

**UNIVERSIDADE REGIONAL DE BLUMENAU**  
**CENTRO DE CIÊNCIAS EXATAS E NATURAIS**  
**CURSO DE SISTEMAS DE INFORMAÇÃO – BACHARELADO**

**APLICATIVO PARA CRIAR FORMULÁRIOS PARA COLETA  
DE DADOS EM AMBIENTE ANDROID**

**MARCOS BUSANA**

**BLUMENAU**  
**2011**

**2011/2-20**

**MARCOS BUSANA**

**APLICATIVO PARA CRIAR FORMULÁRIOS PARA COLETA  
DE DADOS EM AMBIENTE ANDROID**

Trabalho de Conclusão de Curso submetido à  
Universidade Regional de Blumenau para a  
obtenção dos créditos na disciplina Trabalho  
de Conclusão de Curso II do curso de Sistemas  
de Informação— Bacharelado.

Prof. Mauro Marcelo Mattos, Doutor - Orientador

**BLUMENAU  
2011**

**2011/2-20**

# **APLICATIVO PARA CRIAR FORMULÁRIOS PARA COLETA DE DADOS EM AMBIENTE ANDROID**

Por

**MARCOS BUSANA**

Trabalho aprovado para obtenção dos créditos na disciplina de Trabalho de Conclusão de Curso II, pela banca examinadora formada por:

Presidente: \_\_\_\_\_  
Prof. Mauro Marcelo Mattos, Doutor – Orientador, FURB

Membro: \_\_\_\_\_  
Prof. Francisco Adell Péricas, Mestre – FURB

Membro: \_\_\_\_\_  
Prof. Dalton Solano dos Reis, Mestre – FURB

Blumenau, 02 de fevereiro de 2012.

Dedico este trabalho aos meus pais que no esforço e determinação puderam me apoiar, incentivar e proporcionar a realização desta graduação.

## **AGRADECIMENTOS**

Aos meus pais, Nelson e Luzia, que sempre me apoiaram e estiveram presentes nos meus estudos e que estão muito felizes e orgulhosos com essa conquista.

Ao meu orientador, Mauro, por ter acreditado na proposta e na minha capacidade de concluir este trabalho.

Aos meus familiares e amigos, pela ajuda e o apoio.

A Edusoft Tecnologia Ltda. e seus funcionários que me apoiaram e incentivaram durante a graduação.

Ao colegiado do curso de Sistemas de Informação da FURB, pelo aprendizado adquirido durante a minha graduação.

Só tem o direito de criticar aquele que  
pretende ajudar.

Abraham Lincoln

## RESUMO

Este trabalho apresenta um aplicativo para realizar coleta de dados em ambiente Android. O aplicativo permite a criação, aplicação de formulários e visualização dados coletados. Além disso, foi disponibilizada a integração com o software LimeSurvey de tal forma que é possível importar formulários deste aplicativo e exportar os dados coletados para o mesmo. Para o desenvolvimento do aplicativo foi utilizada a linguagem PHP e XML para a persistência dos dados. O aplicativo utiliza o *scripting layer for android* e o PHP *for android*. O aplicativo foi validado em um tablet de 7" com android 2.2.3.

Palavras-chave: Android. Coleta de dados. Criador de formulário, Pesquisa de mercado.

## **ABSTRACT**

This work describes an application to perform data collection in Android environment using tablets. The application enables the creation, application forms in field and view collected data in the mobile device. Furthermore, the integration was performed with the software LimeSurvey so that you can import forms from LimeSurvey to mobile device and to export collected data to LimeSurvey. The application was developed using PHP with SL4A (scripting layer for android) and XML for data persistence. The application was validated on a tablet 7 " with android 2.2.3.

Key-words: Android. Data collection. Form Designer. Market Research.

## LISTA DE ILUSTRAÇÕES E TABELAS

Tabela 1 - Distribuição de mercado de smartphones entre 2009 e 2010.....	14
Figura 1 - Arquitetura do Android.....	16
Figura 2 - Exemplo de opções de interpretadores para o SL4A.....	18
Figura 3 - Android executando um script python.....	18
Figura 4 - Diagrama de fluxo de execução do SL4A.....	19
Figura 5 - Exemplo de uma tela e o código fonte de uma lista feita no PFA.....	21
Figura 6 - Exemplo de uma tela e o código fonte de com um campo feito no PFA.....	21
Figura 7 - Passos para realizar uma pesquisa de mercado.....	24
Quadro 1 – Exemplo de um questionário.....	25
Figura 8 - Tela para cadastro de perguntas do LimeSurvey.....	26
Figura 9 - Relatório dos resultados da pesquisa no LimeSurvey.....	27
Quadro 2 – Requisitos funcionais.....	30
Quadro 3 – Requisitos não funcionais.....	30
Figura 10 - Diagrama de caso de uso.....	31
Figura 11 - Diagrama de atividades.....	32
Figura 12 - Modelo entidade-relacionamento.....	32
Figura 13 - Trecho do código onde são montados os dados coletados.....	34
Figura 14 - Trecho do código onde é montada a tela de lista de formulários.....	35
Figura 15 - Trecho do código da importação do LimeSurvey para o aplicativo.....	35
Figura 16 - Trecho do código do formulário que envia os dados para o LimeSurvey.....	36
Figura 17 - Trecho do código da exportação dos dados do aplicativo para o LimeSurvey.....	37
Figura 18 – Arquivo de XML que grava os formulários.....	37
Figura 19 – Arquivo de XML que gravado as perguntas.....	38
Figura 20 – Arquivo de XML que grava as perguntas.....	39
Figura 21 - Tela de formulários cadastrados.....	40
Figura 22 - Tela de perguntas cadastradas.....	40
Figura 23 - Tela de cadastro de perguntas.....	41
Figura 24 - Lista de tipos de perguntas.....	41
Figura 25 - Tela de perguntas cadastradas.....	42
Figura 26 - Tela de aplicar formulário.....	43
Figura 27 - Tela de dados coletados.....	44

Figura 32 - Tela de dados coletados LimeSurvey .....	47
Figura 33 – Primeira página do relatório de estatísticas rápidas no LimeSurvey .....	47
Figura 34 – Página com os resultados da pergunta dois no relatório do LimeSurvey .....	48
Figura 35 – Página com os resultados da pergunta cinco no relatório do LimeSurvey .....	49
Figura 36 - Tela de monitoramento do consumo de memória e processamento no android ....	50
Quadro 4 – Comparativo entre o sistema desenvolvido e os trabalhos correlatos .....	50
Figura 41 – Aplicativo executando em hardware real .....	51
Quadro 5 – Descrição do caso de uso Cadastrar leiautes .....	56
Quadro 6 – Descrição do caso de uso Exportar leiaute .....	56
Quadro 7 – Descrição do caso de uso Importar leiaute .....	57
Quadro 8 – Descrição do caso de uso Aplicar os formulários.....	57
Quadro 9 – Descrição do caso de uso Salvar dados coletados .....	58
Quadro 10 – Descrição do caso de uso Visualizar dados coletados.....	58

## LISTA DE SIGLAS

API – *Application Programming Interface*

DVM – *Dalvik Virtual Machine*

GPS – *Global Positioning System*

GUI – *Graphical User Interface*

IBGE – *Instituto Brasileiro de Geografia e Estatística*

JSON – *JavaScript Object Notation*

MVC – *Model-View-Controller*

NDK – *Native Development Kit*

PFA – *PHP for Android*

PHP – *Hypertext Preprocessor*

RPC – *Remote Procedure Call*

SDK – *Software Development Kit*

SL4A – *Scripting Layer for Android*

XML – *eXtensible Markup Language*

XMPP – *eXtensible Messaging and Presence Protocol*

## SUMÁRIO

<b>1 INTRODUÇÃO.....</b>	<b>13</b>
1.1 OBJETIVOS DO TRABALHO .....	14
1.2 ESTRUTURA DO TRABALHO .....	14
<b>2 FUNDAMENTAÇÃO TEÓRICA .....</b>	<b>15</b>
2.1 ANDROID.....	15
2.1.1 Arquitetura Android .....	16
2.2 SCRIPTING LAYER FOR ANDROID (SL4A).....	17
2.2.1 Arquitetura do SL4A .....	19
2.3 PHP FOR ANDROID (PFA).....	20
2.4 PESQUISAS DE MERCADO .....	21
2.5 TRABALHOS CORRELATOS .....	26
<b>3 DESENVOLVIMENTO .....</b>	<b>29</b>
3.1 LEVANTAMENTO DE INFORMAÇÕES .....	29
3.2 ESPECIFICAÇÃO .....	30
3.3 IMPLEMENTAÇÃO .....	33
3.3.1 Técnicas e ferramentas utilizadas .....	33
3.3.2 Operacionalidade da implementação.....	39
3.4 RESULTADOS E DISCUSSÃO .....	49
<b>4 CONCLUSÕES .....</b>	<b>52</b>
4.1 EXTENSÕES .....	52
<b>REFERÊNCIAS BIBLIOGRÁFICAS .....</b>	<b>54</b>
<b>APÊNDICE A – Detalhamento dos casos de uso.....</b>	<b>56</b>
<b>APÊNDICE B – Dicionário de dados.....</b>	<b>59</b>

## 1 INTRODUÇÃO

Malhotra (2006, p.40) afirma que, para o marketing, a ênfase é na identificação da satisfação e das necessidades do cliente. Para isso, os gerentes de marketing necessitam de informações sobre os clientes, concorrentes e outras forças do mercado para determinar quais são as necessidades e os níveis de satisfação dos clientes.

Uma pesquisa de mercado é um estudo feito para obter informações sobre os clientes e o mercado que se vai atuar, de modo a orientar as empresas nas tomadas de decisão quanto às estratégias voltadas para produtos ou serviços.

As empresas podem realizar uma pesquisa de mercado a partir de um formulário de pesquisa, que pode ter várias questões sobre as necessidades ou níveis de satisfação do cliente. Além disso, os formulários podem ser aplicados de várias formas, em entrevistas pessoais, correspondências, telefone ou e-mail.

As pesquisas não são só importantes para as empresas. Por exemplo, o Instituto Brasileiro de Geografia e Estatística (IBGE), a cada intervalo de tempo realiza o censo demográfico. Esta é uma das maiores pesquisas para coleta de dados realizada no país. O último censo demográfico foi realizado em 2010, onde foram visitados 5.565 municípios que abrigam aproximadamente 58 milhões de domicílios.

No censo de 2000 foram impressos 79.670.000 de exemplares para a coleta dos dados (IBGE, 2003). Já para o censo 2010 foram comprados 220 mil computadores de mão equipados com receptores de *Global Positioning System* (GPS) para realizar a mesma operação de coleta de dados, “trazendo ganho de qualidade ao reduzir as possibilidades de erros na realização das entrevistas, pré-críticas automáticas e transmissão aos computadores locais” (IBGE, 2010).

Neste contexto, este trabalho visa criar um aplicativo para que se possam realizar operações de coleta de dados usando dispositivos móveis, tais como os celulares e tablets.

Existem vários cenários onde uma pesquisa pode ser aplicada, como: pesquisa eleitoral, pesquisa de satisfação, pesquisa de clima organizacional, pesquisa de marketing, pesquisa de opinião, entre outros.

Em geral, as empresas utilizam planilhas eletrônicas para tabular os dados coletados e emitir relatórios. Analisando o tema, percebeu-se que os custos de soluções comerciais são bastante elevados, o que dificulta o processo de adoção por pequenas empresas.

Hoje os celulares são o bem de consumo mais utilizado no mundo, sendo que em 2007 existiam aproximadamente 3,3 milhões de aparelhos e há uma expectativa de que em 2013 chegue a 5,6 milhões (IBGE, 2007).

Assim, a criação de um aplicativo que auxilie pesquisadores na coleta de dados por meio de formulários eletrônicos poderá agilizar a análise dos resultados. Dependendo da área de negócio onde esta pesquisa é aplicada, a agilidade nos resultados pode ser um fator determinante para o sucesso da mesma, tendo em vista a velocidade que os eventos acontecem nos dias atuais.

## 1.1 OBJETIVOS DO TRABALHO

O objetivo deste trabalho é desenvolver um aplicativo para coleta de dados de pesquisa voltado para dispositivos móveis.

Os objetivos específicos do trabalho proposto são:

- a) permitir ao usuário definir questionários para a coleta de dados;
- b) permitir ao usuário aplicar os questionários criados;
- c) armazenar os dados coletados em um dispositivo móvel;
- d) permitir ao usuário visualizar a totalização dos dados coletados no dispositivo móvel;
- e) permitir a importação de questionários do LimeSurvey;
- f) permitir a exportação de dados coletados para o LimeSurvey.

## 1.2 ESTRUTURA DO TRABALHO

Este trabalho está disposto em quatro capítulos. No primeiro capítulo apresenta-se a introdução, os objetivos e a estrutura do trabalho. No segundo capítulo tem-se a fundamentação teórica, descrevendo a plataforma Android, *Scripting Layer for Android*, PHP *for android* e pesquisa de mercado. No terceiro capítulo é apresentado o desenvolvimento do aplicativo, incluindo detalhes sobre a especificação, implementação e tecnologia utilizada.

No quarto capítulo apresentam-se a conclusão e sugestões para trabalhos futuros.

## 2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo introduz os principais conceitos relacionados ao escopo do trabalho além de apresentar o sistema atual e os trabalhos correlatos. São eles a Plataforma Android, o *scripting layer for android*, o *PHP for android* e a coleta de dados.

### 2.1 ANDROID

Android segundo Google (2011a), é uma pilha de softwares para dispositivos móveis que inclui um sistema operacional, um *middleware* e um conjunto de aplicações chaves. Os desenvolvedores podem criar aplicações para a plataforma usando o *Software Development Kit* (SDK). As aplicações para essa plataforma são escritas usando a linguagem de programação Java e executam sobre o *Dalvik Virtual Machine* (DVM), uma máquina virtual própria para dispositivos móveis com restrições de recursos, como pouca capacidade computacional, baixa capacidade de armazenamento e baterias de curta duração.

Segundo a Canalys (2011), o Android no quarto semestre de 2010 se tornou a principal plataforma para os *smartphones*, atingindo 32,9 milhões de embarques em 2010, conforme indicado na Tabela 1<sup>1</sup>.

Tabela 1 - Distribuição de mercado de smartphones entre 2009 e 2010

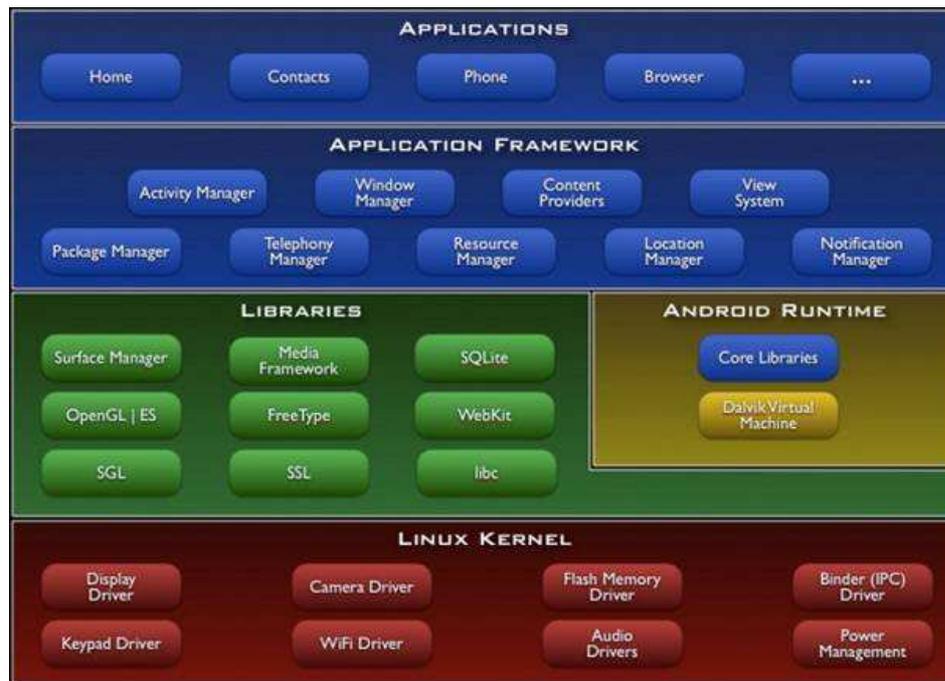
Mercado de <i>smartphones</i> em todo o mundo					
Números do mercado no 4º semestre de 2010 e 4º semestre de 2009					
OS vendidos	4 sem.2010 embarques (milhões)	Participação (%)	4 sem. 2009 embarques (milhões)	Participação (%)	Crescimento 2010/2009
Google*	33,3	32,90%	4,7	8,70%	615,10%
Nokia	31	30,60%	23,9	44,40%	30%
Apple	16,2	16,00%	8,7	16,30%	85,90%
Rim	14,6	14,40%	10,7	20,00%	36%
Microsoft	3,1	3,10%	3,9	7,20%	-20,30%
Outros	3	2,90%	1,8	3,40%	64,80%
Total	101,2	100,00%	53,7	100,00%	88,60%

Fonte: traduzido de Canalys estimates, © Canalys (2011).

<sup>1</sup> Nota: os números do Google nesta tabela se relacionam com o Android, bem como as variantes da plataforma da OMS e Tapas.

### 2.1.1 Arquitetura Android

A arquitetura do sistema operacional Android, segundo Google (2011b), é formada por quatro camadas: núcleo do sistema, bibliotecas, *framework* de aplicação e as aplicações conforme a Figura 1.



Fonte: Aquino (2007, p. 5)

Figura 1 - Arquitetura do Android

A seguir são detalhados os principais componentes desta plataforma conforme Aquino(2007, p. 5):

A primeira camada é formada pelo núcleo do sistema (*kernel*), que funciona como uma camada de abstração entre o hardware e o restante de softwares da plataforma. Utilizando Linux, é responsável pelo gerenciamento de memória, processos, *threads*, pilha de protocolos de rede, módulo de segurança e vários módulos do núcleo de infraestrutura.

A segunda camada contém um conjunto de bibliotecas C/C++ usada por vários componentes do sistema e a *Dalvik Virtual Machine*. A DVM foi escrita de forma que um dispositivo possa executar múltiplas máquinas virtuais concorrentemente de maneira eficiente. Ela usa o *kernel* do Linux para prover a funcionalidade de múltiplas *threads* e gerenciamento de memória de baixo nível. Cada aplicação do Android roda em um processo separado, com sua própria máquina virtual, número do processo e dono do processo. Isso

garante que caso a aplicação apresente erros, ela possa ser isolada e removida da memória sem comprometer o resto do sistema. Ao desenvolver as aplicações em Java, a DVM compila o *bytecode* (.class) e converte para o formato *Dalvik Executable* (.dex), que representa a aplicação do Android compilada. Depois disso, os arquivos .dex e outros recursos do projeto são compactados em um único arquivo com a extensão *Android Package File* (.apk), que representa a aplicação final.

A terceira camada é composta por *frameworks* de aplicações. Os desenvolvedores têm acesso completo à mesma interface de programação de aplicativos que é usada pelas aplicações essenciais da plataforma. Esta camada fornece um conjunto de serviços tais como *Activity Manager*, que gerencia o ciclo de vida das aplicações; *Package Manager*, que mantém quais aplicações estão instaladas no dispositivo; *Windows Manager*, que gerencia as janelas das aplicações; *Telephony Manager*, que são os componentes para acesso aos recursos de telefonia; *Content Providers*, que permitem que as aplicações acessem dados de outras aplicações ou compartilhem os seus próprios dados; *Resource Manager*, que fornece acesso aos recursos gráficos e arquivos de layout; *View System*, que é um conjunto rico e extensível de componentes de interface do usuário; *Location Manager*, que gerencia a localização do dispositivo; *Notification Manager*, que permite que todas as aplicações exibam alertas na barra de status e *XMPP Service* que é o suporte para o uso do protocolo *eXtensible Messaging and Presence Protocol* (XMPP).

A quarta camada é formada pelos aplicativos propriamente ditos e contempla aplicações como cliente de e-mail, programa de envio de mensagens, calendários, mapas, navegador web, contatos, entre outros.

## 2.2 SCRIPTING LAYER FOR ANDROID (SL4A)

O *Scripting Layer for Android* (SL4A) introduz o suporte a linguagens de *script* para o Android, permitindo que sejam criados e interpretados *scripts* interativos diretamente no aparelho com Android (Figura 2). Esses *scripts* têm acesso a muitas *Application Programming Interface* (APIs), disponíveis para aplicativos completos do Android, mas com uma interface muito mais simples, o que facilita o trabalho dos desenvolvedores. Os *scripts* podem ser executados de modo interativo em um terminal ou em segundo plano (GOOGLE, 2011c).



Fonte: Jordan, Greyling (2011, p. 202)

Figura 2 - Exemplo de opções de interpretadores para o SL4A

A Figura 3 apresenta um exemplo de script python sendo executado em um dispositivo Android através do SL4A.

Para instalar o SL4A, basta acessar através do dispositivo móvel o *site* do *php for android*, procurar pelo título “Install SL4A” e realizar o *download* do instalador. Após esta etapa, basta clicar na opção instalar e o processo será executado automaticamente.

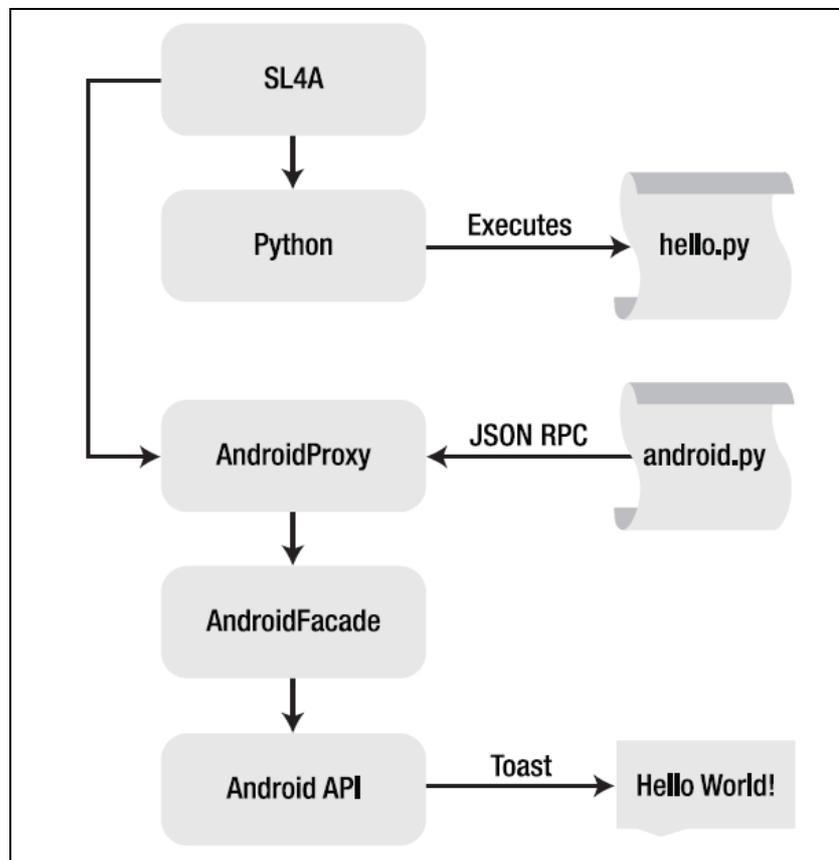


Fonte: Jordan, Greyling (2011, p. 207)

Figura 3 - Android executando um script python

### 2.2.1 Arquitetura do SL4A

Conforme Ferrill (2011), em seu nível mais baixo, SL4A é essencialmente uma série de scripts, o que significa que como um aplicativo que hospeda intérpretes diferentes com processos de cada um dos quais uma linguagem específica. Se acessar o código fonte SL4A no repositório de código vê-se é uma cópia da árvore de origem de cada linguagem, tornando possível o cross-compilados para a arquitetura ARM usando o *Android Native Development Kit* (NDK) e carregando uma biblioteca do SL4A para um interpretador específico. Nesse ponto, o script será interpretado linha por linha.



Fonte: Ferrill (2011, p. 5).

Figura 4 - Diagrama de fluxo de execução do SL4A

A arquitetura básica de SL4A é semelhante ao que se vê em um ambiente na computação distribuída. A Figura 4 mostra o fluxo de execução quando se inicia o SL4A e depois executa um script (neste caso, `hello.py`). Cada script SL4A deve importar o código fonte de um arquivo externo, como `android.py` para Python, que vai definir uma série de funções de *proxy* necessário para se comunicar com o Android API.

A comunicação real entre SL4A e o sistema operacional Android utiliza um

mecanismo de *Remote Procedure Call* (RPC) e JavaScript *Object Notation* (JSON). Normalmente se encontra o RPC sendo usado em uma arquitetura distribuída em que a informação é passada entre um cliente e um servidor. No caso de SL4A, o servidor é o sistema operacional Android, e o cliente é um script SL4A. Isso adiciona uma camada de separação entre SL4A e o sistema operacional Android para evitar qualquer script malicioso de fazer algo nocivo.

### 2.3 PHP FOR ANDROID (PFA)

De acordo com a Irontec (2010), o PHP for Android (PFA) é uma extensão, que foi criada para que dispositivos móveis com o sistema operacional Android executem aplicações desenvolvidas na linguagem PHP. Desta forma, as aplicações PHP têm acesso às funções do dispositivo, como *Bluetooth* e objetos GUI. O SL4A também oferece um ambiente de tempo de execução e um editor, permitindo que o PHP e outros *scripts* possam ser editados no próprio aparelho. Estes *scripts* têm acesso a muitas das APIs disponíveis para os aplicativos do Android, apresentando uma interface extremamente simplificada (PAULO, 2010).

Os requisitos mínimos para utilizar o PFA são o suporte a Android OS 1.5 ou superior, estar habilitada a opção de “*fontes desconhecidas*”(que estará nas configurações no menu aplicativos), estarem instalados o SL4A e o PFA.

Para instalar o PFA, basta acessar o *site do php for android* pelo celular, procurar pelo título “*Install PhpForAndroid.apk*” e realizar o *download* do instalador. Após esta etapa, basta clicar na opção instalar e o processo será executado automaticamente. Depois de instalado basta clicar em *open* e depois em *install* para finalizar o processo. As Figuras 5 e 6 apresentam exemplos de aplicações que acompanham na instalação do PFA.



Fonte: Irontec (2010).

Figura 5 - Exemplo de uma tela e o código fonte de uma lista feita no PFA



Fonte: Irontec (2010).

Figura 6 - Exemplo de uma tela e o código fonte de com um campo feito no PFA

## 2.4 PESQUISAS DE MERCADO

Silva (2005) diz que, a pesquisa de mercado é um dos principais pontos de partida para qualquer planejamento de *marketing*. Sem ela, a organização entra no mercado às cegas. É a

referência para as tomadas de decisões de planejamento e a base das ações de execução das atividades.

A pesquisa possibilita à organização reconhecer os consumidores em qualquer mercado. Isso é possível graças ao levantamento que se faz das características pessoais como sexo, idade, nível de renda, etc., além das percepções, preferências e necessidade diferentes dos consumidores. Essas informações são fundamentais para o processo de *marketing*.

De acordo com o Honorato (2004, p.85), todas as áreas de uma empresa podem se beneficiar das informações obtidas da pesquisa de *marketing*. As empresas realizam pesquisas para identificar problemas, potenciais e participações de mercado, tendências, previsões e também para buscar a solução de problemas, procurando facilitar o processo de avaliação de ameaças e oportunidades requeridas no planejamento estratégico.

O processo de pesquisa de *marketing* envolve várias fases. A condução desse processo depende da formulação do problema, ou seja, do tipo de informação esperado para o auxílio na tomada de decisão. O *design* de uma pesquisa pode apresentar algumas variações, embora todos apresentem fundamentalmente o mesmo conteúdo, os quais podem ser agrupados em situação-problema, objetivos, teoria e método.

A maioria das pesquisas de marketing relaciona-se com a identificação e a definição de oportunidades de novos negócios. Quando se identifica uma oportunidade, é fundamental conduzir uma pesquisa para poder estabelecer estratégias de maneira adequada (HONORATO, 2004, p.92).

Ao formular um problema de pesquisa, o profissional de marketing precisa se aprofundar nas investigações inerentes à questão, procurando estabelecer às ações necessárias a situação, não se limitando simplesmente a sintomas da existência de um problema.

Os objetivos são decorrentes da formulação do problema. Assim, quando o problema é claro, os objetivos também o são, tornando o processo de pesquisa mais rápido e preciso (HONORATO, 2004, p.93).

A concepção da pesquisa consiste no detalhamento dos procedimentos necessários para obtenção das informações desejadas, tem como finalidade determinar as possíveis respostas às questões de pesquisa, proporcionando as informações necessárias para a tomada de decisões.

As pesquisas podem ser classificadas de maneira ampla, como exploratória e conclusiva. A pesquisa exploratória é a pesquisa que tem como principal objetivo descobrir idéias, percepções, gerar hipóteses mais precisas para um estudo mais aprofundado. Já a pesquisa conclusiva é a pesquisa concebida para auxiliar o profissional de marketing a

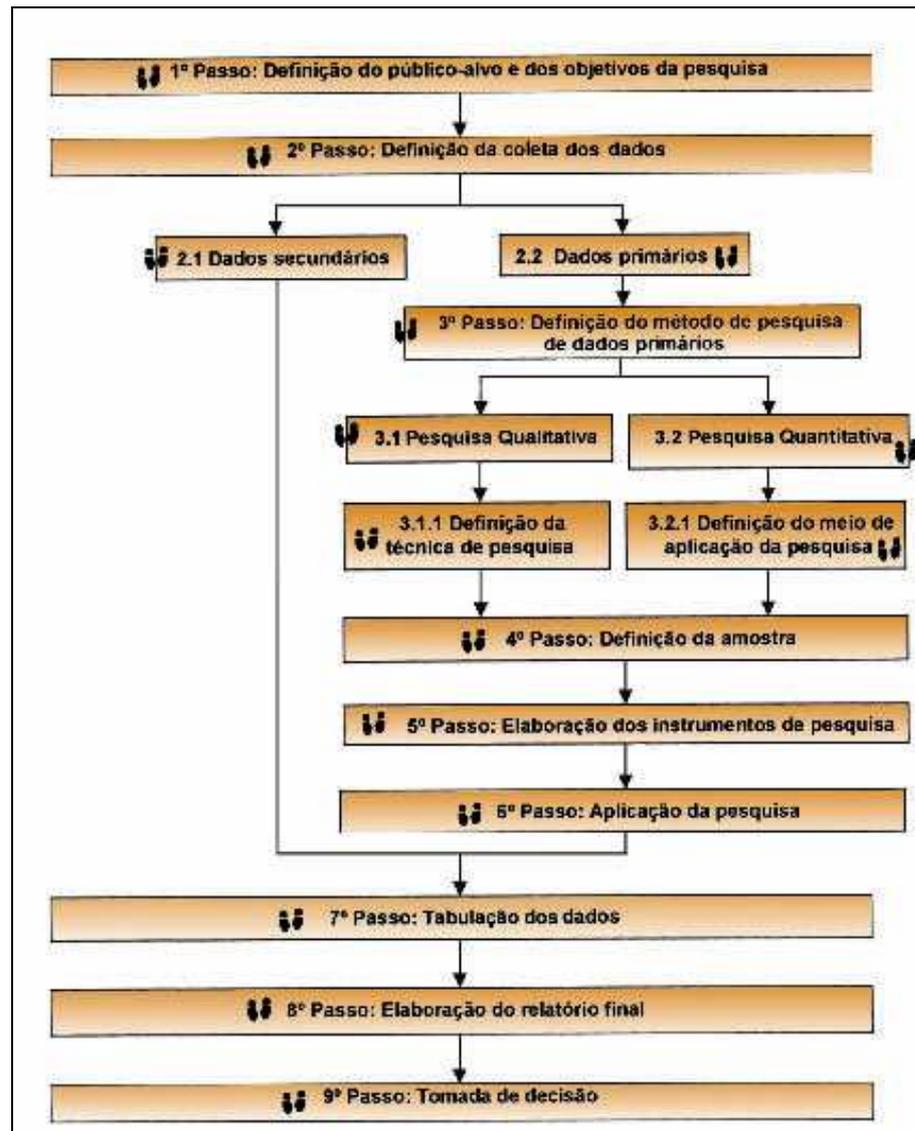
determinar, avaliar e selecionar as melhores informações para a tomada de decisão (HONORATO, 2004, p.96).

Um dos componentes de planejamento de pesquisa é o plano amostral, que consiste em elaborar os processos adequados de amostragem. A maioria das pesquisas tem por objetivo obter informação sobre características de uma população. As características ou parâmetros da população são expressos em números que podem ser obtidos realizando um censo ou extraíndo uma amostra (HONORATO, 2004, p.101).

Os questionários são uma das principais causas de erros em pesquisas de marketing. Um questionário mal elaborado pode invalidar um plano amostral bem concebido. Problemas na elaboração de um questionário geralmente decorrem de: falta de objetividade e clareza, falha na seqüência das perguntas ou na divisão por tópicos, perguntas mal elaboradas ou mal colocadas. Por causa de inexistência de princípios científicos para a concepção de um questionário, não há critério específico para sua elaboração. Um questionário ideal é o resultado de experiência (HONORATO, 2004, p.105).

De acordo com Gomes (2005), a estrutura de uma pesquisa de mercado se divide em nove passos que estão listados abaixo e ilustrados na Figura 7:

- a) definição do público-alvo e objetivos da pesquisa: qual a real necessidade e os motivos que o levam a realizar a pesquisa;
- b) definição da coleta dos dados: nessa etapa se define como serão levantados os dados da pesquisa;
- c) definição do método de pesquisa de dados primários: qual o tipo de pesquisa mais adequado para o propósito escolhido? Definir os procedimentos para a realização da pesquisa;
- d) definição da amostra: conhecimento do tamanho do público a ser pesquisado;
- e) elaboração dos instrumentos de pesquisa: nessa etapa se elabora os questionários, formulários e roteiros de pesquisas;
- f) aplicação da pesquisa: a forma como será abordada garantirá o bom desempenho da mesma;
- g) tabulação dos dados: reunir os dados e processá-los de maneira a facilitar a análise;
- h) elaboração do relatório final: a elaboração do relatório final permitirá analisar os dados obtidos na pesquisa e identificar aspectos relevantes para a tomada de decisão;
- i) tomada de decisão: com os resultados obtidos a partir da pesquisa será possível determinar as ações necessárias em benefício do negócio.



Fonte: Gomes (2005).

Figura 7 - Passos para realizar uma pesquisa de mercado

O Quadro 1 apresenta um exemplo de como devem ser formuladas as perguntas com uma ordem lógica. O questionário que é apresentado é o da Sociedade Mericon de Nápoles. Trata-se de uma pesquisa para conhecer a opinião dos médicos sobre a propaganda de produtos farmacêuticos.

PESQUISA SOBRE PROPAGANDA MÉDICA MERICOM	
CONSULTAS EMPRESARIAIS – ANÁLISE MERCADOLÓGICA – SONDAgens SOBRE A OPINIÃO PÚBLICA RUA RAFFAELE DE CESARA 7 – NÁPOLES TELEFONE 63730	
1. QUAL A SUA OPINIÃO SOBRE A PROPAGANDA DIRETAMENTE AO PÚBLICO DE PRODUTOS FARMACEUTICOS NÃO SUBMETIDOS À PRESCRIÇÃO MÉDICA?	
- Não altera absolutamente a opinião do médico em relação à casa	1
- É possível pra casos muito conhecidos	2
- As boas farmácias deveriam evitá-lo	3
- O médico aceita se a propaganda é séria e reservada	4
- Dispõem desfavoravelmente o médico para a prescrição de especialidade daquela casa	5
- Outras respostas (especificar).....	6
2. OUVES-SE DE ALGUNS QUE O MÉDICO EVITA PRESCREVER ESPECIALIDADE DE CASAS QUE DIFUNDEM O PRÓPRIO NOME ENTRE O PÚBLICO PARA VENDA DE PRODUTOS NÃO SUJEITOS À PRESCRIÇÃO MÉDICA. QUAL A SUA OPINIÃO SOBRE A RESPEITO?	
- Acho que é verdade	1
- Não é verdade	2
- As vezes acontece	3
- Nenhuma opinião	4
- Outras respostas (especificar).....	5
3. SE UMA FARMACIA QUE GAZA DE SUA CONFIANÇA E DA QUAL PRESCREVE ALGUMAS ESPECIALIDADES LANÇASSE UM PRODUTO NÃO SUBMETIDO A PRESCRIÇÃO MÉDICA FAZENDO DIRETAMENTE A PROPAGANDA SOBRE O PÚBLICO. SEM CONTUDO USAR VISTOSAS FORMAS PUBLICITÁRIAS, QUAL SERIA SEU COMPORTAMENTO EM RELAÇÃO AQUELA CASA?	
- Continuarei a prescrever as especialidades	1
- Evitarei a prescrição daquela especialidade	2
- Prescreverei ainda mais seus produtos	3
- Outras respostas (especificar).....	4
4. ACREDITA QUE AO PACIENTE AGRADA A PRESCRIÇÃO DE ESPECIALIDADES DE FARMACIAS MUITO CONHECIDAS PELO PÚBLICO?	
- Sim	1
- Não	2
- Outras respostas (especificar).....	3
Dados relativos à pessoa entrevistada:	
Nome e Sobrenome.....	
Especialização.....	
Idade.....	
de 25 a 35	1
de 36 a 45	2
de 46 a 60	3
acima de 60	4

Fonte: Tagaliacarne (1978).

Quadro 1 – Exemplo de um questionário

## 2.5 TRABALHOS CORRELATOS

Chequeto (2007) desenvolveu o Magregister 1.0, um sistema gerador de interfaces para dispositivos móveis para coleta de dados. O sistema teve como principal objetivo desenvolver um aplicativo *desktop* para modelagem de interfaces, um gerador de código JME e a disponibilização dos códigos gerados para serem executados em dispositivos móveis

O LimeSurvey é um software para aplicações de questionários on-line escrito em PHP, que pode utilizar banco de dados MySQL, PostgreSQL ou Microsoft SQL Server. O sistema permite aos usuários publicar e coletar respostas de questionários. As pesquisas podem incluir ramificações, *layout* e projeto personalizados. O sistema oferece análise estatística com base nos resultados do questionário. As pesquisas podem ser acessíveis ao público ou acesso controlado, através do uso de chaves para cada participante do questionário (MCGRATH, 2007).

Nas Figuras 8 e 9 podem ser visualizadas algumas telas do LimeSurvey. Essas telas são as de cadastro de perguntas e o relatório com os resultados obtidos em uma pesquisa.

The screenshot displays the LimeSurvey administration interface in a browser window. The page title is 'LimeSurvey Demo'. The browser address bar shows the URL: <http://demo.limesurvey.org/admin/admin.php?action=addquestion&sid=57282&gid=1925>. The interface is organized into several sections:

- Administration -- Logged in as: admin**: Contains various utility icons and a 'Surveys' dropdown menu set to 'new temporary survey'.
- Survey new temporary survey (ID:57282)**: Contains icons for survey management and a 'Groups' dropdown menu set to 'group 1'.
- Group group 1 (ID:1925)**: Contains icons for group management and a 'Questions' dropdown menu set to 'Please Choose...'.

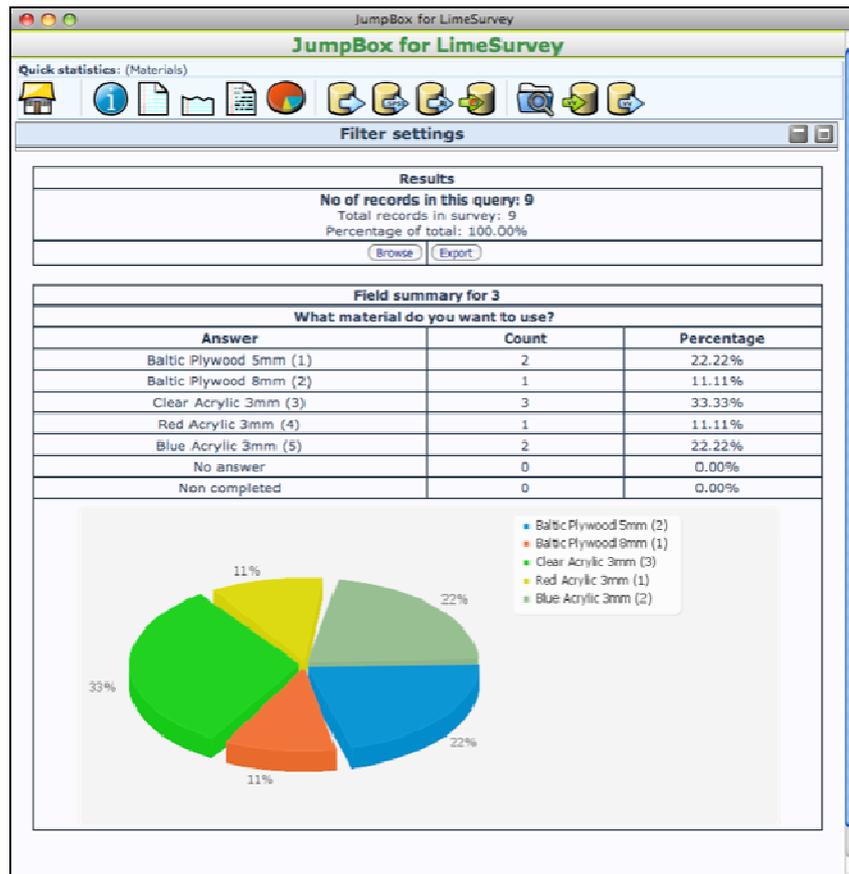
The main content area is titled 'Add Question' and contains the following form fields:

- Code:** 2 (Required)
- Question:** How do you feel about this question?
- Help:** (Empty text area)
- Type:** Long free text
- Validation:** (Empty text area)
- Mandatory:** Yes  No
- Question Attributes:** display\_rows: (Empty text area)

Below the form, there are two options: 'Add Question' and 'Import Question'. The 'Import Question' section includes a 'Select CSV File:' field with a 'Choose File' button and a note 'no file selected', followed by an 'Import Question' button. The footer of the page shows 'LimeSurvey Version 1.53' and a 'Like it? Donate to LimeSurvey' link.

Fonte: Limesurvey (2007).

Figura 8 - Tela para cadastro de perguntas do LimeSurvey



Fonte: Limesurvey (2007).

Figura 9 - Relatório dos resultados da pesquisa no LimeSurvey

O LimeSurvey 2.0 atualmente esta sendo desenvolvido com base no padrão *Model-View-Controller* (MVC) e utilizando o framework CakePHP (LIMESURVEY, 2007).

As principais características do software são:

- número ilimitado de questionários ao mesmo tempo;
- número ilimitado de perguntas em um questionário (limitado somente pela sua base de dados);
- número ilimitado de participantes de um questionário;
- questionários com multi-linguagens;
- gerenciamento de usuários;
- 20 diferentes tipos de questões;
- integração com imagens em um questionário;
- criação de versão impressa de questionários;
- condições para as perguntas de acordo com respostas anteriores;
- conjuntos de resposta reutilizáveis;
- importar perguntas prontas;

- l) pesquisas de avaliação;
- m) pesquisas anônimas e não anônimas;
- n) pesquisas abertas e fechadas para grupo de participantes;
- o) registro público opcionais para pesquisas;
- p) opção para os participantes salvarem as respostas e continuar o questionário mais tarde;
- q) editor de templates para criar seu próprio *layout* de página;
- r) questionários com data de expiração automatizada;
- s) realiza funções de importação e exportação para texto, CVS, PDF, SPSS, R, QueXML e formato MS EXCEL;
- t) análise de estatística e gráfica básica, com recursos de exportação;
- u) suporta mais de 50 idiomas diferentes no *frontend* e *backend*.

A QuestManager é uma empresa de tecnologia especializada em soluções profissionais para pesquisa de mercado e *feedback management*. O software permite o acompanhamento dos resultados em tempo real. A análise dos números pode ser feita antes do fechamento do projeto. Outra vantagem do sistema é permitir a personalização completa dos questionários. É possível criar layouts exclusivos, com a identidade visual da empresa, e utilizar recursos multimídia (imagens, áudio, animação e vídeo) para ilustrar todo o instrumento de coleta. O QuestManager pode ser aplicado em diversos tipos de pesquisa tais como: pesquisa de mercado, pesquisa eleitoral, pesquisa de satisfação de clientes, pesquisa de opinião, pesquisa de mídia, auditoria, aplicação *conjoint*, pesquisa *ad-hoc* (QUESTMANAGER, 2002).

### 3 DESENVOLVIMENTO

Neste capítulo são descritas as informações levantadas, detalhes da especificação, os diagramas de casos de uso e entidade relacionamento, a operacionalidade do sistema e ao final, os resultados e discussão.

#### 3.1 LEVANTAMENTO DE INFORMAÇÕES

A criação de um aplicativo que auxilie pesquisadores na coleta de dados por meio de formulários eletrônicos, pode contribuir para uma maior agilidade na análise dos resultados. Dependendo da área de negócio onde a pesquisa de mercado é aplicada, a agilidade nos resultados pode ser um fator determinante para o sucesso da mesma, tendo em vista a velocidade que os eventos acontecem nos dias atuais.

Assim se propôs criar um aplicativo para dispositivos móveis com algumas funcionalidades. As funcionalidades que devem ser suportadas pelo aplicativo são as seguintes:

- a) criar formulários com múltiplas perguntas com os seguintes tipos de respostas: texto, escolha simples e múltipla escolha;
- b) permitir que o usuário aplique os formulários criados no próprio dispositivo;
- c) os dados coletados pelo formulário devem ser salvos no dispositivo;
- d) os dados deverão estar disponíveis através de um relatório para o usuário visualizá-los no dispositivo;
- e) permitir a integração com o LimeSurvey, através da importação dos formulários e a exportação dos dados coletados. A integração será realizado no seguinte sentido os formulários criados no LimeSurvey serão importados no aplicativo. E os dados coletados no aplicativo através dos formulários importados serão exportados para o LimeSurvey.

### 3.2 ESPECIFICAÇÃO

Esta seção descreve os requisitos funcionais (RF) e não funcionais (RNF), bem como os diagramas de casos de uso e diagrama de atividades desenvolvidos para o sistema. A ferramenta Enterprise Architect (EA), em sua versão 7.5.850, foi utilizada na elaboração do diagrama de atividades.

O Quadro 2 apresenta os requisitos funcionais previstos para o sistema e sua rastreabilidade, ou seja, vinculação com o(s) caso(s) de uso associado(s).

<b>Requisitos Funcionais</b>	<b>Caso de Uso</b>
RF01: O aplicativo deve permitir ao usuário o cadastramento de leiautes. Esses leiautes deverão ser compostos por tabelas, com seus respectivos campos, tipos, formatos e relacionamentos.	UC01
RF02: O aplicativo deve importar os leiautes do LimeSurvey em arquivos XML.	UC02
RF03: O aplicativo deve exportar os dados coletado para o LimeSurvey.	UC03
RF04: O aplicativo deve disponibilizar os leiautes em forma de formulário.	UC04
RF05: O aplicativo deve salvar os dados coletados através dos formulários em um arquivo XML.	UC05
RF06: O aplicativo deve disponibilizar os dados coletados.	UC06

Quadro 2 – Requisitos funcionais

O Quadro 3 lista os requisitos não funcionais previstos para o sistema.

<b>Requisitos Não Funcionais</b>
RNF01: Ser desenvolvido PHP com o <i>framework</i> PFA.
RNF02: Utilizar XML.
RNF03: O aplicativo deverá ser executado no sistema operacional Android 1.5 ou superior.

Quadro 3 – Requisitos não funcionais

Na Figura 10, tem-se o diagrama de casos de uso do aplicativo desenvolvido, sendo que o detalhamento dos principais casos de uso está descritos no Apêndice A.

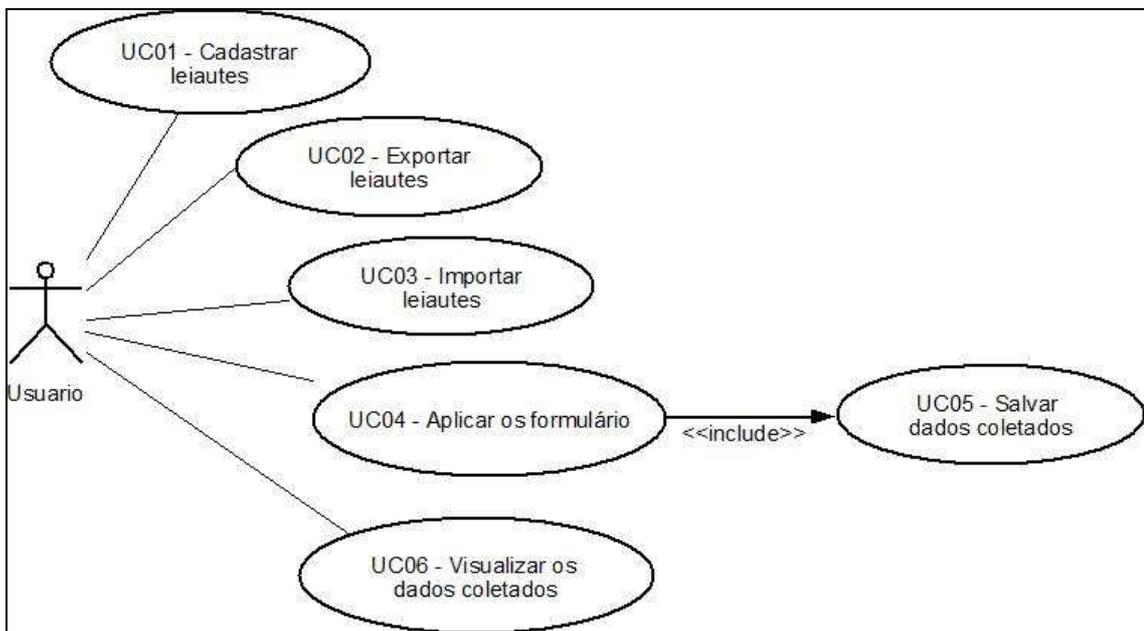


Figura 10 - Diagrama de caso de uso

Na Figura 11 tem-se o diagrama de atividades que mostra as principais transações do aplicativo.

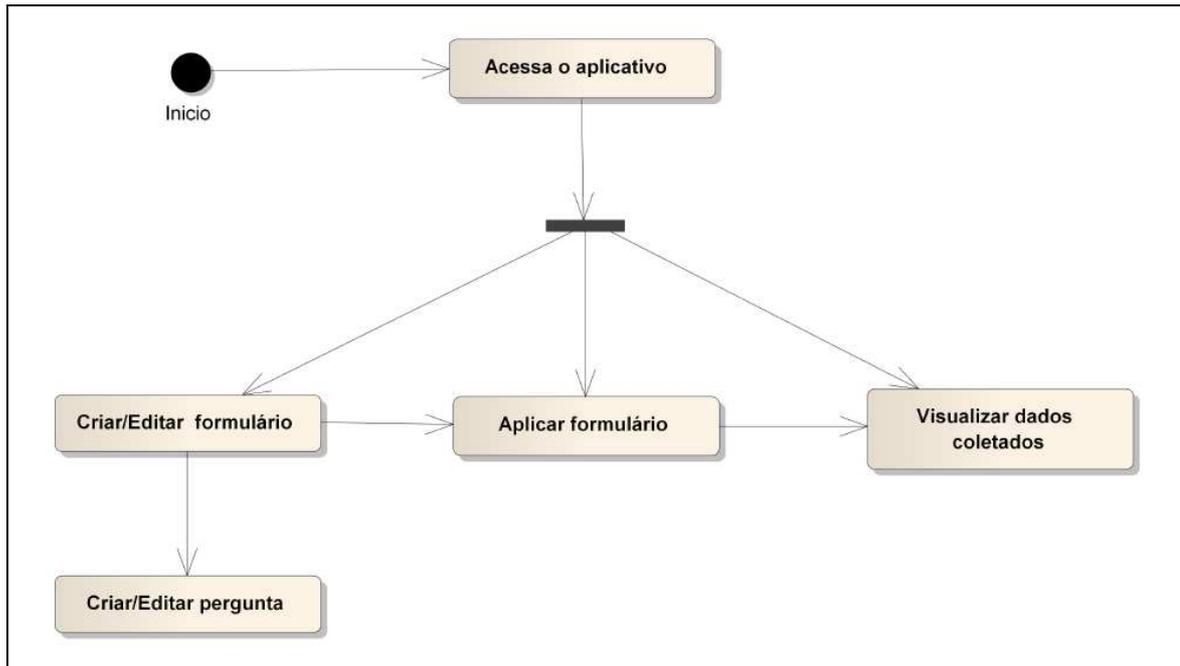


Figura 11 - Diagrama de atividades

Através da tela inicial do aplicativo, o usuário poderá escolher a ação que ele irá tomar naquele momento. Ele poderá tomar a ação de criar ou editar um formulário, em seguida o aplicativo irá redirecioná-lo para a tela de criar ou editar perguntas. O usuário também poderá optar pela ação de aplicar formulários já criados, realizando a coleta de dados e na sequência ele poderá visualizar os dados coletados com alguns totalizadores.

A Figura 19 apresenta o modelo entidade e relacionamento que representam as entidades que serão persistidas em arquivos XML.

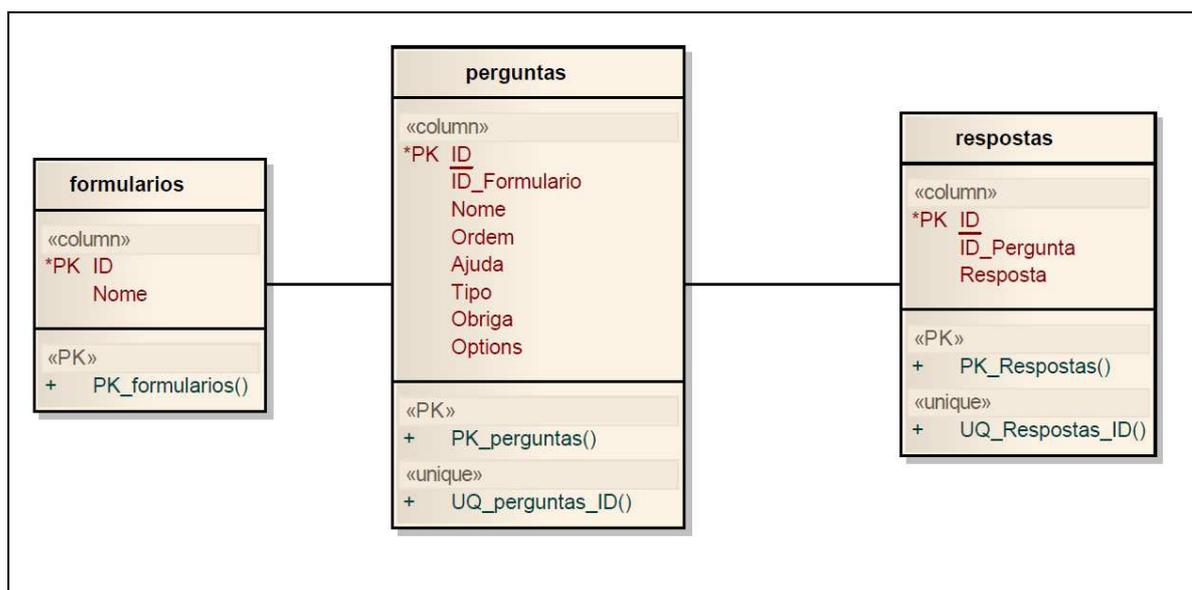


Figura 12 - Modelo entidade-relacionamento

A seguir é apresentada uma breve descrição das entidades utilizadas para o

desenvolvimento do sistema, o dicionário de dados está descrito no Apêndice B:

- a) formularios: entidade responsável por guardar informação iniciais do formulário. A entidade formulário será um único arquivo XML.
- b) perguntas: entidade responsável por guardar as informações referentes as perguntas do formulário. Para cada formulário criado existe um arquivo XML de perguntas.
- c) respostas: entidade responsável por guardar as informações referentes as respostas das perguntas. Para cada formulário criado existe um arquivo XML de respostas.

### 3.3 IMPLEMENTAÇÃO

A seguir são mostradas as técnicas e ferramentas utilizadas e a operacionalidade da implementação.

#### 3.3.1 Técnicas e ferramentas utilizadas

No desenvolvimento do sistema, foi utilizado o Android SDK 15 para emular o dispositivo móvel no computador, o SL4A r5x como interprete de scripts dentro do android, o PFA 0.3 como interpretador dos arquivos PHP dentro do SL4A, o *Hypertext Preprocessor* (PHP) 5.3.3 como a linguagem de programação, o Aptana Studio 3 como ambiente de desenvolvimento e *Extensible Markup Language* (XML) para armazenar as informações cadastradas no aplicativo. Para a manipulação dos arquivos XML foi utilizado à biblioteca externa Very Fast XML Parser, pois o PFA não possui uma biblioteca nativa para a manipulação de XML.

A Figura 13 traz um trecho do código onde são montados os dados coletados que irão resultar em uma tabela no aplicativo e logo abaixo dessa tabela irá trazer gráficos de barras para as perguntas que são do tipo lista ou múltipla escolha. Da linha 44 até a linha 62 percorre-se o *array* com todas as respostas montando-se a tabela que irá aparecer na visualização dos dados. Na linha 54 a variável *estatística* esta totalizando as respostas

para cada item, para que seja usada nas linhas 74 até a 85 para montar o gráfico de barras.

O gráfico é montado da seguinte forma: obtém-se o total de respostas para aquele item multiplicado-se por 100 e depois divide-se pelo total de questionários respondidos. Desta forma obtém-se o percentual de respostas para aquele item. Posteriormente é criado um div (elemento HTML) com o tamanho da porcentagem obtida.

```

49-     foreach ($cam as $key => $res) {
50         $chave = str_replace('id_', '', $key);
51
52-         if($camposF[$chave][0][1] == "3"){
53-             if($res == 'oth-'){
54                 $estatistica[$chave][$res] = $estatistica[$chave][$res]+1;
55-             }else{
56                 $contents .= '<td width="30%">'.$camposF[$chave][0][2][$res].'</td>';
57                 //contabiliza as respostas iguais de uma pergunta
58                 $estatistica[$chave][$res] = $estatistica[$chave][$res]+1;
59             }
60-         }else{
61-             if(str_replace('other', '', $key) == $key){
62                 $contents .= '<td width="30%">'.$res.'</td>';
63-             }else{
64                 
65             }
66         }
67         $respAnt = array($res, $camposF[$chave][0][2][$res]);
68     }
69     $contents .= '</tr>';
70 }
71
72 $contents .= '<tbody>';
73
74
75- foreach($estatistica as $key => $esta){
76     $contentsEstat .= '<br />
77     <hr />
78     <legend>'.$camposF[$key][0][0].'</legend>';
79-     foreach($esta as $e => $re){
80         //tira a porcentagem de uma questão
81         $perc = round(((($re*100)/$totalResposta), 2);
82         //monta o grafico de barras com um div colocando o tamanho dele com a porcentagem obtida
83         $contentsEstat .= '<p>'.$camposF[$key][0][2][$e].' - '.$perc.'% ('.$re.')'.'</p>
84             <div style="background: #0069AC; width:'.$perc.'% ">&nbsp;&nbsp;&nbsp;</div>';
85     }

```

Figura 13 - Trecho do código onde são montados os dados coletados

Na Figura 14 é apresentado o trecho do código onde é montada a tela de lista de formulários. Nas demais telas foi seguido o mesmo padrão de codificação que é o seguinte:

- a) primeiro faz-se a leitura do leiaute em HTML (linha 11);
- b) posteriormente leem-se os dados que estão no XML (linha 14);
- c) a seguir monta-se a tabela com os dados obtidos no XML (linhas 18 a 31);
- d) depois altera-se o conteúdo que é estático no HTML pela tabela que foi montada (linhas 32 e 34);
- e) finalmente chama-se o método que invoca o navegador para executar o HTML alterado (linha 36).

```

8 //cria objeto de para leitura e escrita de xml
9 $array2xml = new multidimensional_array2xml();
10
11 $listaHtml = file_get_contents("/sdcard/s14a/scripts/android/html/listaForm.html");
12
13 $arr = array();
14 $array = $array2xml->xml2array("/sdcard/s14a/scripts/android/xml/formularios.xml", $arr);
15 $contents = '';
16 $GLOBAL['maiorIdF'] = 0;
17
18 foreach ($array as $n => $layout) {
19     $contents .= '
20     <tr style="background:#F0F0F0">
21     <td width="5%"><input type="checkbox" value="'. $layout['id'].'" id="layout_'. $layout['id'].'" name="layout"/></td>
22     <td width="90%">'. $layout['nome']. '</td>
23     <td width="5%"><a href="#" onclick="btnAction(\`editar;'. $layout['id']. '\`)"><a href="#" onclick="btnAction(\`visualizar;'. $layout['id']. '\`)"><a href="#" onclick="btnAction(\`dadosColetados;'. $layout['id']. '\`)">';
27
28     if($GLOBAL['maiorIdF'] < $layout['id']){
29         $GLOBAL['maiorIdF'] = $layout['id'];
30     }
31 }
32 $lista = str_replace('%conteudo%', $contents, $listaHtml);
33
34 file_put_contents("/sdcard/s14a/scripts/android/html/temp.html", $lista);
35
36 $wb = $droid->webViewShow('file:///sdcard/s14a/scripts/android/html/temp.html', FALSE);
37
38 sleep('12');
39 file_put_contents("/sdcard/s14a/scripts/android/html/listaForm.html", $listaHtml);

```

Figura 14 - Trecho do código onde é montada a tela de lista de formulários

Na Figura 15 é apresentado o trecho do código onde é realizada a importação dos formulários do LimeSurvey para o aplicativo no Android. O LimeSurvey já disponibiliza uma forma de exportar os formulários em XML. Nesse trecho de código é realizada a operação de selecionar os dados do XML do LimeSurvey e criar um XML de acordo com o formato do aplicativo desenvolvido. Na linha 776 é transformado o XML do LimeSurvey em um *array*; na linha 779 é criado um *array* com o *id* do formulário e o nome; na linha 781 são copiadas todas as perguntas do formulário para a variável *questoes*; da linha 785 até a linha 859 são percorridas todas as perguntas do formulário do modelo de XML do LimeSurvey e transferidas para o modelo do aplicativo.

```

776 $arr = new xml2array($xml);
777 $arr = $arr->getResult();
778
779 $arrNew = array('id'=>$surveyid, 'nome'=>utf8_encode($arr['questionnaire']['title']['#text']));
780
781 $questoes = $arr['questionnaire']['section']['question'];
782 $questoesNew = array();
783
784 $count = 0;
785 foreach ($questoes as $key => $value) {
786     if(is_int($key)){
787         $questoesNew[$count] = array(
788             'id' => $count, 'nome' => utf8_encode($value['text']['#text']), 'ordem' => $count,
789             'ajuda' => utf8_encode($value['directive']['help']['#text']), 'obriga' => ($value['directive']['mandatory']['#text']=='Y'?1:0)
790         );
791
792         if($value['subQuestion']){
793             $questoesNew[$count]['tipo'] = 5;
794             $sub = $value['subQuestion'];
795
796             foreach ($sub as $k => $v) {
797                 if(is_array($v) && $k !== 'text'){
798                     $questoesNew[$count]['options'][$v['@varName']] = utf8_encode($v['text']['#text']);
799
800                 }elseif($k == '@varName'){
801                     $questoesNew[$count]['options'][$v] = utf8_encode($sub['text']['#text']);
802                 }
803             }
804             elseif(count($value['response']) > 2){
805
806             }
807
808             $count ++;
809         }elseif($count < 1000){
810
811         }
812     }
813     $arrNew['perguntas'] = $questoesNew;

```

Figura 15 - Trecho do código da importação do LimeSurvey para o aplicativo

Na Figura 16, é apresentado o trecho do código onde esta montado um formulário para

enviar os dados para o LimeSurvey. Nesse formulário na linha 9 na *tag form* no atributo *action* nele esta o caminho para onde o aplicativo deve mandar o dados coletado, nele tem a palavra “servidor” entre dois símbolos de porcento. Essa palavra será substituída pelo domínio onde o LimeSurvey esta instalado. Já na linha 10 tem a palavra “respostas” entre dois símbolos de porcento. Essa palavra será substituída por campos com os conteúdos dos XML das respostas do questionário. Nas linhas 13 a 16 tem um *script* que submete o formulário automaticamente para o LimeSurvey.

```

1 <html>
2 <head>
3 <title>Formulario de exportação</title>
4 </head>
5 <body>
6
7 
8 <p>Aguarde exportando os dados</p>
9 <form id="formLista" action="http://%servidor%/limesurvey/admin/integrar/importar.php" method="post" style="display: none">
10 %resposta%
11 <input type="submit" value="enviar" />
12 </form>
13 <script type="text/javascript">
14 var form = document.getElementById("formLista");
15 form.submit();
16 </script>
17 </html>

```

Figura 16 - Trecho do código do formulário que envia os dados para o LimeSurvey

Na Figura 17, é apresentado o trecho do código que recebe os dados do formulário da figura 16 e importa dos dados coletados para o LimeSurvey. Os dados são inseridos diretamente na tabela de respostas do LimeSurvey. Nesse trecho de código e realizada a operação de pegar os dados do XML que está no aplicativo e inserido os dados na tabela:

- na linha 91 e 92 são transformados os dados do XML em um *array*;
- nas linhas 94 a 98 verifica se existe o questionário cadastrado no LimeSurvey;
- nas linhas 105 a 119 é realizado os *inserts* na tabela.

```

88 foreach($post as $key => $p){
89     $sid = str_replace('respostas_', '', $key);
90
91     $sarr = new xml2array($p);
92     $sarr = $sarr->getResult();
93
94     $Query = "SELECT sid FROM ".$dbprefix.'surveys WHERE sid = ".$sid;
95     $QueryResult = db_execute_assoc($Query);
96     $Row = $QueryResult->FetchRow();
97     if($Row['sid'])
98         echo "<p>Não foi encontrado o questionario ".$sid."<p>";
99
100    $Query = "SELECT gid FROM ".$dbprefix.'groups WHERE sid = ".$sid;
101    $QueryResult = db_execute_assoc($Query);
102    $grupo = $QueryResult->FetchRow();
103
104    $sarr = $sarr['array'];
105    foreach($sarr as $k => $res){
106        $sql = "INSERT INTO ".$Row[0]."";
107        $campos = $valores = '';
108
109        foreach($res as $r => $p){
110            $sidP = explode('_', $r);
111            $campos .= ",".(count($sidP) == 2 ? $sid.'X'.$grupo['gid'].'X'.$sidP[1] : $sidP[2]);
112            $valores .= ",".$p['#text']."";
113        }
114        $Query = "INSERT INTO ".$dbprefix.'survey_'.$sid.' (startlanguage $campos)
115                VALUES ('pt-BR' $valores)";
116
117        $connect->SetFetchMode(ADODB_FETCH_ASSOC);
118        $dataset=$connect->Execute($Query,false);
119    }
120 }

```

Figura 17 - Trecho do código da exportação dos dados do aplicativo para o LimeSurvey

A Figura 18 traz um exemplo de como o XML que grava os formulários é montado. Neste XML somente é gravado o nome do formulário e um código de identificação. A *tag array* é o nó principal, dentro da *tag array* vem as *tag n* e um número sequencial. Cada *tag n* guarda um formulário com as *tag id* que é o código de identificação e a *tag nome* que é o nome do formulário.

```

<?xml version="1.0" encoding="utf-8"?>
<array>
  <n1>
    <id>1</id>
    <nome>Satisfacao c/ mobilidade urbana</nome>
  </n1>
  <n2>
    <id>2</id>
    <nome>Satisfacao c/ a Proeb</nome>
  </n2>
  <n3>
    <id>3</id>
    <nome>Iluminacao publica</nome>
  </n3>
</array>

```

Figura 18 – Arquivo de XML que grava os formulários

Na Figura 19 é apresentado o arquivo de XML onde estão os dados das perguntas do formulário de “id” 1. Para cada formulário é criado um arquivo XML para gravar os dados referentes às perguntas. Este XML tem a seguinte formatação: a *tag* *array* é o nó principal, dentro do qual existe a *tag* *id* que é o número de identificação do formulário, a *tag* *nome* com o nome do formulário e a *tag* *perguntas* que contém todas as perguntas do formulário. Dentro da *tag* *pergunta* vem as *tag* *n* e um número seqüencial. Cada *tag* *n* guarda uma pergunta com a *tag* *id* que é o número identificador da pergunta, a *tag* *nome* que contém a descrição da pergunta, a *tag* *ordem* que é o número da sequência das perguntas no formulário, a *tag* *ajuda* que contém uma descrição para um auxílio ao usuário ao responder a pergunta, a *tag* *tipo* que identifica qual é o tipo da pergunta, a *tag* *obriga* que identifica se a pergunta é obrigatória ou não e a *tag* *options* que contém as opções para perguntas de múltiplas escolhas e lista.

```

<?xml version="1.0" encoding="utf-8"?>
<array>
  <id>1</id>
  <nome>Satisfacao c/ mobilidade urbana</nome>
  <perguntas>
    <n1>
      <id>1</id>
      <nome>Nome</nome>
      <ordem>1</ordem>
      <ajuda>completo</ajuda>
      <tipo>1</tipo>
      <obriga>1</obriga>
    </n1>
    <n2>
      <id>2</id>
      <nome>Sexo</nome>
      <ordem>2</ordem>
      <ajuda></ajuda>
      <tipo>3</tipo>
      <obriga>1</obriga>
      <options>
        <n1>masculino</n1>
        <n2>feminino</n2>
      </options>
    </n2>
    <n3>
      <id>3</id>
      <nome>nivel de satisfacao</nome>
      <ordem>3</ordem>
      <ajuda></ajuda>
      <tipo>3</tipo>
      <obriga>1</obriga>
      <options>
        <n1>bom</n1>
        <n2>regular</n2>
        <n3>ruim</n3>
      </options>
    </n3>
  </perguntas>
</array>

```

Figura 19 – Arquivo de XML que gravado as perguntas

Na Figura 20 é apresentado o arquivo de XML onde os dados coletados são salvos. Assim como para as perguntas, para cada formulário é criado um arquivo para os dados coletados. Este XML tem a seguinte formatação: a *tag* `array` é nó principal, dentro dele vem as *tag* `n` e um número sequencial. Cada *tag* `n` guarda as respostas de um formulário respondido. Dentro da *tag* `n` estão as *tags* `id_número` que é o número da pergunta que foi respondida e dentro a resposta. Se for uma pergunta descritiva irá conter o que foi descrito na resposta e, se for uma resposta de múltiplas escolhas ou lista irá aparecer a identificação da resposta que foi selecionada.

```
<?xml version="1.0" encoding="utf-8"?>
<array>
  <n0>
    <id_1>Joao</id_1>
    <id_2>n1</id_2>
    <id_3>n1</id_3>
  </n0>
  <n1>
    <id_1>Maria</id_1>
    <id_2>n2</id_2>
    <id_3>n3</id_3>
  </n1>
  <n2>
    <id_1>Ana</id_1>
    <id_2>n2</id_2>
    <id_3>n2</id_3>
  </n2>
  <n3>
    <id_1>Pedro</id_1>
    <id_2>n1</id_2>
    <id_3>n3</id_3>
  </n3>
</array>
```

Figura 20 – Arquivo de XML que grava as perguntas

### 3.3.2 Operacionalidade da implementação

A Figura 21 apresenta a tela inicial do aplicativo, onde o usuário pode visualizar os formulários cadastrados. Nesta tela podem ser realizadas as opções de criar, editar e excluir formulários, além de aplicar formulários e visualizar dados coletados. Clicando no botão `exportar` enviam-se os dados coletados no dispositivo móvel para o LimeSurvey em um servidor externo, e clicando-se no botão `importar` recuperam-se os layouts dos formulários que estão no LimeSurvey para o dispositivo móvel.

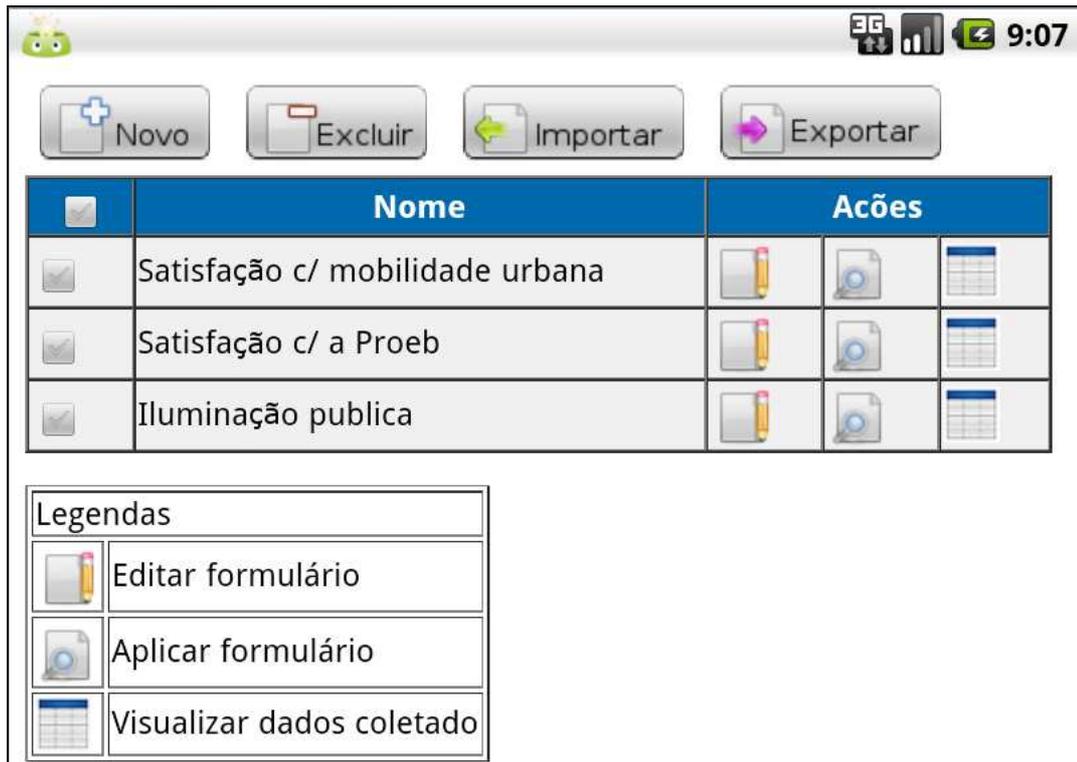


Figura 21 - Tela de formulários cadastrados

Ao clicar no botão novo o usuário é redirecionado para a tela de lista de perguntas (Figura 22). Nesta tela o usuário irá cadastrar o nome do formulário e clicando no botão novo ele poderá criar uma nova pergunta ou clicar em salvar que um novo formulário será criado.



Figura 22 - Tela de perguntas cadastradas

Clicando no botão novo o usuário é redirecionado para a tela de cadastro de perguntas (Figura 23). Nessa tela há cinco campos que são:

- a) título da pergunta: essa informação irá aparecer antes do campo no formulário;
- b) ajuda da pergunta: essa informação irá aparecer após o campo no formulário;
- c) ordem da pergunta: qual será a posição da pergunta no formulário;
- d) tipo da pergunta: qual será o campo que aparecerá no formulário que podem se do tipo texto, texto longo, lista, lista múltipla escolha (Figura 24);
- e) pergunta obrigatória: esse item indicará se a pergunta será de preenchimento obrigatório.

Titulo da pergunta \*

Ajuda da pergunta

Ordem da pergunta \*

Tipo da pergunta

Pergunta obrigatória

Figura 23 - Tela de cadastro de perguntas

Titulo da pergunta \*

Ajuda da pergunta

Ordem da pergunta \*

Tipo da pergunta

- Texto
- Texto Longo
- Lista
- Lista Multipla Escolha

Figura 24 - Lista de tipos de perguntas

Clicando em `salvar` na tela de cadastro de perguntas, o aplicativo salva as informações e redireciona o usuário para a tela de perguntas cadastradas onde o usuário poderá editar, excluir ou criar uma nova pergunta (Figura 25).



Figura 25 - Tela de perguntas cadastradas

Retornando a lista de formulários e selecionando o botão `aplicar formulário` o usuário é redirecionado para a tela de lista de perguntas (Figura 26). Nessa tela são listadas todas as perguntas cadastradas no formulário, de acordo com a ordem cadastrada. As perguntas com asterisco (\*) são de preenchimento obrigatório.

PESQUISA SOBRE PROPAGANDA MÉDICA MERICOM

QUAL A SUA OPINIÃO SOBRE A PROPAGANDA DIRETAMENTE AO PÚBLICO DE PRODUTOS FARMACEUTICOS NÃO SUBMETIDOS À PRESCRIÇÃO MÉDICA? \*

Não altera absolutamente a opinião do médico em relação à ca

OUVE-SE DE ALGUNS QUE O MÉDICO EVITA PRESCREVER ESPECIALIDADE DE CASAS QUE DIFUNDEM O PRÓPRIO NOME ENTRE O PÚBLICO PARA VENDA DE PRODUTOS NÃO SUJEITOS À PRESCRIÇÃO MÉDICA. QUAL A SUA OPINIÃO SOBRE A RESPEITO?

Acho que é verdade

SE UMA FARMACIA QUE GAZA DE SUA CONFIANÇA E DA QUAL PRESCREVE ALGUMAS ESPECIALIDADES LANÇASSE UM PRODUTO NÃO SUBMETIDO A PRESCRIÇÃO MÉDICA FAZENDO DIRETAMENTE A PROPAGANDA SOBRE O PÚBLICO. SEM CONTUDO USAR VISTOSAS FORMAS PUBLICITÁRIAS, QUAL SERIA SEU COMPORTAMENTO EM RELAÇÃO AQUELA CASA?

Figura 26 - Tela de aplicar formulário

Ao clicar no botão `visualizar dados coletados` na tela de lista de formulários o usuário é redirecionado para a tela de dados coletados, conforme Figura 27. Nesta tela irá aparecer uma tabela com todos os dados coletados do formulário e com gráfico de barras totalizando as respostas e abaixo dos gráficos existe um botão `voltar` para retorna para a tela inicial do aplicativo.

QUAL A SUA OPINIÃO SOBRE A PROPAGANDA ...	OUVE-SE DE ALGUNS QUE O MÉDICO EVITA PR...	SE UMA FARMACIA QUE GAZA DE SUA CONFIANÇA...	ACREDITA QUE AO PACIENTE AGRADA A PREESC...	Nome e Sobrenome	Especialidade
Não altera absolutamente a opinião do médico em relação à casa	Não é verdade	Continuarei a prescrever as especialidades	Sim	Joao da Silva	ortoped
O médico					

Figura 27 - Tela de dados coletados

Já para cadastrar um questionário no LimeSurvey basta clicar no terceiro botão da direita para a esquerda que o usuário será redirecionado para a tela criar, importar ou copiar questionário (Figura 28). Nessa tela são cadastrados os campos de idioma, título do questionário, uma descrição sobre o que trata o questionário, mensagem de boas-vindas, mensagem de final entre outros. Na integração só será usado o campo de título, os demais campos só irão aparecer quando for realizada a coleta de dados pelo próprio LimeSurvey. Os tipos de campos que estão aptos para realizar a integração são os seguintes:

- a) ordem;
- b) sexo;
- c) sim/não;
- d) lista (dropdown);
- e) múltipla escolha;
- f) texto livre curto;
- g) texto livre longo;
- h) textos curtos múltiplos.

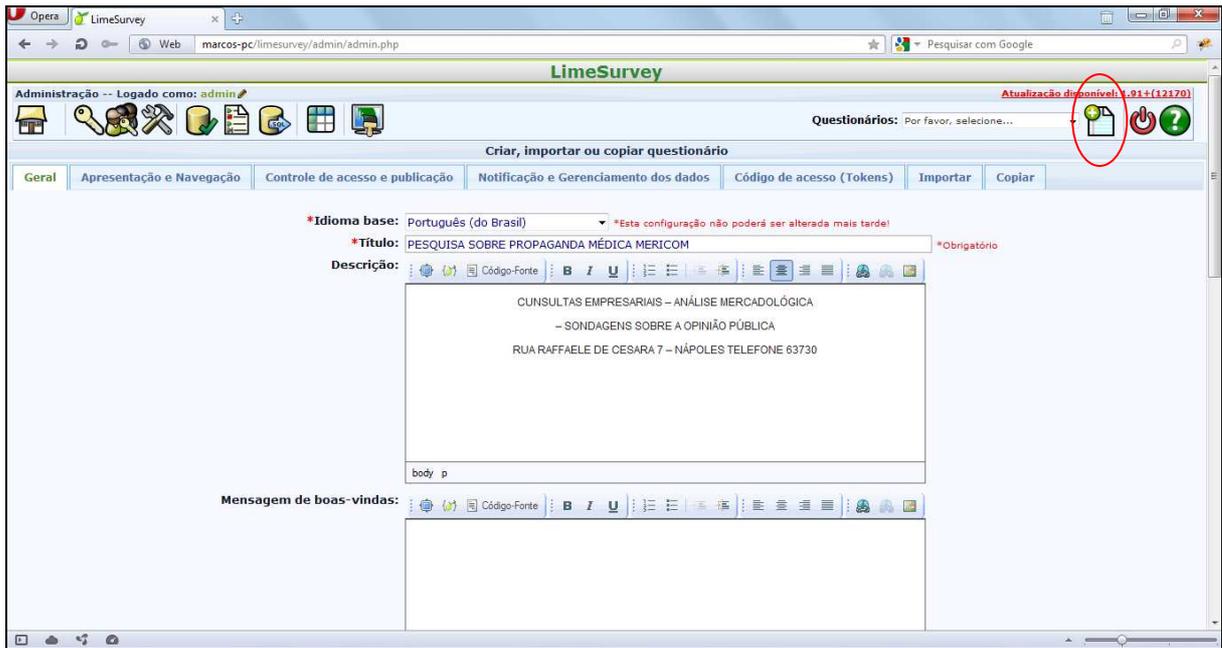


Figura 28 - Tela para criar, importar ou copiar questionário

No LimeSurvey para criar uma pergunta basta selecionar um questionário cadastrado dentre os que aparecem no campo no canto superior esquerdo da tela e em seguida selecionar um grupo de questões que irão aparecer logo abaixo do campo de questionário. Em seguida clicar no botão de adicionar perguntas (Figura 29). Nesta tela são cadastrados os campos de pergunta, ajuda, tipo de pergunta e se a pergunta é obrigatória. Estes são os campos serão usados na integração.

Figura 29 - Tela para adicionar pergunta

A Figura 30 é a tela para cadastrar e editar opções de resposta. Neste caso o usuário é redirecionado para essa tela quando ele escolhe uma pergunta de múltipla escolha. Assim o usuário irá adicionar as opções de respostas para a sua pergunta.



Figura 30 - Tela para cadastrar e editar opções de resposta

Assim que o questionário estiver pronto no LimeSurvey o usuário acessa o aplicativo e clica no botão importar e ele será redirecionado para a tela de importação dos questionários. Para escolher o questionário desejado basta posicionar o cursor sobre o nome e clicar no botão importar (Figura 31). Ao clicar no botão importar o aplicativo baixa as perguntas do questionário selecionado e deixa-as disponíveis para o usuário realizar a coleta de dados.

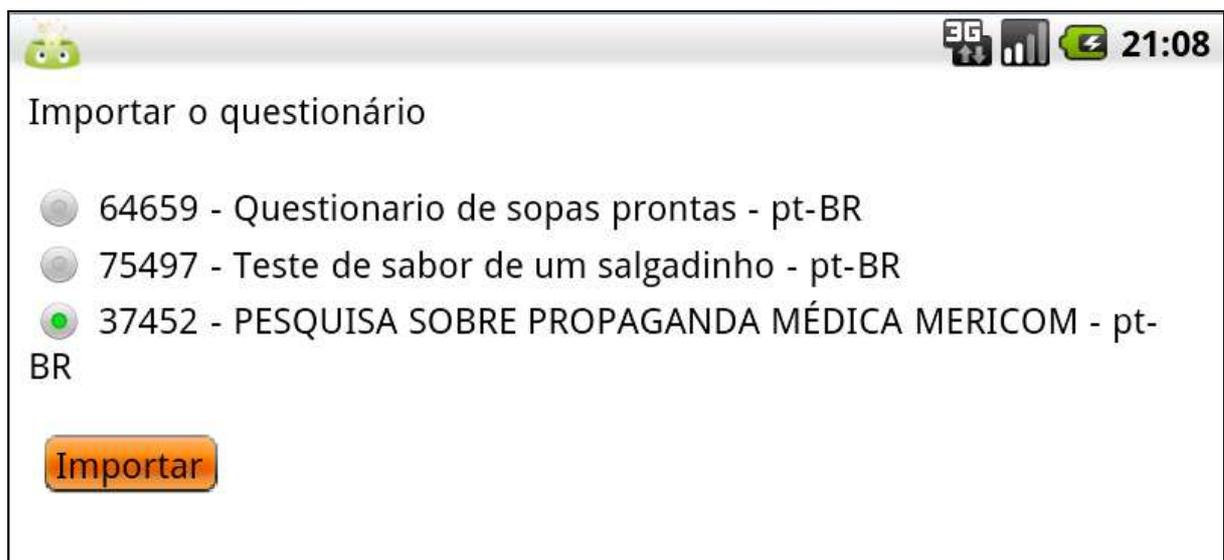


Figura 31 - Tela de importação dos questionários

Para exportar os dados coletados no aplicativo deve-se retornar para a lista de formulários. Lá o usuário deve selecionar questionário que deseja fazer a exportação e clicar em exportar. Dessa forma o aplicativo envia os dados para o LimeSurvey. Para o usuário verificar os dados no LimeSurvey, ele deve acessar o mesmo, selecionar o questionário desejado e clicar em respostas & estatísticas. Assim o usuário será redirecionado para a tela de registros onde deverá clicar no ícone de mostrar resposta. Neste momento ele será redirecionado para a tela de respostas cadastradas (Figura 32).

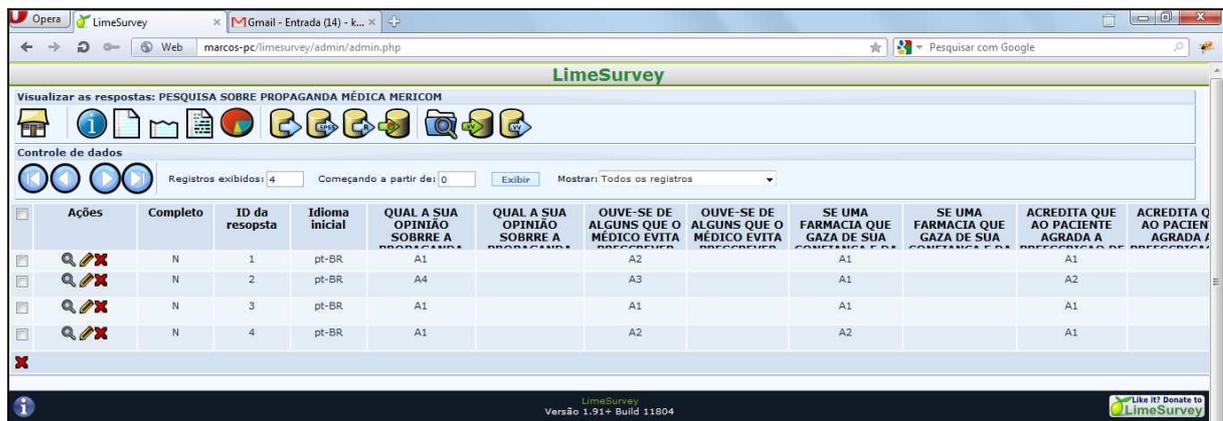


Figura 32 - Tela de dados coletados LimeSurvey

A Figura 33 apresenta um exemplo de relatório gerado pelo LimeSurvey a partir do questionário exemplo apresentado no Quadro 1. Esta figura caracteriza a primeira página do relatório o qual contém: a informação do nome e o número do questionário, assim como a quantidade total de questionários respondidos.

Estatísticas rápidas	
Questionário 37452 'PESQUISA SOBRE PROPAGANDA MÉDICA MERICOM'	
Resultados	
Questionário 37452	
Número de registros nesta consulta:	4
Total de registros no questionário:	4
Porcentagem do total:	100.00%

Figura 33 – Primeira página do relatório de estatísticas rápidas no LimeSurvey

As Figuras 34 traz os dados estatísticos gerados pelo relatório do LimeSurvey para segunda pergunta cadastrada. Essa parte do relatório traz as seguintes informações:

- código e o nome da questão;
- uma tabela com as opções de respostas cadastradas na questão, o total de resposta para cada opção e a percentagem de respostas para ela;
- um gráfico de pizza demonstrado as percentagem para cada resposta da questão.

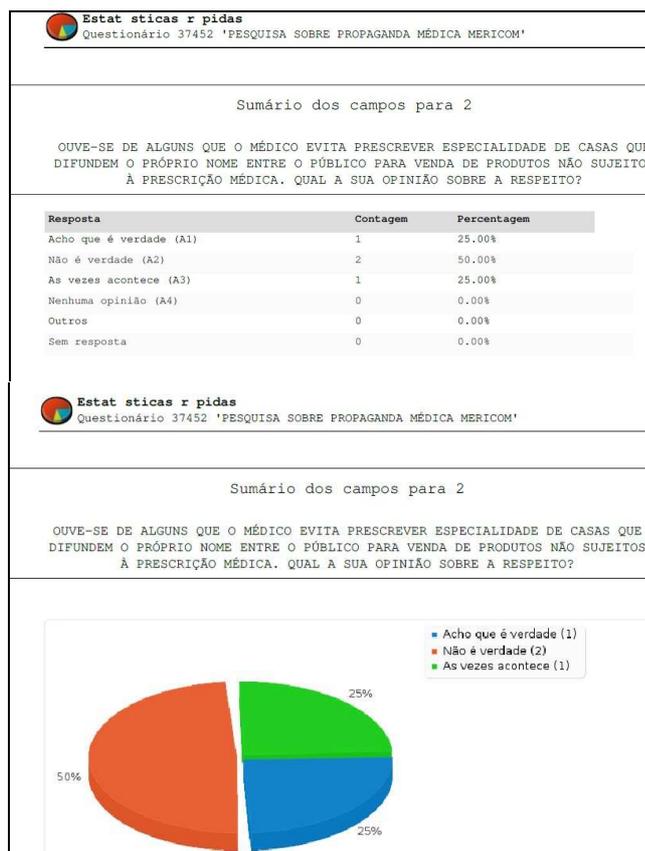


Figura 34 – Página com os resultados da pergunta dois no relatório do LimeSurvey

As Figura 35 traz os dados estatísticos gerados pelo relatório do LimeSurvey para a quinta e sexta pergunta cadastrada. Deve-se observar que o relatório não traz as respostas descritas na questão, pois o relatório é estatístico e não traz os dados descritivos. Para esse tipo de questão o LimeSurvey só informará se a questão foi respondida ou não. O mesmo acontece para as respostas que foram respondidas com a opção “outros”. Neste caso só haverá a indicação de que esta opção foi selecionada mas não a resposta fornecida.



Figura 35 – Página com os resultados da pergunta cinco no relatório do LimeSurvey

### 3.4 RESULTADOS E DISCUSSÃO

O desenvolvimento deste trabalho permitiu criar um aplicativo para Android com a linguagem PHP capaz de atender todos os requisitos levantados. Com isto está sendo disponibilizado um recurso adicional para que pesquisadores possam realizar pesquisas de coleta de dados em campo de uma forma mais eficiente.

A Figura 36 apresenta a imagem do aplicativo de monitoramento de recursos do Android obtida através do programa Smart Monitor (versão 1.5). A imagem foi capturada no momento em que foi realizado o processo de excluir um formulário. Com esse monitoramento se observou que o consumo de memória fica estável entre 72% a 77%. Já o consumo de processamento fica alto ao executar alguma função do aplicativo criado, chegando a picos de 97% como é o caso ao excluir um formulário.

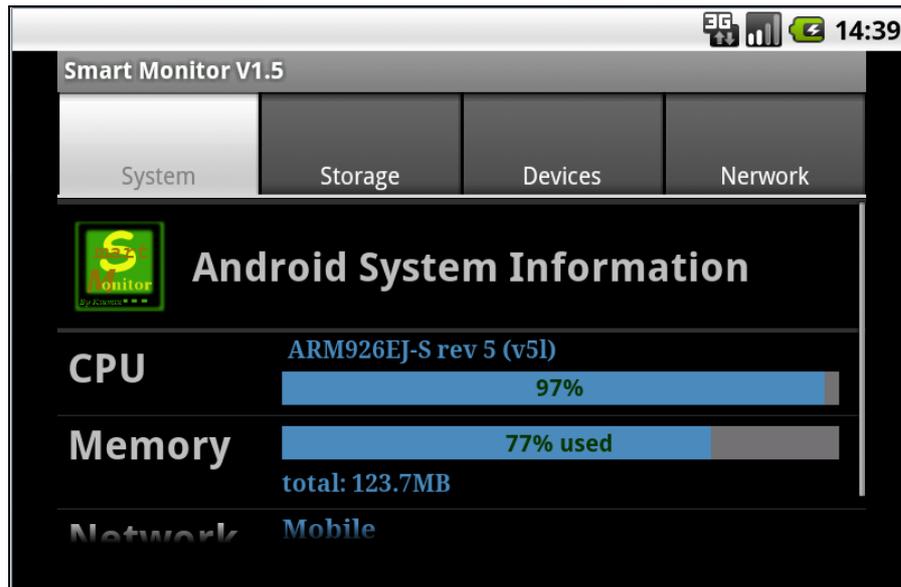


Figura 36 - Tela de monitoramento do consumo de memória e processamento no android

O Quadro 4 apresenta uma comparação do sistema desenvolvido com os trabalhos correlatos, baseado na fundamentação teórica e nas principais necessidades do desenvolvimento de software.

Funcionalidades/ características	Este trabalho	CHEQUETO (2007)	LimeSurvey	QuestManager
Plataforma	móvel (android)	móvel (JME)	<i>web</i>	<i>web</i>
Criação de formulários	sim	sim	sim	sim
Aplicação dos formulários	sim	sim	sim	sim
Relatório	sim	não	sim	sim
Tecnologias utilizadas	PHP, XML, PFA, SL4A	JME, DB2	PHP, MySQL, PostgreSQL, SQL Server	Java, MySQL, jQuery, PostgreSQL, Eclipse IDE

Quadro 4 – Comparativo entre o sistema desenvolvido e os trabalhos correlatos

Deve-se destacar que o armazenamento das informações salvas em XML é dependente do espaço que o dispositivo móvel tiver disponível em determinado momento. Normalmente, ao instalar-se o SL4A, os arquivos PHP e os XMLs ficam gravados no cartão de memória do dispositivo consumindo espaço útil do cartão de memória. Para os dados gravados no XML não foi utilizado nenhum tipo de criptografia assim todos os dados ficam visíveis caso o usuário abra o XML.

Deve-se destacar que o sistema foi validado em um tablet de 7" com Android 2.2.3. A Figura 37 caracteriza uma tela do sistema executando o questionário utilizado como referência neste trabalho. O tablet usado para testes foi um iBAK-793GPS com 512MB de memória embutida, 256 MB de RAM DDR2, processador de 600Mhz, câmera de 1.3 megapixels, slot para conexão 3G, wifi 802.11b/g, GPS e Bluetooth. O sistema operacional é o Android Froyo versão 2.2 . Neste tablet as vezes acontecem alguns travamentos por exemplo no momento de exportar os dados coletados, salvar dados coletados e visualizar os dados coletados. Nesses momentos era realizada a operação de retornar para a tela inicial do android e depois retornar para o aplicativo. Naturalmente a velocidade de transferência dos dados é altamente dependente da qualidade da conexão utilizada: *wireless* ou 3G.

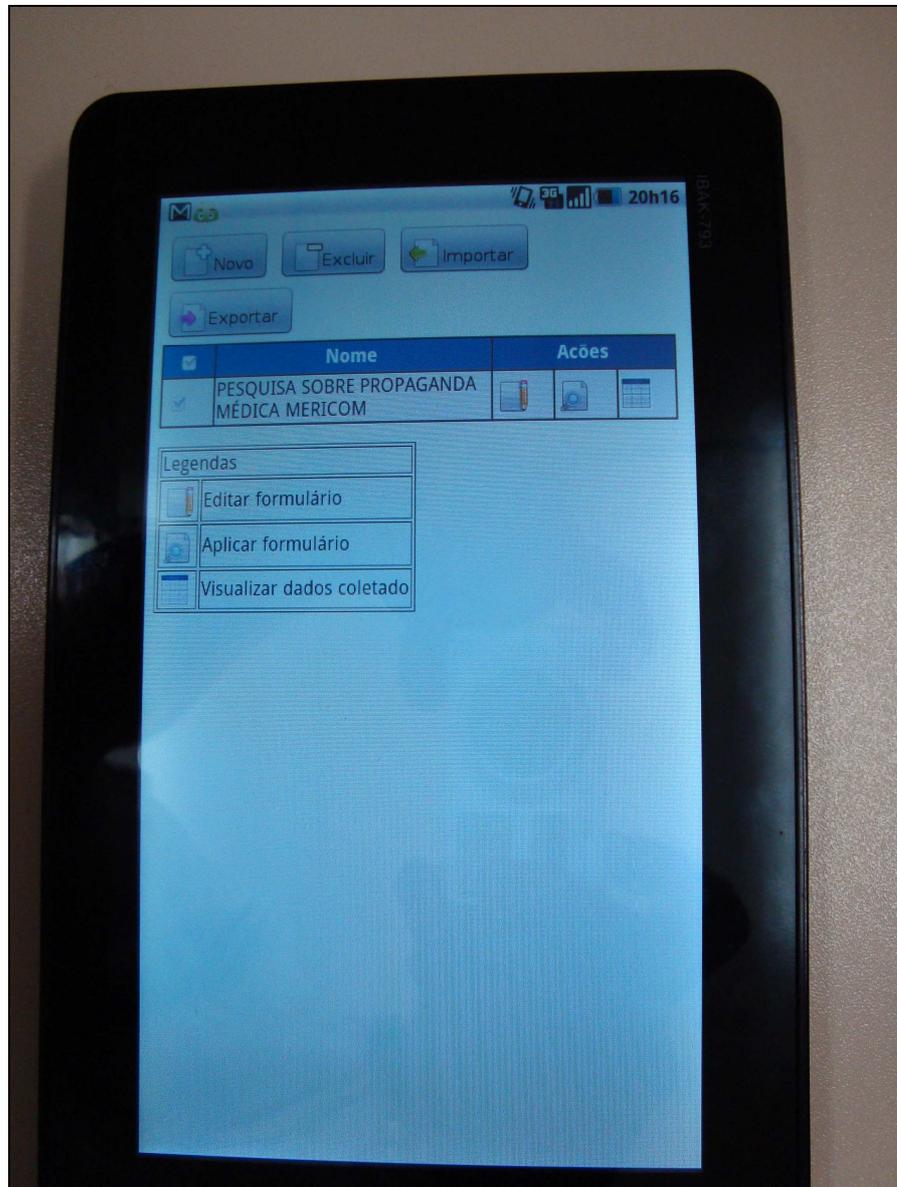


Figura 41 – Aplicativo executando em hardware real

## 4 CONCLUSÕES

Neste trabalho se propôs o desenvolvimento de uma aplicação para dispositivos móveis desenvolvida em linguagem PHP onde se criou um aplicativo que possibilita aos pesquisadores criar formulários de pesquisa no próprio dispositivo móvel e aplicá-los em campo. Além disso demonstrou-se a integração do aplicativo com a ferramenta LimeSurvey.

No desenvolvimento foram encontradas algumas dificuldades para a criação do aplicativo. Uma delas está relacionada com a criação da interface, por conhecimento mais profundo não foi possível utilizar a GUI que o SL4A oferecia. Em função disso foi necessário aprofundar o assunto de tal forma a viabilizar uma solução. A saída foi a criação de interface em HTML. Outro aspecto foi a falta de uma biblioteca nativa no PFA para manipulação de arquivos XML. Neste caso foi utilizada uma biblioteca externa (Very Fast XML Parser) para viabilizar a operação.

O aplicativo desenvolvido possibilita ao pesquisador criar formulários com várias perguntas que podem ser de 4 tipos: texto, texto longo, lista, múltipla escolha. Assim qualquer usuário poderá fazer uma coleta de dados sem precisar de um servidor externo. Esses formulários são aplicados no próprio dispositivo, onde o aplicativo monta o formulário com as perguntas ordenadas conforme o usuário definiu no cadastro. Os dados coletados podem ser visualizados no dispositivo móvel ou exportados para o software LimeSurvey. Além disso, permitiu-se a importação de formulários desenvolvidos no LimeSurvey para o dispositivo móvel fechando-se o ciclo de integração de ambos.

Esse aplicativo tem como sua principal vantagem estar no próprio dispositivo e assim não precisar de internet para realizar as pesquisas, assim em qualquer lugar que o pesquisador estiver ele pode realizá-la

### 4.1 EXTENSÕES

Como sugestão de implementação de funcionalidades para trabalhos futuros, pode-se destacar:

- a) criar um instalador para o aplicativo;
- b) criar interface web própria para visualizar dados em um servidor externo;

- c) criar interface web própria para criar os formulários em um servidor externo;
- d) ampliar o número de tipos de perguntas que podem ser criadas no dispositivo móvel;
- e) ampliar as possibilidades de relatórios que podem ser visualizados no dispositivo móvel.

## REFERÊNCIAS BIBLIOGRÁFICAS

AQUINO, Juliana F. S. **Plataformas de desenvolvimento para dispositivos móveis**. 2007. 14 f. Monografia (Pós Graduação em Informática) – Departamento de Informática, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro.

CANALYS. **Expert analysis for the high-tech industry**. [S.l.], 2011. Disponível em: <<http://www.canalys.com/pr/2011/r2011013.html>>. Acesso em: 03 abr. 2011.

CHEQUETO, Gilson. **Magregister 1.0: Gerador de Interfaces de Coleta de Dados para PDA's**. 2007. 79f. Trabalho de conclusão de curso (Bacharelado em Ciência da Computação) – Centro de Ciências Exatas e Naturais – Curso de Ciências da Computação, Fundação Universidade Regional de Blumenau, Blumenau, 2007.

FERRIL, Paul. **Pro Android Python with SL4A**. Apress : Nova York, 2011.

GOMES, Isabela Motta. **Manual Como Elaborar uma Pesquisa de Mercado**. Belo Horizonte: SEBRAE/MG, 2005.

GOOGLE. **Activity**. [S.l.], 2011a. Disponível em: <<http://developer.android.com/reference/android/app/Activity.html> >. Acesso em: 10 maio 2011.

GOOGLE. **What is Android**. [S.l.], 2011b. Disponível em: <<http://developer.android.com/guide/basics/what-is-android.html>>. Acesso em: 01 maio 2011.

GOOGLE. **Scripting Layer for Android brings scripting languages to Android**. [S.l.], 2011c. Disponível em: < <http://code.google.com/p/android-scripting> >. Acesso em: 05 ma. 2011.

HONORATO, Gilson. **Conhecendo o marketing**. Barueri : Manole, 2004.

IBGE. **Metodologia do censo demográfico 2000**. Rio de Janeiro : IBGE, 2003. 565 p, il. +, 2 CD-ROM. (Relatórios metodológicos, v.25).

IBGE. **Censo 2010, Síntese das etapas da pesquisa** . Rio de Janeiro : IBGE, 2010.

IBGE. **IBGE vai ao campo e às cidades traçar novo retrato do país**. [S.l.], 2007. Disponível em: <[http://www.ibge.gov.br/home/presidencia/noticias/noticia\\_visualiza.php?id\\_noticia=865&id\\_pagina=1&titulo=IBGE-vai-ao-campo-e-%E0s-cidades-tracar-novo-retrato-do-pais](http://www.ibge.gov.br/home/presidencia/noticias/noticia_visualiza.php?id_noticia=865&id_pagina=1&titulo=IBGE-vai-ao-campo-e-%E0s-cidades-tracar-novo-retrato-do-pais)>. Acesso em: 03 abr. 2011.

IRONTEC. **PHP for Android project**. [S.l.], 2010. Disponível em: <<http://phpforandroid.net/>>. Acesso em: 02 mar. 2011.

JORDAN, Lucas; GREYLING, Pieter. **Practical Andorid Projects**. New York : Apress, 2011.

LIMESURVEY. **The free & open source survey software tool**. [S.l.], 2007. Disponível em: <<http://www.limesurvey.org/>>. Acesso em: 24 maio 2011.

MALHOTRA, Naresh K. **Pesquisa de marketing: uma orientação aplicada**. 4. ed. Porto Alegre : ArTmed, 2006. xi, 720 p, il.

MCGRATH, Ben. **Simple polling with LimeSurvey**. [S.l.], 2007. Disponível em: <<http://www.linux.com/archive/articles/62327>>. Acesso em: 26 maio 2011.

PAULO, Ivan Mosquera. **Haciendo posible PHP sobre Android**. [S.l.], 2010. Disponível em: <<http://code.irontec.com/2010/haciendo-posible-php-sobre-android/>>. Acesso em: 02 mar. 2011.

QUESTMANAGER. **Software de pesquisa online e sistema de feedback com coletor offline**. [S.l.], 2002. Disponível em: <<http://www.questmanager.com/>>. Acesso em: 31 maio 2011.

SILVA, Severino Francisco da. **Marketing de serviços: fundamento, análises e práticas no setor de saúde**. Maceió : EDUFAL, 2005. 327 p.

TAGLIACARNE, Guglielmo. **Pesquisa de Mercado: técnica e pratica; tradução de Maria de Lourdes Rosa da Silva ; revisão técnica e adaptação de Noberto Roveda e de Roberto Simoes ; prefacio de Alfredo Carmo**. -2.ed. - São Paulo : Atlas, 1978. - 468p. :il.

## APÊNDICE A – Detalhamento dos casos de uso

Esta seção contém o detalhamento dos principais casos de uso descritos na seção 3.2.

No Quadro 5 apresenta-se o caso de uso "Cadastrar leiautes".

Nome do Caso de Uso	Cadastrar leiautes
Descrição	Usuário acessa aplicação, clica no menu criar leiaute, insere os campos, informa qual são os tipos dos campos e clica em salvar leiaute.
Ator	Usuário
Pré-condição	Sistema deve estar instalado no dispositivo móvel.
Fluxo principal	<ol style="list-style-type: none"> <li>1. Sistema informa leiautes cadastrados</li> <li>2. Usuário o indica se quer editar, apagar ou cadastra um leiaute;</li> <li>3. Ao optar por editar ou cadastrar o sistema direciona o usuário para a tela de leiaute;</li> <li>4. Usuário insere e/ou edita os campos indicando seus tipos.</li> <li>5. Usuário clica em salvar.</li> <li>6. Sistema salva os leiaute cadastrado</li> </ol>
Cenário – Visualização	Sistema mostra os registros de leiautes cadastrados para o Usuário.
Cenário – Exclusão	<ol style="list-style-type: none"> <li>1. Sistema mostra os leiautes cadastrados;</li> <li>2. Usuário seleciona um leiaute para exclusão;</li> <li>3. Sistema exclui o leiaute e mostra os leiautes brestantes.</li> </ol>
Pós-condição	Usuário visualizou, editou, apagou ou cadastrou um leiaute.

Quadro 5 – Descrição do caso de uso Cadastrar leiautes

No Quadro 6 apresenta-se o caso de uso "Exportar leiautes".

Nome do Caso de Uso	Exportar leiautes
Descrição	Usuário acessa a aplicação, seleciona o leiaute que deseja exportar e clica no botão exportar.
Ator	Usuário
Pré-condição	Sistema deve estar instalado no dispositivo móvel. O usuário deve ter um leiaute cadastrado.
Fluxo principal	<ol style="list-style-type: none"> <li>1. Usuário acessa a lista de leiautes</li> <li>2. Usuário seleciona o leiaute;</li> <li>3. Usuário clica no botão exportar.</li> <li>4. Sistema cria um arquivo XML com as informações do leiaute.</li> </ol>
Pós-condição	Usuário exportou um leiaute.

Quadro 6 – Descrição do caso de uso Exportar leiaute

No Quadro 7 apresenta-se o caso de uso “Importar leiautes”.

Nome do Caso de Uso	Importar leiautes
Descrição	Usuário acessa a aplicação, clica em importar leiautes e indica uma arquivo XML que deseja se importado.
Ator	Usuário
Pré-condição	Sistema deve estar instalado no dispositivo móvel. O usuário deve ter um arquivo XML para ser importado.
Fluxo principal	<ol style="list-style-type: none"> <li>1. Usuário acessa a lista de leiautes</li> <li>2. Usuário clica no botão importar leiaute;</li> <li>3. Usuário indica o arquivo XML que deve ser importado.</li> <li>4. Sistema cria um novo leiaute com as informações do arquivo.</li> </ol>
Pós-condição	Usuário importou o leiaute.

Quadro 7 – Descrição do caso de uso Importar leiaute

No Quadro 8 apresenta-se o caso de uso "Aplicar os formulários".

Nome do Caso de Uso	Aplicar os formulários
Descrição	Usuário acessa a aplicação, seleciona leiaute e clica no botão aplicar formulário
Ator	Usuário
Pré-condição	O usuário deve ter um leiaute cadastrado.
Fluxo principal	<ol style="list-style-type: none"> <li>1. Usuário acessa a lista de leiautes</li> <li>2. Usuário seleciona o leiaute;</li> <li>3. Usuário clica no botão aplicar formulário.</li> <li>4. Sistema cria uma tela com os campos criados no leiaute.</li> <li>5. Usuário preenche os campos.</li> <li>6. Sistema salva as informações digitadas no formulário em um arquivo XML</li> </ol>
Pós-condição	O usuário deve ter um leiaute cadastrado.

Quadro 8 – Descrição do caso de uso Aplicar os formulários

No Quadro 9 apresenta-se o caso de uso "Salvar dados coletados".

Nome do Caso de Uso	Salvar dados coletados
Descrição	O usuário preencher os campos do formulário e clicar no botão salvar
Ator	Usuário
Pré-condição	O usuário deve ter um leiaute cadastrado e preencher os campos do formulário.
Fluxo principal	<ol style="list-style-type: none"> <li>1. Usuário acessa a lista de leiautes</li> <li>2. Usuário seleciona o leiaute;</li> <li>3. Usuário clica no botão aplicar formulário.</li> <li>4. Sistema cria uma tela com os campos criados no leiaute.</li> <li>5. Usuário preenche os campos.</li> </ol>

	<ol style="list-style-type: none"> <li>6. O usuário clica em salvar</li> <li>7. Sistema insere todas as informação preenchidas pelo usuário em um <i>array</i>.</li> <li>8. Sistema transforma o <i>array</i> em um XML e salva o XML em um arquivo.</li> </ol>
Pós-condição	O usuário deve ter um leiaute cadastrado.

Quadro 9 – Descrição do caso de uso Salvar dados coletados

No Quadro 10 apresenta-se o caso de uso "Visualizar dados coletados".

Nome do Caso de Uso	Visualizar dados coletados
Descrição	Usuário acessa a aplicação, seleciona o leiaute e clica no botão visualizar dados.
Ator	Usuário
Pré-condição	<p>O usuário deve ter um leiaute cadastrado.</p> <p>O usuário deve ter um leiaute cadastrado.</p>
Fluxo principal	<ol style="list-style-type: none"> <li>1. Usuário acessa a lista de leiautes</li> <li>2. Usuário seleciona o leiaute;</li> <li>3. Usuário clica no botão visualizar dados.</li> <li>4. Sistema monta uma tela com os dados coletados</li> </ol>
Pós-condição	O usuário deve ter um leiaute cadastrado.

Quadro 10 – Descrição do caso de uso Visualizar dados coletados

## APÊNDICE B – Dicionário de dados

O dicionário de dados descreve em mais detalhes as entidades da modelagem do XML utilizada no trabalho desenvolvido. Os tipos de dados de cada campo são descritos a seguir:

- a) *varchar*: tipo de campo para armazenamento de strings de caracteres e seu tamanho é definido em *bytes* com largura variável, os valores entre parênteses definem o comprimento máximo em *bytes* de caracteres;
- b) *int*: tipo de campo para armazenamento de números inteiros;
- c) *smallint*: tipo de campo para armazenamento de números inteiros, fornece 2 bytes de armazenamento numérico;
- d) *text*: tipo de campo para armazenamento de grandes strings ou binários.

O dicionário de dados apresenta o nome do XML, o nome do campo, o tipo do campo, a descrição do campo e a observação para o caso do campo fazer parte de chave primária e/ou chave estrangeira.

O Quadro 11 contém o dicionário de dados da entidade “formulario”.

<b>Tabela: formularios</b>			
<b>Nome da coluna</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observação</b>
id	<i>int</i>	Código de registro do formulário.	Chave primária.
nome	<i>varchar(255)</i>	Nome do formulário.	

Quadro 11 – Dicionário de dados da entidade “formulario”

O Quadro 12 contém o dicionário de dados da entidade “perguntas”.

<b>Tabela: perguntas</b>			
<b>Nome da coluna</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observação</b>
id	<i>int</i>	Código de registro do formulário.	Chave primária.
id_formulario	<i>int</i>	Código do formulário que a pergunta pertence.	Chave estrangeira.
nome	<i>varchar(255)</i>	Nome da pergunta.	
ordem	<i>int</i>	Sequência que define a ordem que as perguntas devem aparecer no formulário	
ajuda	<i>varchar(255)</i>	Descrição para ajudar o usuário a responder a pergunta	

tipo	<i>int</i>	tipo da pergunta que deve aparecer no formulário	1: texto 2: texto longo 3: lista simples 4: lista múltipla escolha 5: textos curtos múltiplos 6: ordem
obriga	<i>smallint</i>	Inteiro de valor 1 ou 0 informando se a pergunta é obrigatória ou não.	
options	<i>varchar(255)</i>	Opções de respostas para a pergunta	

Quadro 12 – Dicionário de dados da entidade “perguntas”

O Quadro 13 contém o dicionário de dados da entidade “respostas”.

<b>Tabela: respostas</b>			
<b>Nome da coluna</b>	<b>Tipo</b>	<b>Descrição</b>	<b>Observação</b>
id	<i>Int</i>	Código de registro da resposta.	Chave primária.
id_formulario	<i>int</i>	Código da pergunta que a resposta pertence.	Chave estrangeira.
resposta	<i>text</i>	Descrição da repostas.	

Quadro 13 – Dicionário de dados da entidade “respostas”