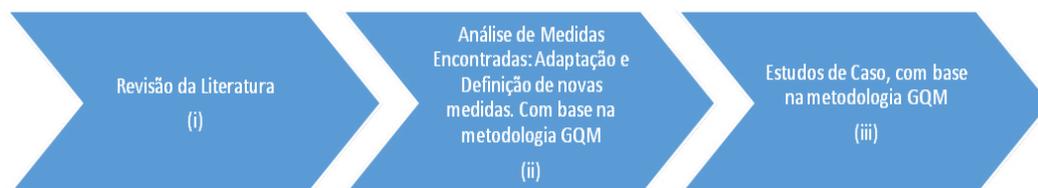
Os resultados alcançados por este trabalho visam auxiliar o analista de qualidade na identificação de quais atributos estão impactando a confiança do usuário no funcionamento desse tipo de sistema e, em consequência, que os desenvolvedores usem os resultados dessas avaliações para melhorar a qualidade dessas aplicações, aumentando a aceitação e utilização das mesmas.

---

<sup>3</sup>Esse projeto foi financiado pela Fundação Cearense de Apoio Tecnológico e Pesquisas (FUNCAP) com o número de processo INC-0064-00012.01.00/12.

### 1.3 Metodologia Utilizada

Para alcançar o objetivo mencionado na seção anterior, a seguinte metodologia é utilizada nesta dissertação, conforme ilustrada na Figura 1.



**Figura 1: Metodologia utilizada nesta dissertação**

Para a etapa (i), é necessária uma revisão bibliográfica (realizada neste trabalho por meio do uso de uma *string* de busca em três diferentes bases). Com isso, deve ser possível identificar se existem na literatura medidas para a avaliação da confiança no funcionamento em sistemas de *software* ubíquos. Caso existam, é necessário verificar se elas levam em consideração as características particulares desses sistemas (mencionadas na Seção 1.1), caso isto não ocorra, é necessária uma adaptação das medidas encontradas. Ainda sobre a etapa (i), o mapeamento sistemático realizado por (SANTOS 2014) serve como ponto de partida para esta busca, pois nele foram identificados os principais autores e trabalhos na área de medições em sistemas ubíquos, bem como de palavras que ajudam a compor a *string* de busca utilizada nesta etapa.

Além da adaptação de medidas já existentes, é preciso também verificar a necessidade da definição de novas medidas. Com isso, na etapa (ii), é utilizado o paradigma *Goal-Question-Metric* (GQM), apresentado em (BASILI et. al. 1994).

Para a avaliação relacionada a etapa (iii), são utilizadas três aplicações. O uso das medidas nessas aplicações permite a coleta de dados quantitativos e qualitativos. A análise dos resultados obtidos permite avaliar se as medidas de qualidade de software apresentadas neste trabalho são capazes de fornecer indícios sobre a característica de qualidade confiança no funcionamento. Cabe mencionar que, os estudos de caso para avaliação das medidas deste trabalho são experimentos, usando as questões da metodologia GQM e não seguem o rigor dos estudos de casos da engenharia de

software experimental.

## 1.4 Organização da dissertação

Além deste capítulo introdutório, esta dissertação está estruturada em mais quatro capítulos e três apêndices.

**Capítulo 2 – Referencial Teórico e Trabalhos Correlatos:** tem por objetivo fornecer o embasamento teórico acerca das áreas que compõem esse trabalho, que são Sistemas Ubíquos, Qualidade de Software e Confiança no Funcionamento em Sistemas de Software. Além disso, apresenta um subconjunto de trabalhos da área, encontrados por meio de uma busca baseada em *string*. Onde, para a definição da *string*, foram utilizadas as palavras-chave relacionadas ao trabalho, bem como os objetivos apresentados na Seção 1.2.

**Capítulo 3 – Medidas de Qualidade de *Software* para Avaliação da Confiança no Funcionamento em Sistemas Ubíquos:** apresenta as medidas de qualidade de software para avaliação da confiança no funcionamento de sistemas ubíquos.

**Capítulo 4 – Avaliação das Medidas: Exemplo de Uso e Critérios Qualitativos:** apresenta os resultados obtidos com a aplicação das medidas propostas nos estudos de caso e os benefícios alcançados com esses resultados.

**Capítulo 5 – Conclusão:** apresenta um resumo dos resultados alcançados neste trabalho e seus direcionamentos futuros.

**Apêndice A – Formulário para traçar o perfil do usuário:** formulário utilizado com as pessoas que participam dos testes relacionados aos estudos de caso apresentados no Capítulo 4.

**Apêndice B – Questionário para ser utilizado pelo usuário:** questionário a ser utilizado como método de coleta de algumas medidas apresentadas no Capítulo 3, onde o ponto de vista a ser avaliado é o do usuário final.

**Apêndice C – Questionário para ser utilizado pelo desenvolvedor:** questionário a ser utilizado como método de coleta de algumas medidas apresentadas no Capítulo 3, onde o ponto de vista a ser avaliado é o do desenvolvedor.

## 2 REFERENCIAL TEÓRICO E TRABALHOS CORRELATOS

Neste capítulo é apresentada uma revisão e análise bibliográfica das áreas de estudo relacionadas a esta pesquisa, a qual busca facilitar o entendimento dos desafios e soluções apontados neste trabalho. A Seção 2.1 apresenta definições e princípios da área de Sistemas Ubíquos. A Seção 2.2 apresenta as principais definições da área de Qualidade de *Software*. A Seção 2.3 apresenta os conceitos da área de Confiança, em especial a Confiança no Funcionamento de Sistemas de *Software*. A Seção 2.4, por sua vez, apresenta um subconjunto de trabalhos da área, comparando-os com os objetivos desta dissertação. Por fim, a Seção 2.5 apresenta as considerações finais do capítulo.

### 2.1 Sistemas Ubíquos

Imagine-se entrando em um laboratório de pesquisa que você ainda não conhece, acompanhado somente de um dispositivo móvel onde está instalada uma aplicação que servirá como um guia de visitas do local, como o exemplo ilustrado na Figura 2.



Figura 2: GREAt Tour<sup>4</sup> - Guia de visitas móvel do laboratório GREAt

Ao entrar, você é imediatamente notificado sobre a sala na qual se encontra, acompanhado de um mapa geral que retrata os demais ambientes. Guiado pelas informações disponibilizadas pelo aplicativo, você é capaz de conhecer todo o laboratório, obtendo informações acerca de funcionários, pesquisadores, projetos de pesquisa desenvolvidos e trabalhos relacionados. À medida que se movimenta pelo ambiente, você é

---

<sup>4</sup>Guia de visitas móveis sensível ao contexto, produto de uma Linha de Produto de Software Sensível ao Contexto chamada Mobliline, desenvolvida pelo grupo GREAt e que pode ser encontrada em [Marinho et al. 2012]

atualizado de informações, dependendo de sua posição, sem a necessidade de realizar operações diretas com o dispositivo (e.g, clicar em um botão).

Portanto, munido apenas de um dispositivo móvel você é capaz de conhecer e passear por um local sobre o qual você não tem nenhum conhecimento. Esse é um típico exemplo de ambiente que remete à ideia de computação ubíqua, idealizada em 1991, por Marc Weiser.

Considerado o pai da computação ubíqua, Weiser afirmou que no futuro computadores irão habitar os mais triviais objetos: etiquetas de roupas, xícaras de café, interruptores de luz, canetas, entre outros, de forma invisível para o usuário. Com isso, as pessoas devem aprender a conviver com computadores, e não apenas interagir com eles (WEISER 1991).

No mundo idealizado pelo autor, a computação move-se para fora das estações de trabalho e computadores pessoais, tornando-se indistinguível em nossa vida cotidiana. Nesta concepção, o computador tem a capacidade de obter informações do ambiente no qual está e utilizá-las para prover serviços de maneira natural. Assim, surge a capacidade de computadores agirem de forma autônoma e “inteligente” no ambiente.

A fim de se obter uma definição mais precisa e completa sobre computação ubíqua, outras definições podem ser encontradas na literatura. Alguns exemplos são:

- “Computação ubíqua é o uso de um conjunto de computadores dos mais variados tamanhos, formatos e funções, que de forma coordenada e autônoma, auxiliam as pessoas na realização das diversas tarefas cotidianas. Esse auxílio é realizado de tal forma que a infraestrutura computacional responsável fica escondida no ambiente.” de (LIMA 2011).
- “A computação ubíqua está presente quando serviços computacionais tornam-se disponíveis para os usuários de tal forma que o computador não é mais uma ferramenta visível ou essencial para acessar esses serviços. Assim, os serviços computacionais são acessados em qualquer tempo ou lugar, de forma transparente, através do uso de dispositivos comuns” (SPINOLA et al 2007).

Apesar de definidas em diferentes contextos de pesquisa, as diversas definições encontradas para sistemas ubíquos concordam com relação às ideias-chave que podem ser extraídas do trabalho de (WEISER 1991): invisibilidade, transparência, naturalidade e, conseqüentemente, autonomia.

É possível, então, estabelecer um conjunto de princípios associados aos sistemas ubíquos e que se fazem pertinentes a este trabalho de pesquisa (LIMA 2011):

- Onipresença dos serviços - usuários podem se mover nos limites físicos de um sistema ubíquo e trocar de dispositivos sem perder o acesso aos serviços;
- Captura de experiências e intenções - o sistema deve ser capaz de perceber o ambiente, as atividades e necessidades do usuário, sem que seja necessário determiná-las explicitamente;
- Heterogeneidade de serviços e dispositivos - habilidade de possuir uma grande variedade de dispositivos e serviços;
- Interoperabilidade - capacidade de diferentes tipos de dispositivos e serviços interagirem automaticamente sem a intervenção do usuário;
- Descentralização - as responsabilidades devem ser distribuídas entre vários dispositivos no ambiente, onde cada um é responsável pela execução de um conjunto de funções e tarefas;
- Mínima intervenção do usuário - os dispositivos eletrônicos que compõem o ambiente ubíquo devem estar o mais longe possível da percepção dos usuários, com o principal objetivo de não requisitar qualquer operação direta sobre eles;
- Tolerância à falha - a habilidade que o sistema possui de corrigir a si próprio depois da ocorrência de falhas sem envolver os usuários nesse processo;
- Adaptação ao comportamento - habilidade de se adaptar ao contexto e prover serviços e/ou informações relevantes para os usuários.

Diante desse contexto, neste trabalho de pesquisa, os sistemas ubíquos considerados são, por natureza, sensíveis ao contexto.

A computação sensível ao contexto aborda a ideia de que os computadores podem tanto perceber quanto reagir ao ambiente no qual estão inseridos buscando facilitar as atividades humanas. Os dispositivos usam informações sobre as circunstâncias em que se encontram e, baseados em regras ou estímulos inteligentes, reagem de acordo (DEBASHIS E AMITAVA 2003). Segundo (LIMA 2011) é impossível conceber um sistema ubíquo que não seja sensível ao contexto, já que os princípios particulares desses sistemas, mencionados anteriormente, precisam da sensibilidade ao contexto para ser alcançados.

Uma das principais definições de contexto é apresentada por (DEY 2001). Segundo o autor, contexto é qualquer informação que pode ser usada para caracterizar a situação de uma entidade. Uma entidade pode ser uma pessoa, um lugar ou um objeto considerado

relevante para a interação entre um usuário e uma aplicação, incluindo o próprio usuário e a aplicação.

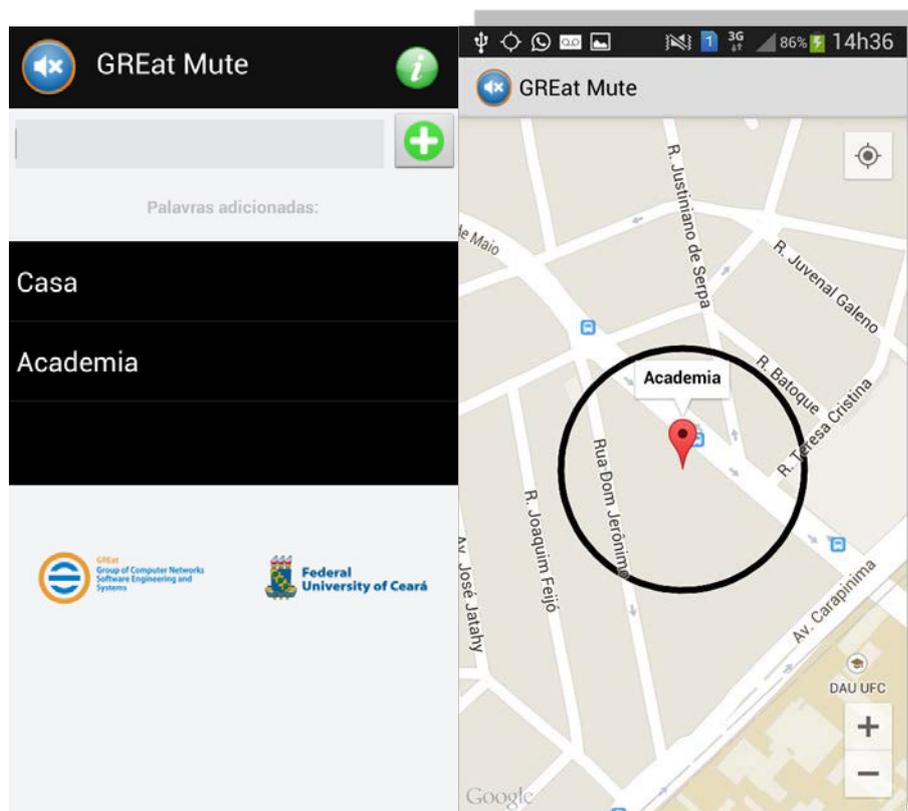
Segundo (DEY 2001), os aspectos relacionados à obtenção das informações contextuais são descritos com base em cinco dimensões, também conhecidas como “cinco W’s H”, definidos a seguir:

- Quem (*Who*) - relaciona-se a identidade do usuário;
- Onde (*Where*) - trata-se da localização do usuário;
- O que (*What*) - foca nas atividades do usuário;
- Quando (*When*) - relacionado ao contexto temporal; e
- Por que (*How*) - está relacionado ao contexto que levou o usuário a fazer determinada ação.

De acordo com o autor, um sistema é dito sensível ao contexto se utiliza informação contextual para disponibilizar informações ou serviços relevantes para o usuário.

Para melhor compreensão da relevância de um serviço que esse sistema pode prover, suponha agora o seguinte cenário: imagine que você está participando de uma importante reunião e opta por colocar o seu dispositivo móvel no modo silencioso, para evitar interrupções. Ao sair da reunião, você esquece de ativar o modo normal do dispositivo e perde uma série de ligações importantes, pois não ouviu as chamadas.

Esse inconveniente poderia ser facilmente evitado com o uso de uma aplicação ubíqua para dispositivos móveis que, utilizando informações sobre compromissos cadastrados na agenda pessoal do usuário e sua atual localização, fosse capaz de ativar ou desativar o modo silencioso em momentos considerados adequados. Um exemplo de aplicação desse tipo é ilustrado na Figura 3.



**Figura 3: GREAt Mute - Aplicação em desenvolvimento no laboratório GREAt**

Nas três aplicações apresentadas até então (sistema de detecção de quedas no Capítulo 1, guia de visitas e aplicação para automatizar o uso do modo silencioso no Capítulo 2), é possível perceber que a aceitação e utilização desses tipos de sistemas requerem um certo nível de confiança por parte do usuário. Visto que, delegar o monitoramento de pessoas que necessitam de cuidados especiais a um sistema de software, passear em um local ao qual nunca foi, baseado somente nas informações disponibilizadas por uma aplicação, e delegar as configurações do seu aparelho móvel a um sistema<sup>5</sup>, são ações baseadas na confiança em seu funcionamento.

Como mencionado no Capítulo 1, diversas características de qualidade (e.g., usabilidade, transparência, disponibilidade, confiança) exercem forte influência sobre a aceitação e utilização de sistemas ubíquos por parte do usuário. Uma das possíveis técnicas para avaliar essas características e, porventura, impulsionar o uso real desses sistemas, é a realização de medições. Porém, para tanto, é necessário o entendimento de diversos conceitos da área de qualidade de *software*, começando pelo significado do termo.

<sup>5</sup>Neste trabalho, ao falar sobre sistema, estão sendo considerados apenas sistemas ubíquos para dispositivos móveis (e.g. smartphones, tablets). Dando ênfase ao seu comportamento exterior (e.g., comportamento percebido pelo usuário final).

## 2.2 Qualidade de Software

Neste trabalho de dissertação de mestrado, a qualidade de software é vista como o grau no qual um produto de software satisfaz as necessidades explícitas e implícitas quando usado sob condições específicas (ISO 25000 2005). A importância dessa norma para este trabalho está no fato de que ela leva em consideração o contexto de uso do sistema que está sendo avaliado. Este aspecto, por sua vez, está intimamente relacionado aos objetivos e motivação desta pesquisa.

Ainda em relação aos conceitos adotados nesta pesquisa, é importante mencionar que, neste trabalho, um atributo de qualidade é visto como uma propriedade ou característica de um sistema que pode ser avaliada por meios quantitativos ou qualitativos. Uma característica de qualidade é uma propriedade que se deseja que o sistema possua e, por fim, uma subcaracterística de qualidade é o refinamento de uma característica. Estas definições também se encontram em conformidade com a ISO 25000.

Diante disso, algumas das principais questões a serem analisadas quando o assunto é qualidade de software é como garanti-la e avaliá-la. Para tanto, surgiram as normas de qualidade<sup>6</sup>, dentre elas, a *SQuaRE*.

### 2.2.1 Norma de Qualidade *SQuaRE*

Ao longo dos anos, as normas ISO<sup>7</sup> tornaram-se sinônimo de preocupação com qualidade. As suas normas oferecem uma base comum para julgar a qualidade de produtos e definir critérios em contratos e negociações.

Conforme mencionado no início deste capítulo, a definição de qualidade adotada nesta pesquisa é proposta pela ISO 25000, também conhecida como norma ou projeto *SQuaRE*. A norma ISO/IEC 25000 é uma evolução das normas ISO/IEC 9126 e ISO/IEC 14598 que tratam da qualidade de produtos de software.

Segundo (SANTOS 2014), o projeto *SQuaRE*, ilustrado na Figura 4, reorganizou o conjunto dos 10 componentes que formam as normas ISO/IEC 9126 e ISO/IEC 14598,

---

<sup>6</sup>Uma norma de qualidade nasce, dentre outras, da necessidade de padronização, necessidade esta que pode ser identificada na indústria ou no meio acadêmico.

<sup>7</sup>A ISO é uma família de normas que tem caráter genérico, servindo de base para qualquer organização em qualquer ramo de atividade, que queira realizar o controle de qualidade dos produtos ou serviços oferecidos (Koscianski 2007).

mas não realizou mudanças radicais no material preexistente. O modelo hierárquico de qualidade proposto na norma 9126 continuou válido, assim como diversos aspectos organizacionais, propostos pela 14598.

Requisitos de Qualidade 2503n	Modelo de Qualidade 2501n	Avaliação 2504n
	Gerenciamento de Qualidade 2501n	
	Medições 2501n	

**Figura 4: Partes componentes da norma SQuaRE**

Como podemos observar na Figura 4, cada divisão é composta por um conjunto de documentos, referenciados da forma 250Xn (onde X varia entre 1 e 4) e trata de um assunto em particular, definidos de maneira sucinta, a seguir:

- Divisão de Gerenciamento da Qualidade: onde são definidos todos os termos que serão utilizados em todos os outros documentos e como está organizada a norma.
- Divisão do Modelo de Qualidade: onde é apresentado o modelo de qualidade e guias práticos sobre como utilizá-lo.
- Divisão de Medição de Qualidade: inclui um modelo de referência para medições da qualidade de um produto, definições matemáticas de medidas e orientações práticas para sua aplicação.
- Divisão de Requisitos de Qualidade: onde, os documentos que formam essa divisão ajudam na especificação de requisitos de qualidade.
- Divisão de Avaliação de Qualidade: composta por requisitos, recomendações e *guidelines* para avaliação de um produto de software.

É importante ressaltar também que a norma *SQuaRE* identifica três tipos de qualidade, que correspondem a diferentes situações envolvendo um produto de *software*:

- Qualidade Interna: é avaliada na representação estática do produto e, é exatamente a arquitetura interna que é levada em conta. A qualidade de organização do código ou a complexidade algorítmica são exemplos de critérios

que podem indicar, respectivamente, prováveis custos de manutenção e provável velocidade de execução.

- **Qualidade Externa:** considera o produto como uma caixa-preta. A realização de testes corresponde a verificar a qualidade externa. A qualidade externa é uma estimativa da qualidade em uso; ambas podem ser muito próximas, mas não são equivalentes.
- **Qualidade em Uso:** corresponde ao ponto de vista de um usuário. A qualidade em uso, quando se refere a um programa sendo executado, depende de diversos fatores como o hardware utilizado, o treinamento do usuário, as condições ambientes, as tarefas realizadas e etc.

## 2.2.2 Medidas de Qualidade

Segundo a (ISO 25000 2005), uma medida de qualidade de software é uma variável ao qual é atribuído um valor, como o resultado da medição. De acordo com (KOSCIANSKI E SOARES 2007), dentre as dificuldades para se definir medidas de qualidade de software, pode-se citar duas: a variedade de aspectos a considerar e a presença de muitos elementos intangíveis.

Para tanto, segundo (SCALET 1999), alguns critérios devem ser levados em consideração pelo avaliador ao elaborar as medidas. Tais critérios podem ser vistos na Tabela 1, adaptada de (KOSCIANSKI E SOARES 2007).

**Tabela 1: Critérios Qualitativos para Definição de Medidas**

<i>Significância</i>	<i>Os resultados obtidos devem agregar informação útil a avaliação da qualidade.</i>
Custo e Complexidade de Aplicação	O custo e a complexidade de uma medida devem ser compatíveis com a avaliação a ser realizada.
Repetibilidade	As medidas devem ser repetíveis, exceto algumas que dificilmente podem ser repetidas por envolverem condições não controláveis.
Objetividade	A opinião de pessoas envolvidas deve ser limitada ou totalmente evitada.

**Tabela 1 (Continuação): Critérios Qualitativos para Definição de Medidas**

<i>Significância</i>	<i>Os resultados obtidos devem agregar informação útil a avaliação da qualidade.</i>
Imparcialidade	As medidas devem ser imparciais, e deve se ter em mente que uma escolha não criteriosa de parâmetros pode levar a resultados diferentes e favorecer ou não um produto.
Evidência para Validação	As medidas devem prover evidências para sua validação. Por exemplo, se o resultado de uma avaliação é uma curva de tendência, é preciso prover informação a respeito do grau de confiabilidade dos valores apresentados.

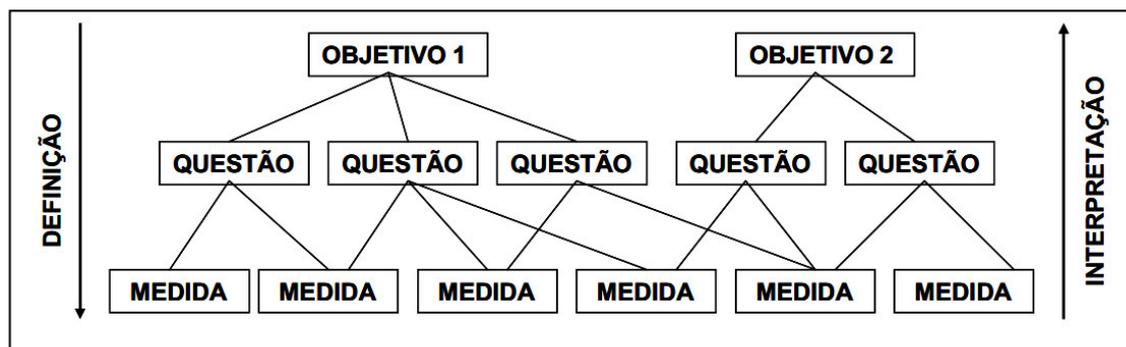
De acordo com (ROCHA et al 2012), ao definir medidas de qualidade de *software* é necessário também ter clareza sobre a adequação e utilidade destas, para que seja possível garantir que as informações obtidas com a sua coleta apoiem a tomada de decisão, respeitando os objetivos definidos para o sistema que está sendo avaliado. Para tanto, segundo (SANTOS 2014), é possível utilizar diferentes metodologias para a definição de medidas, dentre elas o GQM, o GQiM e o PSM.

O *Goal-Question-Metric* (GQM) é uma abordagem orientada a objetivos para a mensuração de produtos e processos de *software*. Essa abordagem é baseada na premissa de que as medições de *software* devem ser orientadas aos objetivos de medição, que por sua vez devem ser orientados aos objetivos da organização (BASILI et al. 1994).

Devido à dificuldade em se derivar objetivos de medição no GQM, surgiu, como auxílio, a abordagem *Goal-Question-Indicator-Metric* (GQiM). A metodologia é composta por dez passos e procura alcançar os objetivos de negócio preparando um plano de medições.

O *Practical Software Measurement* (PSM), por sua vez, é uma abordagem para especificar formalmente medidas de qualidade de *software* e como aplicá-las em um processo de medição (HUGHES, 2000).

Nesta dissertação de mestrado, a metodologia utilizada para a definição de medidas é o GQM. Essa metodologia é bastante difundida e utilizada devido à sua simplicidade. Segundo (ROCHA et al 2012), por ter uma estrutura *top-down*, utilizando o GQM é possível identificar as medidas a partir de questões associadas ao objetivo de medição, como ilustrado na Figura 5.



**Figura 5: A metodologia GQM retirada de (ROCHA et al 2012)**

Como se pode observar na Figura 5, o resultado da aplicação do GQM é uma estrutura composta de três níveis. No nível 1, ou nível conceitual, estão os objetivos, no nível 2, ou operacional, estão as questões e, no nível 3, ou quantitativo, encontram-se as medidas.

Ainda segundo (ROCHA et al 2012), com base em (SOLLINGEN e BERGOUTH 1999), após a definição dos objetivos, questões e medidas é necessário produzir um plano de medição e análise. Para isso, é fundamental definir, entre outros:

- Possíveis valores para as medidas;
- Métodos para a coleta das medidas;
- Responsáveis para a coleta das medidas;
- O momento de coleta;
- Procedimentos para a análise das medidas.

Como mencionado no Capítulo 1, o GQM é uma das técnicas mais adequadas quando o objetivo da medição é claro e específico, como é o caso deste trabalho, onde o objetivo é avaliar a confiança no funcionamento de sistemas ubíquos que executam em dispositivos móveis.

Por esse motivo, e por ser este trabalho uma continuidade de (SANTOS 2014), onde o GQM foi utilizado, essa metodologia foi a escolhida para a definição das medidas desta pesquisa e é apresentada com mais detalhes na próxima seção.

### 2.2.3 Goal-Question-Metric (GQM)

Segundo (BASILLI et al 1994), a metodologia GQM é dividida nas seguintes etapas: planejamento, definição, coleta de dados e interpretação de resultados, apresentadas em detalhes a seguir.

Na etapa de planejamento é necessário definir a equipe que participará do GQM, a área de melhoria e os projetos que farão parte da aplicação do método. O principal objetivo dessa fase é coletar todas as informações necessárias para uma avaliação de sucesso. Para isso, como mencionado anteriormente, é necessário estabelecer um time GQM, uma área de melhoria, um projeto de aplicação e o time do projeto.

Na fase de definição é preciso elucidar os objetivos do GQM, as questões a serem respondidas, além de definir e refinar as medidas.

A fase de definição de medidas é composta por:

- Definição do objetivo de medição, com objeto, propósito e foco;
- Entrevistas GQM com especialistas;
- Utilização dos resultados das entrevistas para derivar questões; e, por fim,
- Medidas.

Para a definição do objetivo de medição é utilizado o *template* definido por (BASILLI et al. 1994), apresentado na Tabela 2. De acordo com esse *template* devem ser definidos os objetos sob medição, os propósitos do objetivo, os focos de qualidade e o ponto de vista (quem mede o objeto).

**Tabela 2: Template para a definição do objetivo de medição (Basili et al 1994)**

Analisar	Os sistemas ubíquos que executam em dispositivos móveis
Com o propósito de	Avaliar
Com respeito à	Confiança no Funcionamento
Sob o ponto de vista do	Desenvolvedor e usuário

No caso deste trabalho de pesquisa, o objeto sob medição é o sistema ubíquo para dispositivo móvel, o propósito é avaliar e os focos de qualidade são os cinco atributos de

qualidade relacionados à confiança no funcionamento: Disponibilidade, Integridade, Confiabilidade, Manutenibilidade e Proteção. Por fim, o ponto de vista é o do usuário e o do desenvolvedor.

Segundo (MARTINS 2011), o objetivo de medição é definido em um nível abstrato, por isso, as questões são o refinamento do objetivo em um nível mais operacional e mais simples de interpretar. Logo, as questões devem ser definidas em um nível intermediário de abstração entre as medidas e o objetivo, para facilitar a interpretação dos dados coletados.

Após a definição do objetivo de medição, é necessário realizar as entrevistas GQM com o time estabelecido na fase de planejamento. Com isso, é possível identificar quais questões são relevantes com respeito ao objetivo e, quais as possíveis medidas capazes de respondê-las.

Por fim, nas fases de coleta e interpretação é necessário coletar os dados, com base nas medidas definidas e, interpretá-los buscando responder as questões estabelecidas.

A fase de coleta consiste em coletar, armazenar e processar todas as informações necessárias para calcular as medidas. Definindo, portanto, como será a obtenção dos dados que serão utilizados na medição. De acordo com (VAN SOLINGEN E BERGHOUT, 1999), durante a fase de coleta de dados, é importante observar quatro questões:

- Para uma determinada medida, quem deve coletar os dados necessários?
- Quando o dado deve ser coletado?
- Como o dado pode ser coletado mais eficientemente?
- Para quem deve ser entregue os dados coletados?

Em relação à interpretação, o objetivo nesse estágio é analisar os resultados obtidos na coleta dos dados e utilizá-los para interpretar as questões e os objetivos de medição definidos.

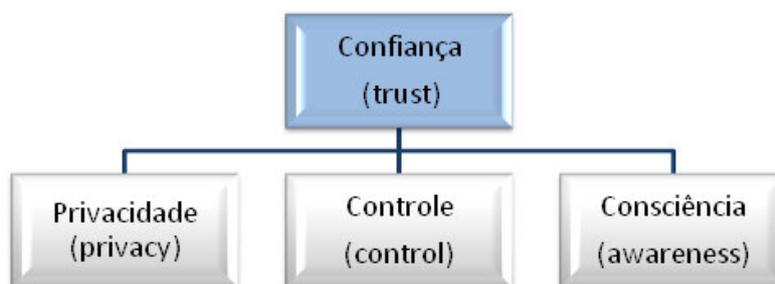
## **2.3 Confiança no Funcionamento de Sistemas de Software**

Apesar do trabalho realizado por (SANTOS 2014), as medidas do autor não são aplicáveis a todas as características levantadas durante o seu mapeamento sistemático. Destacando assim, a possibilidade e necessidade da definição de medidas para outras características, entre elas, a confiança (traduzida do inglês *trust*).

Diferentemente de características como disponibilidade, que possui uma definição mais clara e precisa na literatura (e.g., prover serviço sempre que requisitado (AVIZIENIS et al 2004) (STALLINGS 2012)), o termo confiança traz uma certa subjetividade em sua definição quando usado em diferentes áreas.

Para que as medidas de qualidade relacionadas a este trabalho possam realmente avaliar da maneira mais precisa possível essa característica, é necessário estabelecer de maneira objetiva e direta o que definimos como confiança. Para tanto, é necessário avaliar e respeitar uma série de conceitos estabelecidos nas áreas de qualidade de software, engenharia de software e redes de computadores que possibilitam estabelecer e guiar quais aspectos podem ser considerados como atributos da confiança do usuário em um sistema.

Segundo (SANTOS 2014) e (SCHOLTZ E CONSOLVO 2004), a confiança (traduzida do inglês *trust* em ambos os trabalhos), apresentada na Figura 6, pode ser vista como a crença do usuário que o sistema utilizará os seus dados pessoais corretamente sem causar nenhum prejuízo ou dano. Para tanto, os autores apontam como atributos importantes para a garantia dessa confiança, aspectos relativos à privacidade (traduzida do inglês *privacy*, representa o grau de sigilo das informações pessoais do usuário, seja para outros usuários do sistema ou terceiros), consciência (traduzida do inglês *awareness*, representa a capacidade do usuário de entender quais das suas informações são ser utilizadas pelo sistema e quando) e controle (traduzida do inglês *control*, representa a capacidade do usuário de controlar quais das suas informações podem ser utilizadas pelo sistema e quando podem).



**Figura 6: A confiança e seus atributos adaptada de (SANTOS 2014)**

Já para a norma ISO 25010, o termo confiança (traduzida do inglês *trust*), apresentada na Figura 7, representa o grau de confiança que um usuário ou qualquer outra parte interessada tem de que um produto ou sistema irá se comportar conforme o esperado. Ainda segundo a norma, a confiança deve ser vista como uma subcaracterística da característica satisfação (traduzida do inglês *satisfaction*) que representa, por sua vez, o grau

em que as necessidades dos usuários são satisfeitas, quando um produto ou sistema é utilizado em um contexto de uso específico.

Além da confiança, compõem o grau de satisfação, as subcaracterísticas conforto (traduzida do inglês *comfort*, representa o grau com o qual o usuário está satisfeito com o conforto físico), utilidade (traduzida do inglês *usefulness*, representa o grau de satisfação do usuário ao perceber que o sistema consegue perceber os seus objetivos) e prazer (traduzida do inglês *pleasure*, representa o grau em que um usuário obtém prazer ao cumprir as suas necessidades pessoais).

Diante desse cenário, o nosso primeiro passo é estabelecer a confiança como uma característica de qualidade, assim como satisfação em (ISO 25010) e, após isso, refiná-la em uma série de subcaracterísticas.



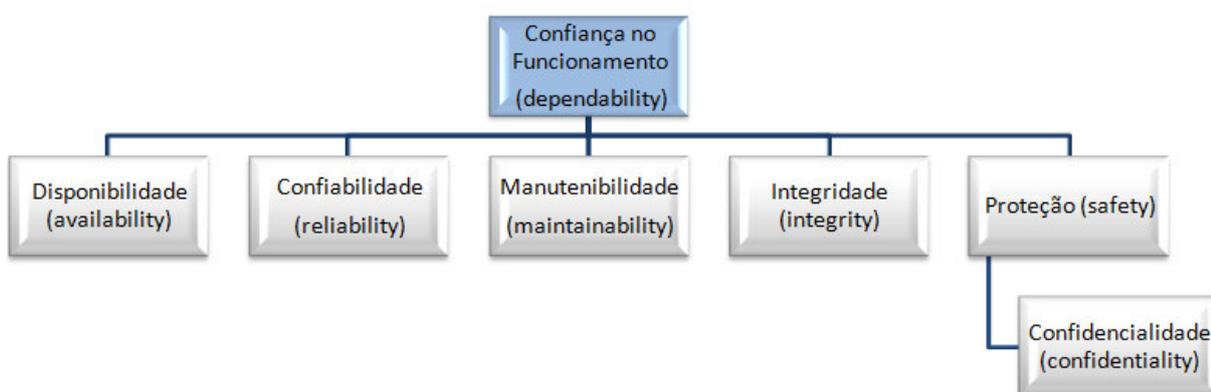
**Figura 7: A satisfação e seus atributos adaptada de (ISO 25010)**

Sendo assim, o restante desse texto é baseado no conceito de confiança no funcionamento (traduzido do inglês *dependability*), definido por (AVIZIENIS et al 2004) e ilustrado na Figura 8. Segundo o autor, o termo é proposto como representação dos esforços das comunidades de confiança no funcionamento e segurança (traduzida do inglês *security*) buscando unificar vários aspectos dessas duas áreas, servindo como uma espécie de guia de termos e conceitos. Assim, diante da precisão e aceitabilidade por parte da comunidade científica do termo estabelecido, é possível definir medidas de qualidade de software que avaliem a confiança do usuário no funcionamento do sistema, levando em consideração um conjunto de atributos mais amplo do que o proposto por (SANTOS 2014) e (SCHOLTZ e CONSOLVO 2004) e com uma maior granularidade do que o proposto pela (ISO 25010).

É importante destacar também que não faz parte das intenções deste trabalho discutir sobre as controvérsias geradas por parte da literatura (e.g., confiança no

funcionamento (*dependability*) é sinônimo de confiança (*trust*) (FOGG e TSENG 1999) (GOILAU 2003) (SOLNER 2012)), mas sim, fazer uso dos conceitos apresentados por (AVIZIENIS et al 2004) em benefício da área de qualidade de software e medições.

Assim sendo, a confiança avaliada com as medidas relativas a este trabalho representa a capacidade do usuário de poder justificadamente confiar no funcionamento de um sistema. Para tanto, essa confiança é justificada com base na confiabilidade (traduzida do inglês *reliability*), disponibilidade (traduzida do inglês *availability*), integridade (traduzida do inglês *integrity*), manutenibilidade (traduzida do inglês *maintainability*), proteção (traduzida do inglês *safety*) e confidencialidade (traduzida do inglês *confidentiality*) do mesmo.



**Figura 8: A confiança no funcionamento e seus atributos**

Segundo (AVIZIENIS et al 2004), a confiabilidade representa a habilidade de prestar serviço conforme o esperado de maneira contínua, a disponibilidade implica em prestar serviço correto sempre que requisitado, a integridade implica na ausência de alterações impróprias para o sistema, a manutenibilidade representa a habilidade de sofrer modificações e reparos, a proteção implica na ausência de consequências catastróficas para o usuário e o ambiente e, por fim, a confidencialidade implica na ausência de divulgação não autorizada de informação. Ainda segundo o autor é importante observar que a segurança (do inglês *security*) e a confiança no funcionamento possuem diversos atributos em comum.

Segundo (ROCHA 2010): “esses atributos podem ser tratados em conjunto desde que disponibilidade, confiabilidade e integridade sejam consideradas simultaneamente, a disponibilidade seja apenas para os usuários autorizados e que na definição de integridade o termo impróprio seja entendido como não autorizado”.

Neste trabalho, o agrupamento dos atributos mencionados por (AVIZIENIS et al

2004) é feito de modo que todos eles estejam associados, como subcaracterísticas, a mesma característica: a confiança no funcionamento. Com isso, levando em consideração o domínio desta pesquisa (sistemas ubíquos), aspectos como a proteção e confiabilidade do sistema passam a ser avaliados levando em consideração informações típicas de sistemas ubíquos (e.g., informações contextuais).

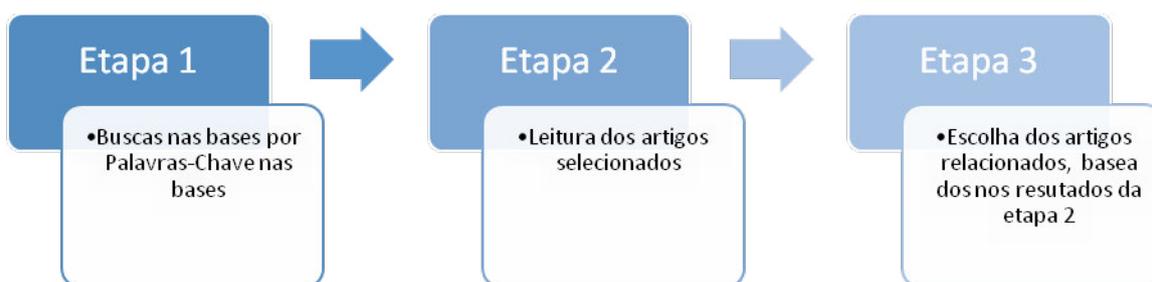
É importante ressaltar que, dependendo do sistema a ser desenvolvido, os atributos mencionados anteriormente podem ter pesos diferentes em relação a confiança no funcionamento. Além disso, é possível que, em certos sistemas, os atributos sejam vistos como antagonísticos (e.g., disponibilidade e proteção). Logo, é importante fazer uso de um conjunto de métodos ou técnicas para garantir que o sistema resultante seja de confiança.

## 2.4 Trabalhos Correlatos

Esta seção apresenta os trabalhos relacionados a esta pesquisa. Ao todo, quatorze artigos são analisados e comparados aos resultados desta dissertação.

### 2.4.1 Metodologia de Busca

O processo de identificação dos trabalhos relacionados, realizado nesta pesquisa, ocorre em três etapas, considerando as bases eletrônicas da IEEE, ACM e Springer. O processo de filtragem dos artigos é representado na Figura 9 e detalhado no restante desta seção.



**Figura 9: Processo de identificação dos trabalhos relacionados**

Como observado na Figura 9, a etapa 1 é caracterizada pela busca nas bases, utilizando *strings* de busca derivadas das palavras-chave apresentadas na Tabela 3. Estas, por sua vez, são derivadas do objetivo da proposta. Primeiro são identificados os objetos de interesse (características e medidas de qualidade), depois o propósito (avaliação da confiança no funcionamento) e, por fim, o contexto (sistemas ubíquos ou sistemas ubíquos

para dispositivos móveis). Durante a busca, foram considerados também sinônimos, traduções para o inglês e palavras alternativas.

**Tabela 3: Palavras-chave utilizadas para a busca de trabalhos**

<b>Objeto</b>	<b>Propósito</b>	<b>Contexto</b>
<i>Características</i>		
<i>Subcaracterísticas</i>		<i>Sistemas Sensíveis ao Contexto</i>
<i>Medidas</i>	<i>Avaliar a confiança no funcionamento</i>	<i>Sistemas Pervasivos</i>
<i>Métricas</i>		<i>Sistemas Ubíquos</i>
<i>Crítérios</i>		<i>Sistemas móveis</i>
<i>Atributos</i>		
<i>Fatores</i>		
<i>Modelo</i>		

O intervalo de tempo considerado é de 1991 até 2015, tendo como base para a escolha o surgimento do conceito de computação ubíqua (1991) até os trabalhos atuais. Foram então encontrados 121 artigos considerados relevantes, publicados em conferências e periódicos da área.

Ainda de acordo com a Figura 9, na segunda etapa, é realizada a leitura completa de cada um dos artigos, de onde é possível extrair: referência, problema abordado, contribuição ou inovação do artigo, semelhanças e divergências entre o artigo e o trabalho em desenvolvimento e, por fim, desafios encontrados.

Na terceira e última etapa, com base nas informações coletadas na etapa 2, foram selecionados 14 trabalhos considerados pertinentes a este, descritos de maneira resumida na Tabela 4.

Apesar de antigas, algumas das referências como (AVIZIENIS et al 2004) e (RANGANATHAN 2005), são de extrema importância pois, além de fundamentarem os trabalhos seguintes, trazem contribuições essenciais para o trabalho em desenvolvimento nesta proposta de dissertação, que vão além da fundamentação teórica, como, por exemplo, a taxonomia utilizada para a confiança no funcionamento apresentada em (AVIZIENIS et al 2004) ou as medidas apresentadas por (SCHOLTZ e CONSOLVO 2004) e (RANGANATHAN 2005).

**Tabela 4: Referência e Descrição dos Trabalhos Relacionados**

<b>Referência</b>	<b>Descrição Resumida</b>
<i>Scholtz e Consolvo, 2004</i>	<i>Propõem um framework para avaliação de aplicações ubíquas</i>
<i>AVIZIENIS et al., 2004</i>	<i>Propõem uma taxonomia para confiança no funcionamento (dependability).</i>
<i>Ranganathan, 2005</i>	<i>Propõe um benchmark para avaliação de medidas e atividades em ambientes da computação ubíqua.</i>
<i>Stajano, 2008</i>	<i>Aborda questões e desafios relacionados à segurança em sistemas ubíquos.</i>
<i>Jia, 2009</i>	<i>Propõe um framework para avaliação da confiança baseado em controle de acesso nos processos de interação humano computador.</i>
<i>Abi-Char, 2010</i>	<i>Propõe um framework para garantia de segurança e privacidade em sistemas sensíveis ao contexto.</i>
<i>Karvonen, 2010</i>	<i>Analisa diferentes aspectos da confiança humana que afetam o uso de sistemas ubíquos inteligentes de transporte. O objetivo é contribuir para as bases conceituais da confiança na interação humano-computador.</i>
<i>Karvonen, 2010</i>	<i>Analisa diferentes aspectos da confiança humana que afetam o uso de sistemas ubíquos inteligentes de transporte. O objetivo é contribuir para as bases conceituais da confiança na interação humano-computador.</i>
<i>Treck, 2011</i>	<i>Avalia uma metodologia qualitativa de avaliação dinâmica para o gerenciamento de confiança em sistemas ubíquos, baseado em simulações</i>
<i>ISO 25010</i>	<i>Propõe características, subcaracterísticas e medidas para a avaliação da qualidade em sistemas de software de maneira geral.</i>
<i>Lee e Yun, 2012</i>	<i>Propõem medidas para a avaliação da usabilidade em serviços ubíquos, tendo como base as interações e serviços.</i>
<i>Santos, 2013</i>	<i>Propõe um modelo de qualidade para avaliação de sistemas ubíquos.</i>
<i>Hofmman e Sollner, 2014</i>	<i>Propõem uma abordagem para aumentar a confiança durante o processo de desenvolvimento de sistemas ubíquos utilizando conceitos e idéias sobre teoria da confiança comportamental.</i>
<i>Sherchan et al., 2014</i>	<i>Apresentam a primeira revisão abrangente da literatura social e ciência da computação na confiança (trust) em redes sociais.</i>
<i>Liu et al., 2014</i>	<i>Apresentam um modelo que descreve o papel da personalização e controle na avaliação de custos e benefícios associados à divulgação de informações privadas dos usuários em sistemas pervasivos.</i>

## 2.4.2 Análise dos Trabalhos Correlatos e Desafios em aberto

A análise dos trabalhos correlatos leva em consideração os seguintes aspectos: as vantagens encontradas em cada um dos trabalhos, as desvantagens e as relações entre eles e esta pesquisa de mestrado. O objetivo da análise é, com base nos aspectos considerados, identificar quais são as características que influenciam na confiança no funcionamento de sistemas ubíquos executando em dispositivos móveis e qual a metodologia utilizada pelos autores para a sua avaliação. No caso da identificação de medidas de qualidade, como método de avaliação, é também analisada a adequação dessas medidas para o domínio em questão, os sistemas ubíquos.

A análise comparativa dos trabalhos correlatos é apresentada, de maneira sucinta, na Tabela 5.

Buscando alcançar o propósito geral desse trabalho, três questões, definidas por (SANTOS 2014) e adaptadas para esta pesquisa (para levar em consideração especificamente a característica de confiança no funcionamento), são consideradas durante a análise dos trabalhos relacionados. Mais informações encontram-se documentadas na Tabela 5 para auxiliar as respostas a estas questões.

1. Quais metodologias, técnicas ou procedimentos (e.g. modelo de qualidade, medidas de qualidade, framework) são utilizados para avaliar a confiança no funcionamento em sistemas ubíquos?

Para responder esta questão é necessário informara existência de diferentes métodos, procedimentos e metodologias existentes na área de qualidade de software que podem ser usados para a avaliação de atributos de qualidade. Dentre esses, é possível citar: modelos de qualidade, medidas de qualidade, framework de medidas, framework de características, processos, análise de requisitos e outros (SANTOS 2014).

Durante a análise dos trabalhos correlatos é possível identificar três metodologias distintas para a avaliação da confiança no funcionamento: framework de características, medidas e características, detalhadas a seguir:

- Framework de características – nesse caso, o trabalho apresenta as características e/ou subcaracterísticas de qualidade como uma lista de fatores

**Tabela 5: Comparação dos Trabalhos Correlatos**

Referência	Considera o domínio dos sistemas ubíquos	Avalia por meio de medidas de qualidade	As medidas são definidas seguindo formato padrão	Existe algum guia que auxilie a aplicação das medidas	Considera a disponibilidade como fator de impacto	Considera a confiabilidade como fator de impacto	Considera a manutenibilidade como fator de impacto	Considera a integridade como fator de impacto	Considera a proteção como fator de impacto
Scholtz e Consolvo, 2004	SIM	NÃO	NÃO	NÃO	NÃO	NÃO	NÃO	NÃO	SIM
Avizienis et al. 2004	NÃO	NÃO	NÃO	NÃO	SIM	SIM	SIM	SIM	SIM
Ranganathan, 2005	SIM	SIM	NÃO	NÃO	NÃO	SIM	NÃO	SIM	SIM
Stajano, 2008	SIM	NÃO	NÃO	NÃO	SIM	SIM	NÃO	SIM	SIM
Jia, 2009	SIM	NÃO	NÃO	NÃO	NÃO	SIM	NÃO	SIM	SIM
Abi-Char, 2010	SIM	NÃO	NÃO	NÃO	NÃO	SIM	NÃO	SIM	SIM
Karvonen 2010	SIM	NÃO	NÃO	NÃO	SIM	SIM	SIM	SIM	SIM
Treck 2011	SIM	NÃO	NÃO	NÃO	SIM	NÃO	NÃO	SIM	SIM
ISO 25010	NÃO	SIM	SIM	SIM	NÃO	NÃO	NÃO	NÃO	SIM
Lee e Yun, 2012	SIM	NÃO	SIM	NÃO	NÃO	SIM	NÃO	NÃO	SIM
Santos, 2013	SIM	SIM	SIM	NÃO	NÃO	NÃO	NÃO	NÃO	SIM
Hofmman e Sollner, 2014	SIM	NÃO	NÃO	NÃO	NÃO	SIM	NÃO	SIM	SIM
Sherchan et al. 2014	SIM	NÃO	NÃO	NÃO	NÃO	SIM	NÃO	NÃO	SIM
Liu et al. 2014	SIM	SIM	NÃO	NÃO	NÃO	SIM	NÃO	NÃO	SIM

- que devem ser avaliados em um sistema ubíquo (SCHOLTZ e CONSOLVO 2004) (LEE e YUN 2012) (JIA 2009) (ABI CHAR 2010).
- Medidas de Qualidade de Software – nesse caso, o trabalho apresenta uma lista de medidas organizadas dentro de características de qualidade (RANGANATHAN 2005) (ISO 25010) (SANTOS 2013) (SANTOS 2014).
- Características de Avaliação – nesse tipo de trabalho é apresentado um conjunto de características que podem ser consideradas em uma avaliação envolvendo sistemas ubíquos, mas não está explicitamente definido como a avaliação pode ser realizada (STAJENO 2008) (KARVONEN 2010) (TRECK 2011) (HOFMAN e SOLLNER 2014) (SCHERCHAN et al 2014) (LIU et al 2014).
- Quais características (atributos ou fatores) de qualidade são utilizadas para avaliar a confiança no funcionamento em sistemas ubíquos?

A partir dos conceitos definidos nos trabalhos de (AVIZIENIS et al 2004), (SCHOLTZ e CONSOLVO 2004) e (ISO 25000), outros trabalhos foram fundamentados (e.g., (LEE e YUN 2012), (LIU et al 2014)). Para o trabalho fruto desta dissertação, a definição adotada para confiança no funcionamento pode ser encontrada em (AVIZIENIS et al 2004).

Como mencionado na seção 2.3, a escolha da definição baseia-se na importância diante dos trabalhos relacionados. Além disso, os atributos considerados por (AVIZIENIS et al 2004) abrangem grande parte das possíveis características que impactam o usuário em relação a sua confiança no funcionamento do sistema, quando comparados aos demais autores e trabalhos analisados.

2. Existem medidas de qualidade de *software* para avaliar a confiança no funcionamento nesses sistemas? É necessário adaptá-las?

Em resposta a terceira questão, é possível encontrar medidas de qualidade de software ou informações que fornecem subsídios para a proposta de medidas nos trabalhos de (SCHOLTZ e CONSOLVO 2004), (RANGANATHAN 2005), (SANTOS 2013), (LEE e YUN 2012) e (ISO 25010).

Os principais problemas relacionados às medidas encontradas nesses trabalhos são a falta de um formato padrão para a definição destas e o foco específico de cada um dos trabalhos quanto aos atributos (ou características) considerados para a avaliação da confiança no funcionamento. Em geral, é possível perceber a restrição da avaliação apenas a um conjunto específico dos atributos de impacto, apontados no próprio trabalho, desconsiderando os demais.

De acordo com (SCHOLTZ e CONSOLVO 2004), a confiança pode ser avaliada considerando questões relativas à privacidade, controle das informações pessoais do usuário e consciência do usuário sobre o uso de suas informações por parte do sistema. As medidas definidas em seu trabalho são associadas somente a proteção (*safety*), não considerando, portanto, atributos como confiabilidade, disponibilidade e confiabilidade.

O mesmo pode ser aplicado para o trabalho encontrado em (RANGANATHAN 2005), porém, nesse caso, o foco das medidas definidas está na confiabilidade e proteção, desconsiderando os demais atributos. Além disso, as medidas definidas pelo autor não seguem uma especificação precisa (e.g., com nome, função de medição e método de coleta) e são apresentadas apenas com nome e descrição, não sendo possível coletá-las por falta de funções de medição.

Dessa forma, utilizando as medidas propostas por (RANGANATHAN 2005) (e.g., a taxa de erros de contexto inferidos), pode não ser possível obter uma medição repetível, pois a falta de função de medição pode possibilitar resultados com um nível de abstração e ambiguidade significativos. Além disso, segundo o autor, o conjunto de medidas definidas em seu trabalho auxilia a avaliação da confiança no funcionamento, porém não é o suficiente para o domínio dos sistemas ubíquos, acarretando na necessidade da definição de outras medidas.

O trabalho de (SANTOS 2014), por sua vez, possui as mesmas características do trabalho de (RANGANATHAN 2005) em relação a definição das medidas, as quais são apresentadas para avaliação da confiança, mas possuem apenas nome e descrição, pois a definição destas, na pesquisa do autor, é apresentada como um trabalho futuro.

Conforme observado na Tabela 5, não existem medidas para avaliar a confiança no funcionamento de sistemas ubíquos e todos os seus atributos (ou fatores de impacto), listados na Figura 8. É possível encontrar medidas que ainda estão incompletas (e.g., não possuem qualquer apoio na sua coleta ou interpretação) ou não avaliam todos os seus atributos. Diante desse cenário, esta dissertação apresenta medidas para esta característica que permitem coleta, interpretação e repetição (caso haja necessidade). Vale ressaltar ainda

que as medidas definidas nesta pesquisa de mestrado podem ser vistas como uma contribuição para a evolução do trabalho realizado por (SANTOS 2014).

Além deste, outros desafios de pesquisa e questões em aberto são identificados durante a revisão da literatura e são apresentados na Tabela 6. A resolução destas questões está fora do escopo deste trabalho, mas espera-se que o mesmo possa servir como um ponto de partida para os trabalhos futuros.

**Tabela 6: Questões e Desafios**

Referência	Questões em aberto / Desafios encontrados
Scholtz e Consolvo, 2004	<ul style="list-style-type: none"> <li>• Como garantir que o processamento das informações pessoais seja de fundamental importância para a confiança no sistema?</li> <li>• Como personalizar a privacidade de acordo com o usuário e/ou dispositivo?</li> <li>• Como garantir que um usuário intruso não interfira nas informações de outro?</li> <li>• Como equilibrar o <i>trade-off</i> entre conhecimento de informações e privacidade?</li> </ul>
Nixon, 2005	<ul style="list-style-type: none"> <li>• Como garantir controle de acesso, gerenciar identidades e lidar com os aspectos sociais e legais em ambientes inteligentes?</li> <li>• Como prover interações seguras em ambientes inteligentes?</li> <li>• Como estabelecer abertura e preservar a privacidade em ambientes inteligentes?</li> </ul>
Ranganathan, 2005	<ul style="list-style-type: none"> <li>• Quais as características em comum aos ambientes ubíquos e outros ambientes que devem ser levadas em consideração para medição da segurança e privacidade nos ambientes inteligentes?</li> <li>• Como expressar a probabilidade do contexto ter sido deduzido de maneira correta?</li> </ul>
Stajano, 2008	<ul style="list-style-type: none"> <li>• Como expressar erros de acurácia com relação ao contexto?</li> <li>• Como identificar os pontos fracos da segurança do sistema?</li> <li>• Quais as partes mais valiosas do sistema?</li> <li>• Onde um possível ataque causaria dano máximo?</li> <li>• Quais as motivações para atacar o sistema?</li> <li>• Como sobrecarregar o sistema ao ponto de gerar uma falha?</li> <li>• É possível efetuar um ataque em grande escala, atingindo vários nós do sistema?</li> </ul>
Jia, 2009	<ul style="list-style-type: none"> <li>• Como garantir que uma entidade maliciosa não tenha acesso a informações que não são permitidas?</li> <li>• Como garantir que entidades maliciosas não manipulem informações?</li> <li>• Como garantir que uma entidade maliciosa não aja fornecendo informações erradas para os usuários?</li> <li>• Como garantir que comportamentos conflitantes entre diferentes entidades do sistema não gerem uma falha de segurança?</li> </ul>

## 2.5 Considerações Finais

Nesse capítulo foi apresentada uma visão geral sobre as áreas de pesquisa relacionadas a este trabalho: Sistemas Ubíquos, Confiança no Funcionamento de Sistemas de Software e Qualidade de Software.

Com o levantamento literário feito, é possível identificar a carência de medidas aplicáveis para o domínio dos sistemas ubíquos. Essa carência pode ser suprida por meio da definição de novas medidas, a qual pode ser realizada com o uso da metodologia GQM.

Com base no que foi apresentado nas seções anteriores deste capítulo e com a análise comparativa apresentada na Tabela 5, é possível observar como contribuição dos trabalhos relacionados, a definição da taxonomia utilizada para a confiança no funcionamento e o auxílio para a definição de medidas, pois é possível extrair de alguns trabalhos medidas que podem ser adaptadas.

As desvantagens concentram-se na maior parte em medidas definidas sem um padrão específico. Além de ser possível perceber que alguns dos trabalhos desconsideram características particulares dos sistemas ubíquos (AVIZIENIS et al 2004) (JIA 2009).

A principal contribuição deste trabalho está, portanto, em identificar e propor medidas de qualidade de *software* para a avaliação da confiança no funcionamento que reflitam as características dos sistemas ubíquos e considerem os atributos da definição de confiança adotada. Estas medidas são apresentadas no próximo capítulo.

## 3 AS MEDIDAS PARA CONFIANÇA NO FUNCIONAMENTO

Este capítulo apresenta 25 medidas de qualidade de *software* para a avaliação da confiança no funcionamento em sistemas ubíquos que executam em dispositivos móveis. Para a definição dessas medidas, a metodologia *Goal-Question-Metric* (GQM) é utilizada. As Seções 3.1, 3.2, 3.3 e 3.4 apresentam, respectivamente, as fases de planejamento, definição, coleta e interpretação, sendo estas correspondentes às etapas do GQM. Por fim, a Seção 3.5 apresenta as considerações finais sobre as medidas apresentadas.

### 3.1 Planejamento

Como mencionado no Capítulo 2, a medição de *software* é uma avaliação quantitativa que permite o melhor entendimento sobre os processos e produtos de *software*. Segundo (BASILLI et al. 2002) (ROCHA et al. 2012), realizando medições é possível auxiliar o planejamento, controle e melhoria de processos e produtos.

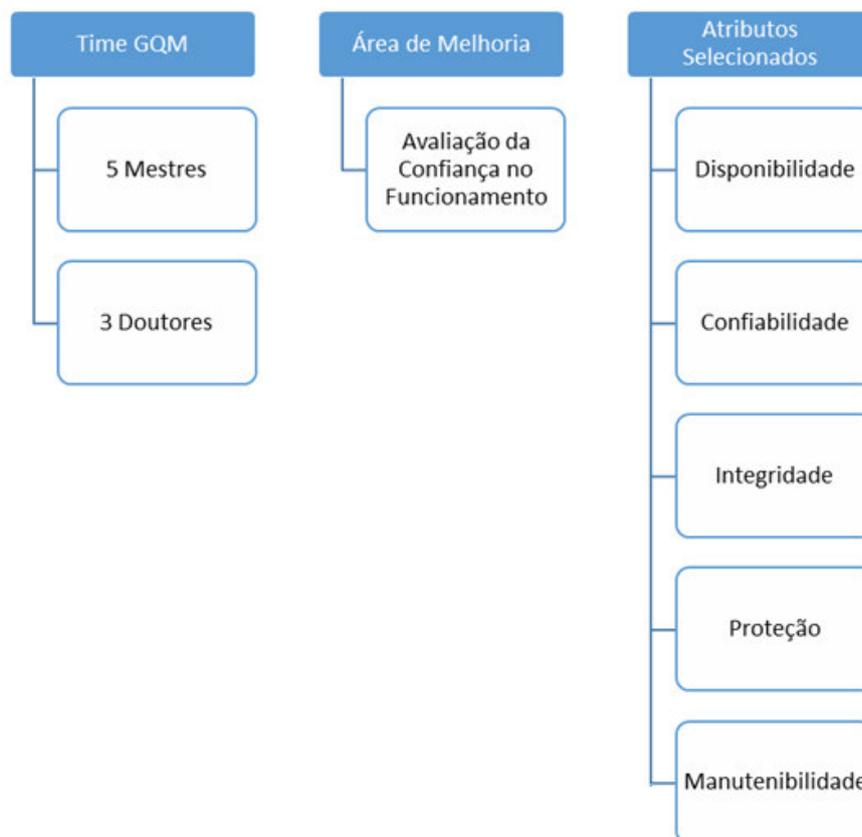
Diante desse cenário, dando continuidade ao trabalho de (SANTOS 2014), esta dissertação também utiliza a metodologia *Goal-Question-Metric* (GQM) para apoiar a definição do conjunto de medidas de qualidade de *software* apresentado. Assim como no trabalho do autor, o objetivo principal da fase de planejamento GQM desta pesquisa é estabelecer um time GQM, uma área de melhoria, um projeto de aplicação e o time do projeto (i.e., todas as pessoas envolvidas e interessadas no processo de definição).

Para este trabalho, o time GQM foi definido de acordo com as áreas de interesse apresentadas no Capítulo 2. Foram selecionados especialistas com conhecimento nas áreas de sistemas ubíquos, qualidade de *software* e confiança em sistemas de *software*. Um participante é considerado especialista se possui mestrado (completo) ou doutorado (completo ou em andamento) em pelo menos uma das áreas de interesse.

Além disso, com base na motivação e objetivos desta pesquisa, foi possível estabelecer uma área de melhoria: a avaliação da confiança no funcionamento em cenários de sistemas ubíquos que executam em dispositivos móveis. O projeto de aplicação e o time de projeto não foram definidos, pois, assim como em (SANTOS 2014), não foi considerado um projeto ou sistema específico, mas sim atributos de qualidade dos sistemas

ubíquos (e.g., disponibilidade, confiabilidade, integridade, proteção, confidencialidade e manutenibilidade).

A etapa de planejamento é ilustrada na Figura 10.



**Figura 10: Planejamento do GQM**

Após a etapa de planejamento, deu-se início a fase de definição, segunda etapa do GQM.

## 3.2 Definição

Como apresentado no Capítulo 2, a fase de definição de medidas é composta por:

- Definição do objetivo de medição, com objeto, propósito e foco;
- Entrevistas GQM com especialistas;
- Utilização dos resultados das entrevistas para derivar questões; e, por fim,
- Definir e (ou) propor medidas.

Cada um desses componentes é apresentado em detalhes nas subseções a seguir (3.2.1 a 3.2.4), de acordo com contexto de pesquisa deste trabalho.

### 3.2.1 Objetivo de medição

Para a definição do objetivo de medição é utilizado o *template* definido por (BASILLI et al. 2002), apresentado no Capítulo 2.

No caso dessa pesquisa, o **objeto** sob medição é o sistema ubíquo que executa em dispositivos móveis, o **propósito** é avaliar e os **focos** de qualidade são os cinco atributos de qualidade selecionados anteriormente: Disponibilidade, Integridade, Confiabilidade, Manutenibilidade e Proteção. Vale ressaltar que o ponto de vista é o do usuário e o do desenvolvedor.

Assim como indica (BASILI et al 2002), após a definição do objetivo de medição, é necessário realizar as entrevistas GQM com o time estabelecido na fase de planejamento. Essas entrevistas são fundamentais para identificar quais são as questões relevantes, com respeito ao objetivo e, quais as possíveis medidas capazes de respondê-las.

No total, oito especialistas participaram das entrevistas GQM, sendo estas realizadas de maneira individual. Os especialistas podem ser classificados da seguinte maneira:

- 2 mestrados e 3 doutorandos na área de sistemas ubíquos
- 2 mestres e 1 doutorando na área de segurança e confiança.

A individualidade dos encontros ocorreu, pois as reuniões dependiam da disponibilidade dos entrevistados, não sendo possível entrevistar dois ou mais especialistas de maneira simultânea.

Para cada especialista as atividades<sup>8</sup> realizadas foram:

- apresentação do problema, motivação e objetivos da pesquisa;
- análise da lista de medidas encontradas na literatura (ou propostas com base nas medidas encontradas); e,
- seleção de medidas, adaptação e (ou) sugestões de medidas, por parte dos especialistas.

Para cada foco de qualidade apresentado foi utilizado um *abstract sheet* (Tabela 7) com o intuito de facilitar as entrevistas e documentação das medidas.

---

<sup>8</sup>Essas atividades estão em conformidade com o que sugere (BASILI et al 2002) e com o estudo realizado por (SANTOS 2014).

**Tabela 7: Abstract Sheet**

<p style="text-align: center;"><b>Foco de qualidade</b></p> <p>Qual é o conhecimento atual dos especialistas com respeito ao foco de qualidade?</p>	<p style="text-align: center;"><b>Fatores de variação</b></p> <p>Quais características particulares dos sistemas ubíquos os especialistas acham que influenciam esse foco de qualidade?</p>
<p style="text-align: center;"><b>Hipóteses de <i>baseline</i></b></p> <p>Quais são as possíveis medidas para mensurar o objeto de estudo de acordo com os especialistas? Qual é o conhecimento atual dos especialistas com respeito a essas medidas?</p>	<p style="text-align: center;"><b>Impacto nas hipóteses de <i>baseline</i></b></p> <p>Quais tipos de dependências existem entre as medidas e os fatores de variação?</p>

### 3.2.2 Entrevistas

Como base no modelo de *abstract sheet* apresentado na subseção 3.2.1, para guiar as entrevistas, alguns tópicos foram levados em consideração e apresentados aos especialistas, são eles:

- **Foco de Qualidade:** para o entrevistado, o que significa o(s)foco(s) de qualidade especificado(s)? Por exemplo, se o foco de qualidade definido foi *Confiabilidade*, então o que significa *Confiabilidade* para o entrevistado?
- **Hipótese de *baseline*:** Para cada foco de qualidade declarado, quais as possíveis medidas para avaliá-lo? E quais os valores esperados para cada medida?
- **Fatores de Variação:** Quais fatores específicos do tipo de sistema avaliado têm possivelmente um impacto nos focos de qualidade? Seguindo a linha dos exemplos anteriores, para o entrevistado, sensibilidade ao contexto pode ter impacto sobre a *Confiabilidade*? Se sim, assim como nas hipóteses de *baseline*, quais das medidas apresentadas ou outras possíveis medidas estão relacionadas a este impacto?

- **Impacto na Hipótese de *baseline*:** Para cada fator de variação declarado: para o entrevistado, qual é a estimativa do impacto do(s)fator(es) de variação no foco de qualidade?

Com base nesse cenário e nos cinco atributos da confiança no funcionamento, os *abstract sheets* das Tabelas 8, 9, 10, 11, 12 e 13 foram gerados.

**Tabela 8: *Abstract Sheet* – Disponibilidade**

<b>Objeto</b>	<b>Propósito</b>	<b>Foco de Qualidade</b>	<b>Ponto de Vista</b>
Sistemas Ubíquos	Avaliar	Disponibilidade	Especialistas, Desenvolvedores e Usuários
<b>Foco de Qualidade</b>		<b>Fatores de Variação</b>	
1 – disponibilidade 2 – adaptações durante o uso (preoces ou tardias) 3 – grau de funcionamento da rede: falhas por tempo de uso		1.1 - tempo médio entre de falhas e tempo médio de reparo 1.2 – tempo que demora pra se adaptar após uma mudança de contexto relevante para o provimento do serviço 1.3 - tipo de rede, conexão de rede, handoff	
<b>Hipóteses de <i>Baseline</i></b>		<b>Impacto dos fatores nas Hipóteses de <i>Baseline</i></b>	
1 – quanto mais próximo de 1, melhor 2 - quanto menor (mais rápido), melhor 3 – quanto menor o número de falhas, melhor		1.1 - quanto maior for a razão entre o tempo médio entre falhas e a soma do tempo médio entre falhas e o tempo médio para reparo, melhor para o grau de disponibilidade 1.2 - quanto menor o tempo para se adaptar, melhor 1.3 - quanto menos influência do tipo de rede e conexão e quanto menor o tempo de <i>handoff</i> , melhor	

Com relação ao primeiro *abstract sheet*, foi analisada a capacidade do sistema de prover serviço sempre que requisitado. Para tanto, os principais aspectos levantados pelos

especialistas, como fatores de variação, foram a mobilidade dos sistemas ubíquos e as adaptações provenientes de mudanças contextuais.

Segundo os especialistas, as mudanças relacionadas a adaptação devem ocorrer de maneira rápida o suficiente para que nenhum atraso possível seja entendido por parte do usuário como uma indisponibilidade do sistema. Diante desse cenário, os especialistas apontaram preocupações relativas ao tempo de adaptação, indicando que quanto menor for, melhor, e ao tipo de conexão de rede utilizada, indicando que a influência da rede ou da troca de rede no provimento de serviço deve ser a menor possível.

Além disso, os especialistas apontaram a medida clássica do grau de disponibilidade (i.e., envolvendo o tempo médio entre falhas e o tempo médio de reparo) como sendo importante para o foco de qualidade em questão, baseado na classificação de falhas apresentada por (ROCHA 2010).

**Tabela 9: Abstract Sheet – Integridade**

<b>Objeto</b>	<b>Propósito</b>	<b>Foco de Qualidade</b>	<b>Ponto de Vista</b>
Sistemas Ubíquos	Avaliar	Integridade	Especialistas, Desenvolvedores e Usuários
<b>Foco de Qualidade</b>		<b>Fatores de Variação</b>	
1 – existência de mecanismos de integridade para os dados transmitidos e para os dados armazenados		1.1, 2, 3 – natureza distribuída dos sistemas ubíquos	
2 – existência de mecanismos de integridade com verificação		1.1, 2, 3 – comunicação com componentes externos	
3 – existência de mecanismos de integridade com verificação e correção			
<b>Hipóteses de <i>Baseline</i></b>		<b>Impacto dos fatores nas Hipóteses de <i>Baseline</i></b>	
1, 2, 3 – quanto mais, melhor		1.1 – quanto mais distribuído for o sistema, mais abaixo na pilha de protocolo está a responsabilidade pela garantia de integridade. (sendo responsabilidade principal da camada de enlace)	
		1.2, 1.3 - quanto mais troca de informações entre as partes envolvidas, maior necessidade do uso de mecanismos de integridade baseados em verificação e/ou correção	

O segundo *abstract sheet*, por sua vez, avalia o foco de qualidade “Integridade”. Para esse foco, os especialistas levantaram como principal fator de variação a natureza distribuída dos sistemas ubíquos. Segundo eles, o impacto desse fator no foco de qualidade é mínimo, pois nesse tipo de sistema a ausência de alterações impróprias não é responsabilidade da camada de aplicação.

Porém, considerando cenários onde a camada de rede não está funcionando conforme o especificado, é importante levar em consideração a existência ou não de mecanismos de integridade. Para tanto, parte dos especialistas apontaram a necessidade de analisar se existem mecanismos de integridade com verificação e (ou) correção. Diante desse cenário, quanto mais mecanismos existentes, melhor.

**Tabela 10: Abstract Sheet – Manutenibilidade**

<b>Objeto</b>	<b>Propósito</b>	<b>Foco de Qualidade</b>	<b>Ponto de Vista</b>
Sistemas Ubíquos	Avaliar	Manutenibilidade	Especialistas, Desenvolvedores e Usuários
<b>Foco de Qualidade</b>		<b>Fatores de Variação</b>	
1.1 – complexidade para a manutenção de sistemas ubíquos, com base no tempo  1.2 – complexidade de evolução de código em sistemas ubíquos, com base no tempo  1.3 – existência ou não de documentação associada ao sistema		1.1, 1.2 – natureza distribuída dos sistemas ubíquos (apenas um componente pode precisar de manutenção e impactar o sistema todo)  1.1, 1.2 – não conhecer todos os possíveis comportamentos do sistema em tempo de desenvolvimento pode impedir que aspectos importantes sejam documentados	
<b>Hipóteses de <i>Baseline</i></b>		<b>Impacto dos fatores nas Hipóteses de <i>Baseline</i></b>	
1 – quanto menor, melhor 2 – quanto menor, melhor 3 – quanto maior, melhor		1.1– quanto maior for a dependência de componentes externos, maior pode ser o tempo para manutenção e/ou evolução  1.2 – quanto mais documentações existentes (seja do sistema ou de componentes externos dos quais o sistema depende), melhor	

Com relação ao terceiro *abstract sheet*, foi analisada a capacidade do sistema de sofrer alterações e reparos. Para tanto, os especialistas destacaram que a natureza distribuída dos sistemas ubíquos impõe que a manutenibilidade do sistema como um todo seja influenciada pela manutenibilidade de suas partes.

Além disso, como podem existir aspectos comportamentais do sistema que só deverão ser conhecidos no momento do uso, alguns aspectos da documentação podem ser comprometidos.

Segundo os especialistas, a manutenibilidade pode ser vista como o esforço em função do tempo para a realização de atividades de manutenção e evolução. Com isso, melhorias podem ser constatadas quando os tempos necessários para cada atividade forem comparados a tempos anteriores utilizados para as mesmas atividades.

A existência de documentação associada a toda e qualquer parte do sistema pode ser vista como um atenuante para o esforço em função do tempo. Além disso, a dependência de componentes externos deve trazer junto a si documentação acessível sobre aquilo que é utilizado.

**Tabela 11: *Abstract Sheet* – Confidencialidade**

<b>Objeto</b>	<b>Propósito</b>	<b>Foco de Qualidade</b>	<b>Ponto de Vista</b>
Sistemas Ubíquos	Avaliar	Confidencialidade	Especialistas, Desenvolvedores e Usuários
<b>Foco de Qualidade</b>		<b>Fatores de Variação</b>	
1 – controle do usuário sobre suas informações utilizadas pelo sistema 1 – entendimento do usuário sobre suas informações utilizadas pelo sistema		1.1 – uso de informações contextuais 1.2 – uso de informações obtidas de maneira implícita 1.3 – autonomia (falta de interação explícita do usuário com o sistema)	
<b>Hipóteses de <i>Baseline</i></b>		<b>Impacto dos fatores nas Hipóteses de <i>Baseline</i></b>	
1– quanto maior, melhor 2 - quanto maior, melhor		1.1, 1.2– quanto maior o uso de informações contextuais, ou obtidas de maneira implícita, ou sem necessidade de interação explícita, mais meios o usuário deve ter para estar ciente de todas essas informações e controlar o uso delas por parte da aplicação (e.g., opção para alterar configurações do sistema)	

Para o foco que qualidade “Confidencialidade” (levantado por 3, dos 8 especialistas), um quarto *abstract sheet* foi gerado, com o objetivo de avaliar a capacidade do sistema de não realizar divulgação de informação não autorizada. Para tanto, o principal fator de variação apontado pelos especialistas envolve o uso de informação contextual.

Segundo os especialistas, o usuário precisa ter controle sobre as suas informações que são utilizadas pelo sistema. Segundo eles, devem haver mecanismos que possibilitem ao usuário alterar essas configurações ou, caso isso comprometa o provimento do serviço, antes de utilizá-lo pela primeira vez o usuário precisa estar ao menos estar ciente de todas as informações que o sistema passará a capturar e fazer uso.

Além das entrevistas, é possível encontrar na literatura o trabalho de (SCHOLTZ e CONSOLVO 2004) que colabora para todos os pontos levantados para esse foco de qualidade.

**Tabela 12: Abstract Sheet – Proteção**

<b>Objeto</b>	<b>Propósito</b>	<b>Foco de Qualidade</b>	<b>Ponto de Vista</b>
Sistemas Ubíquos	Avaliar	Proteção	Especialistas, Desenvolvedores e Usuários
<b>Foco de Qualidade</b>		<b>Fatores de Variação</b>	
1 –entendimento do usuário sobre suas informações utilizadas pelo sistema  2 – políticas de segurança		1.2 – uso de informações contextuais  1.2 – uso de informações obtidas de maneira implícita  1.3 – autonomia (falta de interação explícita do usuário com o sistema)	
<b>Hipóteses de <i>Baseline</i></b>		<b>Impacto dos fatores nas Hipóteses de <i>Baseline</i></b>	
1- quanto maior, melhor  2- quanto mais presentes, melhor		1.1, 1.2, 1.3– quanto maior o uso de informações contextuais, ou obtidas de maneira implícita, ou sem necessidade de interação explícita, mais meios o usuário deve ter para controlar o uso delas por parte da aplicação (e.g., opção para alterar configurações do sistema)  1.3 – quanto maior a autonomia do sistema, mais necessário é o uso de políticas de segurança para conceder permissão as partes envolvidas no provimento do serviço	

O *abstract sheet* relacionado ao foco de qualidade “Proteção” possui diversas informações em comum com o foco de qualidade “Confidencialidade”. Para um dos especialistas entrevistados, apenas um *abstract sheet* foi gerado para esses dois focos.

Além das informações apresentadas no *abstract sheet*, para o foco “Proteção”, foram levantadas questões sobre a presença ou não de políticas de segurança. Segundo os especialistas, a natureza autônoma dos sistemas ubíquos pode gerar situações onde a existência de algum tipo de política possibilite a não existência de situações de perigo para o usuário e suas informações (e.g., login feito de maneira implícita, utilizando informações providas pelo dispositivo do usuário, concede permissão baseada em autenticação indireta).

**Tabela 13: Abstract Sheet – Confiabilidade**

<b>Objeto</b>	<b>Propósito</b>	<b>Foco de Qualidade</b>	<b>Ponto de Vista</b>
Sistemas Ubíquos	Avaliar	Confiabilidade	Especialistas, Desenvolvedores e Usuários
<b>Foco de Qualidade</b>		<b>Fatores de Variação</b>	
1 - precisão do serviço provido 2 - recall do serviço provido 3 - acurácia do serviço provido 4 - corretude da captura contexto 5 - corretude da adaptação 6 - influência do sistema sobre componentes externos utilizados no provimento do serviço mas não exclusivo dele (e.g., bateria do celular) 7 - influência de componentes externos sobre o provimento do serviço (e.g., baixo nível de bateria pode comprometer o ativamento de algum sensor utilizado pelo sistema)		1.1 – uso de informação contextual 1.2 – autonomia no provimento do serviço 1.3 – variedade de informações contextuais utilizadas pelo sistema 1.4 – variedade de informações contextuais visíveis ao sistema 1.5 – uso de componentes externos	

**Tabela 13 (Continuação): *Abstract Sheet* – Confiabilidade**

<b>Objeto</b> Sistemas Ubíquos	<b>Propósito</b> Avaliar	<b>Foco de Qualidade</b> Confiabilidade	<b>Ponto de Vista</b> Especialistas, Desenvolvedores e Usuários
<b>Hipóteses de <i>Baseline</i></b>  1 - quanto maior, melhor 2 - quanto maior, melhor 3 - quanto maior, melhor 4 - quanto maior, melhor 5 - quanto maior, melhor 6- quanto menor, melhor 7 - quanto menor, melhor		<b>Impacto dos fatores nas Hipóteses de <i>Baseline</i></b>  1.1, 1.2, 1.3, 1.4 – quanto maior a capacidade do sistema de discernir entre as informações contextuais acessíveis a ele, utilizadas no provimento do serviço, e as informações contextuais visíveis, porém não necessárias no provimento do serviço, melhor precisão, recall e corretude. (informações relevantes vs informações não relevantes)	

O sexto e último *abstract sheet* tem como foco de qualidade a capacidade do sistema de funcionar conforme o especificado.

Segundo os especialistas, a autonomia dos sistemas ubíquos, que pode ser alcançada com o uso da sensibilidade ao contexto, exerce forte influência sobre a confiabilidade. Além disso, componentes externos, como bateria, foram citados por especialistas como fatores que influenciam no comportamento do sistema, pois o *Android* altera o comportamento do sistema dependendo do nível da bateria e, com isso, pode afetar o comportamento das aplicações.

É importante ressaltar que, assim como em (SANTOS 2014), as informações documentadas nos *abstract sheets* (Tabelas 8 a 13) podem ou não se transformar em medidas de qualidade. Algumas delas, assim como apontado no trabalho da autora, podem apenas servir como um meio para interpretar os resultados encontrados pela coleta de dados.

### 3.2.3 Derivando as Questões

Com base nas informações obtidas com a revisão da literatura e com as entrevistas GQM foi possível derivar as seguintes questões:

Q1.1 - Qual o grau de confiabilidade do serviço provido pelo sistema?

Q1.2 – Qual o grau de consciência e controle do usuário sobre suas informações utilizadas pelo sistema?

Q1.3 - Qual o grau de disponibilidade do serviço provido pelo sistema?

Q1.4 - Qual o grau de integridade dos dados armazenados ou transmitidos pelo sistema?

Q1.5 - Qual o esforço necessário para realizar manutenção ou evolução do sistema?

A questão Q1.1 busca avaliar se o sistema funciona conforme o especificado. A questão Q1.2, por sua vez, busca identificar se o usuário é ciente do uso de suas informações pessoais por parte do sistema e se tem controle sobre o uso destas. Com isso, é possível obter informações úteis para entender se o sistema pode levar a um estado no qual a vida humana, a saúde, a propriedade ou o ambiente está em perigo e se há proteção da informação contra acessos não autorizados. A questão Q1.3 busca identificar se o sistema está disponível sempre que requisitado. A questão Q1.4 busca identificar a ausência de alterações inadequadas no sistema. Por fim, a questão Q1.5 tem por objetivo avaliar o grau de eficácia e eficiência com o qual o sistema pode ser modificado.

O conjunto de medidas proposto com base nas questões derivadas das entrevistas GQM e demais informações documentadas durante as reuniões é apresentado na subseção 3.2.4.

### **3.2.4 Medidas**

As medidas resultantes seguem o formato da norma ISO 25000 e são compostas por identificador, nome, definição, função de medição e método de coleta. Apresentada no Capítulo 2, essa norma é referência quando se trata de medidas de qualidade de software.

O conjunto resultante das medidas apresentadas neste trabalho é descrito nas Tabelas 15 a 19. Nestas tabelas estão presentes:

- Identificador: sigla única capaz de identificar a medida;
- Nome: nome da medida;
- Definição: descrição da medida;
- Função de Medição: fórmula utilizada para a medição;
- Interpretação: intervalo de valores esperados ou aceitáveis;
- Método de Coleta: indicação do método ou procedimento a ser utilizado para a coleta da medida;

- **Referência:** a referência da medida pode ser algum trabalho encontrado na literatura (pois a medida já se encontrava formatada de acordo com a [ISO/IEC 25000]), ou uma adaptação (medida encontrada na literatura e que possui apenas nome e descrição, ou nome e função de medição, necessitando, portanto, de uma adaptação). Por fim, fruto de uma entrevista GQM (sugestão de um especialista ou definição com base em outras medidas); e
- **Subcaracterística associada:** atributo(s) da confiança no funcionamento que está(ão) diretamente associado(s) a medida.

Para cada uma das cinco questões derivadas na etapa de definição do GQM, seguem nas Tabelas A, B, C, D e E, as medidas propostas capazes de respondê-las.

Onde:

- *Retirada* indica que a medida foi extraída do trabalho referenciado onde também é apresentada como uma medida de qualidade.
- *Adaptada* indica que a medida foi extraída de um trabalho onde é apresentada como uma medida, porém, não formatada seguindo algum padrão específico ou, não definida para o domínio em questão neste trabalho.
- *Baseada* indica que a medida foi extraída de um trabalho onde apenas existem conceitos que podem vir a formalizar uma possível medida, segundo os autores.
- *Indicada* em reunião GQM indica que a medida foi sugestão dos especialistas envolvidos no GQM, usando como base seus conhecimentos e experiências na área.

É importante também ressaltar que algumas medidas fazem uso de positivos (P), falsos positivos (FP), negativos (N) e falsos negativos (FN) em suas funções de medição, conforme mostrado na Tabela 14.

**Tabela 14: Sobre os Resultados das Funções de Medições**

	Informação contextual visível ao sistema e importante (relevante) para o provimento do serviço	Informação contextual visível ao sistema, porém, não necessária (não relevante) para o provimento do serviço
Percebida pelo sistema como algo importante para o provimento do serviço	P (positivo)	FP (falso positivo)
Percebida pelo sistema como algo não importante para o provimento do serviço	FN (falso negativo)	N (negativo)

As definições expressas acima para P, FP, N e FN são válidas para todas as medidas nas quais esses critérios aparecem, bem como o significado de relevante e não relevante, mencionado na descrição das medidas e associados na Tabela 14 às informações contextuais utilizadas pelo sistema.

Tabela 15: As 11 Medidas para a questão Q1.1

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GrP</b>	Grau de Precisão	Representa a fração de contextos sentidos ou inferidos que são relevantes para o sistema	$\frac{P}{P + FP}$ P = Positivos FP = Falsos Positivos	Quanto mais próximo de 1, melhor	Automática	Adaptada de [Noury 2007]	Confiabilidade
<b>GrR</b>	Grau de <i>Recall</i> (abrangência ou cobertura)	Representa a fração de contextos relevantes para o sistema que foram corretamente sentidos ou inferidos	$\frac{P}{P + FN}$ P = Positivos FN = Falsos Negativos	Quanto mais próximo de 1, melhor	Automática	Adaptada de [Noury 2007]	Confiabilidade
<b>GrE</b>	Grau de Especificidade	Representa a fração de contextos não relevantes identificados pelo sistema sobre o total de contextos não relevantes	$\frac{N}{N + FP}$ N = Negativos FP = Falsos Positivos	Quanto mais próximo de 1, melhor	Automática	Adaptada de [Noury 2007]	Confiabilidade

Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GDFm</b>	Grau de Desempenho <i>F-Measure</i>	Representa a média ponderada entre o grau de precisão e o grau de recall	$2 * \left( \frac{P}{P+FP} * \frac{P}{P+FN} \right)$ $\frac{P}{P+FP} + \frac{P}{P+FN}$ <p>P = Positivos FN = Falsos Negativos FP = Falsos Positivos</p>	Quanto mais próximo de 1, melhor	Automática	Adaptada de [Noury 2007]	Confiabilidade
<b>GC</b>	Grau de Correção ou Acurácia	Representa a fração de contextos corretamente sentidos ou inferidos	<p>X = A, onde A é</p> $\frac{P + N}{P + N + FN + FP}$ <p>FP = Falsos positivos P = Positivos FN = Falsos Negativos N = Negativos</p>	Quanto mais próximo de 1, melhor	Automática	Adaptada de [Ranganathan 2005]	Confiabilidade

Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
GA	Grau de Adaptação	Busca identificar se o sistema se adaptada sempre que necessário. Implicando na ausência de notificações equivocadas ou provimento de serviço baseado em informações atualizadas.	$X = \frac{(\sum_{j=1}^N \frac{A_j}{B_j}) * 100}{N}$ <p>Aj = nº de vezes que adaptou</p> <p>Bj = nº de vezes que foi requisitada a adaptação</p> <p>mudança contextual</p> <p>N = nº dos tipos das diferentes adaptações</p>	Quanto mais próximo de 100%, melhor.	Automática	Retirada de [Santos 2013]	Confiabilidade

**Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1**

<b>ID</b>	<b>Nome da Medida</b>	<b>Descrição</b>	<b>Função de Medição</b>	<b>Interpretação</b>	<b>Método de Coleta</b>	<b>Ref.</b>	<b>Subcaracterística Associada</b>
<b>GCEA</b>	Grau de Corretude na Execução da Adaptação	Representa a execução correta dos mecanismos ou regras de adaptação implicando em um provimento de serviço conforme o esperado.	$X = A$ , onde A é (1) Alto (2) Médio (3) Baixo	(1) Alto: execução correta das regras ou mecanismos de adaptação. (2) Médio: execução de mais de uma regra ou mecanismo de adaptação, implicando em uma adaptação incorreta. Porém não inviabilizando o provimento do serviço. (3) Baixo: execução de mais de uma regra ou mecanismo de adaptação sobre a mesma estrutura, implicando em uma adaptação incorreta e inviabilizando o provimento do serviço.	Análise do Desenvolvedor	Baseada em [Cheng 2009]	Confiabilidade

---

**Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1**

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GCAS</b>	Grau de Concordância das Ações do Sistema	O número de funções providas pelo sistema deve ser compatível com o número de funções executadas e esperadas (e.g., ao entrar em um ambiente o usuário espera receber uma única informação, logo, o sistema não deve fornecer nenhuma ou mais de uma informação).	$X = A/B$ Onde, A = Número de funções executadas e esperadas B = Número de funções providas pelo serviço ubíquo	Quanto mais próximo de 1, melhor	Automático e Análise do Usuário	Adaptada de [Lee e Yun 2008]	Confiabilidade
<b>GIAS</b>	Grau de independência ou autonomia do sistema	Mede a necessidade de requisições de ajuda, por parte do sistema, para confirmar o resultado de alguma saída e/ou serviço.	$X = A$ , onde A é (1) Alto (2) Médio (3) Baixo	(1) Alto: o sistema não necessita ajuda do usuário para confirmar o resultado de nenhuma saída e/ou serviço. (2) Médio: o sistema necessita da ajuda do usuário para confirmar o resultado de alguma saída e/ou serviço (3) Baixo: o sistema necessita da ajuda do usuário para confirmar o resultado.	Automático e Análise do Usuário	Adaptada de [Scholtz e Consolvo 2004]	Confiabilidade

**Tabela 15 (Continuação): As 11 Medidas para a questão Q1.1**

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	Avaliar se o sistema funciona em <i>background</i>	$X = A$ , onde $A$ é (1) Alto (2) Médio (3) Baixo	(1) Alto: o sistema funciona normalmente em <i>background</i> (2) Médio: Algum aspecto do sistema é comprometido quando o dispositivo está bloqueado e a aplicação rodando, consequentemente, em <i>background</i> (3) Baixo: O sistema não funciona em <i>background</i>	Automático e Análise do Usuário	Indicada em Reunião GQM	Confiabilidade
<b>GInB</b>	Grau de Influência sobre a Bateria	Avaliar a influência do sistema sobre o consumo de bateria do dispositivo.	$X = A$ , onde $A$ é (1) Alto (2) Médio (3) Baixo	1) Baixo: a diferença entre o consumo de bateria com o sistema rodando ou sem rodar não ultrapassa 10% (2) Médio: a diferença entre o consumo de bateria não ultrapassa 25% (3) Alto: a diferença entre o consumo de bateria com o sistema rodando ou sem rodar ultrapassa 25%.	Automático e Análise do Usuário	Indicada em Reunião GQM	Confiabilidade

A medida Grau de Precisão (GrP) busca avaliar a precisão do sistema, isto é, a capacidade do sistema de identificar que houve uma mudança específica de contexto onde, essa mudança é de fato importante para o provimento do serviço. Essa capacidade também é expressa em forma de porcentagem, quanto mais próximo de 100% maior a precisão do sistema e maior o impacto positivo da confiabilidade sobre a confiança no funcionamento do sistema.

A medida Grau de *Recall* (GrR) busca avaliar a cobertura ou abrangência dos contextos sentidos ou inferidos pelo sistema, isto é, a capacidade do sistema de identificar que houve uma mudança de contexto relevante para o provimento de serviço. Essa capacidade é expressa em forma de porcentagem, quanto mais próximo de 100% (ou 1) maior a cobertura ou abrangência do sistema e maior o impacto positivo da confiabilidade sobre a confiança do sistema. A medida representa a razão entre as mudanças que aconteceram e o sistema detectou e todas as mudanças de contexto detectadas pelo sistema, sejam elas relevantes ou não. A interpretação para P e FP é a mesma utilizada na medida M03. Além disso, FN indica as mudanças que houve e o sistema não foi capaz de detectar. O critério FN faz parte também da função de medição de outras medidas, possuindo o mesmo significado descrito em M4.

A medida Grau de Especificidade (GrE) representa a fração de contextos não relevantes identificados pelo sistema sobre o total de contextos não relevantes, a idéia chave dessa medida é que as mudanças de contexto que não são relevantes para o provimento de serviços e o total de mudanças não relevantes identificados seja compatível, conseqüentemente, quanto mais próximo de 1 melhor, pois isto implica em um baixo número de falsos positivos.

A medida Grau de Desempenho (GDFm) representa a média ponderada entre o grau de precisão e o grau de *recall*. Em estatística, a média é o valor que aponta para onde mais se concentram os dados de uma distribuição e pode ser considerado o ponto de equilíbrio. Na média simples todos os valores possuem um mesmo peso, situação diferente na média ponderada, que para cada valor deve-se levar em conta o valor do seu peso.

A medida Grau de Correção (GC) representa a porcentagem de mudanças contextuais corretamente sentidas ou deduzidas. Para isso é avaliada a razão entre a soma de positivos e negativos (representando contextos relevantes) e a soma de falsos positivos, falsos negativos, positivos e negativos (representando todos os contextos).

Diferentemente de GrP, a medida GC pode não ser útil (e.g., gerar valores de interpretação próximos do melhor possível para sistemas onde a necessidade de melhorias é evidente) em alguns cenários (e.g., valores altos para N e valores baixos para P, FN e FP, podem gerar um grau de acurácia alto para um sistema onde o grau de precisão é baixo).

A medida Grau de Adaptação (GA) busca avaliar se as adaptações ocorreram no momento esperado, mesmo que de maneira errada. Essa medida foi definida, pois o sistema pode não conseguir se adaptar, em alguns casos, mesmo diante de uma mudança contextual. Para calcular essa medida, é necessário identificar quais adaptações e mudanças de contexto o sistema possui e que podem ser coletadas e, então, contar quantas vezes uma determinada mudança contextual (requisição para a adaptação ocorrer) aconteceu durante um determinado período de uso da aplicação e quantas dessas vezes ocorreram a adaptação. Como um sistema pode ter várias adaptações, a medida apresenta um somatório com o objetivo de somar todas as adaptações ocorridas, para depois calcular uma média ao dividir por N.

A medida Grau de Corretude na Execução da Adaptação (GCEA) busca avaliar aspectos do sistema que possibilitem garantir que o serviço é entregue ao usuário de forma adequada. Para tanto, são analisadas as regras ou outros possíveis mecanismos (e.g., mecanismos de inferência) utilizados pelo sistema para o provimento de serviço, dada a ocorrência de mudanças contextuais.

Como exemplo da medida GCEA, suponha um sistema onde, dado a localização do usuário, ele sugere uma indicação de lugar para comer. A indicação deverá ser a de local mais próximo a posição atual do usuário. Entretanto, caso o usuário tenha cadastrado seu perfil como “Vegetariano”, o sistema deverá sugerir locais tipicamente vegetarianos, caso estes estejam em um raio de até 2 km, ainda que não seja o restaurante mais próximo ao usuário em determinado instante (i.e. restaurantes vegetarianos tem prioridade sobre restaurantes não vegetarianos). Suponha ainda que, ao passear por uma rua, o aplicativo sugere um local não vegetariano ao usuário. A princípio, a aplicação está em correto funcionamento, no entanto, não realiza o serviço em conformidade com as expectativas e necessidades do usuário, pois no momento da adaptação, não disparou a regra correta. Imaginando ainda o mesmo cenário, outra situação incorreta poderia acontecer, caso o sistema fizesse a sugestão de dois lugares ao usuário, o vegetariano e o não vegetariano. A ocorrência de tal sugestão dupla poderia não indicar um problema, sob o ponto de vista do usuário, no entanto, continuaria sendo fruto de uma adaptação errada em virtude da

execução de duas ou mais regras sobre a mesma adaptação. Nesse sentido, o usuário não seria prejudicado, porém, adaptações incorretas poderiam surgir em outras situações, causadas pelo mesmo motivo (i.e. execução de regras não esperadas). Logo, identificar a raiz dessa não correteza é importante.

A medida Grau de Concordância das Ações do Sistema (GCAS) busca avaliar se o número de funções providas pelo sistema e o número de funções esperadas é compatível. Para isso é utilizada a razão entre o número de funções providas e o número de funções requisitadas. Quanto mais próximo de 1, melhor.

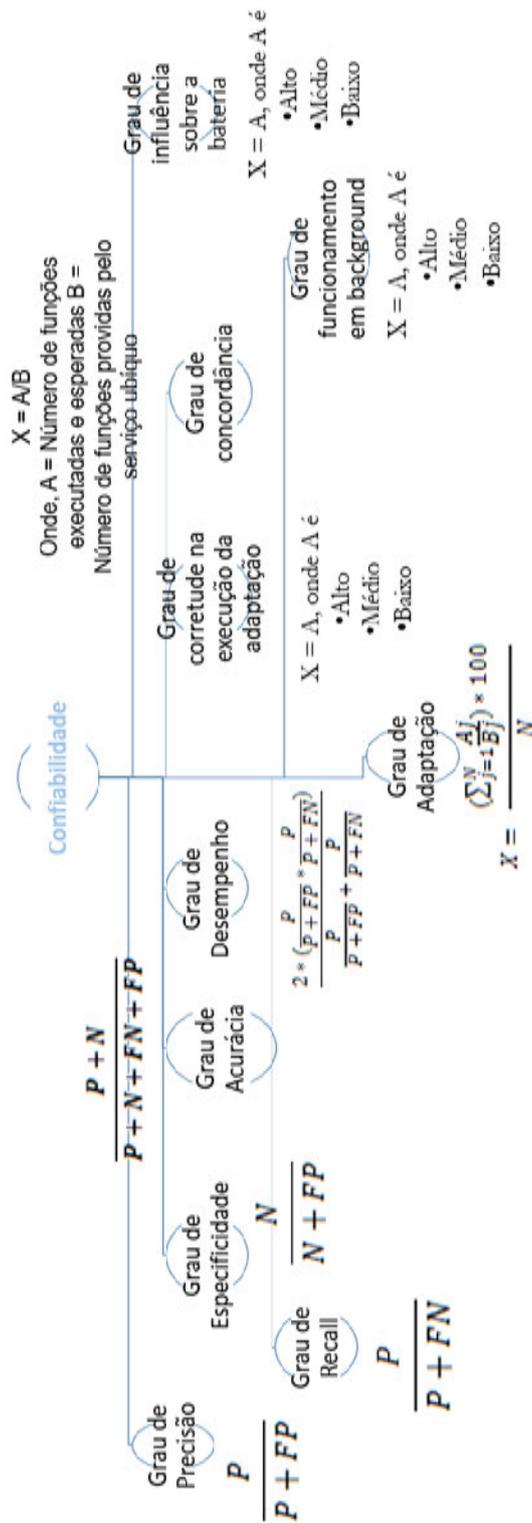
A medida Grau de Independência ou Autonomia do Sistema (GIAS) busca avaliar a independência do sistema para tomar decisões, isto é, a necessidade de requisições de ajuda, por parte do sistema, para confirmar o resultado de alguma saída e/ou serviço. Quanto maior o número de requisições de ajuda mais próximo de ‘Alto’ o valor da medida, quanto menor, mais próximo de ‘Baixo’.

A medida Grau de Funcionamento em *Background* (GFB) busca avaliar se o serviço é entregue ao usuário conforme o especificado e sempre que requisitado, mesmo quando o dispositivo estiver bloqueado (em descanso). Essa medida retrata aspectos específicos dos sistemas desenvolvidos para dispositivos móveis.

A medida Grau de Independência da Bateria (GI<sub>B</sub>) tem como objetivo garantir que o serviço é entregue ao usuário conforme o especificado e sempre que requisitado, mesmo quando o dispositivo estiver com bateria baixa. A ideia-chave desta medida é avaliar como o sistema se comporta quando a bateria do dispositivo está baixa (menos de 25%). Algumas funções nativas do desenvolvimento para sistemas *Android* (o comando *broadcast*, por exemplo) não são ativadas (ou ativadas em intervalos de tempo maiores do que o adequado) quando o dispositivo está com pouca bateria.

A medida Grau de Influência sobre a Bateria (GI<sub>InB</sub>) busca garantir que o serviço é entregue ao usuário conforme o especificado e sempre que requisitado sem comprometer significativamente (e.g., 15% a mais de consumo quando utilizando o sistema) a bateria do dispositivo. Uma forte influência sobre o consumo de bateria pode levar o usuário a abrir mão do uso do sistema, desinstalando-o do seu dispositivo.

A Figura 11 apresenta de maneira sintetizada (com nome, subcaracterística associada e função de medição) as medidas da Tabela 15.



**Figura 11: Medidas de Confiabilidade**

Tabela 16: As 3 Medidas para a questão Q1.2

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
GPUD	Grau de Permissão no Uso de Dados	Representa a existência de recursos disponíveis ao usuário para controlar as permissões de uso sobre suas informações pessoais: privacidade do conteúdo, identidade, localização e outros.	$X = A$ , onde A é (1) Alto (2) Médio (3) Baixo	(1) Alto: o usuário tem controle sobre 75% ou mais dos dados pessoais que são utilizados pelo sistema e pode alterar as permissões a qualquer momento.  (2) Médio: existem recursos disponíveis para o usuário controlar e alterar a permissão de alguns (75% - 50%) dos seus dados pessoais que são utilizados pelo sistema.  (3) Baixo: Existem recurso disponíveis para o usuário controlar ouso de menos de 50% dos seus dados que são utilizados por parte do sistema.	Análise do Usuário	Baseada em [Scholtz e Consolvo 2004]	Confidencialidade

Tabela 16 (Continuação): As 3 Medidas para a questão Q1.2

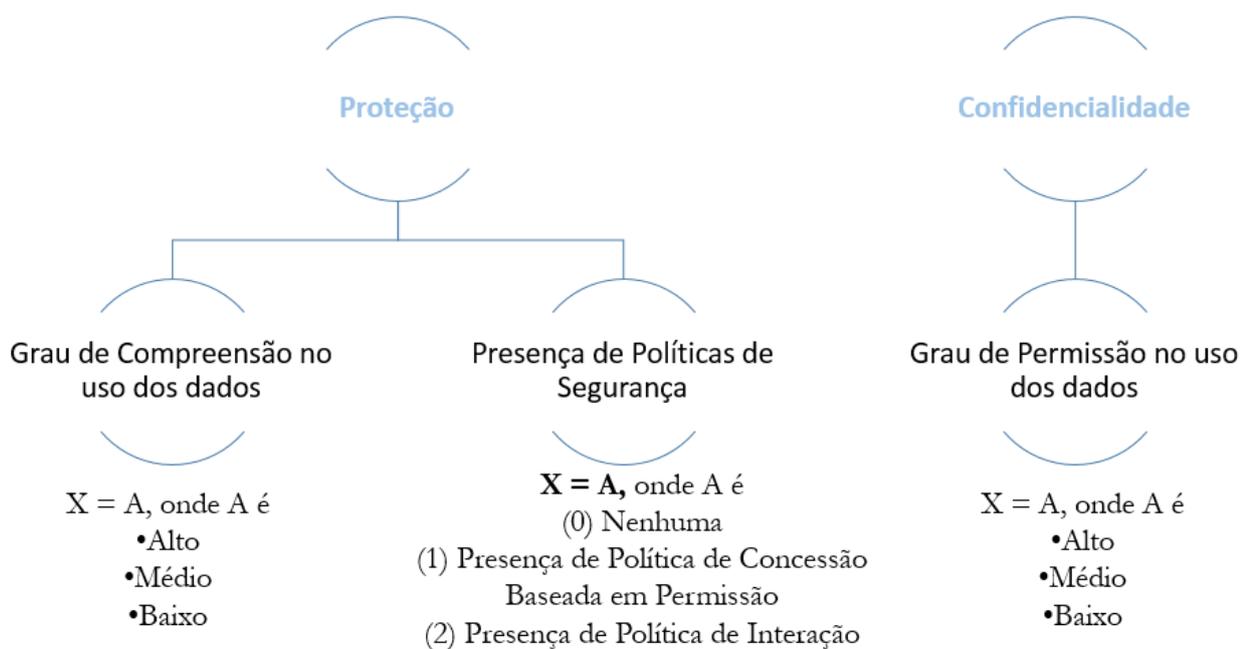
ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GCUD</b>	Grau de Compreensão do Uso de Dados	Mede a capacidade do usuário de entender quais dos seus dados ou informações pessoais são utilizados pelo sistema.	$X = A$ , onde A é (1) Alto (2) Médio (3) Baixo	(1) Alto: o usuário é ciente de mais de 75% das suas informações que são utilizadas pelo sistema. (2) Médio: o usuário é ciente de apenas algumas (75% - 50%) das suas informações que são utilizadas pelo sistema. (3) Baixo: o usuário é ciente de menos de 50% das suas informações que são utilizadas pelo sistema.	Análise do Usuário	Baseada em [Scholtz e Consolvo 2004]	Proteção
<b>PPS</b>	Presença de Políticas de Segurança	Representa a existência de políticas de segurança utilizadas pelo sistema ou pela aplicação	$X = A$ , onde A é (0) Nenhuma (1) Presença de Política de Concessão Baseada em Permissão (2) Presença de Política de Interação	(0) Nenhuma (1) Presença de Política de Concessão Baseada em Permissão (2) Presença de Política de Interação	Automático e Análise do Desenvol-vedor	Baseada em [Kilincet al 2012]	Proteção

A medida Grau de Permissão no Uso dos Dados (GPUD) busca avaliar se há ou não recursos disponíveis ao usuário para controle sobre suas informações pessoais: privacidade do conteúdo, identidade, localização e outros. Por exemplo, é possível alterar permissões? É possível restringir conteúdo? Quanto maior o número recursos disponíveis mais próximos de ‘Alto’ o valor da medida, quanto menor, mais próximo de ‘Baixo’.

A medida Grau de Compreensão no Uso dos Dados (GCUD) busca avaliar se há ou não recursos disponíveis ao usuário para entender e compreender como os dados gravados são utilizados. Por exemplo, em sistemas que utilizam informações sobre a localização do usuário, o usuário é ciente desta utilização? Quanto maior o número de recursos disponíveis mais próximos de ‘Alto’ o valor da medida, quanto menor, mais próximo de ‘Baixo’.

A medida Presença de Políticas de Segurança (PPS) busca avaliar a presença de políticas de segurança utilizadas pelo sistema ou pela aplicação. De acordo com (KILINC et al 2012), referência da qual foram retiradas informações para definição desta medida, as políticas de segurança em sistemas ubíquos e são: Políticas de Concessão Baseada em Permissão e Políticas de Interação.

A Figura 12 apresenta de maneira sintetizada (com nome, subcaracterística associada e função de medição) as medidas da Tabela 16.



**Figura 12: Medidas de Proteção e Confidencialidade**

**Tabela 17: As 6 Medidas para a questão Q1.3**

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GrD</b>	Grau de Disponibilidade	Representa a capacidade do sistema de prover serviço sempre que requisitado	$X = A$ , onde A é (1) Alto (2) Médio (3) Baixo	(1) Alto: o sistema funciona conforme sempre que necessário para a maioria ( $\geq 75\%$ ) dos contextos de uso (2) Médio: O sistema funciona quando requisitado de 50% a 75% dos contextos de uso (3) Baixo: O sistema funciona quando requisitado de 0% a 50% dos contextos de uso	Automática ou Análise do Usuário		Disponibilidade
<b>TxErr</b>	Taxa de Erros	Taxa de ocorrência de erros durante o uso do serviço	$X = A$ , onde A é a quantidade de erros durante o tempo de uso do serviço	Quanto menos erros por tempo de uso, melhor	Automática	Retirada de [Lee e Yun 2008]	Disponibilidade
<b>TAdpt</b>	Tempo de Adaptação	Representa o tempo necessário para que a adaptação ocorra, buscando identificar inicializações precoces ou tardias do sistema.	$X = T$ e $Dp$ , onde T é a média dos tempos que o sistema demorou em se adaptar uma vez que mudou	Alto: minutos Médio: segundos Baixo: milissegundos	Automática	Retirada de [Santos 2013]	Disponibilidade

**Tabela 17 (Continuação): As 6 Medidas para a questão Q1.3**

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GDM</b>	Grau de Discrição de mecanismos de segurança	Tempo usado para interagir com o subsistema de segurança (por exemplo, autenticação) para auxiliar a tarefa principal.	X = A, onde A é (3) Alto (4) Médio (5) Baixo	(1) Alto: minutos (2) Médio: segundos (3) Baixo: milissegundos	Análise do Usuário	Adaptada de [Ranganathan 2005]	Disponibilidade
<b>GIdB</b>	Grau de Independência da Bateria	Avaliar se o sistema funciona sempre da mesma forma, mesmo com o dispositivo em diferentes níveis de bateria	X = A, onde A é (1) Alto (2) Médio (3) Baixo	(1) Alto: o sistema funciona conforme o especificado e sempre que necessário em qualquer nível de bateria (2) Médio: Algum aspecto do sistema é comprometido quando o dispositivo está com pouca bateria (menos de 25%) (3) Baixo: O sistema não funciona quando o dispositivo está com pouca bateria(menos de 25%).	Análise do Desenvolvedor	Indicada em Reunião GQM	Disponibilidade

**Tabela 17 (Continuação): As 6 Medidas para a questão Q1.3**

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GInCE</b>	Grau de Independência de Componentes Externos	Avaliar se o sistema funciona conforme o especificado e sempre que requisitado, independente de componentes externos ao dispositivo no qual ele executa.	$X = A$ , onde A é (1) Alto (2) Médio (3) Baixo	(1) Alto: O sistema funciona conforme o especificado e sempre que necessário mesmo que qualquer componente externo tenha algum problema.  (2) Médio: Algum aspecto do sistema é comprometido quando algum componente externo, facilmente identificável, tem algum problema.  (3) Baixo: O sistema não funciona quando qualquer componente externo tem algum problema.	Análise do Desenvolvedor or	Indicada em Reunião GQM	Disponibilidade

A medida Grau de Disponibilidade (GrD) busca avaliar a disponibilidade do sistema, identificando a capacidade do sistema em prover serviços sempre que requisitado. Para isso, essa avaliação leva em consideração o fato do sistema estar ou não disponível sempre que o usuário necessitar. Para tanto, indica-se que sejam realizadas diversas tentativas em diferentes contextos que representem o máximo possível as situações cotidianas do usuário final, pois, o resultado de uma única tentativa pode não corresponder ao real comportamento do sistema quando utilizado no dia-a-dia.

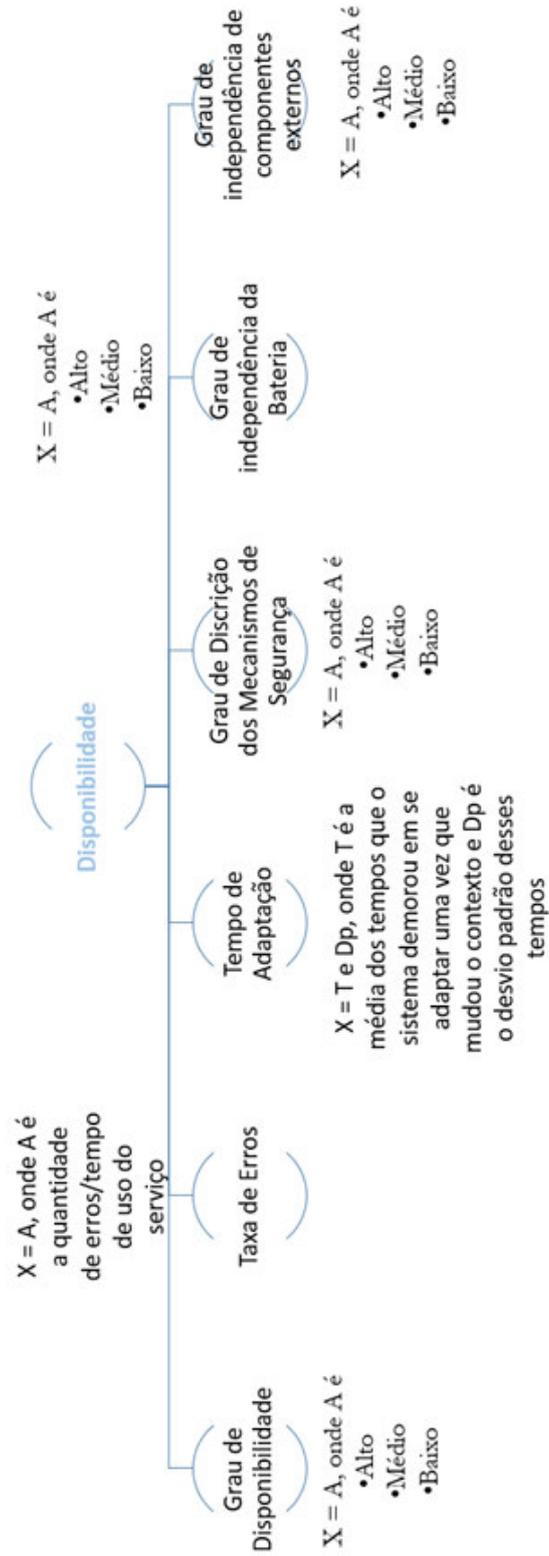
A medida Taxa de Erros (TxErr) busca avaliar a confiança com base na quantidade de erros que acontecem durante o uso do sistema. Para melhores resultados, em relação a confiança no funcionamento, os valores esperados para essa medida são sempre o mais próximo possível de zero, indicando assim a baixa ocorrência de erros.

A medida Tempo de Adaptação (TAdpt) mede o tempo que o sistema demora para se adaptar a partir do instante em que a informação de contexto foi sentida ou inferida. Essa medida é importante porque a informação e/ou o serviço deve ser entregue em um tempo apropriado para o usuário. Para calcular o resultado dessa medida, é necessário calcular a média entre os tempos de adaptação e o desvio-padrão (visto que é necessário verificar se houve resultados muito distantes uns dos outros).

A medida Grau de Discrição dos Mecanismos de Segurança (GDM) busca avaliar o tempo que o sistema precisa para interagir com os mecanismos de segurança existentes. Por exemplo, no momento do *login* o sistema leva tempo longo para autenticar o usuário, isso afeta a confiabilidade e disponibilidade do sistema, pois a prestação de serviço quando requisitado demora e o serviço não é prestado conforme especificado.

A medida Grau de Independência de Componente Externos (GInCE) objetiva garantir que o serviço é entregue ao usuário conforme o especificado e sempre que requisitado independente de componentes externos. Um exemplo de questionamento que essa medida pode ajudar a sanar é: em um sistema que coloca o celular no mudo de acordo com os compromissos da *Google* agenda do usuário. Caso ocorra algum problema de indisponibilidade com o sistema de agendas da *google*, o sistema presente no dispositivo, que interage com essa agenda, também deixará de funcionar?

A Figura 13 apresenta de maneira sintetizada (com nome, subcaracterística associada e função de medição) as medidas da Tabela 17.



**Figura 13: Medidas de Disponibilidade**

Tabela 18: As 2 Medidas para a questão Q1.4

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GIDR</b>	Grau de Integridade dos Dados Transmitidos em Rede	Avaliara existência da implementação de técnicas de integridade sobre os dados transmitidos em rede	$X = A$ , onde A é (0) Alto (1) Médio (2) Baixo	(0) Alto: Implementa técnicas de integridade com verificação e correção. (1) Média: Implementa técnicas de integridade, porém não garante verificação e correção. (2) Baixo: Não implementa técnicas de integridade.	Análise do Desenvolvedor	Indicada em Reunião GQM	Integridade
<b>GIDA</b>	Grau de Integridade dos Dados Armazenados	Avaliar a existência da implementação de técnicas de integridade sobre os dados armazenados	$X = A$ , onde A é (0) Alto (1) Médio (2) Baixo	(0) Alto: Implementa técnicas de integridade com verificação e correção. (1) Média: Implementa técnicas de integridade, porém não garante verificação e correção. (2) Baixo: Não implementa técnicas de integridade.	Análise do Desenvolvedor	Indicada em Reunião GQM	Integridade

**Tabela 19: As 3 Medidas para a questão Q1.5**

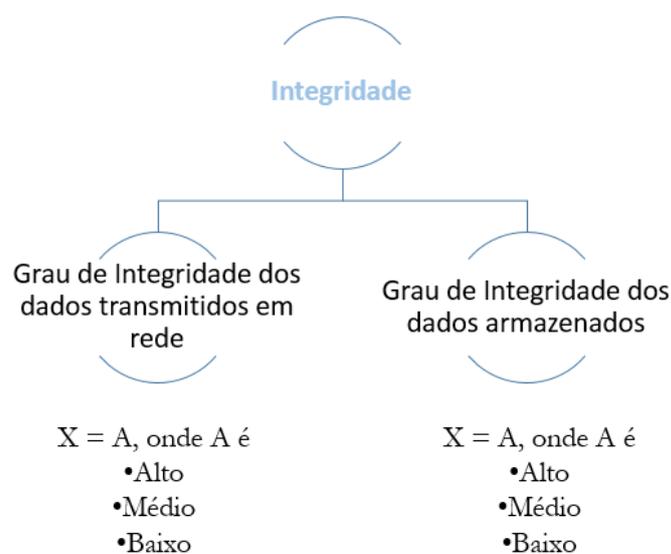
ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
<b>GDoc</b>	Grau de Documentação	Avaliar a existência de documentação associada ao sistema desenvolvido	$X = A$ , onde A é (0) Alto (1) Médio (2) Baixo	(0) Alto: Mais de 75% dos componentes estão documentados. (1) Médio: Apenas uma parte (50% - 75%) dos componentes estão documentados. (2) Baixo: Menos de 50% dos componentes estão documentados.	Análise do Desenvolvedor	Indicada em Reunião GQM	Manutenibilidade
<b>GEMS</b>	Grau de Esforço para Realizar Manutenção do Sistema	Avaliar o esforço, relacionando-o ao tempo, necessário para realizar manutenção do sistema	$X = A$ , onde A é (0) Alto (1) Médio (2) Baixo	(0) Alto: O tempo para realizar manutenção é superior em até 25% ao tempo que foi necessário para implantação. (1) Média: O tempo para realizar manutenção é equivalente ao tempo que foi necessário para implantação. (2) Baixo: O tempo para realizar manutenção é inferior em até 25% ao tempo que foi necessário para implantação.	Análise do Desenvolvedor	Indicada em Reunião GQM	Manutenibilidade

Tabela 19 (Continuação): As 3 Medidas para a questão Q1.5

ID	Nome da Medida	Descrição	Função de Medição	Interpretação	Método de Coleta	Ref.	Subcaracterística Associada
GEES	Grau de Esforço para Realizar Manutenção ou Evolução de Código	Avaliar o esforço, relacionando-o ao tempo, necessário para realizar manutenção ou evolução do código	$X = A$ , onde A é (0) Alto (1) Médio (2) Baixo	(0) Alto: O tempo para realizar manutenção ou evolução é superior em até 25% ao tempo que foi necessário para codificação. (1) Média: O tempo para realizar manutenção ou evolução é equivalente ao tempo que foi necessário para codificação. (2) Baixo: O tempo para realizar manutenção ou evolução é inferior em até 25% ao tempo que foi necessário para codificação.	Análise do Desenvolvedor	Indicada em Reunião GQM	Manutenibilidade

As medidas Grau de Integridade dos Dados Transmitidos em Rede (GIDR) e Grau de Integridade dos Dados Armazenados (GIDA) objetivam avaliar e garantir que os dados utilizados pelo sistema, que são transmitidos via rede ou armazenados, não sejam acessados por quem não tem autorização.

A Figura 14 apresenta de maneira sintetizada (com nome, subcaracterística associada e função de medição) as medidas da Tabela 18.

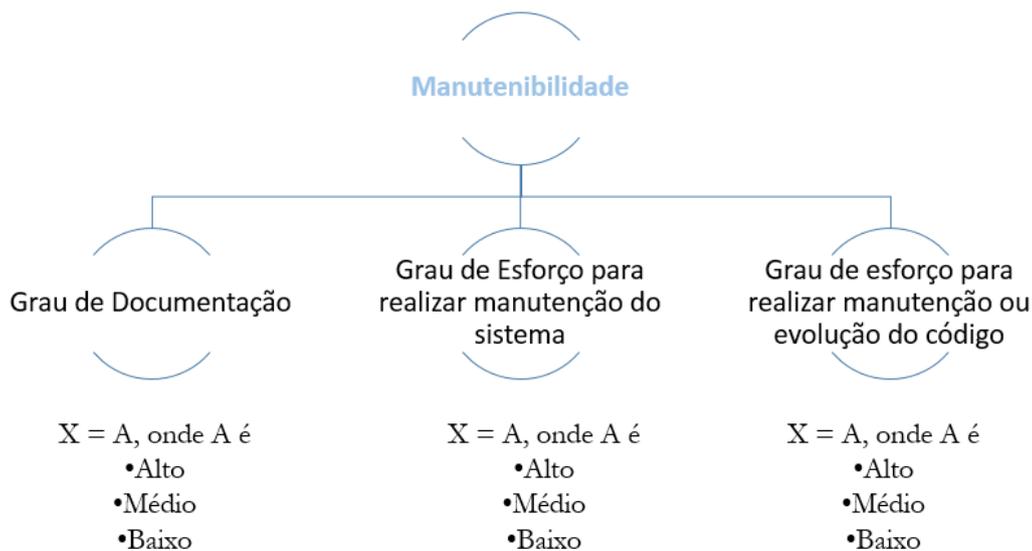


**Figura 14: Medidas de Integridade**

A medida Grau de Documentação (GDoc) tem como objetivo garantir que o sistema desenvolvido possui documentação, auxiliando no seu entendimento e utilização.

As medidas Grau de Esforço para realizar Manutenção do Sistema (GEMS) e Grau de Esforço para realizar Manutenção ou Evolução do Código (GEES), por sua vez, buscam garantir que o tempo necessário para a manutenção ou evolução do sistema e do código seja adequado, não implicando em perdas ou atrasos para o provimento do serviço conforme o especificado.

A Figura 15 apresenta de maneira sintetizada (com nome, subcaracterística associada e função de medição) as medidas da Tabela 19.



**Figura 15: Medidas de Manutenibilidade**

### 3.3 Coleta das Medidas

Como mencionado no Capítulo 2, essa fase do GQM consiste em coletar, armazenar e processar todas as informações necessárias para calcular as medidas (SANTOS 2014). Essa etapa define, portanto, como é realizada a obtenção dos dados que são utilizados na medição. De acordo com (VAN SOLINGEN E BERGHOUT, 1999), durante a fase de coleta de dados, é importante observar quatro questões:

- Para uma determinada medida, quem deve coletar os dados necessários?
- Quando o dado deve ser coletado?
- Como o dado pode ser coletado mais eficientemente?
- Para quem deve ser entregue os dados coletados?

De acordo com (ROCHA et al 2012), como a produção de software é uma tarefa intelectual, a coleta dos dados requer, muitas vezes, observação e coleta manual. Assim, segundo o autor, para garantir que a coleta seja feita de forma correta, é necessário definir como ela é realizada.

Neste trabalho, para coletar as medidas estabelecidas, é necessário utilizar três diferentes métodos:

- Coleta Automática: a coleta de dados automática é feita com base em dados obtidos durante o uso da aplicação (em testes) e armazenados em *log*.

- **Análise do Usuário:** a análise com o usuário é feita por meio de formulários e questionários (apêndices A e B) e procura avaliar a confiança do usuário durante a utilização do sistema.
- **Análise do Desenvolvedor:** a análise com o desenvolvedor é feita por meio de formulários e questionários (apêndices A e C) e busca auxiliar o desenvolvedor na construção de um sistema no qual o usuário possa justificadamente confiar.

Estes métodos, sugeridos por (BASILI ET AL 1994), podem ser encontrados também em outros trabalhos que utilizam o GQM, como por exemplo, em (SANTOS 2014). No entanto, os questionários e formulários são desenvolvidos de acordo com cada pesquisa e, por consequência, pode haver diversas diferenças entre os seus conteúdos.

Para cada medida, o método de coleta a ser utilizado é apresentado na coluna “Método de Coleta”. Recomenda-se que os resultados sejam documentados segundo o formato apresentado no arquivo “Modelo CT”, encontrado no repositório “Qualidade em Sistemas Ubíquos- Pesquisa”, no endereço <https://github.com/andressabezerra>. A padronização na documentação dos resultados torna o processo de avaliação uma atividade repetível. Com isso, os dados obtidos podem ser replicados para fins de comprovação, checagem e pesquisa.

### **3.4 Interpretação dos Resultados**

Esta fase corresponde a quarta e última etapa da metodologia GQM. Como indicado por (VAN SOLINGEN; BERGHOUT, 1999), o objetivo nesse estágio é analisar os resultados obtidos na coleta dos dados e utilizá-los para interpretar as questões e os objetivos de medição definidos. Os valores de interpretação ou intervalo esperado dos dados, apresentados em cada medida, representam aproximações do que seria um bom resultado.

Como apontado por (SANTOS 2014), valores de interpretação específicos não são aplicáveis, pois este resultado depende do propósito do sistema, o que varia de aplicação para aplicação. Além disso, não existem valores históricos com os quais se possa comparar e especificar uma escala de interpretação mais específica.

### **3.5 Discussão sobre as Medidas**

Esta seção busca apresentar as discussões sobre as medidas abordando a adequação e cobertura destas para os sistemas ubíquos para dispositivos móveis. Para tanto, é evidenciada a relação entre as medidas e as possíveis causas para falhas nesse tipo de sistema.

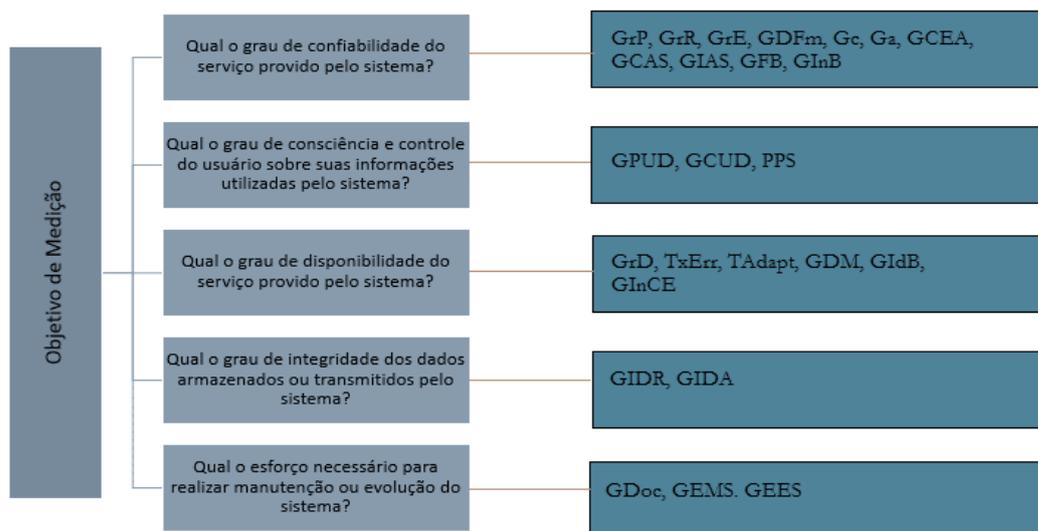
Esta relação foi priorizada, pois como apresentado no Capítulo 2, a ocorrência destas é capaz de afetar a confiança que o usuário possui no funcionamento do sistema. Portanto, apresentando esse relacionamento, é possível mostrar que com a utilização das medidas definidas pode-se minimizar a ocorrência de falhas e, por consequência, fortalecer a confiança no funcionamento.

Para isso, será utilizada a classificação encontrada no estudo de (ROCHA 2010), apresentada em resumo, a seguir:

- Falhas de Dispositivo: Sistemas ubíquos podem ser compostos por diferentes tipos de dispositivos (e.g., computadores de mesa, notebooks, sensores, *smartphones* e outros). Cada dispositivo é suscetível a um conjunto particular de faltas que, em maior ou menor grau, podem contribuir para a falha do dispositivo e do sistema como um todo. Para minimizar esse tipo de falha o desenvolvedor deve estar atento a questões relativas ao uso da bateria, a corretude dos sensores utilizados e outros.
- Falhas de Rede: Os componentes que fazem parte de um sistema ubíquo podem estar conectados através de enlaces de redes, com ou sem fio. Diante desse cenário é preciso projetar e desenvolver sistemas que saibam lidar com problemas como, por exemplo, a baixa força do sinal ou a indisponibilidade de canais de comunicação devido a um alto tráfego na rede.
- Falhas de Aplicação: As falhas de aplicações podem ser causadas por faltas de código (*bug*), erros do sistema operacional, exceções não tratadas e mau uso por parte do usuário. Além disso, as aplicações podem ser vítimas de softwares maliciosos (e.g., vírus e *worms*).
- Falhas de Serviço: Os ambientes ubíquos são repletos de serviços. Esses serviços são responsáveis por prover diferentes funcionalidades para as aplicações e para o próprio sistema, sendo susceptíveis aos mesmos tipos de falhas das aplicações, apresentadas anteriormente.
- Falhas de Adaptação: Em sistemas ubíquos, o subsistema de adaptação é responsável por ajustar tanto os serviços como as aplicações às mudanças de contexto. Durante o processo de adaptação, faltas podem fazer com que a estrutura das aplicações e/ou dos serviços seja modificada de maneira errada. Isso faz com que as aplicações e/ou serviços passem a gerar um comportamento inesperado e, possivelmente, errôneo. Isso pode ocorrer tanto pela especificação errada da regra de adaptação quanto pelo conflito entre regras.

- Falhas de Contexto: O subsistema de contexto é responsável pela gerência do contexto desde a fase de aquisição até a notificação aos demais componentes do sistema. Falhas no subsistema de contexto podem ocasionar a inicialização precoce ou tardia de aplicações, adaptações indevidas ou inferências errôneas.

O conjunto final de medidas é resumido na Figura 16.



**Figura 16: Modelo Final GQM**

A Figura 17 resume a relação entre as medidas definidas (Figura 16) e as causas das falhas em sistemas ubíquos. Na parte superior, estão as falhas em sistemas ubíquos. Na parte lateral esquerda da figura, são descritas as medidas. Os quadrados escurecidos indicam que a medida da linha horizontal está relacionada com a falha da linha vertical e pode ser utilizada para minimizar (direta ou indiretamente) a ocorrência da falha.

Como pode-se observar na Figura 17, as medidas GD, TE, GIdB e GInB podem minimizar falhas de dispositivo pois, estão diretamente associadas a avaliação da capacidade do sistema de prover serviços sempre que requisitado. Assim sendo, portanto, impactadas por questões associadas ao uso da bateria, correteude dos sensores utilizados e outros. Além disso, a baixa força do sinal ou a indisponibilidade de canais de comunicação também podem ser identificados por meio da coleta dessas medidas, assim, incluindo também as medidas GCEA e GIDR, é possível minimizar falhas de rede.

As medidas GD – GICE são capazes de auxiliar a identificação de falhas que podem ser causadas por faltas de código (*bug*), erros do sistema operacional, exceções não

tratadas e mau uso por parte do usuário. Com isso, é possível minimizar falhas de aplicação e serviço.

	<i>Falha de Aplicação</i>	<i>Falha de Dispositivo</i>	<i>Falha de Serviço</i>	<i>Falha de Adaptação</i>	<i>Falha de Contexto</i>	<i>Falha de Rede</i>
GrD						
TxEr						
GrP						
GrR						
GE						
GDFm						
GE						
TAdpt						
GA						
GCEA						
GDMS						
GCUD						
GPUD						
GCAS						
GIAS						
GIAS						
PPS						
GDoc						
GEMS						
GEES						
GIDR						
GIDA						
GFB						
GIdB						
GInB						
GICE						

**Figura 17: Relação entre medidas e falhas**

Por fim, as medidas TE – GCEA, GPUD – GIAS e GIDR – GICE podem minimizar falhas de contexto e adaptação, pois procuram avaliar questões relativas ao contexto, que vão desde a fase de aquisição até a fase de notificação dos demais componentes do sistema, e como estes se ajustam de acordo com as mudanças contextuais.

### 3.6 Considerações Finais

Neste capítulo foi apresentado um conjunto de vinte e cinco medidas de qualidade de *software* para avaliação da confiança no funcionamento em sistemas ubíquos que executam em dispositivos móveis, das quais:

- 4 medidas foram retiradas da literatura  
(GD, TxEr, TAdapt, GA)
- 8 medidas adaptadas  
(GrP, GrR, GE, GDFm, GC, GDMS, GCAS, GIAS)

- 13 medidas propostas com base em informações obtidas da literatura, em leitura de artigos, ou nas reuniões GQM (GCEA, GCUD, GPUD, PPS, GDoc, GEMS, GEES, GIDR, GIDA, GFB, GIdB, GInB, GICE).

No próximo capítulo essas medidas são avaliadas por meio de três estudos de caso.

## 4 AVALIAÇÃO DAS MEDIDAS

Este capítulo apresenta, por meio de estudos de caso, a aplicação das medidas definidas no capítulo anterior. A Seção 4.1 apresenta o planejamento para executar os três estudos de caso. A Seção 4.2 apresenta a coleta e interpretação das medidas na aplicação *GREat Mute*. A Seção 4.3 apresenta a coleta e interpretação das medidas na aplicação *GREat Tour*, a Seção 4.4 apresenta a coleta e interpretação das medidas na aplicação *fAlert*. A Seção 4.5 apresenta a validação dos resultados obtidos com a coleta das medidas nos três estudos de caso e, por fim, a Seção 4.6 apresenta as considerações finais do capítulo.

### 4.1 Planejamento

Com o intuito de avaliar o uso das medidas apresentadas no Capítulo 3, estudos de casos (EC) são utilizados. Esse método de investigação foi escolhido, pois costuma ser utilizado por projetos que utilizam a abordagem GQM na fase de coleta de dados (VAN SOLINGEN e BERGHOUT 1999) e permite a coleta de dados tanto quantitativos como qualitativos.

Vale ressaltar ainda que o estudo de caso trata-se de uma abordagem de investigação, especialmente adequada quando procuramos compreender, explorar ou descrever acontecimentos e contextos, nos quais estão simultaneamente envolvidos diversos fatores (YIN 1994). Eles também podem ser conduzidos em configurações do mundo real e, assim, tem um alto grau de realismo, permitindo, portanto, realizar testes com usuários interagindo com um sistema real. Por ter várias aplicações, os estudos de caso são apropriados para pesquisadores individuais, pois dão a oportunidade para que um aspecto de um problema seja estudado em profundidade dentro de um período de tempo limitado. Além disso, dado a sua flexibilidade, são úteis na exploração de novos processos, pois, auxiliam a compreensão dos novos comportamentos e concepções.

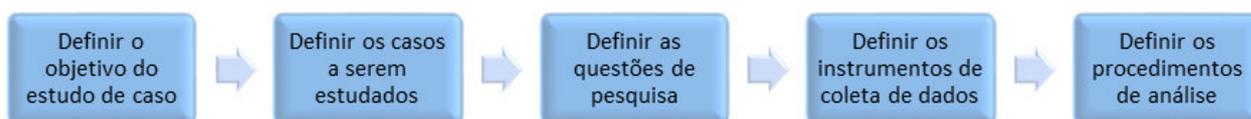
Os estudos de caso se caracterizam ainda por (YIN 1994):

- **Objetivar a descoberta:** o investigador se mantém atento a novos elementos que podem surgir, buscando novas respostas e novas indagações no desenvolvimento do seu trabalho.

- Enfatizar a interpretação contextual: para melhor compreender um problema, deve-se relacionar as ações, os comportamentos e as interações das entidades envolvidas com a problemática da situação.
- Retratar a realidade de forma completa e profunda: o pesquisador enfatiza a complexidade da situação, procurando revelar a multiplicidade de fatos que a envolvem e a determinam.
- Utilizar várias fontes de informação: o pesquisador recorre a uma variedade de dados, coletados em diferentes momentos, em situações variadas e com uma variedade de tipos de informantes.

Como mencionado anteriormente, para a realização dos estudos de caso desta dissertação são utilizadas três aplicações ubíquas para dispositivos móveis, com o intuito de avaliar a confiança no funcionamento nesse tipo de sistema. Os resultados obtidos são apresentados nas seções a seguir.

Para o planejamento dos estudos de caso deste trabalho, foram seguidas as mesmas atividades de planejamento propostas por (RUNESON; HÖST, 2008) e utilizadas por (SANTOS 2014), apresentada na Figura 18.



**Figura 16: Planejamento para os estudos de caso**

Baseado na motivação desta pesquisa, o objetivo do estudo de caso é avaliar a confiança no funcionamento em sistemas ubíquos. Como casos a serem estudados, estão as três aplicações ubíquas mencionadas no início deste capítulo. As questões de pesquisa são as mesmas derivadas no GQM e apresentadas no Capítulo 3. Por fim, os instrumentos de coleta e procedimentos de análise variam de acordo com cada medida e podem ser encontrados em sua definição no capítulo 3.

Para cada uma das aplicações, um grupo de pessoas está envolvido na realização dos estudos de caso, conforme apresentado na Tabela 20.

**Tabela 20: Perfil dos usuários envolvidos nos estudos de caso**

	<b>GREat Mute</b>	<b>fAlert</b>	<b>GREat Tour</b>
<b>Quantidade Total de Pessoas</b>	3	4	2
<b>Apenas Usuário</b>	1	3	0
<b>Desenvolvedor e Usuário</b>	2	1	2
<b>Média de Idade</b>	22 anos	23 anos	22 anos
<b>Escolaridade</b>	Graduandos	Graduandos	Graduandos
<b>Familiaridade com o uso de Smartphones</b>	Alta para todos	Alta para todos	Alta para todos
<b>Familiaridade com o uso de sistemas ubíquos</b>	Média para todos	Média para todos	Média para todos

Como limitação, aponta-se o fato de que parte dos participantes envolvidos nos testes relativos aos estudos de caso são também desenvolvedores dos sistemas avaliados. Apesar disso, considera-se útil e importante a sua participação, pois, como mencionado no capítulo 3, algumas medidas definidas devem ser coletadas e analisadas sob o ponto de vista do desenvolvedor e não apenas do usuário final.

Ainda sobre os testes, os dispositivos apresentados na Tabela 21 foram utilizados. Os resultados obtidos são apresentados nas demais seções deste capítulo.

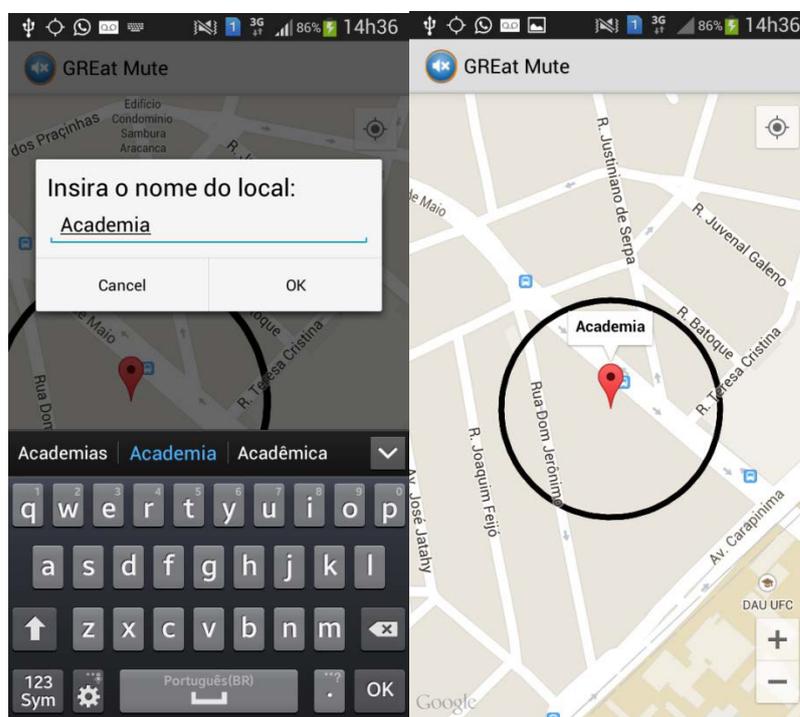
**Tabela 21: Dispositivos usados nos estudos de caso**

Samsung Galaxy S2 Slim Duos	
<b>Dimensões</b>	133 x 66 x 9,7 mm
<b>Tamanho de tela</b>	4,5 polegadas 960x540 pixel
<b>Peso</b>	139g
<b>Tecnologias sem fio</b>	Wi-Fi, 3G, EDGE e GPRS
<b>Armazenamento</b>	8 GB
<b>Memória RAM</b>	1 GB
<b>Processador</b>	Quad-core 1,1 Ghz
<b>Bateria</b>	Li-Ion 2100 mAh
<b>Qualidade do som</b>	
<b>Sensores</b>	Acelerômetro, microfone, proximidade, GPS, barômetro e Bússola
<b>Sistema Operacional</b>	Android 4.0.2 (JellyBean)
Samsung Galaxy S3 Slim Duos	
<b>Dimensões</b>	133 x 66 x 9,7 mm
<b>Tamanho de tela</b>	4,5 polegadas 960x540 pixel
<b>Peso</b>	139g
<b>Tecnologias sem fio</b>	Wi-Fi, 3G, EDGE e GPRS
<b>Armazenamento</b>	16 GB
<b>Memória RAM</b>	1 GB
<b>Processador</b>	Quad-core 1,2 Ghz
<b>Bateria</b>	Li-Ion 2100 mAh
<b>Qualidade do som</b>	
<b>Sensores</b>	Acelerômetro, microfone, proximidade, GPS, barômetro e Bússola
<b>Sistema Operacional</b>	Android 4.2 (JellyBean Plus)
Motorola Moto G2	
<b>Dimensões</b>	141.5 x 70.7 x 6.0-11.0mm (curvo)
<b>Tamanho de tela</b>	5 polegadas HD 720x1280 pixel
<b>Peso</b>	149 gramas
<b>Tecnologias sem fio</b>	802.11 b/g/n, Bluetooth v 4.0,
<b>Armazenamento</b>	8 Gb
<b>Memória RAM</b>	1 Gb
<b>Processador</b>	quad-core 1,2 GHz
<b>Bateria</b>	2070 mAh
<b>Qualidade do som</b>	
<b>Sensores</b>	
<b>Sistema Operacional</b>	Android 4.4.4, KitKat®

## 4.2 GREat Mute

A *GREat Mute*, apresentada em (SANTOS 2014), consiste em uma aplicação para *Smartphones* com sistema operacional *Android*, capaz de silenciar o dispositivo móvel do usuário em determinadas condições nas quais isso é adequado (Figura 19).

Executando em *background*, a aplicação monitora o *Google Calendar* do usuário. Ao encontrar eventos nos quais o usuário considere que não é adequado receber ligações, como “reunião”, “aula” ou “cinema” a aplicação verifica se o usuário encontra-se ou não no local associado ao determinado compromisso e, caso ele esteja no referido local, o dispositivo móvel aciona o modo silencioso.



**Figura 17: GREatMute Versão 2.0**

A aplicação encontra-se atualmente na versão 3.0. Anteriormente, na versão 1.0 a aplicação tinha acesso apenas ao *Google Calendar* do usuário e colocava o dispositivo móvel no silencioso baseada apenas nas informações obtidas na agenda. Caso o usuário não comparecesse ao local do evento no horário marcado, a aplicação não era capaz de identificar essa situação, gerando, por vezes, o provimento do serviço de forma incorreta ou indesejada.

A versão 2.0 utiliza o GPS para verificar a localização do usuário e, a versão 3.0 faz a combinação do endereço obtido do *Google Calendar* e a localização obtida com o GPS,

sem que seja necessário o usuário determinar explicitamente a localização do seu compromisso no mapa.

Para cada uma das versões, os resultados obtidos com a coleta das medidas são apresentados nas seções 4.2.1, 4.2.2 e 4.2.3.

### 4.2.1 GREat Mute versão 1.0

Como mencionado na Seção 4.2, a versão 1.0 da *GREat Mute* utiliza apenas algumas (e.g., nome do compromisso, horário do compromisso) informações obtidas do *Google Calendar* do usuário. Essa aplicação não faz uso de sensores ou informações que indiquem a real localização do usuário nos horários associados à compromissos.

Para a análise dessa versão foram consideradas informações obtidas das coletas realizadas por (SANTOS 2014) em seu trabalho, onde a mesma aplicação é chamada de *UbiMute*. Além disso, foi necessário realizar novos testes com o aplicativo *GREat Mute* 1.0, pois as informações provenientes do estudo de (SANTOS 2014) não eram suficientes para a coleta de todas as medidas desta dissertação.

Foram realizados 150 testes com dois usuários, utilizando os dispositivos *Samsung Galaxy S3* e *Samsung Galaxy S4*. Nesta seção são apresentados os resultados obtidos pela coleta de dados na aplicação utilizando, para cada subcaracterística da confiança no funcionamento, uma questão do GQM (apresentadas no Capítulo 3). Cada questão é comentada de acordo com os resultados das medidas obtidas.

Na Tabela 22 são apresentados os resultados relacionados à questão: Qual o grau de disponibilidade do serviço provido pelo sistema?

**Tabela 22: Resultados - Disponibilidade – GREat Mute v1.0**

Id	Nome	Resultado
<b>GrD</b>	Grau de Disponibilidade	X = Alto
<b>TxErr</b>	Taxa de Erros	X = 1 <sup>9</sup>
<b>TAdapt</b>	Tempo de Adaptação	X = T = 63 milissegundos Dp = 112 milissegundos
<b>GDM</b>	Discrição dos mecanismos de segurança	N/A

<sup>9</sup> A medida TxErr foi coletada e não se identificou nenhum problema de disponibilidade, entretanto, o intervalo de tempo de coleta talvez não tenha sido o suficiente para detectar falhas de comunicação tanto entre a aplicação e o *Google Calendar* quanto no próprio sistema celular.

**Tabela 22 (Continuação): Resultados - Disponibilidade – GREAt Mute v1.0**

Id	Nome	Resultado
<b>GidB</b>	Grau de Independência da Bateria	X = Alto
<b>GInB</b>	Grau de influência sobre a bateria	X = Médio
<b>GInCE</b>	Grau de independência de componentes externos	X = Médio

O grau de disponibilidade do sistema em sua versão 1.0 pode ser considerado alto, apesar de existir dependência em relação a componentes externos (e.g., *google agenda*), esteve disponível para mais de 85% dos casos de teste. Além disso, o tempo de adaptação é aceitável para os usuários que realizaram os testes e o comportamento do sistema, com relação a disponibilidade, não é afetado por índices baixos de bateria.

Na Tabela 23 são apresentados os resultados relacionados à questão: Qual o grau de integridade dos dados armazenados ou transmitidos pelo sistema?

**Tabela 23: Resultados de Integridade – GREAt Mute v1.0**

Id	Nome	Resultado
<b>GIDR</b>	Grau de Integridade dos Dados Transmitidos em Rede	X = Baixo
<b>GIDA</b>	Grau de Integridade dos Dados Armazenados	X = Baixo

Com relação ao grau de integridade, a versão 1.0 não implementa nenhum mecanismo para garantia da integridade dos dados transmitidos pelas diferentes partes envolvidas na entrega do serviço. Também não há garantias de integridade para os dados armazenados no dispositivo. Logo, há necessidade de melhorias sob o ponto de vista do desenvolvedor. Porém, sob o ponto de vista do usuário final, a falta de mecanismos de integridade não é identificável e, como não houve problemas oriundos disto, também não houve nenhum efeito sobre o grau de confiança do funcionamento sob o ponto de vista do usuário no sistema.

Na Tabela 24 são apresentados os resultados relacionados à questão: Qual o esforço necessário para realizar manutenção ou evolução do sistema?

**Tabela 24: Resultados – Manutenibilidade – GREat Mute v1.0**

Id	Nome	Resultado
<b>GDoc</b>	Grau de Documentação	X = Baixo
<b>GEMS</b>	Grau de Esforço para Realizar Manutenção do Sistema	X = Médio
<b>GEES</b>	Grau de Esforço para Realizar Manutenção ou Evolução do Código	X = Médio

Com relação ao grau de manutenibilidade, a coleta das medidas indica a necessidade de documentação, pois a ausência desta implica em esforço extra para realizar a manutenção e evolução do sistema para as versões seguintes.

Na Tabela 25 são apresentados os resultados relacionados à questão: Qual o grau de confiabilidade no serviço provido pelo sistema?

**Tabela 25: Resultados - Confiabilidade - GREat Mute v1.0**

Id	Nome	Resultado
<b>GrP</b>	Grau de Precisão	X = 0,27 ou 27%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,88 ou 88%
<b>GC</b>	Grau de Acurácia	X = 0,92 ou 92%
<b>GCEA</b>	Corretude na Execução da Adaptação	X = Médio
<b>GrE</b>	Grau de Especificidade	X = 65%
<b>GDFm</b>	Grau de Desempenho	X = 78%
<b>GA</b>	Grau de Adaptação	X = 92%
<b>GCAS</b>	Grau de Concordância	X = Médio
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Baixo

Com relação ao grau de confiabilidade, a versão 1.0 da aplicação GREat Mute não obteve bons resultados. A principal questão é a grande geração de falsos positivos.

Os mesmos são decorrentes da falsa identificação, por parte da aplicação, de que o usuário está realmente cumprindo um compromisso cadastrado em sua *Google agenda*.

Além disso, a aplicação tem diversos problemas devido a falta de sincronização aplicação-agenda quando em *background* (e.g., quando o dispositivo se encontra em modo de descanso ou bloqueado).

Na Tabela 26 são apresentados os resultados relacionados à questão: Qual o grau de consciência e controle do usuário sobre suas informações que são utilizadas pelo sistema?

**Tabela 26: Resultados - Proteção - GREAt Mute v1.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GCUD</b>	Grau de Compreensão no Uso dos Dados	X = Alto
<b>GPUD</b>	Grau de Permissão no Uso dos Dados	X = Baixo
<b>PPS</b>	Presença de Políticas de Segurança	X = Nenhuma

Por fim, o grau de consciência e controle do sistema é considerado baixo pois, de acordo com os três aspectos levados em consideração para a sua avaliação, apenas um obteve resultados satisfatórios. Não dar ao usuário a possibilidade de alterar quais informações contextuais podem ser usadas pela aplicação e o momento de uso, causa no usuário final, de acordo com as pessoas que realizaram o teste e as medidas coletadas, certo desconforto.

#### 4.2.2 GREAt Mute versão 2.0

Segundo indica o estudo de (SANTOS 2014), a versão 1.0 da GREAt Mute possui diversos erros de adaptação, decorrentes de mudanças contextuais. Que puderam ser confirmados com a avaliação feita e apresentada na Seção 5.2.1. Isso ocorre, pois, como a aplicação não faz uso de informações relativas à localização do usuário, o contexto muitas vezes não corresponde a situação real do utilizador.

Além disso, os resultados obtidos com a coleta das medidas definidas neste trabalho, também indicam a necessidade de melhorias, pois para a maioria das medidas, os resultados obtidos estão muito abaixo (e.g., menos da metade) daquilo que seria o ideal, de acordo com os valores de interpretação estipulados na definição das medidas.

Diante desse cenário, a aplicação foi evoluída, fazendo uso em sua nova versão, de informações provenientes do GPS do dispositivo móvel do usuário. Nessa versão, é

necessário que o usuário indique, utilizando um mapa fornecido pela própria aplicação, a localização do compromisso cadastrado na sua Google agenda.

Após a evolução da aplicação, novos testes foram realizados, utilizando os dispositivos *Samsung Galaxy S3* e *Samsung Galaxy S4*.

Assim como na seção anterior, os resultados obtidos com a coleta das medidas são apresentados de forma a servirem como respostas para as questões GQM. Cada questão é comentada de acordo com os resultados das medidas obtidas.

Na Tabela 27 são apresentados os resultados relacionados à questão: Qual o grau de disponibilidade do serviço provido pelo sistema?

**Tabela 27: Resultados - Disponibilidade - GREAt Mute v2.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GrD</b>	Grau de Disponibilidade	X = Alto
<b>TxErr</b>	Taxa de Erros	X = 1
<b>TAdapt</b>	Tempo de Adaptação	X = T = 63 milissegundos Dp = 112 milissegundos
<b>GDM</b>	Discrição dos mecanismos de segurança	N/A
<b>GIdB</b>	Grau de Independência da Bateria	X = Médio
<b>GInB</b>	Grau de influência sobre a bateria	X = Médio
<b>GInCE</b>	Grau de independência de componentes externos	X = Médio

Diferentemente da 1.0, como na versão 2.0 há interação com a aplicação para o uso do mapa, o usuário final tem a sensação de que o sistema “demora” mais. Essa sensação é associada a exigência da aplicação de que haja um contato direto aplicação-usuário e, com isso, há a falsa sensação de aumento no tempo do provimento do serviço.

Apesar disso, o resultado obtido com a coleta das medidas permanece o mesmo.

Na Tabela 28 são apresentados os resultados relacionados à questão: Qual o grau de integridade dos dados armazenados ou transmitidos pelo sistema?

**Tabela 28: Resultados - Integridade - GREat Mute v2.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GIDR</b>	Grau de Integridade dos Dados Transmitidos em Rede	X = Baixo
<b>GIDA</b>	Grau de Integridade dos Dados Armazenados	X = Baixo

Assim como na versão 1.0, a versão 2.0 não implementa nenhum mecanismo para garantia da integridade dos dados transmitidos pelas diferentes partes envolvidas na entrega do serviço. Também não há garantias de integridade para os dados armazenados no dispositivo. Como a localização dos compromissos é adicionada a aplicação pelo próprio usuário, no próprio dispositivo, preocupações com os mecanismos de integridade foram postergadas para futuras versões. Entretanto, por existir uma transmissão de dados em uma rede TCP/IP, considerou-se neste trabalho que mecanismos de integridade já estão implementados em camadas inferiores (COULOURIS 2011).

Na Tabela 29 são apresentados os resultados relacionados à questão: Qual o grau de esforço para realizar manutenção ou evolução do sistema?

**Tabela 29: Resultados - Manutenibilidade - GREat Mute v2.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GDoc</b>	Grau de Documentação	X = Médio
<b>GEMS</b>	Grau de Esforço para Realizar Manutenção do Sistema	X = Médio
<b>GEES</b>	Grau de Esforço para Realizar Manutenção ou Evolução do Código	X = Médio

Com relação ao grau de manutenibilidade, com a coleta das medidas é possível perceber que, mesmo com a existência de documentação, ainda há esforço extra para realizar a manutenção e evolução do sistema. Indicando assim a necessidade de uma melhor documentação, de acordo com os desenvolvedores que participaram da coleta destas medidas.

Na Tabela 30 são apresentados os resultados relacionados à questão: Qual o grau de confiabilidade no serviço provido pelo sistema?

**Tabela 30: Resultados - Confiabilidade - GREat Mute v2.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GrP</b>	Grau de Precisão	X = 0,72 ou 72%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,91 ou 91%
<b>GC</b>	Grau de Acurácia	X = 0,66 ou 66%
<b>GCEA</b>	Corretude na Execução da Adaptação	X = Médio
<b>GrE</b>	Grau de Especificidade	X = 87%
<b>GDFm</b>	Grau de Desempenho	X = 81%
<b>GA</b>	Grau de Adaptação	X = 92%
<b>GCAS</b>	Grau de Concordância	X = Médio
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Médio

Com relação ao grau de confiabilidade, diferentemente da versão 1.0, a versão 2.0 obteve resultados considerados bons. Ainda há a geração de falsos positivos, porém, o problema identificado e que deve ser tratado em versões futuras consiste apenas em um atraso no provimento do serviço. O atraso ocorre em relação ao tempo necessário ao provimento do serviço e distância do usuário em relação ao local do compromisso (distâncias de até 50 metros para o local do compromisso e horários de até 4 minutos depois do horário de início).

Além disso, é importante destacar que agora a aplicação tem menos problemas relacionados à sincronização aplicação-agenda quando em *background*.

Na Tabela 31 são apresentados os resultados relacionados à questão: Qual o grau de consciência e controle do usuário sobre as suas informações utilizadas pelo sistema?

**Tabela 31: Resultados - Proteção - GREat Mute v2.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GCUD</b>	Grau de Compreensão no Uso dos Dados	X = Alto
<b>GPUD</b>	Grau de Permissão no Uso dos Dados	X = Alto
<b>PPS</b>	Presença de Políticas de Segurança	X = Baseada em permissão

Por fim, o grau de consciência e controle do sistema é considerado alto, pois, diferentemente da versão anterior, por ter a possibilidade de alterar quais informações contextuais podem ser usadas pela aplicação e o momento de uso, a aplicação torna-se mais atraente para o usuário final.

### 4.2.3 GREat Mute versão 3.0

Nesta seção são apresentados os resultados obtidos pela coleta de dados na aplicação GREat Mute v3.0. Cada questão GQM, apresentada no Capítulo 3, é comentada de acordo com os resultados das medidas obtidas.

Mesmo diante da evolução realizada entre as versões 1.0 e 2.0, e das melhorias obtidas e constatadas com a aplicação das medidas, é possível identificar a necessidade de uma nova evolução. Essa necessidade pode ser fundamentada em dois fatos: os resultados obtidos ainda não são os melhores possíveis, dado os valores de interpretação para cada medida coletada e, por fazer com que o usuário indique de maneira explícita a localização do seu compromisso em um mapa, o sistema perde em ubiquidade.

Com isso, a versão 3.0 continua fazendo uso do mesmo conjunto de informações contextuais, porém, a localização do compromisso é obtida pela aplicação de maneira automática, retirando esse dado da *Google agenda*.

Para a coleta das medidas na nova versão foi necessário realizar novos testes. Ao todo, 150 novos casos de teste foram feitos com dois usuários, utilizando os mesmos dispositivos dos testes das versões anteriores (*Samsung Galaxy S3* e *Samsung Galaxy S4*).

Na Tabela 32 são apresentados os resultados relacionados à questão: Qual o grau de disponibilidade do serviço provido pelo sistema?

**Tabela 32: Resultados - Disponibilidade - GREAt Mute v3.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GrD</b>	Grau de Disponibilidade	X = Alto
<b>TxErr</b>	Taxa de Erros	X = 1
<b>TAdapt</b>	Tempo de Adaptação	X = T = 63 milissegundos Dp = 112 milissegundos
<b>GDM</b>	Discrição dos mecanismos de segurança	X = Baixo
<b>GIdB</b>	Grau de Independência da Bateria	X = Médio
<b>GInB</b>	Grau de influência sobre a bateria	X = Médio
<b>GInCE</b>	Grau de independência de componentes externos	X = Médio

Os resultados obtidos para o grau de disponibilidade são semelhantes aos resultados das versões 1.0 e 2.0. Porém, diferentemente da versão 2.0, não há mais a necessidade de interação direta do usuário com o mapa da aplicação. Isso fez com que a sensação de disponibilidade do sistema, por parte do usuário final, parecesse mais imediata. Implicando em uma melhoria não captada pela coleta das medidas.

Na Tabela 33 são apresentados os resultados relacionados à questão: Qual o grau de integridade dos dados armazenados ou transmitidos pelo sistema?

**Tabela 33: Resultados - Integridade – GREAt Mute v3.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GIDR</b>	Grau de Integridade dos Dados Transmitidos em Rede	X = Médio
<b>GIDA</b>	Grau de Integridade dos Dados Armazenados	X = Médio

Nesta nova versão houve diferenças nas medidas de grau de integridade coletadas em relação às versões anteriores. Essas diferenças ocorreram por causa da nova implementação dos recursos de configuração da aplicação, os quais foram alterados devido à modificações feitas no serviço de calendário da Google.

Durante o desenvolvimento e teste desta versão da GREat Mute, a API disponibilizada pela Google para uso por desenvolvedores não encontrava-se em sua versão final (informação disponibilizada pela própria empresa no seu site <https://developers.google.com/google-apps/calendar/>). Com isso, não havia garantias que os mecanismos disponibilizados no momento seriam mantidos, de forma a não afetar o comportamento futuro das aplicações que os utilizam. Com isso, nesta nova versão, os mecanismos de comunicação entre usuário e aplicação foram afetados devido a essas modificações e obtiveram como resultado da coleta valores intermediários (ou medianos).

A implementação de mecanismos de integridade não foi feita nesta nova versão, reforçando que a aplicação funciona como um serviço web que utiliza TCP/IP nas camadas inferiores.

É importante ressaltar que com o uso de permissões *Android*, é possível fazer com que a aplicação faça sincronização de contas automaticamente, de tempos em tempos, garantindo que os dados conhecidos pela aplicação, dentro do dispositivo, são condizentes com os dados que estão na agenda do *google*, associada a conta de *email* do usuário. Logo, do ponto de vista do usuário final, há melhorias nesse aspecto.

Na Tabela 34 são apresentados os resultados relacionados à questão: Qual o esforço necessário para realizar manutenção ou evolução do sistema?

**Tabela 34: Resultados - Manutenibilidade - GREat Mute v3.0**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GDoc</b>	Grau de Documentação	X = Alto
<b>GEMS</b>	Grau de Esforço para Realizar Manutenção do Sistema	X = Baixo
<b>GEES</b>	Grau de Esforço para Realizar Manutenção ou Evolução do Código	X = Baixo

Assim como na versão anterior (2.0) com coleta das medidas é possível perceber que, com a realização de melhorias na documentação, foi possível reduzir o esforço para realizar a manutenção e evolução do sistema.

Na Tabela 35 são apresentados os resultados relacionados à questão: Qual o grau de confiabilidade no serviço provido pelo sistema?

**Tabela 35: Resultados - Confiabilidade - GREat Mute v3.0**

Id	Nome	Resultado
<b>GrP</b>	Grau de Precisão	X = 0,68 ou 68%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,91 ou 91%
<b>GC</b>	Grau de Acurácia	X = 0,71 ou 71%
<b>GCEA</b>	Corretude na Execução da Adaptação	X = Alto
<b>GrE</b>	Grau de Especificidade	X = 87%
<b>GDFm</b>	Grau de Desempenho	X = 91%
<b>GA</b>	Grau de Adaptação	X = 92%
<b>GCAS</b>	Grau de Concordância	X = Médio
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Alto

Com relação ao grau de confiabilidade, não houve mudanças significativas em relação a geração de falsos positivos ou falsos negativos. As melhorias estão em: funcionar de maneira independente ou autônoma, pois não necessita mais da interação explícita usuário-mapa e por funcionar perfeitamente em *background* (e.g., quando o dispositivo se encontra em modo de descanso ou bloqueado).

Na Tabela 36 são apresentados os resultados relacionados à questão: Qual o grau de consciência e controle do usuário sobre suas informações que são utilizadas pelo sistema?

**Tabela 36: Resultados - Proteção - GREat Mute v3.0**

Id	Nome	Resultado
<b>GCUD</b>	Grau de Compreensão no Uso dos Dados	X = Alto
<b>GPUD</b>	Grau de Permissão no Uso dos Dados	X = Alto
<b>PPS</b>	Presença de Políticas de Segurança	X = Baseada em permissão

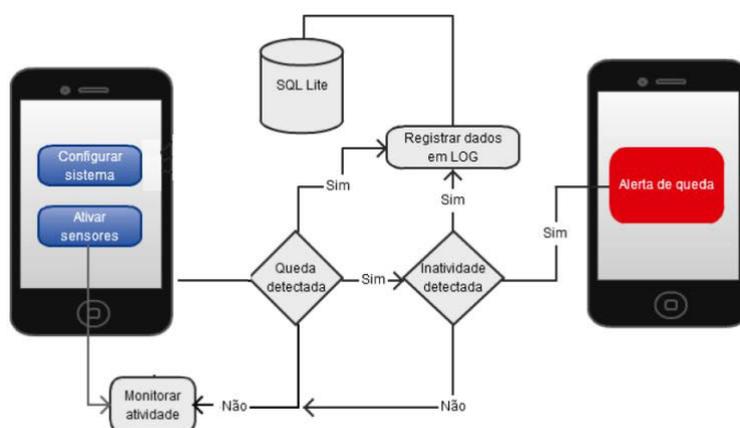
Por fim, para o grau de consciência e controle do sistema, não houve alterações em relação a coleta da versão anterior (2.0)

### 4.3 fAlert

O *fAlert* (*FallAlert*) é um protótipo de aplicação de *software*, para *smartphones* com sistema operacional *android*. A aplicação atua monitorando de forma inteligente pessoas que necessitam de cuidados especiais, como idosos e deficientes físicos (PIVA et al 2014).

O sistema consiste em uma aplicação ubíqua que, em tempo real, utiliza os dados fornecidos pelos sensores presentes nos *smartphones* para detectar possíveis quedas, tais como acelerômetro, microfone e magnetômetro. A aplicação permite a configuração de telefones, e-mails e endereços de emergência para que, uma vez detectado um padrão de queda, um alerta seja enviado para uma entidade pré-configurada.

A Figura 20 mostra o esquema de representação do sistema de detecção de quedas, mostrando as configurações do dispositivo e o processo de ativação do monitoramento dos dados dos sensores.



**Figura 18: Representação do funcionamento do sistema fAlert (PIVA et al 2014)**

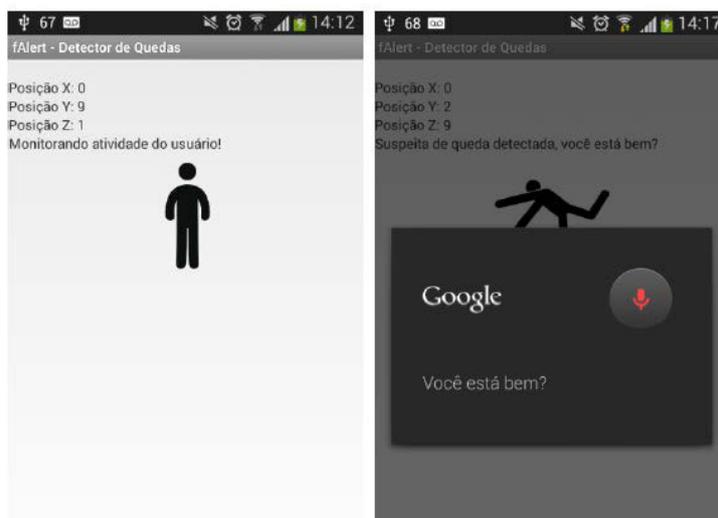
O protótipo de aplicação *fAlert* foi desenvolvido com base no dispositivo *Samsung Galaxy S3* que possui a seguinte configuração, apresentada na Tabela 37.

**Tabela 37: Configuração técnica do aparelho utilizado [PIVA et al 2014]**

Dimensões	136.5 x 70.6 x 8.6 mm
Tamanho da Tela	4.8 polegadas de 720 x 1280 pixels
Peso	133g
Tecnologias sem Fio	Wifi 802.11 b/g/n, 2.5G, 3G, Bluetooth, NFC e GPS
Armazenamento	16GB
Memória RAM	1GB
Processador	Quad-core 1.4 GHz Cortex-A9
Bateria	Li-Ion 2100 mAh
Qualidade do Som	-90.3 dB e -92.6 dB
Sensores	Acelerômetro, microfone, giroscópio, proximidade, magnetômetro, barômetro
Sistema Operacional	Android versão 4.3 JellyBean

A aplicação possui uma interface gráfica intuitiva, apresentada na Figura 21, possuindo apenas duas funcionalidades principais, que facilitam a compreensão e o uso do sistema. A primeira funcionalidade é utilizada para as configurações de contato e idioma, permitindo a seleção de três idiomas: Português do Brasil, Espanhol e Inglês. A segunda funcionalidade é utilizada para a ativação do monitoramento das atividades do usuário por meio dos sensores do dispositivo, que somente estará disponível após a configuração dos dados de contato.

A interface para o monitoramento exibe informações dos três eixos (x,y e z) do acelerômetro, permitindo, assim, mostrar para o usuário que os dados estão sendo obtidos corretamente à medida que ele se move.



**Figura 19: Interface gráfica do fAlert**

### 4.3.1 Mecanismo para a Detecção de Quedas

As soluções baseadas em *smartphones* utilizam dois possíveis mecanismos para a detecção de quedas: reconhecimento de padrões ou *threshold* (FUDICKAR et al 2014). No primeiro caso, o sistema se comunica com sensores externos ao dispositivo, que podem auxiliar o reconhecimento de padrões que identifiquem as atividades do usuário. Assim, é possível fazer com que o sistema aprenda sobre o comportamento do usuário e seja capaz de diferenciar entre situações cotidianas e de perigo. No segundo caso, o sistema utiliza apenas os sensores presentes no *smartphone* para o processo de detecção. Com isso, é possível baratear o custo e prover o serviço sem a necessidade de outro hardware. Sendo este segundo o mecanismo adotado pelo *fAlert*.

O *fAlert* utiliza os dados obtidos por meio do vetor de aceleração linear, produzido pelo acelerômetro presente no dispositivo móvel e identificado pela Equação 1 apresentada a seguir:

$$|A_T| = \sqrt{|A_x|^2 + |A_y|^2 + |A_z|^2}$$

#### **Equação 1: Fórmula para Cálculo da Aceleração Linear do Dispositivo**

Com base na fusão da informação dos três eixos (x,y e z) do acelerômetro e os dados obtidos pela leitura do magnetômetro, a aplicação pode identificar os casos onde o usuário se encontra em queda livre, por um curto período de tempo, seguido da ocorrência de impacto. Caracterizando assim, uma possível ocorrência de queda.

Para que a queda seja confirmada, verifica-se se ocorreu uma variação brusca da aceleração do dispositivo bem como a posição na qual o dispositivo se encontra. Caso o

dispositivo esteja em um ângulo entre 0 e 45 graus em relação ao chão, a possibilidade de queda é identificada.

Depois de identificada uma possível queda, o sistema ativa o *Google Speech Recognizer* para verificar se o usuário se encontra bem ou não. Isso é feito por meio do lançamento de uma janela de diálogo, onde o usuário precisa interagir com o dispositivo utilizando a voz em um tempo mínimo de 25 segundos. A interação deve ocorrer em voz alta, por meio do uso de palavras-chave.

No caso do usuário estar bem, a aplicação volta ao seu estado inicial de monitoramento das atividades do usuário. Caso contrário, quando não há reconhecimento de voz ou o usuário fala que não está bem, o sistema gera o alerta, caracterizado pela realização de uma ligação telefônica para um contato previamente configurado, na opção de menu “*Configurar Sistema*”.

De acordo com os sensores presentes no dispositivo do usuário, há variação nas informações disponíveis para a realização dos cálculos feitos pelo sistema e, conseqüentemente na precisão das detecções realizadas. Os dados disponíveis de acordo com cada sensor são apresentados na Tabela 38.

**Tabela 38: Informações providas pelos Sensores Utilizados para Detecção**

Informações Utilizadas para a Detecção	Acelerômetro	Acelerômetro e Magnetômetro	Acelerômetro, Magnetômetro e Microfone
Queda Livre	X	X	X
Impacto	X	X	X
Movimento Pós Impacto	X	X	X
Estabilidade	X	X	X
Orientação do Dispositivo		X	X
Confirmação Sonora			X

#### 4.2.4 Testes Realizados

Os testes realizados para a detecção de quedas levam em consideração diversas atividades cotidianas do usuário que podem ser detectadas como possíveis quedas, são exemplos: andar, correr, sentar, levantar e deitar.

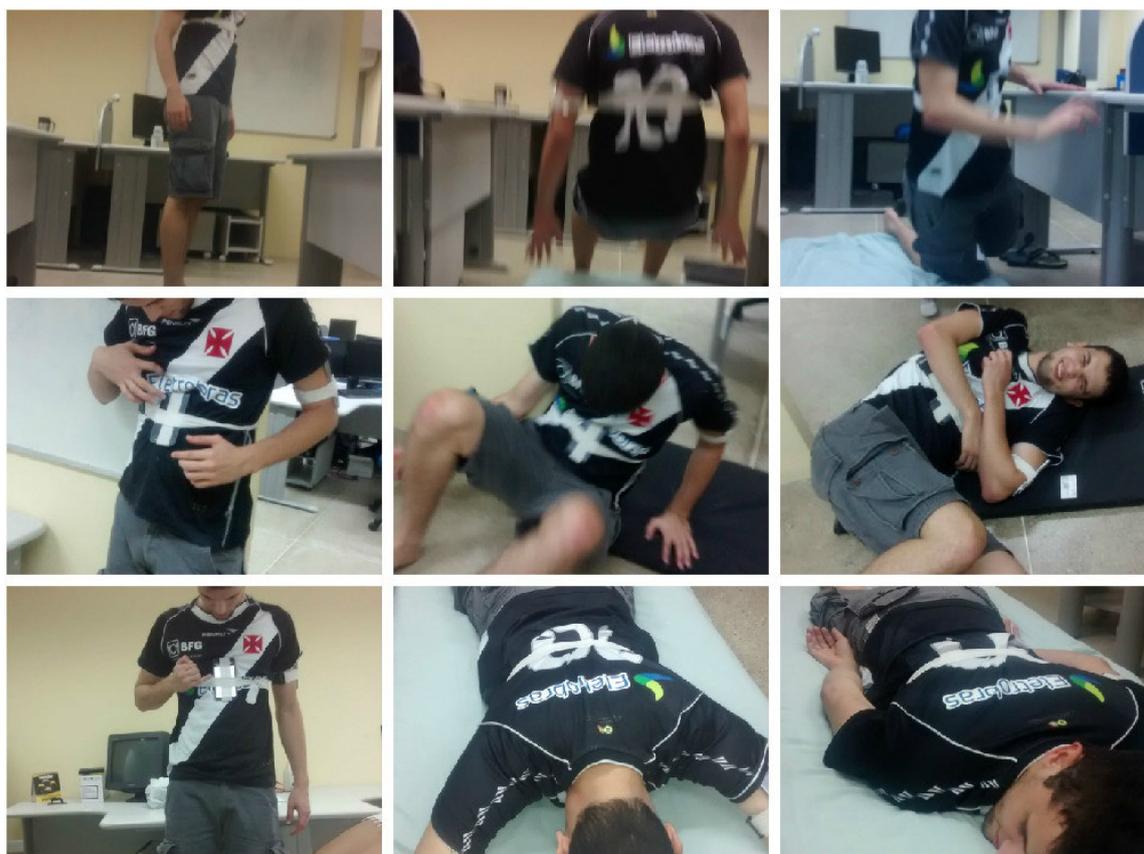
Para produzir resultados mais precisos foram realizados 1.140 testes em quatro usuários jovens e saudáveis, utilizando os dispositivos *Samsung Galaxy S3 I9300*, *Samsung Galaxy S3 Slim Duos* e *LG G2*, colocados no peito, braço e bolso, respectivamente,

ilustrados na Figura 22. Nos testes específicos para os casos de queda foi utilizado um colchão de espuma localizado no chão.



**Figura 20: Dispositivos espalhados pelo corpo do usuário**

É importante salientar também que as quedas foram divididas em categorias: quedas frontais, laterais e para trás. Os testes foram realizados em um ambiente fechado bem como em um ambiente aberto, como ilustram as imagens na Figura 23.



**Figura 21: Casos de queda**

A quantidade de testes realizados por pessoa é apresentada de maneira categorizada na Tabela 39.

**Tabela 39: Quantidade de testes por categoria**

Atividades	<i>Indoor</i>	<i>Outdoor</i>	Frontal	Lateral	Para trás
Andar	15	15	-	-	-
Correr	0	30	-	-	-
Pular	30	0	-	-	-
Subir e Descer Escadas	30	0	-	-	-
Sentar Lento	30	0	-	-	-
Sentar Rápido	30	0	-	-	-
Deitar Lento	30	0	10	10	10
Deitar Rápido	30	0	10	10	10
Cair Lento	30	0	10	10	10
Cair Rápido	30	0	10	10	10

Como pode-se observar na Tabela 39, para cada atividade, 30 testes foram realizados. Esse número foi considerado o suficiente, pois com essa mesma quantidade foi possível definir o *threshold* para o algoritmo de detecção. Para as ações de deitar e cair os testes foram subdivididos de acordo com a direção do movimento.

A princípio, para a avaliação do *fAlert*, é utilizado apenas um subconjunto das medidas de qualidade apresentadas no Capítulo 3. Esse subconjunto foi escolhido com base em diversos trabalhos (e.g.) encontrados na literatura que indicam quais aspectos são importantes e devem ser levados em consideração ao se avaliar esse tipo de sistema.

Adaptadas para a avaliação do *fAlert*, as principais medidas utilizadas são apresentadas em detalhes na Tabela 40. A coleta de todas as medidas foi realizada apenas na última versão do sistema (versão com os sensores de acelerômetro, magnetômetro e microfone) e os resultados são apresentados no fim desta seção.

**Tabela 40: Medidas utilizadas para avaliar o fAlert**

<i>Id</i>	<i>Nome</i>	<i>Descrição</i>	<i>Função de Medição</i>	<i>Interpretação</i>
GrP	Grau de Precisão	Representa a capacidade do sistema de detectar somente quedas, avaliando, para tal, a fração de contextos sentidos ou inferidos que são relevantes para o sistema.	$\frac{P}{P + FP}$	Quanto mais próximo de 1, melhor.
GrR	Grau de Sensibilidade ( <i>Recall</i> )	Representa a capacidade do sistema de detectar quedas, avaliando, para tal, a fração de contextos relevantes para o sistema que foram corretamente sentidos ou inferidos.	$\frac{P}{P + FN}$	Quanto mais próximo de 1, melhor.
GrD	Grau de Disponibilidade	Representa a capacidade do sistema de prestar serviço quando requisitado.	$\frac{TmF}{TmF + TmR}$	Quanto mais próximo de 1, melhor.
GrE	Grau de Especificidade	Representa a fração entre os alertas não gerados, decorrentes da ausência de quedas e a soma destes com o total de alertas gerados para quedas que não ocorreram, complementando assim, informações obtidas por meio da coleta do <i>Grau de Precisão</i> .	$\frac{N}{N + FP}$	Quanto mais próximo de 1, melhor.
GD Fm	Grau de Desempenho	Representa a média ponderada entre o <i>Grau de Precisão</i> e o <i>Grau de Sensibilidade (Recall)</i> .	$2 * \left( \frac{\frac{P}{P+FP} * \frac{P}{P+FN}}{\frac{P}{P+FP} + \frac{P}{P+FN}} \right)$	Quanto mais próximo de 1, melhor.
GC	Grau de Acurácia	Representa a fração de quedas corretamente sentidos ou inferidos	$\frac{P + N}{FP + P + FN + N}$	Quanto mais próximo de 1, melhor.

TmF e TmR representam respectivamente o tempo médio entre falhas e o tempo médio para reparar as falhas. P representa a ocorrência de uma queda detectada corretamente pelo sistema. Por outro lado, N representa que não houve queda e o sistema não detectou. Já FP representa a detecção de uma queda que não ocorreu e FN representa que houve queda, mas o sistema não identificou. Esses critérios, quando combinados, podem fornecer informações importantes sobre a acurácia do sistema, precisão, sensibilidade e outros.

Alguns dos resultados obtidos estão ilustrados nas Figuras 24 a 29. Os gráficos apresentados nessas figuras representam medidas que impactam no grau de confiança do sistema *fAlert*.

#### 4.2.4 Interpretação dos Resultados

Nesta seção são apresentados os resultados obtidos com a coleta das medidas na aplicação *fAlert*. Com a análise dos resultados alcançados é possível interpretar as questões e o objetivo de medição definidos nesta pesquisa.

O valor obtido para cada uma das medidas coletadas nas três configurações utilizadas, e para as três diferentes posições nas quais os dispositivos se encontravam no corpo do usuário, é apresentado graficamente nas Figuras 24-29.

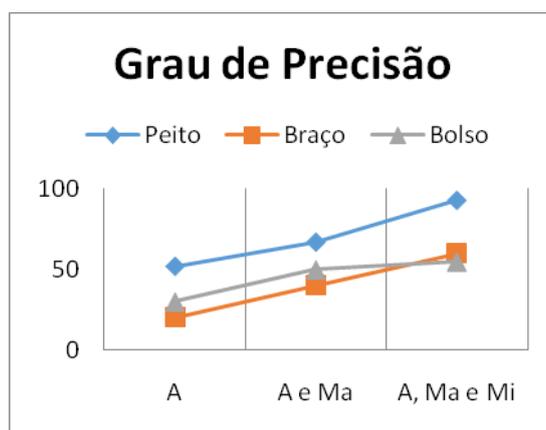


Figura 22: Resultados obtidos para a medida precisão

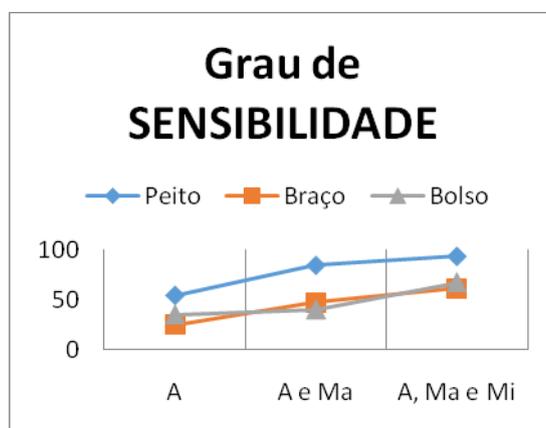
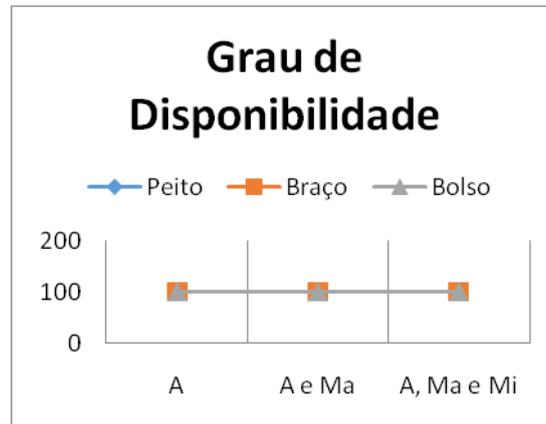
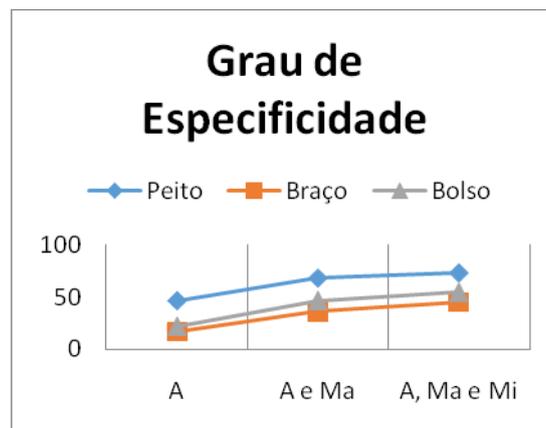


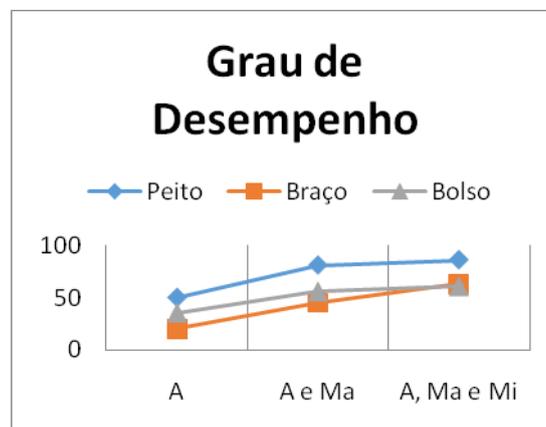
Figura 23: Resultados obtidos para a medida sensibilidade



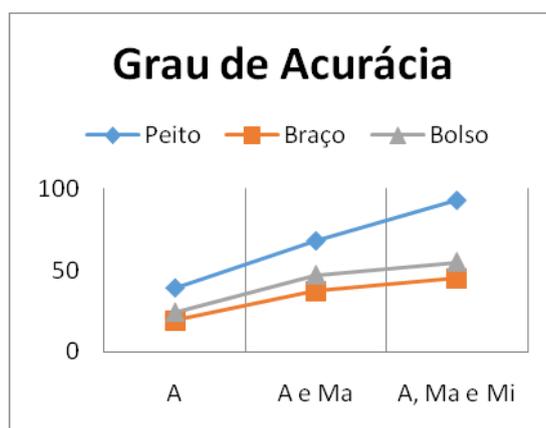
**Figura 24: Resultados obtidos para a medida disponibilidade**



**Figura 25: Resultados obtidos para a medida especificidade**



**Figura 26: Resultados obtidos para a medida grau de desempenho**



**Figura 27: Resultados obtidos para a medida acurácia**

Na primeira configuração (apenas o sensor acelerômetro), é possível perceber que o número de falsos positivos (FP) gerados pelo sistema é significativo. Essa identificação de FP é facilmente percebida porque a função de medição de GrP possibilita ao avaliador identificar a ocorrência de alertas para situações que não são quedas (e.g. sentar, levantar, deitar).

Além da quantidade significativa de falsos positivos gerados, esta primeira configuração gerou também um número considerável de falsos negativos. Nesse caso, independentemente da posição do dispositivo, no pior caso, 60% das quedas não foram detectadas, contribuindo negativamente para a medida de *Sensibilidade*. Os resultados obtidos reforçam a necessidade do uso da combinação de outros sensores do dispositivo.

Na segunda configuração, utilizando os sensores de acelerômetro e microfone, os valores coletados para as medidas foram melhores. O número de falsos positivos reduziu em aproximadamente 30% para a maioria das medidas. Porém, não alcançaram os mesmos níveis dos sistemas de detecção de quedas encontrados nos trabalhos relacionados. Além disso, falsos positivos foram gerados para as ações de sentar, levantar e deitar.

Ainda assim, os resultados obtidos são animadores pois, nos demais trabalhos encontrados, para que esse nível de precisão fosse atingido eram necessárias, muitas vezes, informações obtidas por sensores externos.

Finalmente, na terceira configuração (acelerômetro, microfone e magnetômetro) foi possível eliminar as ações de sentar e levantar, pois não houve a geração de falsos positivos. Para as ações de deitar e cair, os valores coletados com as medidas obtiveram melhores valores do que as duas primeiras configurações, aumentando a precisão e a sensibilidade do sistema.

Por fim, com a distribuição dos dispositivos pelo corpo do usuário, foi possível identificar que o peito é a melhor posição para a alocação do *smartphone* para a detecção de

quedas. Por manter um padrão de movimentos mais estável, independentemente do tipo de queda, o algoritmo de detecção funciona melhor neste caso.

Situações de queda livre seguidas de impacto são bem identificadas pelo dispositivo alocado no braço, porém, a orientação do dispositivo após o impacto e o ângulo formado entre o celular e o chão influenciam o algoritmo erroneamente.

Quando alocado no bolso, o sistema também gera uma quantidade significativa de falsos positivos. Quando o usuário apenas senta bruscamente o sistema identifica a queda livre e o impacto, após isso, dependendo da altura do usuário, o celular se aloca no bolso de maneira horizontal, caracterizando situações onde a orientação do dispositivo corresponde a uma queda. Além disso, quando a detecção ocorre de maneira correta, a distância da boca até o bolso impede a confirmação sonora, causando atrasos ou erros no processo de detecção.

Seguindo o mesmo método de apresentação dos resultados obtidos apresentados na seção anterior (*GREat Mute*), para a versão do *fAlert*, com o uso dos três sensores e dispositivo alocado ao peito (melhor configuração e posição de alocação), as respostas para as questões GQM são apresentadas nas Tabelas 41 a 45.

A Tabela 41 apresenta os resultados para a questão: Qual o grau de disponibilidade do serviço provido pelo sistema?

**Tabela 41: Resultados - Disponibilidade - fAlert**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GrD</b>	Grau de Disponibilidade	X = Alto
<b>TxErr</b>	Taxa de Erros	Não coletada
<b>TAdapt</b>	Tempo de Adaptação	X = T = 3 segundos
<b>GDM</b>	Discrição dos mecanismos de segurança	X = Baixo
<b>GIdB</b>	Grau de Independência da Bateria	X = Alto
<b>GInCE</b>	Grau de independência de componentes externos	X = Alto

O grau de disponibilidade do *fAlert* em sua configuração com o uso de três sensores pode ser considerado alto pois, o sistema encontrou-se disponível em mais de 75% dos casos de teste. Além disso, o sistema não é afetado por índices baixos de bateria. E, de

acordo com os usuários, o tempo de adaptação, mesmo sendo em segundos, é tolerável dado a sensibilidade das circunstâncias de contexto (e.g., 3 segundos é um tempo aceitável para um sistema que consegue distinguir ações de deitar e cair).

A Tabela 42 apresenta os resultados para a questão: Qual o grau de integridade dos dados transmitidos ou armazenados pelo sistema?

**Tabela 42: Resultados - Integridade - fAlert**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GIDR</b>	Grau de Integridade dos Dados Transmitidos em Rede	N/A
<b>GIDA</b>	Grau de Integridade dos Dados Armazenados	N/A

Com relação ao grau de integridade, o fAlert não armazena dados relativos ao processo de detecção. As verificações são feitas de maneira contínua, sem aprendizado ou armazenamento de dados. Logo, não há aplicabilidade para a medida GIDA. O mesmo vale para a medida GIDR.

A Tabela 43 apresenta os resultados para a questão: Qual o esforço necessário para realizar manutenção ou evolução do sistema?

**Tabela 43: Resultados - Manutenibilidade - fAlert**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GDoc</b>	Grau de Documentação	X = Baixo
<b>GEMS</b>	Grau de Esforço para Realizar Manutenção do Sistema	X = Médio
<b>GEES</b>	Grau de Esforço para Realizar Manutenção ou Evolução do Código	X = Médio

Com relação a manutenibilidade, com a coleta das medidas é possível perceber que, mesmo com a inexistência de documentação, o esforço extra para realizar a manutenção e evolução do sistema é menor que o esperado. A explicação associada a isto está na simplicidade do código. Sendo este resultado uma exceção ao esperado.

A Tabela 44 apresenta os resultados para a questão: Qual o grau de confiabilidade do sistema?

**Tabela 44: Resultados - Confiabilidade - fAlert**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GrP</b>	Grau de Precisão	X = 0,90 ou 90%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,93 ou 93%
<b>GC</b>	Grau de Acurácia	X = 0,73 ou 73%
<b>GCEA</b>	Corretude na Execução da Adaptação	X = Alto
<b>GrE</b>	Grau de Especificidade	X = 73%
<b>GDFm</b>	Grau de Desempenho	X = 86%
<b>GA</b>	Grau de Adaptação	X = 89%
<b>GCAS</b>	Grau de Concordância	X = 0,9
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Baixo

Segundo os usuários que realizaram os testes e as pessoas envolvidas no desenvolvimento do sistema, as medidas relativas à confiabilidade estão entre as mais importantes para a avaliação da confiança no funcionamento. Nesse aspecto, o *fAlert* obteve resultados satisfatórios pois, não exigindo custo extra com aquisição de novos equipamentos de hardware ou cursos para instalação e aprendizado do uso de software, é possível obter um sistema com taxas de acerto equivalentes a soluções, encontradas no mercado, como mais “robustas”.

Além disso, é importante destacar que, com a coleta das medidas é possível quantificar o ganho obtido com cada novo sensor associado ao processo de detecção.

A Tabela 45 apresenta os resultados para a questão: Qual o grau de consciência e controle do usuário sobre suas informações que são utilizadas pelo sistema?

**Tabela 45: Resultados - Proteção - fAlert**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GPUD</b>	Grau de Compreensão no Uso dos Dados	X = Alto
<b>GCUD</b>	Grau de Permissão no Uso dos Dados	X = Alto
<b>PPS</b>	Presença de Políticas de Segurança	X = Nenhuma

Por fim, o grau de consciência e controle possui resultados positivos. Apesar de não existir a implementação de nenhuma política de segurança, a aplicação é altamente satisfatória em relação ao uso dos dados do usuário e a permissão para a utilização destes.

Assim, os resultados obtidos com a coleta de todo o conjunto de medidas reforçam os resultados obtidos com as coletas parciais, indicando bons resultados para a configuração com o uso dos três sensores, quando comparado aos resultados obtidos em sistemas semelhantes, encontrados em trabalhos relacionados.

Mesmo diante desse cenário, é possível identificar oportunidades de evolução e, consequentemente, trabalhos futuros. Essa afirmação tem por base o fato de que as medidas apresentadas neste trabalho foram definidas e avaliadas tendo como foco apenas o domínio dos sistemas para dispositivos móveis, não havendo indícios de que possam ser aplicadas em outros domínios. Acarretando, possivelmente, na necessidade de novas avaliações.

Além disso, como mencionado em (FERREIRA ET AL 2015), o *fAlert* é uma prova de conceito (PoC, do inglês *proof of concept*). A PoC serve como modelo prático para confirmar os conceitos teóricos estabelecidos neste trabalho, porém não é possível fazer afirmações sobre a aplicação das medidas em sistemas reais.

## 4.4 GREAt Tour

O GREAt Tour é uma aplicação *Android* que serve como um guia de visitas do laboratório GREAt. A aplicação, sensível ao contexto, captura a posição do visitante em tempo real e exibe informações do local onde ele se encontra, bem como realiza adaptações com base nessa localização.

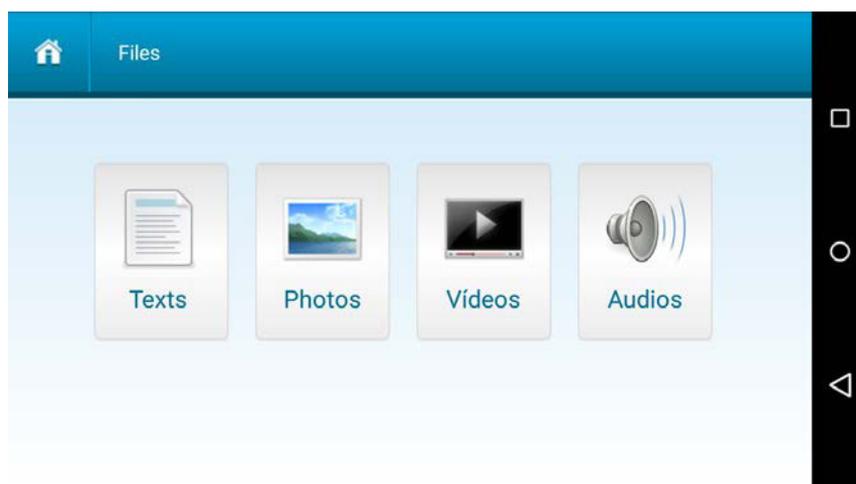


**Figura 28: Telas do GREAt Tour**

A aplicação encontra-se atualmente em estado de evolução para a versão 3.0. Nesta dissertação, a versão avaliada é a 2.0, que pode ser encontrada para download no seguinte link: <http://www.great.ufc.br/ctqs/index.php/producao/aplicacoes/78-artigosctqs/102-greattour>.

Em sua versão 2.0, a aplicação deve permitir que o usuário faça uma visita às alocações do GREAt (e.g., salas, laboratórios) de forma guiada, fornecendo informações a respeito dos ambientes a medida que o usuário se move.

Dentre as informações fornecidas pela aplicação encontram-se: vídeos, imagens, áudio, artigos, *papers*, dissertações, tese e outros, respeitando sempre o critério de exibí-las de acordo com o ambiente no qual o usuário se encontra (Figura 31).



**Figura 29: Informações providas pelo GREAt Tour**

Dentre os requisitos funcionais e não-funcionais, descritos no template de especificação de requisitos da aplicação, encontram-se:

- 1 – O sistema deve ser capaz de guiar um visitante pelas alocações do GREAt;
- 2 – O sistema deve ser capaz de fazer leituras de QRcodes, bem como de *tags* NFC, pois as informações de localidade são associadas pelos códigos presentes nas supracitadas tecnologias;
- 3 – O sistema deve fazer requisições ao banco de dados via *web service*, de forma rápida, para garantir uma experiência não cansativa da aplicação;
- 4 – O sistema deverá retornar informações referentes ao local onde o visitante se encontra;
- 5 – A aplicação deverá fornecer ao usuário um mapa do local atual, indicando em que ponto ele se encontra ao ler a localização;
- 6– O sistema deve ser de fácil utilização;
- 7– O sistema deve ser desenvolvido para Android OS;
- 8 – Os pacotes e classes da aplicação devem ser organizados de tal forma a minimizar o tempo de manutenção, caso necessário;
- 9 – Um novo usuário deve ser capaz de utilizar a aplicação desde que seja feito um cadastro prévio.

É possível perceber, observando requisitos como o 1, 4, 5 e 8 que, as medidas definidas o Capítulo 3, possuem relação direta com a garantia destes.

Diante desse cenário, as medidas definidas no capítulo anterior, são coletadas na versão 2.0 desta aplicação. Para tanto, foram realizados 50 casos de teste com os

dispositivos *Samsung Galaxy Dual Prime* e *Nexus 4*. Os resultados obtidos são apresentados a seguir como respostas para as questões GQM apresentadas no Capítulo 3.

A Tabela 46 apresenta os resultados para a questão: Qual o grau de disponibilidade do serviço provido pelo sistema?

**Tabela 46: Resultados - Disponibilidade - GREAt Tour**

Id	Nome	Resultado
<b>GrD</b>	Grau de Disponibilidade	X = Alto
<b>TxErr</b>	Taxa de Erros	X = 3 3 erros em 50 casos de teste
<b>TAdapt</b>	Tempo de Adaptação	X = T = 1 segundo
<b>GDM</b>	Discrição dos mecanismos de segurança	X = Médio
<b>GIdB</b>	Grau de Independência da Bateria	X = Alto
<b>GInCE</b>	Grau de independência de componentes externos	X = Baixo

O grau de disponibilidade do *GREAt Tour* em sua versão 2.0 não é considerado o melhor possível, segundo as medidas coletadas. O sistema indica a existência de erros, por algumas vezes houve erros de login, geração do conteúdo e até mesmo inicialização do sistema. Além disso, o mecanismo de autenticação não é discreto o suficiente, pois, tentativas de “logar” na aplicação demoravam mais do que o aceitável, segundo o usuário final. Por fim, a alta dependência da aplicação com relação a componentes externos (e.g., qrcode), torna a disponibilidade um aspecto ainda mais crítico, a ser tratado em futuras versões.

A Tabela 47 apresenta os resultados para a questão: Qual o grau de integridade dos dados armazenados ou transmitidos pelo sistema?

**Tabela 47: Resultados - Integridade - GREAt Tour**

Id	Nome	Resultado
<b>GIDR</b>	Grau de Integridade dos Dados Transmitidos em Rede	N/A
<b>GIDA</b>	Grau de Integridade dos Dados Armazenados	N/A

Assim como mencionado nos estudos de caso anteriores, os mecanismos de integridade são implementados em outras camadas do protocolo TCP/IP.

A Tabela 48 apresenta os resultados para a questão: Qual o grau de esforço necessário para realizar manutenção ou evolução do sistema?

**Tabela 48: Resultados - Manutenibilidade - GREat Tour**

Id	Nome	Resultado
<b>GDoc</b>	Grau de Documentação	X = Alto
<b>GEMS</b>	Grau de Esforço para Realizar Manutenção do Sistema	X = Baixo
<b>GEES</b>	Grau de Esforço para Realizar Manutenção ou Evolução do Código	X = Baixo

Com relação a manutenibilidade do sistema, a aplicação obteve os melhores resultados dentre as três aplicações analisadas. A documentação da aplicação possui um nível de conteúdo muito satisfatório e, segundo os desenvolvedores envolvidos, o esforço para evoluir e manter a aplicação é adequado.

A Tabela 49 apresenta os resultados para a questão: Qual o grau de confiabilidade do serviço provido pelo sistema?

**Tabela 49: Resultados - Confiabilidade - GREat Tour**

Id	Nome	Resultado
<b>GrP</b>	Grau de Precisão	X = 0,94 ou 94%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,98 ou 98%
<b>GC</b>	Grau de Acurácia	X = 0,92 ou 92%
<b>GCEA</b>	Grau de Corretude na Execução da Adaptação	X = Alto
<b>GrE</b>	Grau de Especificidade	X = 89%
<b>GDFm</b>	Grau de Desempenho	X = 96%
<b>GA</b>	Grau de Adaptação	X = 97%
<b>GCAS</b>	Grau de Concordância	X = 1
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Baixo

Apesar dos resultados negativos, obtidos com a coleta das medidas de disponibilidade, as respostas obtidas para as questões relativas à confiabilidade, são animadoras. A aplicação obteve resultados próximos do melhor possível, para a maioria das medidas. Indicando que, mesmo não provendo serviço sempre que requisitado, a conformidade com o esperado, quando não ocorrem problemas de disponibilidade, proporcionam um ótimo auxílio para o aumento do grau de confiança no funcionamento do sistema por parte do usuário.

A Tabela 50 apresenta os resultados para a questão: Qual o grau de consciência e controle do usuário sobre suas informações utilizadas pelo sistema?

**Tabela 50: Resultados - Proteção - GREat Tour**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GCUD</b>	Grau de Compreensão no Uso dos Dados	X = Alto
<b>GPUD</b>	Grau de Permissão no Uso dos Dados	X = Alto
<b>PPS</b>	Presença de Políticas de Segurança	X = Baseada em Permissão

Por fim, o grau de consciência e controle possui resultados positivos. Além da implementação do mecanismo de *login*, a aplicação é altamente satisfatória em relação ao uso dos dados do usuário e a permissão para a utilização destes. É importante destacar aqui o uso não apenas das informações do usuário da aplicação, mas, também, dos ambientes e pessoas envolvidas nele. Aspecto este não constatado pelas medidas, mas sim pelas pessoas envolvidas no momento da coleta.

## 4.5 Validação dos Resultados

Com o intuito de validar os resultados obtidos com a coleta das medidas, novos testes foram feitos, aproximadamente 50 dias após a realização dos testes anteriores, para as mesmas versões testadas anteriormente (GREat Mute v3.0, *fAlert* e GREat Tour v2.0). Para tanto, apenas um subconjunto das medidas apresentadas (as que respondem à questão Q1.1 – “Qual o grau de confiabilidade do serviço provido pelo sistema”) foi coletado.

### 4.5.1 GREat Mute

Para a aplicação GREat Mute, o número de testes realizados foi 50, por um único participante, portando o dispositivo LG G2. Os resultados obtidos são apresentados na Tabela 51.

**Tabela 51: Validação dos resultados - Confiabilidade - GREatMute**

<b>Id</b>	<b>Nome</b>	<b>Resultado</b>
<b>GrP</b>	Grau de Precisão	X = 0,82 ou 82%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,85 ou 85%
<b>GC</b>	Grau de Acurácia	X = 0,83 ou 83%
<b>GCEA</b>	Grau de Corretude na Execução da Adaptação	X = Alto
<b>GrE</b>	Grau de Especificidade	X = 79%
<b>GDFm</b>	Grau de Desempenho	X = 88%
<b>GA</b>	Grau de Adaptação	X = 89%
<b>GCAS</b>	Grau de Concordância	X = 0,8
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Baixo

Com relação ao grau de confiabilidade, os resultados dos novos testes permaneceram bons. Porém, houve diferenças com relação aos resultados obtidos anteriormente. Os valores de algumas medidas chegaram a variar em até 15%, para menos, quando comparado aos testes anteriores.

Novamente, houve a geração de falsos positivos. Dessa vez, no entanto, a causa se deu devido ao atraso no provimento dos dados necessários por parte da Google Agenda. Após uma análise e investigação do novo comportamento da aplicação (com relação ao atraso), foi constatado que houve mudanças na API do Google Calendar, entre as datas dos últimos testes e a data dos novos testes. Segundo desenvolvedores e

testadores da aplicação, as mudanças ocorridas justificam o surgimento dos novos casos de comportamento em não conformidade com o especificado.

Foi possível também constatar que, de fato, houve melhorias associadas ao funcionamento autônomo, por não mais necessitar da interação explícita e ao correto funcionamento em *background*, apesar do atraso para sincronização das informações.

#### 4.5.2 fAlert

Para a aplicação *fAlert*, o número de testes realizados foi 110, por um único participante, portando o dispositivo Samsung GalaxyS2. Os resultados obtidos são apresentados na Tabela 52.

**Tabela 52: Validação dos resultados - Confiabilidade - fAlert**

Id	Nome	Resultado
<b>GrP</b>	Grau de Precisão	X = 0,53 ou 53%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,57 ou 57%
<b>GC</b>	Grau de Acurácia	X = 0,56 ou 56%
<b>GCEA</b>	Grau de Corretude na Execução da Adaptação	X = Médio
<b>GrE</b>	Grau de Especificidade	X = 52%
<b>GDFm</b>	Grau de Desempenho	X = 58%
<b>GA</b>	Grau de Adaptação	X = 61%
<b>GCAS</b>	Grau de Concordância	X = 0,6
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Baixo

Diferentemente da coleta anterior, os resultados obtidos nessa nova coleta foram considerados baixos. Para a maioria das medidas, os valores diferem em até 40%, para menos.

Como principal fator associado a essa discrepância, aponta-se o uso de um aparelho celular diferente dos utilizados anteriormente, para o qual, a variação na calibração dos

sensores não havia sido considerada no momento do desenvolvimento do fAlert. Com sensores de sensibilidade diferente, o sistema não funciona conforme o especificado.

Com essa nova coleta das medidas foi possível perceber quão crítico é esse aspecto pois, ao não considerar diferentes sensibilidades de sensores, a aplicação reduz a sua precisão, especificidade e acurácia a pelo menos metade do que é alcançado por soluções semelhantes disponíveis no mercado.

Com isso, é possível obter novos e importantes focos de melhoria no que tange ao desenvolvimento dessa aplicação.

### 4.5.3 GREat Tour

Para a aplicação GREat Tour, o número de testes realizados foi 100, por um único participante, portando o dispositivo Moto G – segunda geração. Os resultados obtidos são apresentados na Tabela 53.

**Tabela 53: Validação dos resultados - Confiabilidade - GREat Tour**

Id	Nome	Resultado
<b>GrP</b>	Grau de Precisão	X = 0,89 ou 89%
<b>GrR</b>	Grau de <i>Recall</i>	X = 0,92 ou 92%
<b>GC</b>	Grau de Acurácia	X = 0,88 ou 88%
<b>GCEA</b>	Grau de Corretude na Execução da Adaptação	X = Alto
<b>GrE</b>	Grau de Especificidade	X = 86%
<b>GDFm</b>	Grau de Desempenho	X = 92%
<b>GA</b>	Grau de Adaptação	X = 94%
<b>GCAS</b>	Grau de Concordância	X = Alto
<b>GIAS</b>	Grau de Independência ou Autonomia	X = Médio
<b>GFB</b>	Grau de Funcionamento em <i>Background</i>	X = Baixo

Na aplicação GREat Tour, por sua vez, os resultados associados a confiabilidade permanecem altos e satisfatórios. Novamente, a aplicação obteve resultados próximos do

melhor possível, para a maioria das medidas. Isso reforça que a conformidade do serviço com aquilo que é esperado pelo usuário, quando não ocorrem problemas de disponibilidade, proporciona um ótimo auxílio para o aumento do grau de confiança no funcionamento.

## 4.6 Considerações Finais

Com o intuito de avaliar a utilidade das 25 medidas apresentadas no Capítulo 3, este capítulo apresentou os resultados da coleta delas em três diferentes aplicações. Com os resultados obtidos foi possível identificar pontos de melhoria nos três sistemas. Tais pontos foram apresentados e discutidos nas seções 4.2, 4.3 e 4.4, e possibilitaram o desenvolvimento de novas e melhores versões dos sistemas avaliados, no que tange a confiança do usuário em seu funcionamento. Evidenciando, com isso, indícios de que as medidas propostas nesta dissertação são, de fato, capazes de avaliar a característica de qualidade *confiança no funcionamento*.

## 5 CONCLUSÃO

Este trabalho de dissertação apresentou um conjunto de medidas de qualidade de *software* para a avaliação da confiança no funcionamento (i.e., *dependability*) de sistemas ubíquos. Essas medidas foram aplicadas em três sistemas para dispositivos móveis. Este capítulo apresenta na Seção 5.1 os resultados alcançados e a sua relevância e na Seção 5.2 as limitações e, por fim, na Seção 5.3, os trabalhos futuros.

### 5.1 Resultados Alcançados

Objetivando prover serviços em todo lugar e a qualquer momento de forma transparente, invisível e natural, os sistemas ubíquos fazem cada vez mais parte do nosso dia-a-dia (WEISER 1991) (SANTOS 2014) (SPÍNOLA 2007). Para serem aceitos e utilizados, é aconselhável que esse tipo de sistema possibilite ao usuário um grau satisfatório de confiança no seu funcionamento (SCOLTZ e CONSOLVO 2004) (SANTOS 2014). Para tanto, de acordo com (AVIZIENIS et al 2004) e como detalhado nos Capítulos 1 e 2, é necessário levar em consideração os seguintes atributos de qualidade: disponibilidade, confiabilidade, proteção, confidencialidade, integridade e manutenibilidade.

Sabendo que uma das possíveis técnicas para a avaliação e garantia da confiança no funcionamento é a realização de medições, o objetivo desta pesquisa foi identificar e definir um conjunto de medidas de qualidade de *software* para a avaliação da confiança no funcionamento de sistemas ubíquos com foco em aplicações para dispositivos móveis. Para isso, diversas atividades, apresentadas no Capítulo 1, foram realizadas no desenvolvimento deste trabalho com o intuito de alcançar o seu objetivo.

Inicialmente foi realizada uma revisão bibliográfica e, a partir dos resultados obtidos, foi percebida a necessidade da adaptação de medidas existentes e definição de novas medidas para a avaliação da confiança no funcionamento em sistemas ubíquos, apoiando-se, principalmente, nas diferenças entre esse tipo de sistema e os sistemas tradicionais.

Após a revisão bibliográfica, o método GQM foi utilizado para adaptar e definir medidas. Como apresentado no Capítulo 2, essa metodologia consiste em: planejamento, definição, coleta e interpretação. Essas quatro etapas foram realizadas e, para tanto, oito

reuniões, com oito especialistas distintos, foram realizadas. Tais especialistas são professores, pesquisadores e estudantes da pós-graduação no grupo de pesquisa GREAt (Grupo de pesquisa em Redes de computadores, Engenharia de software e sistemas). No total foram apresentadas neste trabalho vinte e cinco medidas de qualidade de *software* que avaliam 5 questões de um objetivo de medição. Essas medidas foram aplicadas em três estudos de caso.

Com os estudos de caso foi possível fazer a coleta das medidas, o que permitiu a identificação de pontos de melhoria no que tange ao desenvolvimento dos sistemas avaliados e, com isso, foi possível proporcionar um aumento no grau de confiança no funcionamento deles.

Diante deste cenário, este trabalho gerou a seguinte contribuição para medições de qualidade de *software* de sistemas ubíquos: um conjunto de vinte e cinco medidas de qualidade de *software* para a avaliação da confiança no funcionamento de sistemas ubíquos voltados para dispositivos móveis. Essas medidas foram apresentadas de forma a padronizar a sua nomenclatura, função de medição e métodos de coleta.

Dentro do grupo de pesquisa (GREAt) no qual este trabalho está vinculado, esta dissertação contribuiu para o projeto Maximum, onde o trabalho de (SANTOS 2014) também está inserido. Esta dissertação definiu medidas para a característica confiança no funcionamento que se encaixa dentro das características encontradas por (SANTOS 2014) para avaliar a qualidade de sistemas ubíquos.

Com os resultados apresentados anteriormente foi possível a aceitação dos seguintes trabalhos científicos:

- “Uma Proposta de Medidas de Qualidade para Avaliação da Confiança em Sistemas Ubíquos” **Ferreira, A. B.**, Braga, R. B., Andrade, R. M. C. (aceito no Workshop de Teses e Dissertações em Qualidade de Software do XIII Simpósio Brasileiro de Qualidade de Software - WTDQS/SBQS 2014)

Este trabalho foi apresentado no primeiro ano do mestrado, em um evento específico da área de qualidade de software. Nesse primeiro momento, houve a oportunidade de ouvir de especialistas sobre a relevância em fazer pesquisa na área de medições e dos desafios mediante o paradigma da computação ubíqua.

Mesmo estando no início da pesquisa, foi possível apresentar detalhes parciais sobre as medidas e os estudos de caso que seriam realizados e obter bons *feedbacks*.

- “Avaliação da Confiança em um Sistema *Android* para Detecção e Alerta de Quedas” **Ferreira, A. B.**, Piva, L. S., Braga, R. B., Andrade, R. M. C. (aceito no XX Simpósio Brasileiro de Sistemas Multimídia e Web - WebMedia 2014)

Apresentado após a defesa da proposta de dissertação, pré-requisito do programa de mestrado. Neste trabalho, o segundo autor, aluno de graduação em intercâmbio no departamento de computação da UFC, gerou *logs* durante o desenvolvimento e testes do *fAlert* que foram utilizados para a coleta das medidas já documentadas. Este trabalho foi fundamental como fruto inicial dos resultados alcançados nesta etapa do mestrado e este artigo foi indicado a Best Paper na categoria Short Paper na Trilha Principal do evento e recebeu o prêmio de Menção Honrosa. Por ser um artigo curto, apenas 4 páginas, diversos resultados não foram apresentados. Servindo estes para a concepção de outro artigo, apresentado posteriormente.

- “*fAlert*: Um Sistema *Android* para Monitoramento de Quedas em Pessoas com Cuidados Especiais” Piva, L. S., **Ferreira, A. B.**, Braga, R. B., Andrade, R. M. C. (aceito no Workshop de Ferramentas e Aplicações do XX Simpósio Brasileiro de Sistemas Multimídia e Web - WFA/WebMedia 2014)

A aplicação *fAlert*, concebida com o intuito de ser um dos estudos de caso desta dissertação, também originou um artigo. O feedback dado pela banca presente no evento, mesmo não sendo sobre as medidas de qualidade desta pesquisa, serviu para diversos aspectos relacionados a concepção deste trabalho e futuras contribuições.

- “Avaliação da *Confiança no Funcionamento* de Sistemas de Detecção e Alerta de Quedas” **Ferreira, A. B.**, Braga, R. B., Andrade, R. M. C. (aceito no VII Simpósio Brasileiro de Computação Ubíqua e Pervasiva- SBCUP 2015)

Este artigo é uma evolução do *short paper* aceito no WebMedia 2014. No primeiro, foram apresentados resultados de apenas duas medidas. Nesse segundo, seis medidas foram coletadas e interpretadas, gerando um volume mais significativo de resultados.

## 5.2 Limitações

É importante apontar limitações na execução deste trabalho tanto na escolha dos estudos de caso quanto na coleta das medidas.

Todas as aplicações escolhidas como estudos de caso foram desenvolvidas no próprio grupo de pesquisa do qual este trabalho faz parte. No entanto, as aplicações utilizadas nos estudos de caso fazem parte de um conjunto de aplicações que vem sendo avaliadas também em outros trabalhos, o que as tornam mais robustas, além do código e outros artefatos de desenvolvimento estarem disponíveis para os testes necessários.

Os desenvolvedores que participaram dos testes, apesar de também serem membros do mesmo grupo de pesquisa, foram orientados que o objetivo da avaliação estava relacionado com a evolução e melhoria dos sistemas. Logo, resultados positivos e negativos da parte deles eram igualmente importantes.

Por fim, apresenta-se como limitação, a falta de coleta das medidas de integridade. Os motivos para tal ausência são elencados no Capítulo 4, ao longo da discussão sobre os resultados obtidos com os estudos de caso.

## 5.3 Trabalhos Futuros

As medidas de qualidade de *software* apresentadas neste trabalho possibilitam a avaliação da confiança no funcionamento de sistemas ubíquos para dispositivos móveis. Consequentemente, essas medidas viabilizam também a aceitação e utilização desse tipo de sistema. Entretanto, ainda é preciso investigar desafios que ficaram em aberto, gerando as seguintes perspectivas de trabalhos futuros:

- Analisar se o conjunto de medidas apresentado pode ser utilizado para a avaliação da confiança no funcionamento de outros tipos de sistemas ubíquos. O conjunto de medidas apresentado neste trabalho auxilia a avaliação da confiança no funcionamento, porém, como foram definidas e avaliadas apenas em cenário de sistemas ubíquos que executam em dispositivos móveis, não há indícios de que possam ser aplicadas em outros domínios, acarretando na necessidade de novas avaliações e, possivelmente, novas medidas.
- Coletar as medidas em projetos reais de desenvolvimento de software. As medidas foram aplicadas em sistemas desenvolvidos em projetos de pesquisa, não refletindo detalhes particulares e eventuais do processo de

desenvolvimento de software (e.g., há membros na equipe de testes com aptidão para acompanhar os usuários no momento de coleta de medidas baseadas em “Análise do Usuário”?). Todos os sistemas avaliados nos estudos de caso são provas de conceitos (PoCs, do inglês *proof of concept*). As PoCs avaliadas servem como modelo prático para confirmar os conceitos teóricos estabelecidos nesta pesquisa, porém não é possível fazer afirmações sobre a qualidade das medidas quando coletadas em produtos reais.

- Coletar as medidas de integridade em sistemas onde hajam dados armazenados. As aplicações utilizadas nos estudos de caso ou não armazenavam dados (que pudessem ser avaliados quando à integridade) ou transmitiam dados via rede, usando o protocolo TCP/IP (onde, para as aplicações utilizadas, as garantias de integridade eram feitas em camadas inferiores da pilha de protocolo (COULOURIS 2011)).
- Realizar uma comparação mais aprofundada sobre a característica de confiança no funcionamento (i.e., *dependability*) e confiança (i.e., *trust*) e, eventualmente integrar no modelo de características apresentado em (SANTOS et al 2013).
- Realizar um estudo aprofundado sobre a área de qualidade do contexto (BRINGEL 2006), visto que as medidas definidas para a subcaracterística de confiabilidade levam em consideração informações provenientes de percepção contextual para o cálculo das suas funções de medição.
- Analisar a dependência entre as medidas.  
Assim como em (SANTOS 2014), é necessário verificar se o resultado de uma medida pode impactar ou complementar o resultado de outra, indicando assim uma dependência entre elas.

## REFERÊNCIAS BIBLIOGRÁFICAS

- Abi-Char, P., Mhamed, A., El-Hassan, B. and Mokhtari, M., “A Flexible Privacy and Trust Based Context-Aware Secure Framework”. In *Lecture Notes in Computer Science*, 2010.
- Abowd, G. D., Mynatt, E. D. and Rodden, T., “The Human Experience [of Ubiquitous Computing]”. In *Pervasive Computing*, IEEE, v. 1, n. 1, p. 48–57, 2002.
- Alvez M. e Vasques F., “Confiança no Funcionamento: Conceitos Básicos e Terminologia – Relatório Técnico”, Portugal, 1998.
- Araujo, R. B., “Computação Ubíqua: Princípios, Tecnologias e Desafios”. Em XXI Simpósio Brasileiro de Redes de Computadores, 2003.
- Avizienis, A., Vytautas Magnus Univ., Kaunas, Lithuania, Laprie, J.-C, Randell, B., Landwehr, C., “Basic concepts and taxonomy of dependable and secure computing”, In *IEEE Transactions on Dependable and Secure Computing*, 2004.
- Barbosa, S. D. J. and Silva, B. S. , “Interação Humano-Computador”, Elsevier, 2010.
- Basili, V. R., Caldiera, G., e Rombach, H. D., “The Goal Question Metric Paradigm”. In *Encyclopedia of Software Engineering*, v.2, p 527-532, 1994.
- Berard, E., “Metrics for Object-Oriented Software Engineering, an Internet”, In *comp software-eng*, 1995.
- Bringel Filho, M. J. R., “CxtBAC : Une Famille de Modèles de Contrôle d'accès Sensible au Contexte e pour les Environnements Pervasifs”, Tese de Doutorado – Universidade de Grenoble, 2006.
- Bublitz, F. M., “Infra-estrutura para o Desenvolvimento de Aplicações Cientes de Contexto em Ambientes Pervasivos”, Dissertação de Mestrado - Campina Grande, 2007.
- Chen, G., Kotz, D. “A Survey of Context-Aware Mobile Computing Research”. In *Dartmouth Computer Science Technical Report TR 2000-381*, , New Hampshire, USA, 2000.
- Cheng, B. H. C. “Software Engineering for Self-Adaptive Systems: A Research Roadmap”,

In LNCS, p. 1–26, 2009.

Cooper D. A., Birman K. P., "Preserving Privacy in a Network of Mobile Computers," EmIEEE Symposium on Research in Security and Privacy, 1995.

Coulouris G., Dollimore, J., Kindberg, T. "Distributed Systems: Concepts and Design", 5ª Edição, Addison-Wesley, 2011.

Davis, A., "Identifying and Measuring Quality in a Software Requirements Specification", In IEE Software Metric Symposium, pg. 141-152, 2010.

Debashis S., Amitava M., "Pervasive Computing: A Paradigm for the 21st Century," In Computer, pp. 25-31, 2003.

Dey A. K. "Understanding and Using Context". In Personal and Ubiquitous Computing, 5(1):4–7, 2001.

Ferreira, A. B., Braga, R. B., Andrade, R. M. C., "Avaliação da Confiança no Funcionamento de Sistemas de Detecção e Alerta de Quedas", VII Simpósio Brasileiro de Computação Ubíqua e Pervasiva, 2015.

Fogg, B. J., Tseng, H. , "The Elements of Computer Credibility", In Human Factors in Computing Systems Conference, 15-20, 1999.

Fudickar S., Karth C., Mahr P., Schnor B., "Fall-detection Simulator for accelerometers with in-hardware preprocessing", PETRA, 2012.

Goillau P., Kelly C., Boardman M., e Jeannot E., "Guidelines for Trust in Future". In ATM Systems: Measures, 2003.

Gresse, C., Hoisl, B., Wüst, J., "A process model for planning GQM-based Measurement", Technical Report STTI-95-04-E, 1995.

Gu T., H. K. Pung, e D. Q. Zhang. "A Bayesian Approach for Dealing with Uncertain Contexts". In Proceedings of the 2nd International Conference on Pervasive Computing, volume 176, 2004.

Gwizdka J. "What's in the context". In Proceedings of the Computer Human Interaction, 2000.

- Hamman R., Google Glass: saiba tudo sobre os óculos da Google, Disponível em: <<http://www.tecmundo.com.br/project-glass/25785-google-glass-saiba-tudo-sobre-os-oculos-da-google.htm#ixzz2zzLjAvLT>> Acessado em: 25 de Abril de 2014.
- Hughes, B., “Practical Software Measurement”. McGraw - Hill Maidenhead, 2000.
- ISO/IEC 25000, “Software Engineering - Software Product Quality Requirements and Evaluation (SQuaRE) – Guide to SQuaRE”. v. 2005.
- ISO/IEC 9126 (2001). “Software Engineering – Product Quality – Part 1”.
- Jia, L., Collins, M. and Nixon, P. , “Evaluating Trust-Based Access Control for Social Interaction”. In International Conference on Mobile Ubiquitous Computing, Systems, Services, and Technologies, UBICOMM 2009.
- Karvonen H., “Different aspects of trust in ubiquitous intelligent transportation systems”, In 28th Annual European Conference on Cognitive Ergonomics, pg311-314, 2010.
- Kemp, E. A., Johnson, R. S. and Thompson, A.-J., “Interface Evaluation for Invisibility and Ubiquity: An Example from E-learning”. In 9th ACM SIGCHI New Zealand Chapter’s International Conference on Human-Computer Interaction: Design Centered HCI, 2008.
- Kilinc C., Booth T., and Andersson K., “Wall Droid: Cloud Assisted Virtualized Application Specific Firewalls for the Android OS”. In 11th IEEE International Conference on Trust, Security and Privacy in Computing and Communications pages 877–883, 2012.
- Koscianski, A., Soares, M. dos S. , “Qualidade de software”. São Paulo: Novatec, 2007.
- Kryvinska, N.; Strauss, C.; Zinterhof, P. “Variated Availability Approach to the Services Manageable Delivering” In 5th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011.
- Langeheinrich M., “Privacy by Design – Principles of Privacy Aware Ubiquitous Systems”, In UBICOMP, LNCS 2201, pp 273 291, 2001.
- Lee, J., Yun, M. H., “Usability Assessment for Ubiquitous Services: Quantification of the Interactivity in Inter-Personal Services”, In IEEE International Conference on Management of Innovation & Technology (ICMIT). 2012.

- Leveson N. G., “Software Safety in Embedded Computer Systems”. In Communications of the ACM, , p. 35 – 45, 1991.
- Lima, E. R. R., Araújo, I. L., Santos, I. S., Oliveira, T. A., Monteiro, G. S., Costa, C. E. B., Santos,Z., Andrade, R. M. C. (2013). “GREat Tour: Um Guia de Visitas Móvel e Sensível ao Contexto”. In Webmedia 2013 - Workshop de Ferramentas e Aplicações, 2013.
- Lima, F. F. P., “SysSu - Um Sistema de Suporte para Computação Ubíqua”. Dissertação de Mestrado - Universidade Federal do Ceará, 2011.
- Lima, F. F. P., Rocha, L. S., Maia, P. H. M., Andrade, R. M. C. , “Uma Arquitetura Desacoplada e Interoperável para Coordenação em Sistemas Ubíquos”. In Software Components, Architectures and Reuse (SBCARS), 2011.
- Loureiro E., Ferreira G., Almeida H., Perkusich A., “Pervasive Computing: What Is It Anyway? Ubiquitous Computing: Design, Implementation and Usability”. Information Science Reference, p. 9-36, 2008.
- Marinho, F. G.; Andrade, R. M. C.; Werner, C.; Viana, W.; Maia, M. E. F.; Rocha, L. S.; Teixeira, E.; Ferreira, J. B.; Dantas, V. L. L.; Lima, F.; Aguiar, S. , “MobiLine: A Nested Software Product Line for the domain of mobile and context aware applications”. Science of Computer Programming, 2012.
- Marinho, F. G., Costa, A. L., Lima, F. F. P., Neto, J. B B., Dantas, V. L. L., Andrade, R. M. C., “Uma Proposta de Arquitetura para Linhas de Produto de Software Aninhadas no Domínio de Aplicações Móveis e Sensíveis ao Contexto”. In IV Simpósio Brasileiro de Componentes,Arquiteturas e Reutilização de Software, 2010.
- Miranda, R. A. V., “Provendo segurança e uma semântica de falhas consistente para a comunicação de objetos assíncronos distribuídos”, Dissertação de Mestrado, Campina Grande, 2010.
- Nielsen, J.,“Usability Engineering”,Academic Press, 1993.
- Nixon P., Lacey G. and Dobson S.,“Managing interactions in smart environments”, In Springer Verlag Press, pp. 243, 1999.
- Nixon P A , W. Wagealla, C. English, S. Terzis, “Security, Privacy and Trust Issues in Smart Environments” ,In The Global and Pervasive Computing Group Department of Computer and Information Sciences University of Strathclyde, Glasgow, Scotland, 2005.

- Noury N., Fleury A., Rumeau P., Bourke A. K., Rialle V., Lundy J. E., “Fall detection – Principles and Methods”, Proceedings of the 29th Annual International, 2007.
- Park, R. E., Goethert, W. B. and Florac, W. A. , “Goal-Driven Software Measurement”. A Guidebook. Software Engineering Institute, Carnegie Mellon University, 1996.
- Peres, D. R., “Modelo de Qualidade para Componentes de Software”. Dissertação de Mestrado - Universidade de São Paulo, 2011.
- PetrellD., E. Not, C. Strapparava, O. Stock,, M. Zancanaro. “Modeling Context is Like Taking Pictures”, In Proceedings of the Computer Human Interaction, Workshop on “The What, Who, Where, When, Why and How of Context-Awareness”, Hague, Holanda, 2000.
- Piva, L. S., Ferreira, A. B., Braga, R. B., Andrade, R. M. C. , “fAlert: Um Sistema Android para Monitoramento de Quedas em Pessoas com Cuidados Especiais”, Workshop de Ferramentas e Aplicações do XX Simpósio Brasileiro de Sistemas Multimídia e Web, 2014.
- Poppe, R., Rienks, R., Dijk, B. Van , “Evaluating the Future of HCI: Challenges for the Evaluation of Emerging Applications”. AI for Human Computing, p. 234–250, 2007.
- Poslad, S. , “Ubiquitous Computing: Smart Devices, Environments and Interactions”, Wiley.76, 2009.
- Preece, J.; Rogers, Y.; Sharp H.; Benyon, D.; Holland, S., Carey T. “Human Computer Interaction”. Reading, MA., Addison-Wesley, 1994.
- Pressman, R. S., “Software engineering: a practitioner’s approach”. 7th. ed. McGraw-Hill, 2009.
- Ranganathan, J. Al-Muhtadi, e R. H. Campbell. “Reasoning about Uncertain Contexts in Pervasive Computing Environments”. IEEE Pervasive Computing, 3(2):62–70, 2004
- Ranganathan, J., Al-Muhtadi, J., Biehl, J., Ziebart, B., Campbell, R. H., Bailey, B., “Towards a Pervasive Computing Benchmark”. In Third IEEE International Conference on Pervasive Computing and Communications Workshops, 2005.
- Rocha L.S., “CAEH: Um Método para Verificação de Modelos de Tratamento de Exceção Sensível ao Contexto em Sistemas Ubíquos”, Tese de Doutorado – Universidade

- Federal do Ceará, 2013.
- Rocha, L. S., “Dependabilidade em Sistemas Ubíquos”. Monografia de Qualificação- Universidade Federal do Ceará, 2010.
- Rocha, A. R. C., Souza, G. S., Barcellos M. C., “Medições de Software: Controle Estatístico de Processos”, PBQP Software, 2012.
- Rocha, L. S. et al. “Ubiquitous Software Engineering: Achievements, Challenges and Beyond”, In Brazilian Symposium on Software Engineering , 2011.
- Ruhe G., Rough, “Set based Data Analysis in Goal Oriented Software Measurement”, Third Symposium on Software Metrics, Berlin, March, IEEE Computer Society Press, p.10-19, 1994.
- Runeson, P.; Host, M., “Guidelines for Conducting and Reporting Case Study Research”. In Software Engineering. Empirical Software Engineering, 2008.
- Santos R. M., Oliveira K. M., Andrade R. M. C., Santos I. S., Lima E. R.; “A Quality Model for Human-Computer Interaction Evaluation in Ubiquitous Systems”. In: 6th Latin American Conference on Human Computer Interaction, 2013.
- Santos, R. M., “Características e Medidas de Software para Avaliação da Qualidade da Interação Humano-Computador em Sistemas Ubíquos”. Dissertação de Mestrado - Universidade Federal do Ceará, 2014.
- Satyanarayanan M., “Pervasive Computing: Vision and Challenges”. In IEEE Personal Communications, 2001.
- Scalet, D., Koscianski, A., Vostoupal, T. M., “Qualidade de software segundo normas internacionais”, 1999.
- Schmidt, A., “Implicit Human Computer Interaction Through Context”. Personal and Ubiquitous Computing, v. 4, n. 2, p. 191–199, 2000.
- Scholtz, J., Consolvo, S., “Toward a Framework for Evaluating Ubiquitous Computing Applications”.IEEE Pervasive Computing, 2004.
- Sherchan W, Nepal S., Paris C., “A survey of trust in social networks”, InJournal ACM Computing Surveys, 2014.

- Soligen, V. R.e Berghout, E., “The goal/question/metric method – A practical guide for quality improvement of software development”. Great Britain: Cambridge, McGraw-Hill, 1999.
- Söllner, M., Hoffmann, A., Hoffmann, H., Leimeister, J. M., “Towards a Theory of Explanation and Prediction for the Formation of Trust in IT Artifacts”, In Annual Workshop on HCI Research in MIS, Shanghai, Vol. 4, 2011.
- Söllner, M., Hoffmann, A., Hoffmann, H., Leimeister, J. M., “How to use behavioral research insights on trust for HCI system design”, In CHI'12 Extended Abstracts on Human Factors in Computing Systems (pp. 1703-1708),, 2012.
- Sommerville, I. , “Software Engineering”, 7th edition. Addison-Wesley, 2007.
- Sousa, B., Pentikousis, K. and Curado, M. .“UEF: Ubiquity Evaluation Framework”. In Lecture Notes in Computer Science, v. 6649 LNCS, p. 92–103, 2011.
- Spínola, R. O., Massollar, J. and Travassos, G., “Checklist to Characterize Ubiquitous Software Projects”. Anais do Simpósio Brasileiro de Engenharia de Software, p. 39–55, 2007.
- Stajeno F., “Security for Ubiquitous Computing”, Wiley Press, 2002.
- Stallings W. , “Network Security Essentials applications and standards”, 2000.
- Sun, T. and Denko, M. K., “Performance Evaluation of Trust Management in Pervasive Computing”, In International Conference on Advanced Information Networking and Applications, AINA, 2008.
- Tinghuai M., Yong-Deak K., Qiang M., Meili T., Weican Z., “Context-aware implementation based on CBR for smart home”. In Proceedings of IEEE International Conference on Wireless and Mobile Computing, Network and, vol4, pages 112–115, 2005.
- Treck D., “The evaluation of Qualitative Assessment Dynamics (QAD) methodology for managing trust in pervasive computing environments”. In Pervasive Computing and Applications (ICPCA), 2011.
- Viana, W. C. “Mobilidade ET sensibilidade au context e pour La gestion de documments multimédias personnels: CoMMedia”. Tese (Doutorado) — Université Joseph-Fourier -

Grenoble, 2010. Disponível em: <<http://hal.archives-ouvertes.fr/tel-00499550/>>.

Weiser M., “The Computer for the Twenty-First Century”, *Scientific American*, pp. 94-10, September 1991.

Yau, S. S., Wang, Y. and Karim, F., “Development of Situation-Aware Application Software for Ubiquitous Computing Environments”. In *Computer Software e Applications Conference*, 2002.

## Apêndice A – Formulário para traçar o perfil do usuário

O formulário a seguir foi utilizado antes da realização dos testes mencionados no estudo de uso apresentado no Capítulo 4, com o intuito de traçar o perfil dos usuários envolvidos.

1. Nome
2. Sexo
  - (1) Feminino
  - (2) Masculino
3. Idade (em anos)
4. Ocupação Profissional
5. Formação Acadêmica (graduando, graduado, mestrando, mestre, doutorando ou doutorado)
6. Qual seu grau de experiência no uso de dispositivos móveis (e.g., *smartphones*, *tablets*)?
  - (1)Muito Baixo (2)Baixo (3)Médio (4)Alto (5) Muito Alto
7. Já utilizou aplicações ubíquas (aplicações móveis e sensíveis ao contexto)?
  - (1)Sim (2)Não (3)Não sei
8. Qual sua experiência com aplicações ubíquas?
  - (1)Muito Baixo (2)Baixo (3)Médio (4)Alto (5) Muito Alto
9. Você já desenvolveu softwares?
  - (1)Sim (2)Não
10. Você já desenvolveu softwares ubíquos?
  - (1)Sim (2)Não

## Apêndice B – Questionário utilizado para avaliação realizada pelo usuário

O conjunto de perguntas apresentadas neste questionário pode ser utilizado para a coleta das medidas apresentadas neste trabalho e que envolvem o uso de questionários (apontado como método de coleta na definição da medida), onde o ponto de vista a ser avaliado é o do usuário final. Para as medidas que envolvem métodos de coleta automáticos, que são GCAS, GIAS, GFB, GInB, GPUD, GCUD, GrD e GDM, as perguntas 1, 2, 3 e 4 deste questionário podem servir como fonte de comparação para avaliação dos resultados obtidos. Vale ressaltar que as perguntas 1, 4, 5, 6 e 12 deste questionário foram extraídas de (SANTOS 2014).

**Nome:**

**Data:**

- 1) A aplicação estava disponível quando você necessitou?
  - (1) A aplicação estava sempre disponível para interagir quando era necessário. Você conseguiu executar normalmente a função desejada sem nenhuma dificuldade.
  - (2) Geralmente estava disponível quando era necessário. Você conseguiu executar a aplicação, porém com dificuldades.
  - (3) A disponibilidade não corresponde com sua necessidade de onde e quando utilizar a aplicação. Você não foi capaz de executar funções desejadas.
- 2) Ocorreram erros durante o uso da aplicação (entenda por erro algo inesperado que fez com que a aplicação parasse de executar, e.g., travar ou encerrar sozinha)?
  - (1) Sim (2) Não
- 3) Quantos erros ocorreram durante o uso da aplicação e quanto tempo foi o uso (especificar o tempo em horas)?
- 4) O tempo de adaptação (ocorrência de uma mudança de contexto e a reação do sistema provendo o serviço) foi:

- (1) Minutos (2) Segundos (3) Milissegundos
- 5) Você considera esse tempo:  
(1) Alto (2) Médio (3) Baixo
- 6) A adaptação foi correta (o serviço foi provido conforme o esperado)?  
(1) Sim (2) Não
- 7) O tempo usado para interagir com o subsistema de segurança (por exemplo, autenticação via login) para auxiliar a tarefa principal foi:  
(1) Minutos (2) Segundos (3) Milissegundos
- 8) Você considera esse tempo:  
(1) Alto (2) Médio (3) Baixo
- 9) Você tem capacidade de entender quais dos seus dados ou informações pessoais são utilizados pelo sistema:  
(1) Alto: o usuário se considera ciente de mais de 75% das suas informações que são utilizadas pelo sistema.  
(2) Médio: o usuário se considera ciente de apenas (75% - 50%) das suas informações que são utilizadas pelo sistema.  
(3) Baixo: o usuário se considera ciente de menos de 50% das suas informações que são utilizadas pelo sistema.
- 10) Existem recursos disponíveis para controlar as permissões de uso sobre suas informações pessoais: privacidade do conteúdo, identidade, localização e outros:  
(1) Alto: o usuário tem controle sobre mais de 75% dos seus dados que são utilizados pelo sistema e pode alterar as permissões a qualquer momento.  
(2) Médio: existem recursos disponíveis para o usuário controlar e alterar a permissão de (75% - 50%) dos seus dados pessoais que são utilizados pelo sistema.  
(3) Baixo: O usuário tem controle sobre menos de 50% dos seus dados que são utilizados pelo sistema.
- 11) A aplicação fornece somente informações relevantes?

- (1) Nenhuma informação ou requisição irrelevante é apresentada. Você obteve somente informações e requisições necessárias.
  - (2) A maioria das informações e requisições são relevantes.
  - (3) Informações e requisições irrelevantes foram frequentemente apresentadas.
- 12) A aplicação interage no momento correto?
- (1) A interação ocorreu sempre no momento correto. Quando a interação ocorreu, você concordou que foi no momento correto e você não quis interagir quando ela não ocorreu.
  - (2) A interação ocorreu na maioria das vezes no momento correto. Poucas vezes aconteceu de você não querer a interação quando ela ocorreu, ou de querer interagir quando não ocorreu.
  - (3) Na maioria das vezes, você não concordou com a interação.
- 13) Você notou diferenças no consumo de bateria após a instalação e o uso da aplicação?
- (1) Sim (2) Não
- 14) Você considera essa diferença de consumo
- (1) Alta (2) Média (3) Baixa

## Apêndice C – Questionário utilizado para avaliação realizada pelo desenvolvedor

O conjunto de perguntas apresentadas neste questionário pode ser utilizado para a coleta das medidas definidas neste trabalho que envolvem o uso de questionários (apontado como método de coleta na definição da medida), onde o ponto de vista a ser avaliado é o do desenvolvedor. Para as medidas que envolvem métodos de coleta automáticos, que são GCEA, PPS, GrD, GIdB, GInCE, GIDR, GIDA, GDoc e GEES, as perguntas 6, 7, 8, 9 e 14 deste questionário podem servir como fonte de comparação para avaliação dos resultados obtidos. É importante frisar que algumas das perguntas deste questionário (1, 2, 3, 4, 5 e 6) foram extraídas de (SANTOS 2014).

**Nome:**

**Data:**

1. A aplicação exige alguma intervenção explícita do usuário para receber o serviço?  
(1) Sim (2) Não
2. Quais intervenções o usuário pode ou precisa fazer?
3. Quais sensores do dispositivo são utilizados?
4. Para quais versões do *Android* a aplicação funciona?
5. Para quais dispositivos móveis a aplicação será executada (e.g., *tablet*, *smartphone*)?
6. Qual a quantidade de tecnologias de comunicação a aplicação tem acesso? Descreva quais (e.g., *wifi*, *3G*).
7. A aplicação possui entradas implícitas (aquelas que não precisam ser efetuadas de maneira direta pelo usuário)?  
(1) Sim (2) Não
8. Qual a quantidade de entradas implícitas da aplicação?
9. Descreva as entradas implícitas que a aplicação utiliza.

10. Um usuário leigo da aplicação é capaz de perceber essas entradas implícitas?
- (1) Sim (2) Não
11. Quais entradas implícitas o usuário pode perceber?
12. A entrada implícita pode ser manipulada pelo o usuário?
- (1) Sim (2) Não
13. Quais entradas implícitas o usuário pode manipular?
14. Quantas e quais ações o usuário precisa fazer para a aplicação inicializar?
15. A aplicação possui interfaces para a interação natural como baseada em voz, gestos ou outros?
- (1) Sim (2) Não
16. Existe alguma política de segurança utilizada pela aplicação?
- (0) Nenhuma  
(1) Presença de Política de Concessão Baseada em Permissão  
(2) Presença de Política de Interação
17. Existe documentação associada ao sistema desenvolvido?
- (0) Mais de 75% dos componentes estão documentados.  
(1) Apenas uma parte (75% - 50%) dos componentes estão documentados.  
(2) Menos de 50% dos componentes estão documentados.
18. Qual o esforço relacionado ao tempo necessário para realizar manutenção do sistema
- (0) O tempo para realizar manutenção é superior ao tempo que foi necessário para implantação.  
(1) O tempo para realizar manutenção é equivalente (com variação de até 20%) ao tempo que foi necessário para implantação.  
(2) O tempo para realizar manutenção é inferior ao tempo que foi necessário para implantação.
19. Qual o esforço relacionado ao tempo necessário para realizar manutenção ou evolução do código
- (0) O tempo para realizar manutenção ou evolução é superior ao tempo que foi necessário para codificação.  
(1) O tempo para realizar manutenção ou evolução é equivalente (com variação de até 20%) ao tempo que foi necessário para codificação.

- (2) O tempo para realizar manutenção ou evolução é inferior ao tempo que foi necessário para codificação.
20. Existem técnicas de integridade sobre os dados transmitidos em rede?
- (0) Implementa técnicas de integridade com verificação e correção.
  - (1) Implementa técnicas de integridade, porém não garante verificação e correção.
  - (2) Não implementa técnicas de integridade.
21. Existem técnicas de integridade sobre os dados armazenados?
- (0) Implementa técnicas de integridade com verificação e correção.
  - (1) Implementa técnicas de integridade, porém não garante verificação e correção.
  - (2) Não implementa técnicas de integridade