



Universidade Federal da Bahia
Universidade Salvador
Universidade Estadual de Feira de Santana

TESE DE DOUTORADO

**Um Modelo de Avaliação de Usabilidade de Aplicativos para
Smartphones Baseado na Captura Automática de Interações com
o Usuário**

Artur Henrique Kronbauer

**Programa Multiinstitucional de
Pós-Graduação em Ciência da Computação – PMCC**

Salvador - BA

2013

PMCC-DSc-0010

ARTUR HENRIQUE KRONBAUER

**Um Modelo de Avaliação de Usabilidade de Aplicativos para
Smartphones Baseado na Captura Automática de Interações com
o Usuário**

Tese apresentada ao Programa Multiinstitucional de Pós-Graduação em Ciência da Computação da Universidade Federal da Bahia, Universidade Salvador e Universidade Estadual de Feira de Santana, como requisito parcial à obtenção do título de Doutor em Ciência da Computação.

Orientador: Prof. Dr. Celso Alberto Saibel Santos

Salvador

2013

Kronbauer, Artur Henrique.

Um modelo de avaliação de usabilidade de aplicativos para *smartphones* baseado na captura automática de interações com o usuário / Artur Henrique Kronbauer – Salvador, 2013.

248f. : il.

Inclui anexos

Orientador: Prof. Dr. Celso Alberto Saibel Santos.

Tese (doutorado) – Universidade Federal da Bahia, Instituto de Matemática. Universidade Salvador, Universidade Estadual de Feira de Santana, 2013.

1. Smartphones. 2. Interação homem-máquina - Avaliação. 3. Interação humano-computador. 4. Computação móvel. 5. Programas de computador. I. Santos, Celso Alberto Saibel. II. Universidade Federal da Bahia. Instituto de Matemática. III. Universidade Salvador. IV. Universidade Estadual de Feira de Santana. V. Título.

CDD - 005.3
CDU - 004.42

ARTUR HENRIQUE KRONBAUER

**UM MODELO DE AVALIAÇÃO DE USABILIDADE DE APLICATIVOS PARA
SMARTPHONES BASEADO NA CAPTURA AUTOMÁTICA DE INTERAÇÕES COM O
USUÁRIO**

Esta tese foi julgada adequada à obtenção do título de
Doutor em Ciência da Computação e aprovada em sua
forma final pelo Programa Multiinstitucional de Pós-
Graduação em Ciência da Computação da UFBA-
UNIFACS-UEFS.

Salvador, 29 de Maio de 2013.

Prof. Celso Alberto Saibel Santos (orientador), Docteur en Informatique
Universidade Federal do Espírito Santo – UFES

Prof^ª. Cristiano Maciel, D.SC.
Universidade Federal de Mato Grosso – UFMT

Prof. Jair Leite, D.SC.
Universidade Federal do Rio Grande do Norte – UFRN

Prof. Joberto Sérgio Barbosa Martins, Docteur en Informatique
Universidade Salvador – UNIFACS

Prof^ª. Vaninha Vireira, D.SC.
Universidade Federal da Bahia – UFBA

RESUMO

A onipresença de novas tecnologias associada a diferentes formas de interação, à diversidade de características dos usuários e às interferências contextuais, criam novas barreiras para a avaliação da usabilidade dos aplicativos desenvolvidos para *smartphones*. As propriedades móveis fornecem maneiras de interação até certo ponto inusitadas, altamente dependentes do cenário de interação, o que dificulta a reprodução em laboratório. Um dos desafios nessa área é propor abordagens que permitam realizar testes de usabilidade com a associação de dados quantitativos, contextuais e subjetivos em um mesmo experimento. Sendo assim, este trabalho apresenta um modelo e sua infraestrutura com o objetivo de propor uma nova abordagem que possibilite obter dados estatísticos referentes à usabilidade, contextualizados ao cenário de interação e com interlocuções diretas com os usuários para a obtenção dos seus sentimentos com relação aos aplicativos utilizados. A abordagem proposta permite realizar experimentos por longos períodos de tempo, com a possibilidade da participação de um grande número de usuários e com o potencial para auxiliar a execução de experimentos tanto em cenários reais como em laboratório. Para comprovar a eficiência da abordagem, foram desenvolvidos dois experimentos: o primeiro, com usuários de aplicativos para *smartphones*, com o objetivo de verificar as potencialidades do processo de coleta e análise dos dados; o segundo, com desenvolvedores de softwares e analistas de banco de dados, com a finalidade de avaliar os procedimentos empregados para realizar o mapeamento das tarefas, a instrumentação dos aplicativos com métricas de rastreabilidade e validar os documentos que descrevem os processos envolvidos na configuração do ambiente de armazenamento e exploração dos dados de usabilidade, capturados durante as interações dos usuários finais.

Palavras Chaves: Avaliação de Usabilidade, Dispositivos Móveis, Avaliação Automática de Usabilidade, *Smartphones*.

ABSTRACT

The omnipresence of new technologies associated with different ways of interaction, the diversity of users' characteristics and the contextual interferences, creates new barriers to the usability assessment of applications designed for smartphones. The mobile characteristics provide forms of interaction that are, up to a certain extent, unusual, highly dependent on the interaction scenario, which makes the reproduction in the laboratory difficult. One of the challenges in this area is to propose approaches that will allow usability tests to be carried out in association with quantitative, contextual and subjective data in the same experiment. This thesis presents a model along with its infrastructure aiming to propose a new approach that will permit to acquire statistical data related to the usability, contextualized within the environment of interaction, communicating directly with users in order to obtain their feelings about the applications used. The approach proposed allows experiments to be carried out for long periods of time, enabling the participation of a great number of users, with the potential to help the experiment completion, both in a real environment and in the laboratory. So as to prove the efficiency of this approach, two experiments were developed: the first one, involving users of smartphones applications, with the objective of verifying the potentialities of the data acquisition and analysis process; the second one, with software developers and database analysts, with the purpose of evaluating the procedures used to map the tasks, instrument the applications with traceability metrics and to validate the documents which describe the processes involved in the storage environment setup and the exploration of usability data, captured during the end users interactions.

Keywords: Usability Evaluation, Model Devices, Automatic Usability Evaluation, Smartphones.

Em memória aos meus pais, Arthur
Leondino Kronbauer e Ruth Aranda
Kronbauer.

AGRADECIMENTOS

Primeiramente, agradeço a Deus, pela obstinação, coragem, paciência e dedicação a mim concedidos para realizar este trabalho.

Ao meu orientador Dr. Celso Aberto Saibel Santos, pelos ensinamentos durante esta longa trajetória, dedicando seu tempo e empenho em todas as fases desta pesquisa. Fui aprendendo, com o passar do tempo, lições de vida e de profissionalismo que jamais serão esquecidas.

Ao meu irmão Antonio Kronbauer, por ter participado e acompanhado esta jornada, sempre com conselhos, ensinamentos e palavras de apoio. Além de sermos ligados por laços familiares, sei que posso contar com ele como o meu melhor amigo.

À minha companheira Dra. Shirley Gois Barreto, pela dedicação, cumplicidade, amor e carinho. Mesmo nas horas mais difíceis, tendo muitas vezes que abdicar da minha atenção, teve paciência, cuidado e me auxiliou no que estava ao seu alcance para a realização dos trabalhos que envolveram esta caminhada.

À minha querida cunhada e professora de inglês, Karina Bittencourt, pela paciência e dedicação durante o meu aprendizado de uma nova língua e por todos os textos que me ajudou a escrever. Ela me ajudou a transpor obstáculos que nem eu poderia imaginar.

À professora Dra. Vaninha Vieira pelos trabalhos realizados em conjunto. Suas contribuições foram de suma importância para a lapidação desta tese.

A todos os meus colegas do PMCC que participaram e contribuíram para a realização das atividades do doutorado. Em particular, a Renato Novais e Ernesto Massa que em várias oportunidades, foram meus companheiros de estudo.

Aos meus colegas da Universidade Estadual da Bahia que confiaram na minha potencialidade para realizar este doutorado. Em especial, ao Prof. Fabricio Faro coordenador do curso de Bacharelado em Sistemas de Informação, por ter apoiado e aprovado o processo de afastamento das minhas atividades acadêmicas na referida instituição para a realização desta pós-graduação.

Aos professores do PMCC que deram contribuições para a realização deste trabalho, em especial à Prof. Dra. Christina von Flach G. Chaves pelos ensinamentos na disciplina de Programação Orientada a Aspectos e coautoria no primeiro artigo publicado no início deste trabalho.

A Raoni Kulesza pelas dicas e indicação das bibliografias relacionadas à realização dos experimentos apresentados neste trabalho.

Aos meus amigos Ademar Nogueira do Nascimento, Christianne Orrico Dalforno, Hercules de Souza, Jader Cristiano Magalhães de Albuquerque, Kleber Freire, Monica Massa e Marcus Vinicios da Silva Wagner.

Aos participantes dos experimentos que possibilitaram validar as potencialidades do modelo e da infraestrutura aqui apresentados, além da descrição dos processos necessários para a utilização da infraestrutura.

Aos professores que fizeram parte da banca de avaliação desta tese, Prof. Dr. Cristiano Macial, Prof. Dr. Joberto Sérgio Barbosa Martins, Prof. Dr. Jair Leite, Profa. Dra. Vaninha Vieira e Dr. Celso Aberto Saibel Santos.

A todos os meus familiares, por terem me ensinado que as conquistas só são possíveis com trabalho, honestidade e esforço pessoal.

Não sei quantas almas tenho

Não sei quantas almas tenho.

Cada momento mudei.

Continuamente me estranho.

Nunca me vi nem acabei.

De tanto ser, só tenho alma.

Quem tem alma não tem calma.

Quem vê é só o que vê,

Quem sente não é quem é,

Atento ao que sou e vejo,

Torno-me eles e não eu.

Cada meu sonho ou desejo

do que nasce e não meu.

Sou minha própria paisagem;

Assisto à minha passagem,

Diverso, móbil e só,

Não sei sentir-me onde estou.

Por isso, alheio, vou lendo

Como páginas, meu ser.

O que segue não prevendo,

O que passou a esquecer.

Noto à margem do que li

O que julguei que senti.

Releio e digo : "Fui eu?"

Deus sabe, porque o escreveu.

Fernando Pessoa

SUMÁRIO

1	INTRODUÇÃO	31
1.1	POR QUE O FOCO DESTA PESQUISA SÃO OS <i>SMARTPHONES</i> ?	33
1.2	MOTIVAÇÕES E PROBLEMÁTICAS NA ÁREA DE USABILIDADE.....	35
1.3	OBJETIVOS DESTA PESQUISA	38
1.4	ESTRUTURAÇÃO DO DOCUMENTO.....	40
1.5	RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO.....	40
2	O ESTADO DA ARTE	42
2.1	DEFINIÇÕES DE USABILIDADE	42
2.2	AVALIAÇÕES DE USABILIDADE	48
2.3	TESTES DE USABILIDADE	51
2.3.1	TRABALHOS PUBLICADOS EM 2008	52
2.3.2	TRABALHOS PUBLICADOS EM 2009	53
2.3.3	TRABALHOS PUBLICADOS EM 2010	56
2.3.4	TRABALHOS PUBLICADOS EM 2011	59
2.3.5	TRABALHOS PUBLICADOS EM 2012	63
2.3.6	DISCUSSÃO SOBRE OS TRABALHOS RELATADOS	68
2.4	RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO.....	77
3	METODOLOGIA.....	80
3.1	CONSTRUÇÃO DE UM QUADRO CONCEITUAL.....	81
3.2	PROJETO ARQUITETURAL.....	82
3.3	ANÁLISE E PROJETO DO SISTEMA.....	85
3.3.1	PARADIGMAS E LINGUAGEM DE PROGRAMAÇÃO	86
3.3.2	SISTEMA GERENCIADOR DE BANCO DE DADOS (SGBD)	88
3.3.3	FERRAMENTA PARA AUXILIAR NA ANÁLISE DOS DADOS	89
3.3.4	BASE DE TRANSMISSÃO E ARMAZENAMENTO DOS DADOS	90
3.3.5	TÉCNICAS UTILIZADAS PARA CAPTURA DOS DADOS	91
3.3.5.1	TÉCNICA DE CAPTURA AUTOMÁTICA A PARTIR DA APLICAÇÃO.....	92
3.3.5.2	TÉCNICA <i>EXPERIENCE SAMPLING METHOD</i> (ESM)	93
3.4	CONSTRUÇÃO DO SISTEMA.....	95
3.5	AVALIAÇÃO DO SISTEMA.....	96
3.5.1	EXPERIMENTO REALIZADO EM CAMPO	97

3.5.2	EXPERIMENTO PARA AVALIAR OS PROCESSOS DA INFRAESTRUTURA	98
3.6	RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO.....	99
4	O MODELO E SUA INFRAESTRUTURA.....	101
4.1	VISÃO DE DESENVOLVIMENTO.....	104
4.2	VISÃO LÓGICA	107
4.3	VISÃO DE CASOS DE USO.....	109
4.4	VISÃO DE IMPLANTAÇÃO.....	111
4.5	VISÃO DE PROCESSOS	112
4.6	A INFRAESTRUTURA UXEPROJECT.....	113
4.6.1	AMBIENTE DE DESENVOLVIMENTO.....	114
4.6.1.1	UNIDADE DE MAPEAMENTO.....	114
4.6.1.2	UNIDADE DE RASTREABILIDADE	118
4.6.2	AMBIENTE DE INTERAÇÃO	121
4.6.3	AMBIENTE DE AVALIAÇÃO.....	126
4.7	FORMAS DE CAPTURA DE DADOS	129
4.8	RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO.....	132
5	EXPERIMENTO REALIZADO EM CAMPO	136
5.1	FASE 1 – DETERMINAR OS OBJETIVOS DA AVALIAÇÃO.....	137
5.2	FASE 2 – EXPLORAR PERGUNTAS A SEREM RESPONDIDAS	138
5.3	FASE 3 – ESCOLHER O MÉTODO DE AVALIAÇÃO.....	138
5.4	FASE 4 – IDENTIFICAR E ADMINISTRAR AS QUESTÕES PRÁTICAS.....	139
5.4.1	APLICATIVOS UTILIZADOS NO EXPERIMENTO.....	140
5.4.2	RECRUTAMENTO DOS PARTICIPANTES.....	142
5.4.3	DADOS CONSIDERADOS NO EXPERIMENTO	143
5.5	FASE 5 – DECIDIR COMO LIDAR COM AS QUESTÕES ÉTICAS	145
5.6	FASE 6 – AVALIAR, INTERPRETAR E APRESENTAR OS RESULTADOS.....	146
5.6.1	IDENTIFICAÇÃO DE TAREFAS COM PROBLEMAS DE USABILIDADE	147
5.6.2	ANÁLISE DO DESEMPENHO DOS USUÁRIOS AO LONGO DO TEMPO.....	150
5.6.3	ANÁLISE DA FREQUÊNCIA DE UTILIZAÇÃO DA AJUDA	152
5.6.4	ANÁLISE DO DESEMPENHO DOS USUÁRIOS COMUNS X EXPERIENTE.....	154
5.6.5	ANÁLISE DE INFORMAÇÕES CONTEXTUAIS.....	155
5.6.6	ANÁLISE DA SATISFAÇÃO DOS USUÁRIOS	162
5.6.7	ANÁLISE DOS VALORES DE REFERÊNCIA X DADOS REAIS	166
5.7	RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO.....	169

6	AVALIAÇÃO DA ABORDAGEM POR DESENVOLVEDORES DE SOFTWARE.....	172
6.1	FASE 1 – DETERMINAR OS OBJETIVOS DA AVALIAÇÃO	173
6.2	FASE 2 – EXPLORAR PERGUNTAS A SEREM RESPONDIDAS	174
6.3	FASE 3 – ESCOLHER O MÉTODO DE AVALIAÇÃO	175
6.4	FASE 4 – IDENTIFICAR E ADMINISTRAR AS QUESTÕES PRÁTICAS	176
6.5	FASE 5 – DECIDIR COMO LIDAR COM AS QUESTÕES ÉTICAS	177
6.6	FASE 6 – AVALIAÇÃO DOS DADOS	178
6.6.1	ANÁLISE DA FERRAMENTA MAG.....	179
6.6.2	ANÁLISE DA FERRAMENTA ATD.....	180
6.6.3	ANÁLISE GERAL DO PROCESSO DE MAPEAMENTO DAS TAREFAS	182
6.6.4	CRIAÇÃO E INSERÇÃO DAS MÉTRICAS DE RASTREABILIDADE	183
6.6.5	ANÁLISE GERAL DO PROCESSO DE INSTRUMENTAÇÃO DOS APLICATIVOS...	185
6.6.6	RESULTADOS DA ANÁLISE REFERENTE AO AMBIENTE DE AVALIAÇÃO	187
6.7	RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO	189
7	CONCLUSÕES.....	191
7.1	CONTRIBUIÇÕES	194
7.2	LIMITAÇÕES	196
7.3	PERSPECTIVAS FUTURAS.....	197
	REFERÊNCIAS.....	199

LISTA DE FIGURAS

Figura 1 – <i>Framework</i> sugerido por Coursaris e Kim (2011).	46
Figura 2 – Distribuição percentual das técnicas de avaliação utilizadas entre 2008 e 2012	70
Figura 3 – Distribuição percentual de participantes nos experimentos	70
Figura 4 – Quantidade absoluta de vezes que cada atributo de usabilidade foi avaliado	72
Figura 5 – Percentuais relacionados ao local onde os experimentos foram realizados	75
Figura 6 – Análise quantitativa das plataformas utilizadas nos experimentos	77
Figura 7 – Etapas do desenvolvimento do projeto de pesquisa (Hevner e Chatterjee, 2010).....	80
Figura 8 – Visões no Modelo 4+1 (Kruchten, 1995).	83
Figura 9 – Exemplo de pergunta proposta aos participantes.	94
Figura 10 – Principais Componentes do Modelo.....	103
Figura 11 – Diagrama de Componentes da Unidade de Mapeamento	104
Figura 12 – Diagrama de Componentes da Unidade de Rastreabilidade.	105
Figura 13 – Diagrama de Componentes da Unidade de Avaliação.....	106
Figura 14 – Diagrama de Sequência da Unidade de Mapeamento	107
Figura 15 – Diagrama de Sequência da Unidade de Rastreabilidade.....	108
Figura 16 – Diagrama de Sequência da Unidade de Avaliação.....	109
Figura 17 – Caso de Uso da Unidade de Mapeamento	110
Figura 18 – Caso de Uso da Unidade de Rastreabilidade.	110
Figura 19 – Caso de Uso da Unidade de Avaliação.....	111
Figura 20 – Diagrama de Implantação do Modelo	112
Figura 21 – Principais processos presentes no Modelo Arquitetural	113
Figura 22 – A Ferramenta MAG.....	115
Figura 23 – Fluxo de ações para a inserção dos Aspectos de mapeamento	115
Figura 24 – <i>PointCut</i> e <i>Advice</i> para identificar a interação do usuário	116
Figura 25 – Janela principal da Ferramenta ATD.....	116
Figura 26 – Janela para definição dos valores de referência	117
Figura 27 – Geração de métricas de rastreabilidade.	118
Figura 28 – <i>UXE Metrics Generation</i> – Métricas de Usabilidade e Contexto.....	119
Figura 29 – <i>UXE Metrics Generation</i> – Métricas Subjetivas	120
Figura 30 – <i>UXE Metrics Generation</i> – Controle	120
Figura 31 – Geração da aplicação com as métricas de Rastreabilidade	121

Figura 32 – Execuções previstas no Ambiente de Interação.	122
Figura 33 – Estrutura do XML gerado durante o mapeamento das tarefas.....	123
Figura 34 – Estrutura do XML contendo as métricas de usabilidade e contexto.....	124
Figura 35 – Estrutura do XML contendo as respostas subjetivas.....	124
Figura 36 – Cenários do Ambiente de Interação.	125
Figura 37 – Fluxo de ações na Unidade de Avaliação.....	127
Figura 38 – Modelagem do Banco de Dados.	127
Figura 39 – Modelagem do <i>Data Warehouse</i>	128
Figura 40 – Variáveis contextuais consideradas na Infraestrutura UXEProject.....	131
Figura 41 – Etapas para executar o processo Gerar Aspectos de Mapeamento.....	132
Figura 42 – Tarefas executadas no processo Inserir Métricas de Rastreabilidade.....	133
Figura 43 – Tarefas executadas no processo Preparar Ambiente de Avaliação.....	134
Figura 44 – Interface do Aplicativo Mileage e tarefas instrumentadas para sua avaliação.....	140
Figura 45 – Interface do Aplicativo ^3 (Cubed) e tarefas instrumentadas para sua avaliação.....	141
Figura 46 – Interface do Aplicativo Shuffle e tarefas instrumentadas para sua avaliação.....	142
Figura 47 – Percentual de tarefas completadas com erro.....	150
Figura 48 – Tempo médio para execução das tarefas propostas.....	151
Figura 49 – Tempo médio para a execução das tarefas considerando 8 participantes.....	152
Figura 50 – Percentual de utilização da opção de Ajuda.....	153
Figura 51 – Influência da luminosidade na taxa de erro.....	157
Figura 52 - Influência da luminosidade na velocidade (em seg.).....	157
Figura 53 – Taxa de erro em função da velocidade de deslocamento.....	159
Figura 54 – Velocidade de Execução das tarefas em função da resolução da tela (em seg.).....	159
Figura 55 – Velocidade de execução das tarefas em função do tamanho da tela (em seg.).....	160
Figura 56 – Relação do tamanho e resolução da tela com o percentual de erros.....	161
Figura 57 – Relação da taxa de erro em função do Poder Aquisitivo X Resolução da Tela.....	161
Figura 58 – Respostas associadas ao Aplicativo Cubed.....	163
Figura 59 – Respostas associadas ao Aplicativo Mileage.....	163
Figura 60 – Respostas associadas ao Aplicativo Shuffle.....	164
Figura 61 – Análise da insatisfação dos participantes com os aplicativos.....	165
Figura 62 – Janela principal da Aplicação Places ME.....	177
Figura 63 – Avaliação da descrição dos processos para utilizar a Ferramenta MAG.....	179
Figura 64 – Avaliação da Ferramenta MAG.....	179
Figura 65 – Abordagem proposta com a Ferramenta MAG x Abordagem Tradicional.....	180

Figura 66 – Avaliação da descrição dos processos para a utilização da Ferramenta ATD	181
Figura 67 – Avaliação da dificuldade para utilizar a Ferramenta ATD	181
Figura 68 – Abordagem proposta com a Ferramenta ATD X Abordagem Tradicional	182
Figura 69 – Avaliação da descrição dos processos do mapeamento de tarefas	182
Figura 70 – Comparação das abordagens para identificar os <i>join points</i>	183
Figura 71 – Análise da descrição dos processos referentes a Ferramenta UXE Metrics Generetor	184
Figura 72 – Avaliação do nível de dificuldade para utilizar a Ferramenta UXE Metrics Generator ...	184
Figura 73 – Comparação das metodologias para criar as métricas de rastreabilidade	185
Figura 74 – Avaliação geral da descrição dos processos para a instrumentação dos aplicativos	185
Figura 75 – Avaliação geral do nível de dificuldade para instrumentar aplicativos	186
Figura 76 – Avaliação geral das duas metodologias para instrumentar aplicativos.....	186
Figura 77 – Nível de satisfação com a descrição dos processos das Ferramentas Data Load e ETL ..	187
Figura 78 – Nível de dificuldade para instalar as Ferramentas Data Load e ETL Maker	187
Figura 79 – Nível de satisfação com a descrição dos processos relacionados à Ferramenta OLAP....	188
Figura 80 – Nível de dificuldade para a preparação da Ferramenta OLAP.....	188
Figura 81 – Nível de dificuldade para a realização das análises dos dados.....	189
Figura 82 – Evolução histórica dos <i>smartphones</i>	212
Figura 83 – Exemplo da combinação entre Classes e Aspectos.	215
Figura 84 – Representação do <i>Weaver</i>	215
Figura 85 – Representação do código de baixo nível com as Classes e Aspectos.	216

LISTA DE TABELAS

Tabela 1 – Definição dos atributos que caracterizam usabilidade no escopo deste trabalho	47
Tabela 2 – Diferenças entre os métodos de avaliação.....	49
Tabela 3 – Relação das técnicas de avaliação	50
Tabela 4 – Investigação da estrutura de navegação Web móvel (Kim et al., 2010).	57
Tabela 5 – Número de problemas de usabilidade encontrados (Fetaji et al., 2011).....	61
Tabela 6 – Resumo dos dados coletados no experimento (Sparkes et al., 2012).....	64
Tabela 7 – Avaliação da usabilidade do aplicativo iStepLog (Kirwan et al., 2012).....	66
Tabela 8 – Resumo das características dos experimentos	68
Tabela 9 – Dimensão de Usabilidade.....	71
Tabela 10 – Variáveis Contextuais	74
Tabela 11 – Relação das etapas da metodologia DSR com a presente pesquisa	81
Tabela 12 – Atividades do <i>framework</i> DECIDE (Sharp et al., 2007).	98
Tabela 13 – Resumo da relação das etapas DSR e a execução desta pesquisa.....	100
Tabela 14 – Tecnologias escolhidas para a construção da Infraestrutura.....	100
Tabela 15 – Métricas de Usabilidade	130
Tabela 16 – Sensores implementados na infraestrutura UXEProject	131
Tabela 17 – Descrição das tarefas para executar o processo Gerar Aspectos de Mapeamento	132
Tabela 18 – Descrição das tarefas para executar o processo Inserir Métricas de Rastreabilidade	133
Tabela 19 – Descrição das tarefas executadas no processo Preparar Ambiente de Avaliação	134
Tabela 20 – Distribuição dos usuários de acordo com seus perfis.....	143
Tabela 21 – Características dos <i>Smartphones</i>	144
Tabela 22 – Escala de valores para dados do ambiente	144
Tabela 23 – Percentual de tarefas completadas com erro no Mileage	147
Tabela 24 – Percentual de tarefas completadas com erro no Cubed.....	148
Tabela 25 – Percentual de tarefas completadas com erro no Shuffle.....	148
Tabela 26 – Avaliação das tarefas com maior índice de erros durante 12 meses	149
Tabela 27 – Taxa de erro dos 8 participantes considerando os 12 meses	151
Tabela 28 – Desempenho dos usuários Comuns x Experiente (primeiros 6 meses).....	154
Tabela 29 – Desempenho dos usuários Comuns x Experiente (últimos 6 meses).....	155
Tabela 30 – Relação de tarefas com alto índice de erro.....	158
Tabela 31 – Perguntas submetidas aos usuários	162

Tabela 32 – Valores de Referência x Dados Reais (Cubed)	167
Tabela 33 – Valores de Referência x Dados Reais (Mileage).....	167
Tabela 34 – Valores de Referência X Dados Reais (Shuffle)	168
Tabela 35 – Pontos positivos e negativos das abordagens para identificar os <i>join points</i>	183
Tabela 36 – Configurações dos <i>smartphones</i> em três gerações diferentes	213

GLOSSÁRIO DE TERMOS

Equipe de Avaliação	Constituída de pessoas que executam algum processo relacionado à infraestrutura UXEProject.
Móvel	É a habilidade de poder se movimentar. Na definição aqui empregada, móvel é uma característica de algo que pode ser utilizado em movimento. Portanto, a característica móvel pode ser empregada em um dispositivo computacional se tal dispositivo pode ser facilmente transportado de um local para outro enquanto o usuário interage com ele. Ou seja, a mobilidade é um atributo tanto do dispositivo quanto do usuário.
Usuários	São potenciais utilizadores da infraestrutura UXEProject como, por exemplo, os desenvolvedores de softwares, analistas de banco de dados e analistas de IHC.
Usuário Final	Corresponde ao usuário que interage com aplicativos para <i>smartphones</i> .
Usuário Experiente	São os participantes do experimento realizado em campo com mais tempo de utilização dos aplicativos utilizados na avaliação, ou seja, tiveram contato com os aplicativos dois meses antes do início do experimento.
Usuários Normais	São os participantes do experimento realizado em campo que obtiveram contato com os aplicativos utilizados na avaliação após o início do experimento.

LISTA DE ABREVIATURAS E SIGLAS

AAC	<i>Advanced Audio Coding</i>
Amazon EC2	<i>Amazon Elastic Compute Cloud</i>
API	<i>Application Programming Interface</i>
APK	<i>Android Package File</i>
ATD	<i>Automatic Task Description</i>
BD	<i>Banco de Dados</i>
DECIDE	<i>Determine Explore Choose Identify Decide Evaluate</i>
DSR	<i>Design Science Research</i>
DW	<i>Data Warehouse</i>
ECG	<i>Electrocardiography</i>
ELR	<i>Event Loop Recorder</i>
EPF	<i>Eclipse Process Framework</i>
ER	<i>Entidade e Relacionamento</i>
ESM	<i>Experience Sampling Method</i>
ETL	<i>Extract Transform Load</i>
FTP	<i>File Transfer Protocol</i>
GPS	<i>Global Positioning System</i>
HMD	<i>Head Mounted Display</i>
HTC	<i>High Tech Computer Corporation</i>
IBM	<i>International Business Machines</i>
IDE	<i>Integrated Development Environment</i>
IEEE	<i>Instituto de Engenheiros Eletricistas e Eletrônicos</i>
IHC	<i>Interação Humano-Computador</i>
IMEI	<i>International Mobile Equipment Identity</i>
IVE	<i>Integrated Interactive Information Environment</i>
IMAP4	<i>Internet Message Access Protocol</i>
JAXB	<i>Java Architecture for XML Binding</i>
JVM	<i>Java Virtual Machine</i>
MAG	<i>Mapping Aspect Generator</i>
MIME1	<i>Multipurpose Internet Mail Extensions, part one</i>
MDX	<i>MultiDimensional eXpressions</i>

MMS	<i>Multimedia Messaging Service</i>
MPEG4	<i>Moving Picture Experts Group – Version 4</i>
MP3	<i>MPEG-1/2 Audio Layer 3</i>
AO	Orientado a Aspectos
OLAP	<i>Online Analytical Processing</i>
OLTP	<i>Online Transaction Processing</i>
OMG	<i>Object Management Group</i>
OO	Orientado a Objetos
PCs	<i>Personal Computers</i>
PDA	<i>Personal Digital Assistant</i>
POA	Programação Orientada a Aspectos
POO	Programação Orientada a Objetos
POP3	<i>Post Office Protocol (version 3)</i>
QoS	Qualidade de Serviço
SEM	<i>Structural Equation Model</i>
SGBD	Sistema Gerenciador de Banco de Dados
SMS	<i>Short Message Service</i>
SMTP	<i>Simple Mail Transfer Protocol</i>
SO	<i>Sistema Operacional</i>
SPEM	<i>Software Process Engineering Metamodel</i>
SQL	<i>Structured Query Language</i>
STEMI	<i>S-T Segment Elevation Myocardial Infarctions</i>
TAM	<i>Technology Acceptance Model</i>
UML	<i>Unified Modeling Language</i>
USB	<i>Universal Serial Bus</i>
UX	<i>User eXperience</i>
VGA	<i>Video Graphics Array</i>
WEKA	<i>Waikato Environment for Knowledge Analysis</i>
WI-FI	<i>Wireless Fidelity</i>
WLAN	<i>Wireless Local Area Network</i>
WPAN	<i>Wireless Personal Area Network</i>
XML	<i>eXtensible Markup Language</i>
3G	Terceira geração de padrões e tecnologias de telefonia móvel
4G	Quarta geração de padrões e tecnologias de telefonia móvel

Este capítulo apresenta alguns conceitos introdutórios referentes ao estudo proposto neste trabalho. O seu principal foco é descrever as motivações e problemáticas relacionadas ao tema, bem como os objetivos gerais e específicos que norteiam esta tese.

1 INTRODUÇÃO

Ao longo da última década, diversas tecnologias que revolucionaram a forma de comunicação, entretenimento e a execução de atividades foram apresentados à sociedade. O processo de convergência digital gerou uma gama de novos dispositivos, com destaque especial para os *tablets* e *smartphones*, capazes de executar aplicações inovadoras, com diferentes formas de interação e com razoável poder de processamento. Esse progresso é consequência da evolução do hardware e do software que estão embarcados nestes dispositivos.

A redução do tamanho dos equipamentos, o aumento das funcionalidades e suas características portáteis tornam a sua adoção cada vez mais frequente na execução de nossas tarefas diárias. Muitas pessoas estão migrando dos *desktops* e *laptops* para dispositivos menores em função da facilidade de serem transportados e da abrangência das redes que possibilitam a comunicação de forma ubíqua. Uma prova que evidencia este fato é que, em 2011, os *smartphones* ultrapassaram a venda de PCs (DO et al., 2011), fato que destaca a importância e aceitação destes dispositivos no mercado.

Segundo Queiroz e Ferreira (2009), os dispositivos estáticos (por exemplo, *desktops*) diferenciam-se substancialmente dos dispositivos móveis no que diz respeito à forma, dimensões, mobilidade, conectividade, mecanismos de entrada, tela de visualização e capacidades de processamento e armazenamento. Assim, é prudente rever as abordagens de avaliação quando o objetivo passa a ser avaliar *smartphones*, já que se trata de um dispositivo móvel.

A possibilidade de utilizar as aplicações nos mais variados ambientes tornaram o contexto de uso dos *smartphones* uma importante área de investigação e, conseqüentemente, a verificação da influência das variáveis contextuais no desempenho do usuário. Segundo Korhonen et al. (2010), tanto aspectos

emocionais e físicos, quanto a localização do usuário e a quantidade de pessoas no ambiente, podem causar impacto nas interações. Barnard et al. (2007) também enfatizam que a usabilidade de dispositivos móveis é influenciada pelo contexto social e cultural nos quais os produtos são utilizados.

A grande propagação do mercado dos *smartphones*, inclusive superior à difusão dos computadores, significa que cada vez mais indivíduos de diferentes localidades e níveis culturais serão potenciais consumidores destes dispositivos. Este fato, aliado à adoção destes aparelhos como ferramenta de trabalho em diversos setores, tais como vendas, turismo, educação e construção civil, tem impulsionado o interesse de grupos de pesquisa, fabricantes e prestadores de serviço em aspectos relativos à usabilidade destes dispositivos (Queiroz e Ferreira, 2009).

As dificuldades de interação dos usuários observadas em situações reais de uso referem-se à rigidez das interfaces disponibilizadas nos aplicativos embarcados nos *smartphones*. Isso acarreta prejuízos em termos de usabilidade e os torna limitados para o seu propósito. Além disso, estes dispositivos estão provocando mudanças importantes com relação às interfaces disponíveis para a interação dos usuários. A primeira mudança está relacionada com a possibilidade da adoção de interfaces multimodais com interações por voz e/ou toques na tela. Interfaces com estas características proporcionam interações mais naturais e flexíveis à execução de tarefas, uma vez que possibilitam aos usuários com diferentes níveis de habilidades escolherem o modo de interação mais adequado às suas necessidades. A segunda mudança refere-se às aplicações executadas em arquiteturas voltadas à mobilidade, o que impõe novas restrições como o consumo de energia, conectividade, adequação ao contexto e a segurança das informações (Hansen, 2012).

Tomando como base este cenário, os *smartphones* e suas aplicações são atraentes em termos de oportunidades e nichos de pesquisa, tanto em nível de projeto de novos aplicativos quanto no que se refere à avaliação da usabilidade e experiências proporcionadas aos usuários (Queiroz e Ferreira, 2009).

O crescimento da importância do papel da usabilidade no desenvolvimento de produtos de software pode ser observado pelo aumento do percentual do orçamento de um projeto destinado a esta área. Segundo Queiroz (2001), os gastos com usabilidade eram em torno de 3% em 1971 e passaram para 6% em 1993. Nielsen (2008) realizou o mesmo estudo em 2003 e 2008, constatando que os gastos com usabilidade em projetos de software se mantêm em torno de 10% nos dias atuais.

Segundo a Associação de Profissionais de Usabilidade¹, os investimentos relacionados a essa área têm seus custos justificados e seu retorno garantido através dos seguintes aspectos:

- Aumento da produtividade.
- Redução dos custos de treinamento e suporte.
- Aumento nas vendas e receitas.

¹ Disponível em: http://www.upassoc.org/usability_resources/index.html

- Minimização de alterações no projeto de software em estágios avançados do processo de desenvolvimento.

- Redução do número de erros cometidos pelo usuário.
- Redução dos custos de manutenção.
- Maior satisfação dos clientes.

Diante dos desafios impostos pelo crescente uso de dispositivos móveis e pela ascensão da importância do aspecto de usabilidade no desenvolvimento de produtos, mostra-se pertinente a concepção de um modelo arquitetural e a implementação de uma infraestrutura adequada à avaliação da usabilidade de aplicativos para *smartphones*.

Tomando como base a relevância de estudos nesta área, o foco desta tese está voltado para a avaliação da usabilidade de aplicativos móveis, com principal atenção para aplicativos que são executados em *smartphones*. O cenário de interesse abrange abordagens e técnicas para avaliações da usabilidade de aplicativos em plataformas móveis, levando em consideração as tarefas a serem executadas, informações de contexto, características dos usuários, recursos dos dispositivos, novos paradigmas de interação e a satisfação dos usuários ao interagirem com estes aplicativos.

Nas subseções seguintes, serão abordadas as motivações, as problemáticas relacionadas à área e os objetivos gerais e específicos que orientam este trabalho. A evolução histórica dos *smartphones* pode ser encontrada no Anexo I.

1.1 POR QUE O FOCO DESTA PESQUISA SÃO OS *SMARTPHONES*?

Optou-se por investigar a usabilidade de aplicativos para *smartphones* por quatro razões que são apresentadas a seguir:

1. A utilização generalizada de *smartphones*

Várias pesquisas (Cheng et al., 2007), (Kang et al., 2011), (Palit et al., 2011), (Xu et al., 2011) e (Hansen, 2012) na área de dispositivos móveis indicam a crescente popularidade e importância dos *smartphones*. Além disso, os *smartphones* estão se tornando uma importante ferramenta no auxílio das tarefas diárias dos seus consumidores. Não apenas os relatórios apresentados por importantes agências como a Gartner (2012) e ITU (2011) demonstram um aumento significativo da popularidade dos *smartphones*, como também relevantes pesquisas demonstram esta tendência.

A popularidade dos *smartphones* impõe que os sistemas utilizados nestes aparelhos sejam amplamente aceitos e empregados por um considerável número de pessoas. Além disso, eles estão se tornando cada vez mais importantes para as empresas que oferecem suporte para esta tecnologia, principalmente devido ao número de usuários e serviços atrelados a estes dispositivos, tornando seus negócios mais ágeis e eficazes.

2. Os *smartphones* utilizam inovadores recursos de hardware e software

Quando a Apple lançou o *iPhone*, houve um grande avanço tanto em termos de hardware com de software². O *iPhone* redefiniu as expectativas dos usuários com relação aos benefícios dos dispositivos móveis (Charland, 2011). Antes do *iPhone* já existiam *smartphones* disponíveis no mercado, entretanto, eles eram extremamente limitados quando comparados aos modernos dispositivos criados para utilizar as plataformas Android e iPhone OS.

As inovações destes dispositivos como a forma de interação através de toques na tela e a disponibilidade de diferentes tipos de sensores, criaram novas oportunidades e interesses para os desenvolvedores de aplicativos e pesquisadores. Atualmente, existe a possibilidade da utilização dos sensores para auxiliar na coleta de diferentes informações que podem melhorar a qualidade das avaliações de usabilidade.

Adicionalmente, com os novos recursos de hardware, aumento da velocidade de processamento e melhorias nas ferramentas de desenvolvimento, foi possível a criação de inovadoras aplicações. Cheng et al. (2007) previram que estes aparelhos se tornariam cada vez mais populares porque oferecem a integração de inúmeras funcionalidades em um único dispositivo, associando os tradicionais celulares aos computadores portáteis.

3. Os *smartphones* podem utilizar diferentes tecnologias de redes sem fio

Os *smartphones* têm a capacidade de se conectar a diferentes tecnologias de redes, tais como a EDGE, 3G e 4G atualmente disponíveis para vários aparelhos. Além disso, os dispositivos normalmente têm a capacidade de utilizar recursos de rede locais, tais como *Wireless Local Area Network* (WLAN) e *Wireless Personal Area Network* (WPAN). Com a possibilidade de se conectar a Internet e a outros dispositivos, novas oportunidades são criadas, não limitando seus recursos apenas às informações disponíveis localmente nos aparelhos. Isto significa que os *smartphones* combinam a mobilidade com o poder computacional encontrada em outros aparelhos, tais como, *laptops e tablets*.

4. Os *smartphones* proporcionam novos desafios

Inúmeros desafios são potencializados devido a fatores contextuais que podem interferir na usabilidade, diferentes formas de interação, uma vasta gama de novos aplicativos desenvolvidos e

² Disponível em: <http://www.apple.com/pr/library/2007/01/09Apple-Reinvents-the-Phone-with-iPhone.html>

disponibilizados por desenvolvedores com pouca experiência nesta plataforma, além de limitações na velocidade, estabilidade e confiabilidade das redes de acesso aos dados (Grønli, 2012).

Outro ponto importante, especificamente para a área de Interação Humano-Computador (IHC), está relacionado com a utilização de técnicas de Engenharia de Software que propiciem a coleta de dados das interações dos usuários com os aplicativos de forma automática, contemplando informações dos cenários de utilização em tempo real, sem interferir no desempenho da aplicação.

Como as avaliações de usabilidade para *smartphones* requerem a captura de um grande volume de dados, principalmente para relacionar as interações como o cenário de utilização dos aplicativos, é necessário prever um local para armazenar estes dados e possibilitar as análises de forma rápida, fácil e de diferentes ângulos.

1.2 MOTIVAÇÕES E PROBLEMÁTICAS NA ÁREA DE USABILIDADE

A investigação da forma como se dá a interação do usuário tem se tornado mais importante à medida que novos aplicativos, com funcionalidades e características cada vez mais complexas, são incorporados as mais diversas atividades humanas. Deste modo, a inserção dos conceitos da área de IHC como princípio de projeto vem se consolidando como um fator diretamente relacionado ao sucesso dos aplicativos (Queiroz e Ferreira, 2009).

Segundo Hansen (2012), a partir do momento que um novo dispositivo tecnológico é criado, as suas formas de interação disponibilizadas passam a ser um importante alvo de estudo. Deste modo, fatores como a miniaturização e a mobilidade dos dispositivos computacionais tornam-se passíveis de comprometimento do modo de interação e da geração de novas formas de utilização. Esses fatores impõem novos desafios às atuais tendências dos processos interativos homem-máquina.

Outro fator relevante é decorrente do processo de globalização que se instaurou no mundo atual e que propicia a disseminação de inúmeros produtos em diferentes mercados internacionais, sem a devida apreciação de aspectos socioculturais e antropométricos de suas populações (Queiroz e Ferreira, 2009).

A avaliação da usabilidade de aplicações para *smartphones* é um tema de constantes discussões na área de IHC devido à alta demanda por dispositivos dessa natureza e sua inerente contextualização de uso. A principal preocupação dos pesquisadores e profissionais da área é referente à escolha dos métodos e medidas mais adequados para extrair dados ligados à usabilidade durante uma avaliação. Um dos pontos de discussão iniciais se refere à técnica utilizada na avaliação que pode ser baseada em coleta de dados em

laboratório (*in vitro*) e ou em campo (*in vivo*) (Kawalek, 2008) (Jensen e Larsen, 2007). Outro ponto que reforça as discussões envolve o tipo de dado a ser coletado, quantitativo ou subjetivo (Kjeldskov e Stage, 2004) (Nielsen et al., 2006).

No caso de avaliações de aplicações para *smartphones*, a contextualização do experimento é de fundamental importância e, desta forma, dados contextuais também devem ser associados aos experimentos (Barnard et al., 2007). As necessidades e características do usuário, a interferência do meio externo, as particularidades dos aplicativos e as limitações físicas são fatores que influenciam a interação e devem ser considerados na avaliação de usabilidade (Avouris et al., 2008). Existe ainda a possibilidade de um único dispositivo ser usado em mais de um contexto, em diferentes situações, para diferentes objetivos e tarefas, servindo a um único usuário ou a um grupo deles (Avouris et al., 2008).

Apesar dos esforços para reproduzir em laboratório o cenário natural de uso dos aplicativos, alguns trabalhos defendem que a melhor forma de avaliar a usabilidade de dispositivos móveis é diretamente no seu ambiente natural de uso (Jensen e Larsen, 2007) (Waterson et al., 2002). Essa polêmica é recorrente na área de IHC e teve suas primeiras repercussões nos anos 80, ganhando força após a publicação de Whiteside em 1988 (apud Maguri, 2001). O trabalho mostrou que, embora alguns produtos apresentassem bons resultados em experimentos realizados dentro do laboratório, o mesmo não se repetia quando o produto era usado no mundo real. Esta pesquisa trouxe novamente à tona a importância do contexto de uso na avaliação (Maguri, 2001).

O contexto de uso é definido a partir dos usuários, das tarefas que eles têm que executar, dos ambientes técnicos, organizacionais e físicos e até mesmo do dia e da hora em que o produto está sendo utilizado. Desta forma, para especificar ou medir a usabilidade é necessário definir os objetivos de uso, o contexto e as medidas de usabilidade (Wigelius e Väättäjä, 2009).

Segundo Mallick (2003), as experiências mostram que os seres humanos sempre interagem com sistemas de formas inusitadas. Uma vez que a previsão de tais formas de interação não é uma tarefa trivial, a realização de testes constitui uma atividade crítica para a garantia da usabilidade do aplicativo associado ao contexto de uso considerado. Desta forma, a inclusão de usuários reais nos testes é imprescindível para o delineamento do perfil do usuário e a consequente adequação dos produtos a ele destinados (Queiroz e Ferreira, 2009).

Apesar da defesa de muitos autores que as investigações da usabilidade de dispositivos móveis devem ser realizadas em campo, Coursaris e Kim (2011) realizaram um levantamento sistemático, no período de 2000 a 2010, que permitiu identificar que 47% dos trabalhos de avaliação são realizados em laboratório, 21% em campo, 10% utilizam ambos os cenários e 22% são realizados sem a participação de usuários.

Observa-se através dos relatos existentes na literatura que, tanto as métricas quantitativas, quanto as subjetivas, são amplamente aplicadas para avaliações de aplicativos móveis. Entretanto, as métricas apropriadas para cada experimento, em geral, estão associadas à abordagem usada. Diante desse cenário, várias métricas e abordagens foram propostas para a avaliação de aplicativos móveis (Kawalek et al., 2008), (Heo et al., 2009), (Tossell et al., 2012), (Hansen, 2012), embora ainda não se tenha chegado a um consenso sobre quais as melhores práticas para se alcançar bons resultados em avaliações dessa natureza. É necessário escolher um método apropriado, levando em consideração o tempo disponível para a execução dos procedimentos metodológicos, os dados a serem coletados, o número e a natureza dos problemas identificados e a relação custo/benefício (Betiol e Cybis, 2005).

Kawalek et al. (2008) sugerem a utilização de métodos de avaliação que contemplem diferentes ângulos de observação nos experimentos realizados nesta área, tais como dados quantitativos (métricas de usabilidade), a avaliação subjetiva (sentimento dos usuários) e dados contextuais (por exemplo, condições ambientais e características dos dispositivos). O principal problema atualmente é que se observa uma carência na literatura por abordagens que suportem esses três requisitos combinados em um único experimento, geralmente, apenas um ou dois deles são relacionados.

Segundo Cairns et al. (2008), um dos desafios a ser transposto é a criação de abordagens de avaliação que possam capturar e analisar dados quantitativos, contextuais e subjetivos referentes à interação dos usuários com novas aplicações. A categorização desses dados possibilita a inferência de informações úteis para o desenvolvimento de aplicações que facilitem a execução de tarefas pelos usuários, além de auxiliarem na elaboração de novas versões de um determinado aplicativo.

Várias limitações e falhas podem ser percebidas na maioria dos testes de usabilidade conduzidos em dispositivos móveis. A maioria destas falhas é relacionada a fatores orçamentários, tecnológicos ou mesmo de recursos humanos, enquanto outras estão relacionadas à falta de adequação das abordagens empregadas (Queiroz e Ferreira, 2009).

Outro ponto a ser observado é que inúmeros estudos não consideram o caráter móvel de tais dispositivos, aplicando métodos de avaliação tradicionais. Coursaris e Kim (2011), no seu levantamento sistemático, identificaram as principais abordagens utilizadas na avaliação de dispositivos móveis. Um fato que chama atenção nos resultados observados é que 47% dos estudos avaliam tarefas individuais e descontextualizadas, 46% são voltados para a tecnologia empregada e apenas 14% levam em consideração variáveis contextuais e as características dos utilizadores. Portanto, existe falta de pesquisas empíricas sobre a relevância das características dos usuários e do impacto do meio ambiente na usabilidade móvel.

1.3 OBJETIVOS DESTA PESQUISA

Diante dos problemas e motivações abordados na subseção anterior, o principal objetivo desta tese é propor uma abordagem que possibilite realizar testes de usabilidade contextualizados ao ambiente de uso dos dispositivos, com a capacidade de coletar dados automaticamente, abrangendo as diferentes características dos usuários, com a possibilidade de auxiliar experimentos realizados em campo ou em laboratório, controlados ou não, independentes do período de duração e do número de usuários. Para isso, é proposto um modelo arquitetural, sua infraestrutura e a descrição dos processos necessários para a instrumentação de aplicativos para *smartphones*, possibilitando a obtenção do perfil do usuário e a coleta de dados quantitativos, contextuais e subjetivos, para posterior análise.

Para extrair os dados referentes às interações, contexto de utilização e sentimento dos usuários, é proposta, nesta tese, uma nova abordagem que mescla duas técnicas para a captura de informações:

- A primeira, conhecida como Captura Automática a partir da Aplicação (*Logging*) (Ivory e Hearst, 2001), baseia-se na coleta de dados referentes à interação do usuário com uma aplicação, possibilitando análises estatísticas a respeito da usabilidade. A técnica também possibilita a utilização de sensores disponíveis nos *smartphones* para a coleta de dados contextuais, tais como, a intensidade da luminosidade e posição do aparelho (Froehlich et al., 2007). Estes dados são coletados de forma “transparente” e “automática”.

- A segunda, conhecida como *Experience Sampling Method* (ESM) (Meschtscherjakov et al., 2009), baseia-se na coleta de dados sobre a percepção do usuário com relação a um determinado produto por meio de interlocuções diretas aos usuários.

O modelo proposto estabelece pontos de avaliações alicerçados no confronto entre a ótica da equipe de IHC, o comportamento efetivo do usuário em situações reais de uso da aplicação, contextos de interação e comparação do comportamento dos usuários finais com usuários especialistas.

O processo de avaliação, associado ao modelo, permite a execução de ações que possam trazer melhorias para a aplicação fundamentadas em dados concretos de intenções e usabilidade. Resumidamente, o modelo por ser descrito em quatro fases distintas:

- 1) O mapeamento das interações da equipe de IHC com o aplicativo possibilitando identificar como devem ser executadas as tarefas providas aos usuários.

- 2) A instrumentação do código da aplicação com métricas de usabilidade, contextuais e subjetivas, para possibilitar a coleta de dados referentes à interação, contexto e percepção do usuário.

3) A utilização, por parte dos usuários de *smartphones*, dos aplicativos instrumentados com as métricas para que os dados possam ser coletados e posteriormente transferidos para uma base de dados.

4) A avaliação dos resultados obtidos ao longo do tempo, considerando aspectos de usabilidade, frequência de uso, contextos, experiência dos usuários, entre outras.

A abordagem definida como objetivo principal desta pesquisa fundamenta-se em três premissas básicas:

(i) A captura dos dados de interação do usuário deverá ser automática.

(ii) As técnicas utilizadas para a captura de dados não devem interferir no código fonte original das aplicações.

(iii) A abordagem proposta deve ser de baixo custo financeiro

Essas premissas impõem extensões de técnicas conhecidas e potencializam a interdisciplinaridade entre várias áreas, tais como: IHC, Engenharia de Software, Banco de Dados (BD) e Sistemas Sensíveis ao Contexto. Sendo assim, são almejados os seguintes objetivos específicos:

- Prover recursos necessários para realizar o mapeamento da execução das tarefas disponibilizadas por uma aplicação de forma fácil e interativa.

- Desenvolver uma biblioteca de métricas de usabilidade para permitir a instrumentação das aplicações que serão avaliadas.

- Definir uma biblioteca de métricas que capturem dados dos sensores, possibilitando que as interações possam ser contextualizadas ao cenário de avaliação.

- Prover recursos que possibilitem a coleta de informações subjetivas referente aos sentimentos dos usuários com relação aos aplicativos utilizados, por intermédio de interlocuções diretas, utilizando a técnica ESM.

- Disponibilizar a estrutura que contemple o armazenamento dos dados capturados e possibilite uma posterior análise.

- Escolher uma ferramenta *Online Analytical Processing* (OLAP) para facilitar a avaliação dos dados capturados.

- Validar as potencialidades do modelo arquitetural proposto nesta tese e sua infraestrutura, denominada de UXEProject. Essa verificação está atrelada a realização de um experimento envolvendo usuários de aplicativos para *smartphones*.

- Descrever todos os processos que contemplem a utilização da infraestrutura UXEProject, desenvolvida para materializar o modelo proposto.

- Avaliar a utilização da infraestrutura por seus potenciais usuários como, por exemplo, Desenvolvedores de Software, Especialistas em Banco de Dados, Engenheiros de Software e Especialista em IHC.

1.4 ESTRUTURAÇÃO DO DOCUMENTO

Esta tese está estruturada em sete capítulos. Neste capítulo, foi apresentadas as motivações, as principais discussões referentes ao tema e os objetivos do estudo. O Capítulo 2 aborda os principais conceitos e trabalhos desenvolvidos para a avaliação de aplicativos criados para dispositivos móveis, levando em consideração as motivações, problemas e objetivos descritos ao longo da introdução. No Capítulo 3, é apresentada a metodologia que foi utilizada durante todos os passos desta pesquisa, enfatizando-se os aspectos científicos do trabalho. No Capítulo 4, é descrito o modelo arquitetura proposto nesta tese, bem como a infraestrutura UXEProject criada para a sua validação, sendo estas as principais contribuições deste trabalho. O Capítulo 5 contempla o experimento realizado com usuários de aplicativos para *smartphones*, tendo como objetivo identificar as potencialidades do modelo e infraestrutura desenvolvidos para a instrumentação de aplicativos para *smartphones*, coleta de dados e avaliação das informações. No Capítulo 6, são apresentados os resultados de um experimento para a avaliação da infraestrutura sob o ponto de vista dos seus potenciais utilizadores, com o objetivo de identificar a sua viabilidade de ser utilizada em larga escala pela comunidade que se interessa pela área de IHC. No Capítulo 7, são descritas as conclusões do trabalho, enfatizando as contribuições, limitações e perspectivas futuras.

1.5 RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO

As características apresentadas pelos *smartphones*, tais como, telas sensíveis ao toque, capacidade para instalar localmente diferentes aplicativos, acesso a distintos tipos de redes, incorporação de inúmeros sensores e grande evolução nos componentes de hardware e software, tornam estes dispositivos os mais vendidos no mercado mundial.

Este fato, aliado aos aplicativos inovadores que estão sendo disponibilizados com diversificados propósitos para uma vasta gama de usuários com diferentes características, abre novos campos de pesquisa.

Uma das áreas potencializadas é a avaliação da usabilidade dos aplicativos para *smartphones* já que diferentes aspectos podem interferir na eficácia, eficiência e facilidade de uso, tais como, o contexto onde as interações ocorrem, a ineficiência das redes e o projeto dos aplicativos.

Muitas discussões em torno do assunto estão sendo promovidas, principalmente no que se refere ao local onde os experimentos de usabilidade devem ser conduzidos (em campo ou laboratório), com relação aos tipos de dados que devem ser coletados e analisados (quantitativos, qualitativos e subjetivos) e referentes à abordagem utilizada.

Diante deste cenário, nesta tese é proposta uma nova abordagem que possibilite realizar testes de usabilidade em campo e em laboratório, considerando as diferentes características dos participantes, independente do número de usuários, sem restrições de tempo e com baixo custo.

Para viabilizar a proposta, será especificado um modelo, sua infraestrutura e a descrição dos processos necessários para a instrumentação de aplicativos para *smartphones*, possibilitando a obtenção de dados quantitativos, contextuais e subjetivos para posterior análise.

A abordagem associa duas técnicas para a captura de dados: (i) o *Logging* para coletar dados das interações com as aplicações e sensores; e (ii) a ESM para obter dados sobre a percepção do usuário.

Para a execução da abordagem, é necessário mapear as tarefas providas aos usuários, instrumentar o código das aplicações com métricas de usabilidade, utilizar os aplicativos instrumentados para coletar os dados e avaliar os resultados.

Com a abordagem proposta, pretende-se ampliar as possibilidades de captura e análise dos dados, contribuindo para significativos avanços na área de avaliações de usabilidade para *smartphones*. A proposta mescla a utilização de técnicas de Engenharia de Software como, Programação Orientada a Aspectos (POA), a utilização da tecnologia existente nos aparelhos como, por exemplo, sensores e acesso a redes de comunicação, além de novas formas de armazenamento e processamento dos dados, oriundas da computação na nuvem (*Cloud Computer*).

Neste capítulo, são abordados os principais tópicos e conceitos relacionados ao tema principal deste trabalho. Inicialmente, são apresentados alguns conceitos clássicos de usabilidade e posteriormente, uma revisão literária dos principais trabalhos que relatam experimentos envolvendo aplicativos para smartphones entre os anos de 2008 a 2012. Os resultados do levantamento bibliográfico realizado são discutidos e sumarizados no final do capítulo.

2 O ESTADO DA ARTE

Os *smartphones* estão associados a um grande número de tarefas no nosso cotidiano, tendo como consequência a criação de uma demanda da sociedade por estes dispositivos, assim como da infraestrutura tecnológica envolvida para o seu funcionamento. Como os *smartphones* são utilizados praticamente em todos os lugares e possuem a capacidade de acessar recursos externos, eles fornecem uma combinação ideal para o desenvolvimento de aplicações dos mais variados tipos.

Um dos grandes desafios é prover aos usuários aplicativos que sejam eficientes, eficazes, atrativos, de fácil aprendizagem, flexíveis, úteis e de simples utilização. Sendo assim, a usabilidade se encaixa como uma das áreas de pesquisa mais importantes no contexto de dispositivos móveis. Como esta área é o foco desta tese, nas próximas seções será abordado este assunto em profundidade.

2.1 DEFINIÇÕES DE USABILIDADE

Com os avanços na área de hardware e software empregados nos *smartphones*, existem muitos tópicos de interesse que estão abertos para novas pesquisas. Nesta seção, o foco será a definição do conceito de usabilidade, a área escolhida para o desenvolvimento desta tese.

A popularização dos *smartphones* tem tornado os sistemas computacionais cada vez mais onipresentes e desenvolvidos para um segmento crescente de usuários com os mais variados perfis. Com isso, as preocupações com a usabilidade destes sistemas tornaram-se maiores do que costumavam ser há alguns anos atrás.

O termo usabilidade é frequentemente usado para se referir à capacidade de um produto, em geral, ser usado de maneira fácil. Porém, somente essa definição simplifica demasiadamente o problema, que envolve atender às expectativas do usuário de forma completa.

A primeira definição de usabilidade foi introduzida por Miller, em 1971, através de métricas relacionadas à “facilidade de uso” (Santos, 2008). Miller identificou vários critérios para mensurar a facilidade de uso, como, por exemplo, tempo de aprendizado, número de erros e tolerância a falhas (Miller, 1971).

Bennett em 1979 deu início à questão da usabilidade no contexto da IHC, área de pesquisa que estuda a comunicação entre pessoas e sistemas, apontando que a usabilidade sugere interação e satisfação dos usuários. Mais tarde, em 1986, Shackel ampliou o conceito de usabilidade incluindo uma visão centrada no usuário e no ambiente em que o sistema está sendo utilizado e apresentou quatro critérios para medir usabilidade: Eficácia, Aprendizagem, Flexibilidade e Atitude (Santos, 2008).

No decorrer dos anos 90, alguns autores já argumentavam que investir na usabilidade de produtos era tão importante quanto investir em suas funcionalidades.

Shackel (1991) concebeu um modelo de percepção de produtos baseado na aceitação. Segundo o autor, a aceitação é diretamente dependente de quatro fatores:

- Utilidade – refere-se ao mapeamento das necessidades dos usuários com as funcionalidades do produto.
- Capacidade de agradar – está relacionada às avaliações subjetivas do usuário.
- Custos – estão vinculados aos encargos econômicos, sociais e organizacionais decorrentes da aceitação do produto.
- Usabilidade – diz respeito à facilidade de utilização prática das funcionalidades do produto.

Nielsen (1993) considerou que a aceitação de um sistema podia ser mensurada a partir da confiabilidade, custos, compatibilidade, aplicabilidade e usabilidade. Para a definição de usabilidade, o autor empregou os seguintes critérios:

- Aprendizado – aprender a utilizar um novo sistema deve ser fácil o suficiente para que o usuário comece a usá-lo rapidamente.
- Eficiência – o sistema deve ser eficiente de tal forma que os usuários possam atingir uma alta produtividade.

- Memorização – um usuário ocasional do sistema não precisa reaprender a utilizar o sistema, o sistema precisa ser intuitivo.

- Erros – o sistema deve se recuperar de erros e avisar ao usuário quando este os cometer.
- Satisfação – o usuário deve ficar satisfeito com o uso do sistema.

A ISO 9241-11 (1998), por sua vez, define usabilidade apenas com três características:

- Efetividade – diz respeito à precisão e à completude com as quais os usuários alcançam metas específicas.

- Eficiência – refere-se aos recursos necessários em face à completude e precisão para que os usuários alcancem suas metas.

- Satisfação – refere-se ao conforto e à aceitação do uso do sistema pelos usuários.

Segundo a ISO/IES 9126-1 (2001), usabilidade é “a capacidade que um produto de software apresenta para ser compreendido, aprendido, utilizado e ser atraente para o utilizador, quando usado em condições específicas”. De forma mais particular, cada atributo pode ser definido como:

1. Inteligibilidade – representa a facilidade com que o usuário pode compreender as suas funcionalidades e avaliar se o mesmo pode ser usado para satisfazer as suas necessidades específicas.

2. Aprendizagem – identifica a facilidade de aprendizado do sistema pelos seus potenciais usuários.

3. Operacionalidade – é como o produto facilita a sua operação por parte do usuário, incluindo a maneira como ele tolera erros de operação.

4. Atratividade – envolve características que possam atrair um potencial usuário para o sistema, o que pode incluir desde a adequação das informações prestadas para o usuário até os requintes visuais utilizados na sua interface gráfica.

Shneiderman (2004) apresenta cinco “regras de ouro” que são fatores mensuráveis para avaliação da usabilidade e qualidade de uma interface de sistema. Essas regras são bem parecidas com as apresentadas por Nielsen (1993), porém apresentadas de uma forma mais objetiva.

1. Tempo de aprendizagem – Tempo necessário para um usuário iniciante aprender os comandos com o objetivo de executar uma determinada tarefa.

2. Velocidade de realização – Tempo necessário para um usuário concluir uma tarefa do sistema.

3. Taxa de erro dos usuários – Número e tipo de erros cometidos pelos usuários.

4. Retenção ao longo do tempo – Forma pela qual os usuários utilizam o sistema depois de passado determinado tempo (horas, dias ou semanas). Esta medida está relacionada com o tempo de aprendizagem e com a frequência de utilização.

5. Satisfação subjetiva – Grau de satisfação do usuário ao utilizar diversas funcionalidades do sistema.

Com o objetivo de aprimorar os conceitos anteriores sobre usabilidade, Zhang e Adipat (2005) redefiniram alguns atributos e propuseram novos atributos que devem ser considerados nos teste de usabilidade de aplicativos:

1. Capacidade de aprendizado – a facilidade com que os usuários podem terminar uma tarefa pela primeira vez, usando uma aplicação (facilidade de utilização).

2. Eficiência – rapidez com que os usuários podem realizar uma tarefa utilizando um aplicativo.

3. Nível de memorização – facilidade com a qual os usuários podem se lembrar de como usar uma aplicação.

4. Erros – os erros que um usuário comete.

5. Satisfação – a atitude demonstrada pelo usuário ao utilizar um aplicativo.

6. Integridade – eficácia e precisão com que os usuários alcançam seus objetivos.

7. Simplicidade – o grau de conforto disponibilizado pelo aplicativo para a execução de suas tarefas.

8. Compreensibilidade – a facilidade que os usuários possuem para entender o conteúdo e a proposta apresentada pelo aplicativo.

9. Desempenho na aprendizagem – eficácia de aprendizagem ao utilizar o aplicativo.

Segundo Zhang e Adipat (2005), a lista apresentada acima deve ser vista apenas como orientações sobre o que uma avaliação de usabilidade pode conter e não como uma lista de verificação que os testes devem contemplar.

Coursaris e Kim (2011) propuseram um *framework* que contempla fatores contextuais na avaliação de usabilidade para dispositivos móveis. A estrutura é ilustrada na Figura 1 e contém três elementos. O primeiro, o círculo externo, mostra os quatro fatores contextuais definidos pelos autores como fatores impactantes na usabilidade de dispositivos móveis (usuários, ambiente, tecnologia e tarefas/atividades). O segundo, o círculo interno, mostra as principais dimensões de usabilidade (eficácia, eficiência, satisfação, capacidade de aprendizado, flexibilidade, operabilidade, entre outros). O terceiro elemento é representado por um retângulo localizado acima dos fatores contextuais, onde são listadas as consequências (melhorar a integração com o sistema, aumentar a adoção, promover a retenção, entre outros).

No trabalho mais recente, proposto por Hansen (2012), foram identificados quatro desafios importantes na área de usabilidade para *smartphones* que merecem estudos específicos já que, na visão do autor, interferem diretamente na usabilidade destes dispositivos. São eles:

- Contexto – a capacidade de identificar o ambiente e ajustar o conteúdo do aplicativo com base nas variáveis do ambiente.

- Multimodalidade – combinação de vários modos de interação com os aplicativos, tais como a utilização de comandos de voz, toques na tela e movimento do dispositivo.
- Conectividade – como lidar com as variações das condições da rede.
- Métodos de entrada de dados restritivos – botões e descrições com tamanho reduzido interferem na eficácia e eficiência das entradas de dados.

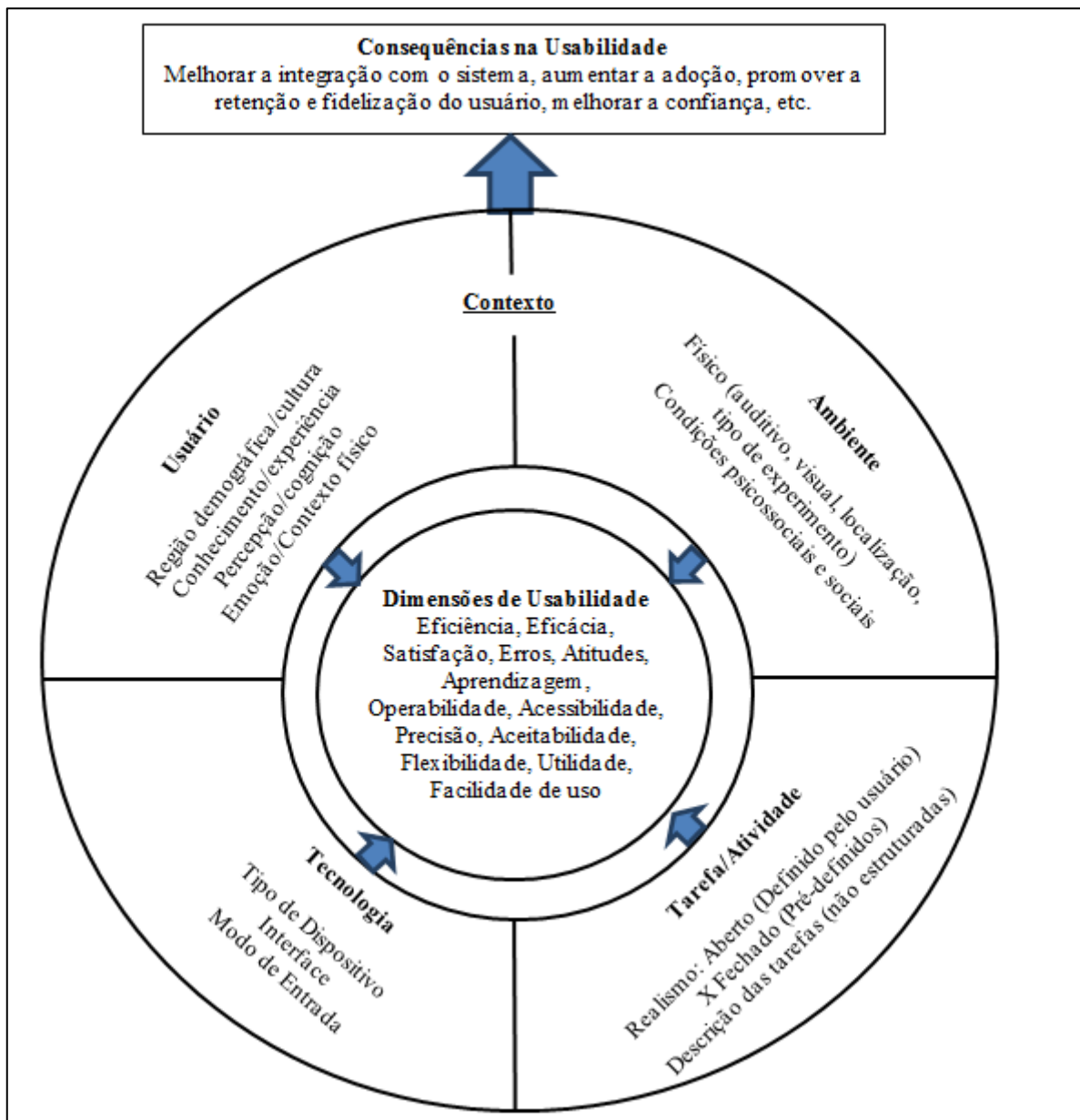


Figura 1 – *Framework* sugerido por Coursaris e Kim (2011).

Considerando todas as definições de usabilidade apresentadas nesta seção e tomando como base o *framework* proposto por Coursaris e Kim (2011), faz-se necessário unificar os principais atributos para que

possam ser empregados no restante desta tese sem ambiguidades. Além disso, será necessário descrever o significado dos atributos considerados para que não exista margem para equívocos.

Os atributos de usabilidade e os termos adotados no contexto desta pesquisa podem ser contemplados na Tabela 1. É importante salientar que muitos atributos citados anteriormente foram unificados por serem considerados sinônimos ou representarem praticamente a mesma coisa. Por exemplo, os atributos “erro”, “eficácia” e “precisão” foram unificados e denominados apenas de “eficácia”. Sendo assim, serão considerados apenas nove atributos daqui em diante. Além disso, a proposta apresentada nesta tese possibilita a investigação da usabilidade tomando como base esses atributos.

Tabela 1 – Definição dos atributos que caracterizam usabilidade no escopo deste trabalho

Atributo	Descrição
Eficiência	Rapidez com que os usuários realizam uma tarefa no aplicativo.
Eficácia	Precisão com que os usuários realizam uma tarefa no aplicativo.
Satisfação	Nível de aceitação do usuário quanto à utilização do aplicativo.
Aprendizagem	Tempo necessário para o usuário utilizar o aplicativo de forma eficiente.
Operabilidade	Capacidade do aplicativo se manter operacional independente das circunstâncias.
Acessibilidade	Adaptabilidade do aplicativo às necessidades especiais ou restrições físicas do usuário.
Flexibilidade	Formas de interação disponibilizadas pelo aplicativo para realizar uma mesma tarefa.
Utilidade	Conformidade entre as tarefas disponibilizadas e o objetivo do aplicativo.
Facilidade de uso	Nível de entendimento por parte dos usuários de como executar uma tarefa no aplicativo.

A experiência do usuário (*User eXperience* - UX) é outro termo amplamente mencionado em contraste com a usabilidade. Embora algumas pessoas usem os termos como sinônimos, a experiência do usuário é um conceito mais amplo do que a usabilidade (Saffer, 2007). A usabilidade está preocupada principalmente com as características de design de produtos interativos em termos de como é fácil usá-lo. Por outro lado, a UX tem foco na interação do indivíduo com um determinado produto, levando em consideração os pensamentos, sentimentos e percepções do resultado dessa interação (Tullis e Albert, 2008). Os conceitos referentes à UX, portanto, englobam outras questões como, por exemplo, credibilidade e envolvimento.

O objetivo da UX é identificar se os aplicativos são atraentes o suficiente para causar às pessoas a sensação de imersão que sentimos quando lemos um bom livro, a sensação de desafio durante um bom jogo ou a fascinação diante do desenrolar de um drama (Benyon, 2005). Desta forma, a UX extrapola o tema proposto nesta pesquisa.

2.2 AVALIAÇÕES DE USABILIDADE

A área de usabilidade pode trazer benefícios para a engenharia de software, embora ainda existam alguns obstáculos a serem transpostos como, por exemplo, a necessidade de mudar a mentalidade dos desenvolvedores para que seus produtos sejam centrados no usuário (Bak et al., 2008). A avaliação de usabilidade foi proposta principalmente para trazer benefícios econômicos, tais como, o aumento das vendas, maior produtividade por parte do Usuário Final, diminuição dos custos de capacitação e redução da necessidade de centrais de ajuda (Bak et al., 2008).

A avaliação da usabilidade dos dispositivos móveis e dos seus aplicativos associados é fator determinante para o sucesso deste segmento (Zhang e Adipat, 2005). As necessidades e características dos usuários, os contextos de uso, as especificidades dos aplicativos e as limitações físicas são fatores que influenciam a interação e devem ser considerados tanto no projeto das interfaces quanto na avaliação de usabilidade (Betioli e Cybis, 2005).

Segundo Kangas e Kinnunen (2005), a avaliação da usabilidade é uma atividade essencial para garantir a perfeita utilização dos dispositivos móveis, sendo necessário ser empregada em todas as fases do ciclo de vida de um projeto de software. Na área de IHC, estas avaliações podem ser classificadas como Formativas ou Somativas. Caso o software esteja em fase de desenvolvimento, a avaliação é classificada como Formativa. Por outro lado, caso o software já esteja pronto, a avaliação é chamada de Somativa (Barbosa e Silva, 2010).

A avaliação Formativa é caracterizada fundamentalmente por processos contínuos de investigação, adaptáveis tanto ao processo global da interface quanto a aspectos específicos, emergentes em decorrência da investigação e da modificação de partes da interface.

As principais características da avaliação Formativa são:

1. Oferecer informações que podem ser utilizadas nas decisões do projeto.
2. Realizada durante a elaboração do projeto, utiliza rascunhos ou protótipos.
3. Requer um número reduzido de participantes nos experimentos que normalmente são os potenciais usuários.
4. Visa indicar problemas, apontando soluções que devem ser aplicadas antes do projeto ser finalizado.

A avaliação Somativa, por sua vez, tem o propósito de fornecer diagnósticos globais da interface investigada ao término de diferentes etapas de seu desenvolvimento. Estes diagnósticos se fundamentam usualmente em parâmetros de referência e confrontações entre metas pré-fixadas e os

resultados atingidos. Além disso, podem incluir a avaliação das reações do usuário ao interagir com o aplicativo.

A avaliação Somativa possui as seguintes características:

1. Verifica se o sistema está cumprindo com os seus objetivos de forma eficaz e eficiente.
2. Os dados coletados são normalmente quantitativos através de respostas numéricas, entretanto, também podem ser considerados dados subjetivos quando retratam as experiências dos usuários durante a interação com o aplicativo.
3. Normalmente, são diagnosticados problemas de interface e comunicabilidade.
4. Nem sempre são identificadas as causas dos erros, mas os dados coletados devem propiciar indícios a respeito das causas e forma de solucioná-los.

Na literatura, são abordados diferentes métodos de avaliação de usabilidade. Para uma contextualização, a Tabela 2 apresenta os diferentes aspectos envolvidos nos métodos de avaliação de usabilidade conforme a categorização proposta por Sharp et al. (2007).

Tabela 2 – Diferenças entre os métodos de avaliação

Aspectos	Diferenças
Momento da Avaliação	Avaliação Formativa – Durante o ciclo de desenvolvimento. Utiliza cenários, <i>storyboards</i> , modelagens conceituais e protótipos do aplicativo.
	Avaliação Somativa – Após o produto estar pronto. Verificar a conformidade do aplicativo com padrões previamente estabelecidos.
Técnica para Coleta de Dados	Através da opinião dos usuários – Utiliza questionários ou entrevistas com a finalidade de mensurar o nível de satisfação do usuário.
	Observação dos usuários em seu ambiente natural – Utiliza recursos para a obtenção dos dados, tais como: câmeras, gravadores, arquivos de <i>logs</i> e sistemas paralelos para quantificar, através de métricas, a interação do usuário com o aplicativo.
	Observação dos usuários em laboratório – Permite um controle mais rígido sobre as ações e os dados que serão coletados durante os experimentos.
	Coleta da opinião de especialistas – Os especialistas examinam a interface e identificam potenciais dificuldades para os usuários.
Tipo de Dado Coletado	Quantitativo – São os dados que representam a grandeza resultante de uma contagem ou medição como, por exemplo, o tempo gasto para completar uma tarefa.
	Qualitativo – São dados obtidos por análises interpretativas.
	Subjetivo – São dados que precisam ser explicitamente expressos pelos participantes da avaliação, como opiniões, preferências e satisfação.
Tipo de Avaliação	Preditiva – É caracterizada pela análise de especialistas de IHC em uma determinada aplicação.
	Interpretativa – Os dados são obtidos durante a interação do usuário.
	Experimental – Os dados são obtidos durante a interação do usuário, entretanto, em ambientes controlados como os laboratórios.

A principal distinção entre os métodos de avaliação está relacionada com a técnica utilizada para a coleta de dados. Segundo Maguire (2001), as técnicas podem ser divididas em função da participação ou não dos usuários durante as avaliações. Os métodos Analíticos são avaliados normalmente utilizando protótipos e modelos formais de um sistema específico, com a participação de especialistas da área de IHC e desenvolvedores. Por outro lado, os métodos Empíricos utilizam os usuários para a coleta de dados. Segundo o autor, os métodos Empíricos ainda podem ser subdivididos em:

- Informais – observação direta das interações dos usuários.
- Formais – elaboração de experimentos, questionários e entrevistas.

Na Tabela 3, é apresentada uma descrição resumida das principais técnicas empregadas para a avaliação da usabilidade de softwares, segundo a estruturação proposta por (Barbosa e Silva, 2010).

Tabela 3 – Relação das técnicas de avaliação

Técnicas de Avaliação	Descrição
Analíticos – Não utilizam a participação de usuários	
Avaliação Heurística (Nielsen e Mack, 1994)	Envolve especialistas que avaliam o <i>design</i> com base em um conjunto de critérios de usabilidade ou heurísticas. O <i>design</i> é examinado em busca de instâncias nas quais esses critérios são violados.
Percurso Cognitivo (Nielsen e Mack, 1994)	Avalia uma proposta de projeto de IHC no contexto de tarefas específicas. É uma técnica na qual um avaliador percorre caminhos por uma determinada interface buscando colocar-se no lugar do usuário. O objetivo é verificar se há problemas potenciais para o usuário, além de tentar prever onde haverá dificuldades de interação.
Inspeção Semiótica (Souza, 2005)	Avalia a qualidade de comunicação de uma interface de acordo com o ponto de vista do <i>designer</i> .
Empíricos – Utilizam dados coletados junto aos usuários	
Teste de Usabilidade (Rubin e Chisnell, 2008)	O objetivo é avaliar a interação do usuário de acordo com critérios previamente definidos. Normalmente, os dados para comparação são capturados durante a interação do usuário com o sistema.
Avaliação de Comunicabilidade (Nielsen e Mack, 1994)	Avalia a qualidade de comunicação de uma interface de acordo com o ponto de vista do usuário.
Prototipação em Papel (Snyder, 2003)	Avalia a usabilidade de um protótipo representado em papel através de simulações de uso com a participação de usuários.

A adoção de uma determinada técnica é influenciada pelo estágio em que se encontra o software no seu ciclo de desenvolvimento (Maguire, 2001). Por exemplo, algumas são mais apropriadas para avaliações antes do aplicativo estar pronto, como as Avaliações Heurísticas, Percurso Cognitivo e Prototipação em Papel. Por outro lado, Inspeção Semiótica, Avaliação de Comunicação e Teste de Usabilidade são mais adequados para avaliações após o produto estar em fase de teste ou ser colocado em produção.

De acordo com Zhang e Adipat (2005), independente da técnica utilizada para avaliação de aplicações para dispositivos móveis, é fundamental que as avaliações levem em consideração as possíveis influências que o meio externo pode ocasionar nas interações. Em comparação com outras áreas de desenvolvimento de software, é possível afirmar que existem desafios específicos a serem transpostos no caso dos dispositivos móveis. Conforme Kenteris et al. (2009), estudos de usabilidade envolvendo dispositivos móveis devem contemplar as limitações das redes sem fio, a adaptação a diferentes contextos de interação e a influência das aplicações no cotidiano do usuário. Estes aspectos, geralmente não são importantes em estudos de usabilidade em softwares desenvolvidos para *desktops*. Além disso, avaliações que abrangem *smartphones* é uma área ainda pouco estudada, devido aos frequentes avanços tecnológicos observados nos últimos anos (Coursaris e Kim, 2011). Atualmente, pode-se afirmar que a definição de um método geral para a avaliação efetiva de dispositivos móveis é uma questão em aberto (Jones e Marsden, 2006) (Hansen, 2012), principalmente em função da escassez de técnicas que possibilitem coletar e relacionar diferentes tipos de dados.

2.3 TESTES DE USABILIDADE

Os testes de usabilidade de artefatos de software são importantes para compreender o sistema a partir da perspectiva dos usuários, ou seja, para obter informações relevantes dos usuários com relação ao uso do sistema. Eles são úteis para encontrar problemas nos recursos existentes e identificar os aspectos que estão adequados. É importante entender o que os usuários querem do produto e se eles estão satisfeitos com as atuais características, com o objetivo de tornar o produto mais fácil de ser utilizado (Krug, 2009).

Apesar das pesquisas de usabilidade de software abrangerem uma variedade de áreas como aplicativos móveis comerciais (Venkatesh et al., 2003), jogos (Korhonen e Koivisto, 2006) e *Web sites* (Petrie e Power, 2012), esta tese se concentra na investigação de usabilidade para *smartphones*.

Há pelo menos três razões principais para a escolha desta área:

- Primeiramente, as pesquisas com foco em usabilidade para *smartphones* ainda são escassas já que estes dispositivos começaram a ser utilizados efetivamente em 2008 (Hansen, 2012).
- Em segundo lugar, existe a possibilidade da utilização dos sensores para auxiliar na coleta de diferentes informações que podem melhorar a qualidade das avaliações.
- Por fim, há novas restrições introduzidas por estes aparelhos, tais como: (i) limitações das redes sem fio; (ii) fatores contextuais que podem interferir na usabilidade; (iii) diferentes formas de

interação; e (iv) uma vasta gama de novos aplicativos desenvolvidos e disponibilizados por desenvolvedores com pouca experiência nesta plataforma (Grønli, 2012).

Nas próximas subseções, será apresentado um apanhado das principais pesquisas empíricas publicadas nos últimos cinco anos na área de avaliação de aplicativos para *smartphones*. Para facilitar o posicionamento cronológico, será feita uma divisão de acordo com o ano da publicação dos trabalhos.

2.3.1 Trabalhos publicados em 2008

O ano de 2008 foi marcado pelo início da proliferação acentuada dos *smartphones* no mercado de dispositivos móveis. Apesar disso, já nesta época são observados pelo menos dois trabalhos relevantes na área de avaliação de usabilidade de aplicativos para *smartphones*.

Burigat et al. (2008) realizaram uma avaliação sobre três técnicas de navegação (*DoubleScrollbar*, *Grab&Drag* e *Zoom-Enhanced Navigator – ZEN*) utilizadas em *smartphones* para possibilitar a apresentação de informações que ultrapassam o tamanho da tela do dispositivo. A intenção dos autores foi medir o desempenho e satisfação dos usuários quando submetidos a visualizar mapas e Web sites em telas pequenas. Os resultados da pesquisa foram constatados em função da análise do tempo gasto para o usuário completar as tarefas, o número de ações necessárias, a precisão espacial e as preferências subjetivas dos participantes com relação às técnicas de navegação investigadas.

Os autores concluíram que a falta de informações do nível de zoom na técnica ZEN tornava a interação do usuário menos intuitiva do que nas outras técnicas, onde esta informação sempre está disponível. Embora esta questão não afete significativamente a quantidade de ações realizadas pelos usuários, a falta de atualizações interativas pode trazer problemas de usabilidade. As avaliações identificaram ainda que, embora as três técnicas forneçam um planejamento sequencial de navegação, a técnica ZEN possibilita o melhor mapeamento mental do espaço de informação considerado e mais precisão na execução das tarefas propostas. Por outro lado, foi observado que os usuários tiveram mais dificuldade de iniciar as tarefas utilizando a técnica ZEN, já que requer o toque em regiões precisas da tela para iniciar a visualização do alvo desejado.

Sodnik et al. (2008) desenvolveram um experimento para avaliar diferentes formas de interação de usuários com *smartphones* durante a condução de veículos. A pesquisa contou com a utilização de um simulador e tinha como objetivo avaliar a eficiência e a eficácia de duas interfaces auditivas com relação a uma interface tradicional (visual). Os dados coletados e analisados envolveram o desempenho do condutor, a

conclusão das tarefas propostas, a carga de trabalho necessária para realizar as interações, a percepção e a satisfação dos usuários.

Os autores concluíram que as duas interfaces auditivas foram eficazes no cenário proposto, porém não permitiram interações mais rápidas que a interface visual principalmente no caso de tarefas curtas como, por exemplo, apagar uma mensagem. Outro fato observado foi que o desempenho na condução do veículo é melhor com interfaces auditivas e a carga de trabalho necessária para completar as tarefas é menor. Além disso, os resultados comprovaram maior satisfação dos usuários quando utilizaram as interfaces auditivas. Os resultados do estudo não são surpreendentes, já que existe uma forte competição por atenção visual entre a interface visual e a tarefa primária (condução do veículo). Assim, constatou-se que ambos os tipos de interfaces são eficazes, mas a interface visual foi menos eficiente já que tende a distrair o motorista.

2.3.2 Trabalhos publicados em 2009

Em 2009, a expansão das vendas de *smartphones* e a crescente disponibilidade de novos aplicativos com inúmeras funcionalidades práticas aumentaram o interesse dos pesquisadores da área de IHC em estudos empíricos de usabilidade para estes dispositivos.

Os pesquisadores Fitchett e Cockburn (2009) compararam as técnicas de rolagem de tela Flick e Tilt, muito utilizadas em *smartphones*. No experimento realizado, foram comparadas as duas técnicas para a leitura de textos e execução de tarefas. As avaliações foram realizadas com os usuários parados e caminhando. As principais conclusões evidenciadas pelos autores foram: (i) a técnica Flick é melhor para a execução de tarefas quando os usuários se encontram parados; (ii) ambas as técnicas apresentaram resultados semelhantes quando as tarefas são realizadas com o usuário em movimento; e (iii) os usuários preferem a técnica Flick.

Nas Olimpíadas de Beijing – China em 2008 foi disponibilizado vários aplicativos por meio de um portal Web para que os espectadores pudessem obter informações atualizadas nos seus *smartphones*. Os aplicativos correspondiam a um guia olímpico, um mapa para locomoção na cidade, um menu de leitura dos destaques e uma rede de relacionamento (*phrasebook*) que permitia compartilhar fotos. Chin e Salomaa (2009) realizaram um trabalho de investigação referente à usabilidade das aplicações, a experiência dos usuários e o número de acessos de cada participante.

Os resultados apontaram uma grande popularidade do portal, com expressiva utilização dos mapas e compartilhamento de fotos. Segundo os participantes do experimento, o mapa para localização e o

menu de destaques foram os mais úteis. No entanto, o *phrasebool* foi considerado o aplicativo mais fácil de ser usado, seguido do guia olímpico.

Lai et al. (2009) desenvolveram uma aplicação para *smartphones* chamada Life360, com a finalidade de investigar a atitude e o comportamento das pessoas com relação ao ambiente de interação. O objetivo da pesquisa é propor uma nova abordagem para identificar os diferentes estilos de vida e personalidades que caracterizam uma determinada população. A aplicação, apresenta questionários aos usuários, com uma frequência de 8 a 12 interlocuções diárias, coletando informações referentes à localização atual dos entrevistados, a atividade em que estão envolvidos, as pessoas que estão no ambiente, o estado emocional dos entrevistados, entre outras questões.

Além dos dados referentes ao cotidiano dos participantes, os autores realizaram entrevistas cognitivas para avaliar o Life360 com relação ao projeto da interface, avaliação da usabilidade e a execução das tarefas durante todo o período de realização do experimento.

Com relação aos resultados referentes à usabilidade, os participantes reclamaram basicamente dos seguintes fatores:

- Ter necessariamente de rolar a tela até o final das possíveis respostas, mesmo se já tivessem escolhido a resposta desejada, em função das mesmas não ficarem visíveis em uma única tela.
- Ter de utilizar uma caneta para selecionar botões de rádio, devido ao pequeno diâmetro deste componente definido na aplicação.
- A difícil interação com as teclas do *smartphone* quando eram submetidos a responder questões abertas.

Em outro trabalho publicado por Ebner et al. (2009), foi avaliado um aplicativo no contexto da aprendizagem ubíqua (*u-Learning*), desenvolvido pela *Graz University of Technology* – Áustria. O aplicativo foi concebido como uma ferramenta para gerenciar o aprendizado.

O objetivo do experimento foi investigar a usabilidade do aplicativo baseado em metáforas que generalizam o comportamento do usuário, contexto e interação. Foram também considerados seis fatores que contemplam a experiência dos usuários: atratividade, clareza, eficiência, segurança, estímulo e inovação.

O estudo propôs seis tarefas para cada participante e revelou diferentes problemas de usabilidade no aplicativo, tais como:

- A necessidade de utilização de *zoom* e barras de rolagem na maioria das telas.
- As telas que não possibilitavam a utilização do *zoom* dificultaram a interatividade e compreensão do conteúdo.
- Em alguns locais, o espaço entre as opções de interação foram projetadas com pouca distância entre elas, inviabilizando a interação por meio de toques na tela.

- O aplicativo não permitia o armazenamento do *login* e senha do usuário para facilitar um novo acesso.
- As listas de discussões e fóruns tinham uma estruturação confusa de difícil entendimento por parte dos usuários.

Em um trabalho para avaliar as potencialidades da utilização do *Bluetooth*, Hansen e Ghinea (2009), criaram um aplicativo para as plataformas Java ME e Windows Mobile com o objetivo de controlar remotamente um computador pessoal. Um exemplo prático da utilização do aplicativo é manejar uma apresentação no Microsoft PowerPoint a partir do *smartphone*, permitindo avançar ou retroceder os *slides* de acordo com a necessidade do apresentador. Como o aplicativo possibilita mapear no *smartphone* qualquer tecla de um *desktop*, inúmeros outros aplicativos podem ser controlados a distância, limitado ao alcance do *Bluetooth*.

A usabilidade do aplicativo foi avaliada com o auxílio de 16 pessoas, sendo todos funcionários de uma grande empresa na área de tecnologia da informação. As tarefas propostas aos voluntários foram: instalar o aplicativo, acionar o *Bluetooth* do *smartphone* e estabelecer conexão com o *Bluetooth* do *desktop*, mapear as teclas do *desktop* no *smartphone* e utilizar o aplicativo para controlar remotamente o *desktop*.

Após a execução das tarefas, os voluntários responderam um questionário com vinte e duas perguntas, com respostas formatadas em função da escala de Likert (1932) com seis opções. O objetivo do questionário foi avaliar a usabilidade do aplicativo em função dos seguintes atributos: aprendizagem, eficácia, facilidade de uso e satisfação. Nos resultados obtidos, foi possível observar certo descontentamento com relação à tarefa de estabelecer a conexão com o *Bluetooth* do *desktop* e, em certos casos, dificuldade para instalar o aplicativo devido à incompatibilidade de versões do Java. Por outro lado, nos demais itens avaliados, foram constatados que o escore médio para cada pergunta ficou acima de 4.3, logo a maioria dos participantes do experimento ficou satisfeito ou muito satisfeito.

O último trabalho analisado nesta subseção é o de Bødker et al. (2009). Nele, as investigações concentraram-se nos aplicativos que disponibilizam as principais funcionalidades dos *smartphones*, tais como: envio de e-mail, acesso a Internet, troca de mensagens via SMS, uso da câmera fotográfica, utilização do GPS e MP3. O objetivo dos pesquisadores foi correlacionar os interesses dos usuários com as funcionalidades disponíveis para responder a seguinte pergunta: Qual é o modelo de negócios eficaz para abordar a relação entre a preferência do usuário e o ajuste do *smartphone* para as suas tarefas diárias?

Os resultados obtidos apontaram que o contexto tem uma profunda influência sobre a percepção da utilidade e usabilidade na execução de uma série de tarefas. Por exemplo, os participantes mencionaram a falta de privacidade das mensagens SMS recebidas, já que aparecem no visor sem um comando prévio do usuário, permitindo que as pessoas próximas possam ler a mensagem. Outro fato citado foi o tempo de espera entre a entrada de um caractere e a disponibilidade do dispositivo para receber outro caractere,

dificultando a usabilidade. Por fim, os participantes relatam que a utilização do GPS é extremamente precária devido à lentidão para disponibilizar a localização, necessita um acesso rápido e estável com a Internet, não disponibiliza instruções de áudio, etc.

Os autores concluíram que os *smartphones* e seus aplicativos devem ser disponibilizados com facilidades de ajustes de configurações compatíveis com as necessidades e características dos usuários, bem como de acordo com o contexto de utilização.

2.3.3 Trabalhos publicados em 2010

O primeiro trabalho descrito nesta subseção aborda uma nova proposta de interface para ser utilizada no contexto da Web 2.0. A sugestão apresentada por Kim et al. (2010) define uma estrutura de navegação baseada em *Tags* com múltiplos níveis de botões. O objetivo da abordagem é ajustar a navegação na Internet, criada para *desktops*, a fim de contemplar os requisitos necessários para a utilização em *smartphones*.

Para avaliar a nova estrutura de navegação, foi desenvolvido um experimento com 33 pessoas, sendo 17 homens e 16 mulheres, com idades que variam de 20 a 28 anos, sendo 87,9% estudantes universitários em nível de graduação. A tarefa proposta no experimento foi encontrar fotos e vídeos, sem um limite de tempo estipulado, usando a estrutura antiga e a nova. Os participantes foram divididos aleatoriamente, 16 utilizaram a estrutura convencional de navegação e 17 utilizaram a nova estrutura proposta na pesquisa. Após a execução da tarefa os usuários foram solicitados a responder um questionário com o objetivo de avaliar a percepção subjetiva dos participantes, contemplando a utilidade, prazer, facilidade de uso, satisfação e intenção futura. Os participantes responderam a cada pergunta em função da escala de Likert (1932) com sete opções.

Os resultados encontrados após a execução do experimento estão na Tabela 4. Conforme os valores médios apresentados é possível concluir que a nova abordagem teve uma boa aceitação por parte dos participantes do experimento, com uma avaliação superior em todos os quesitos analisados, quando comparada com a abordagem tradicional.

Tabela 4 – Investigação da estrutura de navegação Web móvel (Kim et al., 2010).

Atributos Avaliados	Nova Interface	Interface Tradicional
Utilidade	5,82	3,44
Prazer	5,80	4,15
Facilidade de Uso	5,63	5,54
Satisfação	5,88	3,69
Intenção de uso futuro	5,68	3,13

Li e Yeh (2010) realizaram um estudo sobre a adaptação do comércio eletrônico para *smartphones*. No trabalho, eles propõem um novo modelo de projeto para aplicativos desta natureza, correlacionando a estética do projeto à utilidade, à facilidade de uso e à customização com a finalidade de aumentar a confiança do consumidor.

Para validar o modelo, os autores realizaram um experimento conduzido com a participação de 200 estudantes universitários tailandeses, utilizando um simulador onde os usuários eram submetidos a realizar compra de câmeras fotográficas, aluguel de carro e reserva de hotéis. Após a execução das tarefas, os usuários foram solicitados a responder um questionário moldado com a escala de Likert (1932) com cinco opções. Os resultados apontaram, em todos os itens avaliados, uma aceitação média acima de 70%, o que demonstra a satisfação da maioria dos participantes do experimento à proposta do trabalho.

Outro trabalho relevante foi proposto por Maly et al. (2010). Os autores desenvolveram uma ferramenta chamada de *Integrated Interactive Information Visualization Environment (IVE)* que permite a coleta e análise de dados para a avaliação de usabilidade de aplicativos para *smartphones*. O estudo de caso proposto para validar a ferramenta IVE baseou-se na instrumentação de um aplicativo chamado *NaviTerier*, que tinha como objetivo conduzir deficientes visuais em casas e edifícios desconhecidos (Vystrcil, 2008). O sistema funcionava como um guia e proporcionava ao usuário uma descrição do local e, com um sintetizador de voz, fornecendo as instruções de navegação.

O objetivo de testar a usabilidade do *NaviTerier* foi avaliar se as instruções eram apropriadas, possibilitando que o usuário receba informações fáceis de serem memorizadas e permitindo a sua livre movimentação pelo ambiente. O experimento contou com a participação de 12 usuários e a única tarefa avaliada foi percorrer o prédio histórico da Faculdade de Engenharia Elétrica da Universidade Czech Technical, situada em Praga. Foram coletados quatro tipos de dados para a análise de usabilidade: (i) o arquivo de *log* gerado pela aplicação; (ii) imagens de um câmara de vídeo; (iii) gravação de áudio; e (iv) observação direta dos avaliadores.

Os autores concluíram, após o experimento, que apesar de terem detectado os mesmos problemas de usabilidade com e sem a utilização da ferramenta IVE, seria necessário o desenvolvimento de novos componentes que possibilitassem a análise dos dados de forma mais fácil e eficiente. Sendo assim,

desenvolveram seis novos componentes para auxiliar nas análises e realizaram um novo experimento para avaliar a ferramenta.

No segundo experimento, foi utilizado praticamente o mesmo cenário, com os mesmos equipamentos e participantes. Apenas foram introduzidas algumas modificações no percurso. A conclusão foi que a ferramenta IVE possibilitou a análise dos dados de forma mais rápida e precisa quando comparado à avaliação dos dados do primeiro experimento. Com relação à usabilidade do aplicativo NaviTerier, foi relatado que, nas situações onde existiam portas com acesso por meio de cartões de identificação, o aplicativo não apresentava bons resultados e exigia inúmeras repetições das instruções para o usuário concluir o percurso estabelecido.

O último trabalho descrito aqui trata de um aplicativo desenvolvido para gerenciar uma sala de reuniões inteligente. O objetivo do aplicativo proposto por Grønli et al. (2010) foi prover uma metodologia eficaz de compartilhar informações, distribuir tarefas, redigir atas, compartilhar documentos e prover recursos contextuais úteis sem a necessidade de equipamentos adicionais.

O aplicativo foi dividido em três módulos:

- O módulo do apresentador da reunião – serve para gerenciar os recursos utilizados na apresentação (*slides*, vídeos, entre outros) e foi desenvolvido para a plataforma Android.
- O módulo dos participantes – permite que os participantes manipulem os documentos disponibilizados e possui uma versão para as plataformas Java ME, Windows Mobile, Android e iPhone OS.
- O módulo servidor – utiliza os conceitos de computação na nuvem para prover o armazenamento e disponibilizar o compartilhamento de informações, além de prover a integração com sistemas externos como o *Google Calendar* e o *Google Docs*.

Para a adesão ao sistema, os participantes devem inicialmente fazer um cadastro no módulo servidor, informando seus dados pessoais, *e-mail* e o endereço de *Bluetooth* do seu *smartphone*. As reuniões são previamente agendadas em um calendário *online*. Quando os participantes se aproximam do local onde a reunião será realizada o sistema envia uma mensagem de boas-vindas automaticamente para os seus *smartphones*. Após o início da reunião, os participantes inscritos vão recebendo informações adicionais ao assunto que está sendo apresentado de acordo com a sequência e velocidade do apresentador. Os participantes podem navegar nas informações recebidas, criar suas próprias observações e destacar partes importantes para que possam ser recuperadas posteriormente.

Para a execução do teste de usabilidade do aplicativo, foram recrutados 40 participantes inseridos no mercado de trabalho ou estudantes universitários, com conhecimento prévio da utilização de *smartphones* e com idades que variam de 20 a 55 anos.

Foram propostas dez tarefas para serem executadas pelos participantes como, por exemplo, registrar-se no sistema, verificar a mensagem de boas vindas, verificar as informações enviadas após a mudança de um *slide*, navegar nas informações recebidas, entre outras.

Após realização do experimento, os participantes responderam um questionário com quinze perguntas contendo respostas nos moldes da escala de Likert (1932) com quatro opções. Os atributos analisados foram: a interface do aplicativo, o desempenho do usuário, as informações providas pelo aplicativo e a impressão geral com relação à abordagem proposta.

As conclusões dos pesquisadores foi que os usuários tiveram um bom desempenho, conseguiram utilizar o aplicativo de forma eficiente e eficaz, ficaram satisfeitos com a interface proposta e compreenderam perfeitamente as funcionalidades disponibilizadas. Além disso, os pesquisadores concluíram que os participantes não ficaram constrangidos devido aos dados serem armazenados em um servidor na nuvem.

2.3.4 Trabalhos publicados em 2011

Em 2011, a venda de *smartphones* ultrapassou a aquisição de computadores pessoais, assim, o primeiro trabalho investigado avalia os fatores que afetam esta preferência, com o foco nos consumidores.

Inicialmente, Kang et al. (2011) adotaram o *Technology Acceptance Model* (TAM) que é um dos métodos mais populares para analisar a aceitação de novas tecnologias e consiste em cinco atributos de investigação: utilidade, facilidade, atitude, intenção e uso real do sistema. Os autores conduziram uma pesquisa em campo com 100 participantes que responderam a um questionário seguindo as opções de respostas definidas na escala de Likert (1932) com cinco opções. Posteriormente, os autores usaram o *Structural Equation Model* (SEM) para realizar uma análise empírica com a participação de 262 alunos de graduação ou pós-graduação coreanos. O questionário respondido pelos participantes também seguiu as opções de respostas definidas na escala de Likert, mas agora, com sete opções.

O objetivo da pesquisa foi investigar e compreender as características de mercado para *smartphones*, com a finalidade de apoiar a disseminação desta tecnologia e a concepção de novos produtos, bem como o desenvolvimento de estratégias de *marketing* para o mercado de telefonia móvel.

Os autores constataram alguns aspectos interessantes, tais como:

- A utilidade é um atributo que influencia diretamente a adoção de aplicativos para *smartphones*.

- A facilidade de uso influencia indiretamente a adoção de aplicativos para *smartphones* e está vinculada com a experiência do usuário.
- O *designer* do *smartphone* não afeta diretamente a facilidade de uso dos aplicativos, entretanto é necessário o desenvolvedor do aplicativo disponibilizar configurações parametrizadas.
- A facilidade de uso não influencia na utilidade dos aplicativos.
- A disponibilidade da rede, serviços de Internet e os aplicativos afetam na utilidade e facilidade de uso.

Com o avanço nas vendas de *smartphones*, principalmente para usuários em uma faixa etária escolar, que abrange o ensino médio, a graduação e a pós-graduação, um novo nicho de mercado conhecido como *Mobile Learning (M-Learning)* está em evidência. Reunindo os conceitos de *e-Learning* e computação móvel, a *m-Learning* traz ao aluno novas perspectivas em relação à utilização de Ambientes Virtuais de Aprendizagem (AVAs) com a liberdade de locomoção.

Vários pesquisadores e desenvolvedores já se dedicam a aliar a tecnologia com a educação por meio de softwares educacionais nos quais o usuário pode acessar suas apostilas, jogos, exercícios, fóruns, *chats*, *e-mails* e vídeo-aulas de maneira integrada em um ambiente *online* que permita aos professores e alunos um grande nível de interação em qualquer lugar e a qualquer hora, o que facilitaria a administração das tarefas diárias dos usuários com pouco tempo para estudar (Mühlbeier, 2012).

Neste cenário, Fetaji et al. (2011) propuseram uma nova metodologia chamada *Mobile Learning Usability Attribute Testing (MLUAT)* para avaliar a usabilidade de aplicativos para *m-learning*. A metodologia MLUAT combina a técnica de inspeção com a pesquisa qualitativa para avaliar a eficácia de ambientes *m-learning* com base em cinco atributos de usabilidade: tempo para aprender, desempenho na aprendizagem, eficiência, taxa de erros e satisfação subjetiva.

Para medir as potencialidades da nova metodologia, os autores desenvolveram um experimento para avaliar o aplicativo *MobileView* (Fetaji e Fetaji, 2008) o qual é um AVA projetado para ser utilizado com *smartphones*.

O experimento foi conduzido com a utilização de três metodologias distintas – MLUAT, Testes Qualitativos e Avaliações Heurísticas – com o objetivo de compará-las em termos dos resultados obtidos.

O número de participantes no experimento variou de acordo com cada metodologia: no MLUAT, foram selecionados cinco usuários, no Teste Qualitativo, foram utilizados vinte usuários e na Avaliação Heurística participaram três especialistas. Os atributos de usabilidade investigados e o número de problemas encontrados com cada metodologia podem ser observados na Tabela 5.

A conclusão que os autores chegaram foi que a metodologia MLUAT detecta mais problemas de usabilidade que as duas outras metodologias utilizadas, sendo assim, mais apropriada para avaliações na área de aplicativos para *m-learning*.

Tabela 5 – Número de problemas de usabilidade encontrados (Fetaji et al., 2011)

Atributos Avaliados	Heurística	Qualitativo	MLUAT
Estado de visualização do aplicativo	5	2	7
Correspondências entre o aplicativo e o mundo real	3	2	3
Controle de aprendizagem	2	1	4
Consistência e fidelidade a padrões	3	2	5
Prevenção e recuperação de erros	2	1	2
Aprendizagem do aplicativo	2	3	5
Flexibilidade e eficiência de uso	2	1	3
Reconhecimento, diagnóstico e recuperação de erros	3	2	5
Designer para telas pequenas	9	11	15

Hegarty e Wusteman (2011) realizaram um trabalho de avaliação de usabilidade na área de aplicativos para acervos bibliográficos, mais especificamente para avaliar o aplicativo EBSCOhost Mobile que disponibiliza acesso ao acervo bibliográfico por meio de *smartphones*.

O experimento de usabilidade realizado pelos autores teve como objetivo medir cinco atributos de usabilidade: capacidade de aprendizado, navegação, eficiência, eficácia e estética. Participaram do teste seis estudantes universitários utilizando um *iPhone*. O experimento foi conduzido em laboratório e contou com a aplicação de questionários e a técnica *Think Aloud*.

Os participantes realizaram quatro tarefas distintas:

- Tarefa 1 – realizar uma pesquisa utilizando um BD específico.
- Tarefa 2 – recuperar um arquivo em um formato HTML.
- Tarefa 3 – encontrar um determinado artigo e enviá-lo para um e-mail.
- Tarefa 4 – utilizar uma das funcionalidades da pesquisa avançada.

Durante a execução da primeira tarefa, apenas um participante teve dificuldades em função de ter escolhido uma das opções do menu equivocadamente. No caso da segunda tarefa, quatro participantes cometeram erros e foi necessário utilizar a opção de voltar para reiniciar a tarefa, o que consequentemente causou demora além do esperado para a conclusão. Na terceira tarefa, três participantes cometeram erros e apesar de considerarem a tarefa fácil, um dos participantes relatou que não achou óbvio o ícone que permitia enviar o e-mail. Finalmente, na execução da quarta tarefa, dois participantes tiveram problemas para selecionar o período da pesquisa e um dos participantes teve dificuldades para escolher o BD desejado.

As conclusões que os pesquisadores chegaram é que o aplicativo permite uma rápida aprendizagem, entretanto, alguns ícones como, por exemplo, para enviar e-mail, não são tão óbvios como deveriam ser. A aplicação foi elogiada em termos de navegabilidade, apesar dos participantes obterem resultados abaixo do esperado com relação à eficiência, já que o tempo médio para concluir as tarefas foi acima do esperado. A eficácia dos participantes na conclusão das tarefas foi regular, já que vários erros

foram constatados. Com relação à estética, os participantes avaliaram positivamente. Em geral, a aplicação foi considerada boa em termos de utilização, apesar da constatação de alguns ajustes que podem ser obtidas com pequenas alterações na interface. Os pesquisadores acreditam que, com as alterações identificadas no experimento, a aplicação possibilitará menor tempo para a conclusão das tarefas e aumentará a proporção de tarefas concluídas com precisão.

O último trabalho descrito nesta seção aborda a criação de um sistema que ajusta as configurações de alguns aplicativos Android em função de variáveis contextuais coletadas dos sensores e informações referentes ao contexto social do usuário (Grønli et al., 2011).

O sistema foi projetado com três componentes principais: um cliente Android, um aplicativo que executa em um servidor na nuvem e os serviços remotos do Google. Os dados que são gerenciados pela aplicação também são armazenados remotamente na nuvem.

Após a instalação do cliente Android no *smartphone*, o dispositivo deverá ser registrado para ter acesso aos serviços do Google, possibilitando que seja disponibilizado o calendário, a agenda de compromissos do usuário e os seus contatos. O próximo passo é fazer o *login* no site disponibilizado para registrar os clientes Android e escolher as aplicações que deseja obter os ajustes automáticos em função das variáveis contextuais.

Para possibilitar a relação entre o contexto social (trabalho, lazer, etc.) e os compromissos agendados e contatos, foram projetadas *tags* especiais que relacionam as informações. Desta forma, é possível identificar se o usuário está no seu ambiente de trabalho ou em uma atividade de lazer. Por exemplo, se o aplicativo associar a agenda do usuário a uma reunião de trabalho, todas as possíveis pessoas envolvidas serão associadas.

Os sensores do *smartphone* também foram utilizados como entrada para a aplicação. O acelerômetro foi usado para registrar se o usuário estava em movimento e, neste caso, a aplicação é apresentada com uma interface mais simples com botões em escala aumentada. Já o sensor de luminosidade, permite que a cor de fundo da aplicação seja alterada em função da intensidade da luz no ambiente.

O experimento contou com a participação de 38 usuários, os quais foram submetidos à execução de 13 tarefas como, por exemplo:

- Selecionar a aba de contatos e verificar se estão relacionadas às pessoas envolvidas com a próxima reunião.
- Selecionar a aba de sensores e verificar as mudanças na aplicação quando o *smartphone* é movimentado ou exposto à luz.
- Voltar o *smartphone* para o estado inicial e verificar se após 15 segundos o *layout* dos aplicativos volta ao estado normal.

- Entrar na agenda do Google, criar um novo compromisso e verificar se os contatos exibidos serão atualizados.

No final, os participantes responderam a 17 questões baseadas na escala de Likert (1932) com quatro possibilidades de resposta. Os dados coletados foram analisados com o método estatístico *OneSample T-Test*.

Quanto à avaliação da eficácia, os resultados observados foram bons na maioria das funcionalidades avaliadas, entretanto as mudanças de configuração quando o *smartphone* sofre movimentos nem sempre foi consistente, o que gerou muitas divergências nas pontuações atribuídas a este quesito.

Outro atributo que apresentou divergências foi o nível de compreensão do conteúdo apresentado e dificuldade de utilizar alguns recursos devido a mudanças abruptas no *layout* da tela. Os autores acreditam que este resultado mudaria com o passar do tempo, à medida que as pessoas compreendessem as mudanças de configuração.

Com relação à satisfação dos usuários, foram investigados três pontos chaves: (i) a utilização da computação na nuvem; (ii) a integração com aplicativos externos, neste caso específico, aplicativos disponibilizados pela Google; e (iii) a disposição do usuário para novas experiências.

Para o primeiro item (computação na nuvem), foram observadas 11 respostas que desaprovam a utilização deste recurso em função da desconfiança com relação à segurança e exposição de dados pessoais. Nos demais itens, foi possível observar que os participantes gostaram da integração com aplicativos externos e estavam propensos a participar de novas experiências com o sistema proposto.

2.3.5 Trabalhos publicados em 2012

Após a pesquisa da bibliografia referente a 2012, foi possível constatar que os aplicativos para *smartphones* abrangeram novas áreas e passaram a requerer estudos mais intensos de usabilidade com a finalidade de serem adotados no dia a dia das pessoas. Um exemplo disso é o primeiro trabalho descrito nesta seção.

Sparkes et al. (2012) avaliaram um sistema manuseado pelo próprio paciente para o acompanhamento da sua frequência cardíaca. Segundo os autores, este tipo de procedimento não é invasivo, não constrange o paciente e torna-se mais prático e barato. A sua eficiência, entretanto, está diretamente relacionada com o manejo apropriado do aparelho que mede a frequência cardíaca, bem como, do aplicativo

instalado no *smartphone* que recebe as informações. O paciente, neste caso, precisa saber configurar o aplicativo e transmitir as informações corretamente.

Para o experimento, foram recrutados cinco pacientes com arritmia cardíaca, idade acima de 46 anos, diferentes familiarizações com aparelhos tecnológicos e alfabetizados na língua inglesa. Para os pacientes foi entregue o *Event Loop Recorder* (ELR) responsável em fazer as medições da frequência cardíaca e um *smartphone BlackBerry* contendo o aplicativo para capturar os dados e possibilitar as transmissões.

As técnicas de captura de dados utilizadas no experimento contemplaram: (i) anotações do pesquisador; (ii) uma câmera para gravar a execução da técnica *Think Aloud* (pensar em voz alta); e (iii) uma entrevista após a conclusão da tarefa de configuração do aplicativo. Infelizmente, não foi relatada a pontuação dos atributos investigados de usabilidade no artigo. Entretanto, na Tabela 6, são apresentados alguns itens observados na pesquisa.

Tabela 6 – Resumo dos dados coletados no experimento (Sparkes et al., 2012)

Itens avaliados	Part. 1	Part. 2	Part. 3	Part. 4	Part. 5
Tempo para realizar a tarefa.	17:45	8:50	11:50	21:00	17:45
Número de intervenções do pesquisador.	1	0	1	5	5
Não leu as instruções até o final.		X	X		
Problemas com o botão de início de gravação (configuração de horário e local).	X	X	X	X	X
Ansiedade relacionada ao <i>smartphone</i> .	X	X	X	X	X
Problemas com o <i>smartphone</i> .				X	
Problemas com a terminologia usada.	X			X	

Outra aplicação na área médica, chamada de *S-T Segment Elevation Myocardial Infarctions* (STEMI), foi proposta por Bradley et al. (2012). Ela é embarcada em *smartphones* usados em emergências, onde o paciente atendido apresenta sintomas de infarto do miocárdio. Basicamente, a aplicação recolhe os dados de um aparelho de eletrocardiograma (ECG) e transfere as informações para um hospital onde os especialistas podem diagnosticar a lesão sofrida pelo paciente. Com base nos dados do exame, os médicos providenciam os recursos necessários para atender o paciente imediatamente a sua chegada ao pronto socorro do hospital.

Os pesquisadores que criaram o aplicativo realizaram um teste de usabilidade onde os participantes executam livremente as tarefas necessárias para capturar as informações do ECG e posteriormente transmiti-las para um servidor disponibilizado pelo hospital. Após a utilização do aplicativo, os participantes responderam a um questionário que avalia alguns aspectos de usabilidade.

Os resultados obtidos no primeiro experimento apontaram vários problemas de usabilidade, que serviram de base para várias correções na interface. Após as correções, um novo experimento foi realizado, no qual foi constatada uma melhora significativa em termos de eficiência, eficácia e facilidade de uso.

Para avaliar a usabilidade dos teclados disponibilizados pelos principais fabricantes de *smartphones*, Schaub et al. (2012) criaram um experimento conduzido em laboratório onde a tarefa dos participantes foi inserir senhas para desbloquear o aparelho ou obter acesso a algum tipo de aplicativo ou site. O experimento foi conduzido com 80 participantes e investigou as seguintes plataformas: iPhone OS, Android, Windows Phone, Symbian, MeeGo.

As principais métricas quantitativas utilizadas para medir a usabilidade foi o tempo médio para a inserção completa de uma senha e o número de erros cometidos pelo participante.

Além da usabilidade, os autores investigaram quais eram os padrões de senhas mais seguras. Para isso, foram realizados testes onde os participantes inseriam suas senhas sob o olhar de um observador com o objetivo de avaliar quantos caracteres o intruso poderia identificar. Cada participante inseria cinco senhas com padrões de caracteres diferentes.

Os resultados com relação ao tempo médio para a inserção das senhas apontaram que os participantes conseguiram melhor desempenho na plataforma iPhone OS e foram, em média, mais lentos na plataforma Symbian.

Com relação à taxa de erros, o Symbian com o padrão de teclado QWERTZ foi o *smartphone* que propiciou o maior número de erros, por outro lado, o Symbian com o padrão de teclado T9 teve uma taxa de erros muito baixa, sendo superado apenas pelo Windows Phone.

Outro ponto a ser destacado nos resultados relatados é que os dispositivos com pior desempenho em termos de usabilidade se destacam como sendo os mais seguros. Segundo os autores, este resultado está diretamente relacionado com a quantidade de ações necessárias para que o usuário possa mudar de opção de entrada para inserir caracteres especiais, tais como, variação entre letras maiúsculas e minúsculas, caracteres de pontuação, números e caracteres especiais muito frequentes em senhas.

Kirwan et al. (2012) criaram um aplicativo para *smartphones* chamado *iStepLog*, com a finalidade de aumentar a retenção de usuários que se engajam em programas de atividade física para melhorar a qualidade de vida e combater a obesidade.

Os objetivos do estudo foram: (i) analisar o efeito da aplicação para o monitoramento e relatos dos níveis de atividade física dos participantes; (ii) medir a utilidade e usabilidade da aplicação; e (iii) examinar a relação entre a utilidade, facilidade de uso e utilização efetiva do aplicativo.

Para a execução do experimento foram recrutadas, por e-mail, 50 pessoas adeptas do programa para acompanhamento físico chamado 10.000 Passos. Os requisitos para o recrutamento dos participantes eram: ter um *iPhone* e pertencer ao programa há mais de três meses com assiduidade. Os participantes foram

instruídos a instalar o aplicativo no seu *smartphone* e utilizá-lo por três meses. A tarefa que eles tinham que executar era registrar as suas atividades físicas no aplicativo, da mesma forma que procediam quando utilizavam o website do programa. Após o estudo, os participantes responderam um questionário *on-line* com a finalidade de avaliar a usabilidade e utilidade do aplicativo.

As perguntas e a pontuação atribuída ao aplicativo podem ser contempladas na Tabela 7 onde é possível observar que tanto a usabilidade como a utilidade do aplicativo foram aprovadas pela maioria dos participantes do experimento.

Para comparar a utilização do aplicativo com o uso do Web *site*, os pesquisadores selecionaram 150 pessoas que só forneciam os dados de suas atividades físicas por meio do Web *site*. O objetivo dessa investigação paralela foi avaliar a frequência e o número de informações fornecidas em ambas às formas disponibilizadas pelo programa. Após o término do experimento, foram coletados os dados estatísticos da utilização do Web *site* e do aplicativo. Os resultados obtidos com esta avaliação é que, em média, os usuários do aplicativo *iStepLog* informaram seus dados 40 vezes durante o período do experimento, com uma média de utilização de 71% contra 29% dos usuários do Web *site*.

Tabela 7 – Avaliação da usabilidade do aplicativo *iStepLog* (Kirwan et al., 2012)

Itens avaliados	Classificação Média	Concorda ou Fortemente Concorda
1) Eu acho que o aplicativo <i>iStepLog</i> é amigável.	4.45 (0.66)	91% (40/44)
2) Eu gosto da apresentação e <i>layout</i> do aplicativo <i>iStepLog</i> .	4.24 (0.66)	89% (39/44)
3) Eu facilmente encontrava as opções desejadas no aplicativo <i>iStepLog</i> .	4.48 (0.59)	95% (42/44)
4) Eu facilmente inseria e editava os meus dados na aplicação.	4.39 (0.65)	91% (40/44)
5) Eu tinha facilidade de sincronizar os meus dados com o servidor do programa 10.000 passos.	4.21 (0.81)	80% (35/44)
Avaliação correspondente à usabilidade	4.35 (0.67)	89% (39/44)
6) Era conveniente usar o aplicativo <i>iStepLog</i> .	4.50 (0.70)	89% (39/44)
7) Prefiro usar o aplicativo <i>iStepLog</i> em vez do website.	4.35 (0.75)	84% (37/44)
8) O <i>iStepLog</i> me incentivou a fazer com mais frequência atividade física.	4.00 (0.83)	80% (35/44)
9) Eu não visitei o website com mais frequência porque eu usei o aplicativo <i>iStepLog</i> .	4.33 (0.76)	84% (37/44)
10)Eu gostaria de continuar usando o aplicativo <i>iStepLog</i> .	4.47 (0.70)	89% (39/44)
Avaliação correspondente à utilidade	4.33 (0.75)	84% (36.8/44)

Os últimos trabalhos analisados são os propostos por Spyridonis et al. (2012, 2013). Ambos os trabalhos tratam de um aplicativo para a plataforma Android, chamado *PainDroid*, que permite aos pacientes com quadros dolorosos mapearem os locais afetados, possibilitando a criação de um diário que posteriormente poderá ser avaliado em uma consulta médica.

O *PainDroid* é um aplicativo multimodal que disponibiliza uma imagem tridimensional do corpo humano (manequim) onde o usuário pode indicar o local da dor. As interações podem ocorrer por meio de toques na tela, movimentos do aparelho que interferem na posição da imagem ou, ainda, por meio de óculos com a tecnologia *Head Mounted Display* (HMD) que permite a interação de usuários com limitações de movimento.

Dois sensores são utilizados para o reconhecimento da movimentação do dispositivo e, conseqüentemente, das rotações da imagem. Quando o usuário inclina o dispositivo para cima, para baixo, esquerda ou direita, o acelerômetro registra essas mudanças e altera a interface de acordo com a direção indicada. Os movimentos rotacionais para a esquerda e para a direita são registrados usando o magnetômetro.

Para realizar as avaliações de usabilidade, foi proposto um experimento com a participação de 7 pacientes de clínicas reumáticas, sendo que dois desses pacientes apresentavam limitações de movimento e utilizaram o óculos HMD para realizarem suas interações. O experimento foi conduzido em laboratório onde os participantes tinham que realizar 10 tarefas triviais para o uso do aplicativo como, por exemplo, rotacionar a imagem do corpo humano para apresentá-la em diferentes ângulos, indicar o local e o tipo da dor.

Após o término das tarefas, os participantes responderam a um questionário com 13 perguntas formatadas de acordo com a escala de Likert (1932) com cinco respostas possíveis. Além disso, participaram de uma entrevista com duração aproximada de 20 minutos.

O primeiro atributo de usabilidade investigado foi a facilidade de aprender e utilizar o aplicativo, sendo constatado que a maioria dos avaliadores achou o sistema fácil de aprender e desnecessário o apoio de um técnico para utilizar o aplicativo.

O segundo atributo avaliado foi relacionado à eficácia que foca a integridade e precisão com que os usuários atingem determinados objetivos. Após as análises, foi observado que a maioria dos usuários encontrou dificuldade para girar o manequim e deixá-lo na posição desejada. Este fato foi atribuído à sensibilidade dos sensores utilizados para a detecção dos movimentos.

No atributo relacionado à compreensão, foi detectado que os participantes conseguiram, com facilidade, compreender o conteúdo e as funcionalidades disponibilizadas no aplicativo.

Finalmente, na avaliação referente à satisfação do usuário, foi constatado que alguns avaliadores gostariam de outra imagem do manequim e um *designer* mais rebuscado no menu referente aos tipos de dores. Quando os avaliadores foram perguntados se usariam o aplicativo novamente, a maioria indicou uma resposta positiva e confirmaram a utilidade do aplicativo. Entretanto, as respostas foram muito variadas com relação à utilização de movimentos como forma de interação.

2.3.6 Discussão sobre os trabalhos relatados

Esta subseção traz a análise dos dados provenientes dos testes de usabilidade descritos nas seções anteriores, abrangendo os trabalhos publicados entre os anos de 2008 a 2012. Para facilitar a sumarização, serão utilizadas algumas tabelas que condensam as principais informações a serem analisadas, tais como, os aspectos metodológicos, características dos experimentos, atributos de usabilidade utilizados e aspectos contextuais. Como referência para as discussões, serão tomados como base o *framework* proposto por Coursaris e Kim (2011) (seção 2.2) e os atributos de usabilidade considerados no escopo desta tese, definidos na Tabela 1.

A Tabela 8 apresenta o resumo das características das avaliações descritas nas subseções 2.4.1 a 2.4.5, incluindo os autores, as técnicas utilizadas, o número de participantes, o tipo de aplicativo avaliado e o assunto investigado nos experimentos. São relacionados vinte e um trabalhos, sendo a maioria de autores diferentes, o que indica o interesse recente pela área de avaliação de usabilidade.

Tabela 8 – Resumo das características dos experimentos

Autores	Técnicas utilizadas	Nº participantes	Tipo de aplicativo	Assunto investigado
Burigat et al. (2008)	<i>Logging</i> e questionário	20	Visualização de mapas e sites Web (<i>browser</i>)	Usabilidade de técnicas de navegação
Sodnik et al. (2008)	Observação direta, entrevistas e dados do dispositivo (<i>Logging</i>)	18	Aplicativos básicos de <i>smartphones</i> e simulador de carro	Usabilidade de interface visual X Interface auditiva
Fitchett e Cockburn (2009)	Observação direta e entrevistas	14	Visualização de texto e tarefas	Usabilidade das técnicas de rolagem de tela
Chin e Salomaa (2009)	<i>Logging</i> em servidores Web e questionários	180	Aplicativos Web relacionados à Olimpíada de 2008	Usabilidade, preferência por aplicativos e experiência do usuário
Lai et al. (2009)	Questionário <i>on-line</i> e entrevistas cognitivas	11	Aplicação para responder questões do cotidiano	Usabilidade, estilo de vida e personalidade
Ebner et al. (2009)	<i>Thinking Aloud</i> e Questionário	17	Aplicativo para gerenciar o aprendizado (<i>m-learning</i>)	Usabilidade e experiência do usuário
Hansen e Ghinea (2009)	Questionário	16	Aplicativo para controlar remotamente um <i>desktop</i>	Usabilidade
Bødker et al. (2009)	Questionários, grupos focais e entrevistas individualizadas	16	Aplicativos básicos existentes em todos os <i>smartphones</i>	Usabilidade, modelo de negócio para <i>smartphones</i> e UX

Autores	Técnicas utilizadas	Nº participantes	Tipo de aplicativo	Assunto investigado
Kim et al. (2010)	Questionário	33	Aplicativo para navegação Web (<i>browser</i>)	Usabilidade das estruturas de navegação
Li e Yeh (2010)	Questionário	200	Aplicativo para comércio eletrônico móvel (<i>m-commerce</i>)	Usabilidade e confiabilidade
Maly et al. (2010)	<i>Logging, Thinking Aloud</i> (gravadores de áudio) e observação direta (câmeras de vídeo e anotações)	12	Aplicativo para auxílio a locomoção de deficientes visuais (<i>m-health</i>)	Usabilidade e sensibilidade ao contexto
Grønli et al. (2010)	Questionário	40	Aplicativo para compartilhar recursos em sala de reuniões	Usabilidade e computação na nuvem
Kang et al. (2011)	Questionários e análises baseadas na TAM e SEM	100 e 262	Aplicativos básicos existentes em todos os <i>smartphones</i>	Usabilidade e aspectos que influenciam na adoção de aplicativos para <i>smartphones</i>
Fetaji et al. (2011)	MLUAT, Testes Qualitativos (Questionários) e Avaliações Heurísticas	MLUAT = 5 Qualitativo = 20 Heurística = 3	Aplicativo para EAD (<i>m-learning</i>)	Usabilidade e comparação de técnicas de avaliação na área de IHC
Hegarty e Wusteman (2011)	Questionários e <i>Think Aloud</i>	6	Aplicativo para realizar pesquisas bibliográficas	Usabilidade
Grønli et al. (2011)	Questionário	38	Aplicativos sensíveis ao contexto	Usabilidade, variáveis contextuais e experiência do usuário
Sparkes (2012)	Anotações, <i>Think Aloud</i> e entrevista	5	Aplicativo para acompanhamento da frequência cardíaca (<i>m-health</i>)	Usabilidade
Bradley et al. (2012)	Questionários	Não especificado	Aplicativo para enviar dados de pacientes infartos (<i>m-health</i>)	Usabilidade
Schaub et al. (2012)	<i>Logging</i> e observação direta	80	Aplicativos que requerem senha de acesso	Usabilidade e segurança de acesso
Kirwan et al. (2012)	<i>Logging</i> e Questionário <i>on-line</i>	Smartphone = 50 Website = 150	Aplicativo para relatar atividades físicas executadas (<i>m-health</i>)	Usabilidade
Spyridonis et al. (2012)	Questionário e entrevista	7	Aplicativo para relatar locais do corpo com dor (<i>m-health</i>)	Usabilidade, experiência do usuário e sensibilidade ao contexto

Com relação às técnicas utilizadas para a coleta de dados nos experimentos, pode ser observado que 15 utilizam questionários, 4 *logging*, 3 observações diretas dos avaliadores, 3 entrevistas com os usuários, 4 a técnica *Think Aloud* e 6 outras técnicas menos tradicionais. A Figura 2 apresenta a distribuição percentual das técnicas, sendo possível observar que a grande maioria dos experimentos utiliza questionários como base para a coleta de dados.

Segundo Courage e Baxter (2005), os questionários representam uma das técnicas mais comuns de obtenção de dados em experimentos de usabilidade, tendo como vantagens a possibilidade de envolver uma grande quantidade de dados e usuários, permitir uma análise rápida e fácil dos resultados, além de ser uma técnica barata financeiramente quando comparada a outras com o mesmo objetivo. Por outro lado, Zhang e Adipat (2005) afirmam que as avaliações envolvendo dispositivos móveis devem contemplar fatores contextuais na coleta dos dados. Isso nem sempre é considerado com a utilização de questionários onde, na maioria das vezes, são formatados com questões fechadas e respostas previamente definidas. Normalmente, as questões de usabilidade estão atreladas ao aplicativo, não incorporando informações do cenário de interação, das relações sociais e do estado emocional do usuário.

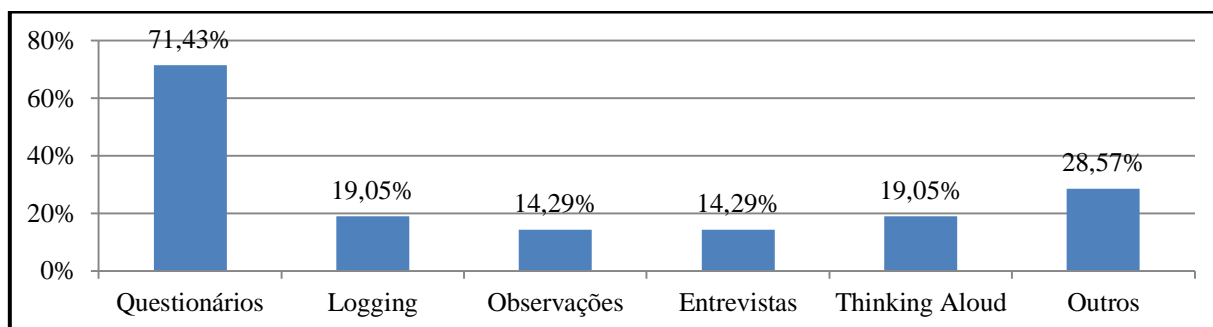


Figura 2 – Distribuição percentual das técnicas de avaliação utilizadas entre 2008 e 2012

Com relação ao número de participantes nos experimentos, é possível perceber que todas as avaliações relatadas contemplam pelo menos o número descrito na afirmação de Nielsen e Mack (1994): “cinco participantes em uma avaliação de usabilidade são suficientes para detectar 95% dos problemas em uma aplicação”.

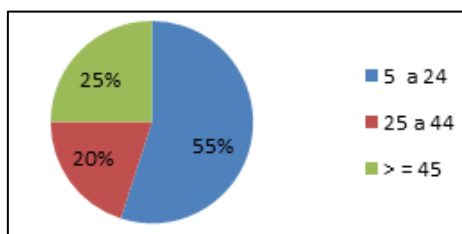


Figura 3 – Distribuição percentual de participantes nos experimentos

Para melhorar a análise dos dados, o número de participantes foi dividido em três faixas distintas: onze experimentos utilizaram entre 5 a 24 participantes, quatro utilizaram entre 25 e 44 participantes e cinco avaliações foram realizadas com um número acima de 44 usuários. Na Figura 3, é possível ver esta distribuição percentualmente, sendo perceptíveis que mais da metade dos estudos foram conduzidos com até 24 participantes.

Com relação aos tipos de aplicativos utilizados nos experimentos, pode ser observado que eles são bastante variados, abrangendo várias áreas, com especial destaque para a área de saúde (*m-health*) com seis aplicativos, totalizando 28,57%.

Os assuntos investigados englobam vários aspectos, tais como técnicas de navegação, tipos de interações, técnicas de rolagem de tela, preferências por aplicativos, experiência do usuário, segurança de acesso, modelos de negócios, entre outros. Entretanto, é importante salientar que todos os experimentos contemplam avaliações de usabilidade, o pré-requisito mais importante para a inclusão do trabalho no estudo proposto neste capítulo.

Na Tabela 9, são apresentados os atributos de usabilidade considerados nos experimentos, levando em consideração a unificação e descrição dos nove atributos definidos na Tabela 1.

Tabela 9 – Dimensão de Usabilidade

Autores	Atributos de usabilidade considerados nas avaliações
Burigat et al. (2008)	Eficiência, eficácia, satisfação e facilidade de uso.
Sodnik et al. (2008)	Eficiência, eficácia, satisfação, flexibilidade, facilidade de uso.
Fitchett e Cockburn (2009)	Eficiência, eficácia, satisfação e facilidade de uso.
Chin e Salomaa (2009)	Eficiência, eficácia, satisfação, utilidade e facilidade de uso.
Lai et al. (2009)	Eficiência, eficácia, satisfação, operabilidade, flexibilidade e facilidade de uso.
Ebner et al. (2009)	Eficiência, eficácia, satisfação e facilidade de uso.
Hansen e Ghinea (2009)	Eficácia, satisfação, aprendizagem e facilidade de uso.
Bødker et al. (2009)	Eficiência, satisfação, aprendizagem, flexibilidade e facilidade de uso.
Kim et al. (2010)	Satisfação, utilidade e facilidade de uso.
Li e Yeh (2010)	Satisfação, utilidade e facilidade de uso.
Maly et al. (2010)	Eficácia, satisfação, acessibilidade, utilidade e facilidade de uso.
Grønli et al. (2010)	Eficácia, satisfação, aprendizagem e facilidade de uso.
Kang et al. (2011)	Satisfação, utilidade e facilidade de uso.
Fetaji et al. (2011)	Eficiência, eficácia, satisfação, aprendizagem, operabilidade, utilidade e facilidade de uso.
Hegarty; Wusteman (2011)	Eficiência, eficácia, satisfação, aprendizagem e facilidade de uso.
Grønli et al. (2011)	Eficácia, satisfação, operabilidade, flexibilidade e facilidade de uso.
Sparkes et al. (2012)	Eficiência, eficácia, satisfação, aprendizagem, acessibilidade e facilidade de uso.
Bradley et al. (2012)	Eficiência, eficácia e facilidade de uso.
Schaub et al. (2012)	Eficiência, eficácia e facilidade de uso.
Kirwan et al. (2012)	Satisfação, utilidade e facilidade de uso.
Spyridonis et al. (2012)	Eficácia, satisfação, aprendizagem, acessibilidade, flexibilidade e facilidade de uso.

Para facilitar uma avaliação quantitativa dos atributos de usabilidade utilizados nos experimentos analisados, a Figura 4 apresenta um gráfico com o número absoluto de vezes que cada um dos atributos da Tabela 1 foi contemplado nos estudos. Observando os dados apresentados no gráfico, é possível perceber que os atributos eficiência, eficácia e satisfação, definidos na ISO 9241-11 (1998), são avaliados em mais de 50% dos casos. Além disso, é possível destacar que o atributo facilidade de uso é contemplado em 100% das análises, confirmando a importância de medir o nível de entendimento por parte dos usuários de como executar uma tarefa em avaliações de usabilidade para *smartphones* (Coursaris e Kim, 2011).

Os atributos operabilidade e acessibilidade foram os menos investigados, o que pode ser até certo ponto surpreendente, já que são dois atributos muito importantes no contexto da usabilidade para dispositivos móveis.

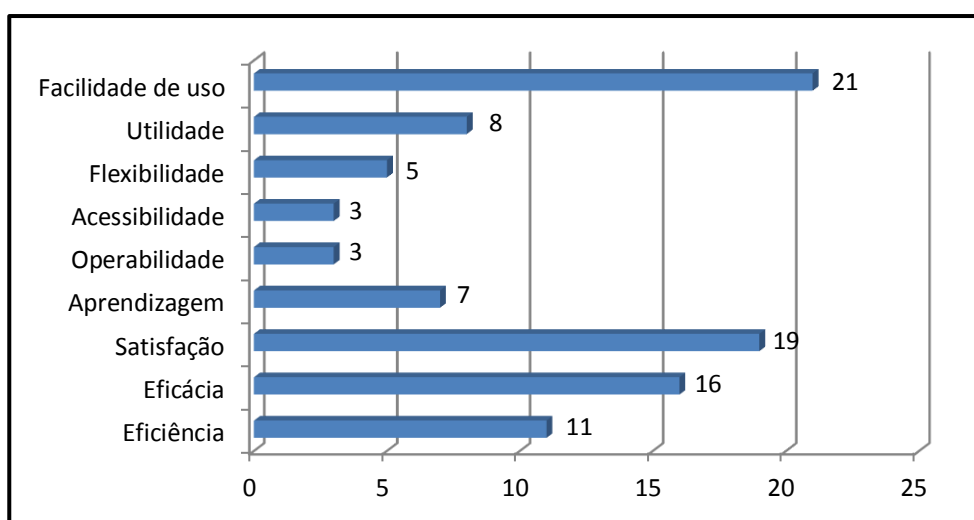


Figura 4 – Quantidade absoluta de vezes que cada atributo de usabilidade foi avaliado

A operabilidade de aplicativos para *smartphones* é uma área de estudo normalmente vinculada a dois fatores: (i) a disponibilidade e velocidade das redes de comunicação que disponibilizam serviços para a tecnologia móvel; e (ii) a capacidade de processamento dos *smartphones*. No trabalho de Betiol e Cybis (2005), foi observado que, a priori, os participantes dos testes de usabilidade acreditam que uma conexão instável e lenta é mais prejudicial para a interatividade com um aplicativo móvel do que o tamanho reduzido das interfaces.

A acessibilidade no âmbito de dispositivos móveis é uma área com forte apelo social, com muitos pesquisadores se dedicando ao assunto. No trabalho de Yesilada et al. (2011) são investigadas as barreiras de acessibilidade comuns entre usuários de dispositivos móveis e usuários com deficiência visual, pouca visão e deficiência motora. O estudo foi realizado com a utilização do método Percurso de Barreiras (Brajnik, 2006) que permite teste e agregação de barreiras entre os grupos selecionados. A motivação do estudo ocorreu devido ao fato de que estudos anteriores propostos por Trewin (2006) e Wobbrock (2006)

indicam que os usuários sem nenhum tipo de deficiência, ao acessarem dispositivos móveis, encontram barreiras semelhantes às encontradas pelos usuários com alguma deficiência acessando a web através de *desktops*. Outro trabalho nesta área, proposto por Chen (2008), investiga problemas comuns entre usuários que acessam a web via dispositivos móveis e usuários com alguma deficiência.

Considerando dispositivos com tecnologias específicas como as telas sensíveis ao toque, Piccolo et al. (2011) realizaram uma pesquisa a fim de identificar problemas de acessibilidade de usuários com deficiência visual, total ou parcial, ao utilizarem leitores de tela em *smartphones*. Como resultado deste estudo, o trabalho define um conjunto de diretrizes a ser considerado no *design* de soluções baseadas em leitores de tela e telas sensíveis ao toque. Trabalho semelhante também é proposto por McGookin et al. (2008) que levantaram requisitos de acessibilidade em dispositivos com tecnologia *touchscreen* para deficientes visuais e fornece diretrizes para *designers*. Porém, as diretrizes são mais abrangentes, direcionadas a todos os dispositivos com tela sensível ao toque, de maneira a atender as necessidades do público com esta deficiência.

Segundo Sakamoto et al. (2012), diversos trabalhos abordam problemas de acessibilidade no uso de dispositivos móveis, a maioria considerando públicos com determinados tipos de deficiência, principalmente visual. Porém, a maioria desses trabalhos não investigam requisitos de acessibilidade com pessoas que não possuem deficiência, mas que também são afetados pela ausência da acessibilidade em seus domínios. Portanto, também se faz necessário mostrar os problemas enfrentados por esse público, como superá-los e os benefícios trazidos pela construção de interfaces mais acessíveis.

A Tabela 10 retrata as variáveis contextuais especificadas no *framework* proposto por Coursaris e Kim (2011). O atributo Usuário corresponde a principal característica dos participantes do experimento enfatizada pelos autores. O atributo Tarefa/Atividade refere-se à atividade desenvolvida pelo usuário ao utilizar o aplicativo. O atributo Ambiente é utilizado para descrever onde o experimento ocorreu (laboratório, em campo ou simulador) e quais variáveis contextuais foram consideradas, como o movimento do aparelho (horizontal vertical), velocidade de deslocamento, luminosidade, barulho, atenção, entre outros. Finalmente, o atributo Tecnologia diz respeito ao sistema operacional utilizado no *smartphone*.

É possível observar nos dados informados na Tabela 10 que as variáveis contextuais Usuário e Tarefa/Atividade apresentam uma grande diversidade. Quanto aos usuários, pode ser destacado que 47,61% dos experimentos foram conduzidos com estudantes. Acredita-se que este público é conveniente para os pesquisadores em função da facilidade de acesso, já que são pessoas integralizadas ao ambiente onde os estudos são desenvolvidos. Com relação às tarefas e atividades, é possível observar que estão relacionadas ao tipo de aplicação que está sendo avaliada.

Tabela 10 – Variáveis Contextuais

Autor	Usuário	Tarefa/Atividade	Ambiente	Tecnologia
Burigat et al. (2008)	Universitários	Navegar em mapas e Web sites	Laboratório	Windows Mobile
Sodnik et al. (2008)	Motoristas	Escrever mensagens de texto, realizar chamadas, alterar configurações, apagar imagens e escutar músicas enquanto dirige.	Laboratório Considera a distração	Symbian OS
Fitchett e Cockburn (2009)	Estudantes de Pós-Graduação na área de Computação	Leitura de texto e contagens de pontos na tela	Laboratório Considera o deslocamento	iPhone OS
Chin e Salomaa (2009)	Espectadores das Olimpíadas de 2008	Navegar em aplicativos Web	Campo	iPhone OS, Symbian OS e Android
Lai et al. (2009)	Participantes remunerados	Responder questionários <i>on-line</i>	Campo	Não especificado
Ebner et al. (2009)	Estudantes Universitários	Interagir com um aplicativo para EAD	Laboratório	iPhone OS e Symbian OS
Hansen e Ghinea (2009)	Funcionários de uma empresa da área tecnológica	Instalar e configurar um aplicativo para controlar remotamente um <i>desktop</i>	Laboratório	Windows Mobile e Java ME
Bødker et al. (2009)	Dinamarqueses	Acessar a Internet, trocar e-mails, enviar SMS, utilizar a câmera, GPS e MP3	Campo	iPhone OS
Kim et al. (2010)	A maioria é universitária em nível de graduação	Procurar fotos e vídeos	Laboratório	Não especificado
Li e Yeh (2010)	Universitários de Taiwan	Comprar uma câmera, alugar um carro e reservar um hotel	Simulador	Não especificado
Maly et al. (2010)	Deficientes visuais	Percorrer um prédio guiado por um aplicativo	Campo	Symbian OS
Grønli et al. (2010)	Pessoas com experiência prévia na utilização de <i>smartphones</i>	Participar de uma reunião e compartilhar informações através do aplicativo	Laboratório	Java ME, Windows Mobile, Android e iPhone OS
Kang et al. (2011)	Universitários coreanos	Atividades corriqueiras desempenhadas em <i>smartphones</i>	Campo	Não especificados
Fetaji et al. (2011)	Estudantes de graduação e especialistas de IHC	Atividades disponibilizadas por Ambientes Virtuais de Aprendizagem	Campo e Laboratório	iPhone OS, Symbian OS, Android e BlackBerry OS
Hegarty e Wusteman (2011)	Universitários	Realizar pesquisas bibliográficas	Laboratório	iPhone OS
Grønli et al. (2011)	Estudantes e profissionais da área de computação	Testar a adaptabilidade de um sistema sensível ao contexto	Laboratório Considera o deslocamento, luminosidade e o contexto social	Android

Autor	Usuário	Tarefa/Atividade	Ambiente	Tecnologia
Sparkes et al. (2012)	Pacientes com arritmia cardíaca	Configurar um aplicativo para coletar a frequência cardíaca e transmitir os dados.	Laboratório e campo (casa do paciente)	BlackBerry OS
Bradley et al. (2012)	Médicos	Obter dados de um aparelho ECG e transmiti-los para um servidor	Campo	iPhone OS
Schaub et al. (2012)	Universitários	Informar senhas de acesso	Laboratório	iPhone OS, Symbian OS, Android, Windows Mobile e MeeGo.
Kirwan et al. (2012)	Usuários de um programa de controle de atividade física	Relatar sua atividade física diária.	Campo	iPhone OS
Spyridonis et al. (2012)	Pacientes de clínicas médicas	Relatar locais de dor no corpo	Laboratório	Android

A variável Ambiente define onde o experimento é realizado e quais as variáveis externas ao aplicativo são consideradas nas análises. Para embasar as discussões relacionadas a este assunto, são apresentados os valores percentuais dos locais onde os experimentos foram conduzidos (Figura 5). Em termos de valores absolutos, foram realizados 11 experimentos em laboratório, 7 em campo, 2 em campo e laboratório e 1 utilizando simuladores, o que demonstra nitidamente a preferência por avaliações em laboratório.

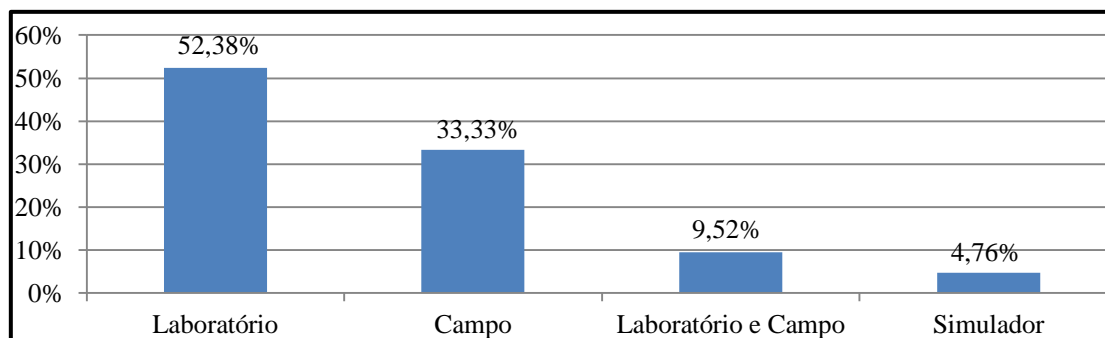


Figura 5 – Percentuais relacionados ao local onde os experimentos foram realizados

As discussões relacionadas com o local onde os experimentos devem ser realizados são recorrentes e abordadas em inúmeros trabalhos na comunidade de IHC (Stienmetz et al., 2012). Os principais argumentos a favor de avaliações em laboratório fundamentam-se no alto grau de controle e relativa facilidade para coletar dados. Além disso, testes de usabilidade em campo requerem, em determinadas situações, muitas práticas e uma logística complexa para serem realizados (Betiol e Cybis, 2005).

Segundo Jensen e Larsen (2007), experimentos em campo são mais difíceis e caros com relação aos recursos financeiros, de tempo e mão de obra. Dessa forma, apenas uma fração relativamente pequena de pesquisas na área de dispositivos móveis toma como base experimentos em campo. Esta afirmação foi

comprovada nos estudos realizados por Kjeldskov e Graham (2003), Coursaris e Kim (2011) e nas análises apresentadas nesta tese.

Apesar da maior frequência de estudos em laboratório, a realização de avaliações em campo trazem vantagens referentes à integração com o ambiente natural de uso dos aplicativos nos experimentos realizados, possibilitando a caracterização real do cenário envolvido nas investigações (Stienmetz et al., 2012). De acordo com Kjeldskov e Stage (2004), é possível simular o contexto da interação que ocorrerá em laboratório, entretanto realizar essa contextualização nem sempre é trivial e o nível de realismo de tais simulações é, em muitos casos, questionável. Além disso, experimentos em laboratório normalmente não são realizados com a duração desejada para que o usuário tenha tempo suficiente para adquirir experiência com relação ao manuseio do aplicativo e o incorpore na sua rotina diária. Esse fato pode ocasionar a impossibilidade de se realizar investigações que necessitem de tempo para que o usuário consiga se habituar às tarefas que necessita realizar. Estas questões, muitas vezes, só podem ser investigadas em ensaios de campo de longa duração com a participação dos usuários finais em seu ambiente natural.

Jensen e Larsen (2007) defendem que experimentos em laboratório, na maioria das vezes, serão susceptíveis a erros devido ao fato de normalmente não serem contempladas todas as variáveis que podem influenciar a avaliação. Há perguntas que permitem grandes variações em experimentos de campo como, por exemplo: (i) Quando o usuário interagiu? (ii) Onde ocorreu a interação? (iii) Por quanto tempo? (iv) Com que frequência? (v) Em que cenário? Dificilmente estas questões podem sofrer variações significativas quando o experimento é realizado em laboratório.

Diante das questões levantadas, Jensen e Larsen (2007) acreditam que os ensaios em campo podem complementar e contribuir para uma maior compreensão do comportamento do usuário e suas preferências. Além disso, afirmam que as investigações em campo por longos períodos de tempo podem resultar em coletas de informações mais precisas, o que não pode ser obtido através de outros métodos.

Outro ponto a ser destacado nos dados apresentados na Tabela 10 é com relação aos dados contextuais analisados. Pode ser observado que dos 21 experimentos relacionados, apenas os trabalhos de Sodnik et al. (2008), Fitchett e Cockburn (2009) e Grønli et al. (2011), analisam dados externos ao aplicativo, o que corresponde a um percentual de 14,28%.

Esta informação vai de encontro aos requisitos definidos para avaliações de aplicativos para dispositivos móveis defendidas por inúmeros pesquisadores na área de IHC (Zhang e Adipat, 2005), (Jensen e Larsen, 2007), (Coursaris e Kim, 2011), (Hansen, 2012) entre outros. Segundo eles, levar em consideração o contexto envolvido nos experimentos é de extrema importância já que o ambiente externo e as características dos aparelhos usados poderão influenciar os resultados obtidos. Por exemplo, fatores como o ruído do ambiente, configuração sociais, Qualidade de Serviço (QoS) da rede, entre outras variáveis de ambiente, podem influenciar a usabilidade do aplicativo e devem ser consideradas nos experimentos.

Outro fato relevante apresentados nestes estudos, é que os três experimentos que consideraram informações contextuais foram executados em laboratório, ou seja, nenhuma investigação realizada em campo relaciona os dados de usabilidade a fatores externos.

Para finalizar a discussão dos dados obtidos no estudo realizado, é apresentado um gráfico com o número de vezes que cada plataforma foi utilizada nos experimentos (Figura 6). É possível perceber a maior utilização do iPhone OS, seguido do Android e Symbian OS. Esta tendência reflete o que ocorre no mercado americano, onde o iPhone OS teve 51,2% das vendas contra 44,2% da plataforma Android no quarto trimestre de 2012³. No caso do Symbian OS, existem estudos que apontam a plataforma ainda como a mais utilizada em todo o mundo devido ao número de dispositivos existentes no mercado⁴.

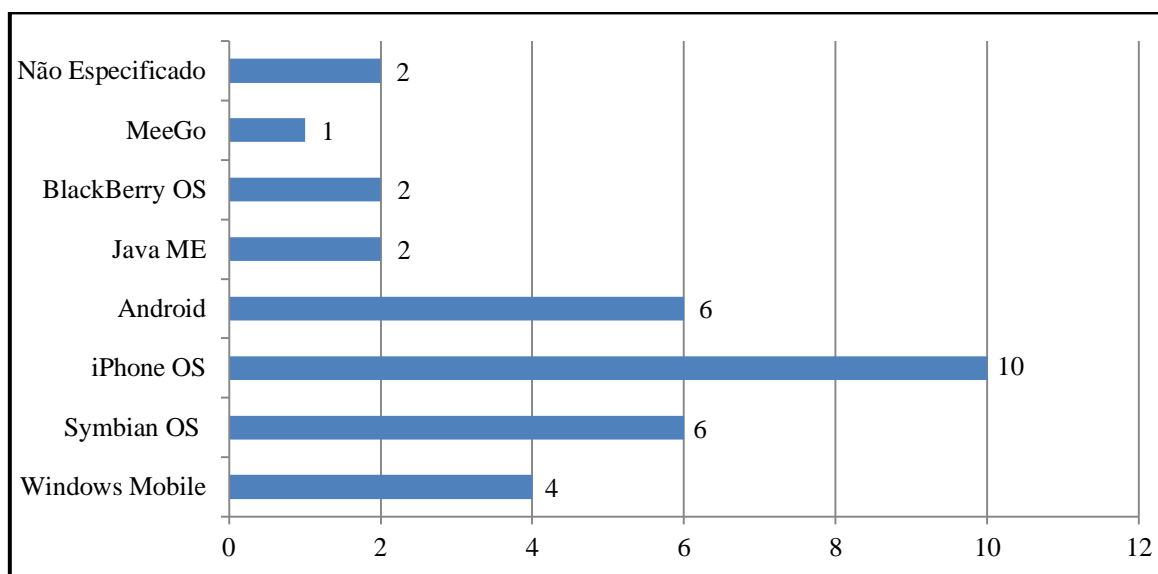


Figura 6 – Análise quantitativa das plataformas utilizadas nos experimentos

2.4 RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO

Vários campos de pesquisa estão vinculados ao sucesso mercadológico dos *smartphones*, entre elas podem ser destacados os avanços no acesso a dados e aplicações remotas, redes móveis, gerenciamento eficiente de energia, sensibilidade ao contexto e aplicações adaptativas. A maioria destes campos de estudo

³ Disponível em: <http://exame.abril.com.br/tecnologia/iphone/noticias/ios-ainda-e-a-plataforma-de-smartphones-mais-vendida-nos-eua>

⁴ Disponível em: <http://www.tudocelular.com/nokia/noticias/n25165/symbian-plataforma-movel-mais-usada.html>

surgiu como subáreas da computação ubíqua e pervasiva, ganhando notoriedade à medida que a computação móvel avançava, conquistava adeptos e proporcionava novas aplicabilidades.

A IHC foi outra área de pesquisa que teve que encarar novos desafios, tanto para a especificação de interfaces apropriadas aos novos aplicativos, com na evolução de diferentes formas de interação com os dispositivos. Como novas tecnologias requerem avaliações para entender as necessidades e exigências do público consumidor, os testes de usabilidade tornaram-se indispensáveis para os avanços nessa área.

Nas últimas duas décadas, diversos pesquisadores definiram atributos de usabilidade e criaram *frameworks* contemplando suas propostas. Entretanto, percebe-se que não existe um consenso sobre quais atributos devem ser contemplados. Desta forma, foi necessário unificá-los já que muitos são similares e, embora tenham nomenclaturas diferentes, abordam os mesmos aspectos. Sendo assim, foram considerados nove atributos que serão utilizados no decorrer deste trabalho para identificar os aspectos de usabilidade. São eles: eficiência, eficácia, satisfação, aprendizagem, operabilidade, acessibilidade, flexibilidade, utilidade e facilidade de uso.

Um dos aspectos mais enfatizados entre os pesquisadores da área é que o contexto de interação deve estar associado aos dados de usabilidade já que os novos aplicativos são utilizados nos mais variados ambientes, com inúmeras interferências externas. Além disso, grande parte dos autores defende a realização de experimentos em campo, justificando que em laboratório não é possível reproduzir todas as variáveis contextuais existentes no cotidiano dos usuários.

Com a perspectiva de entender o que está sendo proposto nos novos experimentos de usabilidade, foi realizado um estudo dos trabalhos com relatos das técnicas e abordagens propostas nos últimos cinco anos. Após a análise de 21 trabalhos onde há avaliação da usabilidade em aplicativos para *smartphones*, foi constatado que apenas três investigaram dados contextuais. Além disso, os três experimentos foram conduzidos em laboratório.

Outro ponto a destacar é que 71,43% da coleta de dados foram realizadas por meio de questionários e nenhum dos trabalhos utiliza dados capturados dos sensores dos *smartphones* para serem contemplados nas avaliações de usabilidade. Além disso, as abordagens utilizadas não contemplam a captura da impressão dos usuários com relação à usabilidade do aplicativo durante as interações do usuário.

A conclusão é que existe carência de abordagens investigativas que possam considerar os requisitos de avaliações de usabilidade sugeridas em inúmeros trabalhos da área. Sendo assim, são necessárias novas propostas que contemplem essas abordagens e disponibilizem ferramentas adequadas e flexíveis, capazes de serem configuradas de acordo com as necessidades dos pesquisadores.

O sucesso nas avaliações de usabilidade de aplicativos para *smartphones* requer a coleta de diferentes informações, por longos períodos de tempo e com a participação de diferentes perfis de usuários, já que as interações sofrem influências do cenário de utilização. Estas restrições introduzem a necessidade de

uma proposta que contemple o tratamento de grandes volumes de informações, com a disponibilidade de ferramentas para a coleta de dados automática, associada a um ambiente que permita a transmissão contínua dos dados, integrada a um local de armazenamento com escalabilidade que possibilite a análise e correlação das informações de forma fácil e eficiente. Estes requisitos foram idealizados inicialmente por Balagtas-Fernandez e Hussmann (2009) e são contemplados na proposta desta tese.

Este capítulo apresenta a metodologia adotada para o desenvolvimento deste trabalho, abordando as distintas técnicas utilizadas em função das etapas executadas. O objetivo principal é descrever a concepção das técnicas empregadas e correlacionar com as ações tomadas durante o desenvolvimento do trabalho, possibilitando que outras pessoas possam seguir o mesmo caminho para atingir resultados semelhantes ou estender o que foi realizado na presente pesquisa.

3 METODOLOGIA

Os projetos de pesquisa científica têm como principal objetivo organizar as etapas necessárias para responder às perguntas relacionadas com os problemas humanos. A criação de artefatos inovadores é uma parte importante do processo e a descrição das etapas para reproduzir tais artefatos é fundamental na compreensão dos problemas. A metodologia *Design Science Reseach (DSR)*, proposta por Hevner e Chatterjee (2010), foi utilizada para direcionar as etapas executadas durante a realização desta tese. Essa metodologia contempla cinco fases distintas, conforme pode ser observado na Figura 7.

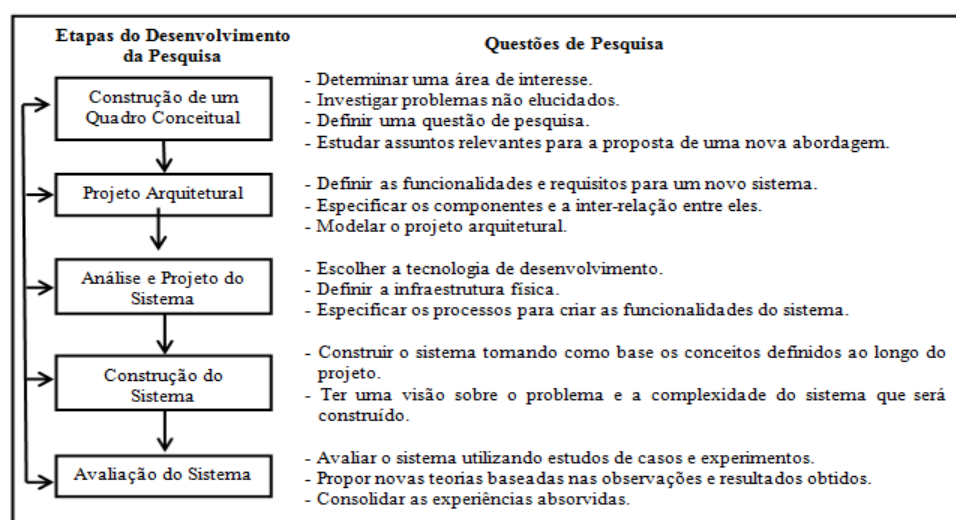


Figura 7 – Etapas do desenvolvimento do projeto de pesquisa (Hevner e Chatterjee, 2010).

A Tabela 11 apresenta uma visão geral de como o andamento deste trabalho se integra com as etapas do desenvolvimento do projeto descrito por Hevner e Chatterjee (2010). Nas próximas seções deste capítulo, será aprofundada a correlação da metodologia de pesquisa DSR com as etapas desenvolvidas ao longo desta pesquisa.

Tabela 11 – Relação das etapas da metodologia DSR com a presente pesquisa

Etapas de Desenvolvimento	Questões de Pesquisa
Construção de um Quadro Conceitual	<ul style="list-style-type: none"> - Estudo aprofundado da literatura com a intenção de identificar pontos ainda não explorados na área de avaliação de dispositivos móveis. - Investigação de funcionalidades das abordagens existentes com a intenção de definir novos requisitos que possam complementar as atuais propostas. - Estudo de disciplinas relevantes que pudessem ser utilizadas para subsidiar a criação de uma nova abordagem.
Projeto Arquitetural	<ul style="list-style-type: none"> - Definição do modelo de uma nova abordagem para avaliação de aplicativos para dispositivos móveis, abrangendo sua arquitetura, modularização e extensibilidade. - Demarcação das funcionalidades de cada componente do modelo e as inter-relações entre eles.
Análise e Projeto do Sistema	<ul style="list-style-type: none"> - Projeto de uma infraestrutura que pudesse contemplar as especificações do modelo. - Definição dos processos envolvidos na criação da infraestrutura. - Especificação de soluções tecnológicas e escolha das mais convenientes para cada componente representado na infraestrutura.
Construção do Sistema	<ul style="list-style-type: none"> - Construção de um sistema que tem como base os conceitos definidos nas fases anteriores. - Elaboração de uma prova de conceito, instanciado para a investigação de aplicativos para <i>smartphones</i>.
Avaliação do Sistema	<ul style="list-style-type: none"> - Especificação e execução de experimentos que pudessem validar as funcionalidades do sistema e a descrição dos processos envolvidos para a sua utilização. - Consolidação das experiências adquiridas por intermédio da descrição dos resultados obtidos nos experimentos.

3.1 CONSTRUÇÃO DE UM QUADRO CONCEITUAL

Como foi descrito na seção anterior, a primeira etapa para a elaboração de um projeto de pesquisa se constitui na construção de um quadro conceitual que possibilite a identificação do problema e a motivação para resolvê-lo. A definição do problema é utilizada como requisito para o desenvolvimento dos artefatos. De acordo com Hevner e Chatterjee (2010), esta etapa serve para motivar os pesquisadores a investigar a solução do problema e ajudar a comunidade a entender o raciocínio que será utilizado para resolvê-lo.

O conhecimento do estado da arte, no campo específico da investigação, é obtido a partir da leitura de pesquisas semelhantes, publicadas em anais de conferências e periódicos, sendo essencial para a

contextualização do problema e o posicionamento do trabalho diante da comunidade, como foi realizada no Capítulo 2 (Estado da Arte) desta tese.

Uma visão aprofundada sobre o tema de estudo é importante para o entendimento de quais problemas foram sanados anteriormente e dos que ainda estão em aberto, identificação das abordagens adotadas e justificativas para a atual pesquisa. É importante que os pesquisadores estejam seguros de que a pesquisa contém novos aspectos ainda não abordados, evitando desgaste de energia com problemas anteriormente solucionados (Hansen, 2012).

Muitas vezes, novas ideias em uma determinada área de pesquisa são estimuladas a partir da leitura de outras pesquisas com o mesmo tema. Este fato ocorre devido à descoberta de aspectos relevantes em pesquisas anteriores que não foram tratados ou, ainda, a continuidade de trabalhos precocemente concluídos (Grønli, 2012).

Em cada etapa da elaboração desta pesquisa, foram estudados os trabalhos correlatos que circundavam o tema específico investigado no momento. Em vários casos, a identificação de problemas foi o resultado de uma experiência anterior vivenciada por autores da área em estudo. Em outros casos, foi escolhido um determinado caminho devido à socialização de conhecimento proporcionado por outros autores.

3.2 PROJETO ARQUITETURAL

A etapa que contempla o desenvolvimento de um sistema arquitetural tem como propósito identificar a complexidade do problema, definir os requisitos necessários para a sua solução e a modularização do problema em partes menores. A especificação dos módulos, seus requisitos funcionais e não funcionais, seus artefatos de entrada e saída, possibilitam identificar as tarefas que envolvem a resolução do problema (Grønli, 2012).

A visão geral do sistema projetado fornece informações importantes, não apenas para o planejamento e desenvolvimento dos módulos, mas também como uma ferramenta de comunicação. Em um ambiente empresarial, um modelo abstrato do sistema fornece aos clientes uma visão geral de como este funciona. Para projetos de pesquisa, os modelos podem informar aos pesquisadores o comportamento do sistema e como as suas partes vitais cooperam (Hansen, 2012).

A representação de um sistema no nível arquitetural compreende questões estruturais, entre as quais se destacam: a divisão do sistema em componentes ou subsistemas, interconexões entre os

componentes, seleção de alternativas de projeto e atribuições de funcionalidades a componentes do projeto. Porém, embora os aspectos estruturais recebam sempre maior destaque nas definições da arquitetura, estes não são os únicos a serem considerados para se obter uma completa e compreensível descrição arquitetural de um sistema.

Conforme afirma Falessi et al. (2011), a arquitetura do sistema deve ser vista e descrita em função de diferentes perspectivas, abrangendo a identificação dos seus componentes, relacionamentos estáticos, interações dinâmicas, propriedades, características e restrições. A maior parte das representações de projetos arquiteturais reconhece a existência de diferentes visões explicitamente, entretanto existem alguns modelos que focam em uma visão particular no sentido de explorar as suas características específicas, distinguindo-as das demais (Falessi et al., 2011).

Nesta pesquisa, a abordagem utilizada é denominada de Modelo de Visão Arquitetural 4+1 (Kruchten, 1995), a qual estrutura a representação dos modelos em cinco visões concorrentes. Conforme pode ser observada na Figura 8, cada visão trata um conjunto de objetivos específicos do projeto arquitetural de acordo com os interesses dos diferentes *stakeholders* (papéis interessados no desenvolvimento do projeto, como usuários finais, programadores, engenheiros, entre outros). Para contemplar a descrição do modelo proposto nesta tese a Figura 8 é uma adaptação da proposta original de Kruchten (1995).

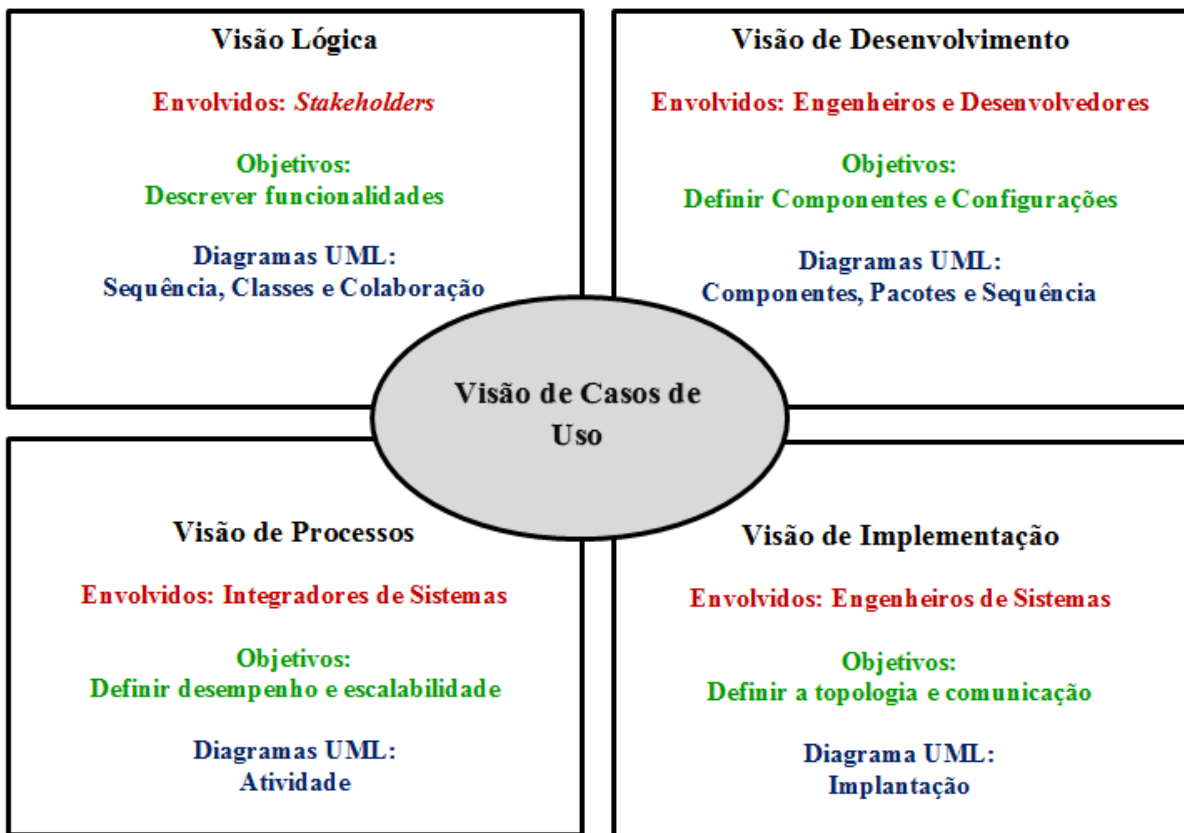


Figura 8 – Visões no Modelo 4+1 (Kruchten, 1995).

Os engenheiros descrevem suas decisões de projeto em quatro visões, usando a quinta visão apenas para ilustrar e validar as demais. As visões propostas pelo modelo são:

- Visão Lógica: descreve a perspectiva estática do sistema em nível conceitual, demonstrando os requisitos comportamentais do sistema em um conjunto de abstrações. Concentra-se nas funcionalidades que o sistema disponibiliza para os *stakeholders*. Os diagramas UML usados para representar a visão lógica incluem: Diagrama de Classes, Diagrama de Colaboração e Diagrama de Sequência.

- Visão de Desenvolvimento: descreve a organização dos módulos do sistema no seu ambiente de desenvolvimento. Ilustra o sistema do ponto de vista do desenvolvedor utilizando os diagramas de Componentes ou de Pacotes para representar os módulos e suas relações. O diagrama de Sequência também pode ser utilizado para definir a ordem cronológica das ações necessárias para a execução das fases e execuções.

- Visão de Processo: Permite visualizar as partes dinâmicas do sistema, explicar os processos e como eles se comunicam, focando no comportamento do sistema. O diagrama de Atividades é usado nesta visão.

- Visão Implantação (Física): mostra o sistema do ponto de vista do engenheiro. Preocupa-se com a topologia dos componentes (no contexto físico) assim como a comunicação entre esses componentes. O diagrama UML utilizado para descrever esta visão é o diagrama de Implantação.

- Visão de Caso de Uso: é utilizada para ilustrar as funcionalidades do sistema, auxiliando as descrições das outras visões e introduzindo ações e condições em cada caso de uso.

Kruchten (1995) ressalta que, nem sempre, um projeto arquitetural requer representações em todas as visões do modelo. Visões que não são úteis podem ser omitidas da descrição arquitetural. Por exemplo, a visão de processo poderia ser eliminada caso existisse somente um processo no sistema.

Para a especificação do projeto arquitetural deste trabalho, foi empregada a linguagem de modelagem conhecida como *Unified Modeling Language* (UML) para a representação gráfica dos componentes conforme recomendado pela abordagem 4 + 1.

A UML é uma linguagem ou notação de diagramas para especificar, visualizar e documentar modelos de projetos orientados a objetos. Basicamente, permite que os engenheiros visualizem seus projetos por meio de diagramas padronizados. Junto com uma notação gráfica, a UML também especifica significados, isto é, semântica para as representações. Atualmente, a UML é a norma mais usual para a descrição de componentes e suas inter-relações (Guedes, 2011).

A principal razão para usar UML neste projeto foi porque esta padronização permite comunicar certos conceitos mais claramente do que as linguagens alternativas. A linguagem natural é muito imprecisa e difícil de retratar conceitos mais complexos. As linguagens de programação são precisas, mas muito detalhadas. Assim, utilizar UML é uma boa alternativa quando precisamos de certa precisão, mas não

queremos nos perder em detalhes. Isso não significa evitar detalhes, ao contrário, utiliza-se UML para salientar detalhes importantes do projeto.

Após vários testes com ferramentas disponíveis no mercado para a criação de projetos arquiteturais que contemplassem os diagramas UML, a ferramenta escolhida foi a *Visual Paradigm for UML Enterprise Edition*⁵, versão 10.0. Basicamente, a escolha dessa ferramenta ocorreu em função do rápido aprendizado, da sua integração com as principais linguagens de programação e da variada gama de opções de diagramas disponíveis, contemplando todos os aspectos arquiteturais necessários para o desenvolvimento do projeto.

No Capítulo 4, será apresentado o modelo arquitetural proposto nesta pesquisa, utilizando as definições sugeridas na abordagem 4+1. Desta forma, o modelo será representado em diagramas UML utilizando as visões que possibilitam evidenciar as contribuições, prover mecanismos formais de validação da arquitetura e permitir a replicação do trabalho desenvolvido.

3.3 ANÁLISE E PROJETO DO SISTEMA

Para a continuidade da pesquisa, foi realizada a análise e projeto de uma infraestrutura com a finalidade de contemplar os requisitos definidos no projeto arquitetural. Para isso, foi necessário desenvolver uma análise detalhada das tecnologias que pudessem dar subsídios para a construção do projeto.

Nas próximas subseções, serão abordadas as tecnologias e técnicas utilizadas para contemplar os requisitos funcionais proposto no projeto arquitetural, considerando algumas premissas, tais como:

(i) A captura dos dados de interação do usuário deverá ser automática. Esta premissa objetiva a captura de dados nos mais diversos cenários, com o mínimo de intrusão no cotidiano dos usuários, além de contemplar experimentos realizados em campo ou em laboratório.

(ii) As técnicas utilizadas para a captura de dados não devem interferir no código fonte original das aplicações. Isso possibilita que a proposta desta pesquisa possa ser empregada em diferentes aplicativos, bem como as investigações podem ser iniciadas e finalizadas de acordo com a necessidade dos avaliadores. Além disso, permite que os resultados observados possam servir de subsídio para o próprio desenvolvedor do aplicativo ou para empresas que terceirizam este trabalho.

⁵ Disponível em: <http://www.visual-paradigm.com/>

(iii) A abordagem proposta deve ser de baixo custo financeiro. Esta decisão tem a intenção de propor uma abordagem viável economicamente, possibilitando a sua utilização em larga escala.

Para a realização deste levantamento, foram especificados cinco pontos de interesse: (i) os paradigmas e a linguagem em que os componentes são desenvolvidos; (ii) o Sistema Gerenciador de Banco de Dados (SGBD) utilizado; (iii) a ferramenta para dar suporte a análise dos dados; (iv) a base de transmissão e armazenamento dos dados; e (v) as técnicas utilizadas para a captura dos dados. Estes pontos de interesse serão discutidos nas próximas subseções.

3.3.1 Paradigmas e linguagem de programação

Para o desenvolvimento dos componentes, foi escolhido o paradigma Orientado a Objetos (OO) em função das especificações do modelo arquitetural ter sido representado nesse paradigma, utilizando os diagramas UML.

O paradigma OO é uma estratégia de projeto em que os projetistas pensam em termos de “coisas”, em vez de operações ou funções (Sommerville, 2005). Uma vez escolhido o paradigma OO para a definição do projeto, é recomendado que a análise e a programação sigam o mesmo caminho (Sommerville, 2005).

Para contemplar a segunda premissa definida neste trabalho, foi necessário aliar o paradigma OO ao paradigma Orientado a Aspectos (OA) (Kiczales, 1997). Para isso, é necessário utilizar uma linguagem que proporcione o desenvolvimento de códigos associando Programação Orientada a Objetos (POO) e Programação Orientada a Aspectos (POA) (Goetten Junior e Winck, 2006).

O principal objetivo da POA consiste em separar os interesses funcionais (aspectos relacionados ao domínio da aplicação) dos interesses sistêmicos (características de suporte a aplicação). Neste caso, interesses são as características relevantes de uma aplicação agrupadas por similaridade. Um interesse pode ser dividido em um ou mais Aspectos que representam os requisitos da aplicação (Goetten Junior e Winck, 2006).

No caso deste projeto, os interesses sistêmicos são todos os códigos referentes à avaliação de usabilidade acoplada ao código fonte de uma aplicação que será avaliada. Como a POA trabalha em paralelo com outro paradigma, neste caso a POO, os conceitos aplicados em POO são extensíveis a POA. Maiores detalhes a respeito de POA podem ser encontrados no Anexo II desta tese.

Embora a estrutura do modelo proposto neste trabalho não exija necessariamente a utilização de POA, a recomendação para seu uso deve ser destacada principalmente por agregar características importantes, tais como:

- **Extensibilidade** – possibilidade de instrumentar as aplicações para serem avaliadas sem esforço de programação.

- **Reusabilidade** – possibilidade de instrumentar diferentes aplicações sem necessidade de modificar o código da infraestrutura previamente desenvolvido.

- **Separação de interesses** – possibilidade de separar os interesses transversais (código referente à infraestrutura de avaliação de usabilidade) do código relacionado à aplicação que será avaliada.

Outro fator determinante para a definição da linguagem utilizada no projeto está relacionado com o sistema operacional dos *smartphones* que se pretende utilizar. Segundo Gartner (2011), desde o lançamento da primeira versão do Sistema Operacional (SO) Android, em outubro de 2008, o crescimento da plataforma tem sido exponencial. Em 2011, os *smartphones* com essa plataforma representavam mais da metade de todas as vendas mundiais neste seguimento de mercado. Em paralelo a isso, também ocorreu uma grande demanda por aplicativos que executam nessa plataforma. Em maio de 2011, o Google informou que mais de 200 mil aplicativos estavam disponíveis no Android Market e que essas aplicações foram instaladas 4,5 bilhões de vezes no total (Barra, 2011).

O Android é a primeira plataforma para aplicações móvel completamente gratuita e de código aberto (*open source*). Para operadoras, fabricantes e usuários avançados, isso é muito vantajoso pois é possível utilizar gratuitamente o sistema operacional, além da facilidade em personalizá-lo (Lecheta, 2009).

Tomando como base a tendência de mercado e levando em consideração a gratuidade e a possibilidade de acesso ao código fonte, a plataforma Android foi adotada para a realização das provas de conceito do modelo especificado neste projeto.

A linguagem Java é à base de desenvolvimento de aplicativos para a plataforma Android, desta forma, utilizam-se todos os recursos dessa linguagem. A única diferença com relação às aplicações convencionais Java é que as classes dos aplicativos são convertidas para o formato Dalvik (*Dalvik Executable*), que representa a aplicação do Android compilada (Lecheta, 2009). De modo análogo aos outros aplicativos em Java, depois da aplicação Android ter sido compilada, juntamente com outras bibliotecas, é gerado um arquivo único, neste caso com a extensão *Android Package File (APK)* que é a aplicação pronta para ser instalada no *smartphone* (Lecheta, 2009).

Em função das exposições anteriores, a linguagem Java⁶ na versão 7.0 foi escolhida para a materialização do modelo devido aos seguintes aspectos:

⁶ Disponível em: <http://www.java.com>

- É disponibilizada gratuitamente.
- Possui compatibilidade com a plataforma Androide.
- Viabiliza a integração de POO com POA utilizando o AspectJ⁷.

Segundo Goetten Junior e Winck (2006), o AspectJ é um compilador que possibilita estender a linguagem de programação Java para compatibilizá-la com a POA. O AspectJ além de compatibilizar os paradigmas OO e OA, ainda atende outros requisitos extremamente importantes e essenciais:

- Compatibilidade total – todo programa Java válido é também um programa AspectJ válido.
- Compatibilidade de plataforma – todo programa AspectJ pode ser executado em uma JVM.
- Compatibilidade de ferramentas – deve ser possível estender ferramentas existentes para suportar o AspectJ de uma forma natural; isto inclui a *Integrated Development Environments* (IDE), ferramentas de documentação e ferramentas de projeto.

- Compatibilidade para o programador – Ao programar com AspectJ, o programador deve sentir-se como se estivesse utilizando uma extensão da linguagem Java.

A IDE utilizada para a construção do projeto foi o Eclipse⁸ na versão Indigo Service Release 1. A escolha desta ferramenta ocorreu principalmente pelo fato de ser gratuita e possuir *plugins* que possibilitam integrar a linguagem Java com o AspectJ e a plataforma Android, em um mesmo ambiente de desenvolvimento.

3.3.2 Sistema Gerenciador de Banco de Dados (SGBD)

Outras escolhas importantes nesta fase do projeto estão vinculadas como a modelagem, armazenamento e gerência dos dados coletados durante as avaliações de usabilidade.

Para o armazenamento dos dados, foi definido que o SGBD seria o MySQL Community Server⁹, versão 5.1.58. O MySQL é um SGBD que utiliza a linguagem SQL (Linguagem de Consulta Estruturada, do inglês *Structured Query Language*) como interface. É, atualmente, um dos bancos de dados mais populares, com mais de 10 milhões de instalações pelo mundo (Pereira et al., 2012). Entre os usuários do BD MySQL estão: NASA, HP, Nokia, Sony, Lufthansa, Alcatel, Cisco Systems, Google, entre outros (Pereira et al., 2012).

⁷ Disponível em: <http://javafree.uol.com.br/artigo/871488/AspectJ-em-20-minutos.html#ixzz2EZ7yqnxN>

⁸ Disponível em <http://www.eclipse.org/indigo/>

⁹ Disponível em <http://dev.mysql.com/downloads/>

Em virtude da escolha do MySQL para armazenamento dos dados, a ferramenta de gerência e modelagem que permitiu desenvolver o Diagrama de Entidade-Relacionamento (ER) das tabelas foi o MySQL Workbench 5.5.35 CE¹⁰. Esta ferramenta conta com um ambiente gráfico interativo e foi utilizada ao longo do projeto para modelar o BD e o *data warehouse* (DW). Como o MySQL Workbench conecta-se com o SGBD utilizando a família de protocolos TCP-IP, o BD pode ser gerenciado a partir de qualquer computador com acesso a Internet, independente de onde o SGBD esteja instalado.

3.3.3 Ferramenta para auxiliar na análise dos dados

Para realizar as análises dos dados contidos no DW, foi escolhida uma ferramenta OLAP. Segundo Kimbol e Ross (2002), uma ferramenta OLAP se refere a um conjunto de tecnologias voltadas ao acesso e análise de dados, tal que o objetivo final seja transformar dados em informações analíticas e prover relatórios sob diferentes perspectivas em uma determinada área de atuação.

Primak (2008) destaca que as ferramentas OLAP permitem uma visão conceitual multidimensional dos dados e representam os dados sob a metáfora de um cubo, que é uma forma de visualização mais útil aos usuários do que a tradicional visão tabular utilizada nos sistemas *Online Transaction Processing* (OLTP). Utilizando tais ferramentas, os usuários podem realizar suas próprias análises e relatórios. Segundo Butuza (2011), a plataforma OLAP é uma combinação de procedimentos com processamento analítico e apresentações gráficas.

Atualmente, existem no mercado diferentes plataformas OLAP, o que pode gerar indecisões no momento da escolha da mais apropriada. Como as ferramentas proprietárias exigem o custo de aquisição, o que vai de encontro à terceira premissa deste trabalho, optou-se por buscar uma solução sem custos financeiros. O principal problema encontrado foi com relação à escassa documentação destas ferramentas, o que dificulta a utilização. Além disso, normalmente são levantadas dúvidas a respeito da evolução continuada das ferramentas, uma vez que algumas delas são mantidas por uma comunidade de desenvolvedores, não havendo uma empresa mantenedora formalmente constituída.

Uma exceção entre as plataformas livres é a Pentaho¹¹, que vem evoluindo suas ferramentas e disponibilizando manuais informativos sobre a instalação e utilização da plataforma. Desenvolvido a partir

¹⁰ Disponível em <http://dev.mysql.com/downloads/workbench>

¹¹ Disponível em <http://community.pentaho.com/>

de 2004 pela Pentaho Corporation, o software foi considerado a melhor aplicação para *business intelligence* (BI) em 2008 pela InfoWorld (Dineley, 2008).

Târnaveanu (2012) apresenta, em seu trabalho, uma descrição e avaliação consistente com relação ao Pentaho, concluindo que é uma plataforma de análise flexível, escalável, segura e fácil de gerenciar. Em outro trabalho similar, Majchrzak et al. (2011) comparam a ferramenta Pentaho com outras ferramentas gratuitas e consideram, entre outros parâmetros, a utilização da Unidade Central de Processamento e memória RAM. Após a repetição de experimentos com variados parâmetros de entrada, a constatação é que a plataforma Pentaho apresenta o melhor desempenho. Os autores ainda ressaltam a maturidade da plataforma e acreditam que ela possa se igualar a outras ferramentas proprietárias existentes no mercado.

Com base nos relatos apresentados nos dois últimos parágrafos, foi escolhida a plataforma Pentaho Analysis Services, na versão 3.9.0, como a ferramenta OLAP para a realização das análises no escopo desta pesquisa.

3.3.4 Base de transmissão e armazenamento dos dados

Nesta fase, a principal preocupação foi definir um ambiente no qual os dados de usabilidade, capturados das interações dos usuários, pudessem ser armazenados e posteriormente analisados.

Segundo Zanger et al. (2010), uma nova tendência para armazenamento de dados é chamada de Computação na Nuvem (*Cloud Computing*). A ideia refere-se à utilização da capacidade de armazenamento, processamento e memória de servidores compartilhados e interligados por meio da Internet, seguindo o princípio da computação em Grade (*Grid*). Além disso, estes ambientes normalmente disponibilizam um conjunto de softwares que podem ser utilizados por seus clientes.

Segundo Rani (2012), a Computação na Nuvem tem atraído muita atenção principalmente pela possibilidade de melhorar o desempenho computacional com pouco investimento em hardware e software. Ela também pode oferecer uma infinidade de recursos, que normalmente são cobrados de acordo com o uso. Isso normalmente diminui consideravelmente os custos, tanto para os provedores do serviço quanto para os usuários.

O modelo e a infraestrutura propostos neste trabalho são bons exemplos da integração da computação móvel com a Computação na Nuvem, já que os dados de usabilidade colhidos nos experimentos são armazenados e avaliados utilizando a infraestrutura provida por um servidor na nuvem.

Após uma avaliação das principais empresas que oferecem este recurso, optou-se pela Amazon Elastic Compute Cloud (Amazon EC2). As principais razões que levaram a esta escolha foram: a gratuidade do serviço por um ano, espaço de armazenamento, facilidade para instalação dos softwares necessários para a implementação do projeto e disponibilidade de ferramentas integradas ao ambiente.

A configuração disponibilizada pela Amazon foi: Sistema Operacional Ubuntu¹² versão 11.10 Oneiric Server, memória RAM de 613 Mbytes e espaço para armazenamento de dados com 18 Gbytes.

Outro recurso de infraestrutura necessário para a realização do projeto está relacionado com a transferência dos dados capturados nos *smartphones* para o Amazon EC2. Sendo assim, foi necessária a instalação de um servidor de FTP (*File Transfer Protocol*) para contemplar este recurso. O software escolhido foi o ProFTPD¹³ versão 1.3.4. por ser o servidor de FTP mais usual no sistema operacional Ubuntu e com ampla documentação para instalação e gerencia de recursos.

3.3.5 Técnicas utilizadas para captura dos dados

Esta subseção aborda as técnicas que são utilizadas para coletar os dados de interação dos usuários. É de suma importância ressaltar que as decisões tomadas nesta fase são direcionadas em função das três premissas definidas na seção 3.3.

Outro ponto importante na adoção da técnica corresponde aos tipos de dados que se deseja obter. Levando em consideração que os dados almejados referem-se à usabilidade do aplicativo (dados quantitativos), contexto de utilização (dados contextuais) e impressão do usuário (dados subjetivos), optou-se por adotar mais de uma técnica, caracterizando desta forma o modelo como híbrido.

A primeira técnica é chamada de Captura Automática a partir da Aplicação (*Logging*) e contempla a coleta de dados referentes à interação do usuário com uma aplicação e a utilização de sensores para a coleta de dados contextuais. A segunda técnica é conhecida como Método de Amostragem de Experiência (*Experience Sampling Method - ESM*) e baseia-se no envio de perguntas ao usuário com a finalidade de identificar a sua satisfação de uso relacionada ao aplicativo.

¹² Disponível em: <http://www.ubuntu.com/>

¹³ Disponível em: <http://wiki.ubuntu-br.org/proftpd>

3.3.5.1 Técnica de captura automática a partir da aplicação

O *Logging* é uma técnica de avaliação de usabilidade que consiste no monitoramento e coleta automática de estatísticas relativas ao uso do aplicativo. Outras denominações podem ser encontradas na literatura como, por exemplo, captura automática de uso real do sistema por Nielsen e Mack (1994) ou coleta contínua de dados sobre o desempenho do usuário por Shneiderman e Plaisant (2004).

Segundo Ivory e Hearts (2001), os dados monitorados consistem de eventos disparados pelos usuários ao interagirem com um aplicativo, implicando respostas correspondentes do sistema, bem como a captura automática do relógio do sistema, visando à coleta de informações sobre a temporização dos eventos. Esses eventos permitem identificar padrões de comportamento dos usuários, tais como, velocidade de conclusão de tarefas, taxas de erros e frequência de uso de mecanismos de ajuda. Segundo os autores, este tipo de dado é extremamente valioso para alterações em procedimentos operacionais, criação de planos de expansão, otimizações de tarefas e personalizações para determinados grupos de usuários.

A captura de informações sobre o uso real de um aplicativo mostra-se particularmente útil para fins de avaliar a usabilidade porque reflete o desempenho dos usuários ao executarem tarefas cotidianas (Froehlich et al., 2007). Além disso, é possível coletar dados relativos a um grande número de usuários, atuando em diferentes circunstâncias, de um modo discreto e "transparente" (Nielsen e Mack, 1994).

Os benefícios advindos do monitoramento automático não se estendem apenas aos comandos e outros recursos do sistema de uso corrente. Recursos não utilizados ou raramente acessados poderão ser analisados com propósitos de otimização e/ou de implementação de mecanismos que os tornem mais acessíveis ao usuário. Em última instância, as análises feitas sobre tais recursos poderão conduzir à sua substituição por outros mais eficazes ou à sua remoção definitiva (Shneiderman e Plaisant, 2004).

Caso sejam realizadas análises estatísticas com os dados capturados, torna-se possível usar o *logging* como uma técnica de alertar à equipe de usabilidade sobre quaisquer alterações nas necessidades da comunidade usuária do sistema, traduzidas por mudanças no modo como utilizam o aplicativo (Nielsen e Mack, 1994).

A implantação da captura automática e análise dos dados de interatividade exigem a utilização de métricas para permitir a coleta dos dados de interesse. Além disso, é importante que sejam disponibilizadas APIs que facilitem a implementação de códigos específicos para a captura dos eventos de interação (Ickin et al., 2012). Sendo assim, nesta fase, é necessário realizar estudos referentes à quais dados devem ser coletados e, posteriormente, definir as métricas que contemplem os requisitos desejados (Moldovan e Tarta, 2006).

Segundo Ivory e Hearts (2001), um grande inconveniente desta técnica é que apenas ações diretas podem ser monitoradas, já que nenhuma informação subsidiária é registrada para respaldar a análise de outras atividades relacionadas ao cenário em que o experimento ocorre. Em outras palavras, o método pode mostrar o que os usuários fizeram, mas não por que o fizeram (Nielsen e Mack, 1994). Neste sentido, a proposta apresentada neste trabalho diverge destas afirmações já que se utilizam os sensores dos dispositivos para identificar algumas variáveis do cenário de interação, possibilitando cruzar os dados de usabilidade com os contextuais.

Outra dificuldade do método refere-se ao volume de dados a serem capturados, transmitidos e analisados. Entretanto, segundo González et al. (2008), o problema pode ser contornado com a utilização de ferramentas OLAP e ferramentas de mineração de dados, sendo essa uma das propostas deste trabalho.

O último problema reportado refere-se à necessidade de incorporar componentes de medida ao código fonte das aplicações para possibilitar quantificar as ações dos usuários. Normalmente, este código se torna intrusivo no contexto das aplicações (Moldovan e Tarta, 2006). Esse problema pode ser solucionado com a utilização da Programação Orientada a Aspectos (Tarby et al., 2009).

O modelo proposto neste trabalho captura automaticamente os dados de usabilidade por meio de métricas de usabilidade criadas com POA. Esta estratégia é uma boa alternativa para que o código funcional do aplicativo a ser avaliado não seja poluído por códigos que dizem respeito à captura de informações de usabilidade, possibilitando ainda que essas medições possam ser disponibilizadas ou retiradas da aplicação de acordo com os interesses da Equipe de Avaliação. Essa característica é o principal fator motivador para a utilização de POA já que outras metodologias advindas da Engenharia de Software como, por exemplo, a utilização de padrões de projetos, não cumpre esse papel com a eficácia desejada (Tarby et al., 2009).

3.3.5.2 Técnica *Experience Sampling Method* (ESM)

Para a captura de dados subjetivos, relacionados à percepção que o usuário tem a respeito de um aplicativo, foi necessário investigar técnicas psicológicas apropriadas para a coleta deste tipo de informação.

A técnica para coleta de dados denominada de ESM foi originalmente proposta nos anos 80, na área de psicologia social. Esta técnica requer que os participantes relatem suas experiências psicológicas reportando o grau de satisfação após um determinado evento acontecer (Brandstätter, 1983). Segundo Csikszentmihalyi e Larson (1992), os relatos durante ou logo após um determinado evento ocorrer reduzem

os vies de memória e potencializa a captura do estado psicológico associado ao evento estudado com mais precisão.

Originalmente, os participantes dos experimentos envolvendo a técnica ESM recebiam avisos sonoros com uma determinada frequência temporal para lembrá-los de preencher questionários curtos (em papel) sobre suas experiências momentâneas, sentimentos e pensamentos (Csikszentmihalyi e Larson, 1992). Um ponto negativo na proposta inicial é que o cronograma de avisos era definido aleatoriamente e raramente ocorria em sincronismo com o evento investigado (Chen e Fan, 2012).

Segundo Froehlich et al. (2009), nos dias atuais, com a popularidade dos dispositivos móveis, a técnica ESM obteve uma série de avanços propiciados pela tecnologia. Os avisos podem ser enviados via mensagens SMS ou estarem integrados no próprio contexto de um aplicativo móvel. No lugar de respostas em formulários de papel, é normalmente disponibilizada uma interface na qual o participante do experimento interage para responder o solicitado. A sincronização temporal melhorou, sendo possível atrelar diretamente ao acontecimento de um evento. Como as respostas são enviadas para uma base de dados remota, os pesquisadores podem acompanhar os resultados praticamente em tempo real (Chen e Fan, 2012).

Segundo Meschtscherjakov et al. (2009), o ESM possibilita medir duas dimensões: o tipo de emoção (positiva ou negativa) e a intensidade da emoção. Para isso, é usado um conjunto de figuras indicando estados emocionais relacionados à pergunta. A sequência das figuras refere-se à intensidade da emoção e podem ser interpretadas da esquerda para a direita (Figura 9) como: muito insatisfeito, insatisfeito, indiferente, satisfeito e muito satisfeito.



Figura 9 – Exemplo de pergunta proposta aos participantes.

Ickin et al. (2012), além de utilizar a técnica ESM para identificar a satisfação dos usuários a respeito de determinados eventos, ainda emprega a técnica para obter informações sobre o contexto das interações dos usuários, tais como, localização (trabalho, escola, residência, etc), contexto social (sozinho, com amigos, com a família, etc) e nível de mobilidade (parado, caminhando, motorizado, etc).

Hicks et al. (2010) associam a apresentação das mensagens com eventos capturados por sensores do *smartphone* como, por exemplo, ao passar por algum local específico (utilizando GPS), ao se aproximar de uma pessoa do seu ciclo de amizade (utilizando *bluetooth*), ou ainda de acordo com um determinado movimento (utilizando o acelerômetro).

Tomando como base os inúmeros trabalhos desenvolvidos usando a técnica ESM para dispositivos móveis (Barrett e Barrett, 2001) (Consolvo e Walker, 2003) (Intille et al., 2003) (Raento et al., 2005) (Sohn, 2006) (Khan et al., 2008) (Kukkonen et al., 2009) (Froehlich et al., 2009) (Hicks et al., 2010), optou-se por utilizar a técnica nesta pesquisa para criar interlocuções diretas com os usuários e coletar dados subjetivos a respeito da usabilidade dos aplicativos investigados.

3.4 CONSTRUÇÃO DO SISTEMA

Nesta fase, os trabalhos direcionaram-se para a elaboração de processos que auxiliaram a codificação da abordagem proposta nesta pesquisa. A Engenharia de Software provê um número razoável de técnicas relacionadas com o desenvolvimento controlado de sistemas. Com base nessas técnicas, os desenvolvedores devem ser capazes de lidar com a vasta gama de informações que compreendem o desenvolvimento de uma infraestrutura. Além disso, é de suma importância que esses desenvolvedores sejam capazes de organizar o trabalho e os recursos envolvidos no processo de codificação (Pham e Pham, 2011).

O processo de desenvolvimento de uma infraestrutura é definido como o conjunto de elementos responsáveis por transformar os requisitos dos usuários em uma solução computacional. Assim, o processo é basicamente composto por um conjunto ordenado de tarefas que estão relacionadas com uma série de outros elementos. Por exemplo, os recursos consumidos e modificados por essas tarefas, os produtos obtidos com a realização dessas tarefas, os procedimentos e condutas seguidos durante a execução, entre outros (Hirama, 2011).

Segundo Pham e Pham (2011), a modelagem de processos pode trazer para uma equipe de desenvolvimento uma série de benefícios como:

- Facilitar a comunicação e entendimento entre as pessoas.
- Possibilitar a melhoria continuada do processo de software.
- Facilitar a reutilização de componentes.
- Apoiar a gerência do projeto.

Há várias maneiras de representar um modelo de processo de software como, por exemplo, por meio de fluxogramas, descrições textuais, modelos orientados a objeto e meta-modelos.

Entre essas abordagens encontra-se o *Eclipse Process Framework* (EPF)¹⁴ que é um projeto de código aberto, tem como objetivo fornecer um ambiente e ferramentas para auxiliar a descrição de processos. O EPF utiliza o meta-modelo *Software Process Engineering Metamodel* (SPEM 2.0) que define a terminologia padrão para estruturar o conteúdo de métodos e processos (OMG, 2008). Baseado nas definições deste meta-modelo, é fornecido um conjunto de ferramentas (EPF Composer) que permitem a autoria, gerenciamento de conteúdo, configuração, visualização e publicação de processos e métodos. (Haumer, 2007).

Em função destas funcionalidades, foi escolhido o EPF Composer para a descrição e controle dos passos para o desenvolvimento da codificação dos componentes propostos no modelo que descreve a abordagem desta tese. Além disso, foi empregado o EPF Composer para explicitar todas as etapas necessárias para a utilização da infraestrutura que é utilizada como prova de conceito da abordagem proposta.

3.5 AVALIAÇÃO DO SISTEMA

Há inúmeras definições para avaliação já que é uma prática aplicada em muitas áreas, tais como, educação, administração, políticas públicas, além da computação. Segundo Marshall e Rossman (2011), avaliação é o processo de julgar o mérito e o valor de um sistema de informação.

Avaliação deve ser contrastada com os conceitos de verificação e validação. Verificação é o processo de julgar a aderência de um sistema de informação com a sua especificação e validação é o processo de julgar a eficiência que um sistema tem para resolver o problema para o qual foi concebido (Wainer, 2007).

As avaliações em Ciência da Computação, com o objetivo de extrair conhecimento, podem tomar como base os seguintes tipos de pesquisa: analítica, quantitativa, qualitativa e bibliográfica. As pesquisas quantitativas e qualitativas são coletivamente chamadas de pesquisas empíricas.

A pesquisa analítica é o método mais comum de se obter conhecimento sobre programas e algoritmos. Esta metodologia tem como base a definição de pressuposições sobre os dados do programa ou sobre o hardware onde o programa vai ser executado e provar matematicamente que o programa tem

¹⁴ Disponível em: <http://www.eclipse.org/epf/>

algumas propriedades interessantes. A análise de complexidade assintótica de algoritmos e o modelo de distribuição de probabilidade dos dados são alguns exemplos de pesquisa analítica (Mezard et al., 2002).

A pesquisa quantitativa é baseada na medida (normalmente numérica) de algumas variáveis objetivas, com ênfase na comparação dos resultados. Segundo Wainer (2007), os métodos quantitativos mais usuais na área de Ciência da Computação são:

- Uso de dados sintéticos: *benchmarks*, simulações e competições.
- Técnicas estatísticas para a comparação de conjuntos de medidas.
- Uso de questionários.
- Experimentos e casos de uso.

A pesquisa qualitativa baseia-se na observação cuidadosa do ambiente onde o sistema está ou será utilizado e do entendimento das várias perspectivas dos usuários. Segundo Wainer (2007), os métodos qualitativos mais usuais na área de Ciência da Computação são:

- Estudos qualitativos observacionais.
- Pesquisa-ação (ou estudos qualitativos intervencionistas).

As práticas mais comuns em pesquisas bibliográficas é realizar uma revisão sistemática ou uma meta-análise. Em ambas, o objetivo é abordar os principais artigos publicados que reportam uma determinada área de interesse e resumir os vários resultados. A revisão sistemática termina em uma avaliação qualitativa e quantitativa desses vários resultados como foi apresentado no final do Capítulo 2 desta tese.

Os experimentos realizados como parte dos estudos apresentados nesta pesquisa utilizam, na maioria dos casos, uma abordagem quantitativa abrangendo: (i) a execução das etapas para instrumentar os aplicativos com a abordagem proposta; (ii) a coleta de dados das interações dos usuários; e (iii) a avaliação do ambiente disponibilizado para a análise dos dados.

3.5.1 Experimento realizado em campo

Este experimento foi realizado com usuários de aplicativos para dispositivos móveis, para testar na prática a eficiência da abordagem proposta nesta pesquisa. O experimento abrange a avaliação da usabilidade de três aplicativos, com usuários reais, nos mais variados cenários de interação. Com relação à abordagem proposta, este experimento possibilitou a validação dos métodos de captura de dados e análise dos resultados.

A abordagem proposta, neste trabalho, tem como abrangência de estudo métodos e técnicas de avaliação Somativa (ver seção 2.2), já que utiliza as técnicas de *Logging* e *ESM*, as quais requerem que o aplicativo esteja pronto para ser utilizado. Além disso, as avaliações demandam a participação de usuários.

Os passos realizados durante todas as fases do experimento foram demarcados pelo *framework* DECIDE proposto por Sharp et al. (2007). O DECIDE orienta o planejamento, a execução e a análise de uma avaliação de IHC. As atividades do *framework* são interligadas e executadas iterativamente à medida que o avaliador articula os objetivos da avaliação, os dados e recursos disponíveis. As atividades do *framework* estão descritas na Tabela 12.

Tabela 12 – Atividades do *framework* DECIDE (Sharp et al., 2007).

D	Determinar os objetivos da avaliação de IHC. O avaliador deve determinar os objetivos da avaliação e identificar por que e para quem tais objetivos são importantes.
E	Explorar perguntas a serem respondidas com a avaliação. Para cada objetivo definido, o avaliador deve elaborar perguntas específicas a serem respondidas durante a avaliação.
C	Escolher (<i>Choose</i>) os métodos de avaliação a serem utilizados. O avaliador deve escolher os métodos mais adequados para responder às perguntas e atingir os objetivos esperados, considerando também o prazo, o orçamento, os equipamentos disponíveis e o grau de conhecimento e experiência dos avaliadores.
I	Identificar e administrar as questões práticas da avaliação como, por exemplo, o recrutamento dos usuários que participarão da avaliação, a preparação e o uso dos equipamentos necessários, os prazos e o orçamento disponível, além da mão de obra necessária para conduzir a avaliação.
D	Decidir como lidar com as questões éticas. Sempre que usuários são envolvidos numa avaliação, o avaliador deve tomar os cuidados éticos necessários.
E	Avaliar (<i>Evaluate</i>), interpretar e apresentar os dados. O avaliador deve considerar: o grau de confiabilidade dos dados; se o método de avaliação mede o que deve medir; se os resultados podem ser generalizados; e o quanto os materiais, métodos e ambiente de estudo se assemelham à situação real investigada.

3.5.2 Experimento para avaliar os processos da infraestrutura

Esse experimento foi concebido para avaliar a abordagem definida nesta tese na visão dos profissionais da área de Ciência da Computação, tais como, especialistas de IHC, engenheiros de software, desenvolvedores de sistemas e analistas de BD.

Os objetivos principais deste experimento foram:

- Avaliar os passos necessários para criar as métricas de usabilidade com a abordagem proposta e compará-la com a maneira tradicional que, neste caso, é realizar a codificação das métricas de usabilidade.

- Avaliar o ambiente de análise dos dados para obter o parecer dos especialistas em BD.

- Validar a descrição dos passos necessários para utilizar a infraestrutura, possibilitando investigar a viabilidade da abordagem a ser aplicada em larga escala por profissionais da área de ciência da computação.

O experimento seguiu o *framework* DECIDE para a sua concepção. Os dados foram obtidos por questionários aplicados após a execução das etapas propostas aos participantes. Na abrangência desta tese, questionários são um conjunto de perguntas com respostas pré-definidas ou perguntas de resposta fechada que são respondidas pelos participantes dos experimentos.

Uma das soluções mais utilizadas para a elaboração de respostas é a escala de Likert (1932), a qual requer que os entrevistados indiquem seu grau de concordância ou discordância com declarações relativas ao que está sendo medido. Atribuem-se valores numéricos e/ou sinais às respostas para refletir a força e a direção da reação do entrevistado à declaração. As declarações de concordância devem receber valores positivos ou altos enquanto as declarações das quais discordam devem receber valores negativos ou baixos. As escalas podem ir, por exemplo, de 1 a 5 ou de +2 a -2.

As principais vantagens da escala de Likert em relação às outras, segundo Conti e Pudney (2011), são: (i) a simplicidade da construção; (ii) o uso de afirmações que não estão explicitamente ligadas à atitude estudada, permitindo a inclusão de qualquer item que se verifique, empiricamente, ser coerente com o resultado final; e (iii) a amplitude de respostas permitidas apresenta informação mais precisa da opinião do respondente em relação a cada afirmação. Em função das vantagens citadas, as questões propostas no experimento realizado para avaliar os processos da infraestrutura concebida nesta pesquisa tomam como base a escala de Likert e podem ser encontradas no Anexo IX e X desta tese.

3.6 RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO

Os passos para a elaboração desta pesquisa seguiram as especificações da metodologia de pesquisa DSR, que é estruturada em cinco etapas, as quais estão correlacionadas, na Tabela 13, com os passos executados para o desenvolvimento desta tese.

Tabela 13 – Resumo da relação das etapas DSR e a execução desta pesquisa

Etapas DSR	Etapas da Pesquisa
Construção de um Quadro Conceitual	Estudo da literatura
Projeto Arquitetural	Definição do modelo.
Análise e Projeto do Sistema	Projeto da infraestrutura e especificações tecnológicas
Construção do Sistema	Elaboração de uma prova de conceito
Avaliação do Sistema	Especificação e execução de experimentos e avaliação dos resultados.

Para a construção do projeto arquitetural, foi utilizado o Modelo de Visão Arquitetural 4+1, o qual estrutura a representação do modelo em cinco visões: Lógica, de Processo, de Desenvolvimento, de implantação e de Casos de Uso. Cada visão trata um conjunto de objetivos específicos da arquitetura de acordo com os interesses dos diferentes papéis dos envolvidos no projeto. A visão de cenário ilustra o sistema em função das outras quatro visões, utilizando a linguagem UML para as representações.

Na análise e projeto do sistema, foram definidas as tecnologias e técnicas para a construção da infraestrutura levando em consideração os paradigmas, a linguagem em que os componentes devem ser desenvolvidos, o SGBD, a ferramenta para dar suporte a análise dos dados, a base de transmissão e armazenamento dos dados e as técnicas utilizadas para a captura dos dados. Na Tabela 14, é apresentado um quadro resumo das tecnologias escolhidas.

Tabela 14 – Tecnologias escolhidas para a construção da Infraestrutura

Recursos Necessários	Tecnologias Adotadas
Paradigmas de desenvolvimento	Orientação a Objetos e Orientação a Aspectos
Linguagem de desenvolvimento	Java
Extensão para POA	AspectJ
Plataforma para <i>smartphones</i>	Android
IDE para desenvolvimento	Eclipse na versão Indigo Service Release 1
SGBD	<i>MySQL Community Server</i> , versão 5.1.58
Ferramenta para manipular o BD	<i>MySQL Workbench 5.5.35 CE</i>
Ferramenta OLAP	<i>Pentaho Analysis Services</i> na versão 3.9.0
Computação na Nuvem	<i>Amazon Elastic Compute Cloud (Amazon EC2)</i>
Servidor de FTP	<i>ProFTPd</i> versão 1.3.4
Técnicas de captura dos dados	<i>Logging</i> e <i>ESM</i>

Para a descrição dos processos envolvidos com a construção do sistema e os passos necessários para o uso da infraestrutura, foi utilizado o *Eclipse Process Framework*. Por fim, para as avaliações da abordagem proposta, foi especificado um experimento em campo, com usuários de aplicativos para *smartphones*, com a finalidade de verificar as potencialidades da infraestrutura construída. Além disso, foi executado um experimento envolvendo especialistas na área de desenvolvimento de software e banco de dados com o objetivo de avaliar a utilização da abordagem por potenciais usuários.

Este capítulo aborda a concepção do modelo arquitetural proposto nesta tese e a infraestrutura UXEProject criada para evidenciar a sua materialização. O modelo está subdividido em visões, utilizando a notação UML, enquanto que a infraestrutura é apresentada em ambientes, nos quais os componentes estão instanciados. Para finalizar, são discutidas as formas de captura de dados empregadas na infraestrutura.

4 O MODELO E SUA INFRAESTRUTURA

Na análise apresentada na Seção 2.3.6, que sumarizou e discutiu alguns trabalhos publicados nos últimos cinco anos para a avaliação de usabilidade de aplicativos para *smartphones*, percebeu-se a carência de abordagens voltadas à captura automática de dados, contemplando diferentes tipos de informações que viabilizem avaliar a usabilidade de diferentes ângulos.

Apesar de muitos autores considerarem as variáveis contextuais como sendo essenciais para avaliações de usabilidade em *smartphones*, poucos trabalhos práticos consideraram este aspecto, além de utilizarem a coleta de dados por meio de questionários e conduzirem os experimentos em laboratório na maioria dos casos (Coursaris e Kim, 2011).

Em um trabalho proposto por Balagtas-Fernandez e Hussmann (2009), ficam evidentes os anseios da área de IHC por novas abordagens que supram os requisitos de avaliações de aplicativos para dispositivos móveis, contemplando diferentes aspectos. Entre estes aspectos estão à forma de instrumentar aplicativos para serem avaliados, a captura de dados contextuais e de interação por meio de *Logging*, além do local para armazenamento e posterior análise desses dados.

Neste capítulo será apresentado um modelo e sua materialização numa infraestrutura computacional para a avaliação da usabilidade de aplicativos para *smartphones*, contemplando alguns dos requisitos desejados pela comunidade de IHC para esse fim. O modelo foi concebido com as seguintes características:

- Propiciar coletas de dados no ambiente natural de interação do usuário com as aplicações, ou seja, possibilitar a realização de experimentos em campo. Entretanto, não inviabiliza a sua utilização para a realização de experimentos controlados em laboratório.

- Permitir avaliações que envolvam o cruzamento do perfil do usuário com dados quantitativos, contextuais e subjetivos. O perfil do usuário é obtido no momento que o usuário realizar o *download* do aplicativo para o seu *smartphone*.

- Possibilitar a realização de experimentos por longos períodos de tempo. Para isso, foi projetada uma infraestrutura de transmissão, armazenamento e análise de dados baseada em uma estrutura de computação na nuvem.

- Disponibilizar uma maneira simples e rápida de realizar as análises dos dados capturados. Este requisito é propiciado pela adoção de uma ferramenta OLAP.

- Dar suporte à instrumentação das aplicações de forma não intrusiva, sem alterações diretas no código fonte das aplicações. Essa característica foi possível com a utilização de POA.

- Possibilitar que adeptos da abordagem possam utilizá-la sem a necessidade de escreverem códigos de programação. Para isso, foram construídas ferramentas que possibilitam determinar as tarefas a serem investigadas e gerar o código necessário para a instrumentação dos aplicativos de forma automática.

O modelo arquitetural proposto tem como base quatro fases distintas que abrangem todos os passos envolvidos no processo.

A primeira fase refere-se à definição das tarefas que serão avaliadas. Resumidamente, essa ação requer que a equipe de avaliação interaja com a aplicação para indicar os passos (supostamente ideais) para a execução de uma tarefa. Para possibilitar o mapeamento das tarefas, é necessário que a aplicação seja preparada para indicar as ações interativas do usuário. Além disso, é necessário executar, em paralelo com o aplicativo, uma ferramenta que captura os métodos executados nas interações para realizar uma tarefa.

A segunda fase trata da instrumentação do código das aplicações com métricas de usabilidade, contextuais e subjetivas. Os códigos fontes que retratam essas métricas são gerados automaticamente, com o auxílio de uma ferramenta que une a estrutura das métricas previamente definidas em uma biblioteca com os métodos capturados no mapeamento das tarefas.

A terceira fase diz respeito à disponibilização do aplicativo instrumentado com as métricas de captura dos dados para a interação dos usuários. À medida que o usuário interage com o aplicativo, os dados são capturados e transmitidos para a infraestrutura projetada na nuvem.

A última fase refere-se à recepção, armazenamento e análise dos dados coletados. As avaliações destes dados possibilitam:

- Confrontar se o usuário final interage com a aplicação da maneira prevista pela equipe de desenvolvimento.

- Identificar problemas de usabilidade em função do contexto de interação.
- Investigar problemas específicos de usabilidade atrelados a uma determinada tarefa.
- Analisar a curva de aprendizado dos usuários.
- Avaliar o desempenho dos usuários na execução de tarefas.
- Confrontar o desempenho de usuários novos com usuários experientes.
- Utilizar o perfil dos usuários para analisar padrões de comportamento.
- Aproveitar as informações subjetivas fornecidas pelos usuários para constatar a sua experiência com o aplicativo, bem como, mensurar suas expectativas, grau de satisfação e fidelização.
- Evidenciar o comportamento dos usuários de *smartphones* quando passam a utilizar novos aplicativos.
- Avaliar a utilização e eficácia das opções de ajuda disponibilizadas no aplicativo.

O modelo arquitetural foi projetado com base em componentes em todas as suas unidades, possibilitando que seus recursos possam ser redefinidos de acordo com as necessidades dos avaliadores. Na Figura 10, são apresentados os três principais componentes e seus relacionamentos. As setas indicam a transferência de informações entre os componentes de alto nível que representam o modelo.

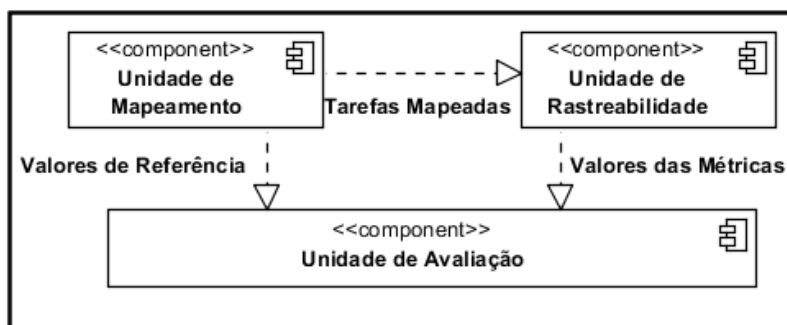


Figura 10 – Principais Componentes do Modelo

A Unidade de Mapeamento é responsável por disponibilizar os recursos necessários para a realização do mapeamento das tarefas que serão posteriormente investigadas no aplicativo que será avaliado. O resultado da sua execução é prover as Tarefas Mapeadas e Valores de Referência que indicam informações referentes à tarefa mapeada como, por exemplo, tempo para a sua execução, número de ações necessárias para executá-la, entre outras.

A Unidade de Rastreabilidade provê uma biblioteca de métricas que utiliza as tarefas mapeadas, possibilitando a instrumentação do código fonte da aplicação com métricas para a captura de dados de usabilidade, contextuais e subjetivos. Os valores obtidos durante as interações dos usuários são enviados para a Unidade de Avaliação.

A Unidade de Avaliação se responsabiliza por receber os dados transmitidos, transferi-los para uma base de dados e possibilitar as avaliações.

As próximas seções apresentam o modelo arquitetural, utilizando o esquema formal de descrição 4+1, proposto por Kruchten (1995) e já discutido na seção 3.2. Essa formalização permite evidenciar as contribuições da arquitetura, prover mecanismos formais para sua validação e replicação.

4.1 VISÃO DE DESENVOLVIMENTO

A visão de desenvolvimento descreve a organização das unidades do sistema e a forma de comunicação entre seus componentes. Os diagramas apresentados possibilitam a visualização dos componentes, suas funcionalidades e as portas de comunicação de dados. Para a descrição desta visão, foram utilizados diagramas de Componentes subdivididos nas unidades de Mapeamento, Rastreabilidade e Avaliação que representam as subdivisões da representação arquitetural do modelo.

A **Unidade de Mapeamento** é subdividida em três componentes responsáveis em prover as funcionalidades do mapeamento das tarefas. A Figura 11 ilustra o diagrama de Componentes, suas interfaces, bem como as portas de entrada e saída previstas para a passagem de dados de um componente para outro.

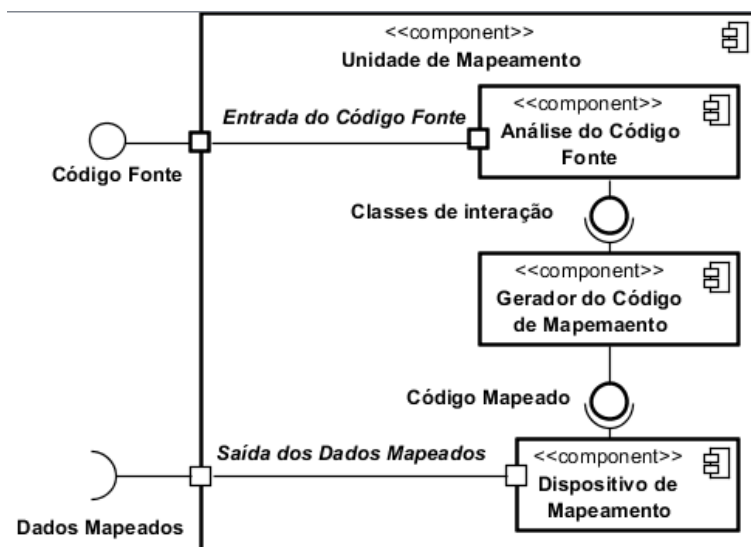


Figura 11 – Diagrama de Componentes da Unidade de Mapeamento

Inicialmente, é previsto que seja disponibilizado o código fonte da aplicação. O objetivo desta ação é permitir que o componente responsável em realizar a análise do código possa identificar quais classes referem-se ao tratamento das interações do usuário.

Essas classes são identificadas e fornecidas como pré-requisito para que o componente responsável em gerar o código de mapeamento possa acoplar, ao código fonte original, informações que possibilitem a identificação das tarefas. O resultado da ação deste componente é realizar a instrumentação do código original da aplicação, permitindo que suas tarefas possam ser mapeadas.

Após o código fonte estar preparado para a realização dos mapeamentos, ele deve ser passado para o componente representado pelo dispositivo onde as tarefas serão mapeadas. Este dispositivo deve possibilitar o mapeamento das tarefas e a disponibilização destas tarefas para outras unidades do modelo.

A **Unidade de Rastreabilidade** é responsável pela coleta de diferentes tipos de informações: (i) os dados referentes à interação do usuário como, por exemplo, acertos e tempo para executar uma tarefa; (ii) os dados de contexto como, por exemplo, luminosidade e deslocamento; e (iii) os dados subjetivos, referentes às experiências proporcionadas pelo aplicativo aos seus usuários.

Para a instrumentação do código fonte com as métricas de captura e realização das coletas de dados, a Unidade de Rastreabilidade foi dividida em três componentes internos mostrados na Figura 12.

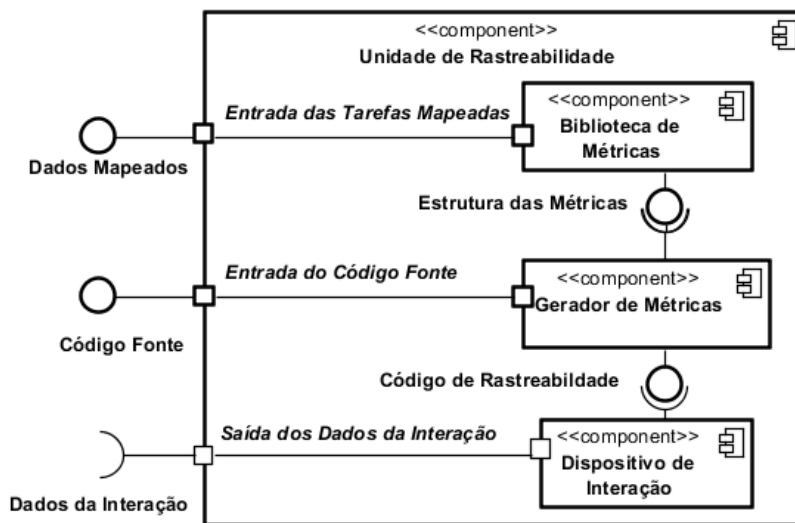


Figura 12 – Diagrama de Componentes da Unidade de Rastreabilidade.

O componente previsto na infraestrutura para disponibilizar a estrutura de todas as métricas é chamado de Biblioteca de Métricas. Este componente se utiliza das tarefas mapeadas para unir a estrutura das métricas a informações referentes ao código fonte da aplicação que será avaliada. O resultado do processamento deste componente é a geração das estruturas de métricas adaptadas ao aplicativo que será instrumentado.

O componente responsável pela Geração das Métricas recebe como entrada o código fonte da aplicação e as estruturas das métricas adaptadas para serem acopladas a este código. O resultado do processamento deste componente é a geração de um novo código fonte com as métricas inseridas.

No componente que representa o Dispositivo de Interação, deve ser instalado o novo código da aplicação contendo as métricas a serem utilizadas para a captura de dados. Este dispositivo deve possibilitar a interação do usuário permitindo que os dados sejam capturados.

A **Unidade de Avaliação** foi estruturada em quatro componentes distintos, responsáveis pela parte da infraestrutura onde os dados ficarão armazenados para serem avaliados ao longo do tempo. A Figura 13 apresenta uma visão arquitetural dos componentes desta unidade.

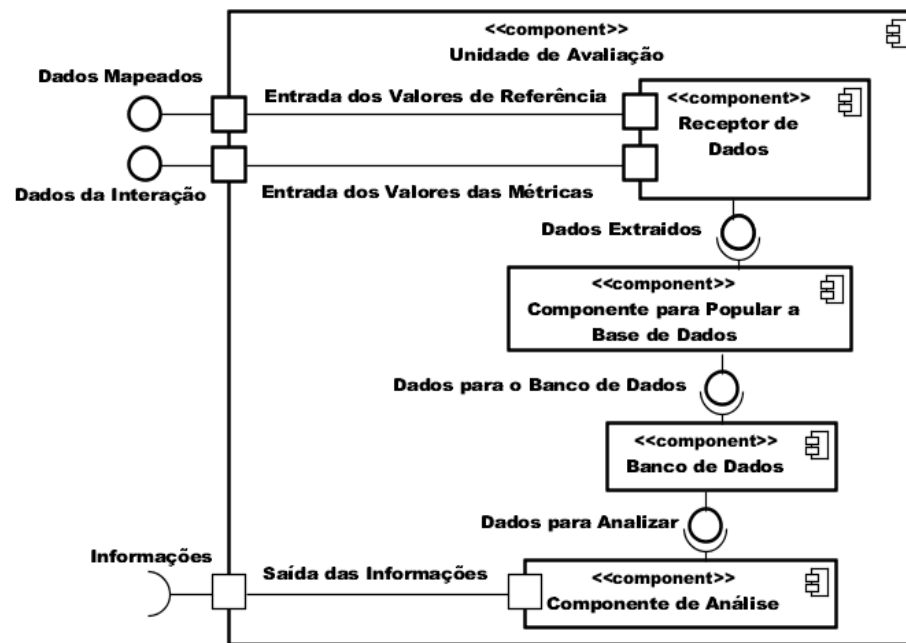


Figura 13 – Diagrama de Componentes da Unidade de Avaliação.

O primeiro componente é responsável pela recepção dos dados referentes ao mapeamento das tarefas e das interações dos usuários com os aplicativos utilizados nos experimentos.

O segundo componente tem como função realizar a verificação da chegada de novos dados de interação e realizar o seu desempacotamento. Os dados extraídos são passados para o terceiro componente que representa o repositório de dados.

O último componente tem a responsabilidade de acessar os dados existentes no repositório e disponibilizar recursos para facilitar a extração de informações de usabilidade relevantes para os avaliadores.

4.2 VISÃO LÓGICA

A visão Lógica na descrição 4+1 documenta a colaboração entre os componentes do modelo, evidenciando o processamento de tarefas no sistema e a ordem cronológica em que ocorrem. Para a representação dessa visão foi escolhido o diagrama UML de Sequência, subdividido novamente em função das três unidades arquiteturais do modelo.

Na Figura 14, é apresentado o diagrama de sequência que retrata a troca de mensagens e dados entre os componentes que fazem parte da **Unidade de Mapeamento**. A Equipe de Avaliação, que pode ser representada por pessoas com diferentes atribuições, é responsável em disponibilizar o código da aplicação.

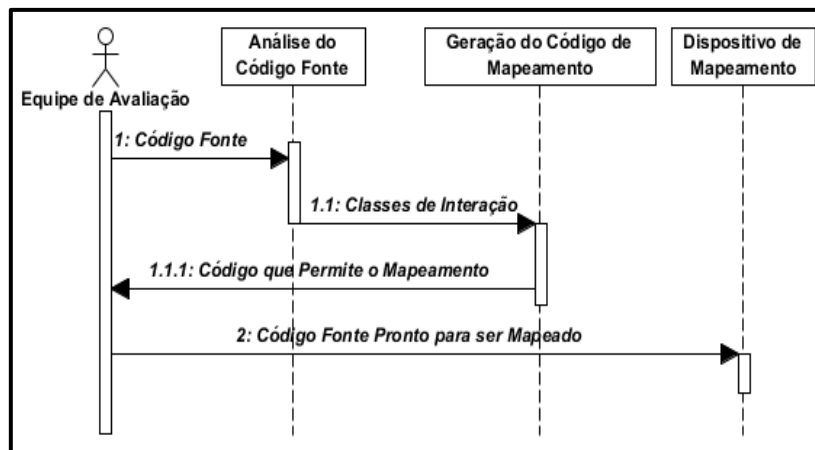


Figura 14 – Diagrama de Sequência da Unidade de Mapeamento

O componente responsável pela análise do código fonte verifica quais são as classes relacionadas à interação, ou seja, as que possuem métodos e atributos que geram eventos e disponibilizam componentes para a interação dos usuários com o aplicativo. Após a conclusão da análise, as classes com essas características são encaminhadas para o componente que gera os códigos de mapeamento.

O componente que prepara a aplicação para ser mapeada insere nas classes recebidas um novo método que ouve e filtra os eventos a serem disparados durante as interações dos usuários. Com a conclusão da fase anterior, a Equipe de Avaliação tem em mãos um novo código da aplicação que está preparado para dar suporte ao mapeamento das tarefas a serem avaliadas.

A última fase requer que a Equipe de Avaliação compile o aplicativo e instale a nova aplicação com o filtro de eventos no dispositivo onde serão realizados os mapeamentos das tarefas.

Para que as tarefas sejam mapeadas, é necessária a instalação de uma ferramenta para ser executada em paralelo à aplicação com a finalidade de capturar os métodos filtrados nas interações da Equipe

de Avaliação. Os métodos capturados é que representam o mapeamento das tarefas. A referida ferramenta será detalhada na descrição da infraestrutura que instancia o modelo (Seção 4.6.1).

Na Figura 15, é apresentado o diagrama de Sequência que retrata a troca de mensagens e dados entre os componentes que fazem parte da **Unidade de Rastreabilidade**.

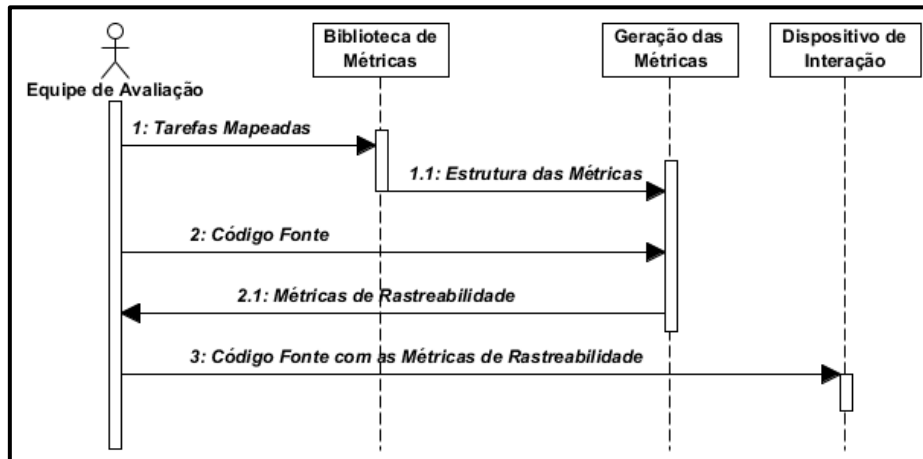


Figura 15 – Diagrama de Sequência da Unidade de Rastreabilidade.

Inicialmente, a Equipe de Avaliação deve disponibilizar para a Biblioteca de Métricas os métodos que representam as tarefas mapeadas. A Biblioteca de Métricas ao receber os métodos, forma as estruturas das métricas que serão utilizadas para rastrear a aplicação. Posteriormente, a Equipe de Avaliação precisa informar a localização do código fonte da aplicação para que as métricas sejam associadas ao aplicativo. O resultado é um novo código fonte com as métricas de rastreabilidade.

Após a execução do processo anterior, a Equipe de Avaliação necessita compilar o novo código gerando o aplicativo com as novas funcionalidades. Finalmente, a Equipe de Avaliação deve disponibilizar o aplicativo para que o usuário final possa instalá-lo em seu dispositivo. É recomendável que os aplicativos sejam disponibilizados em um site Web onde os dados referentes ao perfil do Usuário Final e o dados do seu *smartphone* possam ser informados.

Após a instalação do aplicativo no *smartphone* do Usuário Final, os dados de usabilidade, contexto e subjetivos são capturados e enviados para a Unidade de Avaliação.

O diagrama de Sequência da **Unidade de Avaliação** é apresentado na Figura 16. Nesta figura podem ser observadas as trocas de mensagens e dados entre os seus componentes.

O componente responsável pela recepção dos dados é um servidor FTP, que deve ter um endereço válido na Internet para que os dados de mapeamento e rastreabilidade possam ser enviados pelo dispositivo.

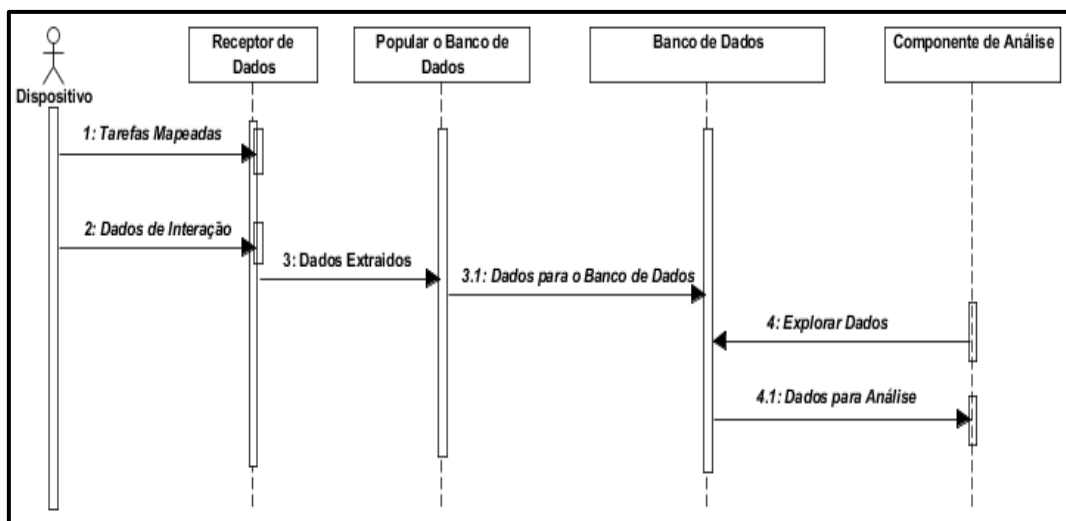


Figura 16 – Diagrama de Sequência da Unidade de Avaliação.

O próximo processo é identificar a chegada de novos dados de rastreabilidade e colocá-los em um BD. Este processo é de responsabilidade do componente preparado para popular o repositório de dados.

O componente que representa o repositório de dados é um SGBD que, além de guardar os dados, deve disponibilizá-los para o processo de análise executado por um componente específico para este propósito como, por exemplo, um minerador de dados ou uma ferramenta OLAP.

4.3 VISÃO DE CASOS DE USO

Nesta seção, são apresentados os diagramas de Caso de Uso para cada unidade do modelo, ressaltando os requisitos funcionais e as interações dos Atores com o sistema.

Para a identificação dos requisitos funcionais da **Unidade de Mapeamento** é apresentado um Caso de Uso que relaciona as interações da Equipe de Avaliação durante a fase de mapeamento das tarefas (Figura 17).

Inicialmente, é possível identificar que a Equipe de Avaliação pode ser representada por Atores com diferentes papéis na elaboração do sistema. As interações deixam claros os requisitos funcionais necessários para a definição do modelo, tais como, a disponibilização do código fonte, a preparação dos aplicativos para serem mapeados, a instalação dos aplicativos no *smartphone* e o mapeamento das tarefas.

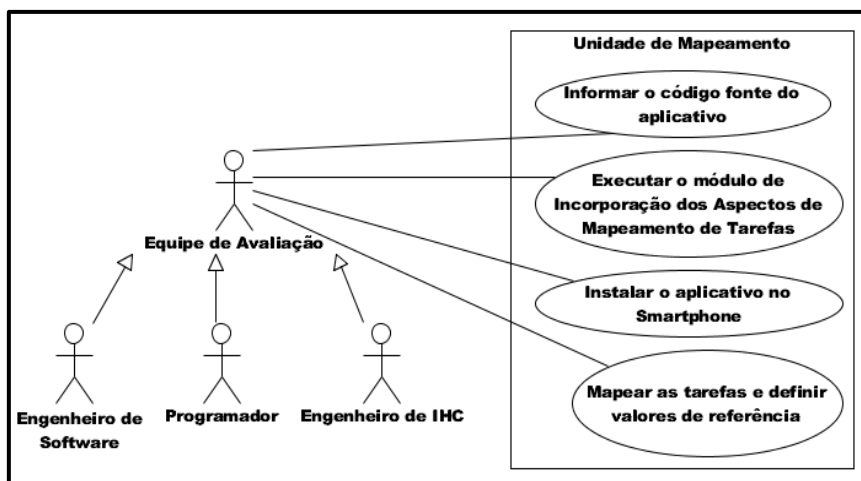


Figura 17 – Caso de Uso da Unidade de Mapeamento

Na Figura 18, é apresentado um diagrama de Caso de Uso referente às ações dos Atores relacionados à **Unidade de Rastreabilidade**. O primeiro Ator (Equipe de Avaliação) é responsável em fornecer os arquivos com as tarefas mapeadas, o código fonte da aplicação e disponibilizar o novo aplicativo com as métricas de captura de dados para que o Usuário Final tenha acesso. O segundo Ator (Usuário Final), deve instalar o aplicativo em seu *smartphone* e utilizá-lo permitindo que os dados de usabilidade, contextuais e subjetivos sejam capturados.

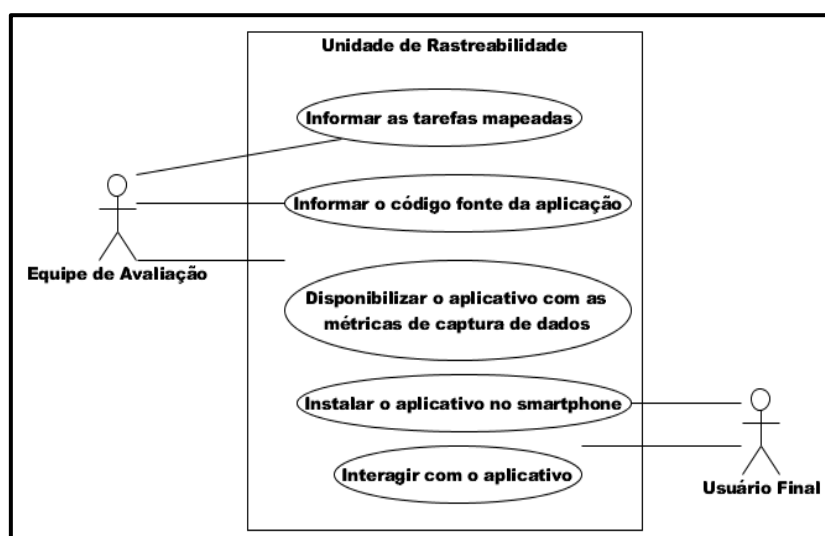


Figura 18 – Caso de Uso da Unidade de Rastreabilidade.

No diagrama de Caso de Uso da **Unidade de Avaliação** apresentado na Figura 19, pode ser observado que o *smartphone* é responsável em transmitir os dados referentes ao mapeamento das tarefas e as métricas com os valores coletados nas interações realizadas com o aplicativo.

A Equipe de Avaliação, por sua vez, resgata as informações referentes às tarefas mapeadas para instrumentar os aplicativos com as métricas de usabilidade. Além disso, é responsável em realizar a análise dos dados, definir parâmetros e configurações de interesse para que as informações de usabilidade sejam extraídas conforme as suas necessidades.

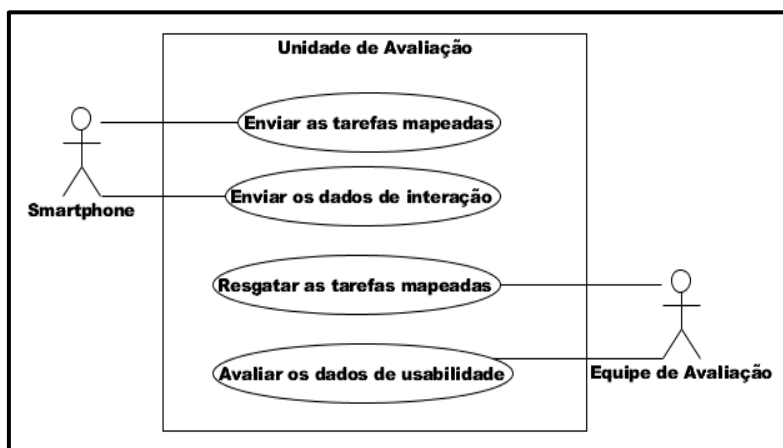


Figura 19 – Caso de Uso da Unidade de Avaliação

4.4 VISÃO DE IMPLANTAÇÃO

A visão de Implantação descreve os componentes do modelo posicionados em nós físicos de computação, permitindo avaliar a topologia e a forma de comunicação. No diagrama de Implantação apresentado na Figura 20, é possível identificar três ambientes distintos: o Ambiente de Desenvolvimento, o Ambiente de Interação e o Ambiente de Avaliação dos Dados.

As ações que possibilitam preparar o aplicativo para prover o mapeamento das tarefas e a inserção das métricas de rastreabilidade são realizadas no Ambiente de Desenvolvimento. Após os aplicativos serem gerados com estas funcionalidades, é necessário enviá-los para o *smartphone*. A forma de comunicação mais simples entre o Ambiente de Desenvolvimento e o Ambiente de Interação é via USB, embora existam outras formas de se fazer isso, como através uma rede local, um cliente Web ou *Bluetooth*.

O Ambiente de Interação é representado pelo dispositivo que irá executar o aplicativo, possibilitando o mapeamento das tarefas ou a captura dos dados de usabilidade, contextuais e subjetivos. Os dados das interações são enviados para um servidor de FTP instalado no Ambiente de Avaliação. Esta comunicação só ocorre quando o dispositivo está numa rede *Wi-Fi*, evitando que os dados sejam

transmitidos por outras redes que normalmente são tarifadas pelo provedor do serviço. É importante salientar que os dados não são perdidos quando não existir a disponibilidade de uma rede *Wi-Fi*, já que o modelo prevê que ficarão armazenados na memória local do *smartphone*.

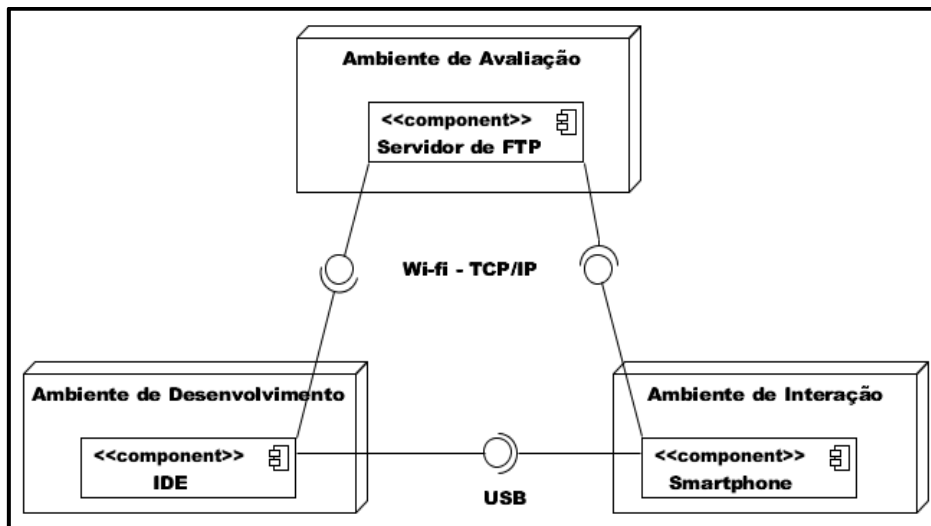


Figura 20 – Diagrama de Implantação do Modelo

Finalmente, no Ambiente de Avaliação, é prevista a existência de um servidor de FTP, que possibilite a transferência dos dados referente ao mapeamento das tarefas e das interações dos usuários finais. Como a Equipe de Avaliação necessita dos dados de mapeamento para gerar as métricas de rastreabilidade, é necessário que o servidor disponibilize esses dados para que sejam utilizados no Ambiente de Desenvolvimento. Esta comunicação é realizada via uma rede *Wi-Fi*.

4.5 VISÃO DE PROCESSOS

Um processo no âmbito deste trabalho é definido como sendo “um conjunto de atividades inter-relacionadas executadas para obter um conjunto específico de produtos, resultados ou serviços” (Guia PMBOK, 2008) ou ainda, como “uma sequência de passos realizados com um determinado objetivo” (IEEE, 2003).

Os processos foram divididos em três atividades principais que desencadeiam uma série de passos para possibilitar o mapeamento das tarefas, inserir as métricas de rastreabilidade e preparar o ambiente de avaliação dos dados. A Figura 21 apresenta o fluxo e a ordem de execução destas atividades.



Figura 21 – Principais processos presentes no Modelo Arquitetural

O primeiro processo do fluxo é Gerar Aspectos de Mapeamento. Ele é responsável pela criação de arquivos contendo código orientado a Aspectos que possibilita o mapeamento de atividades executadas em um determinado aplicativo. Neste processo, são identificados os métodos executados durante a realização de uma tarefa no aplicativo que será avaliado.

O segundo processo define os passos para a Inserção de Métricas de Rastreabilidade que possibilitam capturar os dados contextuais, subjetivos e de usabilidade. O objetivo de coletar os dados relacionados às interações dos usuários finais no aplicativo é enviá-los para um servidor FTP para posterior análise.

O último processo, Preparação do Ambiente de Avaliação, corresponde à elaboração da infraestrutura de recepção de dados e posterior armazenamento em uma base de dados que futuramente possa ser analisada para a descoberta de conhecimento com relação à usabilidade da aplicação que será avaliada.

As tarefas de cada processo e o fluxo de execução serão detalhadas nas próximas subseções durante a descrição da Infraestrutura UXEProject.

4.6 A INFRAESTRUTURA UXEPROJECT

Para contemplar os requisitos descritos na especificação do modelo e validar as funcionalidades propostas, foi criada uma infraestrutura denominada de UXEProject. Nesta seção, os principais conceitos envolvidos na infraestrutura são descritos, assim como as ferramentas utilizadas para conceber os requisitos funcionais do modelo apresentado nas seções anteriores deste capítulo.

A descrição da infraestrutura UXEProject será apresentada de acordo com a visão de Implantação proposta na seção 4.4. Sendo assim, ela foi dividida em três ambientes: Desenvolvimento, Interação e Avaliação. Esses ambientes caracterizam fases distintas do processo de captura, interação e avaliação dos resultados de acordo com a proposta do modelo e requerem inúmeras ferramentas de apoio, as quais caracterizam, de maneira única, a infraestrutura UXEProject.

4.6.1 Ambiente de Desenvolvimento

Nesse ambiente, são tratados todos os aspectos de instrumentação do código fonte das aplicações que serão avaliadas. Ele está dividido em duas unidades: (i) Unidade de Mapeamento e Unidade de Rastreabilidade, que retratam dois dos três componentes principais do modelo.

Um ponto importante a ser destacado, é que as unidades previstas neste ambiente utilizam a POA (Kiczales, 1997) para incorporar funcionalidades definidas no modelo, sem alterar diretamente os códigos fontes das aplicações, conforme as recomendações durante a especificação arquitetural do modelo.

4.6.1.1 Unidade de Mapeamento

Essa unidade é subdividida em dois módulos que são responsáveis em prover todas as funcionalidades para o mapeamento das tarefas de uma aplicação. O mapeamento das tarefas é obtido através da coleta dos métodos que são executados na aplicação durante a interação da Equipe de Avaliação.

Além disso, a Unidade de Mapeamento prevê que sejam informados valores de referência para a execução da tarefa como, por exemplo, tempo médio para completá-la, número de ações necessárias, número máximo de ações inconsistentes aceitáveis, entre outras.

No **Módulo de Incorporação dos Aspectos de Mapeamento das Tarefas**, é criada a lógica dos aspectos que permite guardar os passos executados pela Equipe de Avaliação no momento em que define como realiza uma tarefa. Para isso, foi criado um Aspecto que pode ser incorporado a qualquer aplicação. Este Aspecto identifica todos os métodos executados no caminho da tarefa e utiliza-se de um filtro para guardar apenas os métodos que tratam eventos do usuário, assim possibilitando que todos os passos necessários para concluir uma tarefa sejam capturados.

Para gerar o Aspecto de Mapeamento automaticamente, foi desenvolvida uma ferramenta chamada *Mapping Aspect Generator* (MAG). Na Figura 22, é possível visualizar a única tela da ferramenta na qual é disponibilizada uma caixa de texto para a Equipe de Avaliação informar o caminho onde está o código fonte do projeto. O restante do processo é realizado automaticamente pela ferramenta.

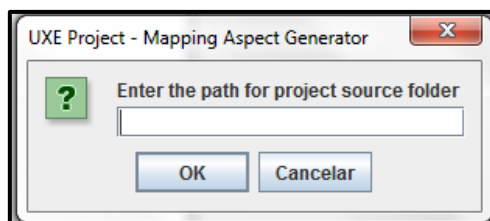


Figura 22 – A Ferramenta MAG

A Figura 23 apresenta o fluxo de ações necessário para preparar uma aplicação a ser mapeada usando a MAG.

Passo 1 – A ferramenta MAG importa o código fonte da aplicação a ser mapeado.

Passo 2 – A ferramenta percorre todas as classes e coloca em uma lista as classes que tratam interações de usuários.

Passo 3 – A MAG utiliza a lista de classes para gerar um Aspecto que adiciona o método *onUserInteraction*¹⁵ nas classes que ainda não o possuem e cria um filtro para detectar as interações dos usuários.

Passo 4 – A Equipe de Avaliação deve utilizar o AspectJ para juntar o Aspecto gerado com o código fonte da aplicação.

Passo 5 – O resultado obtido após o *Weaver* é o aplicativo pronto para possibilitar o mapeamento de suas tarefas.

Passo 6 – Para finalizar o processo, basta embarcar a aplicação em um dispositivo que permita realizar as interações.

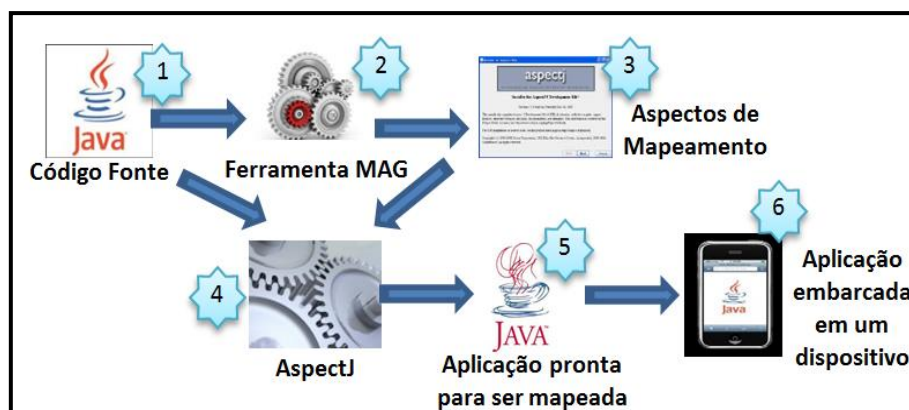


Figura 23 – Fluxo de ações para a inserção dos Aspectos de mapeamento

O ponto chave da ferramenta MAG é a geração do Aspecto que contém o filtro que identifica quando ocorre uma interação do usuário e qual classe, método e parâmetro da aplicação foram utilizados.

¹⁵ Maiores detalhes em [http://developer.android.com/reference/android/app/Activity.html#onUserInteraction\(\)](http://developer.android.com/reference/android/app/Activity.html#onUserInteraction())

Para identificar quando ocorre uma interação na plataforma Android, é necessário utilizar o método *onUserInteraction* das classes que representam atividades. O trecho de código da Figura 24 ilustra um *pointcut* e um *advice* para capturar uma ação do usuário e atribui à variável de controle o valor verdadeiro.

```
pointcut userInteractionPointCut() : execution(void *.onUserInteraction());  
before() : userInteractionPointCut()  
{  
    userInteraction = true;  
}
```

Figura 24 – *PointCut* e *Advice* para identificar a interação do usuário

No **Módulo de Definição das Tarefas**, é prevista a possibilidade da Equipe de Avaliação interagir com o aplicativo para mapear a sua forma de execução das tarefas. Essa ação possibilita que os métodos executados no aplicativo sejam capturados, sendo possível descrever automaticamente os passos para realizar uma tarefa independente de formalismos. Além disso, esse módulo prevê que a Equipe de Avaliação forneça valores aceitáveis em termos de usabilidade para a realização da tarefa.

Para realizar esse processo, foi desenvolvida a ferramenta *Automatic Task Description* (ATD). A ATD deve ser embarcada em um dispositivo e ser executada simultaneamente com a aplicação a ser mapeada. Assim, à medida que a Equipe de Avaliação interage com a aplicação, os métodos executados são capturados de forma automática para serem utilizados como passos para a conclusão de uma tarefa.

A base da ferramenta ATD consiste na utilização de um Aspecto que identifica quando ocorre uma interação do usuário com uma aplicação que foi preparada para ser mapeada, ou seja, a ferramenta ATD captura a execução do método *onUserInteraction* e identifica quais classes, métodos e parâmetros da aplicação foram utilizados. A janela principal da ferramenta pode ser contemplada na Figura 25.

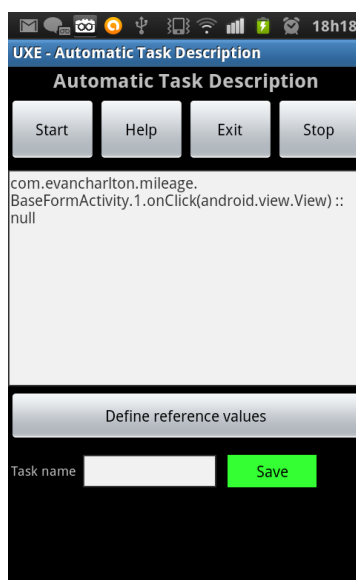


Figura 25 – Janela principal da Ferramenta ATD

Os botões superiores possibilitam ao Engenheiro marcar o início e o final do mapeamento, além das funções correspondentes a ajuda e abandono da tarefa. Os métodos mapeados serão apresentados na parte central e, na parte inferior, a ferramenta disponibiliza uma caixa de texto para a digitação do nome da tarefa que foi mapeada e o botão para realizar a gravação dos dados referentes a ela.

O botão *Define reference values* abre uma nova janela para o lançamento das informações referentes à usabilidade da tarefa, como pode ser observado na Figura 26.

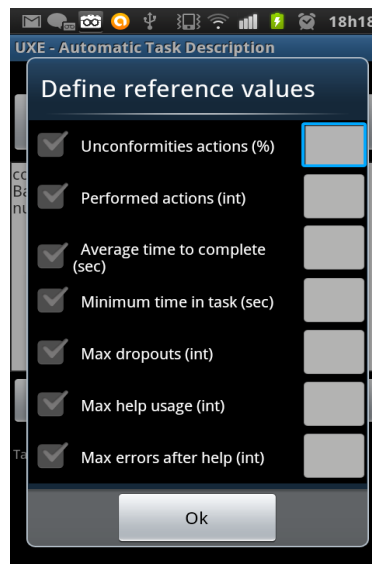


Figura 26 – Janela para definição dos valores de referência

O resultado após o mapeamento de uma tarefa é a criação de um arquivo XML que armazena: (i) todos os métodos que fazem parte de um percurso para a execução da tarefa; (ii) o método que indica ajuda; e (iii) o método que indica abandono da tarefa. Também são armazenados os valores de referência que são estimativas propostas pela Equipe de Avaliação, tais como, o número de ações, o tempo para a execução, o número de erros tolerados para uma tarefa, entre outros.

Depois de gerado, o arquivo XML é enviado para o servidor de FTP existente na Unidade de Avaliação, onde os métodos que representam a tarefa são utilizados na criação das métricas de rastreabilidade. Além disso, os valores de referência são armazenados no BD para serem utilizados posteriormente como parâmetros de comparação com o desempenho real dos usuários.

contexto, (ii) Aspecto com as métricas subjetivas, e (iii) Aspecto com o controle de transmissão e persistência dos dados.

As Figuras 28, 29 e 30 apresentam as janelas da ferramenta *UXE Metrics Generation* e possibilitam identificar os passos necessários para a geração do Aspecto que conterá as métricas de rastreabilidade.

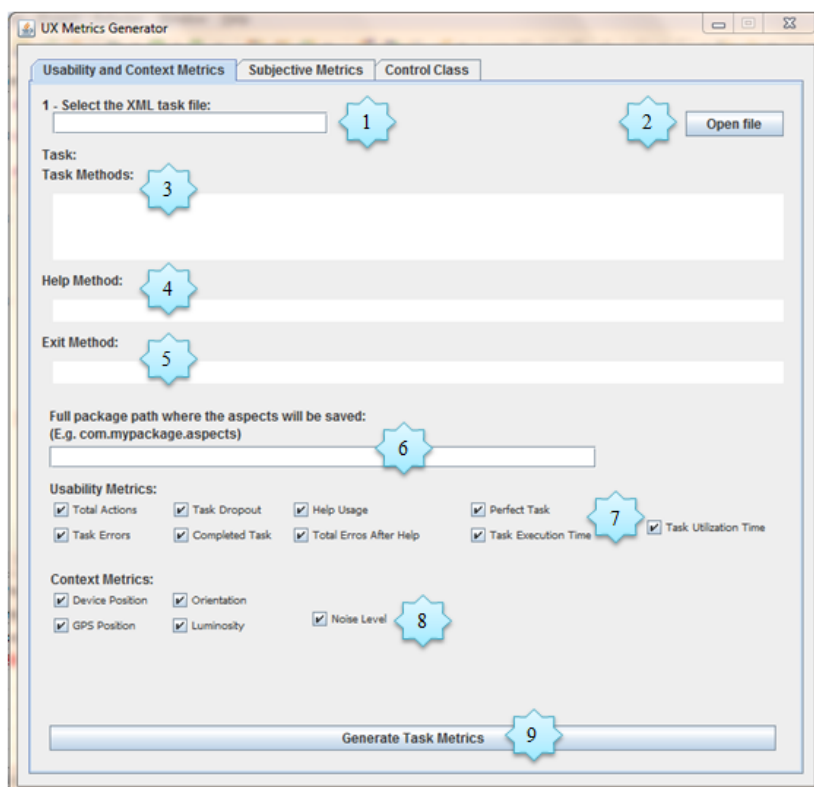


Figura 28 – *UXE Metrics Generation* – Métricas de Usabilidade e Contexto

A primeira caixa de texto (passo 1) possibilita informar o nome do arquivo XML com a definição dos métodos relacionados à tarefa que será instrumentada. No botão *Open File* (passo 2), é possível abrir a tarefa e seus métodos mapeados serão exibidos nas caixas de texto seguintes (passos 3, 4 e 5). No passo 6, deve ser especificado o caminho do pacote onde deve ser salvo o Aspecto que será gerado. Nos passos 7 e 8, é possível escolher quais métricas de usabilidade e contexto serão utilizadas. Finalmente, o passo 9 disponibiliza o botão para gerar e salvar o arquivo contendo as métricas criadas com Orientação a Aspectos. Um exemplo do Aspecto gerado pode ser observado na sua íntegra no Anexo IV.

As questões subjetivas, ao contrário das métricas de usabilidade e contexto, não estão vinculadas a tarefas, mas a aplicação com um todo, logo, é necessário definir o identificador da aplicação (Figura 29 – passo 1), descrever a pergunta que será proposta para o usuário (passo 2), especificar o caminho do pacote onde o Aspecto deve ser salvo (passo 3) e, finalmente, gerar e salvar o Aspecto (passo 4). Um exemplo do Aspecto gerado pode ser observado no Anexo V.

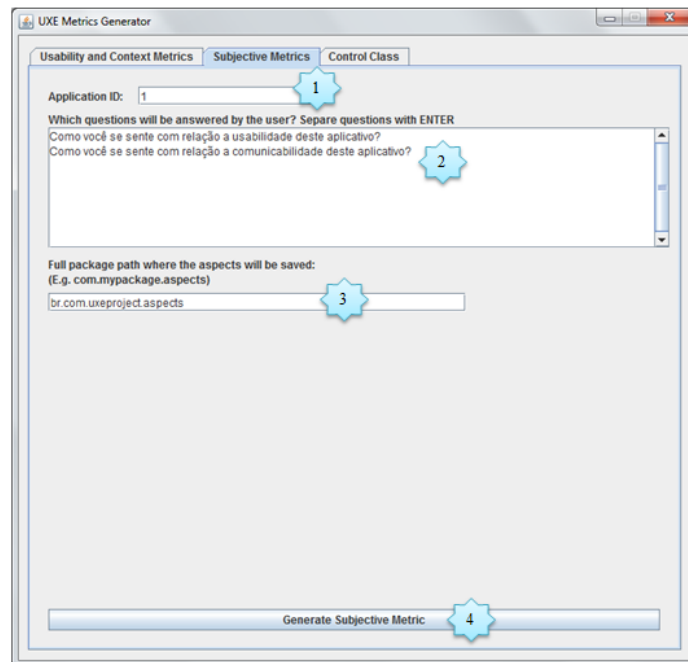


Figura 29 – UXE Metrics Generation – Métricas Subjetivas

A última tela da aplicação (Figura 30) disponibiliza a especificação das configurações necessárias para acessar um servidor FTP (passo 1). Esse serviço será necessário para a transferência dos dados coletados por intermédio das métricas e questionários subjetivos instrumentados na aplicação que será avaliada. No passo 2, é definido o caminho do pacote relacionado à aplicação e, no passo 3, o código será gerado e gravado.

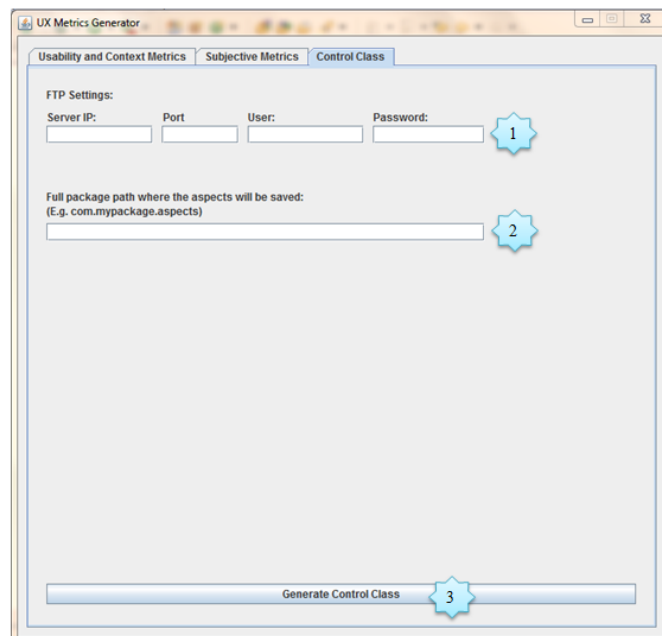


Figura 30 – UXE Metrics Generation – Controle

No **Módulo de Incorporação dos Aspectos ao Código Fonte**, é realizado o *Weaver* do código fonte do aplicativo com os Aspectos gerados no módulo anterior, resultando na instrumentação do aplicativo para ser rastreado durante a interação do Usuário Final.

Na Figura 31, são apresentadas as etapas necessárias para a geração do aplicativo pronto para ser rastreado. A etapa 1 representa as métricas de rastreabilidade, subjetivas e de controle. A etapa 2 apresenta a aplicação que será instrumentada para posterior avaliação. Na etapa 3, os aspectos contendo as métricas de rastreabilidade e subjetividade devem ser incorporados ao projeto inicial da aplicação. Posteriormente, deve ser compilado o novo projeto com o AspectJ. O resultado é a geração da aplicação pronta para ser avaliada, ou seja, com as métricas que serão utilizadas para a captura dos dados de usabilidade, contexto e subjetividade.

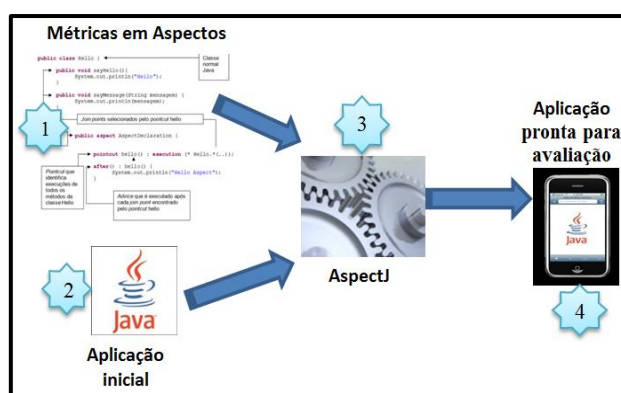


Figura 31 – Geração da aplicação com as métricas de Rastreabilidade

4.6.2 Ambiente de Interação

O foco deste ambiente são as interações da Equipe de Avaliação e dos Usuários Finais. A Equipe de Avaliação interage com o aplicativo para mapear as tarefas (passos 1 a 3 da Figura 32). O resultado dessa interação é a criação de um arquivo XML (Figura 33) com os métodos referentes à execução da tarefa e alguns dados de usabilidade sob a ótica da Equipe de Avaliação. Este arquivo será usado para dar continuidade à instrumentação do código fonte da aplicação, bem como funcionar como uma das entradas de dados para o Ambiente de Avaliação.

Por outro lado, o Ambiente de Interação também prevê a interação do Usuário Final. É nesse momento que as interações do usuário serão rastreadas, com geração de dados que poderão ser interpretados posteriormente no Ambiente de Avaliação (passos 4 a 6 da Figura 32).



Figura 32 – Execuções previstas no Ambiente de Interação.

Na Figura 33, é apresentada a estrutura de um arquivo XML produzida após o mapeamento de uma tarefa. Os principais elementos que caracterizam o arquivo são os seguintes:

`<task>` contém os atributos (*appInfo*) que indica o identificador do aplicativo e (*name*) que guarda o nome da tarefa.

`<referenceValues>` serve para guardar os valores de referência de usabilidade para a tarefa. Este elemento contém uma lista de valores de referência com dois atributos: (*reference*) que indica o nome desse valor e (*value*) que guarda o valor.

`<methods>` guardam os métodos executados para realizar a tarefa. Este elemento contém um atributo (*class*) usado pela biblioteca *Java Architecture for XML Binding* (JAXB) para converter um objeto em um formato XML. Além disso, possui um elemento `<method>` formado por outros dois elementos que guardam a assinatura e os parâmetros dos métodos mapeados.

`<help>` armazena o método relacionado com a ajuda, possui dois elementos usados para guardar a assinatura e os parâmetros dos métodos mapeados.

`<back>` armazena o método relacionado com o retorno de uma ação executada e possui a mesma estrutura do elemento `<help>`.

```

<task appInfo="app.top.level.package-versionCode" name="taskName" id="0">
  <referenceValues class="java.util.ArrayList">
    <referenceValue reference="UNCONFORMITIES" value="Unconformities actions (8)"/>
    <referenceValue reference="PERFORMED_ACTIONS" value="Performed actions (int)"/>
    <referenceValue reference="AVERAGE_COMPLETE_TIME" value="Average time to complete (seconds)"/>
    <referenceValue reference="MIN_UTILIZATION_TIME" value="Minimum time in task (sec)"/>
    <referenceValue reference="DROPOUTS" value="Max dropouts (int)"/>
    <referenceValue reference="MAX_HELP" value="Max help usage (int)"/>
    <referenceValue reference="ERRORS_AFTER_HELP" value="Max errors after help (int)"/>
  </referenceValues>
  <methods class="java.util.ArrayList">
    <method>
      <signature>package.for.the.Class.method(complete.className.Attribute1 ...)</signature>
      <parameters>attr11=value11&attr12=value12&attr13=value13&attr21=value21</parameters>
    </method>
    <method>
      <signature>package.for.the.Class.method2(complete.className.Attribute0)</signature>
      <parameters>attr01=value01</parameters>
    </method>
  </methods>
  <help>
    <method>
      <signature>package.for.HelpClass.method()</signature>
      <parameters></parameters>
    </method>
  </help>
  <back>
    <method>
      <signature>package.for.BackClass.method()</signature>
      <parameters></parameters>
    </method>
  </back>
</task>

```

Figura 33 – Estrutura do XML gerado durante o mapeamento das tarefas

Dois arquivos XML são gerados a partir das interações do usuário final. O primeiro contém as métricas de usabilidade e contexto. O segundo contém as respostas dadas pelo usuário referentes às perguntas promovidas, de tempos em tempos, durante as interações do usuário com o aplicativo.

A Figura 34 apresenta a estrutura de um arquivo XML gerado durante a execução de uma tarefa. No arquivo, o elemento *<tasks IDUser>* é responsável por armazenar o identificador do aparelho usado na execução da tarefa; *<task name>*, por guardar o nome da tarefa; *<timestamp>*, o horário da realização da tarefa; *<metrics>*, as métricas de usabilidade utilizadas e; *<context>*, por armazenar as informações contextuais (*<luminosity>*, *<movements>* e *<motion>*).

```

<?xml version="1.0" encoding="UTF-8" ?>
<tasks IDUser="1">
  <task name="" timestamp="">
    <metrics>
      <totalActions> </totalActions>
      <taskComplete> </taskComplete>
      <totalError> </totalError>
      <taskCompletedPerfect> </taskCompletedPerfect>
      <executionTime> </executionTime>
      <utilizationTime> </utilizationTime>
      <helpUsed> </helpUsed>
      <errorsAfterHelp> </errorsAfterHelp>
    </metrics>
    <context>
      <luminosity>
        <quantity> </quantity>
      </luminosity>
      <movements>
        <position X="" Y="" Z=""/>
      </movements>
      <motion>
        <location Latitud="" Longitud=""/>
      </motion>
    </context>
  </task>
</tasks>

```

Figura 34 – Estrutura do XML contendo as métricas de usabilidade e contexto.

A estrutura do arquivo XML que armazena as informações referentes às perguntas subjetivas é apresentada na Figura 35. O elemento *<subjectiveMetric>* contém os atributos: (*question*) que contém a descrição da pergunta; (*IDUser*) que corresponde ao ID do telefone (*International Mobile Equipment Identity* – IMEI); e o (*appName*) que indica o ID da aplicação testada. O elemento *<subjectiveData>* contém o conjunto de valores informado pelo usuário representado pelo atributo *<rating>* que varia de 1 (completamente insatisfeito) até 5 (completamente satisfeito) e o atributo (*timestamp*) que indica o horário exato de cada avaliação.

```

<?xml version="1.0" encoding="UTF-8" ?>
<subjectiveMetric question="" IDUser="" appName="">
  <subjectiveData rating="" timestamp=""/>
</subjectiveMetric>

```

Figura 35 – Estrutura do XML contendo as respostas subjetivas.

Todos os arquivos XML criados a partir das interações apresentadas na Figura 32 são enviados através do protocolo FTP para um servidor disponível na Internet. As transferências ocorrerem em duas situações distintas:

1) No momento do mapeamento das tarefas (Equipe de Avaliação). Para isso, são necessárias as seguintes premissas:

- O dispositivo possuir uma aplicação instrumentada com os Aspectos de mapeamento e possuir a ferramenta ATD.

- O aparelho ter acesso a uma rede Wi-Fi (padrão IEEE 802.11).
- Existirem arquivos XML de mapeamento que não foram enviados para o servidor.

2) No momento da captura das interações do Usuário Final. Para isso são necessárias as seguintes premissas:

- O dispositivo possuir uma aplicação instrumentada com as métricas de usabilidade, contexto e subjetivas.

- O aparelho ter acesso a uma rede Wi-Fi (padrão IEEE 802.11).
- Existirem arquivos XML com métricas que não foram enviados para o servidor.

A Figura 36 ilustra alguns cenários nos quais podem ocorrer as interações e transferências dos arquivos XML. É importante destacar que as capturas de contextos estão vinculadas aos sensores disponíveis nos aparelhos utilizados. Assim, é possível identificar se o usuário está ou não em movimento, qual a posição do dispositivo (horizontal, vertical e mista), a velocidade de deslocamento e a luminosidade do ambiente durante as suas interações.



Figura 36 – Cenários do Ambiente de Interação.

Como a análise dos dados pode levar em consideração o cenário de utilização da aplicação, podem ser incorporadas métricas específicas como, por exemplo, relacionadas ao GPS do *smartphone*, permitindo correlacionar as interações de um usuário com informações relativas ao seu posicionamento, velocidade e percurso realizado. Neste caso, a análise da usabilidade pode levar em conta não somente como o usuário realiza uma tarefa, mas também como o ambiente influencia suas interações.

4.6.3 Ambiente de Avaliação

Para contemplar os componentes definidos no Ambiente de Avaliação, foram realizados os seguintes processos: (i) criar e configurar um servidor de FTP e de BD e disponibilizá-los na Internet; (ii) realizar a modelagem de um BD e um DW para armazenar e possibilitar as análises das informações capturadas durante os experimentos; (iii) criar ferramentas para detectar a presença de novos arquivos no servidor de FTP, popular o BD e fazer a carga do DW; e (iv) escolher uma ferramenta OLAP para apoiar as análises dos dados oriundos da captura.

Para que os dados referentes aos experimentos pudessem ser transmitidos e armazenados em um BD, foi contratado junto à empresa Amazon uma microinstância do serviço conhecido como *Amazon Elastic Compute Cloud (Amazon EC2)*, em consonância com a tendência recente na área de armazenamento de dados utilizando a computação na nuvem (Zhang et al., 2010). Este ambiente disponibiliza o sistema operacional *Ubuntu* versão 11.10 *Oneiric Server*, memória RAM de 613 Mbyte e um espaço para armazenamento de dados com 18 Gbytes.

Para contemplar o servidor de FTP, foi instalado o software *ProFTPD* versão 1.3.4 com a finalidade de gerenciar e controlar a transferência de arquivos. Além disso, foi instalado o servidor Apache versão 2.2.9, para disponibilizar um site Web¹⁶ descrevendo a pesquisa e possibilitando que os usuários possam informar o seu perfil e os dados do seu *smartphone* no momento de realizar o *download* dos aplicativos instrumentados com as métricas de rastreabilidade. Neste site também está disponível o termo de uso dos aplicativos que relata aos usuários a sua participação em um experimento de usabilidade (ver Anexo VIII).

O SGBD escolhido para armazenar os dados foi o *MySQL Community Server*, versão 5.1.58. Com a finalidade de facilitar as avaliações das informações coletadas nos experimentos, foi modelado um BD e um DW, que são analisados com a ferramenta OLAP *Pentaho Analysis Services*, versão 3.9.0.

Para contemplar a carga dos dados no BD, foi desenvolvida uma ferramenta denominada de *DataLoad*, programada para executar a cada 10 minutos. Os passos executados por esta ferramenta são: detectar a chegada de novos arquivos no servidor de FTP, extrair os dados e carregá-los no BD.

A última ferramenta deste projeto realiza a extração, transformação e carga dos dados (Ferramenta ETL Maker), transferindo-os do BD para o DW. Esta ferramenta executa uma vez por dia e realiza a carga no DW de forma incremental.

¹⁶ Disponível em <http://uxeproject.no-ip.org>

A Figura 37 tem o propósito de ilustrar o fluxo de ações realizado na Unidade de Avaliação, contemplando a chegada do arquivo XML com os dados capturados (passo 1), carga no BD (passo 2), armazenamento dos dados (passo 3), ETL (*Extract Transform Load*) (passo 4), armazenamento no DW (passo 5) e análise dos dados com a plataforma *Pentaho* (passo 6).

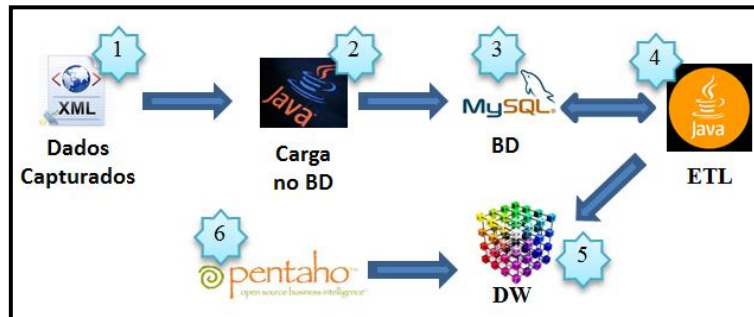


Figura 37 – Fluxo de ações na Unidade de Avaliação

Na Figura 38, são apresentadas as tabelas que fazem parte do BD. Através do modelo de Entidade e Relacionamento (ER) são relacionados os atributos que compõem as tabelas. As descrições das tabelas e dos campos que fazem parte do BD estão disponíveis no dicionário de dados apresentado no Anexo VI.

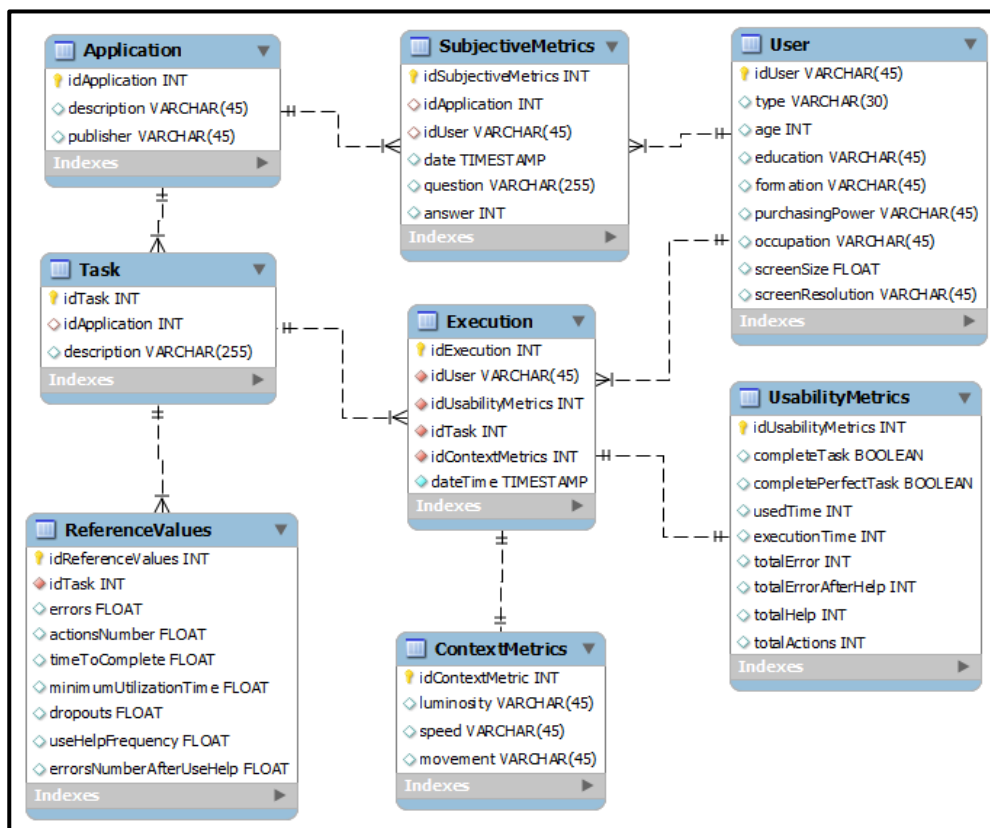


Figura 38 – Modelagem do Banco de Dados.

Todas as análises propostas neste trabalho tomam como base o DW ilustrado na Figura 39. Os fatos disponibilizados, relacionados à execução das tarefas a partir das aplicações, podem ser visualizados a partir de cinco dimensões: Tarefa, Tempo, Aplicação, Contexto e Usuário. Estas dimensões possibilitam diferentes formas de análise como, por exemplo: verificar a usabilidade dos aplicativos e relacionar suas interações ao contexto de uso. O dicionário de dados do DW está disponível no Anexo VII.

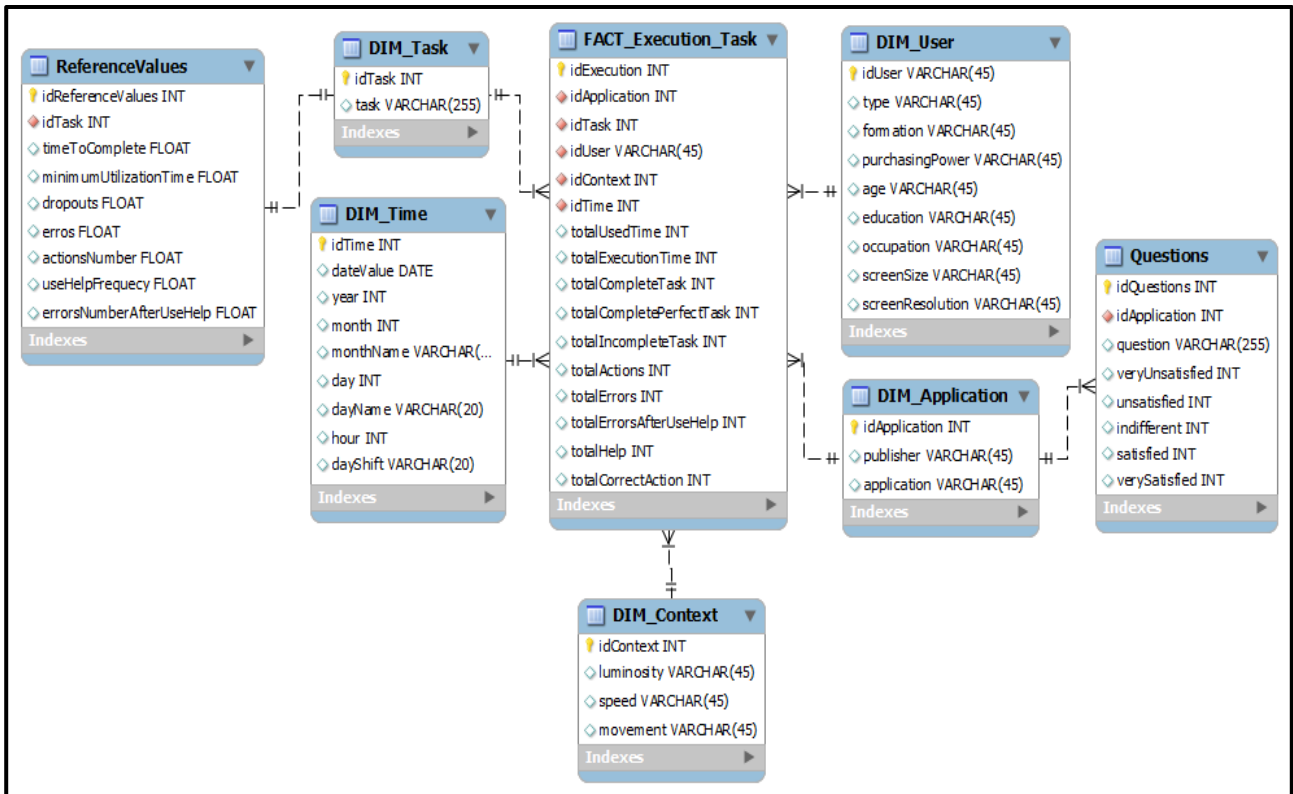


Figura 39 – Modelagem do Data Warehouse

A tabela FACT_Execution_Task é responsável por quantificar as métricas de usabilidade, previstas na Biblioteca de Métricas da infraestrutura. Nesta tabela, são encontrados valores para guardar, por exemplo, o tempo necessário para completar uma tarefa, a quantidade de vezes que uma tarefa foi executada, o total de ações para executar a tarefa, o total de erros e ações corretas, o total de acesso à ajuda e à quantidade de erros após o acesso a ajuda.

A dimensão DIM_Task registra todas as tarefas mapeadas para análise e sua relação com a tabela de fatos permite, por exemplo, avaliar quais tarefas apresentam mais dificuldade para os usuários executá-las.

Para acompanhar as experiências dos usuários ao longo do tempo, é utilizada a dimensão DIM_Time. Por exemplo, pode ser avaliada a frequência com que um usuário executa uma tarefa.

A dimensão DIM_Application, ao relacionar-se com a tabela de fatos, proporciona a obtenção da visão estatística das interações a nível de aplicação como, por exemplo, identificar o total de tarefas completadas com sucesso por aplicativo. A esta dimensão está associada à tabela de questões que são disponibilizadas aos usuários, com a finalidade de identificar sua satisfação em relação a determinados aspectos dos aplicativos.

Para relacionar as interações dos usuários ao contexto em que são realizadas, é utilizada a dimensão DIM_Context. Esta dimensão armazena as informações dos sensores que capturam a luminosidade, a posição do aparelho e a movimentação do usuário.

A caracterização dos usuários está na dimensão DIM_User que contém o perfil dos usuários e as características dos dispositivos utilizados nas interações.

A partir das avaliações realizadas no DW, é possível realizar inferências sobre os dados capturados com a finalidade de obter informações a respeito da usabilidade dos aplicativos, envolvendo a eficiência, eficácia, satisfação, aprendizagem, operabilidade, acessibilidade, flexibilidade, utilidade e facilidade de uso. No Capítulo 5, serão apresentados os resultados das análises realizadas a partir de um experimento conduzido para comprovar as potencialidades da infraestrutura criada.

4.7 FORMAS DE CAPTURA DE DADOS

Para que as interações dos usuários possam ser rastreadas durante a utilização de um aplicativo, são necessárias definições de métricas capazes de quantificar as ações dos usuários, capturar informações providas pelos sensores do dispositivo utilizado e guardar as respostas subjetivas fornecidas diretamente pelo usuário. Além disso, informações relacionadas ao perfil do usuário e características do aparelho utilizado são adquiridas no momento que o Usuário Final faz *download* das Aplicações para o seu *smartphone*.

As métricas disponíveis no modelo proposto nesse trabalho podem ser categorizadas em três grupos distintos: métricas de usabilidade, métricas de contexto e métricas subjetivas.

As **métricas de usabilidade** consideradas na infraestrutura UXEProject são similares àquelas propostas inicialmente por Nielsen (Nielsen, 1994), e mais recentemente adaptadas para diferentes tipos de plataformas (Moldovan e Tarta, 2006). Elas foram divididas em quatro grupos distintos:

- Métricas para captura da assertividade na utilização da aplicação.
- Métricas relacionadas à busca e utilização de tarefas.
- Métricas relacionadas com o desempenho na utilização da aplicação.

- Métricas relacionadas à utilização da ajuda.

A Tabela 15 apresentada cada uma das métricas e seus objetivos. Vale salientar que a infraestrutura proposta, embora tenha como objetivo mensurar a usabilidade de uma aplicação com base nas métricas descritas, não está restrita a um único conjunto de métricas. Novas métricas podem ser incorporadas ao modelo de acordo com o interesse do avaliador e necessidades de contextualização à aplicação. Isto permite a extensão do modelo para incluir outras métricas de usabilidade como, por exemplo, as métricas criadas a partir do padrão ISO 9241-11 (1998), Zhang e Adipat (2005) e Coursaris e Kim (2011).

Tabela 15 – Métricas de Usabilidade

Métricas de Usabilidade	Objetivos
Métricas para captura da assertividade	
Ações em conformidade	Quantificar a dificuldade de interação do usuário ao executar uma tarefa.
Ações sem conformidade	
Métricas relacionadas à busca e utilização de tarefas	
Desistências de tarefas	A desistência pode significar que o usuário está perdido ou à procura da tarefa desejada.
Tarefas concluídas	
Tarefas concluídas e utilizadas	A utilização ou não das tarefas indica se o usuário encontrou o que estava procurando.
Tarefas concluídas e não utilizadas	
Métricas relacionadas ao desempenho	
Tempo médio para completar uma tarefa	Indicar necessidades de melhorias que minimizem o tempo e ações para a execução de tarefas.
Número médio de ações para completar a tarefa	
Número de tarefas concluídas com erros no caminho	
Métricas relacionadas à utilização de ajuda	
Frequência de utilização da ajuda	Mensurar o nível de dificuldade enfrentada pelo usuário e identificar se a ajuda disponível proporciona o efeito esperado.
Inconformidades cometidas após o acionamento da ajuda	

As **métricas de contexto** propiciam relacionar a execução de tarefas com o cenário em que a interação está ocorrendo. A infraestrutura UXEProject contempla este requisito proposto no modelo de duas formas:

1) Por meio dos sensores dos *smartphones* para coletar informações como, por exemplo, a intensidade da luz, a movimentação do usuário, a velocidade de deslocamento e a posição do aparelho (horizontal, vertical e mista). Para isso, basta que o aparelho tenha sensores disponíveis que possam ser utilizados para obter os dados anteriormente citados.

Quando o Usuário Final faz *download* do aplicativo, é necessário que preencha um formulário informando o seu perfil e alguns dados referentes ao dispositivo que irá utilizar.

Na Figura 40, é possível observar as três dimensões contextuais consideradas (Ambiente, Usuário e Dispositivo) e as variáveis contextuais associadas a cada dimensão. Entretanto, é importante

ressaltar que outras dimensões e variáveis podem ser associadas à infraestrutura, porém será necessário estender a Biblioteca de Métricas.

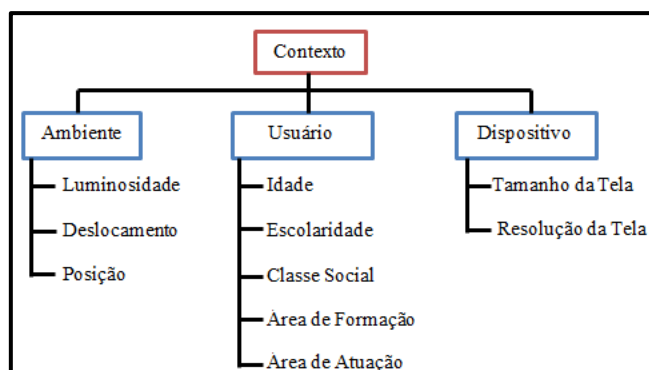


Figura 40 – Variáveis contextuais consideradas na Infraestrutura UXEProject

Na Tabela 16, são listados os sensores que foram instrumentados na infraestrutura, entretanto outros sensores podem ser utilizados para a captura de variáveis contextuais. Para isso, basta estender o conjunto de métricas relacionadas aos sensores, disponível na Biblioteca de Métricas.

Tabela 16 – Sensores implementados na infraestrutura UXEProject

Sensores	Objetivos
Acelerômetro	Permitir a identificação da posição do aparelho (vertical ou horizontal) e se ocorreram mudanças durante a execução de uma tarefa.
GPS	É utilizado para identificar se ocorreram deslocamentos durante as interações do usuário e em qual velocidade.
Sensor de Luminosidade	Permite a captura da intensidade de luminosidade do ambiente.

As **métricas subjetivas** são utilizadas para medir o estado emocional dos usuários ao longo de suas experiências com um aplicativo. A técnica ESM foi utilizada para apresentar perguntas ao usuário e, através das respostas, possibilita medir duas dimensões: o tipo de emoção (positiva ou negativa) e a intensidade da emoção. As perguntas são apresentadas ao usuário de tempos em tempos, de forma aleatória, e estão apenas atreladas ao aplicativo e não às tarefas realizadas e/ou às informações contextuais. Esta decisão foi tomada para propiciar interações com o mínimo de interferência possível, tornar o mecanismo genérico para abranger qualquer aplicativo e simplificar o uso da técnica dentro da abordagem proposta.

Além dos inúmeros trabalhos que utilizam a técnica ESM, citados na Subseção 3.3.5.2, a sua adoção foi em função dos seguintes aspectos (Meschtscherjakov, 2009):

- É apropriada para ser apresentada em dispositivos com tela relativamente pequena.
- É intuitiva, não precisa de muito esforço mental para ser interpretada.
- É capaz de ser respondida usando diferentes modalidades de entrada fornecidas pelos vários modelos de *smartphones*.

4.8 RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO

Neste capítulo, foi descrito o modelo arquitetural para avaliação de usabilidade proposto nesta tese e a infraestrutura criada para contemplar os seus requisitos. Inicialmente, o modelo foi apresentado de acordo com a metodologia 4+1, onde foram utilizadas as visões de Desenvolvimento, Lógica, Casos de Uso, Implementação e Processos.

A infraestrutura UXEProject, construída para materializar o modelo arquitetural, foi dividida em três ambientes conforme estruturado na visão de Implementação:

- No **Ambiente de Desenvolvimento**, com a ferramenta MAG, são gerados os aspectos de mapeamento para habilitar as aplicações a serem mapeadas. Posteriormente, com a utilização da ferramenta ATD, a Equipe de Avaliação realiza o mapeamento das tarefas. A Figura 41 mostra os passos deste processo e a Tabela 17 descreve resumidamente cada um deles.

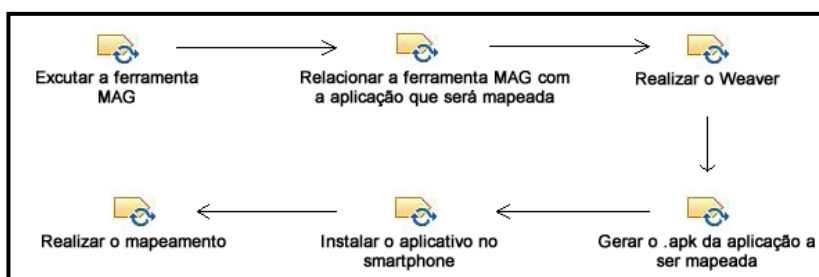


Figura 41 – Etapas para executar o processo Gerar Aspectos de Mapeamento

Tabela 17 – Descrição das tarefas para executar o processo Gerar Aspectos de Mapeamento

Tarefa	Objetivo
Executar a ferramenta MAG	Identificar automaticamente as classes de interação e inserir o método <i>onUserInteraction</i> , possibilitando detectar os métodos que são executados na aplicação durante as interações da equipe de IHC.
Relacionar a ferramenta MAG com a aplicação que será mapeada	Informar na ferramenta MAG o caminho da aplicação que será mapeada.
Realizar o <i>Weaver</i>	Relacionar os Aspectos gerados ao código-fonte da aplicação a ser mapeada.
Gerar o arquivo .apk da aplicação a ser mapeada	Preparar o aplicativo para ser embarcado no <i>smartphone</i> .
Instalar o aplicativo no <i>smartphone</i>	Possibilitar que o aplicativo seja utilizado pela equipe de IHC ao realizar o mapeamento das tarefas.
Realizar o mapeamento	Executar a ferramenta ATD em paralelo à aplicação que será avaliada, capturando os métodos executados pelo usuário ao efetuar determinada tarefa.

Ainda no Ambiente de Desenvolvimento, são gerados automaticamente os Aspectos de rastreabilidade, utilizando a ferramenta UXE Metrics Generation. O resumo do fluxo de execução deste processo é apresentado na Figura 42 e descrito na Tabela 18.

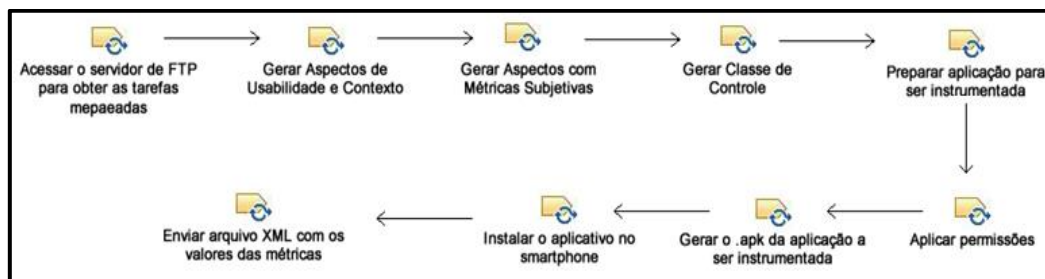


Figura 42 – Tarefas executadas no processo Inserir Métricas de Rastreabilidade

Tabela 18 – Descrição das tarefas para executar o processo Inserir Métricas de Rastreabilidade

Tarefa	Objetivo
Acessar servidor de FTP para obter as tarefas mapeadas	Acessar o servidor FTP para obter os arquivos XML que contêm o mapeamento das tarefas a serem rastreadas.
Gerar Aspectos de Usabilidade e Contexto	Gerar os Aspectos contendo as métricas de usabilidade e de contexto, utilizando a Ferramenta <i>UXE Metrics Generator</i> .
Gerar Aspectos com Métricas Subjetivas	Gerar os Aspectos contendo métricas subjetivas representadas por perguntas a respeito da opinião do usuário com relação ao aplicativo.
Gerar Classe de Controle	Armazenar os dados de acesso ao servidor FTP, tais como: endereço do servidor, porta, usuário e senha.
Preparar aplicação para ser instrumentada	Introduzir no aplicativo os Aspectos com as métricas de rastreabilidade e a classe de controle, habilitando o aplicativo a capturar e transmitir os dados durante as interações do usuário.
Aplicar permissões	Aplicar permissões no arquivo <i>AndroidManifest.xml</i> do projeto Android da aplicação que será rastreada para que se obtenha acesso a Internet.
Gerar o arquivo .apk da aplicação a ser instrumentada	Gerar o arquivo .apk que será instalado no <i>smartphone</i> do Usuário Final.
Instalar o aplicativo no <i>smartphone</i>	Instalar o arquivo .apk no <i>smartphone</i> do Usuário Final, propiciando a captura dos dados subjetivos, contextuais e de usabilidade.
Enviar arquivo XML com os valores das métricas	Enviar para o servidor FTP os dados capturados durante as interações do Usuário Final.

Com a utilização das ferramentas MAG, ATD e *UXE Metrics Generator*, todos os processos de geração de Aspectos referentes à preparação das aplicações para as avaliações são realizados de forma interativa e automática, sem a necessidade da Equipe de Avaliação escrever linhas de códigos.

- No Ambiente de Interação ocorrem as execuções da aplicação no momento de realizar o mapeamento das tarefas e durante as interações do Usuário Final. Em ambas as situações, são gerados arquivos XML contendo os dados que são utilizados nos processos que compreendem a abordagem proposta nesta tese.

• O Ambiente de Avaliação foi estruturado utilizando a computação na nuvem, onde todas as ferramentas necessárias para a recepção, armazenamento e análise dos dados foram instaladas. A Figura 43 demonstra o fluxo de execução das tarefas deste processo e a Tabela 19 descreve cada um deles.

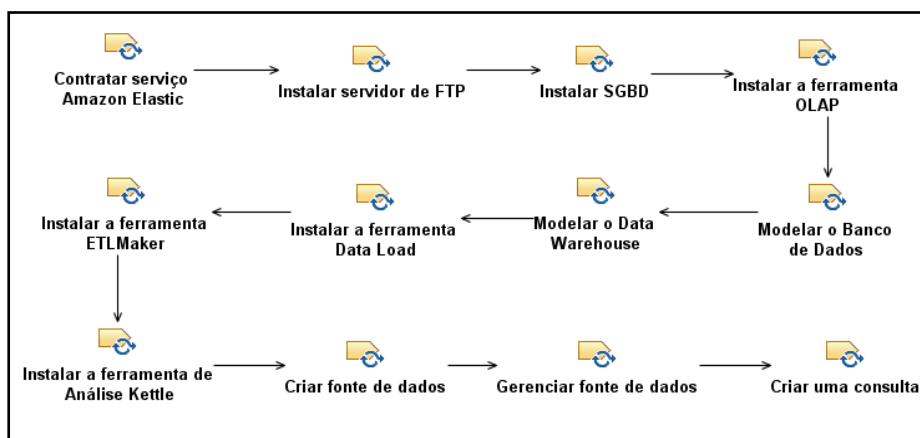


Figura 43 – Tarefas executadas no processo Preparar Ambiente de Avaliação

Tabela 19 – Descrição das tarefas executadas no processo Preparar Ambiente de Avaliação

Tarefa	Objetivo
Contratar Serviço <i>Amazon Elastic</i>	Disponibilizar um ambiente de computação virtual que permita a utilização de serviços web e acesso remoto, possibilitando a instalação e configuração de ferramentas para o armazenamento e análise dos dados de usabilidade.
Instalar servidor FTP	Prover um serviço que permita a transferência dos dados capturados durante as interações do usuário com o aplicativo.
Instalar SGBD	Instalar o BD para armazenamento dos dados e prover a criação do DW.
Instalar a ferramenta OLAP	Instalar uma ferramenta <i>OLAP</i> que permita analisar e manipular uma grande massa de dados sob várias perspectivas.
Modelar o BD	Criar as tabelas e os relacionamentos no BD, possibilitando o armazenamento e a manipulação dos dados.
Modelar o <i>Data Warehouse</i>	Criar as tabelas e relacionamentos de um DW para possibilitar a análise de dados em diferentes tipos de visões.
Instalar a ferramenta <i>Data Loader</i>	Instalar a ferramenta responsável pela extração dos dados dos arquivos XML e transferi-los para o BD. Esta ferramenta foi desenvolvida pela equipe de criação da infraestrutura <i>UXEProject</i> especialmente para este fim.
Instalar a ferramenta <i>ETL Maker</i>	Instalar a ferramenta responsável pela extração, transformação e carga dos dados no DW. Esta ferramenta foi desenvolvida pela equipe de criação da infraestrutura <i>UXEProject</i> .
Instalar a ferramenta de Análise <i>Kettle</i>	Instalar a ferramenta <i>Pentaho Data Integration - PDI (Kettle)</i> para possibilitar a realização das análises dos dados armazenados utilizando o módulo <i>Saiku Analytics</i> .
Criar fonte de dados	Realizar a conexão do <i>Pentaho</i> com o BD e definir as tabelas que farão parte da visão de análise dos dados.
Gerenciar fonte de dados	Criar uma visão para permitir que a ferramenta <i>Pentaho Data Integration - PDI (Kettle)</i> possa gerar as consultas solicitadas pela Equipe de Avaliação.
Criar uma consulta	Criar instruções SQL para extrair informações de usabilidade dos dados armazenados no DW. As instruções SQL são criadas interativamente com a ferramenta <i>Pentaho</i> .

A captura dos dados pode ser atrelada às métricas de usabilidade, contexto e subjetividade disponibilizadas na Biblioteca de Métricas implementada na ferramenta *UXE Metrics Generation*. Além disso, os dados referentes ao perfil do Usuário Final e as especificações de seus *smartphones* são coletados no momento em que é realizado o *download* dos aplicativos. Com o cruzamento destes dados nas avaliações de usabilidade é possível realizar investigações contemplando diferentes pontos de vista.

As principais contribuições que podem ser destacadas com a definição do modelo arquitetural e a sua materialização em uma infraestrutura computacional de suporte são:

- A especificação de um módulo de mapeamento de tarefas automático e a criação de ferramentas para prover esse serviço.
- A definição de uma Biblioteca de Métricas que pode ser estendida devido às propriedades de Extensibilidade e Reusabilidade previstas no modelo.
- A utilização de POA que possibilita a separação do código funcional da aplicação do código referente à captura de informações de usabilidade.
- A criação de uma ferramenta para a geração automática das métricas de rastreabilidade, o que permite a utilização da infraestrutura sem a necessidade do conhecimento de POA.
- A possibilidade de instrumentar diferentes aplicativos com as mesmas métricas previamente definidas.
- A estruturação de um arcabouço de ferramentas projetada na nuvem que possibilita o armazenamento e avaliação dos dados.

Este capítulo detalha o conjunto de experimentos de usabilidade de aplicações para dispositivos móveis com o uso da infraestrutura UXEProject que dá suporte ao modelo conceitual proposto nesta tese. O principal objetivo da realização desse conjunto de experimentos é identificar as potencialidades e restrições da infraestrutura criada. Além disso, as análises consideram dados quantitativos, subjetivos, contextuais e o perfil do usuário, o que tornam a abordagem inovadora na área de avaliação de usabilidade para smartphones. O experimento foi conduzido em campo, inicialmente com a participação de 21 usuários de três aplicações para smartphones, durante um ano.

5 EXPERIMENTO REALIZADO EM CAMPO

Na última década, a comunidade de IHC incorporou outras perspectivas sobre quais elementos deveriam ser observados para permitir que as aplicações móveis satisfizessem às necessidades de seus usuários (Zhang e Adipat, 2005). Devido à multiplicidade e dinamismo dos cenários, não é fácil avaliar a usabilidade contemplando dados quantitativos, subjetivos, contextuais e sobre o perfil dos usuários (Balagtas-Fernandez e Hussmann, 2009). A maioria das abordagens utiliza experimentos controlados ou com dados obtidos por questionários, os quais não consideram variações contextuais ocorridas em um cenário real de interação (Coursaris e Kim, 2011). Um dos desafios nesta área é utilizar avaliações de usabilidade que integrem (i) o perfil dos usuários, (ii) a sua impressão sobre determinados aspectos do aplicativo e (iii) a interferência das variáveis de contexto durante as interações. Em geral, estes três pontos de interesse são investigados de forma isolada, o que dificulta o relacionamento dos resultados (Ardito et al., 2008).

A motivação para realizar o experimento decorre da necessidade de se avaliar como um usuário se adapta às funcionalidades de um aplicativo, qual o grau de satisfação em utilizá-lo e como o ambiente influencia as interações dos usuários sobre um aplicativo na realização de algumas de suas tarefas diárias. Embora este seja um tema amplamente abordado, a área de IHC carece de abordagens que integrem dados quantitativos, subjetivos e contextuais em um único experimento (Lettner e Holzmann, 2012). A infraestrutura UXEProject contribui neste sentido, uma vez que ela permite a extração de dados mesclando duas estratégias para a captura de informações, a de *Logging* e a ESM, conforme apresentado no Capítulo 4.

O experimento para validação do modelo proposto e de sua infraestrutura foi dividido em seis fases distintas. Ele tomou como base o *framework* DECIDE (Sharp et al., 2007) apresentado na seção 3.5.1, o qual norteou a especificação dos passos realizados durante todas as fases do experimento.

5.1 FASE 1 – DETERMINAR OS OBJETIVOS DA AVALIAÇÃO

O primeiro passo foi definir os objetivos a serem alcançados com a realização do experimento. Para essa decisão, considerou-se o modelo e a infraestrutura apresentados no Capítulo 4, já que estabelecem os passos para a instrumentação dos aplicativos a serem avaliados, além da forma de captura e análise dos dados. Outro ponto relacionado aos objetivos do experimento refere-se aos diferentes tipos de dados e a expectativa com relação às informações de usabilidade que podem ser extraídas com o cruzamento desses dados. Sendo assim, as finalidades da etapa de avaliação estão sucintamente enumeradas abaixo:

- 1) Avaliar as potencialidades e limitações da infraestrutura UXEProject para a captura e análise dos dados de usabilidade.
- 2) Conduzir um estudo experimental num ambiente real de uso dos aplicativos pelos usuários.
- 3) Fazer uma análise das informações obtidas no estudo a fim de correlacionar fatores contextuais, métricas de usabilidade e dados subjetivos com a execução de tarefas em um único experimento de usabilidade.
- 4) Evidenciar o comportamento dos usuários de *smartphones* quando passam a utilizar novos aplicativos.
- 5) Identificar problemas de usabilidade específicos em determinadas tarefas.
- 6) Verificar a influência do contexto nas interações.
- 7) Constatar a impressão dos usuários com relação à usabilidade da aplicação sob diferentes aspectos.
- 8) Comparar o desempenho de usuários novatos com o de usuários experientes.

A obtenção dos objetivos acima listados evidencia algumas das contribuições do modelo e infraestrutura propostos nesta tese. Também é possível comprovar os benefícios para a área de IHC sob a ótica dos tipos de informações a serem trabalhadas e quais aspectos de usabilidade podem ser analisados. Por exemplo, espera-se comprovar que as avaliações possam contribuir com a identificação de tarefas que requerem ajustes, avaliar a eficácia do recurso de ajuda, compreender o aprendizado e desempenho dos usuários, avaliar a sua satisfação e fidelização, além de investigar interferências contextuais.

5.2 FASE 2 – EXPLORAR PERGUNTAS A SEREM RESPONDIDAS

Tomando como base os objetivos a serem alcançados, foi elaborado um conjunto de perguntas que direcionam o experimento e a geração e análise dos dados:

- 1) Quais as tarefas mapeadas nos aplicativos que provocam maior dificuldade de interação?
- 2) O que pode ser constatado em relação ao desempenho dos usuários com o passar do tempo?
- 3) Qual é a frequência de utilização da Ajuda dos aplicativos e o desempenho dos usuários após a utilização deste recurso?
- 4) Qual a diferença do desempenho dos usuários comuns comparados a um usuário experiente?
- 5) Que tipo de informações o contexto pode oferecer para melhorar a análise da usabilidade?
- 6) Qual o nível de satisfação dos usuários com os aplicativos propostos no experimento?
- 7) Qual a relação entre os valores de referência, definidos durante o mapeamento das tarefas, com os resultados obtidos pelos usuários no mundo real.

5.3 FASE 3 – ESCOLHER O MÉTODO DE AVALIAÇÃO

O método de avaliação a ser usado no caso específico deste trabalho deve ser realizado almejando atender às seguintes condições:

- 1) Em campo.
- 2) Sem supervisão.
- 3) Por longo período de tempo.
- 4) Com os dados sendo coletados automaticamente.
- 5) Sem restrições quanto ao número de usuários.
- 6) De baixo custo financeiro.
- 7) Sem a necessidade de conhecer como as aplicações a serem avaliadas foram desenvolvidas.
- 8) Com a utilização de uma abordagem que possa ser aplicada a qualquer aplicativo.
- 9) Sem a necessidade de escrever códigos de programação.
- 10) Com a possibilidade de analisar diferentes tipos de dados.
- 11) Com a possibilidade de definir as tarefas a serem analisadas.

O modelo proposto nesta tese e sua infraestrutura UXEProject associada dão suporte à realização do experimento sob as condições apresentadas anteriormente.

5.4 FASE 4 – IDENTIFICAR E ADMINISTRAR AS QUESTÕES PRÁTICAS

Nesta fase, foram levantados inúmeros pré-requisitos, entre os quais podem ser destacados: (i) a escolha das aplicações a serem avaliadas, (ii) a definição das tarefas investigadas, (iii) a definição do grupo de usuários a participar do experimento e (iv) os dados a serem considerados no experimento.

A primeira ação nessa fase foi realizar uma pesquisa exploratória a fim de encontrar aplicativos com funcionalidades atrativas e com possibilidade de serem inseridas no cotidiano das pessoas. A escolha dos aplicativos levou em conta ainda os seguintes pré-requisitos:

- O aplicativo deve ter sido desenvolvido na linguagem Java já que todas as ferramentas e a infraestrutura criada para a validação do modelo foi concebida nessa linguagem, sendo assim, a recomendação é a escolha de aplicativos que executem na plataforma Android¹⁷.
- O código fonte do aplicativo deve estar disponível e com os direitos de uso explícitos.
- O aplicativo deve ter sido construído usando boas técnicas de programação, exibindo uma boa modularização das suas funcionalidades, de forma a permitir que o código fonte possa ser instrumentado com POA.

A partir desses pré-requisitos, foram escolhidas três aplicações para a realização dos experimentos de validação do modelo e de sua infraestrutura. A primeira é chamada de Mileage¹⁸, a segunda é intitulada de ^3 (Cubed)¹⁹ e a terceira é chamada de Shuffle²⁰.

O uso destes três aplicativos nos estudos de casos está fundamentado em dois aspectos importantes no processo de experimentação da infraestrutura UXEProject:

- Mostrar que a infraestrutura e o modelo que lhe deu origem podem ser utilizados em aplicativos para *smartphones*.

¹⁷ <http://www.portalandroid.org/comunidade>

¹⁸ A aplicação está disponível no link <http://evancharlton.com/projects/mileage>

¹⁹ A aplicação está disponível no link <http://abrantix.org/3.php>

²⁰ A aplicação está disponível no link <http://code.google.com/p/android-shuffle/>

- Mostrar que, apesar do processo de concepção do aplicativo não possuir qualquer ligação com o modelo de avaliação de usabilidade proposto, os aspectos de mapeamento das tarefas e das métricas podem ser facilmente integradas ao seu código fonte.

Nas próximas subseções, serão descritas as funcionalidades dos aplicativos e as tarefas que foram instrumentadas para realizar os experimentos.

5.4.1 Aplicativos utilizados no experimento

O primeiro aplicativo selecionado para o experimento de avaliação de usabilidade foi o Mileage. A finalidade dele é auxiliar os usuários no controle do gasto com combustível e outros serviços de manutenção de um automóvel, como troca de óleo, troca de pastilhas de freio, entre outros.

A Figura 44 apresenta a interface do aplicativo, que consiste de um menu principal com 4 botões para acesso às funcionalidades associadas ao controle de consumo/gasto, histórico, estatística e gráficos. Um menu auxiliar também pode ser acionado possibilitando o acesso a outras opções, tais como: cadastrar um veículo, realizar configurações do aplicativo, importar e exportar dados, acrescentar controles de manutenção e acionar a ajuda. Para possibilitar as investigações, foi necessário escolher as tarefas disponibilizadas pelos aplicativos que fossem de interesse para as avaliações e com maior diversidade de interações. As tarefas selecionadas são apresentadas à direita da interface do Mileage na Figura 44.


Interface do Mileage	Tarefas Instrumentadas
	<ol style="list-style-type: none"> 1. Cadastrar um veículo. 2. Entrar com os dados referentes a um novo abastecimento. 3. Realizar uma nova configuração referente ao formato da data. 4. Alterar dados referentes a um item constante no histórico dos abastecimentos. 5. Adicionar controle de manutenção (por exemplo, próxima troca de óleo). 6. Visualizar o gráfico referente à variação do preço do combustível. 7. Visualizar o gráfico referente à distância percorrida. 8. Importar dados armazenados.

Figura 44 – Interface do Aplicativo Mileage e tarefas instrumentadas para sua avaliação.

O segundo aplicativo selecionado foi ^3 (Cubed), um gerenciador de músicas e vídeos. Seu menu principal possibilita escolher as músicas ou vídeos e executá-los. O menu auxiliar permite encontrar capas de álbuns, escolher gêneros musicais, modificar configurações, definir a aparência, escolher temas de apresentação e definir opções de equalização. A interface do aplicativo, assim como as tarefas mapeadas e instrumentadas no seu código, podem ser contempladas na Figura 45.


Interface do ^3 (Cubed)	Tarefas Instrumentadas
	<ol style="list-style-type: none"> 1. Escolher uma música de uma lista de reprodução. 2. Criar uma nova lista de reprodução. 3. Inserir uma música a uma lista de reprodução. 4. Modificar a aparência da aplicação. 5. Escolher um novo tema de apresentação do aplicativo. 6. Acionar o equalizador.

Figura 45 – Interface do Aplicativo ^3 (Cubed) e tarefas instrumentadas para sua avaliação.

O terceiro e último aplicativo escolhido para os experimentos foi o Shuffle, cuja interface é mostrada na Figura 46. Ele é um aplicativo de agenda de atividades que permite vincular tarefas a datas e horários, além de possibilitar o relacionamento a projetos e contextos (por exemplo, casa ou trabalho). As três primeiras opções do menu disponibilizam formas diferentes de visualizar e cadastrar atividades. As três últimas opções permitem cadastrar projetos, contextos e atalhos para encontrar tarefas. Seu menu secundário tem ações vinculadas com as opções do menu principal, além da configuração de preferências, opções de pesquisas e ajuda. As tarefas escolhidas para a instrumentação do Shuffle estão na parte direita da Figura 46.

É importante destacar que, independente das tarefas escolhidas para a avaliação, o usuário não deve perceber nenhum tipo de alteração nas funcionalidades ou aparência dos aplicativos selecionados para a avaliação experimental de usabilidade proposta neste trabalho. A única alteração prevista é a apresentação, eventual, das perguntas subjetivas para o usuário durante a utilização dos aplicativos.

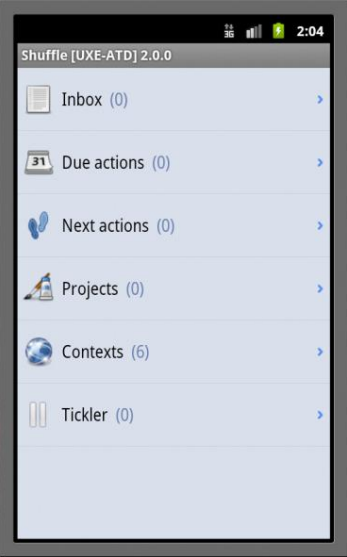
Interface do Shuffle	Tarefas Mapeadas e Instrumentadas
	<ol style="list-style-type: none"> 1. Inserção de uma nova atividade. 2. Excluir todas as atividades da caixa de entrada. 3. Modificar uma atividade previamente cadastrada. 4. Cadastrar um novo projeto com a escolha de um contexto disponível. 5. Cadastrar um novo contexto. 6. Excluir um contexto previamente cadastrado. 7. Modificar um projeto já cadastrado com a escolha de um novo contexto. 8. Realizar <i>backup</i> dos dados utilizando as preferências gerais do aplicativo. 9. Acionar a opção de ajuda. 10. Criar uma atividade com agendamento.

Figura 46 – Interface do Aplicativo Shuffle e tarefas instrumentadas para sua avaliação.

5.4.2 Recrutamento dos participantes

Outra ação realizada nesta fase foi definir o grupo de usuários que participariam dos experimentos. A escolha dos participantes foi em função dos perfis que estavam em análise e das características dos seus *smartphones* (Tabela 20).

Conforme o estudo de Nielsen (1994), de três a cinco participantes são suficientes para a realização de um experimento com o objetivo de detectar problemas de usabilidade, com assertividade de no mínimo 95%. Seguindo este estudo, foi tomado o cuidado para que no mínimo três usuários se enquadrassem em cada perfil de interesse. Por fim, foram selecionados 21 usuários, distribuídos conforme a Tabela 20. Cabe ressaltar ainda que, no experimento, cada usuário é identificado pelo número IMEI do seu dispositivo. Assim, considera-se que cada dispositivo móvel é pessoal e utilizado por apenas um participante. Mantém-se, dessa forma, a privacidade dos mesmos, uma vez que eles não podem ser identificados diretamente.

Tabela 20 – Distribuição dos usuários de acordo com seus perfis

Faixas de Distribuição		Nº de Participantes
Idade	Entre 18 e 28 anos	6
	Entre 29 e 39 anos	5
	Entre 40 e 50 anos	5
	Acima de 50 anos	5
Nível de Escolaridade	Fundamental	3
	Médio	5
	Graduado	8
	Pós-graduado	5
Formação	Exatas	6
	Humanas	3
	Saúde	4
	Outras (inclui os não graduados)	8
Ocupação	Educação	5
	Indústria	4
	Serviços	7
	Comércio	5
Poder Aquisitivo ²¹	Baixo (até 3 salários mínimos)	6
	Médio (até 20 salários mínimos)	10
	Alto (mais de 20 salários mínimos)	5

5.4.3 Dados considerados no experimento

A relação dos dados que seriam considerados no experimento foi definido em função das estratégias de captura providos pela infraestrutura UXEProject. Sendo assim, foram considerados os dados de usabilidade referentes às tarefas mapeadas (definidas na Seção 5.4.1), o perfil dos usuários (apresentado na Tabela 20), as características dos *smartphones*, os dados contextuais obtidos por meio dos sensores e as respostas subjetivas fornecidas pelos usuários.

As características dos *smartphones* consideradas para compor o contexto das interações foram o tamanho e resolução da tela, cujas faixas de valores consideradas estão na Tabela 21.

²¹ Dados baseados na classificação do DIEESE. Disponível em www.dieese.org.br.

Tabela 21 – Características dos *Smartphones*

Características	Faixas de valores consideradas		Nº de Usuários
Resolução da Tela em Pixels	$\chi \leq 320 \times 240$	Baixo	9
	$320 \times 240 < \chi \leq 320 \times 480$	Média	6
	$\chi > 320 \times 480$	Alta	6
Tamanho da Tela em polegadas	$\chi \leq 2.4$	Pequeno	6
	$2.4 < \chi \leq 3.5$	Médio	10
	$\chi > 3.5$	Grande	5

Para contextualizar o ambiente no momento em que ocorrem as interações, são capturados dados referentes ao grau de luminosidade, à posição do aparelho durante as interações e à velocidade de deslocamento do usuário. Estes dados de contexto são capturados diretamente dos sensores dos aparelhos e seus valores de referência estão descritos na Tabela 22.

Tabela 22 – Escala de valores para dados do ambiente

Fatores	Faixa de valores considerados		
Luminosidade	$\chi \leq 100$ lux		Baixo
	$100 < \chi \leq 10000$ lux		Média
	$\chi > 10000$ lux		Alta
Deslocamento	$\chi < 0,2$ m/s		Parado
	$0,2 \text{ m/s} \leq \chi \leq 2,7$ m/s		Caminhando
	$\chi > 2,7$ m/s		Motorizado
Posição	Vertical	Horizontal	Mista

No escopo das avaliações propostas neste experimento, pretende-se analisar três aspectos principais: (i) o desempenho e o comportamento dos usuários ao longo da realização dos experimentos; (ii) as possíveis interferências do contexto durante as interações; e (iii) a satisfação dos usuários com relação à usabilidade dos aplicativos.

O tipo de dado considerado para analisar o primeiro item é classificado como quantitativo, ou seja, representa uma grandeza resultante de uma contagem ou medição (Barbosa e Silva, 2010). Para avaliar o segundo item, os dados objetivos que podem ser obtidos por meio de instrumentos ou softwares são considerados (Barbosa e Silva, 2010). Para analisar o terceiro item, foram consideradas as respostas dos usuários às perguntas a eles apresentadas durante o experimento. O tipo de dado associado à técnica ESM é categorizado como subjetivo, ou seja, deve ser explicitamente expresso pelos participantes do experimento através de sua opinião (Barbosa e Silva, 2010).

5.5 FASE 5 – DECIDIR COMO LIDAR COM AS QUESTÕES ÉTICAS

O Conselho Nacional de Saúde (1996), na Resolução número 196/96²², define pesquisa com seres humanos como toda investigação que, individual ou coletivamente, envolva o contato direto ou indireto com o ser humano, incluindo o manejo de informações por ele geradas. Ainda segundo a mesma resolução, toda pesquisa com seres humanos envolve risco. Risco em pesquisa refere-se à possibilidade de danos à dimensão física, psíquica, moral, intelectual, social ou cultural do ser humano (Leitão e Romão-Dias, 2003).

Segundo Leitão e Romão-Dias (2003), quatro pontos da Resolução 196/96 mostram-se particularmente relevantes para o controle dos riscos nas pesquisas em IHC: o consentimento dos sujeitos, a preservação do anonimato, a proteção de grupos vulneráveis e a garantia de bem-estar desses sujeitos.

Por consentimento dos sujeitos, entende-se a anuência em participar de uma pesquisa após o conhecimento detalhado de seus objetivos. Quando o contato entre o sujeito e o pesquisador é interativo e/ou presencial, o consentimento é facilmente obtido. Já em algumas pesquisas em IHC como, no caso, de experimentos que utilizam a abordagem proposta nesta tese, nem sempre haverá o contato direto com o usuário e, nesta situação, não será possível utilizar os procedimentos tradicionais para a obtenção deste consentimento. Tal impasse impõe dois desafios: o de criar novas formas de explicar os objetivos da pesquisa e o de pensar em uma forma de obter a anuência dos sujeitos.

Neste sentido, para a realização deste experimento, foi construído um *site* onde são disponibilizadas as explicações referentes à pesquisa e um termo de uso dos aplicativos. Para o usuário habilitar-se a realizar o *download* dos aplicativos, é necessário explicitar o seu consentimento em participar do experimento. O termo de uso está disponível no Anexo VIII.

As pesquisas com seres humanos devem também garantir a confidencialidade das informações prestadas e o anonimato dos sujeitos. Segundo Leitão e Romão-Dias (2003), isto quer dizer que dados de identificação, tais como, o nome e a imagem dos sujeitos, não devem ser divulgados. Além disso, as informações coletadas só podem ser divulgadas se sua autoria não for passível de identificação.

No caso da abordagem proposta nesta tese, a identificação de um usuário é exclusivamente através do IMEI de seu *smartphone*. Desta forma, nenhum outro dado de identificação pessoal é necessária, além de ser explicitado no termo de compromisso que os dados jamais serão utilizados individualmente.

Merecem cuidados especiais as pesquisas com os chamados grupos vulneráveis, ou seja, crianças, adolescentes, pessoas com problemas mentais e adultos com capacidade de autodeterminação

²² Disponível em http://conselho.saude.gov.br/web_comissoes/conep/arquivos/materialeducativo/cadernos/caderno01.pdf

reduzida. Sem defesas adequadas, estes grupos são mais suscetíveis aos riscos de uma pesquisa e, por este motivo, devem participar somente quando isto for imprescindível para os objetivos da investigação (Leitão e Romão-Dias, 2003). No caso específico deste experimento, nenhum participante se enquadra nos grupos definidos anteriormente.

O último ponto a ser observado, refere-se à responsabilidade dos pesquisadores em garantir o bem estar dos sujeitos, dando assistência integral às complicações decorrentes dos riscos previstos em seus projetos. Do ponto de vista psicológico, isto implica na necessidade dos pesquisadores em IHC estarem sensíveis aos temores, inseguranças e ansiedades gerados pela exposição dos participantes de seus projetos (Leitão e Romão-Dias, 2003). No caso específico deste experimento, não foi constatado ou relatado qualquer tipo de alteração psicológica que infringisse a ética adotada nesse tipo de estudo.

5.6 FASE 6 – AVALIAR, INTERPRETAR E APRESENTAR OS RESULTADOS

Nesta seção serão apresentados os resultados das avaliações realizadas durante a condução do experimento. A coleta de dados ocorreu no período de 01/12/2011 a 30/11/2012, totalizando doze meses.

Inicialmente, foram instrumentados 32 *smartphones*, entretanto, participaram efetivamente 21 usuários durante os seis primeiros meses e permaneceram utilizando os aplicativos, até o final do experimento, 9 participantes. Desta forma, as avaliações apresentadas nesta seção vão considerar os dados obtidos no decorrer dos seis primeiros meses e os dados referentes aos 9 participantes que completaram todo o período de investigação. Os usuários foram cadastrados ao longo do mês de dezembro e, assim, eles não possuem exatamente o mesmo tempo de participação no experimento. Entretanto, a diferença entre as datas do primeiro e do último cadastrado foi de 18 dias. Todos os participantes tiveram uma prévia demonstração dos aplicativos e o suporte da Equipe de Avaliação no esclarecimento de dúvidas ao longo da realização do experimento. Entretanto, é importante salientar que a interação entre os Usuários e a Equipe de Avaliação não é um pré-requisito estabelecido na abordagem proposta, ficando a critério dos avaliadores. Além disso, a abordagem prevê que a Equipe de Avaliação pode realizar os experimentos sem uma relação direta entre os avaliadores e usuários.

As subseções seguintes detalham os resultados das avaliações em função das perguntas definidas na Seção 5.2.

5.6.1 Identificação de tarefas com problemas de usabilidade

Quando uma tarefa é mapeada, todos os passos que devem ser executados para a sua conclusão são determinados e, desta forma, quando uma ação é realizada sem conformidade com o mapeamento realizado pela Equipe de Avaliação considera-se que ocorreu um erro.

A investigação dos problemas de usabilidade para cada uma das tarefas avaliadas foi obtida a partir da divisão do número de vezes em que a tarefa foi completada com erros pelo total de vezes em que ela foi executada.

Nesta avaliação, foram considerados problemas de usabilidade as tarefas que apresentam uma taxa de erro superior a 10%, entretanto, esse número foi escolhido somente em função da percepção da Equipe de Avaliação, já que não foi encontrada na literatura a indicação de um valor de referência.

As Tabelas 23, 24 e 25 apresentam as tarefas mapeadas em cada aplicativo, com os seus respectivos percentuais de erros, sendo que os dados são referentes aos primeiros seis meses de execução do experimento.

Tabela 23 – Percentual de tarefas completadas com erro no Mileage

ID	Mileage	% Erros
1	Cadastrar um veículo	12,55%
2	Entrar com os dados referentes a um novo abastecimento.	6,87%
3	Realizar uma nova configuração referente ao formato da data.	8,06%
4	Alterar dados referentes a um item constante no histórico dos abastecimentos.	4,94%
5	Adicionar um novo controle de manutenção (por exemplo, a próxima troca de óleo).	6,10%
6	Visualizar o gráfico referente à variação do preço do combustível.	3,50%
7	Visualizar o gráfico referente à distância percorrida.	3,31%
8	Importar dados armazenados.	17,03%

No aplicativo Mileage, como pode ser observado na Tabela 23, as tarefas 1 e 8 apresentam os maiores percentuais de erros, com uma taxa superior a 10%. Na tarefa “cadastrar um veículo”, é provável que os erros ocorram devido à necessidade do usuário ter que rolar a tela do aplicativo para introduzir todos os dados referentes ao mesmo, apesar de não existir uma barra de rolagem indicando este processo. A tarefa “importar dados” requer que o usuário relacione os campos do arquivo a ser importado com os campos do BD utilizado no aplicativo, o que torna a tarefa muito complexa para usuários leigos em computação.

Tabela 24 – Percentual de tarefas completadas com erro no Cubed

ID	^3 (Cubed)	% Erros
1	Escolher uma música de uma lista de reprodução.	8,29%
2	Criar uma nova lista de reprodução.	9,87%
3	Inserir uma música a uma lista de reprodução.	9,98%
4	Modificar a aparência da aplicação.	7,09%
5	Escolher um novo tema de apresentação do aplicativo.	8,19%
6	Acionar o equalizador	5,69%

No Cubed, as tarefas apresentam uma distribuição percentual de erros com pouca variação e todas abaixo de 10%. Esse fato retrata a simplicidade de utilização deste aplicativo e permite concluir que os passos para a execução das tarefas estão adequados para o seu propósito.

Tabela 25 – Percentual de tarefas completadas com erro no Shuffle

ID	Shuffle	% Erros
1	Inserção de uma nova atividade.	7,32%
2	Excluir todas as atividades da caixa de entrada.	3,01%
3	Modificar uma atividade previamente cadastrada.	0,00%
4	Cadastrar um novo projeto com a escolha de um contexto disponível.	0,00%
5	Cadastrar um novo contexto.	2,40%
6	Excluir um contexto previamente cadastrado.	0,00%
7	Modificar um projeto já cadastrado com a escolha de um novo contexto.	3,60%
8	Realizar uma cópia (<i>backup</i>) dos dados utilizando as preferências gerais do aplicativo.	3,15%
9	Acionar a opção de ajuda.	7,10%
10	Criar uma atividade com agendamento.	12,70%

Na análise relacionada ao aplicativo Shuffle (Tabela 25), a única tarefa que apresentou uma taxa de erros acima de 10% foi a de “criar uma atividade com agendamento”. Este fato provavelmente ocorreu porque, após o usuário colocar a data e horário para o início do agendamento, não ficaram visíveis os campos para a inserção dos dados referentes ao término do mesmo, sendo necessário o usuário descobrir este requisito intuitivamente.

Para uma avaliação mais detalhada com relação às três tarefas (duas do Mileage e uma do Shuffle) nas quais foram detectado os índices de erros superiores a 10%, foi realizada uma análise considerando exclusivamente os 9 usuários que permaneceram durante os doze meses de execução do experimento. Na Tabela 26, é apresentada a taxa de erros levando em consideração apenas estas três tarefas.

Tabela 26 – Avaliação das tarefas com maior índice de erros durante 12 meses

Descrição das Tarefas	% Erros
Cadastrar um veículo (Mileage)	12,1%
Importar dados armazenados (Mileage)	18,1%
Criar uma atividade com agendamento (Shuffle)	8,1%

Como pode ser observado na Tabela 26, a tarefa para “cadastrar um veículo” no aplicativo Mileage apresenta um índice de erro próximo à constatação anterior, quando se considerou a participação dos 21 usuários nos seis primeiros meses do experimento (ver Tabela 23). Essa semelhança pode ser decorrente do fato dos usuários, de uma forma geral, executarem poucas vezes esta tarefa durante os doze meses, sendo que a maioria deles só executou a atividade uma ou duas vezes, o que não permite um processo de aprendizado e percepção das particularidades existentes. Outro ponto importante a ser salientado, é que a maioria dos participantes executou essa tarefa nas primeiras interações com o aplicativo, quando ainda estavam se familiarizando com os recursos e o *designer* disponibilizado.

Com relação à tarefa de “importar dados armazenados” do Mileage, é possível observar um pequeno aumento na taxa de erros com relação às constatações anteriores (ver Tabela 23), caracterizando que realmente a tarefa necessita de ajustes, já que evidencia problemas de aprendizagem, flexibilidade, facilidade de uso e comunicabilidade.

A tarefa para “criar uma atividade com agendamento” no aplicativo Shuffle apresentou uma considerável redução no índice de erros com relação às constatações anteriores (ver Tabela 25), caracterizando que a frequência de utilização desta tarefa possibilita a familiarização por parte dos usuários dos passos necessários para a sua execução.

Uma constatação geral com relação à usabilidade das tarefas mapeadas, é que a necessidade de deslocar a tela do aplicativo para a vertical com o objetivo de completar os passos relacionados à execução das tarefas dificulta as interações, neste caso, aumenta a probabilidade de erros, como foi observado nas tarefas para “cadastramento de veículos” e “criar uma atividade com agendamento”.

É relevante destacar que a comprovação de problemas de usabilidade em uma tarefa requer uma análise em conjunto da equipe de desenvolvimento e da equipe de IHC. A definição de uma reestruturação para uma determinada tarefa requer a troca de experiências e interação entre estas duas áreas.

5.6.2 Análise do desempenho dos usuários ao longo do tempo

Para verificar o aprendizado e analisar o desempenho dos usuários no decorrer dos meses nos quais o experimento foi realizado, foram investigadas a eficácia e a eficiência nas interações, utilizando, respectivamente, a taxa de tarefas completadas com erro e a velocidade média para a execução das tarefas. As análises foram realizadas considerando diferentes ângulos, inicialmente subdividindo os resultados em função dos aplicativos e posteriormente analisando o desempenho individual e médio dos usuários que participaram durante um ano do experimento.

Na Figura 47, é apresentado o gráfico referente ao percentual de tarefas completadas com erro ao longo dos seis primeiros meses da realização do experimento. É possível observar que em dezembro de 2011, a taxa de erros é bem maior que nos demais meses de 2012.

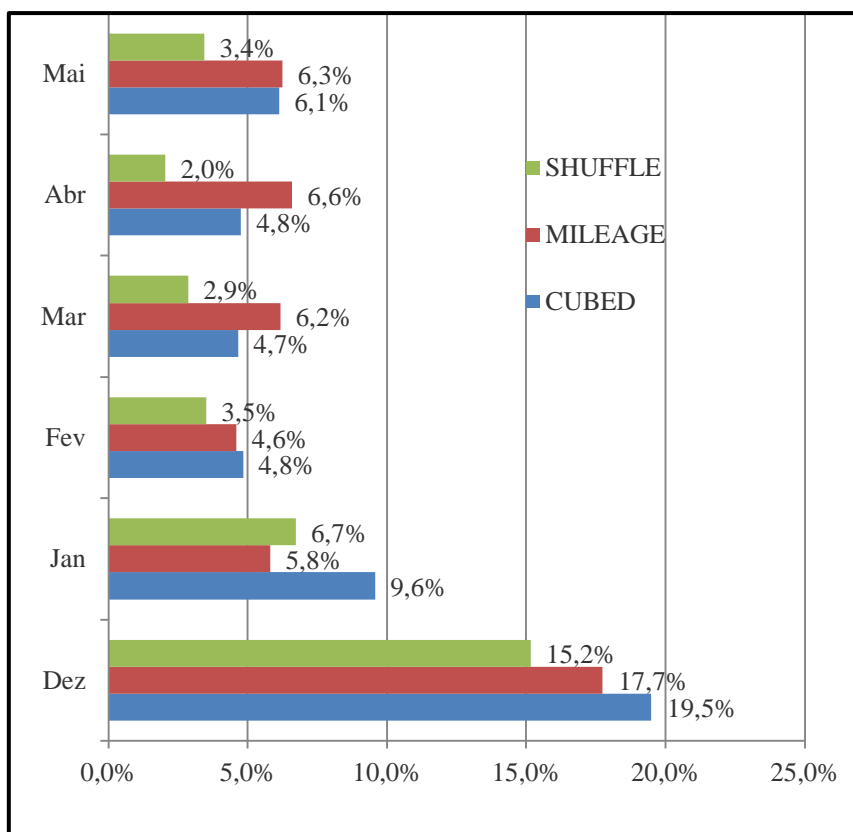


Figura 47 – Percentual de tarefas completadas com erro

Outro fator importante para identificar a evolução do aprendizado dos usuários referente a um novo aplicativo é o tempo médio gasto para a realização das tarefas. Na Figura 48, são apresentados os tempos médios, em segundos, para a conclusão das tarefas disponibilizadas nos três aplicativos. Observa-se

que, no primeiro mês, existe uma discrepância com relação ao tempo médio dos demais meses. Em outras palavras, em dezembro, no primeiro mês de uso dos aplicativos, os usuários necessitaram de muito mais tempo para a conclusão das tarefas do que nos meses subsequentes.

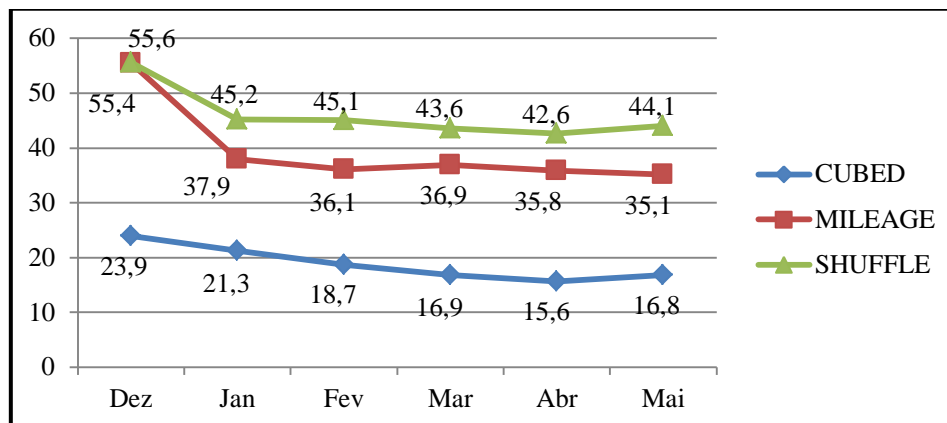


Figura 48 – Tempo médio para execução das tarefas propostas

Após a análise dos dados apresentados nos gráficos que demonstram o desempenho médio dos usuários ao longo dos seis primeiros meses, pode-se afirmar que:

- O aprendizado dos aplicativos por parte dos usuários ocorre durante os dois primeiros meses. Nos meses seguintes, o desempenho se mantém numa faixa constante, com pequenas variações nos valores das métricas utilizadas na análise.
- Após o primeiro mês, a taxa de erros e o tempo necessário para a execução das tarefas são reduzidos significativamente, o que demonstra maior familiarização dos usuários com os aplicativos.

Para uma constatação mais efetiva, o desempenho de 8 dos 9 participantes que permaneceram até o final do experimento foram analisados individualmente. Um dos participantes foi descartado da análise por ser classificado como Usuário Experiente (assunto que será discutido na subseção 5.6.4). A Tabela 27 mostra que a taxa de erros de cada participante durante os doze meses ao utilizarem os três aplicativos.

Tabela 27 – Taxa de erro dos 8 participantes considerando os 12 meses

Participante	Dez	Jan	Fev	Mar	Abr	Mai	Jun	Jul	Ago	Set	Out	Nov
1	18,3%	9,7%	7,1%	7,5%	6,2%	6,1%	5,6%	5,2%	5,1%	4,8%	4,9%	5,0%
2	16,2%	6,8%	6,3%	5,6%	4,2%	5,0%	4,7%	4,3%	4,1%	4,2%	4,3%	4,2%
3	17,4%	7,3%	6,5%	5,2%	4,4%	4,1%	4,3%	3,8%	3,6%	3,7%	3,5%	3,9%
4	15,1%	7,0%	6,2%	5,3%	4,3%	4,7%	4,5%	4,1%	4,4%	4,3%	4,2%	4,1%
5	19,7%	10,2%	7,3%	6,0%	6,8%	6,2%	5,8%	5,2%	5,0%	5,1%	5,3%	5,5%
6	17,7%	7,6%	5,6%	4,1%	3,8%	4,2%	4,0%	4,4%	4,8%	4,6%	4,9%	4,7%
7	14,9%	7,1%	5,0%	4,1%	3,6%	3,8%	3,7%	3,9%	3,6%	3,7%	3,2%	3,5%
8	17,3%	8,4%	5,2%	4,5%	4,7%	4,6%	4,1%	3,8%	3,9%	4,0%	3,6%	4,2%
Média	17,1%	8,0%	6,2%	5,3%	4,8%	4,8%	4,6%	4,3%	4,3%	4,3%	4,2%	4,4%

Analisando os dados apresentados na Tabela 27, é possível chegar às mesmas constatações obtidas após a avaliação das informações exibidas no gráfico da Figura 47. Observa-se que no primeiro mês todos os participantes analisados apresentam uma taxa de erro bem maior que no segundo, com uma estabilização no restante do período investigado.

Uma análise semelhante à anterior foi realizada para avaliar a velocidade de execução das tarefas, considerando os doze meses do experimento e a participação dos mesmos 8 usuários. O gráfico da Figura 49 mostra o tempo médio necessário para os usuários completarem as tarefas nos três aplicativos. Pelo gráfico pode-se chegar às mesmas conclusões obtidas após a análise das informações da Figura 48, quando foi constatado que a velocidade de execução das tarefas diminui após o primeiro mês e se mantém praticamente estável nos meses subsequentes.

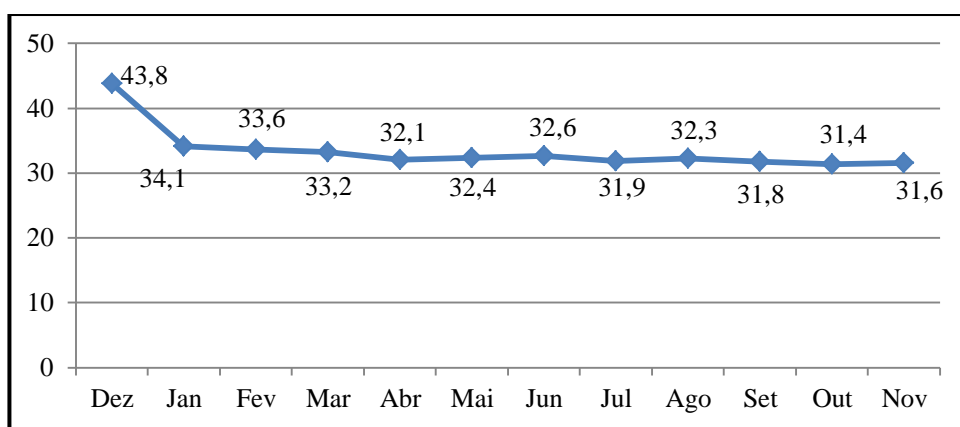


Figura 49 – Tempo médio para a execução das tarefas considerando 8 participantes

5.6.3 Análise da frequência de utilização da ajuda

Para responder à terceira questão, foi realizada uma investigação sobre a frequência de utilização da Ajuda nos aplicativos Mileage e Shuffle, já que o Cubed não disponibiliza este recurso. A outra métrica de verificação utilizada foi analisar a taxa de erros para completar a tarefa após ser acionada a Ajuda.

Na Figura 50, são apresentados os percentuais de utilização da Ajuda dos aplicativos Mileage e Shuffle durante os seis primeiros meses de execução dos experimentos. Nela, pode ser observado que, no mês de Janeiro de 2012, o primeiro mês de uso dos aplicativos, a busca por este recurso é superior aos demais meses. Esta informação possibilita inferir que nesse mês os usuários começaram a utilizar a Ajuda

com mais frequência, o que pode indicar dificuldades para que os usuários possam explorar novos recursos sem um auxílio adicional.

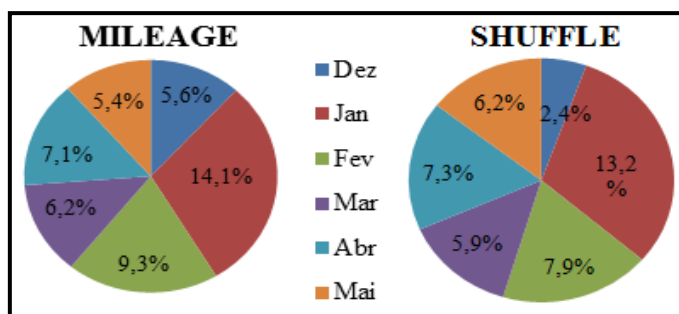


Figura 50 – Percentual de utilização da opção de Ajuda

Para comprovar a eficiência da utilização da Ajuda, avaliamos o percentual de erros cometidos para terminar a tarefa, após ser acionado este recurso. De acordo com os dados identificados, a taxa de erro no Mileage é 31,1% e no Shuffle é de 20,8%, o que caracteriza que este recurso deve ser reestruturado para esclarecer de forma mais efetiva as dúvidas dos usuários.

Para completar a análise, foi investigada a utilização da Ajuda nos seis últimos meses do experimento quando foi constatado que os 9 participantes, que permaneceram utilizando os aplicativos, praticamente não utilizaram a opção de Ajuda neste período, sendo observado à utilização deste recurso em apenas quatro vezes.

Diante das investigações realizadas para responder à terceira questão, foram inferidas as seguintes conclusões:

- A Ajuda é um recurso importante, utilizado com regularidade pelos participantes, principalmente nos primeiros meses de contato com um novo aplicativo.
- Ficou constatado que sua utilização pode ter auxiliado no aprendizado dos aplicativos, nos dois primeiros meses de uso, já que foi observado que ocorreram melhoras significativas no desempenho dos participantes nos meses subsequentes.
- Foi observado alto percentual de erros para terminar a tarefa após a utilização da Ajuda e, apesar da sua eficiência para esclarecer a finalidade de cada tarefa, não é eficiente para ilustrar como esta deve ser realizada.

5.6.4 Análise do desempenho dos Usuários Comuns X Experiente

Para responder à quarta pergunta, foi comparado o desempenho dos usuários Comuns com um usuário Experiente. As métricas utilizadas foram: o tempo médio para completar as tarefas, o percentual de erros cometidos e o percentual de tarefas iniciadas e não concluídas. As análises foram divididas em dois períodos: durante os seis primeiros meses, considerando 21 participantes e, posteriormente, levando em consideração apenas os seis últimos meses, com a participação de 9 usuários.

O critério utilizado pela Equipe de Avaliação para classificar um dos usuários como Experiente foi decorrente da sua familiarização com os aplicativos, antes do experimento ter sido iniciado. A Tabela 28 apresenta valores comparativos entre o desempenho médio dos usuários Comuns com o desempenho do usuário Experiente durante os seis primeiros meses.

Tabela 28 – Desempenho dos usuários Comuns x Experiente (primeiros 6 meses)

Tipo de Usuário	Cubed	Mileage	Shuffle
Tempo médio para completar as tarefas (em seg.)			
Comuns	18,9	39,6	46,0
Experiente	13,0	34,1	34,4
Percentual de ações erradas			
Comuns	11,0%	9,4%	5,9%
Experiente	7,0%	5,9%	3,3%
Percentual de tarefas iniciadas e não concluídas			
Comuns	13,9%	16,5%	14,7%
Experiente	13,6%	13,2%	5,4%

Pode ser constatado que o desempenho do usuário Experiente é superior ao desempenho médio dos demais participantes do experimento. A resposta da quarta questão é mais um indício de que o usuário, ao se tornar mais experiente, consegue obter melhor desempenho na realização das tarefas.

Suzuki et al. (2011) obtiveram resultados semelhantes em um estudo na área de ergometria. No estudo, os usuários foram submetidos à utilização de um aplicativo em três tipos de celular. A constatação foi que os usuários conseguiam obter um desempenho melhor no aparelho em que tinham mais experiência.

Para uma comparação mais efetiva, foram selecionadas as mesmas métricas de comparação da análise anterior, entretanto, considerando os últimos seis meses do experimento (Tabela 29). O objetivo é observar novamente o comportamento dos usuários Comuns em relação ao usuário Experiente, levando em consideração que os usuários Comuns possuem no mínimo seis meses de familiarização com os aplicativos.

Tabela 29 – Desempenho dos usuários Comuns x Experiente (últimos 6 meses)

Tipo de Usuário	Cubed	Mileage	Shuffle
Tempo médio para completar as tarefas (em seg.)			
Comuns	16,7	35,3	43,6
Experiente	13,2	33,8	34,3
Percentual de ações erradas			
Comuns	8,1%	6,2%	3,8%
Experiente	6,8%	5,5%	3,4%
Percentual de tarefas iniciadas e não concluídas			
Comuns	8,7%	9,7%	7,1%
Experiente	7,9%	9,3%	5,2%

Inicialmente, um fato que chama atenção nos dados apresentados na Tabela 29, é que todos os valores referentes aos usuários Comuns são menores que os valores observados na Tabela 28, possibilitando inferir que o desempenho dos usuários melhora de acordo com a frequência de utilização do aplicativo.

O segundo ponto a ser observado é que, a partir dos dados apresentados na Tabela 29, os usuários Comuns se aproximaram mais do desempenho do usuário Experiente em todas as métricas investigadas. Além disso, o usuário Experiente manteve a velocidade para completar as tarefas e a taxa de erros praticamente inalteradas. Estas informações são indícios de que existe um ponto de estabilização na curva de aprendizagem e, mesmo com longo período de utilização, o usuário não obtém melhores resultados.

5.6.5 Análise de informações contextuais

A Computação Ubíqua trabalha com o preceito de que os softwares fazem parte do cotidiano das pessoas e devem estar disponíveis, de forma transparente, “a qualquer hora, em qualquer lugar e a partir de qualquer dispositivo” (Weiser, 1991). Esse preceito vem sendo cada vez mais explorado na construção de aplicativos para *smartphones*. Alcançar a ubiquidade dos aplicativos para *smartphones* implica em monitorar, de forma automática, as informações de contexto relativas ao uso desses dispositivos. Neste caso, contexto pode ser definido como um conjunto de informações que afeta a execução de uma aplicação relacionada às pessoas, objetos, lugares, tempo e espaço em que a aplicação é utilizada (Vieira et al., 2011).

Coletar dados das experiências dos usuários de *smartphones* e associá-los ao contexto no qual as interações ocorrem é um grande desafio para a área de IHC. As situações mudam e os resultados dos testes são altamente dependentes do contexto (Bernard et al., 2007). Por exemplo, uma pessoa interagindo com um

aplicativo móvel, sentada no sofá de sua casa, terá diferentes interferências externas quando comparada à realização da mesma tarefa ao caminhar pela rua. O contexto social é outro fator importante, pois as interações não são apenas influenciadas pelo que o usuário está fazendo, mas por quem está a sua volta durante as interações (Bernard et al., 2007).

Para avaliar a usabilidade de aplicativos móveis de forma mais precisa, é necessário relacionar o contexto com a forma como usuários interagem com estes aplicativos. Para a realização de estudos com esta abrangência, são necessárias abordagens e técnicas que permitam realizar experimentos com a capacidade de coletar dados contextualizados aos cenários onde as interações ocorrem. Este fato desencadeia diversas discussões quanto ao local da realização dos experimentos (em campo ou em laboratório), bem como as técnicas que podem ser utilizadas para a extração do melhor conjunto de dados que caracterizem os experimentos (Kaikkonen et al., 2005) (Kjeldskov e Stage, 2004) (Queiroz e Ferreira, 2009).

Esta subseção foi motivada por estas discussões, tendo como principais objetivos:

- Relacionar os aspectos contextuais que podem influenciar as interações dos usuários em cenários reais de uso dos aplicativos para *smartphones*.
- Utilizar a infraestrutura UXEProject criada com características que possibilitam a extração de dados contextuais dos cenários de avaliação.
- Apresentar os resultados do experimento realizado em campo, relacionando fatores contextuais com métricas de usabilidade.

Inicialmente, são observados os valores percentuais de tarefas completadas com erros em cada um dos aplicativos sob a ótica da variação da luminosidade. O objetivo é identificar a possível influência desta variável contextual nas interações dos participantes do experimento. Para a realização da análise, a luminosidade foi isolada e relacionada com o percentual de tarefas completadas com erros em cada um dos aplicativos, como apresentado na Figura 51. Vale ressaltar que todos os dados analisados fazem parte dos primeiros seis meses de realização do experimento, contando com a participação de vinte e uma pessoas.

Conforme pode ser observado na Figura 51, foi detectado, para todos os aplicativos, que os maiores índices de tarefas completadas com erros ocorrem quando a luminosidade está muito alta ou muito baixa, ou seja, quando as condições do cenário de interação não estão nos parâmetros considerados normais, o que comprova a influência da luminosidade com relação ao número de erros.

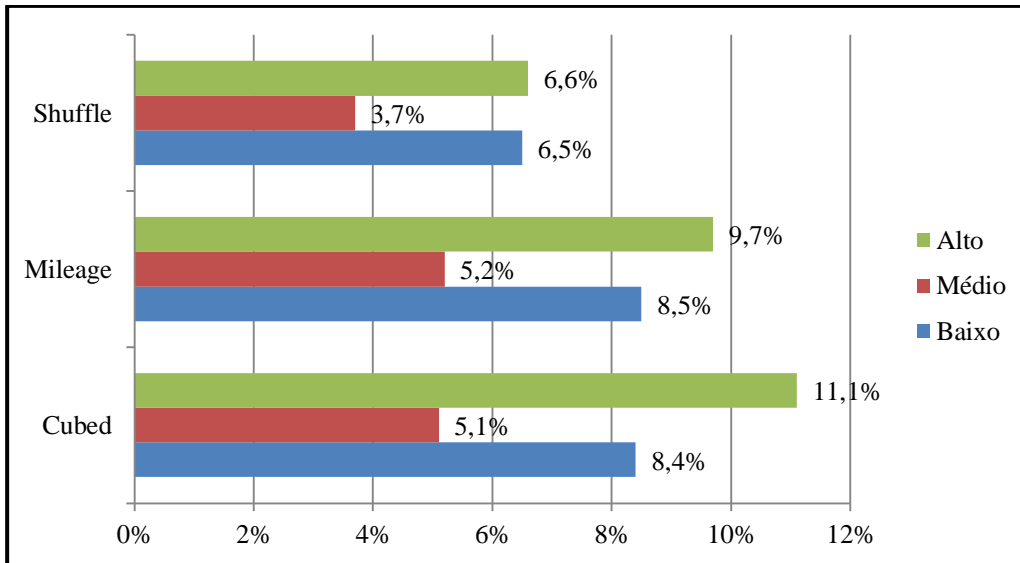


Figura 51 – Influência da luminosidade na taxa de erro

Outro aspecto que pode ser investigado é a influência da luminosidade com relação à velocidade de execução das tarefas. Na Figura 52, é apresentada a velocidade média de execução das tarefas, levando em consideração os aplicativos individualmente e as três variações possíveis de luminosidade no ambiente.

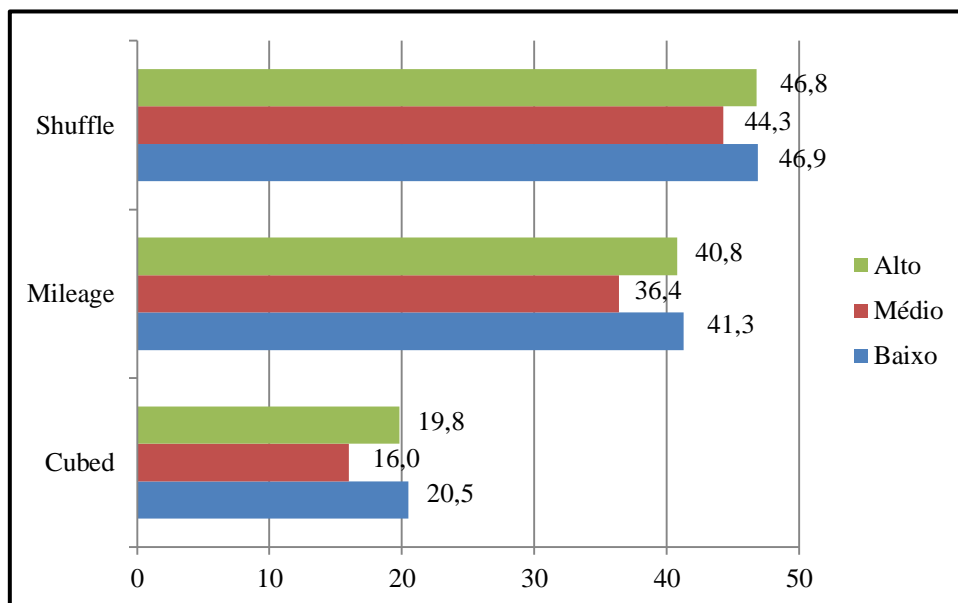


Figura 52 - Influência da luminosidade na velocidade (em seg.)

Novamente, pelos dados apresentados na Figura 52, fica comprovado que a intensidade da luminosidade também influencia na velocidade de execução das tarefas e, em condições normais de luminosidade no cenário de interação, os participantes do experimento obtiveram um desempenho melhor que em condições extremas.

A próxima análise está relacionada com a posição do *smartphone* durante as interações. O objetivo é encontrar problemas de usabilidade em determinadas tarefas referentes à posição de interação (vertical, horizontal ou mista). Na Tabela 30, é possível verificar as tarefas que tiveram uma taxa de erro acima de 10% relacionadas com a posição de interação. Informações desta natureza são úteis para os desenvolvedores da aplicação, já que em versões futuras do aplicativo podem ser inibidas as interações em posições com alta taxa de erros. A Tabela 30 mostra ainda que mais de 50% dos problemas ocorrem quando as tarefas são executadas na posição mista, ou seja, são iniciadas em uma posição e finalizadas em outra.

Tabela 30 – Relação de tarefas com alto índice de erro

Posição	Tarefas – Mileage	% Erros
Vertical	Cadastrar um veículo	16,7%
Misto	Cadastrar um veículo	14,3%
Vertical	Importar dados armazenados.	21,1%
Horizontal	Importar dados armazenados.	18,2%
Misto	Importar dados armazenados.	20,0%
Posição	Tarefas – ^3 (Cubed)	% Erros
Misto	Escolher uma música de uma lista de reprodução.	11,2%
Misto	Inserir uma música a uma lista de reprodução.	16,7%
Vertical	Modificar a aparência da aplicação.	10,6%
Horizontal	Escolher um novo tema de apresentação do aplicativo.	14,0%
Posição	Tarefas – Shuffle	% Erros
Misto	Inserção de uma nova atividade.	12,9%
Misto	Excluir todas as atividades da caixa de entrada.	14,3%
Vertical	Realizar uma cópia (<i>backup</i>) dos dados utilizando as preferências gerais do aplicativo.	14,3%
Misto	Acionar a opção de ajuda.	20,0%

A próxima avaliação refere-se à velocidade de deslocamento dos usuários no momento que realizam as suas interações. Normalmente, a velocidade varia em função de três possibilidades: usuário caminhando, usuário parado ou, ainda, utilizando algum meio de transporte. É apresentada, na Figura 53, a taxa de erros em função desta variável contextual.

Como pode ser observado na Figura 53, em todos os aplicativos é possível identificar que as ações realizadas sem deslocamento apresentam uma taxa de erro inferior às realizadas em movimento. Resultados semelhantes foram observados por Kjeldskov e Stage (2004) em um experimento controlado, realizado em laboratório, onde os participantes utilizaram uma esteira para se locomover enquanto interagiam com um aplicativo para dispositivos móveis.

Outro fato relevante com relação ao contexto de uso das aplicações, é que aproximadamente 70% das interações ocorrem com os usuários parados, com o dispositivo em uma única posição e com luminosidade do ambiente normal. Entretanto, quando estes fatores contextuais mudam, os usuários

cometem mais erros e levam mais tempo para a execução das tarefas. Estas informações sugerem que os aplicativos deveriam, por exemplo: (i) inviabilizar a interação em posições onde se constata maior probabilidade de erros, forçando o usuário a interagir na posição apropriada; (ii) detectar a luminosidade externa e tentar equilibrar a luminosidade irradiada pelo dispositivo a fim de garantir uma boa visualização; e (iii) identificar a movimentação do usuário e disponibilizar apenas as funcionalidades mais usuais, diminuindo a poluição visual.

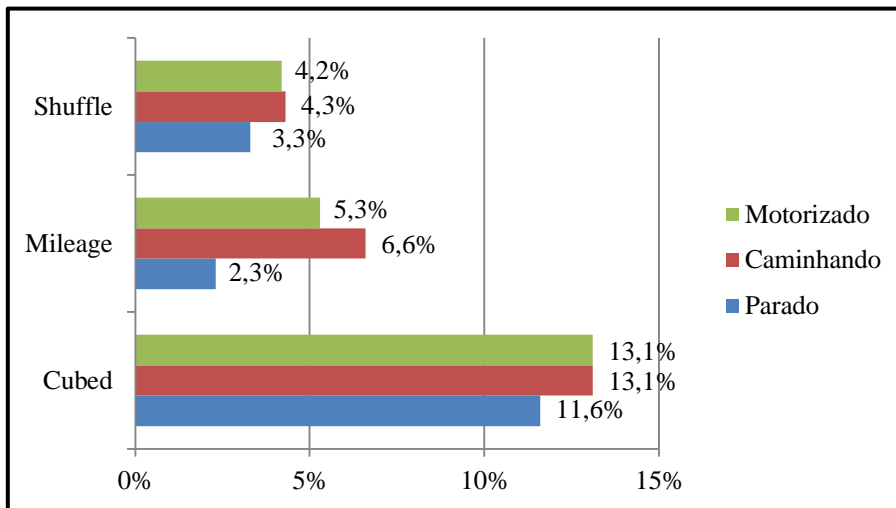


Figura 53 – Taxa de erro em função da velocidade de deslocamento

As próximas análises verificam a existência de interferência das variáveis contextuais relacionadas às características dos *smartphones*, tais como, a resolução e o tamanho da tela. Para realizar esta avaliação, foi investigada a velocidade de execução das tarefas em função das características dos *smartphones*. A Figura 54 apresenta os resultados referentes à resolução da tela.

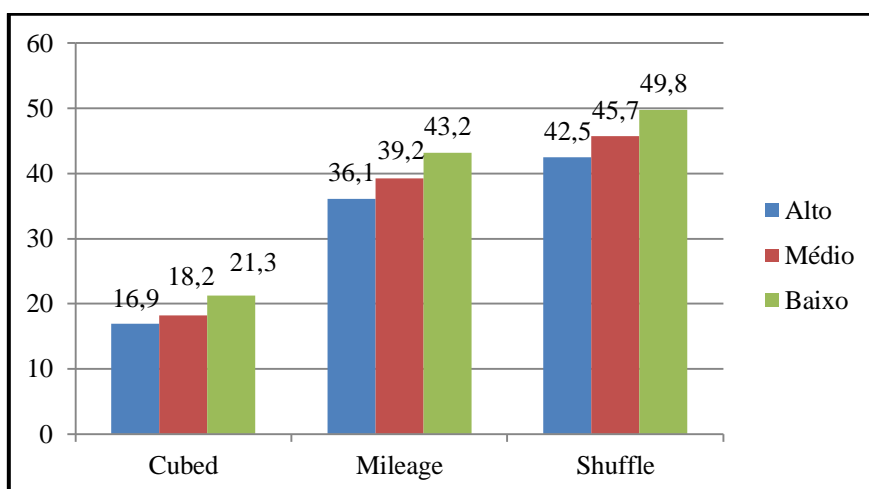


Figura 54 – Velocidade de Execução das tarefas em função da resolução da tela (em seg.)

Os dados apresentados no gráfico da Figura 54 possibilitam identificar que a resolução da tela influencia significativamente na velocidade de execução das tarefas, ou seja, quanto maior for a resolução, mais rapidamente as tarefas são concluídas. É possível verificar, no caso do aplicativo Cubed, que a resolução alta aumenta em média 26,03% a velocidade das tarefas quando comparado com a resolução baixa. No aplicativo Mileage, essa diferença é de 19,66% e, no Shuffle, é de 17,17%. A mesma análise realizada anteriormente foi concebida para a verificação da influência do tamanho da tela nas interações dos usuários. Os resultados podem ser observados na Figura 55.

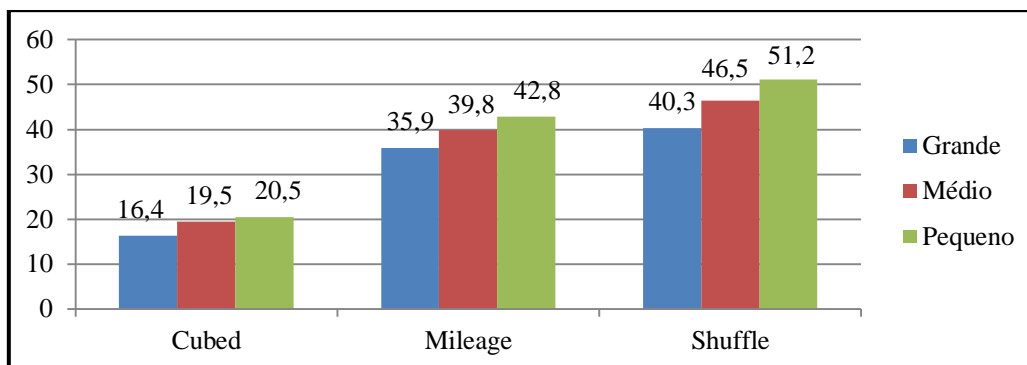


Figura 55 – Velocidade de execução das tarefas em função do tamanho da tela (em seg.)

Analisando o gráfico apresentado na Figura 55, é possível verificar que o tamanho da tela é outra variável contextual que tem influência no desempenho dos usuários de *smartphones*. No aplicativo Cubed, a velocidade média para a execução das tarefas diminuiu em torno de 4,1 segundos quando comparado com a utilização de *smartphones* com tela pequena. No caso do aplicativo Mileage, essa diferença é em torno de 6,9 segundos, e no caso do Shuffle, a diferença é de 10,9 segundos.

Um fato observado no mercado de *smartphones* é que, normalmente, os aparelhos com a tela menor também possuem baixa resolução. Desta forma, foi observado o desempenho dos usuários levando em conta as duas variáveis simultaneamente. A métrica utilizada para medir o desempenho foi o percentual de tarefas completadas com erro. O resultado da análise pode ser observado na Figura 56.

No gráfico da Figura 56, é possível constatar que quanto menores o tamanho e a resolução da tela dos *smartphones* mais erros são encontrados nas tarefas executadas. A diferença entre os extremos, ou seja, telas grandes com resolução alta em relação a telas pequenas com resolução baixa são de 9,3% de tarefas executadas com erro.

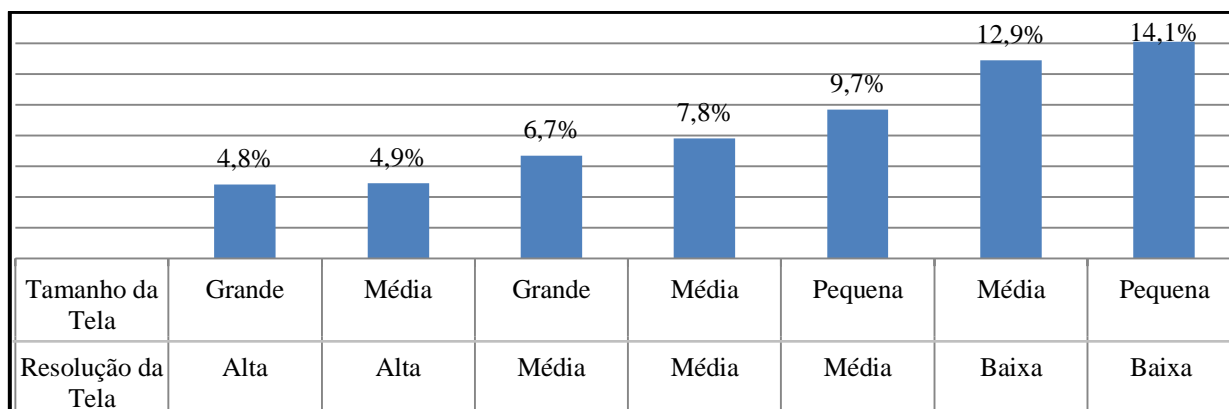


Figura 56 – Relação do tamanho e resolução da tela com o percentual de erros

Ao ser analisada a taxa de tarefas executadas com erros em conjunto com o perfil dos participantes, um fato intrigante foi observado. Ficou constatado que a incidência de erros na classe social de baixa renda é maior que nas demais classes. Para buscar uma explicação para este resultado, investigou-se o tipo de dispositivo utilizado por estes participantes no experimento. A constatação foi que a taxa de erros não estava relacionada ao poder aquisitivo dos usuários, mas à baixa resolução da tela dos dispositivos utilizados. Como a maioria das pessoas de baixa renda utilizaram *smartphones* com baixa resolução, analisando isoladamente a classe social, pode-se chegar a conclusões errôneas. Esta análise caracteriza uma das potencialidades da infraestrutura UXEProject já que permite associar diferentes fatores contextuais em uma única avaliação, diminuindo assim a possibilidade de conclusões erradas.

Na Figura 57, é apresentada a relação do poder aquisitivo e a resolução da tela dos *smartphones*. Pode ser observado que, independente do poder aquisitivo, os erros são mais frequentes quando os usuários utilizam *smartphones* com baixa resolução.

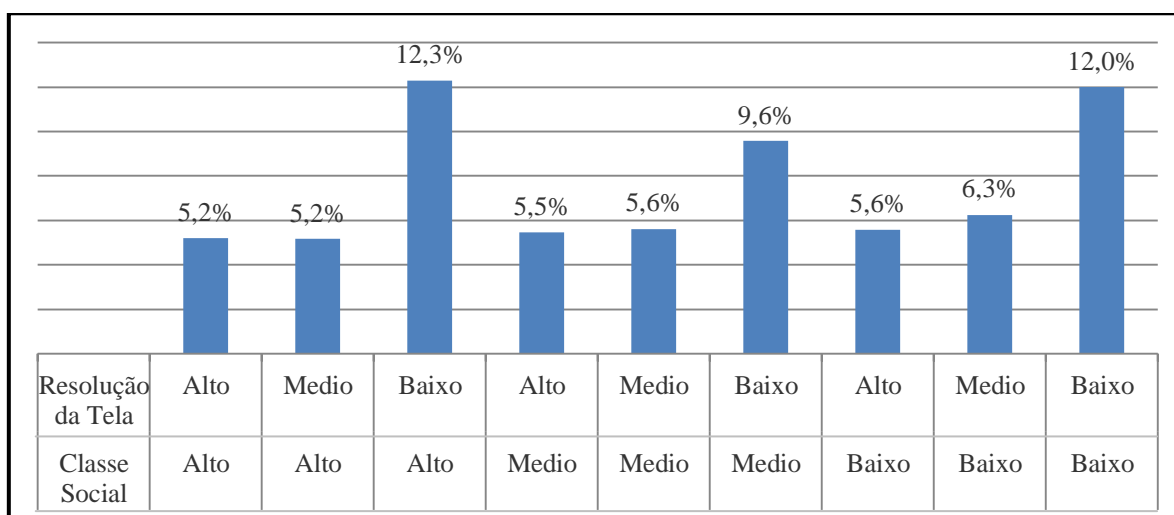


Figura 57 – Relação da taxa de erro em função do Poder Aquisitivo X Resolução da Tela

Nas demais análises referentes ao perfil dos participantes, não foram observados dados conclusivos que pudessem ser atribuídos especificamente a um perfil em particular.

5.6.6 Análise da satisfação dos usuários

Para avaliar a satisfação dos participantes no experimento, foram analisados os dados coletados por meio de interlocução direta com os usuários, por meio da técnica ESM. As perguntas propostas aos usuários estão na Tabela 31, sendo que a pergunta 8 não foi considerada para o aplicativo Cubed devido ao mesmo não possuir a funcionalidade de Ajuda. O objetivo aqui foi coletar dados subjetivos referentes aos sentimentos dos usuários com relação ao uso dos aplicativos. A análise desse tipo de informação permite detectar problemas que geram desconforto ou insatisfação aos usuários, bem como constatar o que mais atrai os usuários para a inserção dos aplicativos em seu cotidiano. As perguntas são apresentadas durante as interações dos usuários com os aplicativos, em intervalos aleatórios de tempo.

Tabela 31 – Perguntas submetidas aos usuários

ID	Perguntas
1	Qual a sua satisfação com o número de ações necessárias para realizar as tarefas propostas?
2	Qual a sua satisfação com a interface?
3	Qual a sua satisfação com a aprendizagem de novas funcionalidades?
4	Qual a sua satisfação com a utilização deste aplicativo em seu cotidiano?
5	Qual a sua satisfação com as funcionalidades do aplicativo?
6	Como você se sentirá ao utilizar este aplicativo novamente?
7	Qual a sua satisfação em relação a comunicabilidade do aplicativo?
8	Qual a sua satisfação com a ajuda disponibilizada?

As três figuras a seguir apresentam os percentuais referentes às respostas dos 21 usuários que participaram do experimento durante os seis primeiros meses.

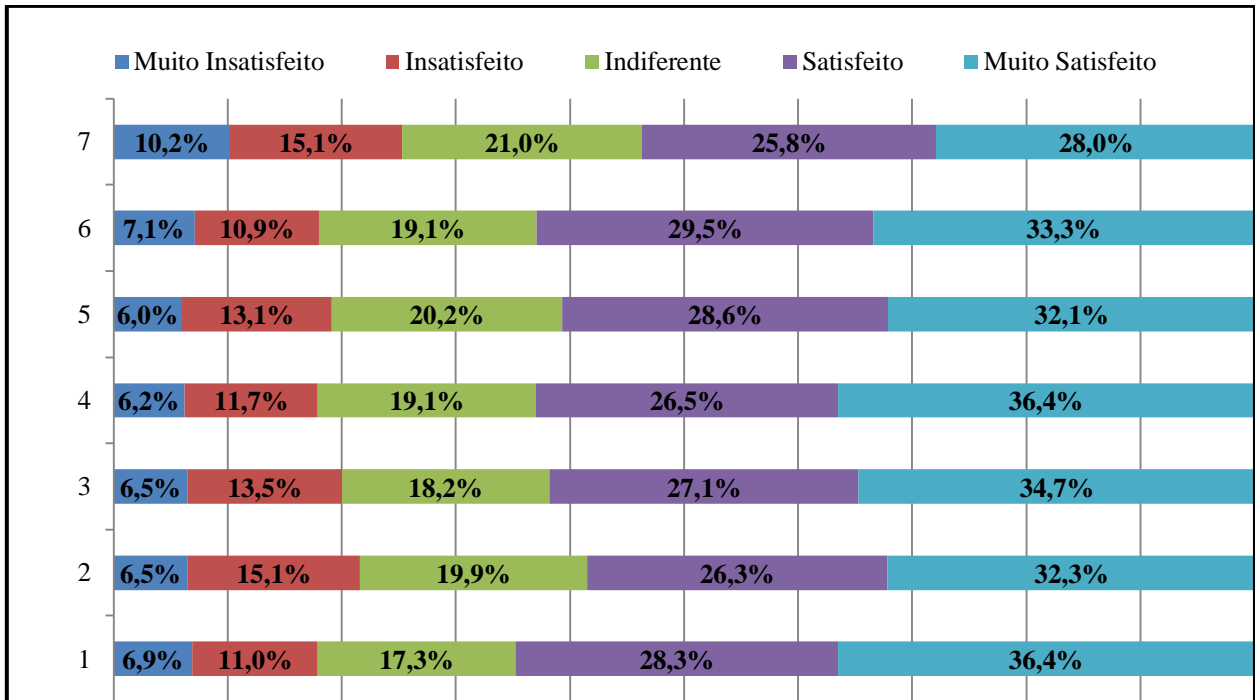


Figura 58 – Respostas associadas ao Aplicativo Cubed

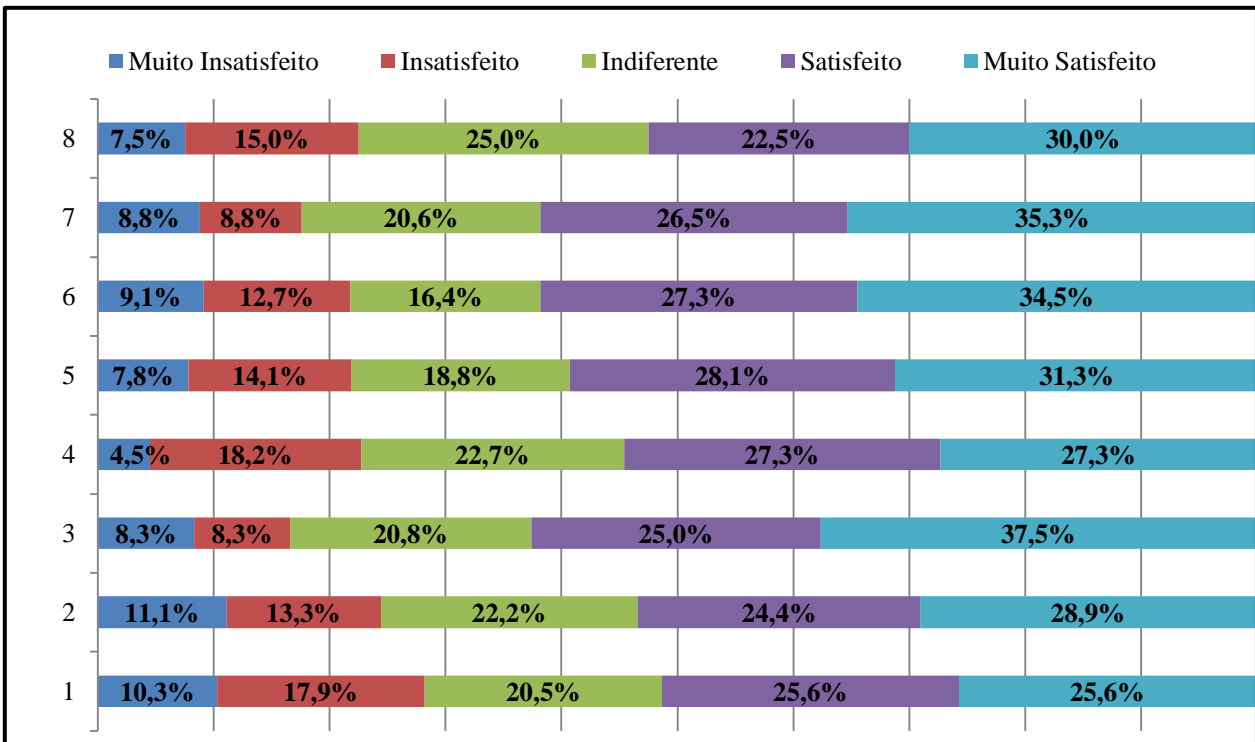


Figura 59 – Respostas associadas ao Aplicativo Mileage

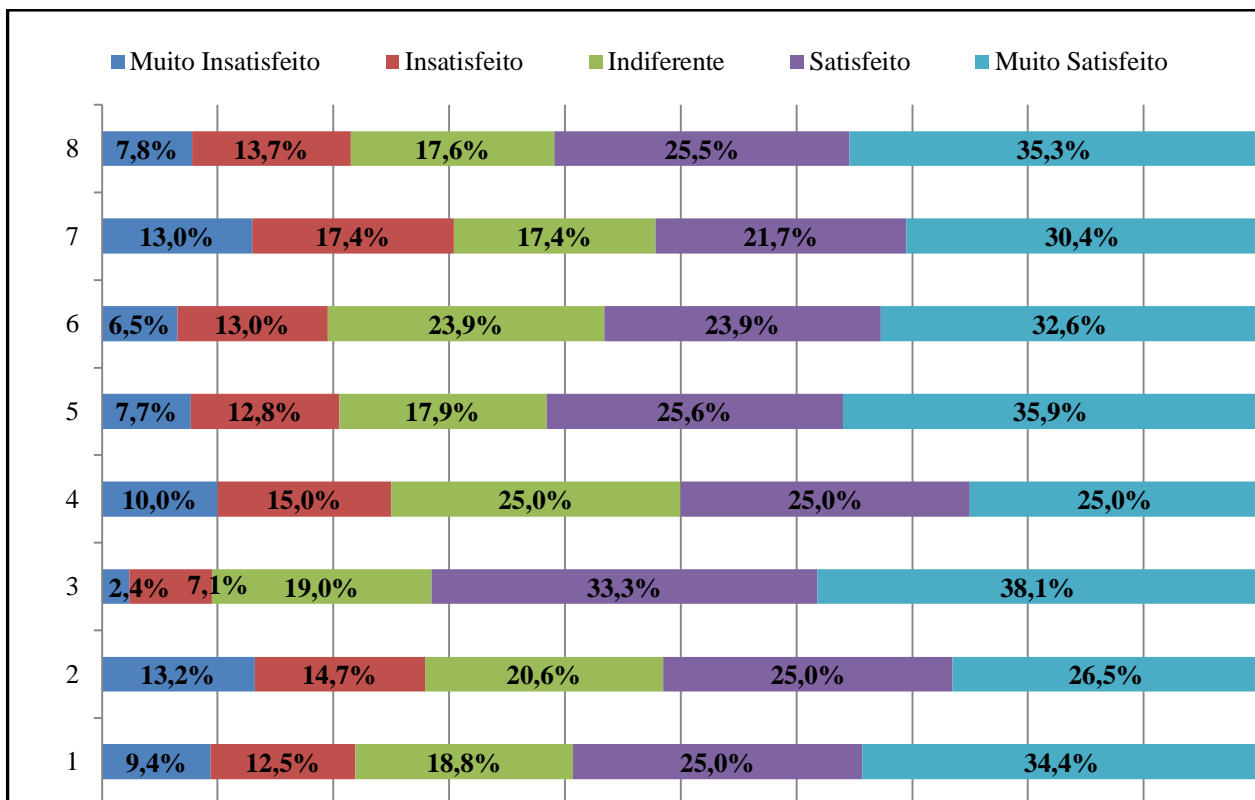


Figura 60 – Respostas associadas ao Aplicativo Shuffle

O fato de destaque nas informações apresentadas nas três figuras é que a distribuição do grau de satisfação se mantém uniforme e crescente do muito insatisfeito para o muito satisfeito. Esta tendência é constatada em todas as perguntas submetidas aos usuários nos três aplicativos.

Outro ponto relevante é que o grupo de usuários satisfeitos ou muito satisfeitos é superior a 50% em todos os quesitos avaliados. Este pode ser um dos motivos do sucesso do experimento durante os seis primeiros meses, já que foi mantido o interesse da maioria dos usuários durante este período. Para alcançar este objetivo é fundamental que os aplicativos sejam interessantes, úteis, atrativos e de fácil interação.

Após os seis primeiros meses, ocorrem algumas evasões no grupo de usuários, restando ao final de um ano apenas 9 participantes no experimento. Sendo assim, foi realizada uma análise abrangendo apenas a Pergunta 6 – Como você se sentirá ao utilizar este aplicativo novamente? (Figura 61).

O objetivo desta análise é correlacionar o nível de insatisfação dos participantes com os três aplicativos, durante o tempo em que permaneceram como integrantes do experimento, permitindo identificar se existe relação da taxa de evasão com o grau de insatisfação. Para isso, foi observado o percentual de vezes em que cada um dos participantes escolheu como resposta às opções: “muito insatisfeito” ou “insatisfeito”.

Em função dos dados apresentados na Figura 61, pode ser observado que, para os 9 participantes que concluíram o experimento, a soma da média da insatisfação não ultrapassa 40%, enquanto que para os outros 12 participantes esse índice ficou acima de 85%. Embora esses dados representem

indícios do motivo da evasão no experimento, é importante destacar que nenhum participante atribuiu mais que 50% de insatisfação a um aplicativo específico, logo havia mais satisfeitos do que insatisfeitos com os aplicativos, como foi detectado anteriormente nesta seção.

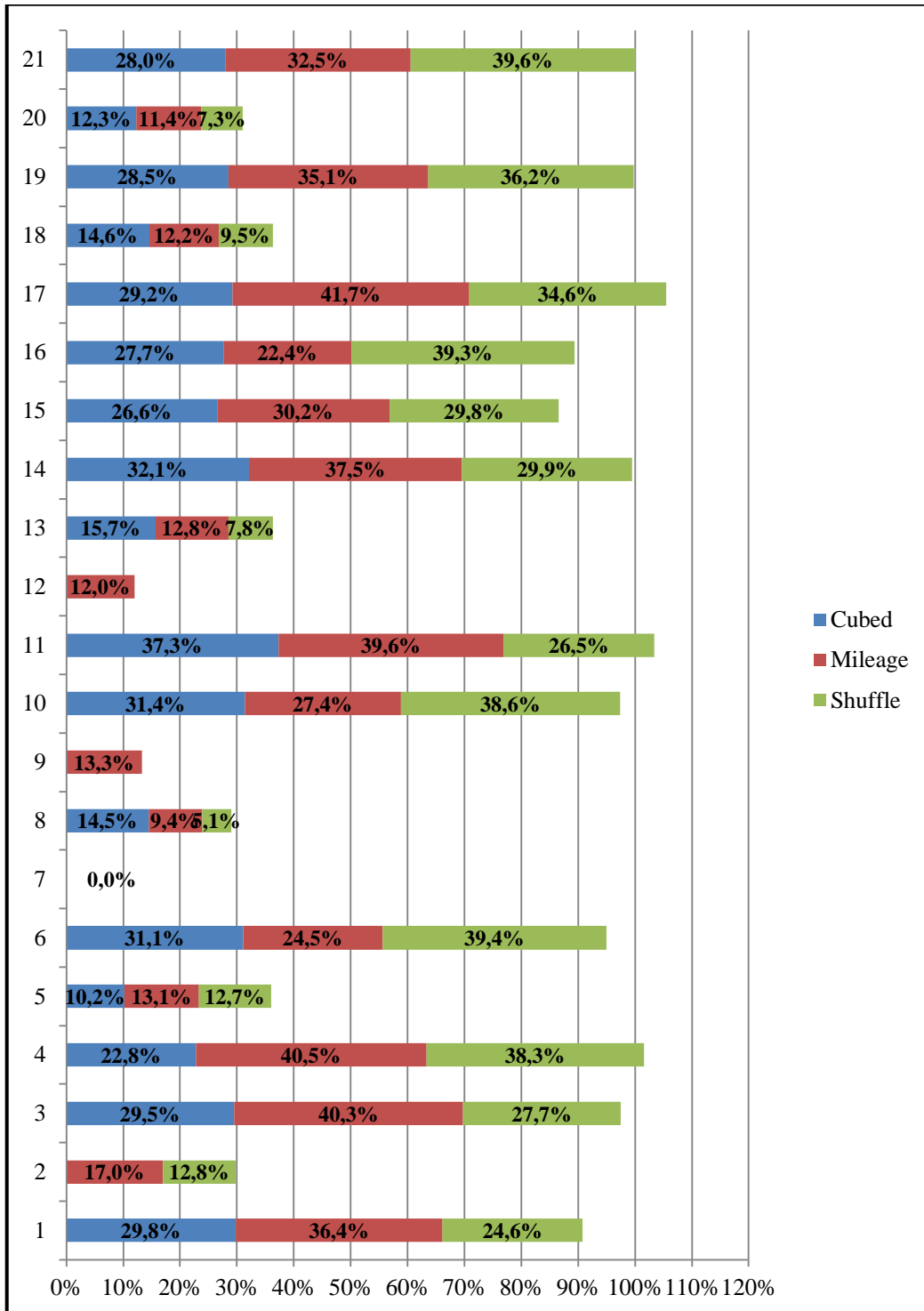


Figura 61 – Análise da insatisfação dos participantes com os aplicativos

5.6.7 Análise dos Valores de Referência X Dados Reais

Nesta subseção será apresentada outra forma de avaliar as tarefas tendo como parâmetro de comparação a visão da Equipe de Avaliação. Tomando como base os dados de referência definidos no momento do mapeamento das tarefas, é possível analisar o desempenho real dos participantes do experimento com relação ao que a Equipe de Avaliação definiu como sendo um comportamento aceitável para cada tarefa.

As métricas utilizadas nesta análise foram: (i) o tempo médio para a execução da tarefa; (ii) o percentual de erro aceitável para cada tarefa; (iii) o número médio de ações considerado admissível para completar a tarefa; (iv) o percentual de acesso a opção de Ajuda; e (v) o percentual de erros após ser acionada a opção de Ajuda para concluir a tarefa.

Os dados reais utilizados para esta análise são referentes aos seis primeiros meses do experimento, com a participação dos 21 usuários. Os dados de referência para cada tarefa foram especificados no momento do seu mapeamento, baseados no desempenho de um dos integrantes da Equipe de Avaliação que previamente já utilizava os aplicativos por dois meses.

A familiarização com os aplicativos, por parte da Equipe de Avaliação, é importante para definir quais tarefas devem ser mapeadas e atribuir valores de referência que estejam próximos da realidade. Além disso, a colaboração da Equipe de Desenvolvimento do aplicativo pode ser uma boa contribuição para este processo, entretanto, vale salientar que, caso um dos objetivos da avaliação seja mensurar a eficiência da Equipe de Desenvolvimento, será prudente não permitir a sua participação no processo.

Na Tabela 32, pode ser observada a comparação dos valores de referência com relação aos dados reais coletados durante a interação dos participantes com o aplicativo Cubed. Inicialmente, é possível identificar que o desempenho real dos participantes com relação à velocidade para executar as tarefas foi maior do que o previsto, com destaque para a Tarefa 3, numa diferença de 9,4 segundos (47% a mais do que o previsto), o que indicaria um problema de eficiência da tarefa.

Levando em consideração a taxa de erro de 10%, pode ser constatado que as Tarefas 2, 3 e 5 ultrapassam este valor. Esse fato vai de encontro à análise apresentada na Tabela 24, onde foi realizada a mesma avaliação tomando por base o percentual de tarefas completadas com erro, sendo constatado que todas as tarefas do aplicativo Cubed não apresentavam problemas de usabilidade. Desta forma, observa-se a importância de avaliar o número de ações erradas, já que para a execução de uma tarefa pode ocorrer mais de uma ação sem conformidade com o previsto.

Tabela 32 – Valores de Referência x Dados Reais (Cubed)

ID	Tempo (seg.)		Erros (%)		Ações (Média)	
	Ref.	Real	Ref.	Real	Ref.	Real
1	12	16,6	10%	8,8%	2	2,3
2	22	26,3	10%	14,8%	3	3,3
3	20	29,4	10%	15,2%	4	4,3
4	7	12,2	10%	7,9%	2	2,4
5	7	12,0	10%	13,0%	2	2,3
6	10	16,6	10%	5,6%	4	4,2

A métrica referente ao número médio de ações está diretamente relacionada com o número de ações sem conformidade, desta forma, a Tarefa 2 com 14,8%, a Tarefa 3 com 15,2% e a Tarefa 5 com 13% são passíveis de melhorias. É importante salientar que o atual formato definido para o mapeamento das tarefas não prevê que o usuário interaja de forma diferente da prevista pela Equipe de Avaliação, ou seja, caso uma tarefa tenha mais de um caminho para ser executada será necessário mapear todas as possibilidades.

Na Tabela 33, pode ser observada a comparação dos valores de referência com relação aos dados reais coletados durante a interação dos participantes com o aplicativo Mileage. Observa-se inicialmente que os participantes demoram mais que o previsto para executar as tarefas, com destaque principal para a Tarefa 8, com uma diferença de 12,7 segundos (47% a mais do que o previsto).

Tabela 33 – Valores de Referência x Dados Reais (Mileage)

ID	Tempo (seg.)		Erros (%)		Ações (média)		Ajuda (%)		Erros após Ajuda (%)	
	Ref.	Real	Ref.	Real	Ref.	Real	Ref.	Real	Ref.	Real
1	57	62,4	10%	14,9%	7,0	8,0	5%	12,9%	5%	29,3%
2	55	60,5	10%	8,7%	7,0	7,6	5%	7,2%	5%	19,8%
3	15	19,6	10%	8,0%	3,0	3,2	5%	0,0%	5%	0,0%
4	34	36,7	10%	5,7%	5,0	5,3	5%	15,8%	5%	33,0%
5	60	61,7	10%	8,4%	12,0	13,0	5%	12,8%	5%	15,7%
6	12	17,8	10%	3,5%	2,0	2,1	5%	0,0%	5%	0,0%
7	12	17,6	10%	3,3%	2,0	2,1	5%	0,0%	5%	0,0%
8	27	39,7	10%	23%	4,0	4,9	5%	14,5%	5%	57,7%

Com relação à métrica que contabiliza os erros, é possível observar que os resultados são os mesmos apresentados na Tabela 23, ou seja, as Tarefas 1 e 8 devem ser reestruturadas em função de exibirem uma taxa de erro superior a 10%. Como esse resultado reflete diretamente no número de ações necessárias para executar as tarefas, na métrica que contabiliza as ações são observadas as mesmas diferenças percentuais com relação ao que foi previsto e o que ocorreu na prática.

A métrica Ajuda permite contabilizar o número de vezes que a opção de Ajuda foi acionada durante a execução de uma tarefa, possibilita identificar quais tarefas deixam dúvidas sobre as etapas para a sua execução ou qual o seu objetivo. É possível visualizar, na Tabela 33, que todas as tarefas para as quais a opção de Ajuda foi acionada tiveram um percentual acima dos 5%, sendo assim, maior que o valor estipulado como normal pela Equipe de Avaliação.

Outro fato a ser destacado, é a frequência de erros para completar a tarefa após acionar a opção de Ajuda que apresenta um percentual bem acima do estipulado pela Equipe de Avaliação, o que comprova a necessidade do recurso apresentar passos detalhados de como executar as tarefas.

Tabela 34 – Valores de Referência X Dados Reais (Shuffle)

ID	Tempo (seg.)		Erros (%)		Ações (Média)		Ajuda (%)		Erros após Ajuda (%)	
	Ref.	Real	Ref.	Real	Ref.	Real	Ref.	Real	Ref.	Real
1	75	98,7	10%	12,6%	9	10,1	5%	7,0%	5%	23,6%
2	15	22,5	10%	3,0%	3	3,1	5%	8,1%	5%	18,3%
3	42	52,3	10%	0,0%	5	5,0	5%	12,3%	5%	0,0%
4	35	41,5	10%	0,0%	5	5,0	5%	13,2%	5%	0,0%
5	31	39,8	10%	2,4%	6	6,1	5%	6,5%	5%	10,7%
6	5	8,4	10%	0,0%	2	2,0	5%	7,9%	5%	0,0%
7	46	56,1	10%	8,3%	4	4,3	5%	3,2%	5%	0,0%
8	9	11,0	10%	7,2%	4	4,3	5%	6,7%	5%	15,3%
9	5	7,9	10%	7,1%	2	2,1	5%	0,0%	5%	0,0%
10	81	121,8	10%	18,4%	7	8,3	5%	6,6%	5%	36,1%

A última análise realizada nesta subseção refere-se ao aplicativo Shuffle, cujos dados apresentados na Tabela 34 demonstram grande discrepância entre os tempos estipulados para a execução das tarefas e o que ocorre na prática. Uma possível explicação para isso está na habilidade e rapidez dos participantes, pois as tarefas neste aplicativo, na maioria das vezes, requerem a descrição das atividades diárias dos usuários. Outro fato relevante é que alguns usuários são mais prolixos que outros para descreverem as suas tarefas diárias, desta forma, é compreensível a diferença de tempo.

Com relação ao percentual de erros, é possível observar que as Tarefas 1 e 10 extrapolam os 10% estipulados pela Equipe de Avaliação. Com relação à Tarefa 10, este problema já tinha sido relatado na Tabela 25, na qual ocorreu a investigação das tarefas completadas com erro. Entretanto, a taxa de erro encontrada para a Tarefa 1 é um fato novo e importante para a usabilidade do aplicativo, já que esta é a uma das tarefas mais executadas pelos usuários.

Com relação à opção de Ajuda, é possível observar a alta taxa de erro após a utilização desse recurso, o que indica necessidade de reestruturação desta opção, inclusive com o estudo dos passos para acionar o recurso e, após a sua utilização, voltar a executar a tarefa do local onde o usuário estava.

5.7 RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO

Neste capítulo foram apresentadas as etapas realizadas para a condução de um experimento em campo com duração de um ano. Foi analisada a usabilidade de três aplicativos – Mileage, Cubed e Shuffle – para *smartphones*, com o objetivo de identificar as potencialidades da infraestrutura UXEProject.

As etapas executadas no experimento seguiram as orientações do *framework* DECIDE. Inicialmente, foram definidos os objetivos da avaliação e elaboradas as oito perguntas a serem respondidas no estudo. Posteriormente, foi adotada a infraestrutura UXEProject para a condução do experimento e tratadas as questões práticas, tais como, escolher os aplicativos que seriam utilizados, recrutar os participantes e definir os dados a serem coletados. Por fim, foi decidido como tratar as questões éticas e avaliar os resultados.

Para identificar as tarefas com problemas de usabilidade, foi analisado o percentual de tarefas completadas com erro em cada aplicativo durante os seis primeiros meses do experimento. Posteriormente, foram analisadas as tarefas que apresentaram percentuais acima de 10% durante os doze meses de experimento para comprovar as constatações preliminares. O resultado encontrado foi que as tarefas para cadastrar um veículo e importar dados armazenados (aplicativo Mileage) apresentam problemas de usabilidade e necessitam de uma reestruturação.

Com o objetivo de analisar o desempenho dos usuários ao longo do tempo, foram avaliados os percentuais de tarefas completadas com erro e o tempo médio para a execução das tarefas durante os seis primeiros meses e, posteriormente, durante os doze meses. As principais conclusões obtidas foram que o aprendizado e a familiarização dos usuários ocorrem nos dois primeiros meses, mantendo-se constante nos meses subsequentes.

Na análise referente à frequência de utilização da ajuda, ficou constatada a maior utilização deste recurso no segundo e terceiro mês do experimento e que isso contribuiu para melhorar o desempenho dos usuários. Por outro lado, foi observada a ocorrência de uma alta taxa de erros entre o acionamento da ajuda e o término da tarefa, logo o recurso não está devidamente projetado para auxiliar os passos de execução da tarefa ou esta funcionalidade apresenta uma falha estrutural, fazendo com que os usuários se atrapalhem.

Na comparação entre usuários Comuns e Experientes, ficou constatada uma diferença significativa de desempenho nos primeiros meses, entretanto, com o passar do tempo, os usuários Comuns se aproximam dos resultados apresentados pelo usuário Experiente. Outro fato interessante é que, após longo

tempo de utilização dos aplicativos a curva de aprendizagem se estabiliza e o desempenho se mantém constante.

Para identificar a influência das variáveis contextuais na usabilidade dos aplicativos, foram realizadas várias investigações:

- A primeira refere-se à luminosidade, sendo constatada a interferência no desempenho dos usuários quando a interação ocorre em lugares com luminosidade muito baixa ou muito alta, influenciando o aumento da incidência de erros e o tempo para completar as tarefas.

- A segunda análise avalia as posições do *smartphone* durante a interação dos usuários. Para uma constatação dos problemas de usabilidade, foram identificadas as tarefas que apresentam uma taxa de erro acima de 10% em função da posição de interação. Ficou constatado que mais de 50% dos problemas ocorrem na posição mista, ou seja, quando ocorrem mudanças entre as posições vertical e horizontal.

- A terceira avaliação refere-se à velocidade de deslocamento dos usuários no momento de realizar as suas interações, sendo observado que ao estarem em movimento cometem mais erros e, em geral, demoram mais tempo para completar as tarefas.

- A quarta análise verifica a interferência da resolução da tela dos dispositivos no desempenho dos participantes, sendo constatada uma perda de velocidade na execução das tarefas em torno de 22% entre os dispositivos com alta resolução para os *smartphones* de baixa resolução.

- O quinto estudo refere-se à interferência do tamanho da tela no desempenho dos usuários, ficando comprovado que a perda de desempenho dos participantes utilizando tela pequena em relação aos que utilizaram tela grande é de 21,3% no aplicativo Cubed, 19,6% no Mileage e 29,18% no Shuffle, comprovando a interferência desta variável contextual na usabilidade.

- A sexta análise verificou as configurações dos *smartphones* em conjunto, já que é comum encontrar no mercado dispositivo com tela pequena com baixa resolução e dispositivos com tela grande com alta resolução. Para realizar esse estudo, foi utilizado o percentual de erros e observado que a melhor configuração tem uma taxa de 10,8% a menos de erros quando comparada com a pior configuração.

- A última avaliação referente às variáveis contextuais envolveu o perfil dos participantes e não foram observados dados que justificassem problemas de usabilidade, exceto com relação à classe social. Nesse caso, foi detectado que participantes da classe social baixa apresentavam uma taxa maior de erro quando comparados às demais classes. Em busca de uma explicação para esse fato, foi realizada uma investigação contemplando a classe social juntamente com a resolução da tela dos *smartphones*, sendo constatado que esta variável teve a maior influência no resultado anterior. Nesse caso, a queda de rendimento por parte da classe social mais baixa se explica pela configuração de seus aparelhos, os quais possuem uma menor resolução da tela.

Na análise referente à satisfação dos usuários, utilizando interlocuções diretas, por meio da técnica ESM, ficou constatado que a maioria dos participantes estava satisfeita com a usabilidade dos aplicativos em teste. Entretanto, como a taxa de evasão após os primeiros seis meses de experimento foi maior do que 50% avaliou-se a satisfação do usuário com relação a uma pergunta específica: “Como você se sentirá ao utilizar este aplicativo novamente?” O resultado encontrado explica a evasão dos 12 participantes, já que a soma dos percentuais nos três aplicativos, considerando as respostas, “muito insatisfeito” e “insatisfeito”, ficaram acima de 85%, mais que o dobro quando comparado com os 9 participantes que permaneceram até o final do experimento.

A última análise compara os valores de referência estipulados pela Equipe de Avaliação com os dados reais coletados durante o experimento. As métricas utilizadas para as comparações foram, o tempo médio para executar as tarefas, o percentual de erros cometidos, o número médio de ações realizadas em cada tarefa, o percentual de utilização da opção de Ajuda dos aplicativos e o percentual de erros entre a utilização da ajuda e o término da tarefa. Os resultados encontrados nesta avaliação foram:

- Em todas as tarefas, a velocidade dos participantes para executá-las ficou a quem do estimado pela Equipe de Avaliação.
- Foram detectadas novas tarefas com problemas de usabilidade que nas análises anteriores não tinham sido observadas.
- O número de ações dos participantes para completar as tarefas está vinculado ao número de acertos e erros para a execução de uma tarefa. Esta é uma limitação da infraestrutura UXEProject, já que tarefas executadas de forma diferente do seu mapeamento não são consideradas, limitando a avaliação dos passos previstos pela Equipe de Avaliação e desprezando, assim, a possibilidade de avaliar se os usuários encontraram maneiras mais eficazes de executar uma determinada tarefa.
- Ficou comprovado que a opção de Ajuda deve ser revista nos aplicativos, já que foram detectadas altas taxas de erro após os usuários voltarem para concluir a tarefa em curso, muito acima do que foi previsto pela Equipe de Avaliação.

Para finalizar, foi constatado que a infraestrutura UXEProject se apresenta como uma excelente opção para a execução de experimentos como o objetivo de avaliar a usabilidade de aplicativos para *smartphones*, possibilita análises de diferentes ângulos, envolve várias métricas para investigar o desempenho dos participantes, com a possibilidade de cruzar diferentes dados na análise dos resultados. Sendo assim, conclui-se que a infraestrutura UXEProject é uma contribuição relevante para a área de IHC, ficando comprovadas as suas potencialidades, além de não terem sido encontrados trabalhos similares com tantos recursos para a captura e análise dos dados.

Este capítulo abordará a avaliação experimental dos processos associados ao uso da infraestrutura UXEProject. É importante para trabalhos dessa natureza que os passos necessários para o uso da infraestrutura sejam avaliados por seus potenciais usuários: os avaliadores de softwares para smartphones. Sendo assim, inicialmente, foi realizado um experimento com alunos de pós-graduação na área de Engenharia de Software para comparar a abordagem proposta com a forma tradicional de realizar as mesmas tarefas de preparação do código para avaliação de usabilidade. Posteriormente, foi dada continuação do experimento com alunos de pós-graduação em banco de dados, para avaliar o processo de montagem da infraestrutura de serviço na nuvem e a descoberta de conhecimento na base de dados gerada. Os resultados serão apresentados e discutidos ao longo do capítulo.

6 AVALIAÇÃO DA ABORDAGEM POR DESENVOLVEDORES DE SOFTWARE

Após comprovar, nos capítulos anteriores, as potencialidades da infraestrutura e do modelo para permitir a captura automática de dados de interação, o foco nesse capítulo é a avaliação da abordagem proposta do ponto de vista dos seus potenciais usuários: os desenvolvedores de software e os avaliadores de aplicações. O ponto de partida dessa avaliação é a formalização dos processos relacionados a cada uma das etapas da abordagem proposta. Estes processos estão relacionados à utilização das ferramentas da infraestrutura UXEProject que possibilitam: (i) realizar o mapeamento das tarefas; (ii) gerar automaticamente as métricas de rastreabilidade e (iii) instrumentar os aplicativos com tais métricas.

A ferramenta escolhida para a descrição dos processos envolvidos para a concepção e utilização da infraestrutura UXEProject foi o EPF Composer. Esta ferramenta foi criada pela IBM, desenvolvida com base no Eclipse IDE e disponibilizada para a comunidade de forma livre, sendo destinada especialmente para engenheiros de processos, gerentes de projeto ou qualquer papel responsável por manter e implementar processos organizacionais ou processos individuais que envolvam atividades e papéis. O uso do EPF Composer permitiu descrever formalmente, do ponto de vista da Engenharia de Software, os processos envolvidos para operacionalizar a utilização da infraestrutura UXEProject.

Para avaliar os processos descritos, foi realizado um experimento dividido em duas etapas. A primeira, com desenvolvedores de software, para validar os passos realizados na Unidade de Mapeamento e na Unidade de Rastreabilidade. A segunda, com estudantes da área de BD, para avaliar as etapas necessárias

para a criação e análise dos dados na Unidade de Avaliação. Para a condução do experimento descrito neste capítulo, novamente foi utilizado o *framework* DECIDE, com a execução das seis fases que envolvem a preparação e realização do experimento. Nas próximas seções, serão abordadas as etapas referentes à realização do experimento de validação.

6.1 FASE 1 – DETERMINAR OS OBJETIVOS DA AVALIAÇÃO

O principal objetivo a ser alcançado no experimento é avaliar a execução da abordagem proposta nesta tese, comparando a forma de instrumentar os aplicativos utilizando as ferramentas que fazem parte da infraestrutura UXEProject com a abordagem tradicional, na qual as métricas de usabilidade, contexto e subjetivas são codificadas pela Equipe de Avaliação ou de Desenvolvimento. Além disso, é necessário avaliar como a infraestrutura foi concebida para a análise dos dados e a facilidade com que os usuários podem descobrir informações relevantes para as suas análises. Nesse contexto, os potenciais usuários são pessoas inseridas nas Equipes de Desenvolvimento ou de IHC, portanto, relacionadas à área de Engenharia de Software, com funcionalidades que abrangem a codificação das métricas para a captura de dados, administração do BD e avaliação da usabilidade.

Embora os códigos relacionados com o mapeamento de tarefas e rastreabilidade das informações sejam gerados automaticamente, inúmeros processos precisam ser realizados para a execução correta da nova abordagem. Além disso, vários processos também são necessários para a criação e utilização do ambiente de análise dos dados. Sendo assim, é necessário documentar estes processos e avaliar a sua eficiência e eficácia do ponto de vista de quem potencialmente poderá utilizar a infraestrutura ou até mesmo reproduzi-la.

De uma forma mais geral, pretende-se detectar os problemas enfrentados pelos usuários ao utilizar a infraestrutura UXEProject, permitindo, assim, identificar processos que precisam ser remodelados e melhorados, bem como, constatar os processos que estão de acordo com as expectativas dos usuários.

Com os resultados observados, será possível avaliar os benefícios da nova abordagem para a área de IHC sob a ótica dos principais interessados, os especialistas na área possibilitando, assim, incentivar a adoção da abordagem em larga escala.

6.2 FASE 2 – EXPLORAR PERGUNTAS A SEREM RESPONDIDAS

Para atingir os objetivos apresentados na seção anterior, foi elaborado um conjunto de perguntas que direcionam o experimento e devem ser respondidas pelos participantes ao final da sua execução. Nos Anexos IX e X, são apresentados os questionários completos, entretanto, para a exemplificação das perguntas propostas aos usuários, seguem abaixo as mais relevantes para a análise desejada.

- 1) Como você classificaria o processo de geração automática dos Aspectos que permite o mapeamento das tarefas com o uso da ferramenta MAG?
- 2) Como você classificaria a atividade de captura automática dos métodos que é utilizado para a instrumentação das métricas de rastreabilidade com o uso da ferramenta ATD?
- 3) Em que nível de dificuldade você classificaria o conjunto de etapas do processo para realizar o mapeamento das tarefas?
- 4) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que dá suporte ao conjunto de etapas do mapeamento das tarefas?
- 5) Como você classificaria a forma de identificar, nas aplicações, as tarefas que necessitam ser avaliadas (Abordagem Tradicional)?
- 6) Como você classificaria o processo de geração automática dos Aspectos que permite a incorporação das métricas de rastreabilidade com o uso da Ferramenta UXE Metrics Generator?
- 7) Como você classificaria a forma de codificação dos Aspectos de Rastreabilidade utilizando a Abordagem Tradicional, ou seja, escrever o código de todos os Aspectos?
- 8) Como você classificaria a execução de todos os processos relacionados com a execução da Abordagem Proposta.
- 9) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que auxilia todos os passos para a utilização da Abordagem Proposta?
- 10) Como você classificaria a execução de todos os processos para a codificação dos Aspectos usando a Abordagem Tradicional.
- 11) Como você classificaria a execução dos procedimentos para instalar as ferramentas Data Load e ETL Maker que fazem a carga dos dados no BD e no DW.
- 12) Como você classificaria a execução dos procedimentos para preparar a ferramenta OLAP para realizar as avaliações dos dados de usabilidade?
- 13) Como você classificaria a execução do processo de análise dos dados após todo o ambiente estar preparado?

Com base nas respostas pretende-se avaliar cinco aspectos principais:

- Analisar a descrição dos processos. As respostas para esta avaliação variam entre muito insatisfeito a muito satisfeito.
- Identificar o nível de dificuldade para a execução dos passos necessários para a instrumentação dos aplicativos com a infraestrutura UXEProject. As respostas desta avaliação variam entre muito difícil a muito fácil.
- Comparar a abordagem tradicional, na qual os desenvolvedores precisam programar o código das métricas, com a abordagem proposta neste trabalho, onde as métricas são criadas automaticamente.
- Investigar a descrição dos passos necessários para criar o ambiente de análise dos dados definido na infraestrutura UXEProject.
- Avaliar o nível de dificuldade enfrentado pelos participantes do experimento para realizar os processos envolvidos.

6.3 FASE 3 – ESCOLHER O MÉTODO DE AVALIAÇÃO

Nesta fase, ficou decidido que o experimento seria conduzido em sala de aula, com alunos de pós-graduação e dividido em duas fases: a primeira com estudantes na área de Engenharia de Software, mais especificamente na disciplina de Programação Orientada a Aspectos; a segunda fase, com estudantes na área de BD, na disciplina Descoberta de Conhecimento em Bases de Dados.

O objetivo do experimento ser desmembrado em duas fases foi em função dos assuntos tratados nas disciplinas e a interligação desses assuntos com partes diferentes da infraestrutura. A primeira fase está relacionada com o mapeamento das tarefas e instrumentação das aplicações com as métricas de rastreabilidade, enquanto que a segunda fase está relacionada com a estrutura criada na nuvem para o armazenamento e análise dos dados.

Ficou definido que o experimento seria executado no final de cada curso, após os alunos terem visto todos os conteúdos previstos para as disciplinas. A escolha do público alvo foi em função da sua formação acadêmica, já que todos são formados em computação e, na sua grande maioria, já atuam profissionalmente nessa área, além de estarem em busca de uma especialização, o que os torna mais propensos a utilizar novas tecnologias.

Outra decisão nesta fase foi que no início do experimento seriam apresentados os objetivos e o roteiro das atividades a serem executadas. Além disso, em ambas as turmas, seriam disponibilizados o

documento gerado no EPF Composer com a descrição e os passos de todos os processos necessários para a instrumentação dos aplicativos, bem como a concepção e utilização da estrutura de avaliação. O documento serve como apoio técnico para os participantes realizarem as atividades propostas.

Por fim, ficou deliberado que os participantes deveriam responder um questionário após a execução das atividades, envolvendo três partes: a primeira, com os dados pessoais, a segunda, referente à experiência do participante na área de pesquisa e, a terceira, com as perguntas relacionadas aos processos executados durante a realização das atividades.

As respostas que contemplam a terceira parte estão formatadas de acordo com a escala de Likert (1932) com cinco opções de respostas, cujo objetivo é mensurar a satisfação dos participantes ou o nível de dificuldade enfrentado para executar os processos. Além disso, são propostas perguntas relacionadas com a descrição dos processos, possibilitando avaliar se as descrições contemplam as expectativas dos usuários e se servem, efetivamente, de apoio para a execução das atividades propostas.

6.4 FASE 4 – IDENTIFICAR E ADMINISTRAR AS QUESTÕES PRÁTICAS

Nesta fase, as preocupações se voltaram aos recursos necessários para a condução do experimento. Entre eles, a preparação do local onde o experimento seria conduzido, com a instalação nos computadores da IDE Eclipse, com os *plugins* que disponibilizam a plataforma de desenvolvimento de aplicativos Android e o AspectJ. Além disso, foi necessário viabilizar a conexão dos equipamentos com a Internet, possibilitando o acesso ao ambiente de avaliação dos dados estruturado na nuvem.

Outro aspecto muito importante foi desenvolver um aplicativo para a plataforma Android com a finalidade de instrumentá-lo com as métricas de rastreabilidade, tendo assim, o objetivo de possibilitar a comparação da abordagem tradicional (codificar os Aspectos) com a nova abordagem proposta, utilizando a infraestrutura UXEProject.

O aplicativo criado foi chamado de Places Me e tem como principal funcionalidade exibir a localização atual de um determinado usuário, em formato textual, com base nas coordenadas de latitude e longitude capturadas através do GPS do *smartphone*. Os endereços são exibidos na janela principal da aplicação, podem ser salvos em uma lista e transmitidos via e-mail. Na Figura 62 é possível observar a janela principal da aplicação.

O aplicativo Places ME atende ao seu propósito, apesar de ser extremamente simples e possuir uma interface que disponibiliza poucos recursos de interação. O importante nesse caso é prover um código

fonte conhecido para ser apresentado e discutido com os participantes da primeira fase do experimento. O objetivo é fazer com que os participantes conheçam detalhadamente as classes, métodos e atributos da estrutura do aplicativo para que possam codificar os Aspectos de rastreabilidade pela abordagem tradicional, possibilitando, assim, a comparação das abordagens.

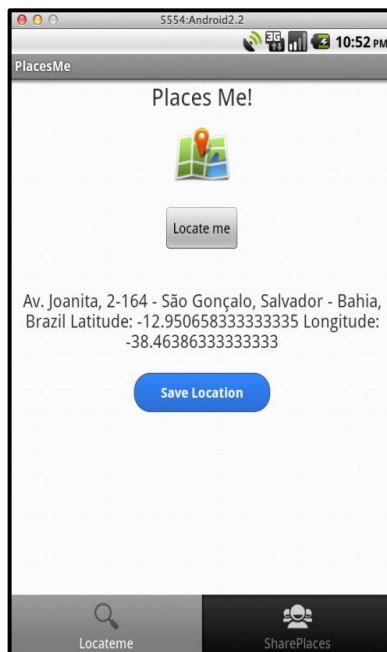


Figura 62 – Janela principal da Aplicação Places ME

6.5 FASE 5 – DECIDIR COMO LIDAR COM AS QUESTÕES ÉTICAS

Com base nas questões éticas discutidas na Seção 5.5, foram tomadas as precauções necessárias para que o experimento fosse conduzido preservando o anonimato dos participantes. Embora os questionários submetidos aos participantes tivessem informações pessoais, tais como, nome, idade, curso de graduação, ano de formatura, atividade profissional e empresa onde trabalha, foi especificado no Termo de Consentimento Livre e Esclarecido (Anexo XI) que tais informações não seriam divulgadas individualmente e serviriam apenas como uma forma de avaliar o perfil dos participantes de forma genérica. O mesmo cuidado foi tomado com relação às informações referentes à experiência dos participantes.

Questões relacionadas a menores de idade não precisaram ser contempladas, já que os 22 participantes que executaram a primeira etapa do experimento tinham idades entre 23 a 49 anos e as 14

pessoas que participaram da segunda fase tinham idades entre 21 e 38 anos. Além disso, nenhum participante pertence a grupos vulneráveis, visto que todos possuem nível superior completo e exercem suas profissões.

6.6 FASE 6 – AVALIAÇÃO DOS DADOS

O primeiro ponto a ser analisado é o perfil dos participantes. Na primeira fase do experimento, foi possível observar que 77,27% dos usuários tinham mais do que três anos de experiência na área de desenvolvimento de software, 86,36% se consideram em um nível médio ou alto na escala de produtividade de software, 81,81% já tinham desenvolvido software na linguagem Java, 100% conheciam o paradigma de Programação Orientado a Objetos, 45,45% não tinham nenhuma experiência com Programação Orientada a Aspectos (POA) e 50% tinham pouca experiência com POA. Na segunda etapa, foi observado que 85,71% dos participantes tinham experiência com o SGBD MySQL e apenas 14,28% tinham conhecimento do funcionamento da ferramenta OLAP Pentaho.

A condução da primeira etapa do experimento ocorreu em quatro fases distintas. Na primeira fase, foram abordados todos os conceitos de POA e realizados alguns exercícios para que os participantes aprendessem e fixassem os conceitos abordados. Na segunda fase, foram apresentadas aos participantes as funcionalidades do Aplicativo Places Me e discutida de forma detalhada a sua construção. Posteriormente, foi solicitado que os participantes criassem cinco métricas de rastreabilidade e as inserissem no Aplicativo Places Me. Na última fase, os participantes seguiram a abordagem proposta com a infraestrutura UXEProject descrita na documentação criada com a ferramenta EPF Composer, criaram as métricas e instrumentaram o aplicativo.

A finalidade da execução da quarta fase foi validar os processos e comparar a abordagem tradicional (codificar os Aspectos) e a abordagem proposta na infraestrutura UXEProject (gerar os Aspectos automaticamente).

A condução da segunda fase do experimento ocorreu em três fases distintas. Na primeira fase, foram abordados os principais conceitos relacionados a BD, DW e Ferramentas OLAP. Na segunda fase, foi apresentada uma visão geral da infraestrutura UXEProject, contemplando a sua estrutura e as finalidades de cada módulo. Na última fase, os participantes seguiram a abordagem proposta na infraestrutura UXEProject e descrita na documentação criada com a ferramenta EPF Composer, bem como executaram as tarefas necessárias para preparar todo o ambiente de recepção, armazenamento e avaliação dos dados de usabilidade.

A análise dos resultados da primeira fase foi dividida em três etapas: os processos para o mapeamento das tarefas, os processos para instrumentar os aplicativos com as métricas de rastreabilidade e uma avaliação geral sobre a abordagem apresentada para a utilização da infraestrutura UXEProject. Os resultados serão apresentados nas próximas cinco subseções. A apresentação dos resultados da avaliação da segunda fase do experimento foi definida em uma única etapa e será apresentada na Subseção 6.7.6.

6.6.1 Análise da Ferramenta MAG

Inicialmente, foi avaliada a satisfação dos participantes com relação à descrição das tarefas para a utilização da Ferramenta MAG (detalhada na seção 4.6.1.1 desta tese). A Figura 63 mostra que 77,28% dos participantes do experimento ficaram “Satisfeitos” ou “Muito Satisfeitos” com a descrição dos processos para utilizar a Ferramenta MAG.

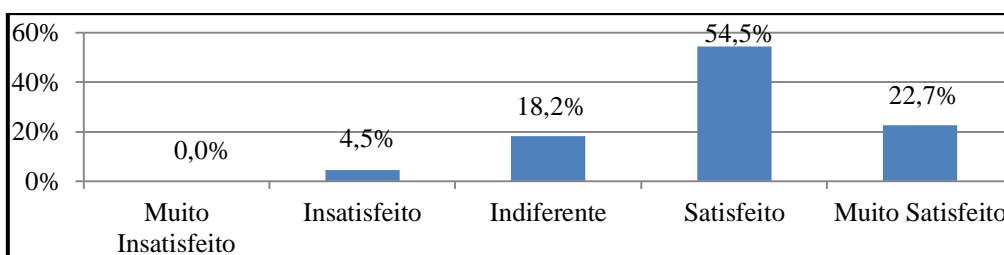


Figura 63 – Avaliação da descrição dos processos para utilizar a Ferramenta MAG

Posteriormente, foi realizada uma análise com a intenção de identificar o grau de dificuldade referente à utilização da ferramenta. Conforme apresentado na Figura 64, a maioria dos participantes acha a sua utilização “Fácil” ou “Muito Fácil”.

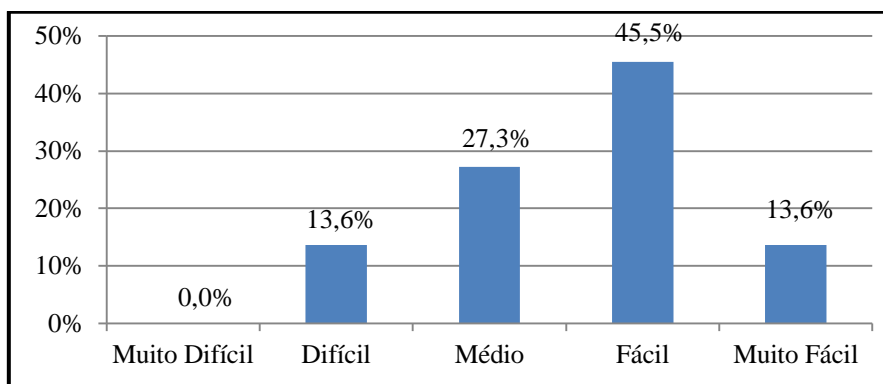


Figura 64 – Avaliação da Ferramenta MAG

Com o objetivo de avaliar a abordagem proposta com a ferramenta MAG e a abordagem tradicional, onde os desenvolvedores de software precisam identificar manualmente as classes de interação que o aplicativo disponibiliza aos usuários, foi especificado na Figura 65, o nível de dificuldade das duas abordagens, de acordo com a visão dos participantes do experimento. Analisando os resultados apresentados, é possível concluir que a abordagem baseada na Ferramenta MAG tende a ser mais fácil que a tradicional.

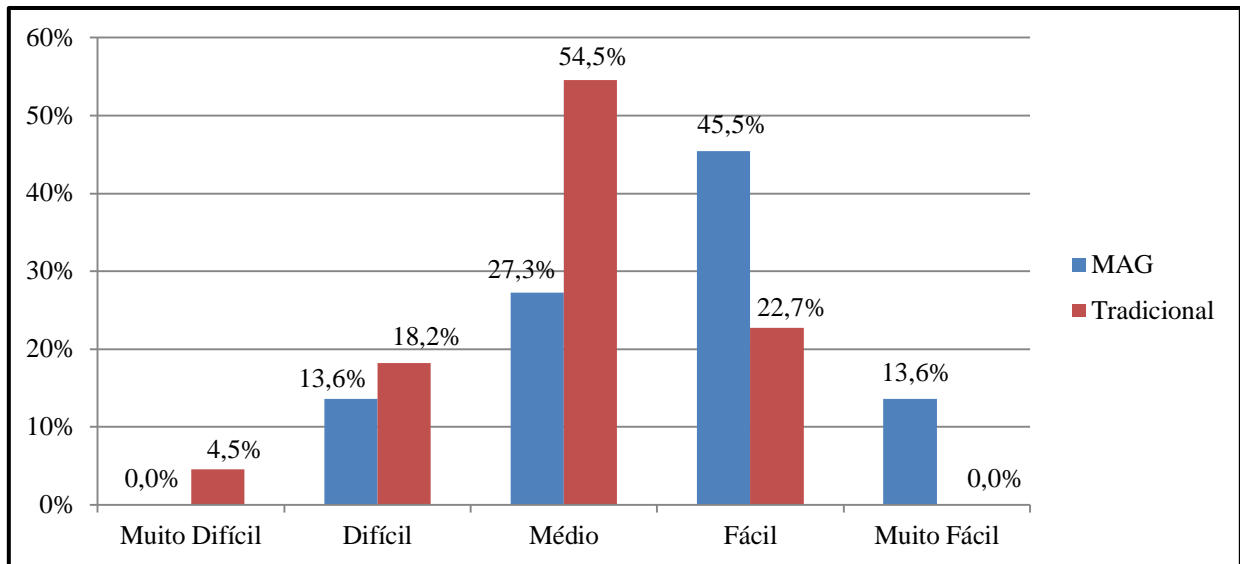


Figura 65 – Abordagem proposta com a Ferramenta MAG x Abordagem Tradicional

6.6.2 Análise da Ferramenta ATD

As próximas análises envolvem a descrição das ações necessárias para efetivar o mapeamento das tarefas, o nível de dificuldade para a utilização da Ferramenta ATD e uma comparação com a abordagem tradicional no que se refere à identificação dos métodos para serem utilizados como *join points*.

Na Figura 66, pode ser observado que mais de 70% dos participantes ficaram “Satisfeitos” ou “Muito Satisfeitos” com a descrição dos processos para a utilização da Ferramenta ATD, logo os processos estão descritos de forma coerente para a maioria dos participantes do experimento.

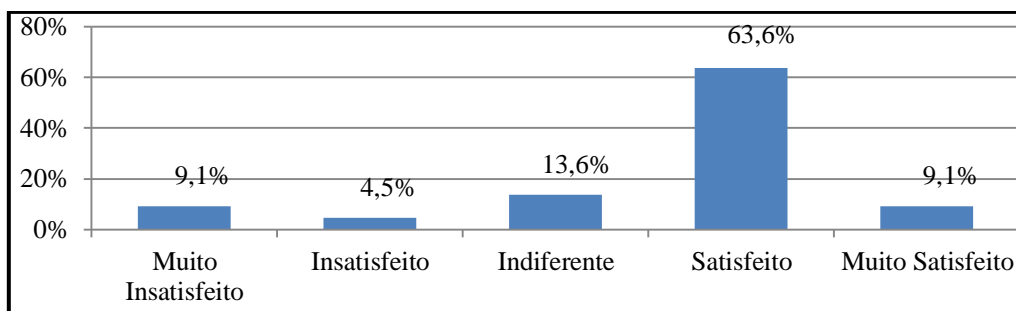


Figura 66 – Avaliação da descrição dos processos para a utilização da Ferramenta ATD

A próxima análise refere-se ao nível de dificuldade ao utilizar a ferramenta ATD para realizar o mapeamento das tarefas. Como pode ser observado na Figura 67, a metade dos participantes do experimento acha “Fácil” ou “Muito Fácil” a utilização da Ferramenta ATD, entretanto os outros 50% classificam a complexidade do processo como “Médio” ou “Difícil”. A principal reclamação dos usuários refere-se à assertividade que o responsável pelo mapeamento das tarefas precisa ter para realizar as etapas corretamente, já que um erro durante o processo invalida todos os passos realizados anteriormente e causa a necessidade de reiniciar o procedimento. Além disso, é necessário conhecer profundamente as funcionalidades do aplicativo e os caminhos para a execução das tarefas disponibilizadas.

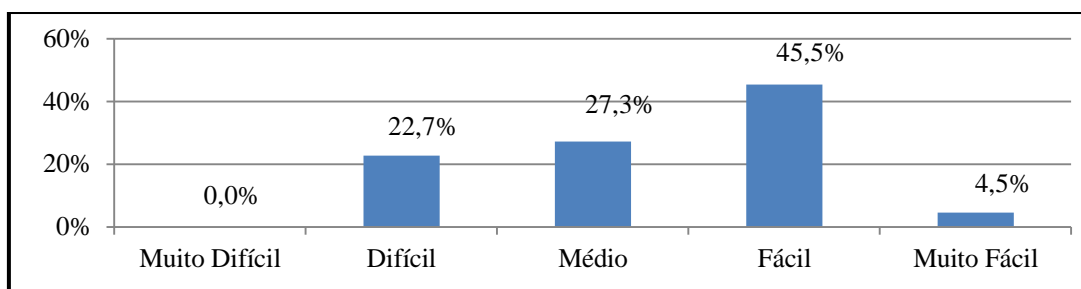


Figura 67 – Avaliação da dificuldade para utilizar a Ferramenta ATD

Para finalizar esta subseção, é apresentada a avaliação dos participantes sob a ótica da abordagem tradicional e da nova abordagem oriunda da infraestrutura UXEProject para identificação dos métodos que serão utilizados para criar a estrutura dos Aspectos (*pointcuts e advices*).

Como pode ser observado na Figura 68, a maioria dos participantes do experimento julga ser uma tarefa de fácil ou média complexidade identificar os métodos que serão quantificados utilizando a abordagem tradicional. Por outro lado, quando a abordagem proposta pela Ferramenta ATD é avaliada, percebe-se um grande percentual de usuários (22,7%) que avaliou a abordagem proposta como sendo difícil. Entretanto, vale ressaltar que a complexidade utilizando a abordagem tradicional aumenta significativamente se o desenvolvedor que irá criar as métricas de usabilidade não conhecer profundamente como o software a ser instrumentado foi construído.

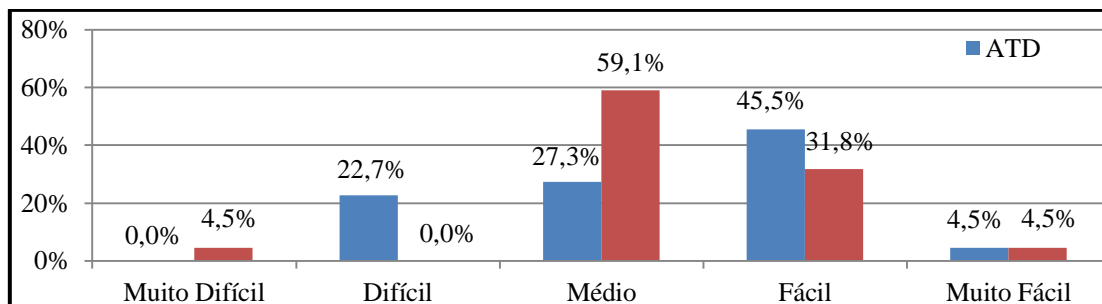


Figura 68 – Abordagem proposta com a Ferramenta ATD X Abordagem Tradicional

6.6.3 Análise geral do processo de mapeamento das tarefas

Nesta subsecção, será analisada a percepção dos participantes do experimento referente ao conjunto de etapas necessárias para executar o mapeamento das tarefas. Para obter os resultados foram analisados: (i) o nível de satisfação com a descrição dos passos contidos no documento que dá suporte ao conjunto de etapas e, (ii) a comparação da abordagem tradicional com a abordagem proposta na infraestrutura UXEProject.

Como pode ser observado na Figura 69, a maioria dos participantes se enquadra na classe dos satisfeitos ou muito satisfeitos, logo a descrição dos passos para realizar o mapeamento das tarefas está coerente com seu objetivo.

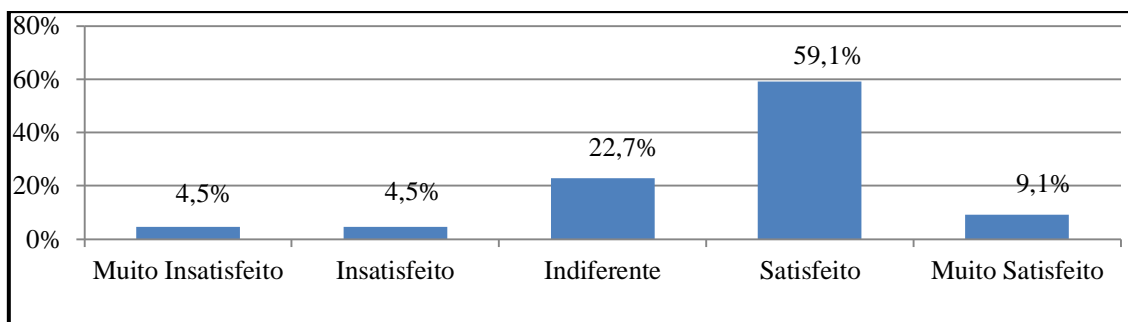


Figura 69 – Avaliação da descrição dos processos do mapeamento de tarefas

Na Figura 70, pode ser observado que existe certo equilíbrio na opinião dos participantes do experimento com relação ao nível de facilidade de utilização das duas abordagens. Desta forma, é importante ressaltar que o objetivo de ambas é identificar quais serão os *join points* na aplicação a serem empregados nos Aspectos que conterão as métricas de rastreabilidade, ou seja, quais serão os métodos a serem utilizados

para criar as métricas. Sendo assim, ambas as abordagens apresentam pontos positivos e negativos, como pode ser observado na Figura 70.

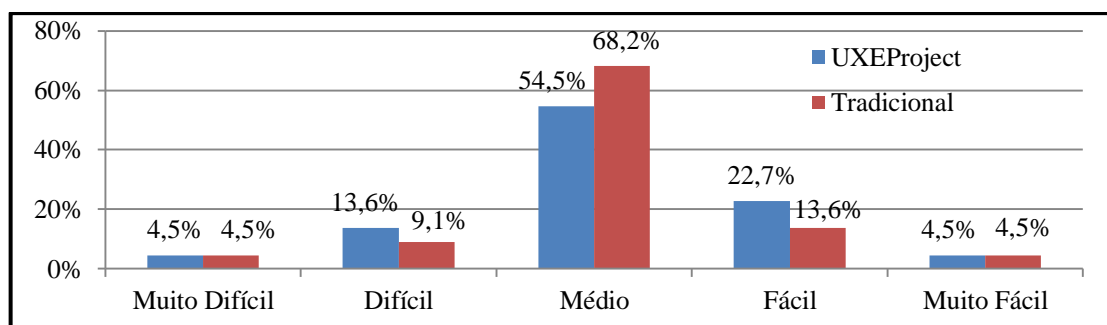


Figura 70 – Comparação das abordagens para identificar os *join points*

Tabela 35 – Pontos positivos e negativos das abordagens para identificar os *join points*

Metodologia	Pontos Positivos	Pontos Negativos
UXEProject	A identificação das classes e métodos que serão utilizados nos <i>pointcuts</i> e <i>advice</i> s é automática, por intermédio das ferramentas que dão suporte ao processo. Desta forma, o engenheiro de software ou especialista da área de IHC não precisa conhecer como o aplicativo foi codificado e nem identificar métodos no código fonte da aplicação.	São necessárias inúmeras ações para realizar o processo. Além disso, qualquer erro cometido requer o reinício do mesmo.
Tradicional	Para o engenheiro de software que desenvolveu o aplicativo é fácil identificar as classes e métodos que serão quantificadas pelos Aspectos.	Caso os Aspectos com as métricas de rastreabilidade sejam desenvolvidos por uma equipe que não criou o aplicativo, a identificação das classes e métodos que serão utilizados nas métricas torna-se uma tarefa complexa.

6.6.4 Criação e inserção das métricas de rastreabilidade

Para a obtenção dos resultados deste processo, foram submetidas três perguntas aos participantes do experimento com o objetivo de analisar a descrição do processo relacionado à utilização da Ferramenta UXE Metrics Generator e comparar a metodologia empregada com a metodologia tradicional.

Na Figura 71, pode ser observado que a maioria dos participantes está satisfeito com relação à descrição do processo para a criação e à instrumentação dos aplicativos com as métricas de rastreabilidade.

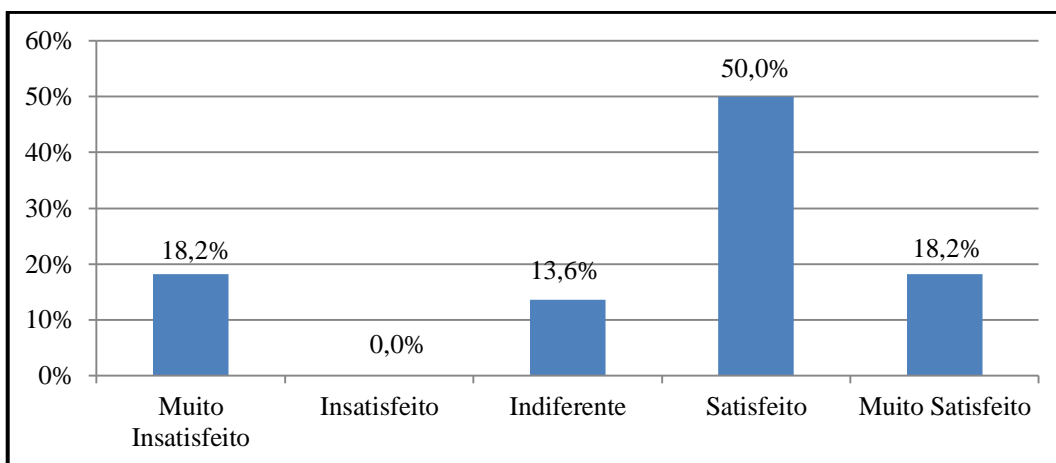


Figura 71 – Análise da descrição dos processos referentes a Ferramenta UXE Metrics Generator

Para utilizar a Ferramenta UXE Metrics Generator na geração automática dos Aspectos que permitem a incorporação das métricas de rastreabilidade, pode ser observado, na Figura 72, que a maioria dos participantes acha fácil ou médio o nível de dificuldade.

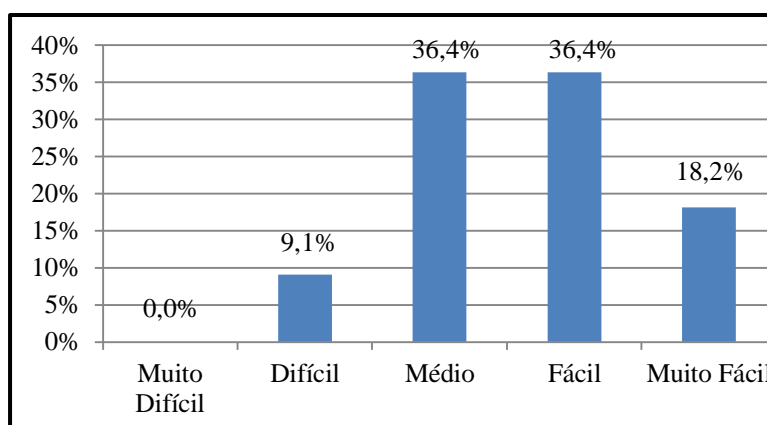


Figura 72 – Avaliação do nível de dificuldade para utilizar a Ferramenta UXE Metrics Generator

Para finalizar a avaliação deste processo, foi comparada a metodologia empregada na Ferramenta UXE Metrics Generator com a metodologia tradicional que, neste caso, significa desenvolver a programação dos Aspectos que contém as métricas de rastreabilidade.

Na Figura 73, podemos observar que os participantes do experimento acham mais difícil o desenvolvimento das métricas com a metodologia tradicional do que desenvolver as métricas com a metodologia proposta com a ferramenta UXE Metrics Generator.

A principal conclusão que pode ser obtida dos resultados apresentados nessa subseção é que a maioria dos participantes então satisfeitos com a descrição dos passos para a criação e inserção das métricas de rastreabilidade nas aplicações. Outra conclusão é que a maioria dos participantes considera ser mais fácil

utilizar a ferramenta UXE Metrics Generator para criar as métricas do que programá-las, já que a infraestrutura UXEProject proporciona uma ferramenta que realiza este processo automaticamente.

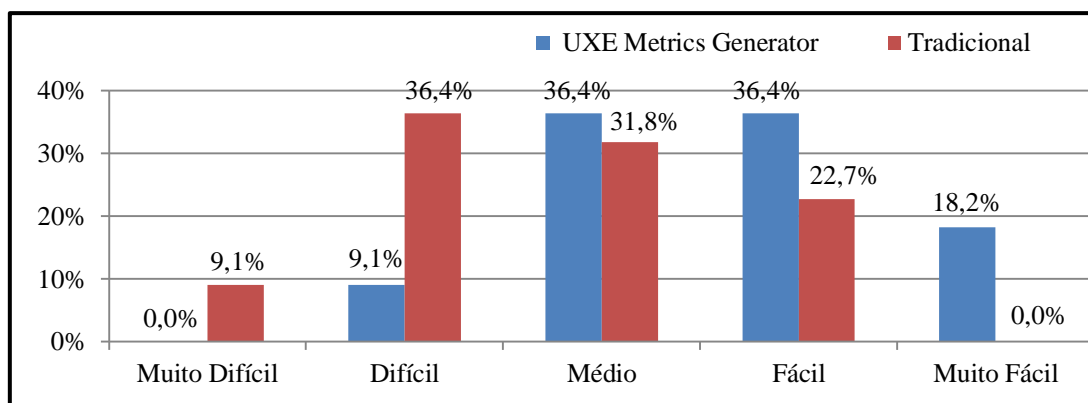


Figura 73 – Comparação das metodologias para criar as métricas de rastreabilidade

6.6.5 Análise geral do processo de instrumentação dos aplicativos

Esta subsecção tem como objetivo traçar uma visão geral da percepção dos participantes sobre a descrição de todos os processos envolvidos para a instrumentação de aplicativos com a utilização da infraestrutura UXEProject. Além disso, pretende-se identificar o nível de dificuldade enfrentado pelos participantes do experimento para a execução dos processos e compará-lo com a forma tradicional de realizar a mesma instrumentação.

Inicialmente, avaliou-se a descrição dos processos. Na Figura 74, pode ser observado que a maioria dos participantes está satisfeita e nenhum deles está insatisfeito ou muito insatisfeito, o que fornece indícios positivos de que a descrição dos processos estaria em conformidade com as necessidades dos usuários.

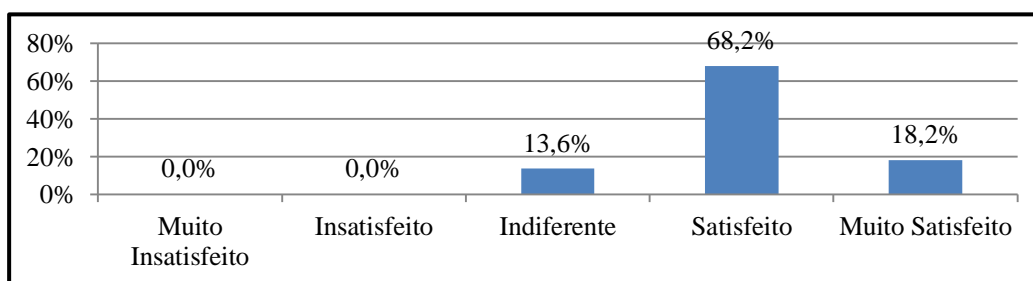


Figura 74 – Avaliação geral da descrição dos processos para a instrumentação dos aplicativos

A próxima avaliação refere-se à dificuldade de execução dos processos envolvidos com a instrumentação de aplicativos utilizando a infraestrutura UXEProject. É possível observar, na Figura 75, que a opinião dos participantes está dividida com o mesmo percentual (27,3%) entre os níveis difícil, médio e fácil. Entretanto, é importante salientar que 18,19% acha muito fácil e que 0% acha muito difícil realizar os processos.

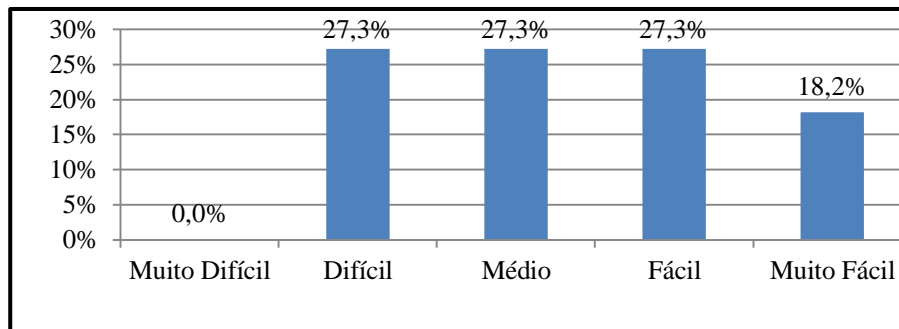


Figura 75 – Avaliação geral do nível de dificuldade para instrumentar aplicativos

Para finalizar esta análise, foi avaliado o nível de dificuldade enfrentado pelos participantes para instrumentar os aplicativos utilizando as duas metodologias. Na tradicional, os processos são necessariamente executados por engenheiros de software devido à necessidade do desenvolvimento do código das métricas. Por outro lado, utilizando a infraestrutura UXEProject, os códigos são gerados automaticamente, mas existe uma série de passos necessários para a realização do mapeamento das tarefas e geração das métricas de rastreabilidade.

Na Figura 76, os resultados apresentados, comprovam que os participantes do experimento consideram, em geral, a metodologia tradicional mais difícil de ser aplicada para a instrumentação dos aplicativos do que a utilização da infraestrutura UXEProject. É importante ressaltar que este resultado é uma das comprovações da relevância deste trabalho para as comunidades de Engenharia de Software e IHC.

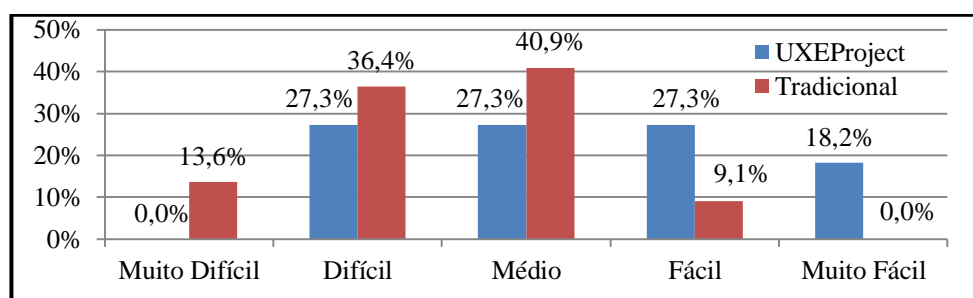


Figura 76 – Avaliação geral das duas metodologias para instrumentar aplicativos

6.6.6 Resultados da análise referente ao Ambiente de Avaliação

A análise dos resultados desta fase do experimento investiga os processos necessários para criar o ambiente de avaliação dos dados de usabilidade capturados durante a interação dos usuários com um determinado aplicativo.

A primeira investigação refere-se à descrição dos processos para a instalação das ferramentas Data Load e ETL Maker. A primeira apoia os processos de alimentação do BD e, a segunda, extrai, transforma e carrega os dados no DW.

A Figura 77 mostra que a maioria dos participantes do experimento ficou satisfeita com a descrição dos processos para a instalação das ferramentas Data Load e ETL Maker.

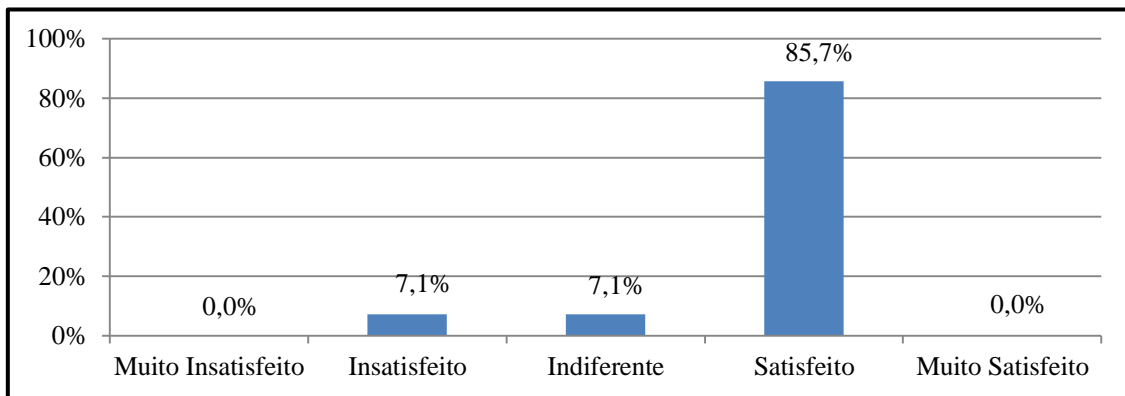


Figura 77 – Nível de satisfação com a descrição dos processos das Ferramentas Data Load e ETL

A Figura 78 apresenta os resultados referentes ao grau de dificuldade para a execução da instalação das duas ferramentas. De acordo com os resultados obtidos, pode ser constatado que a maioria dos participantes considera média ou fácil a execução da instalação.

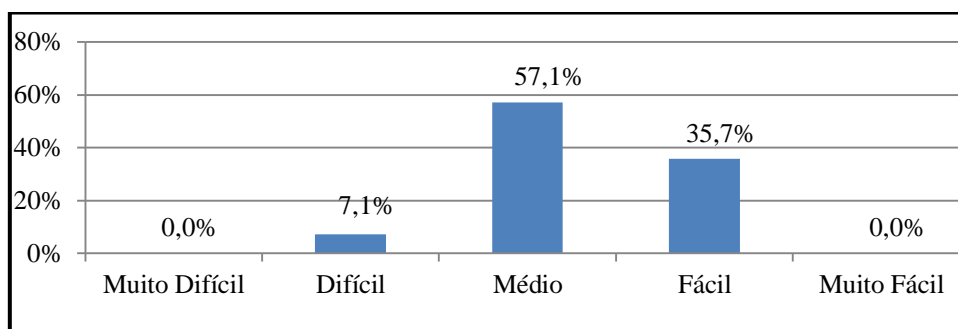


Figura 78 – Nível de dificuldade para instalar as Ferramentas Data Load e ETL Maker

A segunda investigação avalia a descrição e execução do processo referente à preparação da ferramenta OLAP (Pentaho) para dar suporte às investigações de usabilidade. Na Figura 79, nota-se que os participantes ficaram, em geral, satisfeitos com a descrição do processo.

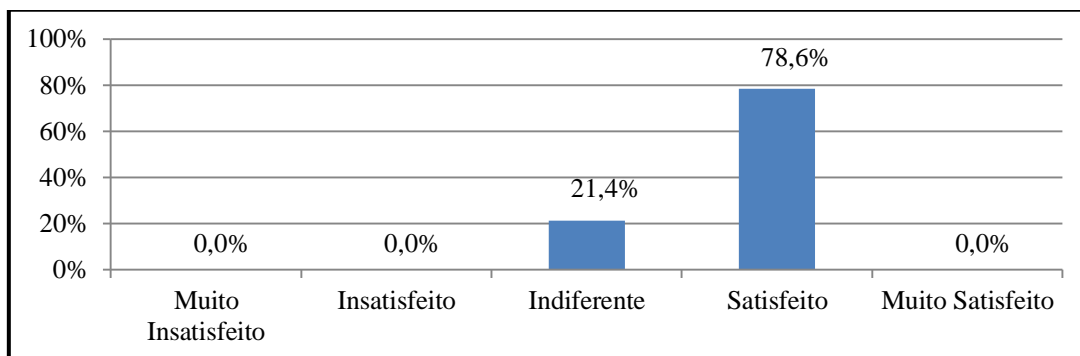


Figura 79 – Nível de satisfação com a descrição dos processos relacionados à Ferramenta OLAP

Com relação ao nível de dificuldade para executar o processo, a maioria dos participantes classificou como médio, como pode ser observado na Figura 80. Um fato relevante que merece destaque refere-se à perda de desempenho da rede e processamento do ambiente na nuvem, potencializando as dificuldades para a finalização do processo. Este problema foi causado devido ao número de usuários executando o mesmo processo simultaneamente, o que não será uma prática normal em um cenário natural de utilização.

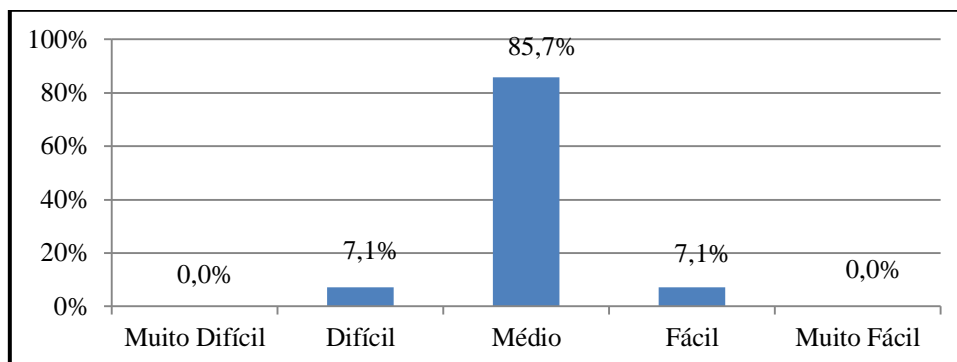


Figura 80 – Nível de dificuldade para a preparação da Ferramenta OLAP

A última investigação realizada neste experimento refere-se à dificuldade para a realização da análise dos dados. Como pode ser observado na Figura 81, metade dos participantes considera fácil a realização da análise dos dados, enquanto que os outros 50% consideram uma atividade de média complexidade.

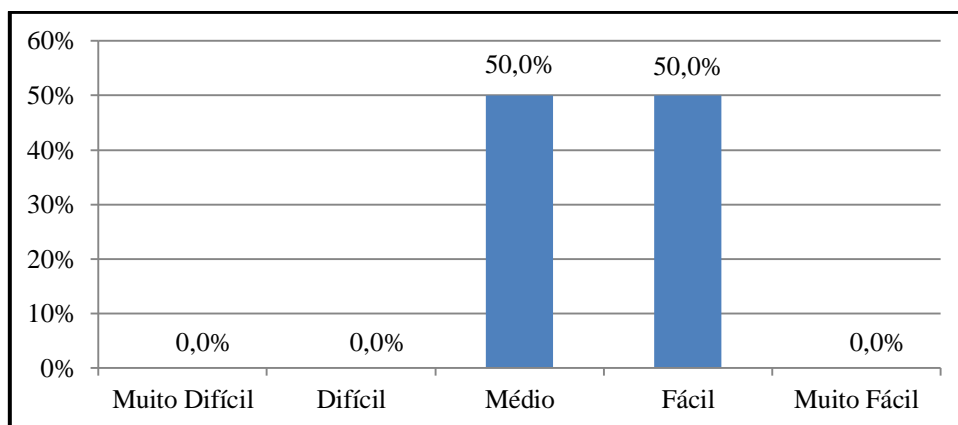


Figura 81 – Nível de dificuldade para a realização das análises dos dados

Após a análise geral dos dados obtidos nesta fase do experimento, pode ser concluído que a maioria dos participantes ficou satisfeita com a descrição dos processos relacionados com a preparação do ambiente de avaliação. Por outro lado, pode se perceber que existe uma tendência dos participantes em classificar a execução dos processos como uma atividade de média complexidade. Uma observação relevante neste caso é que as dificuldades enfrentadas pelos participantes foram decorrentes da pouca familiaridade com o sistema operacional Ubuntu, já que não foram instaladas as APIs da interface gráfica, o que exigiu dos participantes conhecimento dos comandos oriundos do UNIX para a realização do processo. Desta forma, é recomendado que sejam instalados os componentes gráficos para facilitar este processo.

6.7 RESUMO E COMENTÁRIOS FINAIS DO CAPÍTULO

Neste capítulo, foram apresentadas as descrições dos principais processos relacionados com a infraestrutura UXEProject e os resultados de um experimento realizado com os potenciais usuários da infraestrutura. Os objetivos do experimento foram validar o documento criado para a descrição dos processos, comparar a abordagem tradicional com a nova abordagem proposta, identificar os pontos fortes e fracos dos processos necessários para utilizar a infraestrutura.

Com o resultado obtido no experimento, foi possível perceber que utilizar uma descrição formal de processos, em um ambiente centralizado e de fácil acesso, contribui para facilitar a utilização da infraestrutura UXEProject. O grau de satisfação dos participantes foi alto referente à descrição dos processos em todas as avaliações, logo as especificações realizadas estão de acordo com as necessidades dos usuários que participaram do experimento. Os resultados também demonstram que a abordagem proposta com a

infraestrutura UXEProject para a instrumentação dos aplicativos é melhor e mais fácil quando comparado a metodologia tradicional, principalmente em função das ferramentas criadas para a geração automática dos Aspectos necessários para realizar o mapeamento das tarefas e a criação das métricas de rastreabilidade.

As análises referentes ao ambiente de armazenamento e à avaliação dos dados de usabilidade possibilitaram identificar que as ferramentas criadas para executar a carga no BD, a extração, transformação e carga no DW, são fáceis de serem instaladas e possibilitam com que estas tarefas sejam realizadas de forma transparente para os avaliadores. Além disso, a ferramenta Pentaho mostrou ser adequada a seu propósito, facilitando a descoberta de conhecimentos na base de dados.

De modo geral, além de ser constatada a eficiência e eficácia da abordagem proposta pela infraestrutura UXEProject, foi importante a descrição dos processos e a validação por meio da realização de um experimento, fato que é pouco comum na área de IHC, onde se percebe a falta de especificações formais para as abordagens de avaliação propostas na área.

Este capítulo apresenta as conclusões desta tese de doutorado, abordando as suas contribuições e limitações. Além disso, é realizado o posicionamento do trabalho em relação a outras pesquisas realizadas na mesma área, bem como são descritos os trabalhos em andamento e as próximas atividades previstas com a infraestrutura UXEProject.

7 CONCLUSÕES

Os *smartphones* requerem que as interfaces de seus aplicativos estejam adaptadas a particularidades como tela reduzida, mobilidade e redução de consumo. Além disso, a utilização de aplicativos embarcados nesses dispositivos está sujeito à interferência do meio externo, ações em paralelo e contextualizações da aplicação ao cenário de uso. Sendo assim, testes de usabilidade em laboratório são complexos de serem realizados devido à dificuldade de simular esses fatores no ambiente de avaliação (Jensen e Larsen, 2007). Outro ponto a ser destacado é que normalmente não existe uma correlação de diferentes tipos de dados nas análises, tais como, dados estatísticos, contextuais e subjetivos (Ardito et al., 2008).

Segundo Queiroz e Ferreira (2009), várias limitações e falhas podem ser percebidas na maioria dos testes de usabilidade conduzidos em dispositivos móveis. A maioria destas falhas é relacionada a fatores orçamentários, tecnológicos ou mesmo de recursos humanos, enquanto outras estão relacionadas à falta de adequação das abordagens empregadas.

Ao longo do Capítulo 2, foram apresentados inúmeros trabalhos de avaliação da usabilidade de aplicativos para *smartphones* sendo constatado que, em grande parte das pesquisas, são utilizados questionários para a coleta de dados, o que pode dificultar a correlação de diferentes tipos de dados para a descoberta de problemas de usabilidade (Courage e Baxter, 2005). Além disso, na maioria dos casos, não são investigados fatores contextuais, o que é defendido por inúmeros pesquisadores como sendo um fator primordial para os avanços na área de avaliações da usabilidade para os novos dispositivos (Zhang e Adipat, 2005), (Jensen e Larsen, 2007), (Coursaris e Kim, 2011), (Hansen, 2012).

Como alternativa para contornar esses problemas, este trabalho apresentou um modelo e a implementação de uma infraestrutura para a avaliação da usabilidade de aplicações disponibilizadas em *smartphones*, com potencialidade para realizar investigações no ambiente natural do usuário (*in vivo*), bem como facilitar as avaliações em laboratório (*in vitro*). Além disso, possibilita a captura e análise de diferentes tipos de dados, tornando mais precisas as conclusões referentes aos problemas de usabilidade.

O modelo está alicerçado no mapeamento simplificado das tarefas a serem investigadas, na captura de dados referentes à interação do Usuário Final com a aplicação, além de coletar dados referentes ao contexto, levar em consideração o perfil dos usuários e as suas impressões sobre a usabilidade dos aplicativos. Estes diferentes tipos de dados, possibilitam realizar análises alusivas à usabilidade correlacionando diversos aspectos e proporcionando explicações bastante detalhadas sobre alguns problemas das interfaces.

Para validar o modelo, foi construída uma infraestrutura com o objetivo de identificar se as especificações previstas inicialmente poderiam ser concretizadas. Em seguida, a infraestrutura foi utilizada para instrumentar três aplicativos (produzidos e distribuídos por terceiros) com a finalidade de verificar se era aderente ao propósito para o qual foi concebida. O resultado do processo de instrumentação e avaliação dos aplicativos foi positivo, permitindo concluir que as métricas geradas poderiam instrumentar qualquer aplicativo Android que contemple as normas de modularização previsto pela Engenharia de Software.

Posteriormente, foi realizado um experimento em campo envolvendo 21 Usuários Finais utilizando os três aplicativos instrumentados, durante um ano. Esse experimento comprovou que os dados desejados podiam ser efetivamente capturados de forma automática e transparente, sendo, em seguida, armazenados para análise da usabilidade das aplicações. Com a execução das análises, foi possível identificar alguns problemas de usabilidade, relacionando diferentes tipos de dados e visões, comprovando as potencialidades da infraestrutura UXEProject.

Para validar a abordagem proposta do ponto de vista dos potenciais usuários, foi realizado um experimento com desenvolvedores de software e analistas de BD. O propósito do experimento foi verificar se as descrições dos processos estavam em conformidade com as expectativas dos potenciais usuários da infraestrutura, além de avaliar as dificuldades relacionadas com a execução dos processos e de comparar a proposta com a abordagem tradicional para instrumentação do código e avaliação de dados de usabilidade. A partir da análise dos dados obtidos, foi possível constatar que a infraestrutura UXEProject é simples de ser utilizada, a documentação dos seus processos atendem às expectativas dos usuários, trazendo vantagens com relação à abordagem tradicional.

Para retratar a relevância de pesquisas nesta área, podem ser citados alguns trabalhos que demonstram os avanços na área de IHC utilizando as técnicas empregadas na concepção do modelo e sua infraestrutura apresentados nesta tese.

Em 2006, Moldovan e Tarta foram os pioneiros a utilizar POA para criar métricas de usabilidade. Apesar de ser uma ideia inovadora para a época, foi estruturado para avaliar plataformas *desktop*, além disso, as métricas eram inseridas nas aplicações via programação e tinham que ser reestruturadas para cada novo aplicativo a ser instrumentado.

A ferramenta UMARA, desenvolvida por Bateman et al. (2009), foi outra contribuição importante para a área de avaliação de usabilidade utilizando a técnica de *Logging*. As métricas de captura dos dados foram construídas com POA e são geradas automaticamente. Os seus módulos abrangem a instrumentação das aplicações, coleta de dados e análise dos resultados. Apesar de ser desenvolvida para avaliar interfaces construídas com o pacote *swing* da linguagem Java, que não se aplica para dispositivos móveis, deixou um legado indispensável no que se refere à forma de instrumentação de aplicativos utilizando eventos e componentes da interface.

Um dos trabalhos mais relacionados ao da proposta desta tese foi escrito por Balagtas-Fernandez e Hussmann (2009). Os autores anteviram a evolução da área de avaliação de dispositivos móveis e descreveram uma proposta que contempla praticamente todos os aspectos abordados na infraestrutura UXEProject. Os anseios e desejos apresentados pelos autores foram considerados na definição do modelo e na implementação da infraestrutura concebidos nesta tese. Entretanto, novas funcionalidades foram projetadas, principalmente no que se refere ao conjunto de dados que podem ser correlacionados nas avaliações.

O último trabalho, desenvolvido em paralelo ao apresentado nesta pesquisa, foi proposto por Lettner e Holzmann (2012). Ele contempla exatamente a mesma estruturação utilizada na infraestrutura UXEProject para identificar as interações dos usuários e coletar os dados dos sensores. A principal diferença é que o *framework* proposto pelos autores ainda não contempla o mapeamento das tarefas e a utilização da técnica ESM para a coleta de dados subjetivos, sendo estes, segundo os autores, os próximos passos a serem realizados na sua pesquisa. Desta forma, acredita-se que este trabalho pode lhes auxiliar a alcançar seus objetivos, uma vez que foram implementados na infraestrutura UXEProject.

Após a execução desta pesquisa, foi possível constatar que o modelo e a sua infraestrutura propostos neste trabalho alcançam os objetivos pretendidos, atendem as premissas previstas para este estudo e contribuem de forma significativa para a área de avaliação de usabilidade, trazendo inovações e soluções para os problemas atuais, considerando as expectativas de vários pesquisadores na área de IHC.

O experimento realizado em campo proporcionou uma série de informações a respeito da usabilidade dos aplicativos investigados, permitiu identificar problemas de usabilidade considerando as dimensões estabelecidas para este estudo como, por exemplo, a eficiência, eficácia, satisfação, aprendizagem e facilidade de uso.

Por fim, o experimento realizado com os potenciais usuários do UXEProject permitiu avaliar a operabilidade da abordagem, apresentando dados que comprovam a possibilidade da adoção da proposta em larga escala, o que contribuirá para melhorar as avaliações de usabilidade para *smartphones*.

Nas próximas seções, serão descritas as contribuições, as limitações e discutidos os futuros trabalhos relacionados a esta pesquisa.

7.1 CONTRIBUIÇÕES

O principal legado deste trabalho refere-se ao modelo para avaliação da usabilidade de *smartphones* e a sua implementação, através da infraestrutura UXEProject, que contempla os requisitos e as especificações previstos no modelo. Entretanto, para alcançar estes resultados, foi necessário projetar e desenvolver várias ferramentas que dão suporte a infraestrutura criada, mesclar técnicas para a coleta de dados e propor um ambiente integrado e transparente para a Equipe de Avaliação executar a análise dos dados. Sendo assim, outras contribuições de menor impacto para a área de IHC podem ser destacadas:

- Projetar uma forma de mapear tarefas em aplicativos para *smartphones*.
- Desenvolver a ferramenta MAG para habilitar os aplicativos a serem mapeados e a ferramenta ATD para efetivar o mapeamento de forma automática, livre de codificações.
- Estruturar uma biblioteca genérica de métricas, capaz de contemplar diferentes tipos de dados, com a possibilidade de ser estendida de acordo com as necessidades de avaliação.
- Criar a ferramenta UXE Metrics Generator para a geração automática das métricas.
- Instrumentar os aplicativos sem interferir no código fonte original das aplicações, possibilitando a inserção e retirada das métricas de acordo com a necessidade dos avaliadores.
- Projetar um ambiente com a capacidade de receber as informações rastreadas, armazená-las e possibilitar a sua análise com o cruzamento de diferentes tipos de dados.

Com a utilização da infraestrutura para a instrumentação de três aplicativos e a realização de um experimento realizado em campo, com duração de um ano, foi possível constatar que a solução apresentada para a avaliação da usabilidade de aplicativos para *smartphones* atinge diversas expectativas da comunidade de IHC, entre elas podem ser destacadas:

- Realizar testes de usabilidade contextualizados ao ambiente de uso dos dispositivos, contemplando uma das carências na área de IHC.

- Considerar diferentes tipos de dados nas avaliações, o que possibilita correlacionar dados estatísticos, contextuais, subjetivos e o perfil dos usuários. Esta característica é outra carência observada nos estudos apresentados no Capítulo 2 desta tese.

- Possibilitar a instrumentação de um vasto grupo de aplicações de forma automática, ou seja, sem a necessidade de escrever códigos.

- Contemplar avaliações independentes do número de usuários.

- Coletar dados de interação de forma transparente para os Usuários Finais, sem ser intrusivo com relação as suas ações naturais.

- Executar análises continuadas e multidimensionais, propiciando interpretações e explicações referentes à usabilidade dos aplicativos pautada em diferentes ângulos de análise.

No decorrer do desenvolvimento desta pesquisa, uma das estratégias utilizadas para verificar a aceitação do projeto por parte da comunidade de IHC, foi submeter artigos referentes aos resultados observados durante a sua realização. Essa ação possibilitou a obtenção de inúmeras contribuições que permitiram progressos na pesquisa. A seguir estão listados, em ordem cronológica, os artigos científicos publicados ao longo do desenvolvimento desta tese:

- Kronbauer, A. H.; Santos, C. A. S.; Chavez, C. **Um Modelo para Coleta Automática e Avaliação das Ações de Interatividade do Telespectador de TVDI**. IV Latin American Workshop on Aspect-Oriented Software Development, Salvador - BA, Brasil, outubro de 2010.

- Kronbauer, A. H.; Santos, C. A. S. **Um Modelo para Avaliação de Usabilidade de Aplicações Imperativas em GingaJ**. IV Congresso TI e TELECOM, InfoBrasil 2011, Fortaleza - CE, Brasil, março de 2011.

- Kronbauer, A. H.; Santos, C. A. S. **Um modelo de avaliação da usabilidade baseado na captura automática de dados de Interação do usuário em ambientes reais**. XVII WebMedia, Florianópolis - SC, Brasil, outubro de 2011.

- Kronbauer, A. H.; Santos, C. A. S. **Um modelo de Avaliação da usabilidade baseado na interação do usuário em ambientes reais**. Workshop de Teses e Dissertações - XVII WebMedia, Florianópolis - SC, Brasil, outubro de 2011.

- Kronbauer, A. H.; Santos, C. A. S. **Um modelo de avaliação da usabilidade baseado na captura automática de dados de interação do usuário em ambientes reais**. X IHC & V CLIHC, Porto de Galinhas – PE, Brasil, outubro de 2011.

- Kronbauer, A. H.; Santos, C. A. S.; Vieira, V. **Smartphone Applications Usability Evaluation: A Hybrid Model and Its Implementation**. 4th International Conference on Human-Centered Software Engineering, Toulouse, France, October, 2012.

- Kronbauer, A. H.; Santos, C. A. S.; Vieira, V. **Um Estudo Experimental de Avaliação da Experiência dos Usuários de Aplicativos Móveis a partir da Captura Automática dos Dados Contextuais e de Interação**. XI IHC – Simpósio Brasileiro sobre Fatores Humanos em Sistemas Computacionais, Cuiabá – MT, Brasil, novembro de 2012.

7.2 LIMITAÇÕES

Em todas as pesquisas existem limitações, sendo este um dos principais combustíveis para os pesquisadores continuarem suas buscas por novos resultados e aprimorarem o que foi realizado. Ao longo deste estudo, fica claro alguns aspectos que devem ser melhorados e, em alguns casos, ainda realizados. Os pontos listados a seguir explicitam algumas das principais limitações deste trabalho:

- O mapeamento das tarefas é limitado a uma forma de executar uma determinada tarefa, sendo necessários vários mapeamentos para a mesma tarefa quando há mais de uma forma de execução. Este problema limita a investigação da avaliação de novos caminhos executados pelos usuários que não foram identificados previamente pelos desenvolvedores ou avaliadores do aplicativo.

- A infraestrutura foi projetada apenas para a plataforma Android, o que limita a sua utilização para avaliar aproximadamente metade dos aplicativos existentes para *smartphones*.

- O número de participantes do experimento realizado em campo foi modesto diante da possibilidade prevista no modelo que viabiliza avaliações com um grande número de usuários.

- Não foi formalizado um modelo estatístico para validar os dados coletados nos dois experimentos, o que inviabiliza a generalização dos resultados observados.

- Nas análises do experimento realizado em campo, não foram investigados alguns dos elementos da dimensão de usabilidade estabelecidos para este trabalho, tais como, operabilidade, acessibilidade, utilidade e flexibilidade. Entretanto, esses elementos foram previstos na infraestrutura e posteriormente podem ser contemplados em novos experimentos.

- Avaliações com desenvolvedores inseridos no mercado específico de aplicativos para *smartphones* não foram contemplados neste estudo, sendo este um ponto falho, já que restringiu as avaliações da abordagem apenas ao cenário acadêmico.

- Apesar da constatação da viabilidade da abordagem ser utilizada em larga escala, alguns processos para a utilização da infraestrutura UXEProject devem ser automatizados para facilitar a instrumentação dos aplicativos.

7.3 PERSPECTIVAS FUTURAS

Os *smartphones* serão os dispositivos computacionais utilizados para promover a inclusão digital de milhares de pessoas em todo o mundo, sendo necessário que os aplicativos desenvolvidos para estes aparelhos apresentem boa usabilidade, tornem-se cada vez mais úteis e atrativos. Este aspecto fundamenta a continuidade da linha de pesquisa abordada neste trabalho e fortalece a necessidade de aprimorar a abordagem proposta nesta tese.

Inicialmente, como perspectiva para a continuidade deste trabalho, faz-se necessário desenvolver um modelo estatístico para que as informações observadas no experimento realizado em campo possam ser generalizadas.

Apesar da possibilidade de generalização dos resultados obtidos com esta pesquisa, é preciso considerar que a avaliação foi conduzida apenas para a plataforma Android e com apenas três aplicativos. Com o objetivo de consolidar a validação da abordagem apresentada, mostra-se pertinente a aplicação do processo de avaliação da usabilidade ora descrito a outras plataformas e aplicativos, sendo mantidas as hipóteses desta pesquisa, de forma a comprovar (ou refutar) os resultados obtidos neste trabalho. Sendo assim, sugere-se que a infraestrutura UXEProject deva contemplar outras plataformas como, por exemplo, para Iphone OS.

Será importante a realização de novos testes com aplicativos inseridos no mercado, com a adesão de centenas de usuários, possibilitando avaliar a escalabilidade da infraestrutura e a adequação da abordagem para experimentos com um grande número de participantes, de modo a reforçar a validação dos resultados obtidos nesta pesquisa. Além disso, os elementos da dimensão de usabilidade estabelecidos para este trabalho, que não foram contemplados neste estudo, tais como, operabilidade, acessibilidade, utilidade e flexibilidade, devem ser contemplados em um novo experimento.

Os processos para a utilização da infraestrutura devem ser remodelados, permitindo mais eficiência na execução dos passos para a instrumentação dos aplicativos. Após a execução dessas melhorias, novas análises devem ser realizadas com a participação de desenvolvedores e especialistas em IHC, o que possibilita avaliar a abordagem sob a ótica de pessoas inseridas no mercado de aplicativos para *smartphones*.

Apesar da infraestrutura ter sido testada apenas com *smartphones*, o modelo foi proposto para contemplar diferentes tipos de dispositivos móveis, desta forma, faz-se necessário a avaliação da infraestrutura para outros dispositivos como, por exemplo, *tablets*.

A propriedade de reusabilidade do modelo prevê a adequação ao contexto em que o aplicativo será avaliado. Isto inclui características do cenário e dos recursos tecnológicos do dispositivo no qual a

aplicação está sendo executada. Por exemplo, a detecção das formas de interação do Usuário Final (voz, gestos, toques) pode servir de subsídio para determinar a utilização de determinadas métricas ou a supressão das mesmas. Para atender ao requisito de reusabilidade, foi prevista a recuperação de informações sobre os recursos do dispositivo e a configuração do cenário de interesse de investigação no Módulo de Incorporação das Métricas ao Código Fonte. Assim, é possível que diferentes conjuntos de métricas sejam disponibilizadas para uma mesma aplicação como forma de adequar essas métricas ao contexto de uso das aplicações. Por exemplo, as métricas de usabilidade de uma aplicação de acesso a vídeos devem variar se o usuário acessa um vídeo local ou pela Internet, se o usuário possui ou não acesso 3G, entre outras características. A utilização desse recurso deve ser alvo para a evolução da infraestrutura.

Na literatura, encontram-se várias opiniões acerca da relação entre o número de usuários e o número de falhas detectadas. Assim, propõe-se como trabalho futuro a análise desta relação, de forma a identificar um número mínimo de usuários necessário para atingir um percentual pré-especificado de identificação de falhas.

Por fim, para a divulgação dos resultados encontrados durante a realização desta pesquisa, serão submetidos novos artigos a periódicos, com a expectativa que outras contribuições possam ser propostas para a evolução continuada da abordagem proposta neste trabalho, contemplando novas expectativas e anseios dos especialistas na área de IHC.

REFERÊNCIAS

- ARDITO, C.; BUONO, P.; COSTABILE, M. F.; ANGELI, A.; LANZILOTTI, R. **Combining Quantitative and Qualitative Data for Measuring User Experience of an Educational Game**. Proceedings of the International Workshop on Meaningful Measures: Valid Useful User Experience Measurement (VUUM). Reykjavik, Iceland, pp. 27–31, 2008.
- AVOURIS, N.; FIOTAKIS, G.; RAPTIS, D. **On Measuring Usability of Mobile Applications**. In Proceedings International Workshop on Meaningful Measures: Valid Useful User Experience Measurement, Reykjavik, Iceland, pp. 38–44, 2008.
- BAK, J. O.; NGUYEN, K.; RISGAARD, P.; STAGE, J. **Obstacles to usability evaluation in practice: a survey of software development organizations**. In Proceedings of the 5th Nordic conference on Human Computer Interaction: building bridges. New York, NY, USA: ACM, pp. 23–32, 2008.
- BALAGATAS-FERNANDEZ, F.; HUSSMANN, H. **Evaluation of User-Interfaces for Mobile Application Development Environments**. In Human-Computer Interaction. New Trends, Vol. 5610, Springer, pp. 204–213, 2009.
- BARBOSA, S. D. J.; SILVA, B. S. **Interação Humano-Computador**, 1 ed. Elsevier, Rio de Janeiro, 2010.
- BARRETT, L. F.; BARRETT, D. J. **An Introduction to Computerized Experience Sampling in Psychology**. Social Science Computer Review, v. 19, no. 2, pp 175–185, 2001.
- BARNARD, L.; YI, J. S.; JACKO, J. A.; SEARS, A. **Capturing the Effects of Context on Human Performance in Mobile Computing Systems**. Journal Personal and Ubiquitous Computing Archive. v. 11, no. 2, Springer-Verlag, pp. 81–96, 2007.
- BARRA, H. **Android: momentum, mobile and more at Google I/O**. The Official Google Blog, 2011. Disponível em: <http://googleblog.blogspot.com/2011/05/android-momentum-mobile-and-more-at.html>. Acesso em 11 de dezembro de 2012.
- BATEMAN, S.; GUTWIN, C.; OSGOOD, N.; McCALLA, G. **Interactive usability instrumentation**. In. EICS '09: 1st ACM SIGCHI Symposium on Engineering Interactive Computing Systems, ACM, pp. 45–54, 2009.
- BENYON, D.; TURNER, P.; TURNER, S. **Designing Interactive Systems – People, Activities, Contexts, Technologies**. 1 ed. Person Addison-Wesley, New York, 2005.
- BERNARD, L.; YI, J. S.; SEARS A. **Capturing the effects of context on human performance in mobile computing systems**. Personal and Ubiquitous Computing, pp. 81–96, 2007.
- BETIOL, A. H.; CYBIS, W. A. **Usability Testing of Mobile Devices: A Comparison of Three Approaches**. In: INTERACT 2005, LNCS, pp. 470–481, 2005.
- BØDKER, M.; GIMPEL, G.; HEDMAN, J. **Smart Phones and their Substitutes: Task-Medium Fit and Business Models**. In 8th International Conference on Mobile Business, IEEE Computer Society, pp. 24–29, 2009.

- BRADLEY, P. R.; DUREJ, M. T.; GESSNER, A. E.; GOULD, A. J.; KHAN, I. A.; MARTIN, B. N.; MUTTY, M. L.; MYERS, E. C.; PATEK, S. D. **Smartphone Application for Transmission of ECG Images in Pre-Hospital STEMI Treatment**. Proceedings of the IEEE Systems and Information Engineering Design Symposium, pp. 118–123, 2012.
- BRAJNIK, G. **Web Accessibility Testing: When The Method is the Culprit**. In Computers Helping People with Special Needs, LNCS, v. 4061, Springer, pp. 156–163, 2006.
- BRANDSTÄTTER, H. **Emotional Responses to Other Persons in Everyday Life Situations**. Journal of Personality and Social Psychology, v. 45, no. 4, pp. 871–883, 1983.
- BURIGAT, S.; CHITTARO, L.; GABRIELLI, S. **Navigation techniques for small-screen devices: An evaluation on maps and web pages**. International Journal of Human-Computer Studies, v. 66, no. 2, pp. 78–97, 2008.
- BUTUZA, A.; HAUER, I.; MUNTEAN, C.; POPA, A. **Increasing the Business Performance using Business Intelligence**. Analele Universitatii “Eftimie Murgu” Resita, anual XVIII, no. 3, pp. 67–72, ISSN 1453–7397, 2011.
- CAIRNS, P.; COX, A. L. **Research Methods for Human-Computer Interaction**. 1 ed. Cambridge University Press, New York, 2008.
- CHACON, S. **Pro Git**. 1ed. Apress, New York, 2009.
- CHARLAND, A.; LEROUX, B. **Mobile application development**. Magazine Communications of the ACM, v. 54, no. 5, pp. 49–53, 2011.
- CHEN, T. **Input to the Mobile Web is Situationally Impaired**. In Proceedings of the 10th international ACM SIGACCESS conference on Computers and accessibility, ACM, pp. 303–304, 2008.
- CHEN, Q.; FAN, Y. **Smartphone-Based Travel Experience Sampling and Behavior Intervention**. Transportation Research Board (TRB) 91st Annual Meeting, pp. 1–17, 2012.
- CHENG, J.; WONG, S. H. Y.; YANG, H.; LU, S. **SmartSiren: virus detection and alert for smartphones**. In Proceedings of the 5th International Conference on Mobile Systems, Applications and Services. MobiSys '07. New York, NY, USA: ACM, pp. 258–271, 2007.
- CHIN, A.; SALOMAA, J. P. **A user study of mobile web services and applications from the 2008 Beijing Olympics**. In 20th ACM conference on Hypertext and Hypermedia, Torino, Italy: ACM, pp. 343–344, 2009.
- CONSOLVO, S.; WALKER, M. **Using the Experience Sampling Method to Evaluate Ubicomp Applications**. IEEE Pervasive Computing, pp. 24–31, 2003.
- CONTI, G.; PUDNEY, S. **Survey Design and the Analysis of Satisfaction**. The Review of Economics and Statistics, v. 93, no. 3, pp. 1087–1093, 2011.
- COURAGE, C; BAXTER, K. **Understanding your user: a practical guide to user requirements, methods, tools, and techniques**. 1 ed. Morgan Kaufmann Publishers, São Francisco, CA, 2005.

COURSARIS, C. K.; KIM, D. J. **A Meta-Analytical Review of Empirical Mobile Usability Studies**. *Journal Usability Studies*, v. 6, no. 3, pp. 117–171, 2011.

CSIKSZENTMIHALYI, M.; LARSON, R. **Validity and Reliability of the Experience Sampling Method**. *The experience of psychopathology: Investigating mental disorders in their natural settings*, pp. 43–57, 1992.

DIMARZIO, J. F. **Android a Programmers Guide**. 1ed. McGraw–Hill Osborne Media, 2008.

DINELEY, D. **Best of Open Source Software Awards 2008**. InfoWord's 2008. Disponível em: <http://www.infoworld.com/d/open-source/best-open-source-software-awards-2008-785>. Acesso em: 11 de dezembro de 2012.

DO, T. M. T.; BLOM, J.; GATICA-PEREZ, D. **Smartphone usage in the wild: a large scale analysis of applications and context**. In *Proceedings of the 13th International Conference on Multimodal Interfaces. ICMI '11*. New York, NY, USA: ACM. pp. 353–360, 2011.

EBNER, M.; STICKEL, C.; SCERBAKOV, N.; HOLZINGER, A. **A Study on the Compatibility of Ubiquitous Learning (u-Learning) Systems at University Level**. *Universal Access in Human-Computer Interaction. Applications and Services. Springer Lecture Notes in Computer Science*, v. 5616, pp. 34–43, 2009.

FALESSI, D.; CANTONE, G.; KAZMAN, R. **Decision-making techniques for software architecture design: A comparative survey**. *Journal ACM Computing Survey*, v. 43, no. 4, pp. 1–28, 2011.

FETAJI, M.; FETAJI, B.; **Universities go Mobile – Case Study Experiment in Using Mobile Devices**. In the proceedings of 30th International Conference on Information Technology Interfaces, IEEE conference, pp. 123–128, 2008.

FETAJI, B.; EBIBI, M.; FETAJI, M. **Assessing Effectiveness in Mobile Learning by Devising MLUAT (Mobile Learning Usability Attribute Testing) Methodology**. *International Journal of Computers and Communications*, v. 5, no. 3, pp. 178–187, 2011.

FITCHETT, S.; COCKBURN, A. **Evaluating reading and analysis tasks on mobile devices: A case study of tilt and flick scrolling**. In *The 21st Annual Conference of the Australian Computer-Human Interaction*, pp. 255–232, 2009.

FROEHLICH, J.; CHEN, M.; CONSOLVO, S.; HARRISON, B.; LANDAY, J. **MyExperience: A System for In Situ Tracing and Capturing of User Feedback on Mobile Phones**. In: *Proceedings 5th International Conference Mobile Systems, Applications and Services*, pp. 470–516, 2007.

FROEHLICH, J.; DILLAHUNT, T.; KLASNJA, P.; MANKOFF, J.; CONSOLVO, S.; HARRISON B.; LANDAY, J. A. **Ubigreen: Investigating a Mobile Tool for Tracking and Supporting Green Transportation Habits**. *Proceedings of the 27th International Conference on Human Factors in Computing Systems*, ACM, pp. 1043–1052, 2009.

GARTNER. **Gartner Says Worldwide Smartphone Sales Soared in Fourth Quarter of 2011 With 47 Percent Growth**. Disponível em: <http://www.gartner.com/it/page.jsp?id= 1924314>, acesso em 7 de Setembro de 2012.

GOETTEN JUNIOR, V.; WINCK, G. V. **AspectJ – Programação Orientada a Aspectos com Java**. 1 ed. Novatec, São Paulo, 2006.

GONZÁLEZ, M. P.; LORÉS, J.; GRANOLLERS, T. **Enhancing usability testing through data mining techniques**. Information and Software Technology, v. 50, no. 6, pp. 547–568, 2008.

GRØNLI, T.; HANSEN, J.; GHINEA, G. **Android, Java ME and Windows Mobile interplay: The Case of a Context-Aware Meeting Room**. 24th International Conference on Advanced Information Networking and Applications Workshops, IEEE Computer Society, pp. 920–925, 2010.

GRØNLI, T.; HANSEN, J.; GHINEA, G. **Integrated Context-aware and Cloud-based Adaptive Application Home Screens for Android Phones**. 14th International Conference on Human-Computer Interaction. Springer, pp. 427–435, 2011.

GRØNLI, T. **Cloud Computing and Context-Awareness – A Study of the Adapted User Experience**. Thesis submitted for the degree of Doctor, School of Information Systems, Computing and Mathematics – Brunel University, 2012.

GROSS, T.; STEENKISTE, P.; SUBHLOK, J. **Adaptive Distributed Applications on Heterogeneous Networks**. In Proceedings of the 8th Heterogeneous Computing Workshop. Washington, DC, USA: IEEE Computer Society, pp. 209–218, 1999.

GUEDES, G. T. A. **UML 2 – Um abordagem Prática**. 2 ed. Novatec, São Paulo, 2011.

HANSEN, J.; GHINEA, G. **Multi-platform Bluetooth remote control**. IEEE International Conference for Internet Technology and Secured Transactions, pp. 1–2, 2009.

HANSEN, J. **An Investigation of Smartphone Applications: Exploring usability aspects related to Wireless Personal Area Networks, Context-awareness, and Remote Information Access**. Thesis submitted for the degree of Doctor, School of Information Systems, Computing and Mathematics – Brunel University, 2012.

HAUMER, P. **Eclipse Process Framework Composer – Part 1: Key Concepts**. Solution architect, IBM Rational Software, 2007.

HEGARTY, R.; WUSTEMAN, J. **Evaluating EBSCOhost Mobile**. Library Hi Tech, v. 29, no. 2, pp. 320–333, 2011.

HEO, J.; HAM, D.; PARK, S.; SONG, C.; YOON, W. C. **A Framework for Evaluating the Usability of Mobile Phones Based on Multi-level, Hierarchical Model of Usability Factors**. Journal Interacting with Computers, v. 21, pp. 263–275, 2009.

HEVNER, A.; CHATTERJEE, S. **Design Research in Information Systems: Theory and Practice**. 1 ed. Springer, 2010.

HICKS, J.; RAMANATHAN, N.; KIM, D.; MONIBI, M.; SELSKY, J.; HANSEN, M.; ESTRIN, D. **Andwellness: An Open Mobile System for Activity and Experience Sampling**. Proceeding Wireless Health, pp. 34–43, ACM, 2010.

HIRAMA, K. **Engenharia de Software – Qualidade e Produtividade com Tecnologia**. 1 ed. Campus: São Paulo, 2011.

- HUMPHREY, W. S.; KELLNER, M. **Software Process Modeling: Principles of Entity Process Models**. SEI Software Engineering Institute, Carnegie Mellon University, Pittsburgh, Pennsylvania, (CMU/SEI-89-TR-2), 1989.
- ICKIN, S.; WAC, K.; FIEDLER, M.; JANOWSKI, L.; HONG, J.; DEY, A. K. **Factors Influencing Quality of Experience of Commonly Used Mobile Applications**. IEEE Communications Magazine, pp. 48–56, 2012.
- IEEE. **Software Engineering Collection on CD-ROM**. IEEE, New York, 2003.
- INTILLE, S. S.; RONDONI, J.; KUKLA, C.; ANCONA, I.; BAO, L. **A Context-Aware Experience Sampling Tool**. Proceeding CHI - Extended Abstracts on Human Factors in Computing Systems, CHI EA'03, pp. 972-973, ACM, 2003.
- ISO 9241-11. **Guidance on Usability**. International Organization for Standardization, 1998.
- ISO/IES 9126-1. **Software Engineering – Software Product Quality – Part 1: Quality Model**. International Organization for Standardization, 2001.
- ITU - International Telecommunication Union. **Key Global Telecom Indicators for the World Telecommunication Service Sector**. 2011. Disponível em http://www.itu.int/ITU-D/ict/statistics/at_glance/KeyTelecom.html, acessado em 9 de Setembro de, 2012.
- IVORY, M. Y.; HEARST, M. A. **The state of the art in automating usability evaluation of user interfaces**. ACM Computer Survey, v. 3, no. 4, pp. 470–516, 2001.
- JENSEN, K. L.; LARSEN, L. B. **Evaluating the Usefulness of Mobile Services Based on Captured Usage Data from Longitudinal Field Trials**. In Proceedings of 4th International Conference on Mobile Technology, Application and Systems. Singapore, ACM, pp. 675–682, 2007.
- JONES, M.; MARSDEN, G. **Mobile Interaction Design**. 1 ed, John Wiley & Sons: UK, 2006.
- KAIKKONEN, A.; KEKÄLÄINEN, A.; CANKAR, M.; KALLIO, T.; KANKAINEN, A. **Usability testing of mobile applications: a comparison between laboratory and field testing**. Journal of Usability Studies, v. 1, no. 1, pp. 4–16, 2005.
- KANG, Y. M.; CHO, C.; LEE, S. **Analysis of factors affecting the adoption of smartphones**. In Technology Management Conference (ITMC), IEEE International, pp. 919–925, 2011.
- KANGAS, E.; KINNUNEN, T. **Applying user-centered design to mobile application development**. Magazine Communications of the ACM – Designing for the mobile device, v. 48, no. 7, pp. 55–59, 2005.
- KAWALEK, J.; STARK, A.; RIEBECK, M. **A New Approach to Analyze Human-Mobile Computer Interaction**. Journal of Usability Studies, v. 3, no. 2, pp. 90–98, 2008.
- KENTERIS, M.; GAVALAS, D.; ECONOMOU, D. **An innovative mobile electronic tourist guide application**. Personal Ubiquitous Computing Journal, v.13, no. 2, pp.103–118, 2009.
- KHAN, V. J.; MARKOPOULOS, P.; EGGEN, B.; IJSSELSTEIJN, W.; RUYTER, B. **Reconexp: A Way to Reduce the Data Loss of the Experiencing Sampling Method**. Proceedings of the 10th International

Conference on Human Computer Interaction with Mobile Devices and Services, MobileHCI'08, pp. 471–476, ACM, 2008.

KIM, S.; LEE, I.; LEE, K.; JUNG, S.; PARK, J.; KIM, Y. B.; KIM, S. R.; KIM, J. **Mobile web 2.0 with multi-display buttons**. Communications of the ACM, v. 53, n. 1, pp. 136–141, 2010.

KIMBALL, R.; ROSS, M. **The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling**. 2 ed. Wiley: USA, 2002.

KIRWAN, M.; DUNCAN, M. J.; VANDELANOTTE, C.; MUMMERY, W. K. **Using smartphone technology to monitor physical activity in the 10,000 Steps program: a matched case-control trial**. Journal of Medical Internet Research, 2012. Disponível em: <http://www.jmir.org/2012/2/e55/>. Acesso em 22 de janeiro de 2013.

KJELDSKOV, J.; GRAHAM, C. **A review of mobile HCI research methods**. In Mobile HCI, pp. 317–335, 2003.

KJELDSKOV J.; STAGE J. **New Techniques for Usability Evaluation of Mobile Systems**. International Journal of Human-Computer Studies, pp. 599–620, 2004.

KJELDSKOV, J.; SKOV, M. B. **Exploring context-awareness for ubiquitous computing in the healthcare domain**. Personal Ubiquitous Computing, v. 11, no. 7, pp.549–562, 2007.

KORHONEN, H.; KOIVISTO, E. M. I. **Playability heuristics for mobile games**. In Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services. New York, NY, USA: ACM, pp. 9–16, 2006.

KORHONEN, H.; ARRASVUORI, J.; AÄÄNÄNEN-VAINIO-MATTILA, K. **Analysing User Experience of Personal Mobile Products Through Contextual Factors**. In Proceedings of the 9th International Conference on Mobile and Ubiquitous Multimedia, MUM'10, pp. 1–10, 2010.

KRUCHTEN, P. B. **The 4+1 View Model of Architecture**. IEEE Software, v. 12, no. 6, pp. 42–50, 1995.

KRUG, S. **Rocket Surgery Made Easy: The Do-It-Yourself Guide to Finding and Fixing Usability Problems**. 1ed. New Riders Press, 2009.

KUKKONEN, J.; LAGERSPETZ, E.; NURMI, P.; ANDERSSON, M. **Betelgeuse: A Platform for Gathering and Processing Situational Data**. Pervasive Computing, IEEE, v. 8, no. 2, pp. 49–56, 2009.

LAI, J.; VANNO, L.; LINK, M.; PEARSON, J.; MAKOWSKA, H.; BENEZRA, K.; GREEN, M. **Life360: Usability of Mobile Devices for Time Use Surveys**. In. American Association for Public Opinion Research annual conference, Hollywood, FL, pp. 5582–5589, 2009.

LECHETA, R. **Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 1^a ed. Novatec, São Paulo, 2009.

LEITÃO, C.; ROMÃO-DIAS, D. **Pesquisas em IHC: um debate interdisciplinar sobre a ética**. Workshop sobre Interdisciplinaridade em IHC. Congresso Latino-americano de Interação Humano-Computador, pp. 1–2, 2003.

- LETTNER, F.; HOLZMANN, C. **Automated and Unsupervised User Interaction Logging as Basis for Usability Evaluation of Mobile Applications**. Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia. ACM, pp. 118–127, 2012.
- LI, Y. M.; YEH, Y. S. **Increasing trust in mobile commerce through design aesthetics**. Computers in Human Behavior, v. 26, no. 4, pp. 673–684, 2010.
- LIKERT, R. **A Technique for the Measurement of Attitudes**. Archives of Psychology, v. 140, pp. 1–55, 1932.
- LOPES, C. I. V. **D: a language framework for distributed programming**. Ph. D. Thesis, Northeast University, 1997.
- MAGUIRE, M. **Methods to support human-centred design**. International Journal Human-Computer Studies 55, pp. 587–634, 2001.
- MAJCHRZAK, T. A.; JANSEN, T.; KUCHEN, H. **Efficiency evaluation of open source ETL tools**. Proceedings of the ACM Symposium on Applied Computing, pp. 287–294, 2011.
- MALLICK, M. **Mobile and wireless design essentials**. Canada: Wiley Publishing Inc., 2003.
- MALY, I.; MIKOVEC, Z.; VYSTRCIL, J. **Interactive analytical tool for usability analysis of mobile indoor navigation application**. In the 3rd International Conference on Human System Interaction, IEEE Conference Publications, pp. 259–266, 2010.
- MARSHALL, C.; ROSSMAN, G. B. **Designing Qualitative Research**. 5 ed. SAGE Publications Inc. 2011.
- McGOOKIN, D.; BREWSTER, S.; JIANG, W. **Investigating Touchscreen Accessibility for People with Visual Impairments**. Proceedings of the 5th Nordic Conference on Human-Computer Interaction: Building Bridges, ACM, pp. 298–307, 2008.
- MEZARD, M.; PARISI, G.; ZECCHINA, R. **Analytic and Algorithmic Solution of Random Satisfiability Problems**. Science Journals, v. 297, no. 5582, pp. 812–815, 2002.
- MESCHTSCHERJAKOV, A.; WEISS, A.; SCHERNDL, T. **Utilizing Emoticons on Mobile Devices within ESM studies to Measure Emotions in the Field**. MobileHCI, ACM, 2009.
- MIKALAJUNAITE, E. **Android market reaches half a million successful submissions**. Disponível em: <<http://www.research2guidance.com/android-market-reaches-half-a-million-successful-submissions/>>. Acesso em 9 de Setembro de 2012.
- MILLER, R. B. **Human ease of use criteria and their tradeoffs**. IBM Technical Report, 1971.
- MOLDOVAN, G. S.; TARTA, A. **Automatic Usability Evaluation using AOP**. IEEE-TTTC International Conference on Automation, Quality and Testing, Robotics, Cluj-Napoca, pp. 84–89, 2006.
- MÜHLBEIER, A. R. K, MOZZAQUATRO, P, M.; DE OLIVEIRA, I. C.; MONTEIRO, T. B and LOPES, V. **eNIGMA e M-Learning: jogo educativo trabalhando o raciocínio lógico através de dispositivos móveis**. Revista Brasileira de Computação Aplicada, v. 4, no. 2, pp. 92–102, 2012.
- NIELSEN, J. **Usability Engineering**. Academic Press: Cambridge, 1993.

- NIELSEN, J.; MACK, R. L. **Usability Inspection Methods**. John Wiley & Sons: New York, 1994.
- NIELSEN, J. **Usability ROI Declining, But Still Strong**. Jakob's Nielsen Alertbox. 22 jan. 2008, Disponível em: < <http://www.useit.com/alertbox/roi.html>>. Acesso em: 05 de outubro de 2012.
- NIELSEN, C. M.; OVERGAARD, M.; PEDERSEN, M. P.; STAGE, J.; STENILD S. **It's worth the hassle!: the added value of evaluating the usability of mobile systems in the field**. In Proceedings 4th Nordic CHI, Oslo, ACM, pp. 272–280, 2006.
- OMG – Object Management Group. **Software & Systems Process Engineering Meta-Model Specification**, 2008, Disponível em <<http://www.omg.org/spec/SPEM/2.0/PDF/changebarred/>>. Acesso em: 18 de dezembro de 2012.
- PALIT, R.; ARYA, R.; NAIK, K.; SINGH, A. **Selection and execution of user level test cases for energy cost evaluation of smartphones**. In Proceedings of the 6th International Workshop on Automation of Software Test. AST '11. New York, NY, USA: ACM, pp. 84–90, 2011.
- PHAM, A.; PHAM, P. **Scrum em Ação – Gerenciamento e Desenvolvimento ágil de Projetos de Software**. 1ed. Novatec: São Paulo, 2011.
- PEREIRA, A. L.; RAOUFI, M.; FROST, J. C. **Using MySQL and JDBC in New Teaching Methods for Undergraduate Database Systems Courses**. Data Engineering and Management - Lecture Notes in Computer Science, Springer, v. 6411, pp. 245–248, 2012.
- PETRIE, H.; POWER, C. **What do users really care about?: a comparison of usability problems found by users and experts on highly interactive websites**. In Proceedings of the 2012 ACM Annual Conference on Human Factors in Computing Systems. New York, NY, USA: ACM, pp. 2107–2116, 2012.
- PICCOLO, L. S. G.; MENEZES, E. M.; BUCCOLO, B. C. **Developing an Accessible Interaction Model for Touch Screen Mobile Devices: Preliminary Results**. In Proceedings of the 10th Brazilian Symposium on Human Factors in Computing Systems and the 5th Latin American Conference on Human-Computer Interaction, ACM, pp. 222–226, 2011.
- PIVETA, E. **Um modelo de suporte a programação orientada a aspectos**. Dissertação de Mestre em Ciência da Computação. UFSC. 2001.
- PMBOK. **Project Management Body of Knowledg**. 4 ed. Project Management Institute, 2008.
- PRIMAK, F. V. **Decisões com B.I. (Business Intelligence)**. 1ed. Editora Ciência Moderna: Rio de Janeiro, 2008.
- QUEIROZ, J. E. R. **Abordagem híbrida para a avaliação da usabilidade de interfaces com o usuário**. Tese (Doutorado em Engenharia Elétrica) – Pós-graduação em Engenharia Elétrica, Universidade Federal da Paraíba, Campina Grande, 2001.
- QUEIROZ, J. E. R.; FERREIRA, D, S. **A Multidimensional Approach for the Evaluation of Mobile Application User Interfaces**. In: Human-Computer Interaction, San Diego, CA, USA. Lecture Notes in Computer Science. New Trends. Springer Berlin, p.242–251, 2009.

- RAENTO, M.; OULASVIRTA, A.; PETIT, R.; TOIVONEN, H. **Contextphone: A Prototyping Platform for Context-Aware Mobile Applications**. IEEE Pervasive Computing, pp. 51–59, 2005.
- RANI, S. **Grid Computing and Cloud Computing: Description and Comparison**. International Journal of Advanced Research in Computer Science and Electronics Engineering (IJARCSEE), v.1, no. 7, 2012.
- RASCHE, A.; PUHLMANN, M.; POLZE, A. **Heterogeneous adaptive componentbased applications with Adaptive Net**. In Object-Oriented Real-Time Distributed Computing. 8th IEEE International Symposium. pp. 418–425, 2005.
- REED, B. **A Brief History of Smartphones - How the smartphone went from a high-end enterprise device to an everyday consumer staple**. Disponível em: <http://www.pcworld.com/article/199243/a_brief_history_of_smartphones.html>. Acesso em 7 de Setembro de 2012.
- RUBIN, J.; CHISNELL, D. **Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Test**. 2 ed. Wiley Publishing Inc: Indianapolis, 2008.
- SAFFER, D. **Designing for Interaction: Creating Smart Applications and Clever Devices**. 2 ed. New Riders: Berkeley, USA, 2007.
- SAKAMOTO, S. G.; DA SILVA, L. F.; DE MIRANDA, L. C. **Identificando Barreiras de Acessibilidade Web em Dispositivos Móveis: Resultados de um Estudo de Caso Orientado pela Engenharia de Requisitos**. In Brazilian Symposium on Human Factors in Computing Systems, ACM, pp. 23–32, 2012.
- SANTOS, R. C.; GASSENFERTH, W.; MACHADO, M. A. S. **Systems usability evaluation metrics review**. In. Global Business and Technology Association Conference, 2008.
- SAUERBRONN, J. F. R.; CERCHIARO, I. B.; AYROSA, E. A. T. **A debate about alternative methods for academic research in marketing**. Gestão e Sociedade, v. 5, no. 12, pp. 254–269, 2012.
- SCHAUB, F.; DEYHLE, R.; WebER, M. **Password Entry Usability and Shoulder Surfing Susceptibility on Different Smartphone Platforms**. Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia, ACM, 2012.
- SHACKEL, B. **Human Factors for Informatics Usability**. University Press, Cambridge, 1991.
- SHARP, H.; ROGERS, Y.; PREECE, J. **Interaction design: beyond human-computer interaction**. 2 ed. John Wiley & Sons: New York, NY, 2007.
- SHNEIDERMAN, B.; PLAISANT, C. **Designing the User Interface: Strategies for Effective Human-Computer Interaction**. 4 ed. Person Addison-Wesley, 2004.
- SMITH, D. M.; CEARLEY, D. W.; NATIS, Y. V. **Predicts 2011: cloud computing is still at the peak of inflated expectations**. Analysis, v.2, no. 3, 2010.
- SNYDER, C. **Paper Prototyping: the fast and easy way to design and refine user interfaces**. 1 ed. Morgan Kaufmann: San Francisco, 2003.
- SOARES, S.; LAUREANO, E.; BORBA, P. **Implementing Distribution and Persistence Aspects with AspectJ**. ACM SIGPLAN Notices, New York, NY, USA, v. 37, no. 11, pp. 174–190, 2002.

- SODNIK, J.; DICKE, C.; TOMAZIC, S.; BILLINGHURST, M. **A user study of auditory versus visual interfaces for use while driving.** *International Journal of Human-Computer Studies*, v. 66, no. 5, pp. 318–332, 2008.
- SOMMERVILLE, I. **Engenharia de Software.** 6 ed. Person Addison Wesley: São Paulo, 2005.
- SOUZA, C. S. **The Semiotic Engineering of Human-Computer Interaction.** 1 ed. The MIT Press: Cambridge, 2005.
- SPARKES, J.; VALAITIS, R.; and McKIBBON, A. **A Usability Study of Patients Setting Up a Cardiac Event Loop Recorder and BlackBerry Gateway for Remote Monitoring at Home.** *Telemedicine Journal and E-Health the Official Journal of the American Telemedicine Association*, v. 18, no. 6, pp. 484–490, 2012.
- SPYRIDONIS, F. GRØNLI, T.; HANSEN, J.; GHINEA, G. **Evaluating the Usability of a Virtual Reality-Based Android Application in Managing the Pain Experience of Wheelchair Users.** 34th Annual International Conference of the IEEE EMBS - Engineering in Medicine and Biology Society, pp. 2460–2463, 2012.
- SPYRIDONIS, F. GRØNLI, T.; HANSEN, J.; GHINEA, G. **PainDroid: an android-based virtual reality application for pain assessment.** *Journal Multimedia Tools and Applications*, v. 51, Springer, 2013.
- STIENMETZ, J. L.; LEVY, S. E.; BOO, S. **Factors Influencing the Usability of Mobile Destination Management Organization Websites.** *Journal of Travel Research*, pp. 1–12, 2012.
- SUZUKI, S.; BELLOTTI, V.; YEE, N.; JOHN, E. B.; NAKAO, Y.; ASAHI, T.; FUKUZUMI, S. **Variation in Importance of Time-on-Task with Familiarity with Mobile Phone Models.** *Proceedings of the Annual Conference on Human Factors in Computing Systems*, pp. 2551–2554, 2011.
- TARBY, J. C.; EZZEDINE, H.; KOLSKI, C. **Trace-Based Usability Evaluation Using Aspect-Oriented Programming and Agent-Based Software Architecture.** *Human-Centered Software Engineering*, Springer, pp. 257–276, 2009.
- TÂRNAVEANU, D. **Pentaho Business Analytics: a Business Intelligence Open Source Alternative.** *Database Systems Journal*, v. III, no. 3, 2012.
- TOSSELL, C. C.; KORTUM, P.; SHEPARD, C. W.; RAHMATI, A.; ZHONG, L. **Getting Real: A Naturalistic Methodology for Using Smartphones to Collect Mediated Communications.** *Hindawi Publishing Corporation - Advances in Human-Computer Interaction*, pp. 1–10, 2012.
- TREWIN, S. **Physical Usability and the Mobile Web.** *Proceedings of the 2006 International Cross-Disciplinary Workshop on Web Accessibility: Building the mobile web: rediscovering accessibility?* ACM, pp. 109–112, 2006.
- TULLIS, T.; ALBERT, B. **Measuring the User Experience – Collecting, Analyzing and Presenting Usability Metrics.** 1 ed. Morgan Kaufmann: Burlington, 2008.
- VENKATESH, V.; RAMESH, V.; MASSEY, A. P. **Understanding usability in mobile commerce.** *Communications of the ACM*, v. 46, no. 12, pp. 53–56, 2003.

- VIEIRA, V.; TEDESCO, P.; SALGADO, A. C. **Designing context-sensitive systems: An integrated approach**. In: Expert Systems with Applications, v. 38, no. 2, pp. 1119–1138, 2011.
- VYSTRCIL J. **User interface for in-door navigation of visually impaired people via mobile phone**. Master Thesis, CTU in Prague, Czech Republic (in Czech), 2008.
- WAINER, J. **Métodos de pesquisa quantitativa e qualitativa para a ciência da computação**. Jornada de Atualização em Informática. Anais do XXVII Congresso da Sociedade Brasileira de Computação, 2007.
- WATERSON, S.; LANDAY, J. A.; MATTHEWS, T. **In the lab and out in the wild: remote Web usability testing for mobile devices**. In Extended Abstracts of Conference on Human Factors in Computing Systems. Minneapolis, MN. ACM, pp. 796–797, 2002.
- WEISER, M. **The computer for the 21st century**. Scientific American, v. 265, n. 3, pp. 66–75, 1991.
- WIGELIUS, H.; VÄÄTÄJÄ, H. **Dimensions of Context Affecting User Experience in Mobile Work**. In proceedings INTERACT, Springer, pp. 604–617, 2009.
- WOBBROCK, J. O. **The Future of Mobile Device Research in HCI**. In Proceedings Human-Computer Interaction Workshop, ACM, pp. 131-134, 2006.
- XU, Q.; ERMAN, J.; GERBER, A.; MAO, Z. M.; PANG, J.; VENKATARAMAN, S. **Identifying diverse usage behaviors of smartphone apps**. In Proceedings of the ACM SIGCOMM conference on Internet measurement conference. IMC '11. New York, NY, USA: ACM, pp. 329–344, 2011.
- YESILADA, Y.; BRAJNIK, G.; HARPER, S. **Barriers Common to Mobile and Disabled Web Users**. In Interacting with Computers, Springer, v. 23, no. 5, pp. 525–542, 2011.
- ZHANG, D.; ADIPAT, B. **Challenges, Methodologies, and Issues in the Usability Testing of Mobile Applications**. International Journal of Human-Computer Interaction, v. 18, no. 3, pp. 293–308, 2005.
- ZHANG, Q.; CHENG, L.; BOUTABA, R. **Cloud computing: state of the art and research challenges**. Journal of Internet Services and Applications, Springer, pp. 7–18, 2010.

ANEXO I – A Evolução dos Smartphones

Segundo Reed (2010), a primeira tentativa realizada para a construção de um celular com múltiplas funcionalidades foi proposta pela IBM, ao lançar no mercado o IBM Simon em 1993. Ele integrava serviços de voz e dados, além de incorporar funcionalidades de um *Personal Digital Assistant* (PDA) e de uma máquina de fax. Foi construído com a disponibilidade de um teclado sensível ao toque que possibilitava a realização das chamadas telefônicas. Sua memória RAM era de 1 MB e sua velocidade de processamento era de 16 MHz.

O Pilot lançado pela Palm em 1996 é outro precursor importante dos *smartphones*. Apesar de não permitir chamadas telefônicas, incorporava algumas funcionalidades atualmente disponíveis nos *smartphones*, entre elas, podemos destacar a execução de softwares residentes e a tela sensível ao toque. Utilizava um processador Motorola 68328 com velocidade de 16 MHz e tinha, em sua primeira versão, 128 KB de memória RAM.

Em 1998, foi lançado o Nokia 9110 Communicator. Esse dispositivo trouxe algumas inovações atualmente utilizadas nos *smartphones*, tais como: o designer do teclado e a execução de aplicativos com janelas deslizantes. Além disso, utilizava um sistema operacional que incorporava os protocolos SMTP, IMAP4, POP3 e MIME1, que possibilitavam a troca de mensagens via correio eletrônico.

No início de 2002, a empresa canadense Research In Motion (RIM) lançou o *BlackBerry* 5810, um dispositivo que possibilitava enviar e receber e-mail, navegar na internet, enviar e receber SMS, continha um modem, possibilitava sincronização com computadores pessoais e disponibilizava porta Universal Serial Bus (USB). A sua principal limitação é que o usuário tinha que usar um fone de ouvido para falar ao telefone.

O Treo 600 foi um *smartphone* produzido pela Handspring, em 2003. Mais tarde, foi comercializado com a marca da PalmOne, após a fusão das duas empresas. Ele oferecia uma série de recursos integrados, tais como, verificar o calendário durante uma chamada telefônica, discar diretamente da lista de contatos, tirar fotos e enviar e-mails. Disponibilizava um botão de navegação com cinco direções e uma opção que permitia acesso rápido às funções mais usuais do telefone. Em sua especificação original²³, utilizava o processador ARM com 144 MHz e memória RAM de 32 MB.

Lançado em 2007, o *iPhone* é um *smartphone* desenvolvido pela Apple Inc. com funções avançadas para áudio, câmera digital, Internet, mensagens de texto (*Short Message Service* – SMS), mensagem de voz (*visual voicemail*), conexão wifi local e suporte a vídeo chamadas (*FaceTime*). A interação

²³ Disponível em: http://www.gsmarena.com/palm_treo_600-622.php

com o usuário é feita através de uma tela sensível ao toque. A sua mais nova versão é o *iPhone 5*²⁴, lançada em 2012, o qual contempla um processador de quatro núcleos com velocidade de processamento de 1,3 GHz, memória RAM de 1 GB, câmera de 8 megapixels que permite gravações de vídeo FULL HD com 1080p. Seu visual arrojado e a sua facilidade de interação o torna um dos *smartphones* mais vendidos no mundo atualmente.

Outra plataforma que revolucionou os *smartphones* foi o Android²⁵, um sistema operacional baseado no Linux. Ele foi construído para contemplar as especificidades apresentadas pelos dispositivos móveis, sendo atualmente utilizado por grandes fabricantes de *smartphones* e *tablets*, tais como: Sony Ericsson, Motorola, LG, Samsung, HTC, entre outros. O Android foi desenvolvido pela Open Handset alliance, liderada pelo Google. Segundo DiMarzio (2008), a plataforma se adapta a dispositivos de vídeo VGA bem como aos *layouts* mais tradicionais de *smartphones*, com suporte gráfico para gráficos 2D e 3D baseados em OpenGL. Tem a capacidade de tratar mensagens de SMS e MMS. O navegador disponível no sistema é baseado no *framework* de código aberto conhecido como WebKit. As aplicações escritas em Java são compiladas em formato binário (*bytecodes*) Dalvik e executadas utilizando a máquina virtual Dalvik, desenvolvida especialmente para uso em dispositivos móveis, o que permite a distribuição de programas em *bytecode* e possam ser executados em qualquer dispositivo Android, independentemente do processador utilizado. Apesar das aplicações Android serem escritas na linguagem Java, ela não é uma *Java Virtual Machine* (JVM), já que não executa *bytecode* JVM. O sistema suporta os formatos de áudio e vídeo como: MPEG-4, H.264, MP3 e AAC. O Android é capaz de fazer uso de câmeras de vídeo, tela sensível ao toque, *Global Positioning System* (GPS), acelerômetros, entre outros tipos de sensores.

Embora o Android tenha sido lançado oficialmente em 2007, apenas em 2009, com o lançamento do *smartphone* Motorola Droid, é que a plataforma começou efetivamente a ganhar mercado (Reed, 2010). Atualmente, o *smartphone* mais moderno que utiliza a plataforma Android é o Galaxy S4²⁶ da Samsung, com memória RAM interna de 64 GB, processador ARM Cortex-A9 MPCore com 4 núcleos e velocidade de processamento de 1.9 GHz ou com 8 núcleos e processamento de 1.6 GHz. Com 7,9 mm de espessura e pesando apenas 130 g, o Galaxy S4 possui uma tela de 5 polegadas que usa a tecnologia Full HD Super AMOLED e tem 441 pixels por polegada. A tela do *smartphone* se adapta automaticamente à iluminação local. O objetivo é tornar a tela mais confortável para ser usada em qualquer ambiente, horário e iluminação. Esse aparelho foi lançado em 2013 para concorrer com o *iPhone 5* da Apple.

Uma das últimas inovações para *smartphones* foi proposta pela Sprint, que desenvolveu uma rede chamada WiMAX (padrão IEEE 802.16), que é atualmente a rede sem fio mais rápida disponível

²⁴ Disponível em: <http://www.apple.com/iphone/>

²⁵ Disponível em: <http://www.android.com>

²⁶ <http://g1.globo.com/tecnologia/noticia/2013/03/samsung-anuncia-seu-novo-smartphone-galaxy-s4.html>

comercialmente. O EVO 4G, lançado em 2010, é um dos pioneiros a utilizar a rede WiMAX, sendo esta uma das suas principais inovações. Além de sua conectividade, o telefone também possibilita que os usuários possam mantê-lo em pé, enquanto olham para a sua tela (Reed, 2010). A Figura 82 apresenta o marco histórico da evolução dos *smartphones* conforme foi apresentado nesta seção.

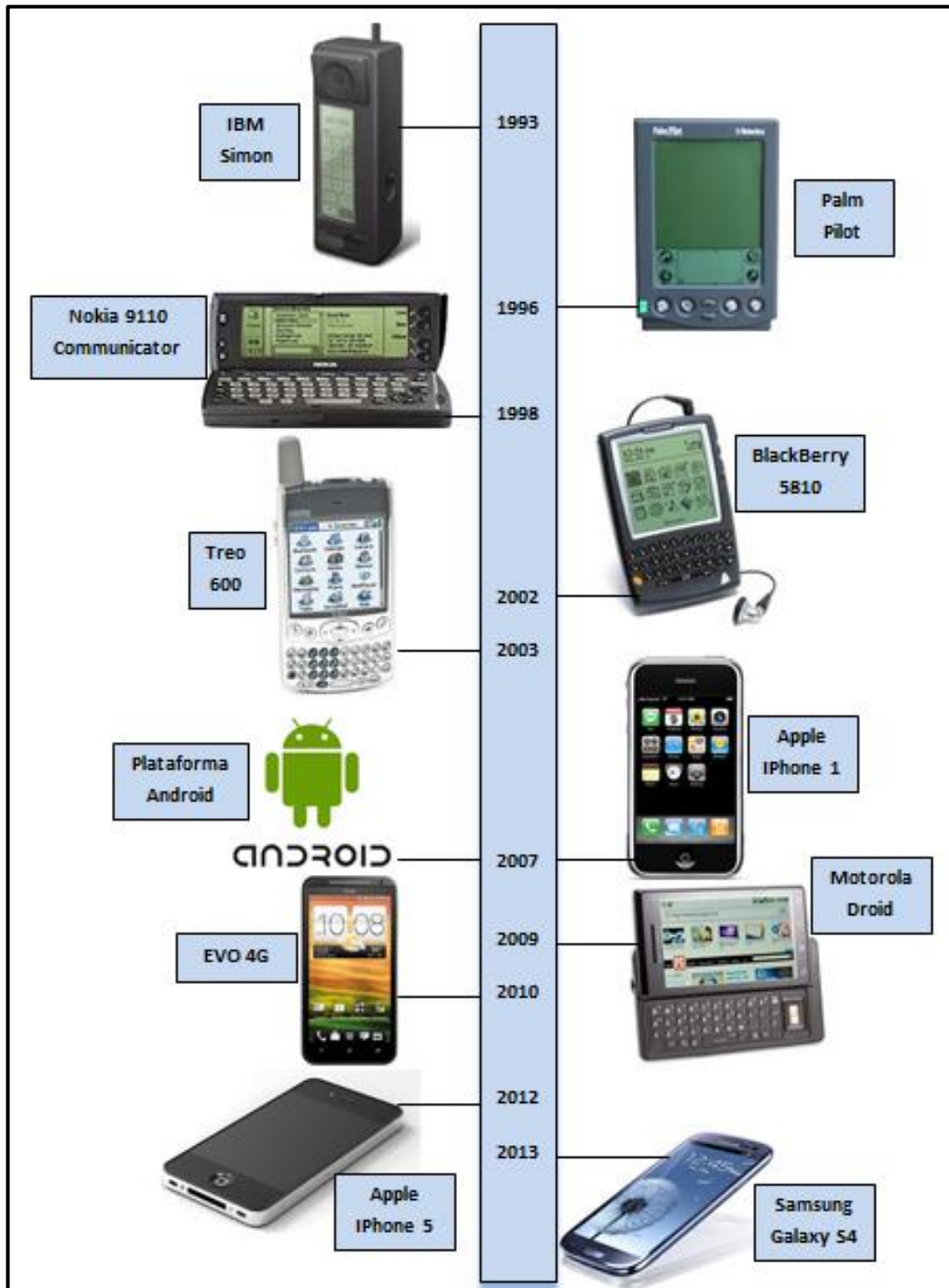


Figura 82 – Evolução histórica dos *smartphones*

Antes de finalizar esta seção, é importante destacar a evolução do hardware ocorrida ao longo das últimas duas décadas. Na Tabela 36, é possível observar o aumento da velocidade de processamento e memória de uma geração para a outra.

Tabela 36 – Configurações dos *smartphones* em três gerações diferentes

Smartphone	Ano de lançamento	Velocidade de Processamento	Memória
IBM Simon	1993	16 MHz	1 MB
Treo 600	2003	144 MHz	32 MB
Samsung Galaxy S4	2013	1.9 GHz	64 GB

As inovações relacionadas aos recursos de hardware, atreladas à evolução das redes sem fio (EDGE, 3G, 4G e WLAN) e à abundante oferta de provedores de serviços Internet, geraram novas oportunidades e desafios para o desenvolvimento de sistemas para *smartphones* (Hansen, 2012).

Ao mesmo tempo em que o hardware utilizado nos *smartphones* se tornou mais poderoso, os softwares que são embarcados nestes dispositivos também passaram por várias etapas evolutivas. Particularmente, a evolução das plataformas Android e iPhone OS é um dos fatores de maior impacto para o sucesso dos *smartphones*. Segundo Mikalajunaite (2011), tanto o Google Play²⁷ como a App Store²⁸ ultrapassaram a marca de 500.000 aplicativos disponíveis. A maturidade das plataformas e a popularidade dos aplicativos estão dando às empresas um novo canal para promover produtos, oferecer novos serviços e expandir as formas de alcançar potenciais clientes.

Além disso, a capacidade de comprar e baixar aplicativos diretamente para os *smartphones* tem provado ser um serviço popular para consumidores e desenvolvedores. Os desenvolvedores são capazes de publicar seus aplicativos rapidamente e os usuários podem navegar em uma biblioteca composta por milhares de aplicações que abrangem, por exemplo, jogos, softwares educacionais e soluções empresariais. Além disso, os sistemas de classificação de produtos possibilitam aos usuários finais opinarem sobre a qualidade e preço das aplicações oferecidas. A facilidade de baixar aplicativos têm motivado as pessoas a buscar, com mais frequência, novos softwares que auxiliem na realização de suas tarefas diárias (Hansen, 2012).

Com a evolução do hardware e software para *smartphones*, ligada à facilidade de obtenção de aplicativos utilizados frequentemente para auxiliar na realização das tarefas diárias das pessoas, torna-se evidente a necessidade de pesquisas na área de usabilidade destes aplicativos, com o objetivo de identificar suas potencialidades e problemas enfrentados por seus usuários.

²⁷ Disponível em: <https://play.google.com/store/apps>

²⁸ Disponível em: www.store.apple.com

ANEXO II – Programação Orientada a Aspectos (POA)

A Programação Orientada a Aspectos foi criada em Palo Alto, nos laboratórios da Xerox, em meados dos anos 90, com a finalidade de estender conceitos de modularização definidos na Programação Orientada a Objetos (POO). Seu objetivo principal é melhorar a separação de interesses das aplicações (Kiczales, 1997). A POA não trabalha isoladamente, ao contrário, é um paradigma que estende outros paradigmas de programação, tais como, a POO e a programação estruturada.

Segundo Lopes (1997), as aplicações estão ampliando os limites das técnicas de programação atuais, de modo que certas características de um sistema afetam seu comportamento de tal forma que as técnicas atuais não conseguem capturar essas propriedades de forma satisfatória.

Um dos elementos centrais da OA é o conceito de Interesse, que são as características relevantes de uma aplicação. Um interesse pode ser dividido em uma série de aspectos que representam os requisitos. Os aspectos podem ser agrupados no domínio da aplicação, compondo os interesses funcionais, que formam a lógica de negócio. Ou podem ser agrupados em elementos que prestam suporte aos interesses funcionais nomeados por interesses sistêmicos, e também chamados de ortogonais ou transversais (Soares et al., 2002).

Segundo Piveta (2001), quando duas propriedades devem ser compostas de maneira diferente e ainda se coordenarem, diz-se que elas são ortogonais entre si. A tentativa de implementar um interesse sistêmico com a aplicação da Orientação a Objetos tem como resultado códigos que se espalham por todo o programa. Para isso dá-se o nome de código espalhado (*Scattering Code*).

Em uma abordagem tradicional Orientada a Objetos, Classes são as entidades responsáveis em representar o domínio do problema, mas não são suficientes para separar os interesses sistêmicos (sincronização e coordenação de múltiplos objetos, tratamento de exceções, persistência, auditoria, etc) dos interesses funcionais (relacionados ao domínio da aplicação) (Kiczales, 1997).

Normalmente, a implementação de interesses sistêmicos ou transversais espalha-se por inúmeros módulos do sistema com conseqüente espalhamento do código por várias classes (*Scattering*). Além disso, em uma única classe pode ocorrer o espalhamento de vários interesses transversais acarretando o entrelaçamento de código (*Tangled*). Esses dois fenômenos provocam o aumento da complexidade do código o que dificulta a manutenção e reutilização do mesmo (Kiczales, 1997).

A POA permite a separação dos interesses pela criação de uma nova dimensão chamada de Aspecto. Assim, os interesses funcionais são codificados nas Classes e os interesses sistêmicos nos Aspectos. Um exemplo desta junção pode ser observado na Figura 83, onde a Classe Produto é combinada com os Aspectos de Auditoria e Controle de Exceções, com a produção de um novo código ao combinar a Classe e os Aspectos.

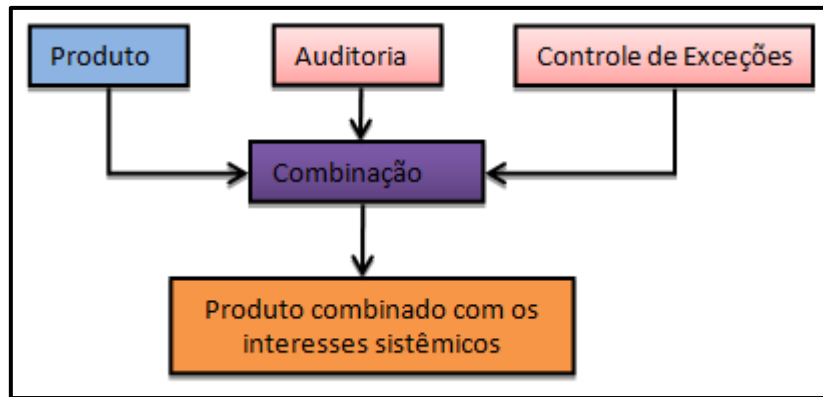


Figura 83 – Exemplo da combinação entre Classes e Aspectos.

A combinação entre Classes e Aspectos ocorre por meio de um componente de software chamado *Weaver*. É um processo que antecede a compilação e gera um código intermediário capaz de produzir a operação desejada ou de permitir a sua realização durante a execução do programa. A Figura 84 representa este processo.

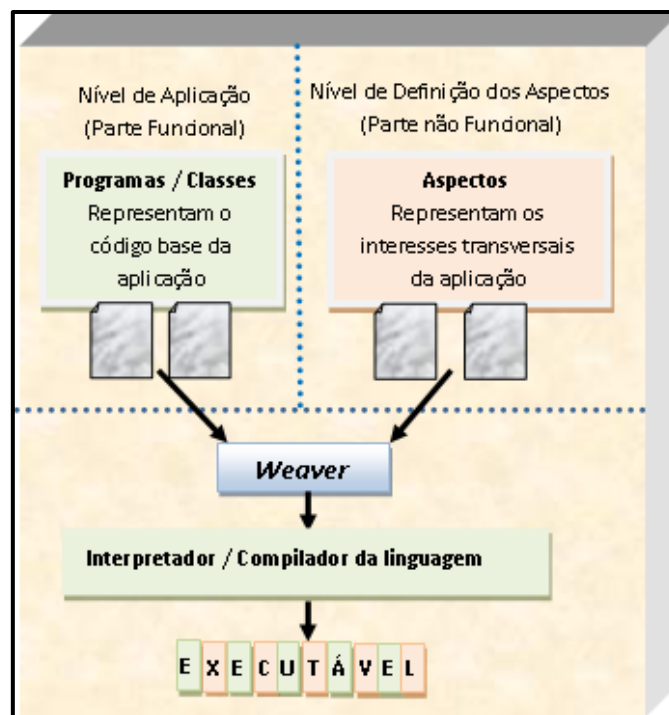


Figura 84 – Representação do *Weaver*.

As classes referentes ao código do negócio nos sistemas não sofrem qualquer alteração para suportar a POA. Isso é feito no momento da combinação entre os componentes e os aspectos. Como pode ser observado na Figura 85, somente o programa de baixo nível é que intercala fisicamente o Código das Classes com o Código dos Aspectos.

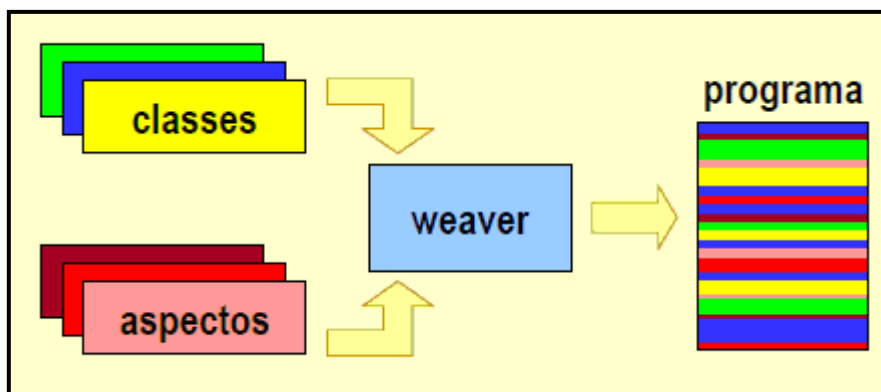


Figura 85 – Representação do código de baixo nível com as Classes e Aspectos.

A estruturação da POA depende da definição dos seguintes componentes:

- **Join Point** – representa um determinado local no fluxo das instruções desenvolvido nas classes. Por exemplo, o início ou o final da execução de um método ou, ainda, o acesso para modificação ou leitura de um atributo.
- **Pointcut** – define regras que permitem especificar quando um *Join Point* será quantificado com uma lógica criada nos Aspectos.
- **Advice** – é o código implementado em um Aspecto responsável em executar alguma lógica quando uma regra definida nos *Pointcuts* é validada, ou seja, quando um *Join Point* é atingido.
- **Aspect** – é o módulo onde os *Advices* e *Pointcuts* são codificados e permite a especificação do código não funcional do sistema.

ANEXO III – Exemplo de um Aspecto para o Mapeamento de Tarefas

```
package br.com.uxeproject.atd.aspect;
import android.content.Context;
import android.content.Intent;
import android.telephony.TelephonyManager;
import android.util.Log;

public aspect UserInteraction
{
    public void com.evancharlton.mileage.DeleteActivity.onUserInteraction()
    {
        super.onUserInteraction();
    }

    public void com.evancharlton.mileage.ImportExportView.onUserInteraction()
    {
        super.onUserInteraction();
    }

    public void com.evancharlton.mileage.io.input.ImportView.onUserInteraction()
    {
        super.onUserInteraction();
    }

    public void com.evancharlton.mileage.io.output.ExportView.onUserInteraction()
    {
        super.onUserInteraction();
    }

    public void com.evancharlton.mileage.Mileage.onUserInteraction()
    {
        super.onUserInteraction();
    }

    public void com.evancharlton.mileage.SettingsView.onUserInteraction()
    {
        super.onUserInteraction();
    }

    public void com.evancharlton.mileage.views.intervals.AddServiceIntervalView.onUserInteraction()
    {
        super.onUserInteraction();
    }

    public void com.evancharlton.mileage.views.intervals.ServiceIntervalsView.onUserInteraction()
    {
        super.onUserInteraction();
    }
}
```

```

private TelephonyManager manager;
private String androidID;
private Context appContext;

private boolean userInteraction = false;

/**
 * JoinPoint to get all executions of methods named onUserInteraction
 * This process will allow the Global JoinPoint to get the first execution
 * after the UI and send this one to ATD
 */

before() : execution(void *.onUserInteraction())
{
    userInteraction = true;
    Log.d("UserInteraction", "Some UI happened");
}

/**
 * It's a Global JoinPoint, get all executions of all methods.
 *
 * The first method execution that was dispatched from an instance of
 * {@link android.content.Context} will be used to get some initial parameters
 * like cellphoneID and androidID.
 *
 * The first method execution preceded by an user interaction will be used to
 * send the method name and the hash codes of received parameters to ATD app.
 */

before() : execution(* *.*(*))
{
    String longString = thisJoinPoint.toLongString();
    Log.d("GLOBAL", longString);

    Object target = thisJoinPoint.getTarget();
    if ( target != null && target instanceof Context )
    {
        appContext = ((Context)target).getApplicationContext();

        if ( manager == null )
        {
            androidID = android.provider.Settings.System.getString(
                appContext.getContentResolver(), android.provider.Settings.System.ANDROID_ID);
            manager = (TelephonyManager)appContext.getSystemService(Context.TELEPHONY_SERVICE);

            Intent methodIntent = new Intent("BroadcastATD_appInfo");
            methodIntent.putExtra("androidID", androidID);
            methodIntent.putExtra("deviceID", manager.getDeviceId());
            appContext.sendBroadcast(methodIntent);
            Log.i("InitParams", "Initial params loaded");
        }
    }
}

```

```

if ( appContext != null && userInteraction )
{
    userInteraction = false;
    String method = longString.substring(0,longString.lastIndexOf("("));
    method = method.substring(method.lastIndexOf(" ",method.lastIndexOf("("))+1);
    StringBuilder parametersHashCode = new StringBuilder();
    for (Object arg : thisJoinPoint.getArgs())
    {
        if ( arg.getClass().isPrimitive() )
        {
            //get object real value (e.g. a boolean returns "true" or "false")
            parametersHashCode.append(arg.toString());
        }
        else
        {
            //hash code for this complex object
            parametersHashCode.append(arg.hashCode());
        }
        parametersHashCode.append(";");
    }

    Log.i("Method to send...", method);
    Intent methodIntent = new Intent("BroadcastATD_method");
    methodIntent.putExtra("method", method);
    methodIntent.putExtra("parameters", parametersHashCode.toString());
    appContext.sendBroadcast(methodIntent);
}
}
}

```

ANEXO IV – Exemplo de um Aspecto com Métricas de Usabilidade e Contexto

```
package com.evancharlton.mileage.aspect;
import java.io.StringWriter;
import java.util.ArrayList;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.*;
import javax.xml.transform.dom.*;
import javax.xml.transform.stream.*;
import org.aspectj.lang.JoinPoint;
import org.aspectj.lang.reflect.CodeSignature;
import org.w3c.dom.Document;
import org.w3c.dom.Element;
import org.w3c.dom.Text;
import android.content.Context;
import android.hardware.Sensors;
import android.hardware.SensorEvent;
import android.hardware.SensorEventListener;
import android.hardware.SensorManager;
import android.telephony.TelephonyManager;
import android.util.Log;
import android.os.Bundle;
import android.widget.TextView;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.app.Activity;

public aspect nameTask_Aspect
{

    declare parents : Mileage implements LocationListener;

    pointcut methodExecution() : execution(* *.*(*));

    pointcut resetAspect() : execution(public static void com.evancharlton.mileage.aspect.AspectsHelper. taskEnded());

    private int totalActions = 0;
    private int totalErrors = 0;
    private int totalErrorsAfterHelp = 0;
    private int totalHelpUsage = 0;
    private int taskState = 0;
    private long taskStartedTime;
    private long taskFinishedTime;
    private long taskCompletedTime;
    private boolean taskAbandoned = false;
    private boolean taskStarted = false;
    private boolean usedHelp = false;
    boolean accelerometerAvailable = false;
```

```

boolean isEnabled = false;
private Location location;
LocationManager locationManager;

private void writeXMLTaskFile(JoinPoint jp)
{
    if(jp.getThis() instanceof Context)
    {
        Context actualContext = (Context)jp.getThis();
        try
        {
            DocumentBuilderFactory dbfac = DocumentBuilderFactory.newInstance();
            DocumentBuilder docBuilder = dbfac.newDocumentBuilder();
            Document doc = docBuilder.newDocument();
            TelephonyManager tm = TelephonyManager.getInstance(actualContext.getSystemService
                (Context.TELEPHONY_SERVICE));

            String deviceId = tm.getDeviceId();
            Element root = doc.createElement("tasks");
            root.setAttribute("IDUser", deviceId);
            doc.appendChild(root);
            Element task = doc.createElement("task");
            task.setAttribute("name", "nameTask");
            task.setAttribute("timestamp", Long.toString(this.taskStartedTime));
            root.appendChild(task);
            Element metrics = doc.createElement("metrics");
            Element context = doc.createElement("contexts");
            task.appendChild(metrics);
            Element totalActions = doc.createElement("totalActions");
            metrics.appendChild(totalActions);
            Text actionsValue = doc.createTextNode(Integer.toString(this.totalActions));
            totalActions.appendChild(actionsValue);
            Element taskComplete = doc.createElement("taskComplete");
            metrics.appendChild(taskComplete);
            Text taskCompleteValue = doc.createTextNode(Boolean.toString(this.taskAbandoned == false));
            taskComplete.appendChild(taskCompleteValue);
            Element totalErrors = doc.createElement("totalErrors");
            metrics.appendChild(totalErrors);
            Text errorsValue = doc.createTextNode(Integer.toString(this.totalErrors));
            totalErrors.appendChild(errorsValue);
            Element taskCompletedPerfect = doc.createElement("taskCompletedPerfect");
            metrics.appendChild(taskCompletedPerfect);
            Text taskPerfectValue = doc.createTextNode(Boolean.toString(this.totalErrors == 0));
            taskCompletedPerfect.appendChild(taskPerfectValue);
            Element executionTime = doc.createElement("executionTime");
            metrics.appendChild(executionTime);
            Text taskExecutionTimeValue = doc.createTextNode(Long.toString(this.taskCompletedTime));
            executionTime.appendChild(taskExecutionTimeValue);
            Element utilizationTime = doc.createElement("utilizationTime");
            metrics.appendChild(utilizationTime);
            Element helpUsed = doc.createElement("helpUsed");
            metrics.appendChild(helpUsed);
            Text helpUsedValue = doc.createTextNode(Integer.toString(totalHelpUsage));
        }
    }
}

```

```

helpUsed.appendChild(helpUsedValue);
Element errorsAfterHelp = doc.createElement("errorsAfterHelp");
metrics.appendChild(errorsAfterHelp);
Text errorsAfterHelpValue = doc.createTextNode(Integer.toString(totalErrorsAfterHelp));
errorsAfterHelp.appendChild(errorsAfterHelpValue);
task.appendChild(context);
for(int i=0; i < this.totalActions; i++)
{
    Element contextData = doc.createElement("contextData");
    context.appendChild(contextData);
    Element correctAction = doc.createElement("correctAction");
    contextData.appendChild(correctAction);
    Element luminosity = doc.createElement("luminosity");
    contextData.appendChild(luminosity);
    Text luminosityValue = doc.creatTextNode(this.readLightSensor(this.getBlunde()));
    luminosityValue.appendChild(luminosityValue);
    Element movement = doc.createElement("movement");
    contextData.appendChild(movement);
    this.acelerametroReader(false);
    Text movementValue = doc.createTextNode(this.readAccelerometer());
    movementValue.appendChild(movementValue);
    Element speed = doc.createElement("speed");
    contextData.appendChild(speed);
    this.gpsReader(actualContext);
    Text gpsValue = doc.createTextNode(this.getLocation());
    gpsValue.appendChild(gpsValue);
}
TransformerFactory transfac = TransformerFactory.newInstance();
Transformer trans = transfac.newTransformer();
trans.setOutputProperty(OutputKeys.OMIT_XML_DECLARATION, "yes");
trans.setOutputProperty(OutputKeys.INDENT, "yes");
StringWriter sw = new StringWriter();
StreamResult result = new StreamResult(sw);
DOMSource source = new DOMSource(doc);
trans.transform(source, result);
String xmlString = sw.toString();
AspectsHelper.saveXMLFile(actualContext, xmlString, "tasks.xml");
AspectsHelper.taskEnded();
}
catch (Exception e)
{
    System.out.println(e);
}
}
}

```

```

public void readLightSensor (Bundle savedInstanceState)
{
    this.onCreate(savedInstanceState);
    setContentView(R.layout.main);
    TextView textLightSensor = (TextView)findViewById(R.id.lightSensor);
    textLightSensorData = (TextView)findViewById(R.id.lightSensorData);
}

```

```

mySensorManager = (SensorManager) getSystemService(Context.SENSOR_SERVICE);
myLightSensor = mySensorManager.getDefaultSensor(Sensor.TYPE_LIGHT);
if (myLightSensor == null)
{   textLightSensor.setText("No Light Sensor!");
}
else
{   textLightSensor.setText(myLightSensor.getName());
    mySensorManager.registerListener(lightSensorEventListener, myLightSensor,
                                    SensorManager.SENSOR_DELAY_NORMAL);
}
}

```

```

SensorEventListener lightSensorEventListener = new SensorEventListener()
{   public void onAccuracyChanged(Sensor arg0, int arg1)
    {
    }

    public void onSensorChanged(SensorEvent arg0)
    {   if(arg0.sensor.getType()==Sensor.TYPE_LIGHT)
        {   textLightSensorData.setText("Light Sensor Date:" + String.valueOf(arg0.values[0]));
        }
    }
};

```

```

public void gpsReader(Context c)
{
    location = null;
    locationManager = (LocationManager) c.getSystemService(c.LOCATION_SERVICE);
    locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0, 0, this);
}

```

```

public void onLocationChanged(Location location)
{
    this.location = location;
}

```

```

public void onStatusChanged(String arg0, int arg1, Bundle arg2)
{
}

```

```

public void onProviderEnabled(String arg0)
{
}

```

```

public void onProviderDisabled(String arg0)
{
}

```

```

public Location getLocation()
{   return location;
}

```

```

public void accelerometerReader(boolean doEnable) throws UnsupportedOperationException
{
    for (String aSensor : Sensors.getSupportedSensors())
    { if (aSensor.equals(Sensors.SENSOR_ACCELEROMETER))
      { accelerometerAvailable = true;
        }
      }
    if (!accelerometerAvailable)
    { throw new UnsupportedOperationException( "Accelerometer is not available.");
      }
    if (doEnable)
    { setEnableAccelerometer(true);
      }
    }
}

```

```

public void setEnableAccelerometer(boolean doEnable) throws UnsupportedOperationException
{ if (!accelerometerAvailable)
  { throw new UnsupportedOperationException( "Accelerometer is not available.");
    }
  if (doEnable && !this.isEnabled)
  { Sensors.enableSensor(Sensors.SENSOR_ACCELEROMETER);
    this.isEnabled = true;
  }
  else
  { if (!doEnable && this.isEnabled)
    { Sensors.disableSensor(Sensors.SENSOR_ACCELEROMETER);
      this.isEnabled = false;
    }
  }
}

```

```

public float[] readAccelerometer() throws UnsupportedOperationException, IllegalStateException
{ if (!accelerometerAvailable)
  { throw new UnsupportedOperationException( "Accelerometer is not available.");
    }
  if (!this.isEnabled)
  { throw new IllegalStateException("Accelerometer was set to disabled!");
    int sensorValues = Sensors.getNumSensorValues(Sensors.SENSOR_ACCELEROMETER);
    float[] out = new float[sensorValues];
    Sensors.readSensor(Sensors.SENSOR_ACCELEROMETER, out);
    return out;
  }
}

```

```

before() : methodExecution()
{
    if(!thisJoinPointStaticPart.getSignature().getDeclaringTypeName().contains("AspectsHelper") &&
        !thisJoinPointStaticPart.getSignature().getDeclaringTypeName().contains("AccelerometerListener") &&
        !thisJoinPointStaticPart.getSignature().getDeclaringTypeName().contains("AspectActions"))
    {
        String executedMethod = "";
        CodeSignature cs = (CodeSignature)thisJoinPointStaticPart.getSignature();
    }
}

```



```

String[] parameters = cs.getParameterNames();
String parametersString = "";
for(int i = 0; i < parameters.length; i++)
{
    if(i != 0)
    {
        parametersString += ", ";
    }
    parametersString += cs.getParameterTypes()[i].getName() + " ";
    parametersString += parameters[i];
}
executedMethod = thisJoinPointStaticPart.getSignature().getDeclaringTypeName()+"."+
                thisJoinPointStaticPart.getSignature().getName()+"("+parametersString+")";
verifyTaskExecutedMethod(executedMethod, thisJoinPoint);
}
}

private void verifyTaskExecutedMethod(String executedMethod, JoinPoint jp)
{
    if((taskState == 0) &&
        (executedMethod.equals("com.evancharlton.mileage.AddFillUpView.onCreate(android.os.Bundle
                                                                    savedInstanceState)")))
    {
        this.taskStartedTime = System.currentTimeMillis();
        taskStarted = true;
        totalActions++;
        taskState = 1;
    }
    else if((taskState == 1) &&
            (executedMethod.equals("com.evancharlton.mileage.AddFillUpView.showMessage(boolean success)")))
    {
        this.taskFinishedTime = System.currentTimeMillis();
        this.taskCompletedTime = this.taskFinishedTime - this.taskStartedTime;
        totalActions++;
        taskState = 0;
        this.writeXMLTaskFile(jp);
        this.totalActions = 0;
    }
    else if(executedMethod.equals("void ajuda()"))
    {
        totalActions++;
        usedHelp = true;
        totalHelpUsage++;
    }
    else if(executedMethod.equals
            ("name.bagi.levente.pedometer.PedometerSettings.saveServiceRunningWithNullTimestamp(boolean)"))
    {
        totalActions++;
        taskState = 0;
        this.taskAbandoned = true;
        this.writeXMLTaskFile(jp);
        this.taskStarted = false;
    }
}

```

```

        this.taskAbandoned = false;
        this.totalActions = 0;
    }
    else
    {
        if(taskStarted && ((executedMethod.contains("onClick") || executedMethod.contains("onCreate"))))
        {
            totalActions++;
            totalErrors++;
        }
    }
}
after() : resetAspect()
{
    Log.d("XML", "aspecto reiniciado");
    this.resetAspectMethod();
}

public void resetAspectMethod()
{
    this.totalActions = 0;
    this.taskState = 0;
    this.totalErrors = 0;
    this.totalErrorsAfterHelp = 0;
    this.totalHelpUsage = 0;
    this.taskCompletedTime = 0;
    this.taskStartedTime = 0;
    this.usedHelp = false;
    this.taskStarted = false;
    this.taskAbandoned = false;
    this.taskFinishedTime=0;
    accelerometerAvailable = false;
    isEnabled = false;
    location = null;
    locationManager = null;
}
}
}

```

ANEXO V – Exemplo de um Aspecto com Métricas Subjetivas

```
package com.evancharlton.mileage.aspect;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.OutputStreamWriter;
import java.io.StringWriter;
import java.util.ArrayList;
import javax.xml.parsers.DocumentBuilder;
import javax.xml.parsers.DocumentBuilderFactory;
import javax.xml.transform.OutputKeys;
import javax.xml.transform.Transformer;
import javax.xml.transform.TransformerFactory;
import javax.xml.transform.dom.DOMSource;
import javax.xml.transform.stream.StreamResult;
import org.aspectj.lang.JoinPoint;
import org.w3c.dom.Document;
import org.w3c.dom.Element;
import android.app.Activity;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.telephony.TelephonyManager;
import android.util.Log;

public aspect Mileage_SubjectiveMetrics
{
    private int timesExecuted = 0;
    private ArrayList<Integer> ratings = new ArrayList();
    private ArrayList<Long> timestamps = new ArrayList();
    private long lasttimestamp = 0;
    private boolean notYetSent = true;
    final CharSequence[] options = { "Completely Satisfied", "Satisfied", "Mildly Satisfied", "Unsatisfied", "Completely
        Unsatisfied" };

    pointcut methodExecution() : execution(* *.*(*));

    after() : methodExecution()
    {
        if (thisJoinPoint.getThis() instanceof Context && thisJoinPoint.getThis() instanceof Activity)
        {
            Activity a = (Activity) thisJoinPoint.getThis();
            final Long timestamp = System.currentTimeMillis();
            if (timesExecuted == 0 || (timestamp >= (lasttimestamp + 180000)))
            {
                AlertDialog.Builder builder = new AlertDialog.Builder(a);
                builder.setTitle("Using your mobile phone now, you are:");
                builder.setItems(options, new DialogInterface.OnClickListener()
                {
                    public void onClick(DialogInterface dialog, int item)
                    {
                        ratings.add(Math.abs(item - 5));
                    }
                });
            }
        }
    }
}
```

```

        timestamps.add(timestamp);
        notYetSent = true;
    }
});
AlertDialog alert = builder.create();
alert.show();
lasttimestamp = timestamp;
timesExecuted++;
}
else if (ratings.size() > 0 && notYetSent)
{
    if (thisJoinPoint.getThis() instanceof Context)
    {
        notYetSent = false;
        writeXMLSubjetctiveData(thisJoinPoint);
        AspectsHelper.uploadFileFTP("subjectivedata.xml", (Context) thisJoinPoint.getThis());
        ratings.clear();
        timestamps.clear();
    }
}
}
}
}

```

```

private void writeXMLSubjetctiveData(JoinPoint jp)
{
    Context actualContext = (Context) jp.getThis();
    try
    {
        DocumentBuilderFactory dbfac = DocumentBuilderFactory.newInstance();
        DocumentBuilder docBuilder = dbfac.newDocumentBuilder();
        Document doc = docBuilder.newDocument();
        TelephonyManager tm = (TelephonyManager) actualContext.getSystemService(Context.TELEPHONY_SERVICE);
        Element root = doc.createElement("subjectiveMetric");
        root.setAttribute("question", "Qual a sua satisfação com relação a usabilidade deste aplicativo?");
        root.setAttribute("IDUser", tm.getDeviceId());
        root.setAttribute("appName", "Mileage");
        doc.appendChild(root);
        for (int i = 0; i < timestamps.size(); i++)
        {
            Element subjectiveData = doc.createElement("subjectiveData");
            subjectiveData.setAttribute("rating", Integer.toString(ratings.get(i)));
            subjectiveData.setAttribute("timestamp", Long.toString(timestamps.get(i)));
            root.appendChild(subjectiveData);
        }
        TransformerFactory transfac = TransformerFactory.newInstance();
        Transformer trans = transfac.newTransformer();
        trans.setOutputProperty(OutputKeys.OMIT_XML_DECLARATION, "yes");
        trans.setOutputProperty(OutputKeys.INDENT, "yes");
        StringWriter sw = new StringWriter();
        StreamResult result = new StreamResult(sw);
        DOMSource source = new DOMSource(doc);
        trans.transform(source, result);
    }
}

```

```
String xmlString = sw.toString();
Log.d("XML", xmlString);
AspectsHelper.saveXMLFile(actualContext, xmlString, "subjectivedata.xml");
}
catch (Exception e)
{
    System.out.println(e);
}
}
}
```

ANEXO VI – Dicionário de Dados do BD

Tabela User

Nesta tabela, estão armazenados os dados referentes aos usuários das aplicações utilizadas nos experimentos. Ela possibilita identificar quais as tarefas cada usuário executou e se esse mesmo usuário é um usuário normal ou um engenheiro de testes.

	Nome	Tipo	Descrição
PK	idUser	varchar(45)	Identificador único do usuário. Esse conjunto de caracteres é uma cópia do IMEI do celular.
	type	varchar(30)	Indica o tipo do usuário. Os valores possíveis são: Especialista ou Normal.
	age	int	Idade do usuário (em anos).
	education	varchar(45)	Grau de escolaridade do usuário. Os valores possíveis são: Fundamental, Médio, Graduado e Pós-graduado.
	formation	varchar(45)	Área de formação do usuário. Os valores possíveis são: Exatas, Humanas, Saúde, Outras (inclui os não formados).
	purchasingPower	varchar(45)	Poder aquisitivo do usuário. Os valores possíveis são: Baixo (até 3 salários mínimos), Médio (até 20 salários mínimos) e Alto (mais do que 20 salários mínimos). Dados baseados na classificação do DIEESE.
	occupation	varchar(45)	Área de trabalho atual do usuário. Os valores possíveis são: Educação, Indústria, Comércio e Serviços.
	screenSize	float	Tamanho da tela do <i>smartphone</i> que será utilizado pelo usuário. Os valores possíveis são: Pequeno (até 2.4 polegadas), Médio (entre 2.4 e 3.5 polegadas) e Grande (acima de 3.5 polegadas)
	screenResolution	varchar(45)	Resolução da tela do <i>smartphone</i> que será utilizado pelo usuário. Os valores possíveis são: Baixa (Menor ou igual a 320X240 pixels), Média (entre 320X240 pixels e 320X480 pixels) e Alta (acima 320X480 pixels)

Tabela Execution

Nesta tabela, estão armazenados os dados relativos à execução de cada tarefa e relaciona o usuário, a tarefa, as métricas de usabilidade, as métricas contextuais, a data e a hora da execução.

	Nome	Tipo	Descrição
PK	idExecution	int	Identificador único da execução.
FK	idUser	varchar(45)	Identificador único do usuário.
FK	idUsabilityMetrics	int	Identificador único das métricas de usabilidade.
FK	idTask	int	Identificador único da tarefa.
FK	idContextMetrics	int	Identificador único das métricas contextuais.
	dateTime	timestamp	Guarda a data e hora da execução da tarefa.

Tabela Task

Esta tabela contém os dados descritivos das tarefas como, por exemplo, o nome e a qual aplicação pertence.

	Nome	Tipo	Descrição
PK	idTask	int	Identificador único da tarefa.
FK	idApplication	int	Identificador único da aplicação a qual essa tarefa pertence.
	description	varchar(45)	Descrição da tarefa.

Tabela ReferenceValues

Esta tabela contém os valores de referência das tarefas sob o ponto de vista da Equipe de Avaliação. Os valores são definidos durante o mapeamento das tarefas para que possam ser comparados com os valores reais obtidos pelos usuários.

	Nome	Tipo	Descrição
PK	idReferenceValues	int	Identificador único dos valores de referência.
FK	idTask	int	Identificador único da tarefa a qual esses valores de referência estão associados.
	erros	float	Percentual aceitável de ações erradas.
	actionsNumber	float	Média de ações para completar a tarefa.
	timeToComplete	float	Tempo médio para completar a tarefa.
	minimumUtilizationTime	float	Tempo mínimo necessário para comprovar a real utilização da tarefa (em segundos).
	dropouts	float	Número médio aceitável de abandonos da tarefa.
	useHelpFrequency	float	Percentual aceitável de uso da opção de Ajuda.
	errorsNumberAfterUseHelp	float	Percentual aceitável de erros cometidos após utilizar a opção de Ajuda.

Tabela SubjectiveMetrics

Esta tabela contém as questões e as respostas fornecidas pelos usuários com relação a satisfação com determinados aspectos de usabilidade dos aplicativos.

	Nome	Tipo	Descrição
PK	idSubjectiveMetrics	int	Identificador único da tabela.
FK	idApplication	int	Identificador único da aplicação.
FK	idUser	varchar(45)	Identificador único do usuário.
	date	timestamp	Data de resposta da pergunta.
	question	varchar(255)	Pergunta de caráter subjetivo a ser submetida ao usuário.
	answer	int	Inteiro que indicará a resposta do usuário e o grau de satisfação do mesmo. Os valores variam de 1 (muito insatisfeito) a 5 (muito satisfeito).

Tabela ContextMetrics

Nessa tabela, estão os dados contextuais coletados pelos sensores dos *smartphones* durante a execução das tarefas e possibilita identificar as condições do cenário de interação.

	Nome	Tipo	Descrição
PK	idContextMetrics	int	Identificador único do contexto.
	luminosity	varchar(45)	Campo que indica a luminosidade do ambiente medida em LUX. Os valores possíveis são: Baixo (até 100 LUX), Médio (entre 100 e 10.000 LUX) e Alto (acima de 10.000 LUX).
	noise	varchar(45)	Campo que indica a quantidade de ruído medido pelo aparelho. (Não foi implementado no projeto).
	speed	varchar(45)	Campo que indica a velocidade de deslocamento medida pelo GPS do aparelho. Os valores possíveis são: Parado (menor que 0,2 m/s), Caminhando (mais que 0,2 m/s e menos que 2,7 m/s) e Motorizado (acima de 2,7 m/s).
	movement	varchar(45)	Campo que indica a posição do aparelho medido pelo acelerômetro do aparelho. Os Valores possíveis são: Vertical, Horizontal e Mista.

Tabela UsabilityMetrics

Nesta tabela, estão armazenados os valores obtidos durante a execução de cada tarefa e possibilita avaliar o desempenho do usuário e identificar problemas de usabilidade.

	Nome	Tipo	Descrição
PK	idUsabilityMetrics	int	Identificador único da métrica.
	completeTask	boolean	Campo que indica se a tarefa foi completada.
	completePerfectTask	boolean	Campo que indica se a tarefa foi completada sem erros.
	usedTime	float	Guarda quanto tempo a tarefa completada foi utilizada pelo usuário (medido em segundos).
	executionTime	float	Guarda o tempo que o usuário levou para completar a tarefa (medido em segundos).
	totalError	int	Número de erros cometidos pelo usuário até completar a tarefa.
	totalErrorAfterHelp	int	Quantidade de erros após a visualização da ajuda.
	totalHelp	int	Número de acessos a opção de Ajuda durante a execução de uma tarefa.
	totalActions	int	Total de ações que o usuário fez até finalizar a tarefa.

Tabela Application

Esta tabela contém os dados referentes às aplicações avaliadas.

	Nome	Tipo	Descrição
PK	idApplication	int	Identificador único da aplicação.
	description	varchar(45)	Descrição das aplicações.
	publisher	varchar(45)	Descrição da empresa que publicou o software.

ANEXO VII – Dicionário de Dados do DW

Tabela FACT_Execution_Task

Essa tabela permite relacionar todas as tabelas dimensionais do DW.

	Nome	Tipo	Descrição
PK	idExecution	int	Identificador único da execução.
PF	idApplication	int	Identificador único da aplicação.
PF	idTask	int	Identificador único da tarefa.
PF	idUser	varchar(45)	Identificador único do usuário.
PF	idContext	int	Identificador único dos valores de contexto.
PF	idTime	int	Identificador único do tempo.
	totalUsedTime	int	Tempo total que a tarefa completada foi utilizada pelo usuário (em segundos).
	totalExecutionTime	int	Tempo total para completar a tarefa (em segundos).
	totalCompleteTask	int	Total de tarefas completadas com sucesso.
	totalCompletePerfectTask	int	Total de tarefas completadas com sucesso e sem erros.
	totalIncompleteTask	int	Total de tarefas que foram abandonadas pelo usuário.
	totalActions	int	Total de ações que o usuário fez até finalizar a tarefa.
	totalError	int	Quantidade de erros cometidos pelo usuário até a tarefa ser finalizada.
	totalErrorAfterHelp	int	Quantidade de erros após a visualização da Ajuda.
	totalHelp	int	Quantidade de visualizações da ajuda pelo usuário.
	totalCorrectAction		Total de ações corretas.

Tabela DIM_Context

Nessa dimensão, encontram-se os dados referentes às condições em que o ambiente estava no momento da execução da tarefa.

	Nome	Tipo	Descrição
PK	idContext	int	Identificador único do contexto.
	luminosity	varchar(45)	Campo que indica a luminosidade do ambiente medida em LUX. Os valores possíveis são: Baixo (até 100 LUX), Médio (entre 100 e 10.000 LUX) e Alto (acima de 10.000 LUX).
	speed	varchar(45)	Campo que indica a velocidade de deslocamento medida pelo GPS do aparelho. Os valores possíveis são: Parado (menor que 0,2 m/s), Caminhando (mais que 0,2 m/s e menos que 2,7 m/s) e Motorizado (acima de 2,7 m/s).
	movement	varchar(45)	Campo que indica a posição do aparelho medido pelo acelerômetro do aparelho. Os Valores possíveis são: Vertical, Horizontal e Mista.

Tabela DIM_User

Nesta dimensão, estão armazenados os dados referentes aos usuários. Ela possibilita relacionar os usuários com as tarefas executadas, identificar o seu perfil e as características do seu *smartphone*.

	Nome	Tipo	Descrição
PK	idUser	varchar(45)	Identificador único do usuário. Esse número é a cópia do IMEI do celular.
	type	varchar(45)	Tipo do usuário (Experiente ou Normal).
	formation	varchar(45)	Área de formação do usuário. Os valores possíveis são: Exatas, Humanas, Saúde, Outras (inclui os não formados).
	purchasingPower	varchar(45)	Poder aquisitivo do usuário. Os valores possíveis são: Baixo (até 3 salários mínimos), Médio (até 20 salários mínimos) e Alto (mais do que 20 salários mínimos). Dados baseados na classificação do DIEESE.
	age	int	Idade do usuário (em anos).
	education	varchar(45)	Grau de escolaridade do usuário. Os valores possíveis são: Fundamental, Médio, Graduado e Pós-graduado.
	occupation	varchar(45)	Área de trabalho atual do usuário. Os valores possíveis são: Educação, Indústria, Comércio e Serviços.
	screenSize	float	Tamanho da tela do <i>smartphone</i> que será utilizado pelo usuário. Os valores possíveis são: Pequeno (até 2.4 polegadas), Médio (entre 2.4 e 3.5 polegadas) e Grande (acima de 3.5 polegadas).
	screenResolution	varchar(45)	Resolução da tela do <i>smartphone</i> que será utilizado pelo usuário. Os valores possíveis são: Baixa (Menor ou igual a 320X240 pixels), Média (entre 320X240 pixels e 320X480 pixels) e Alta (acima 320X480 pixels).

Tabela DIM_Application

Esta dimensão contém os dados das aplicações e informações referentes as avaliações subjetivas definidas pelos usuários.

	Nome	Tipo	Descrição
PK	idApplication	int	Identificador único da aplicação.
	publisher	varchar(45)	Descrição da empresa que desenvolveu o software.
	application	varchar(45)	Descrição das aplicações.

Tabela DIM_Time

Esta dimensão possibilita a visualização dos dados sumarizados por diferentes períodos.

	Nome	Tipo	Descrição
PK	idTime	int	Identificador único do tempo.
	date_value	date	Campo que contém a data completa.
	year	Int	Campo que contém o ano referente à execução.
	month	int	Campo que contém o mês referente à execução.
	month_name	varchar(20)	Campo que contém o nome do mês referente à execução.
	day	int	Campo que contém o dia em formato numérico referente à execução.
	day_name	varchar(20)	Campo que contém o nome do dia referente à execução.
	hour	int	Campo que contém a hora referente à execução.
	dayShift	varchar(20)	Campo que contém o turno do dia onde ocorreu a execução. Os valores possíveis são (Madrugada, Matutino, Vespertino e Noturno).

Tabela ReferenceValues

Esta tabela contém os valores de referência das tarefas sob o ponto de vista da Equipe de Avaliação. Os valores são definidos durante o mapeamento das tarefas para que possam ser comparados com os valores reais obtidos pelos usuários.

	Nome	Tipo	Descrição
PK	idReferenceValues	int	Identificador único dos valores de referência.
FK	idTask	int	Identificador único da tarefa a qual esses valores de referência estão associados.
	timeToComplete	float	Tempo médio para completar a tarefa (em segundos).
	minimumUtilizationTime	float	Tempo mínimo necessário para comprovar a real utilização da tarefa (em segundos).
	dropouts	float	Número médio aceitável de abandonos da tarefa.
	erros	float	Percentual aceitável de ações erradas.
	performedActionsNumber	float	Média de ações para completar a tarefa.
	useHelpFrequency	float	Percentual aceitável de uso da opção de Ajuda.
	errorsNumberAfterUseHelp	float	Percentual aceitável de erros cometidos após utilizar a opção de Ajuda.

Tabela DIM_Task

Esta dimensão permite a identificação das tarefas.

	Nome	Tipo	Descrição
PK	idTask	int	Identificador único da tarefa.
	task	varchar(45)	Descrição da tarefa

Tabela Questions

Esta tabela contém a totalização das respostas subjetivas fornecidas pelos usuários referente a cada uma das perguntas.

	Nome	Tipo	Descrição
PK	idQuestions	int	Identificador único da pergunta.
FK	idApplication	int	Identificador único do aplicativo que relaciona a pergunta ao aplicativo.
	question	varchar(255)	Descrição da pergunta.
	veryUnsatisfied	int	Valor total de muito insatisfeito.
	unsatisfied	int	Valor total de insatisfeito.
	indifferent	int	Valor total de indiferente.
	satisfied	int	Valor total de satisfeito.
	verySatisfied	int	Valor total de muito satisfeito.

ANEXO VIII – Termo de Uso dos Aplicativos utilizados no Experimento de Campo

TERMO DE USO

Os softwares disponíveis no portal UXE Project (<http://uxeproject.no-ip.org>) foram obtidos gratuitamente na Internet. A versão disponível neste site foi instrumentada com módulos de avaliação de usabilidade. As avaliações realizadas são frutos do modelo proposto e implementado pelo doutorando Artur Henrique Kronbauer, aluno do Doutorado Multiinstitucional em Ciência da Computação (Universidade Federal da Bahia, Universidade Estadual de Feira de Santana e Universidade Salvador).

Os dados capturados podem ser classificados em quatro grupos:

1 – Perfil do usuário – São os dados que devem ser preenchidos para possibilitar o download dos softwares.

2 – Dados estatísticos – Referem-se a dados relacionados a execução das tarefas disponibilizadas nos softwares. São medidos, por exemplo, o tempo para a execução de uma tarefa, número de erros cometidos, acesso a ajuda, entre outras.

3 – Dados contextuais – Relacionam a execução das tarefas disponíveis nos softwares ao contexto onde ocorrem como, por exemplo, a intensidade da luz no ambiente, usuário parado ou caminhando, aparelho na horizontal ou na vertical e o grau de barulho do ambiente.

4 – Dados subjetivos – São obtidos através de questionários submetidos ao usuário para coletar a sua satisfação com relação ao aplicativo em uso.

Os dados coletados serão usados para validar o modelo proposto para avaliação de usabilidade e farão parte de análises que podem estar descritas em artigos e na tese de doutorado do aluno Artur Henrique Kronbauer. Os dados não serão usados com outro objetivo e jamais será exposto individualmente um participante do experimento.

Aceitando este termo, você concorda em participar, gratuitamente, dos experimentos e permitir que os dados referentes às suas interações com as aplicações sejam coletados para uma posterior análise.

ANEXO IX – Avaliação da Infraestrutura UXEProject

PMCC – Programa Multiinstitucional de Pós-Graduação em Ciência da Computação

Avaliação dos processos relacionados à Geração das Métricas de Usabilidade

Usuários: Desenvolvedores de Softwares / Especialistas em IHC

Questionário Proposto aos Participantes do Experimento

Local: _____ Data: ___/___/___

Parte 1 – Dados Pessoais

- 1) Nome: _____
- 2) Idade: _____
- 3) Curso de Graduação: _____
- 4) Ano de Formatura: _____
- 5) Atividade profissional: _____
- 6) Empresa onde Trabalha: _____

Parte 2 - Experiência

- 1) Há quanto tempo você programa?

<input type="checkbox"/>	Menos de 6 meses
<input type="checkbox"/>	Entre 6 meses e 1 ano
<input type="checkbox"/>	Entre 1 ano e 2 anos
<input type="checkbox"/>	Entre 2 anos e 3 anos
<input type="checkbox"/>	Mais que 3 anos

- 2) Em que nível de produtividade você se enquadra como desenvolvedor de software?

<input type="checkbox"/>	Muito Baixo
<input type="checkbox"/>	Baixo
<input type="checkbox"/>	Médio
<input type="checkbox"/>	Alto
<input type="checkbox"/>	Muito Alto

- 3) Você tem experiência com alguma das linguagens de programação abaixo?

<input type="checkbox"/>	Java	
<input type="checkbox"/>	C#	
<input type="checkbox"/>	Python	
<input type="checkbox"/>	Object Pascal	
<input type="checkbox"/>	Objective-C	
<input type="checkbox"/>	Outra linguagem Orientada a Objetos	

4) Qual a sua experiência com Programação Orientada a Aspectos?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

5) Qual o seu nível de preocupação com a usabilidade dos softwares que já desenvolveu?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

6) Em que nível você classificaria a preocupação com a usabilidade de softwares nas equipes de desenvolvimento que participa ou participou?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

7) Em que nível você classificaria a preocupação com a usabilidade de softwares na empresa que trabalha ou trabalhou?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

8) Qual a sua experiência em seguir a descrição de processos na área de Engenharia de Software?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

9) Em que nível de dificuldade você classificaria a codificação de Aspectos?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 10) Em que nível de dificuldade você classificaria a inserção de Aspectos aos softwares apresentados durante o curso?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

Parte 3 - Avaliação do Processo para Mapeamento das Tarefas

- 1) Como você classificaria o processo de geração automática dos Aspectos que permitem o mapeamento das tarefas com o uso da ferramenta Mapping Aspect Generator (MAG)?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 2) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que auxilia o processo de mapeamento das tarefas com o uso da ferramenta MAG?

	Muito Insatisfeito
	Insatisfeito
	Indiferente
	Satisfeito
	Muito Satisfeito

- 3) Como você classificaria a forma de identificação das classes de uma aplicação que disponibilizam as interações com os usuários (Abordagem Tradicional).

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 4) Como você classificaria a atividade de captura automática dos métodos que serão utilizados para a instrumentação das métricas de rastreabilidade com o uso da ferramenta Automatic Task Description (ATD)?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 5) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que auxilia a instrumentação das métricas de rastreabilidade com o uso da ferramenta ATD?

	Muito Insatisfeito
	Insatisfeito
	Indiferente
	Satisfeito
	Muito Satisfeito

- 6) Como você classificaria a forma de identificação dos métodos que serão utilizados para criar a estrutura dos Aspectos (pointcuts e advices) (Abordagem Tradicional).

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 7) Em que nível de dificuldade você classificaria o conjunto de etapas do processo para realizar o mapeamento das tarefas?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 8) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que dá suporte ao conjunto de etapas do mapeamento das tarefas?

	Muito Insatisfeito
	Insatisfeito
	Indiferente
	Satisfeito
	Muito Satisfeito

- 9) Como você classificaria a forma de identificar as tarefas que necessitam ser avaliadas nas aplicações (Abordagem Tradicional)?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 10) Você tem alguma consideração que gostaria de relatar para o aprimoramento do processo de Mapeamento das Tarefas?

Parte 4 - Avaliação do Processo de Geração dos Aspectos de Rastreabilidade

- 1) Como você classificaria o processo de geração automática dos Aspectos que permitem a incorporação das métricas de rastreabilidade com o uso da Ferramenta UXE Metrics Generator?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 2) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que auxilia o processo de incorporação das métricas de rastreabilidade com o uso da Ferramenta UXE Metrics Generator?

	Muito Insatisfeito
	Insatisfeito
	Indiferente
	Satisfeito
	Muito Satisfeito

- 3) Como você classificaria a forma de codificação dos Aspectos de Rastreabilidade utilizando a Abordagem Tradicional, ou seja, escrever o código de todos os Aspectos?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 4) Você tem alguma consideração que gostaria de relatar para o aprimoramento do processo de instrumentação da aplicação com as métricas de usabilidade?

Parte 5 - Avaliação Geral dos Processos

- 1) Como você classificaria a execução de todos os processos relacionados com a execução da Abordagem Proposta?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 2) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que auxilia todos os passos para a utilização da Abordagem Proposta?

	Muito Insatisfeito
	Insatisfeito
	Indiferente
	Satisfeito
	Muito Satisfeito

- 3) Como você classificaria a execução de todos os processos para a codificação dos Aspectos usando a Abordagem Tradicional?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 4) Você tem alguma consideração que gostaria de relatar para o aprimoramento da infraestrutura proposta?

ANEXO X – Avaliação da Infraestrutura UXEProject

PMCC – Programa Multiinstitucional de Pós-Graduação em Ciência da Computação

Avaliação do Processo para Preparar o Ambiente de Avaliação

Usuários: Especialistas em Banco de Dados / Especialistas em IHC

Questionário Proposto aos Participantes do Experimento

Local: _____ Data: ___/___/___

Parte 1 – Dados Pessoais

- 1) Nome: _____
- 2) Idade: _____
- 3) Curso de Graduação: _____
- 4) Ano de Formatura: _____
- 5) Atividade profissional: _____
- 6) Empresa onde Trabalha: _____

Parte 2 - Experiência

- 1) Há quanto tempo você trabalha com Banco de Dados (BD)?

<input type="checkbox"/>	Menos de 6 meses
<input type="checkbox"/>	Entre 6 meses e 1 ano
<input type="checkbox"/>	Entre 1 ano e 2 anos
<input type="checkbox"/>	Entre 2 anos e 3 anos
<input type="checkbox"/>	Mais que 3 anos

- 2) Em que nível de produtividade você se enquadra como analista de BD?

<input type="checkbox"/>	Muito Baixo
<input type="checkbox"/>	Baixo
<input type="checkbox"/>	Médio
<input type="checkbox"/>	Alto
<input type="checkbox"/>	Muito Alto

- 3) Você tem experiência com algum dos Sistemas Gerenciadores de Banco de Dados (SGBD) abaixo?

<input type="checkbox"/>	MySQL	
<input type="checkbox"/>	SQL Server	
<input type="checkbox"/>	PostGreSQL	
<input type="checkbox"/>	Oracle	
<input type="checkbox"/>	Outro - Especifique	_____

4) Qual a sua experiência com MySQL?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

5) Qual a sua experiência com Ferramentas OLAP?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

6) Qual a sua experiência com a Ferramenta Pentaho?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

7) Qual a sua experiência com o Compute Cloud (Computação na Nuvem)?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

8) Qual o seu nível de preocupação com a usabilidade de software?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

9) Em que nível você classificaria a preocupação com a usabilidade de softwares nas equipes de desenvolvimento que participa ou participou?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

10) Em que nível você classificaria a preocupação com a usabilidade de softwares na empresa que trabalha ou trabalhou?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

11) Qual a sua experiência em seguir a descrição de processos na área de Engenharia de Software?

	Nenhuma
	Pouca
	Media
	Alta
	Muito Alta

Parte 3 - Avaliação do Processo para Realizar as Avaliações dos Dados de Usabilidade

1) Como você classificaria a execução dos procedimentos para instalar as ferramentas Data Load e ETL Maker que fazem a carga dos dados no BD e no Data Warehouse.

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

2) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que auxilia a instalação das ferramentas Data Load e ETL Maker?

	Muito Insatisfeito
	Insatisfeito
	Indiferente
	Satisfeito
	Muito Satisfeito

3) Como você classificaria a execução dos procedimentos para preparar a ferramenta OLAP para realizar as avaliações dos dados de usabilidade?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 4) Qual o seu nível de satisfação com a descrição dos passos contidos no documento que auxilia a preparar a ferramenta OLAP para realizar as avaliações dos dados de usabilidade?

	Muito Insatisfeito
	Insatisfeito
	Indiferente
	Satisfeito
	Muito Satisfeito

- 5) Como você classificaria a execução do processo de análise dos dados após todo o ambiente estar preparado?

	Muito Difícil
	Difícil
	Médio
	Fácil
	Muito Fácil

- 6) Você tem alguma consideração que gostaria de relatar para o aprimoramento da infraestrutura proposta?

ANEXO XI – Termo de Consentimento Livre e Esclarecido

Eu,; estou sendo convidado(a) a participar do experimento para avaliação dos processos relacionados a utilização da infraestrutura UXEProject, desenvolvida para dar suporte a avaliações de usabilidade de *smartphones* na área de Interação Humano-Computador.

A infraestrutura foi concebida como parte do projeto de doutoramento do aluno Artur Henrique Kronbauer, sob a supervisão do Prof. Dr. Celso Alberto Saibel Santos, vinculado ao Programa de Doutorado Multiinstitucional em Ciência da Computação. A minha participação no referido experimento será no sentido de auxiliar na identificação de pontos fortes e fracos dos processos relacionados com a infraestrutura.

Recebi esclarecimentos sobre a pesquisa e estou ciente de que minha privacidade será respeitada, ou seja, meu nome será mantido em sigilo. Eu autorizo a utilização do questionário contendo as minhas respostas preenchidas durante a realização do experimento, entendo que as informações serão utilizadas somente para os fins desta pesquisa, bem como serão divulgadas apenas em artigos e na redação da tese de doutorado do aluno Artur Henrique Kronbauer.

Estou ciente que poderei solicitar esclarecimentos quanto a quaisquer dúvidas durante a realização do experimento e terei acesso aos resultados obtidos. Tenho ciência de que poderei me recusar a responder qualquer pergunta e que posso me negar a participar do estudo ou retirar meu consentimento a qualquer momento, sem prévia justificativa.

Manifesto meu livre consentimento em participar.

Salvador, 15 de Outubro de 2012

Nome e assinatura do participante

Nome e assinatura do pesquisador