

UNIVERSIDADE DO ESTADO DO RIO GRANDE DO NORTE

JOSÉ RAVEL DA SILVA PAIVA

**MODELAGEM DO SISTEMA DE REGISTRO DE FICHAS “A” DO SIAB PARA
ACOMPANHAMENTO DE SAÚDE FAMILIAR – *FICHAA-MOBILE ANDROID***

NOVA CRUZ

2012

JOSÉ RAVEL DA SILVA PAIVA

**MODELAGEM DO SISTEMA DE REGISTRO DE FICHAS “A” DO SIAB PARA
ACOMPANHAMENTO DE SAÚDE FAMILIAR – FICHAA-MOBILE ANDROID**

Trabalho de Conclusão de Curso
apresentado à Universidade do Estado do
Rio Grande do Norte como pré-requisito para
obtenção do grau de bacharel em Ciência da
Computação.

RAUL BENITES PARADEDA

FELIPE DENIS MENDONÇA DE OLIVEIRA

NOVA CRUZ

2012

**Catálogo da Publicação na Fonte.
Universidade do Estado do Rio Grande do Norte.**

Paiva, José Ravel da Silva

Modelagem do sistema de registro de fichas "A" do SIAB para acompanhamento de saúde familiar – *FichaA-Mobile Android* / José Ravel da Silva Paiva - Nova Cruz, RN, 2012.

68f.

Orientador(a): Prof. Raul Benites Paradedá

Monografia (bacharel). Universidade do Estado do Rio Grande do Norte. Curso de Ciência da Computação

1. *Android* – Monografia. 2. SIAB. 3. Usabilidade. I. Paradedá, Raul Benites. II. Universidade do Estado do Rio Grande do Norte. III. Título.

UERN/BC

CDD 004

JOSÉ RAVEL DA SILVA PAIVA

**MODELAGEM DO SISTEMA DE REGISTRO DE FICHAS “A” DO SIAB PARA
ACOMPANHAMENTO DE SAÚDE FAMILIAR – FICHAA-MOBILE ANDROID**

Trabalho de Conclusão de Curso
apresentado à Universidade do Estado do
Rio Grande do Norte como pré-requisito para
obtenção do grau de bacharel em Ciência da
Computação.

Aprovado em ____/____/____.

Banca Examinadora

RAUL BENITES PARADEDA

Universidade do Estado do Rio Grande do Norte

FELIPE DENIS MENDONÇA DE OLIVEIRA

Universidade do Estado do Rio Grande do Norte

TIAGO ALBUQUERQUE GAMELEIRA

Universidade do Estado do Rio Grande do Norte

AQUILES MEDEIROS FILGUEIRA BURLAMAQUI

Universidade Federal do Rio Grande do Norte

Dedico este trabalho aos meus pais Antônio e Sônia, meus irmãos Rafael, Rafaelly e Rogério, aos meus sobrinhos Raynara e Davi, e por fim, mas não menos importantes a minha esposa Socorro, a minha enteada Adriely e ao meu filho Riê que irá nascer em breve.

AGRADECIMENTOS

Agradeço primeiramente a Deus que me deu capacidade e força para vencer os obstáculos e dificuldades que encontrei.

Aos meus pais, que sempre, com muito carinho, me apoiaram e me incentivaram e incentivam a correr atrás dos meus objetivos. Aos meus familiares e amigos, pela compreensão e ajuda dada durante o curso.

À minha esposa e amiga Socorro, pela tolerância, apoio e compreensão, em todas as etapas do curso e desenvolvimento deste trabalho.

Ao meu filho Adryan Riê que irá nascer em breve, e que mesmo sem saber já é a minha maior fonte de inspiração e incentivo.

Ao meu orientador Raul Paradedá, e ao meu co-orientador Felipe Denis pelas orientações, dedicação e incentivos que me deram para conclusão deste trabalho.

A todos os funcionários permanentes e temporários da Universidade do Estado do Rio Grande do Norte, dos quais posso citar os professores Aquiles (atualmente na UFRN), Adriana, Bartira, Camila, Djalma, Glaucia, Isaac, Isânio (atualmente na DATAPREV), Leonardo (atualmente na UFERSA), Marcos Antônio, Rodolpho e a todos os demais que fizeram e ainda fazem parte desta grande instituição de ensino.

E a todos os colegas do curso, aqueles que caminharam e infelizmente desistiram e principalmente aqueles que continuaram na batalha mesmo depois de quedas que com esforço conseguiram ser superadas.

A vida ganhará plenitude no dia em que o ser humano encontrar um ideal pelo qual valha a pena viver e morrer.

Sören Kierkegaard

RESUMO

Este trabalho consiste na descrição da modelagem do sistema de registro de fichas “A” do SIAB denominado *FichaA-Mobile Android*. A fim de facilitar o processo de cadastramento das fichas, onde os Agentes Comunitários de Saúde (ACS) passarão a preencher estas de forma digital utilizando *Tablets* com o Sistema Operacional (SO) *Android* e com o *FichaA-Mobile Android* instalado, proporcionando agilidade, eficiência e praticidade. Para que isso seja possível, a criação do sistema precisa ser semelhante ao modelo que os ACS fazem uso atualmente, porém, utilizando recursos de interfaces gráficas de fácil compreensão. Para que isso seja atendido, faz-se necessário o estudo dos conceitos de usabilidade voltados para dispositivos móveis, principalmente na construção do *layout* do sistema. Os ACS poderão, além de cadastrar as fichas “A”, gerar relatórios e sincronizar os dados com o SIABMUN (Versão Municipal do SIAB). O principal objetivo deste trabalho é criar a modelagem do sistema para servir como referencial teórico para uma futura implementação do mesmo. Para uma melhor compreensão relacionada ao desenvolvimento da modelagem do sistema, neste trabalho são apresentados alguns conceitos relevantes ao desenvolvimento de aplicações com o SO *Android*, assim como Usabilidade e modelagem de sistema com diagramas UML (*Unified Modeling Language*).

PALAVRAS-CHAVE: *Android*, SIAB, Ficha “A”, Usabilidade.

ABSTRACT

This work is a description of the system modeling log sheets “A” of the SIAB called FichaA-Mobile Android. In order to facilitate the registration process of the chips, where the Community Health Agents (CHA) will fill these digitally using Tablets with Android Operating System (OS) and with the FichaA-Mobile Android installed, providing agility, efficiency and practicality. To make this possible, the creation of the system needs to be similar to the model that CHA makes use today, however, using features easy to understand graphical interfaces. For this to be satisfied, it is necessary to study the concepts of usability-oriented mobile devices, especially in the construction of the system layout. The CHA may, and register the chips 'A', generate reports and synchronize data with SIABMUN (Municipal Version of the SIAB). The main objective of this work is to create a modeling system to serve as a framework for future implementation. For a better understanding related to the development of system modeling, this paper presents some concepts relevant to developing applications with the Android OS, as well as Usability and system modeling with diagrams UML (Unified Modeling Language).

Keyword: Android, SIAB, Sheet “A”, usability.

LISTA DE ILUSTRAÇÕES

Figura 1– Arquitetura do <i>Android</i> .	19
Figura 2 - Gráfico distribuição das versões	22
Figura 3 - Diagrama de Caso de Uso do FichA- <i>Mobile Android</i> .	30
Figura 4 - Diagrama de Classes de Análise	37
Figura 5 - Diagrama ER do FichA- <i>Mobile Android</i>	38
Figura 6 – <i>Layout</i> da tela de Login	39
Figura 7 - <i>Layout</i> tela principal ACS	40
Figura 8 - <i>Layout</i> da aba Família	41
Figura 9 - <i>Layout</i> da aba Adulto	42
Figura 10 - <i>Layout</i> da aba Criança	42
Figura 11 - <i>Layout</i> da aba Moradia	43
Figura 12 - Uso dos componentes: <i>RadioButton</i> , <i>Checkbox</i> e <i>Spinner</i> .	44
Figura 13 - Opções de Menu	44
Figura 14 - Armazenar e Encaminhar	47
Figura 15 - Instalação da Plataforma no <i>SDK</i>	64
Figura 16 - Criação do <i>AVD</i>	64
Figura 17 - Iniciando o <i>AVD</i>	65
Figura 18 - Emulador em execução	65
Figura 19 - Selecionando <i>plug-in ADT</i>	66
Figura 20 - Instalação do <i>Plug-in</i>	66
Figura 21 - Configuração das preferências do <i>Eclipse</i>	67
Figura 22 - Criando Projeto <i>Hello World Android</i> no <i>Eclipse</i>	68
Figura 23 - Primeiro projeto sendo executado no emulador	68

LISTA DE TABELAS

Tabela 1 – Distribuição das Famílias no Município de Santo Antônio.....	14
Tabela 2 - Distribuição das versões.....	23
Tabela 3 - Uso apropriado e não apropriado do <i>SQLite</i>	24

LISTA DE ABREVIATURAS E SIGLAS

ACS	Agente Comunitário de Saúde
ADT	<i>Android Development Tools</i>
API	<i>Application Programming Interface</i>
APK	<i>Android Package File</i>
AVD	<i>Android Virtual Device</i>
BD	Banco de Dados
BSD	<i>Berkeley Software Distribution</i>
DBF	<i>Data Base File</i>
ER	Entidade Relacionamento
ESF	Estratégia Saúde da Família
GPS	<i>Global Position System</i>
IDE	<i>Integrated Development Environment</i>
JDK	<i>Java Development Kit</i>
MCU	Modelo de Caso de Uso
OHA	<i>Open Handset Alliance</i>
PSF	Programa Saúde da Família
SDK	<i>Software Development Kit</i>
SGBD	Sistema Gerenciador de Banco de Dados
SIAB	Sistema de Informação da Atenção Básica
SO	Sistema Operacional
UML	<i>Unified Modeling Language</i>
XML	eXtensible Markup Language

SUMÁRIO

Lista de Ilustrações	IX
Lista de Tabelas	X
Lista de Abreviaturas e Siglas	XI
1. Introdução.....	13
1.1 Justificativa.....	15
1.2 Objetivos.....	16
1.2.1 Objetivos Gerais.....	16
1.2.2 Objetivos Específicos.....	16
1.3 Estrutura do trabalho.....	17
2 Fundamentação Teórica.....	18
2.1 O que é <i>Android</i> ?.....	18
2.2 Arquitetura.....	18
2.3 Versões do <i>Android</i>	22
2.4 SGBD do <i>Android</i> - <i>SQLITE</i>	23
2.5 O que é Usabilidade?.....	25
2.6 Usabilidade e os dispositivos móveis.....	26
2.7 Práticas para desenvolvimento de interface com o usuário.....	27
3 Modelagem do sistema <i>FichaA-Mobile Android</i>	29
3.1 Descrição do sistema.....	29
3.2 Diagrama de casos de uso.....	30
3.2.1 Ilustração gráfica.....	30
3.2.2 Descrição textual.....	31
3.3 Diagrama de Classes.....	36
3.4 Diagrama Entidade-Relacionamento.....	37
3.5 Implementação das Interfaces.....	38
3.6 Propostas de Sincronização dos dados.....	45
3.6.1 Sincronização através do <i>JavaDBF</i>	45
3.6.2 Sincronização dos dados através de um SGBD.....	46
3.6.3 Sincronização através de um Servidor Web.....	46
4 Conclusão.....	48
5 Perspectivas de Trabalhos Futuros.....	49
6 Referências Bibliográficas.....	50
Apêndice A - Descrição dos Arquivos do SIAB – Módulo Municipal.....	52
Apêndice B - Ficha de cadastramento familiar – FichaA.....	58
Anexo A - Instalação do <i>Android SDK</i> e <i>Plug-in ADT</i> no <i>Eclipse</i>	62

1. INTRODUÇÃO

No ano de 1994 o Ministério da Saúde criou o Programa de Saúde da Família (PSF), atualmente chamado de Estratégia Saúde da Família (ESF), idealizado para aproximar os serviços de saúde da população. Com o objetivo de cumprir o princípio constitucional do Estado de garantir ao cidadão seu direito de receber atenção integral à saúde. O trabalho realizado pelo programa ESF gerou uma quantidade significativa de dados. O material recolhido e arquivado manualmente já se demonstrava insuficiente para o aproveitamento dos dados coletados. Por esse motivo o Sistema de Informação da Atenção Básica (SIAB) foi criado para gerenciar as informações obtidas nas ações e atividades feitas às comunidades pelas Equipes do programa ESF.

Através do SIAB, obtêm-se informações sobre cadastros de famílias, condições de moradia, e saneamento, situação de saúde, produção e composição das equipes de saúde (BRASIL, 2003).

O SIAB foi implantado em âmbito nacional em 1998 pelo Departamento de Atenção Básica/Secretaria de Atenção à Saúde, em conjunto com o Departamento de Informação e Informática do SUS/DATASUS/SE (SIAB, 2011). O SIAB possui dois módulos, o módulo Municipal (SIABMUN) e o módulo Estadual / Regional. O Módulo Municipal do SIAB está em funcionamento no município de Santo Antônio/RN desde o início de 2003.

Uma das funcionalidades do SIAB é o cadastro de famílias, esse processo é feito exclusivamente pelo Agente Comunitário de Saúde (ACS) nas primeiras visitas feita a família, onde são preenchidos os dados sobre os integrantes das famílias, situação de saúde e condições de moradias e saneamento em um formulário impresso conhecido como Ficha A (Apêndice B). Os ACS precisam carregar diversas fichas A durante as visitas feitas as famílias da Comunidade, visto que cada Ficha A representa uma única família e deve ser sempre atualizada quando for identificada alguma alteração na família. Após o preenchimento, as fichas são levadas ao Digitador da Secretaria de Saúde do Município para inserir essas informações no SIABMUN. Devido à caligrafia dos ACS surge um problema durante esse processo, onde em alguns casos se faz necessário à presença do próprio ACS

que preencheu a ficha para ditar as informações obtidas, e assim ter uma melhor inserção dos dados no SIAB. De acordo com o manual do SIAB, os dados obtidos permitem à equipe de saúde conhecer as condições de vida das pessoas de sua área de abrangência e melhor planejar suas intervenções (SIAB, 2003).

Esse processo gera retrabalho, redundância, e custos com material impresso, visto que o município de Santo Antônio/RN possui atualmente 47 ACS distribuídos em 9 ESF e a maioria dos ACS possuem entre 100 e 200 famílias, conforme os dados da Tabela 1.

Tabela 1 – Distribuição das Famílias no Município de Santo Antônio.

EQUIPE (ESF)	QTD ACS	Nº DE FAMILIAS
ESF ZONA URBANA I	06	833
ESF ZONA URBANA II	05	706
ESF ZONA URBANA III	06	833
ESF ZONA URBANA IV	05	819
ESF REDENÇÃO	05	388
ESF CAJAZEIRAS	04	338
ESF JUCA	06	529
ESF ANGICOS	03	321
ESF TIMBAÚBA	07	658
TOTAL	47	5.425

Fonte: SIAB/CNES 2012.

A fim de facilitar esse processo, este trabalho se propõe a criar a modelagem do sistema de registro das “fichas A” do SIAB, intitulado *FichaA-Mobile Android*. Gerando uma bibliografia para auxiliar uma futura implementação do aplicativo móvel, proporcionado de maneira geral, agilidade, praticidade, eficiência, e principalmente economia. Visto que, devido aos grandes avanços tecnológicos, os dispositivos móveis tem se tornado cada vez mais poderosos no que diz respeito a sua capacidade de processamento, armazenamento e comunicação e ao mesmo tempo mais acessíveis aos consumidores devido a sua redução gradual de preço.

1.1 JUSTIFICATIVA

A modelagem do *FichaA-Mobile Android* vai servir como referencial teórico para a implantação deste aplicativo que é importante tendo em vista que o município de Santo Antônio/RN tem um custo elevado com material impresso (Ficha A), entre outros fatores importantes e, ainda, os resultados obtidos não tem sido positivos, em consequência das constantes mudanças nas famílias das comunidades, como por exemplo: inclusão e/ou exclusão de algum integrante ou da própria família em si; condição de saúde referida; mudanças no domicílio e etc. Essas mudanças geram um grande número de atualizações a serem feitas no SIABMUN, onde se faz necessário um agendamento prévio semanal ou mensal por parte dos Agentes de Saúde para atualização das Fichas. Pelo fato dessa atualização se tornar algo repetitivo e massivo para os ACS, a maioria deles acaba atualizando as famílias apenas na Ficha A impressa e não na Ficha A do SIABMUN, o que gera inconsistência nos dados obtidos.

Como este trabalho servirá como base para a implantação do sistema proposto, irá permitir que a utilização do *FichaA-Mobile Android* possa diminuir os custos do Município e melhorar a qualidade da informação, visto que a atualização passará a ser feita durante a própria visita do ACS a Família, e semanalmente ou mensalmente haverá a sincronização dos dados com o SIABMUN.

O momento para desenvolver a modelagem do *FichaA-Mobile Android* é oportuno devido aos grandes avanços tecnológicos, principalmente na área de tecnologia móvel, e também com a possibilidade de obter *tablets* com a tecnologia *Android* a um custo bem acessível aos consumidores, e por possuírem telas maiores e com maior capacidade de processamento em detrimento aos *smartphones*. Portanto o projeto é considerável viável.

1.2 OBJETIVOS

1.2.1 OBJETIVOSGERAIS

Desenvolver a modelagem do banco de dados, modelagem da aplicação *FichaA-Mobile Android*, e aplicar os conceitos de Usabilidade para dispositivos móveis na criação das telas do sistema, com o objetivo de proporcionar um referencial teórico para a implantação de um aplicativo que permita o registro digital das “fichas A” do SIAB, geração de relatórios estatísticos para visualização dos dados e melhor planejamento das intervenções das Equipes de Saúde, e sincronização dos dados com o SIABMUN.

1.2.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos desse projeto de pesquisa são:

- Pesquisar sobre o Sistema Operacional (SO) *Android* e o *SQLite*, banco de dados utilizado pelo *Android*;
- Pesquisar sobre modelagem de Banco de Dados e modelagem do Sistema;
- Pesquisar sobre Usabilidade em dispositivos móveis;
- Modelar o Banco de Dados e o sistema proposto;
- Implementar as telas do sistema aplicando Usabilidade para dispositivos móveis;

1.3 ESTRUTURA DO TRABALHO

Após as considerações iniciais sobre o Sistema de Informação da Atenção Básica, assim como, a justificativa para o desenvolvimento do trabalho e os objetivos gerais e específicos, os próximos capítulos serão organizados da seguinte forma:

O Capítulo 2 abordará os aspectos relacionados ao SO *Android*, Usabilidade em dispositivos móveis e práticas para desenvolvimento de interface com o usuário.

No Capítulo 3 será apresentada a modelagem do sistema proposto, como: descrição do sistema, diagramas de casos de uso, diagramas de classes, diagrama ER do banco de dados, *layout* das interfaces e propostas de sincronização dos dados.

O Capítulo 4 exibirá a conclusão deste trabalho tiradas a partir dos estudos realizados.

No Capítulo 5 será apresentado as perspectivas futuras referentes às melhorias que deverão ser feitas no aplicativo e inclusão de novas funcionalidades.

O apêndice A apresenta a descrição dos arquivos do SIAB – módulo municipal, que irá servir como referencial para modelagem do banco de dados do *FichaA-Mobile Android*.

No apêndice B é apresentado o modelo impresso da ficha de cadastramento familiar (Ficha A) utilizada pelos ACS atualmente, e que servirá como modelo para desenvolvimento das telas do aplicativo.

E por fim, o Anexo A que apresenta o passo a passo para instalação do *Android SDK* e do *Plug-in ADT* no *Eclipse*, para dar início ao desenvolvimento de aplicações para o SO *Android*.

2 FUNDAMENTAÇÃO TEÓRICA

2.1 O QUE É *ANDROID*?

O *Android* é um conjunto de *softwares* para dispositivos móveis, compostos por aplicações-chaves, *middleware*, Sistema Operacional, e o *Android SDK* que é a plataforma para desenvolvimento das aplicações. (*ANDROID DEVELOPERS*, 2012b).

O *Android* é criação da *Open Handset Alliance (OHA)* que é um grupo formado por gigantes do setor de tecnologia e comunicação e liderada pela empresa *Google*. Atualmente, o grupo é composto por 85 empresas (operadoras de celulares, companhias de semicondutores, fabricantes de *handsets*, companhias de *software* entre outros), entre elas existem nomes já consagrados no mercado como: *China Mobile*, *Sprint Nextel*, *Intel*, *Broadcom*, *Nvidia*, *Samsung*, *Sony Ericsson*, *HTC*, *LG*, *Motorola* e muito mais.

O grupo foi criado com o objetivo de construir uma plataforma moderna e flexível para o desenvolvimento de aplicações corporativas e de definir uma plataforma única e aberta para celulares, com o intuito de deixar os consumidores mais satisfeitos com o produto final. (LECHETA, 2010). O anexo A deste trabalho apresenta o passo a passo para instalação do *Android SDK*, bem como, a instalação das plataformas de desenvolvimento.

2.2 ARQUITETURA

Os principais componentes do SO *Android* são mostrados na Figura 1. Sua arquitetura é dividida em cinco camadas: Aplicações, *Framework* de aplicações, Bibliotecas, *RunTime* e *Kernel*.

Figura 1– Arquitetura do Android.



Fonte: (ANDROID DEVELOPERS, 2012a).

Aplicações

O SO *Android* fornece um conjunto de aplicações básicas escritas na linguagem de programação *Java*. Algumas das aplicações básicas são clientes de e-mail, programa SMS, calendário, mapas, navegador, contatos e outros.

Framework de Aplicações

Ao fornecer uma plataforma de desenvolvimento aberta, a empresa *Google* através do SO *Android* oferece aos desenvolvedores a capacidade de criar aplicações extremamente ricas e inovadoras. Os desenvolvedores são livres para aproveitar as informações do dispositivo de *hardware*, acessar informações locais, executar serviços em segundo plano, definir alarmes, adicionar notificações à barra de *status*, manipular recursos de *GPS* (*Global Position System*) e muito mais.

Os desenvolvedores têm acesso completo às mesmas aplicações utilizadas pelo núcleo das *APIs* (*Application Programming Interface*) do *framework*. A arquitetura de aplicação é projetada para simplificar a reutilização de componentes. Podendo o desenvolvedor, alterar qualquer componente do sistema, para que esse

se adeque às suas necessidades ou as do aplicativo que está desenvolvendo. Este mesmo mecanismo permite que componentes sejam substituídos pelo usuário.

Juntamente a todas as aplicações são definidos serviços e sistemas, incluindo:

- Um rico e extensível conjunto de *Views* que podem ser usadas para construir aplicações, incluindo listas, grades, caixas de texto, botões, e até mesmo um navegador *web* embutido;
- Provedores de conteúdo que permitem que aplicações acessem dados de outras aplicações (como contatos), ou compartilhem seus dados;
- Um Gerenciador de Recursos, fornecendo acesso a recursos como localização de *strings*, gráficos e arquivos de *layout*;
- Um Gerenciador de Notificação que permite que todas as aplicações abertas apresentem alertas personalizados na barra de *status*;
- Um Gerenciador de Atividade que gerencia o ciclo de vida das aplicações e fornece um retorno comum de navegação.

Bibliotecas

O *Android* possui um conjunto de bibliotecas C/C++ usadas por vários componentes do sistema. Essas capacidades são expostas aos desenvolvedores através do *framework* de aplicação do *Android*. Algumas das principais bibliotecas estão listadas abaixo (*ANDROID DEVELOPERS, 2012a*).

- *System C library* – uma implementação *BSD (Berkeley Software Distribution)* derivada do padrão de biblioteca de sistema C (*libc*), fornecida para execução de sistemas baseados em *Linux*;
- *Media Libraries* – baseado no *OpenCore* da *PacketVideo* as bibliotecas suportam a execução e gravação de diversos formatos populares de áudio e vídeo, bem como arquivos e gravação de imagens estáticas, incluindo *MPEG4, H.264, MP3, AAC, AMR, JPG E PNG*;
- *Surface Manager* – gerencia o acesso ao subsistema apresentado e problemas relacionados às camadas gráficas 2D e 3D e múltiplas aplicações;

- *LibWebCore* – uma moderna *engine* de navegação *web* que alimenta tanto o navegador *web* do *Android* quanto um visualizador *web* embutido;
- *SGL* – abaixo da *engine* gráfica 2D;
- *3D libraries* – uma implementação baseada nas *APIs OpenGL ES 1.0* as bibliotecas usam tanto *hardware* de aceleração 3D (quando disponível), como *software* de otimização 3D;
- *FreeType* – fonte de renderização de *bitmaps* e vetores;
- *SQLite* – um leve e poderoso banco de dados disponível para todas as aplicações.

Android RunTime

O *Android* inclui um conjunto de bibliotecas que fornecem a maioria das funcionalidades disponíveis nas principais bibliotecas da linguagem de programação *Java*.

Cada aplicação *Android* executa em seu próprio processo, com sua própria instância da máquina virtual denominada *Dalvik*. Esta máquina virtual foi desenvolvida para que um dispositivo possa ser executado em várias máquinas virtuais de forma eficiente. A máquina *Dalvik* executa arquivos do *formato Dalvik Executable (.dex)* que é otimizado para ocupar o mínimo de memória. A máquina virtual é baseada em registradores, e executa classes compiladas pelo compilador da linguagem de programação *Java*. Depois que o *bytecode (.class)* é compilado ele é convertido para o formato *.dex (Dalvik Executable)*, que representa a aplicação do *Android* compilada. Depois disso, os arquivos *.dex* e outros recursos como imagens são compactados em um único arquivo com a extensão *.apk (Android Package File)*, que representa a aplicação final, pronta para ser distribuída e instalada.

Kernel

O *Android* precisa do *Linux* em sua versão 2.6 para execução dos serviços centrais do sistema, tais como segurança, gerenciamento de memória, gerenciamento de processos, pilha de rede e modelo de *driver*. O *kernel* também atua como uma camada de abstração entre o *hardware* e o resto da pilha de *software* (ANDROID DEVELOPERS, 2012a).

2.3 VERSÕES DO ANDROID

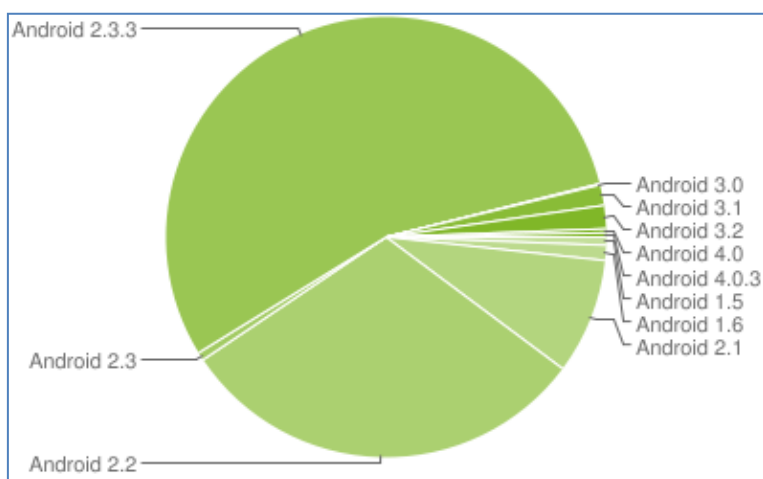
O *Android* possui diferentes versões. A primeira versão com integração total aos serviços *Google* foi a 1.1, lançada em fevereiro de 2009, e utilizada pelo dispositivo denominado HTC G1. A partir da versão 1.5, todas as versões passaram a receber nomes de sobremesas/doces em inglês e seguir uma ordem alfabética.

A segunda versão mais utilizada do *Android* é a 2.2 (*Froyo*) lançada em maio de 2010 e que trouxe melhorias importantes em comparação com as versões anteriores. Por exemplo, o aumento na velocidade de processamento, suporte ao *Adobe Flash*, instalação de aplicativos no cartão *SD*, compartilhamento de conexão 3G por cabo *USB* ou *Wi-Fi* entre outros (*ANDROID DEVELOPERS, 2012c*).

A Figura 2 e a Tabela 2 mostram a distribuição atual das versões do *Android*. Os dados são baseados no quantitativo de dispositivos *Android* que acessaram o *Android Market* dentro de um período de 14 dias, terminando em 03 de janeiro de 2012 (*ANDROID DEVELOPERS, 2012b*).

O *Android Market* é a loja oficial de aplicativos para dispositivos *Android* que inclui aplicativos, livros, filmes e música, acessível aos usuários em dispositivos *Android* e para qualquer pessoa na *Web*. Em 06 março de 2012 o *Google* lançou o *Google play* com o objetivo de substituir e estender o *Android Market*, permitindo que se possa não apenas baixar aplicativos e jogos, mas também músicas, vídeos e livros (*Google Inc, 2012*).

Figura 2 - Gráfico distribuição das versões



Fonte: (*ANDROID DEVELOPERS, 2012b*)

Tabela 2 - Distribuição das versões

Plataforma	Codename	Nível da API	Distribuição
Android 1.5	Cupcake	3	0,6%
Android 1.6	Donut	4	1,1%
Android 2.1	Eclair	7	8,5%
Android 2.2	Froyo	8	30,4%
Android 2.3 - Android 2.3.2	Gingerbread	9	0,6%
Android 2.3.3 - Android 2.3.7		10	54,9%
Android 3.0	Honeycomb	11	0,1%
Android 3.1		12	1,5%
Android 3.2		13	1,7%
Android 4.0 - Android 4.0.2	Ice Cream Sandwich	14	0,3%
Android 4.0.3		15	0,3%

Fonte: (ANDROID DEVELOPERS, 2012b)

2.4 SGBD DO ANDROID – SQLITE

SQLite é uma biblioteca em linguagem C que implementa um banco de dados SQL embutido, sem servidor, configuração zero, transacional, além de ser pequeno, rápido e confiável. As aplicações que utilizam a biblioteca *SQLite* podem ter acesso a banco de dados SQL sem executar um processo SGBD separado e a biblioteca *SQLite* lê e escreve diretamente para e do arquivo do banco de dados em disco. O código para o *SQLite* está no domínio público e é, portanto, livre para uso para qualquer finalidade, comercial ou privada (SQLITE, 2012a).

Características do *SQLite*

O *SQLite* possui algumas características que são incomuns e que o torna diferente de muitos outros bancos de dados SQL. Algumas dessas características serão destacadas a seguir.

SQLite é diferente da maioria dos outros bancos de dados SQL no qual o seu principal objetivo é ser simples: simples para administrar; simples de utilizar; simples de incorporar em projetos maiores e simples de manter e personalizar.

SQLite é uma boa escolha para dispositivos e sistemas que devem trabalhar de forma autônoma e sem apoio humano. *SQLite* é uma boa opção para uso em

celulares, PDAs, set-top box, e/ou eletrodomésticos (SQLITE, 2012b). Conforme a Tabela 3.

Tabela 3 - Uso apropriado e não apropriado do SQLite

Uso apropriado do SQLite	Uso não apropriado do SQLite
Dispositivos e sistemas embarcados	Websites com alto trafego
Websites com trafego médio	Alta concorrência
Aplicações desktop	Aplicações cliente/servidor
Celulares	Entre outros.
Set-top box	
Entre outros.	

Fonte: (SQLITE, 2012b)

Configuração zero

SQLite não precisa ser “instalado” e não há nenhum processo servidor que precisa ser iniciado, parado ou configurado. Também não há necessidade de um administrador para criar uma nova instancia do banco ou atribuir permissões de acesso para usuários. SQLite não usa arquivos de configuração. Nada precisa ser feito para informar ao sistema que está executando o SQLite e nenhuma ação é necessária para se recuperar após uma falha no sistema ou falta de energia, portanto não há nada para solucionar.

Arquivo único de banco de dados

Um banco de dados SQLite é um arquivo único de disco comum, que pode ser localizado em qualquer lugar na hierarquia de diretório. Arquivos de banco de dados podem ser facilmente copiados para um cartão de memória, dispositivos USB ou e-mail para o compartilhamento.

Outros bancos de dados SQL tendem a armazenar dados como uma grande coleção de arquivos. Muitas vezes, esses arquivos estão em uma localização padrão que só o próprio banco de dados pode acessar. Isso torna os dados mais seguros, mas também torna mais difícil o acesso.

Arquivo estável e multi-plataforma

O formato de arquivo *SQLite* é multi-plataforma, ou seja, um arquivo de banco de dados escrito em uma máquina pode ser copiado e utilizado em uma máquina diferente, com uma arquitetura diferente *big-endian* ou *little-endian*, 32-bit ou *64-bit* (*SQLITE, 2012c*).

2.5 O QUE É USABILIDADE?

A norma NBR ISO 9241 cujo título é “Requisitos econômicos para trabalho de escritório com computadores” é composta de dezessete partes, onde em sua parte 11, “Orientações sobre Usabilidade”, define usabilidade como: “Medida na qual um produto pode ser usado por usuários específicos para alcançar objetivos específicos com eficácia, eficiência e satisfação em um contexto específico de uso”.

A norma define os seguintes objetivos específicos:

- a) Eficácia: Acurácia e completude com as quais usuários alcançam objetivos específicos estão ligadas aos objetivos.
- b) Eficiência: Recursos gastos em relação à acurácia e abrangência com as quais usuários atingem objetivos estão ligados aos procedimentos a serem executados para atingir o objetivo.
- c) Satisfação: Ausência do desconforto e presença de atitudes positivas para com o uso de um produto.
- d) Contexto de uso: usuários, tarefas, equipamentos (*hardware, software* e materiais), e o ambiente físico e social no qual um produto é usado (NBR ISO 9241-11, 2002, p.3).

Segundo Nielsen (1993), a usabilidade é um dos aspectos que está diretamente ligado à aceitabilidade de um produto e se aplica a todos os aspectos do sistema que o usuário pode interagir, como: procedimentos de instalação e manutenção.

Nielsen a define através de cinco aspectos: aprendizagem, eficiência, memorização, erros e satisfação, para que ela possa ser avaliada e medida, e também que deve ser sempre medida relativamente a determinados usuários executando determinadas tarefas.

- a) Aprendizagem: o sistema deve ser de fácil aprendizado para que o usuário possa começar a utilizá-lo rapidamente.
- b) Eficiência: o sistema deve ser eficiente de tal forma que os usuários possam atingir uma alta produtividade.
- c) Memorização: o sistema deve ser intuitivo, para que o usuário ocasional não precise reaprender a utilizá-lo novamente.
- d) Erros: o sistema deve ser facilmente recuperável de erros e deve parar o usuário quando este cometer erros.
- e) Satisfação: grau de satisfação do usuário com o uso do sistema.

Observando as definições, concluímos que a usabilidade está associada às características de qualidade de produto no que diz respeito à adaptação do usuário, tipos de usuário e as características do ambiente de utilização do produto. Essas características podem aumentar o grau de satisfação e eficiência do usuário, e conseqüentemente, sua produtividade.

2.6 USABILIDADE E OS DISPOSITIVOS MÓVEIS

De acordo com *LEE et al. (2005)*, a usabilidade de um dispositivo é extremamente dependente de alguns fatores, tais como: características do dispositivo, o ambiente e o usuário.

- a) Características do dispositivo: os dispositivos móveis possuem características intrínsecas diferentes, que podem influenciar a usabilidade total do dispositivo, como: tempo de inicialização (o tempo de inicialização pode influenciar a usabilidade de determinadas aplicações); interface com o usuário (características próprias do dispositivo podem incapacitá-lo a desempenhar certas funções, como, a perda de produtividade na inserção de dados, devido ao tamanho

das teclas do teclado do dispositivo); e integridade dos dados (tolerância a perdas de dados e necessidade de armazenamento no próprio dispositivo).

- b) Características do ambiente: o ambiente de trabalho do usuário influencia diretamente na escolha do dispositivo. O dispositivo móvel deve trabalhar em condições normais de funcionamento e sobre condições extremas. Deve ser levado em consideração o tempo de utilização do dispositivo, se o usuário trabalha a maior parte do tempo em pé, e as condições de temperatura do ambiente (calor, frio etc.).
- c) Características do usuário: algumas características pessoais do usuário influenciam até certo ponto, na interação do usuário com o dispositivo móvel. Flexibilidade e destreza (um usuário com dedos maiores pode precisar de um teclado maior, para ter um melhor uso do dispositivo), conhecimento e capacidade (a falta de conhecimento e capacidade de aprender do usuário deve ser levado em consideração, com isso, os dispositivos devem ser intuitivos e fáceis de utilizar).

Analisando as informações do autor, podemos concluir que a usabilidade é um dos fatores mais importantes no desenvolvimento de aplicações móveis, e está diretamente ligada à aceitação da aplicação pelo usuário, e que se deve levar em consideração não apenas a interface com o usuário, e sim, as características acima descritas. Além disso, o estudo da Usabilidade é imprescindível para uma melhor apresentação dos campos de inserção de dados no sistema, devido à ficha A (Apêndice B) manipulada pelos ACS apresentam muitas informações.

2.7 PRÁTICAS PARA DESENVOLVIMENTO DE INTERFACE COM O USUÁRIO

A interface com o usuário de cliente móvel é a área primária de interação do usuário com o dispositivo móvel, e permite ao usuário emitir e receber informações das aplicações no dispositivo. Uma interface bem projetada melhora a experiência total do usuário e evita colocar em risco o uso da aplicação (*LEE et al., 2005*).

LEE et al. (2005), descreve um conjunto de práticas específicas para desenvolvimento de interfaces com o usuário de dispositivos móveis, algumas delas são apresentadas a seguir:

- a) Dependências de *hardware*: As características fundamentais de cada tipo de dispositivo móvel afetam a interface com o usuário (por exemplo, teclado, tela etc.) precisam ser levadas em consideração.
- b) Reduza a rolagem horizontal: Nos dispositivos móveis o espaço da tela é limitado. Por esse motivo, é importante nas perspectivas de usabilidade, minimizar ou, de preferência, evitar o uso da rolagem horizontal.
- c) Reduza a inserção de informações e a digitação: A inserção de dados no dispositivo móvel deve ser levado em consideração, lembrando que talvez o usuário tenha de entrar com os dados através de um teclado virtual. Que pode ser ao mesmo tempo difícil e sujeito a erros. Por exemplo, é relevante o uso de componentes como caixas de seleção, botões de opção e etc. Desse modo, diminuirá a quantidade real de digitação e fará com ele tenha uma experiência mais agradável.

As práticas para desenvolvimento de interface com o usuário, assim como, as definições e aspectos de usabilidade relatados anteriormente, servirão como base para implementação das telas do sistema, a fim de tornar fácil a utilização e aprendizagem do mesmo. Por exemplo, a ideia inicial era de que o sistema pudesse ser utilizado em *smartphones* e *tablets* com SO *Android*, mas levando em consideração as dependências de *hardware*, é mais recomendado limitar o uso do sistema aos *tablets*, pelo fato de possuírem telas maiores, e conseqüentemente facilitar a inserção de dados pelo usuário quando se fizerem necessários. Outro ponto importante diz respeito à aprendizagem do usuário com o sistema. Desse modo, é necessário fazer um treinamento para ambientação do sistema pelos usuários. As telas serão desenvolvidas baseando-se na mesma sequência das informações nos quais os ACS preenchem nos formulários de papel (Ficha A) atualmente, tornando intuitivo o preenchimento dos dados no *FichaA-Mobile Android*.

3 MODELAGEM DO SISTEMA FICHA-MOBILE ANDROID

O Sistema de *FichaA-Mobile Android* foi idealizado a partir de necessidade identificada no Município de Santo Antônio, de informatizar o processo de cadastro das fichas A do SIABMUN, passando a ser feito de forma digital, baseando-se na ficha A impressa (Apêndice B) que é comumente utilizada pelos ACS. Proporcionando de maneira geral praticidade, eficiência e agilidade.

Para uso do aplicativo recomenda-se *tablets* com telas de 7 polegadas para melhor aproveitamento das informações presentes na ficha A, conexão *WI FI* e 3G para sincronização dos dados através de um servidor *web* (ver seção 3.6.3), conexão *USB* e/ou *Bluetooth* para sincronização dos dados através do *JavaDBF* e/ou através do SGBD (ver seção 3.6.1 e 3.6.2), e por fim, processador de 800mhz ou superior.

3.1 DESCRIÇÃO DO SISTEMA

O sistema deve disponibilizar um sistema de *Login* confiável, garantindo que apenas os ACS cadastrados pelo Coordenador dos ACS tenham acesso ao sistema. O Coordenador entrará no sistema utilizando um usuário e senha padrão que lhe permitirá acesso a tela principal especifica para o seu perfil de usuário, lhe permitindo cadastrar, alterar e excluir os ACS. Os ACS por sua vez, só entraram no sistema, após terem sido cadastrados pelo Coordenador, com seu usuário e senha cadastrados por ele. O ACS poderá alterar apenas seu usuário e senha, os demais dados ficarão a cargo do Coordenador.

O ACS terá acesso a uma tela principal especifica com as principais funções do sistema como: cadastrar, alterar e excluir as famílias, gerar relatórios e sincronizar os dados. O cadastro das famílias será baseado no formulário impresso do SIAB, acrescentando alguns dados considerados relevantes ao sistema, como o nome dos adultos e das crianças pertencentes às famílias que no SIABMUN não são armazenados, e um telefone para contato. Os relatórios poderão ser exibidos em tela ou salvos em *PDF* e a sincronização dos dados poderá ser feita através de conexão via *Bluetooth* ou *USB*.

Baseando-se nesta descrição do sistema, nos tópicos a seguir serão descritos os diagramas referentes à modelagem do sistema: Diagrama de Casos de Uso, Diagrama de Classes e a modelagem do Banco de dados com o Diagrama Entidade-Relacionamento.

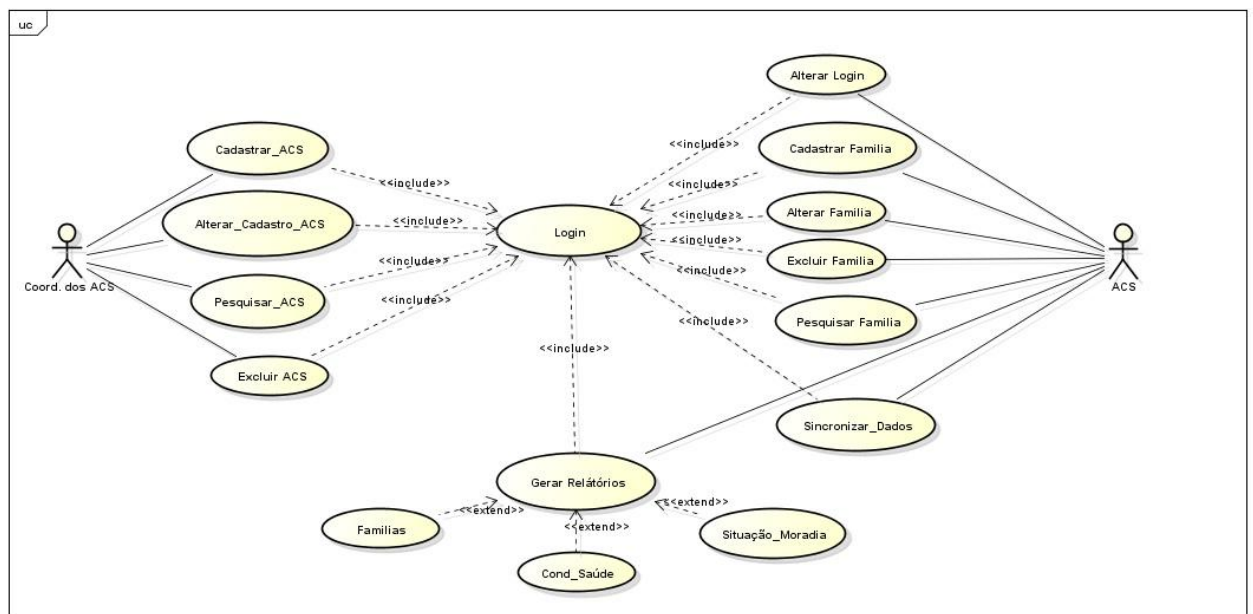
3.2 DIAGRAMA DE CASOS DE USO

Segundo Bezerra (2007), O modelo de casos de uso (MCU) é uma representação das funcionalidades externamente observáveis do sistema e dos elementos externos ao sistema que interagem com ele. De acordo com o mesmo autor, o MCU direciona e força os desenvolvedores a moldarem o sistema de acordo com as necessidades do usuário, e não o usuário de acordo com o sistema.

Levando em consideração a importância do Diagrama de casos de uso nas etapas de desenvolvimento de um sistema, foi elaborado o Diagrama de casos de uso do *FichaA-Mobile Android*, que será apresentado nos tópicos seguintes.

3.2.1 ILUSTRAÇÃO GRÁFICA

Figura 3 - Diagrama de Caso de Uso do FichaA-Mobile Android



Fonte: (Astah UML)

Para elaboração do diagrama de casos de uso representado na Figura 12, foi utilizado o software Astah UML.

3.2.2 DESCRIÇÃO TEXTUAL

Caso de Uso: *Login*

Objetivo: permitir que somente o Coord. dos ACS e os ACS cadastrados acessem o sistema;

Pré-condição ou requisitos: Estar cadastrado no sistema;

Ator: Coord. dos ACS, ACS

Cenário/Fluxo Principal:

1. O Coord. dos ACS e/ou ACS informa seu nome de usuário e senha de acesso e confirma;
2. Verifica se os dados estão corretos;
3. O sistema mostra a tela principal (a tela principal do Coord. será diferente da tela dos ACS, terá funcionalidades diferentes);

Cenário/Fluxo Alternativo:

1. Caso o usuário e/ou senha estejam incorretos, o sistema exibirá mensagem de alerta de usuário e/ou senha inválidos e solicitará nova tentativa;

Caso de Uso: *Cadastrar_ACS*

Objetivo: permitir o cadastro dos ACS do município, para que os mesmo tenham acesso ao sistema;

Pré-condição ou requisitos: O Coord. dos ACS está devidamente cadastrado e logado no sistema;

Ator: Coord. dos ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela para cadastro dos dados;
2. O Coord. dos ACS informará os dados do ACS a ser cadastrado, Como: Segmento, Área, Cod_Microarea, nome, *username* e senha;
3. Após a digitação dos dados, o Coord. confirma;

Cenário/Fluxo Alternativo:

1. Se algum dado estiver incompleto, o sistema exibe uma mensagem informando que o dado é obrigatório e solicita sua digitação;
2. Caso o usuário já exista, o sistema exibe uma mensagem de alerta informando que o mesmo já está cadastrado e solicitará um novo usuário;

Caso de Uso: Alterar_Cadastrar_ACS

Objetivo: permitir que o Coord. possa alterar os dados dos ACS cadastrados no sistema;

Pré-condição ou requisitos: O Coord. dos ACS está devidamente cadastrado e logado no sistema, e ter algum ACS cadastrado;

Ator: Coord. dos ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela com os ACS cadastrados;
2. O Coord. dos ACS selecionará o ACS a ter seus dados alterados;
3. O Coord. digitará os dados e depois confirma;

Cenário/Fluxo Alternativo:

1. Caso o ACS em questão não esteja na lista exibida na tela dos ACS cadastrados, o sistema exibirá mensagem informando que o ACS não está cadastrado;
2. Se algum dado estiver incompleto, o sistema exibe uma mensagem informando que o dado é obrigatório e solicita sua digitação;
3. Caso o usuário já exista, o sistema exibe uma mensagem de alerta informando que o mesmo já está cadastrado e solicitará um novo usuário;

Caso de Uso: Pesquisar_ACS

Objetivo: permitir que o Coord. possa verificar os ACS cadastrados no sistema;

Pré-condição ou requisitos: O Coord. dos ACS está devidamente cadastrado e logado no sistema, e ter algum ACS cadastrado;

Ator: Coord. dos ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela com os ACS cadastrados;
2. O Coord. dos ACS selecionará o ACS e em seguida definir qual ação será executada: Alterar ou Excluir ACS;

Cenário/Fluxo Alternativo:

1. Caso não exista nenhum ACS cadastrado, o sistema exibirá mensagem de alerta, informando que não existe cadastrado a ser exibido;

Caso de Uso: Excluir_ACS

Objetivo: permitir que o Coord. possa Excluir os dados dos ACS cadastrados no sistema;

Pré-condição ou requisitos: O Coord. dos ACS está devidamente cadastrado e logado no sistema, e ter algum ACS cadastrado;

Ator: Coord. dos ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela com os ACS cadastrados;
2. O Coord. dos ACS selecionará o ACS a ser excluído e depois confirmar a exclusão;

Cenário/Fluxo Alternativo:

1. Caso o ACS em questão não esteja na lista exibida na tela dos ACS cadastrados, o sistema exibirá mensagem informando que o ACS não está cadastrado ou já foi excluído;
2. Caso não exista nenhum ACS cadastrado, o sistema exibirá mensagem de alerta, informando que não existe ACS a ser excluído;

Caso de Uso: Alterar *Login*

Objetivo: permitir que o ACS possa alterar sua senha cadastrada no sistema;

Pré-condição ou requisitos: O ACS está devidamente cadastrado e logado no sistema;

Ator: ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela solicitando a senha atual e a nova senha;
2. Após a digitação dos dados, o ACS confirma;

Cenário/Fluxo Alternativo:

1. Caso a senha atual esteja incorreta, o sistema exibirá mensagem de alerta de senha inválida e solicitará nova tentativa;

Caso de Uso: Cadastrar_Familia

Objetivo: permitir que o ACS possa cadastrar as famílias de sua área no sistema;

Pré-condição ou requisitos: O ACS está devidamente cadastrado e logado no sistema;

Ator: ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela de cadastrado da família (Ficha A), dados referentes aos adultos, crianças, contato e condições de moradia da família;
2. Após a digitação dos dados, o ACS confirma;

Cenário/Fluxo Alternativo:

1. Se algum dado estiver incompleto, o sistema exibe uma mensagem informando que o dado é obrigatório e solicita sua digitação;

Caso de Uso: Alterar_Família

Objetivo: permitir que o ACS possa alterar os dados das famílias de sua área cadastradas no sistema;

Pré-condição ou requisitos: O ACS está devidamente cadastrado e logado no sistema, e a família em questão esteja cadastrada;

Ator: ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela com a lista das famílias cadastradas;
2. O ACS selecionará a família em questão e alterará os dados necessários;
3. Após a digitação dos dados, o ACS confirma;

Cenário/Fluxo Alternativo:

1. Se algum dado estiver incompleto, o sistema exibe uma mensagem informando que o dado é obrigatório e solicita sua digitação;

Caso de Uso: Pesquisar_Família

Objetivo: permitir que o ACS possa verificar todas as famílias de sua área cadastradas no sistema;

Pré-condição ou requisitos: O ACS está devidamente cadastrado e logado no sistema, e ter alguma família cadastrada;

Ator: ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela com a lista das famílias cadastradas;
2. O ACS selecionará a família em questão e em seguida definirá qual ação será executada: Alterar ou Excluir família;

Cenário/Fluxo Alternativo:

1. Caso não exista nenhuma família cadastrada, o sistema exibirá mensagem de alerta, informando que não existe família a ser exibido;

Caso de Uso: Excluir_Família

Objetivo: permitir que o ACS possa excluir as famílias de sua área cadastradas no sistema;

Pré-condição ou requisitos: O ACS está devidamente cadastrado e logado no sistema, e ter alguma família cadastrada;

Ator: ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela com a lista das famílias cadastradas;
2. O ACS selecionará a família em questão e em seguida definirá qual ação será executada: Alterar ou Excluir família;

Cenário/Fluxo Alternativo:

1. Caso não exista nenhuma família cadastrada, o sistema exibirá mensagem de alerta, informando que não existe família a ser excluída;
2. Caso a família em questão não esteja na lista exibida na tela das famílias cadastradas, o sistema exibirá mensagem informando que a família não está cadastrada ou já foi excluída;

Caso de Uso: Gerar Relatórios

Objetivo: permitir que o ACS possa gerar relatórios estatísticos referentes aos dados cadastrados sobre as famílias de sua área;

Pré-condição ou requisitos: O ACS está devidamente cadastrado e logado no sistema, e ter alguma família cadastrada;

Ator: ACS;

Cenário/Fluxo Principal:

1. O sistema exibir á tela com a lista das opções de relatórios: Família, Condição de Saúde, e Situação de Moradia;

- 1.1. Família: Quantidade de pessoas (adultos e crianças) por faixa etária ou geral;
- 1.2. Condição de Saúde: Quantidade de Hipertensos, Gestantes, Diabéticos e etc.;
- 1.3. Situação de Moradia: Tipo de casa, transporte, energia, água e etc.;
2. O ACS selecionará a opção desejada e os dados serão exibidos em tela ou em arquivo *PDF*;

Cenário/Fluxo Alternativo:

1. Caso não exista nenhuma família cadastrada, o sistema exibirá mensagem de alerta, informando que não existem dados a ser exibido;

Caso de Uso: Sincronizar_Dados

Objetivo: permitir que o ACS possa sincronizar os dados armazenados no *FichaA-Mobile Android* com o SIAB (Versão desktop municipal);

Pré-condição ou requisitos: O ACS está devidamente cadastrado e logado no sistema, e existir dados a serem sincronizados;

Ator: ACS;

Cenário/Fluxo Principal:

1. O sistema exibirá tela com as opções de sincronizações disponíveis: Via *USB* ou Via *Bluetooth*;
2. Escolher opção disponível;
3. Estabelecer Conexão e enviar os dados;

Cenário/Fluxo Alternativo:

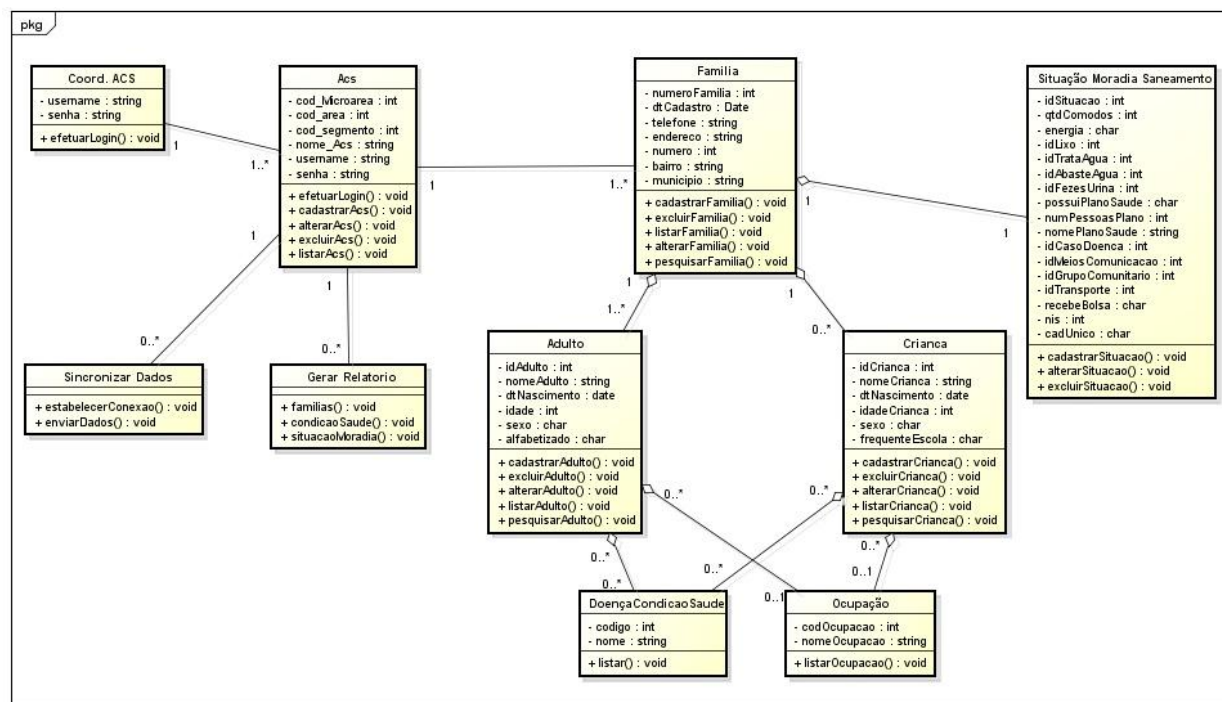
1. Caso não exista nenhum dado armazenado na Base de dados, o sistema exibirá mensagem de alerta, informando que não existem dados a serem Sincronizados;

3.3 DIAGRAMA DE CLASSES

O diagrama de classes é a representação da estrutura do sistema que resulta de um processo abstrato onde são identificados os objetos importantes para o sistema. De acordo com Bezerra (2007), há três níveis sucessivos pelos quais o modelo de classes passa: análise, especificação e implementação. A Figura 13 representa o diagrama de classes de análise, onde foram identificados as classes e

objetos importantes para o sistema, levando em consideração o que o sistema se propõe a fazer. É válido salientar que no modelo de classes de análise, não é levado em consideração às restrições inerentes à tecnologia utilizada. Para elaboração do diagrama de classes representado na Figura 4, foi utilizado o *software Astah UML*.

Figura 4 - Diagrama de Classes de Análise

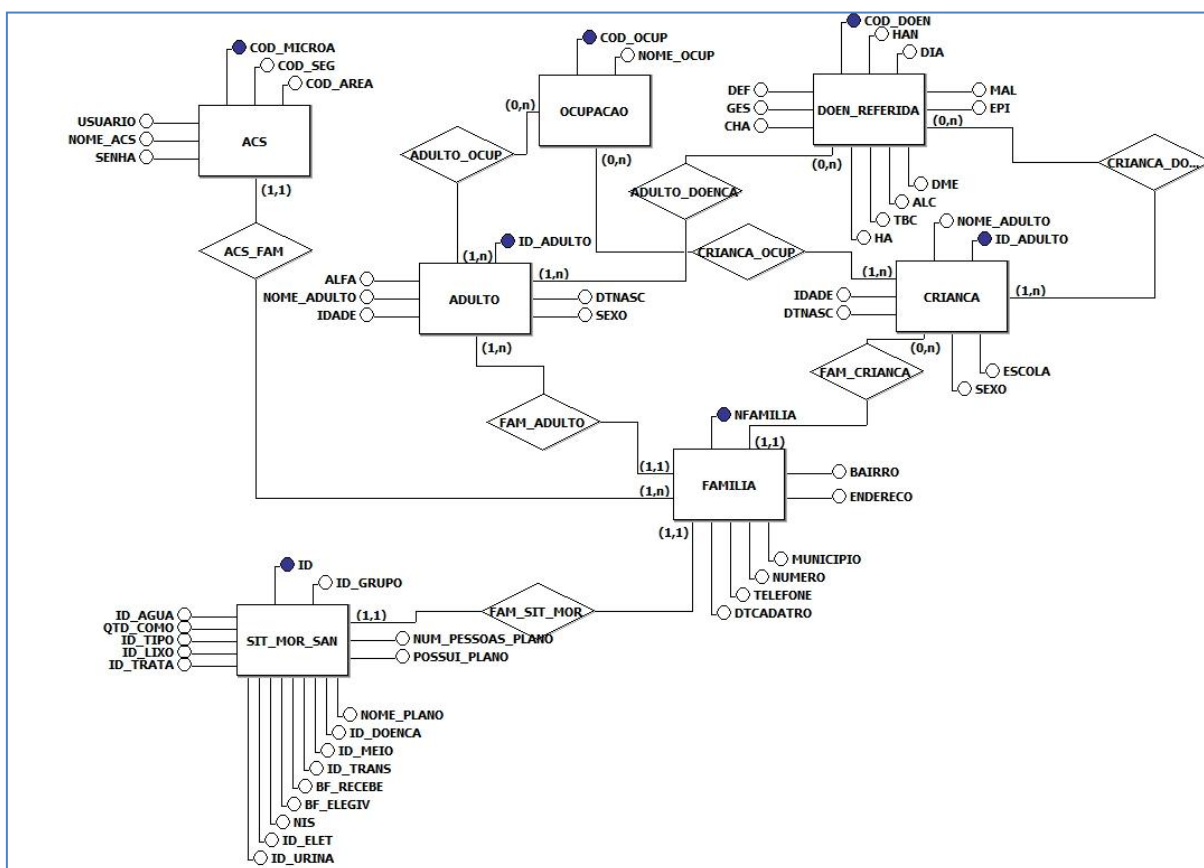


Fonte: (Astah UML)

3.4 DIAGRAMA ENTIDADE-RELACIONAMENTO

De acordo com *SILBERSCHATZ et al (2006)*, o modelo de dados Entidade-Relacionamento foi desenvolvido para facilitar o projeto de banco de dados, permitindo especificação de um esquema que representa a estrutura lógica geral de um banco de dados. A Figura 5 representa o diagrama ER conceitual do sistema *FichaA-Mobile Android*, onde foi utilizado o *software brModelo* para desenvolvimento do mesmo.

Figura 5 - Diagrama ER do FichaA-Mobile Android



Fonte: (Astah UML)

3.5 IMPLEMENTAÇÃO DAS INTERFACES

Realizado os estudos pertinentes sobre a tecnologia *Android* e sobre usabilidade, passou-se a ser desenvolvido o *layout* das telas do sistema proposto. As telas foram desenvolvidas em *Java* e *XML* (*eXtensible Markup Language*) utilizando a *IDE* (*Integrated Development Environment*) – *Eclipse*, o *Plug-in ADT* (*Plug-in* do *Android* para o *Eclipse*) e o *Android SDK* (*Software Development Kit*).

De antemão, optou-se por desenvolver uma interface gráfica ilustrada e intuitiva. A intenção foi que o ambiente ficasse presencialmente básico e fácil de usar, para que o ACS possa interagir com o aplicativo sem quaisquer dificuldades, e garantir os aspectos sobre usabilidade definidos por Nielsen (1993).

O sistema possuirá diversas telas, mas como primeiro protótipo, foram desenvolvidas, as telas de *login*, a tela principal referente aos ACS, e a tela de cadastro das fichas.

A tela de *login* exibida na Figura 6 será o ponto inicial da interação do usuário com o aplicativo e irá permitir o acesso, apenas aos ACS cadastrados e ao coordenador dos ACS. Após a validação do usuário e senha, os usuários serão direcionados para uma tela principal específica para cada tipo de usuário com os seus respectivos menus, e conseqüentemente, direcionados as telas referentes às opções apresentadas.

Figura 6 – Layout da tela de Login



Fonte: (AVD – Emulador 4.0)

A tela principal específica para os ACS é representada na Figura 7, e nela contém um menu com as opções referentes aos ACS, como: Ficha A; Pesquisar; Relatórios; Sincronizar dados; e Sair. As opções de menu serão acessadas através do botão MENU do dispositivo. A tela principal específica para o Coordenador dos ACS é semelhante a dos ACS, mudando apenas as opções de menu, que seriam: Cadastrar ACS; Pesquisar; Alterar; Excluir; e Sair.

Figura 7 - Layout tela principal ACS



Fonte: (AVD – Emulador 4.0)

A tela de cadastro das fichas A é a tela mais complexa de ser implementada e será a tela de maior interação com o usuário. A grande problemática foi encontrar uma solução viável para a inserção das muitas informações presentes na Ficha A impressa (Apêndice B) em um dispositivo de tamanho de tela reduzida. A solução encontrada foi dividir as informações em quatro abas: Família, Adulto, Criança e Moradia.

A aba Família representada pela Figura 8, recebe os dados referentes ao contato da família, como endereço, cidade e telefone.

Figura 8 - Layout da aba Família

FichaA-Mobile

11:04

Família Adulto Criança Moradia

CONTATO

Endereço:

Número: Bairro:

Município:

Nova Cruz

Telefone:

Fonte: (AVD – Emulador 4.0)

As abas Adulto e Criança representadas pelas Figuras 9 e 10 armazenam os dados referentes às pessoas com 15 anos e mais e pessoas com até 14 anos, respectivamente. Ainda nas abas adulto e criança estão presentes quatro botões com funções para adicionar, excluir e navegar entre os adultos e crianças pertencentes a cada família.

Figura 9 - Layout da aba Adulto

FichaA-Mobile

Escola Adulto Criança Moradia

PESSOAS COM 15 ANOS E +

Nome:

Nascimento: Idade:

Sexo: Feminino Masculino Alfabetizado: Sim Não

Ocupação: Autonomo

Doença ou Condição de Saúde:

ALC CHA DEF DIA EPI
 GES HA TB HAN MAL

NOVO Anterior Proximo Excluir

Fonte: (AVD – Emulador 4.0)

Figura 10 - Layout da aba Criança

FichaA-Mobile

Escola Adulto Criança Moradia

PESSOAS DE 0 A 14 ANOS

Nome:

Nascimento: Idade:

Sexo: Feminino Masculino Escola: Sim Não

Ocupação: Autonomo

Doença ou Condição de Saúde:

ALC CHA DEF DIA EPI
 GES HA TB HAN MAL

NOVO Anterior Proximo Excluir

Fonte: (AVD – Emulador 4.0)

A aba Moradia é responsável por receber os demais dados da ficha, como: tipo de casa, quantidade de cômodos, meios de transporte dentre outros. A Figura 11 representa a respectiva aba.

Figura 11 - Layout da aba Moradia

FichaA-Mobile

SITUAÇÃO DE MORADIA E SANEAMENTO

Tipo de Casa:
Tijolo/Adobe

Cômodos: Energia: Sim Não

Destino do Lixo:
Coletado

Tratamento da água no domicílio:
Filtração

Abastecimento de Água:
Rede pública

Destino Fezes e Urina:
Sistema de esgoto

Plano de Saúde:
 Sim Não

Nome do Plano: Pessoas:

Em Caso de Doença Procura:
Hospital

Meios de Comunicação que mais utiliza:
Rádio

Participa de Grupos Comunitários
Cooperativa

Meios de Transporte que mais utiliza:
Ônibus

A Família é beneficiária do Bolsa Família:
Bolsa Família Sim Não CAD-Único: Sim Não

NIS do responsável:

Fonte: (AVD – Emulador 4.0)

Observando as telas de cadastro das fichas A, apresentadas anteriormente, podemos observar o uso de componentes do *Android*, como: *Spinner*, *Checkbox* e *RadioButton*, com o objetivo de diminuir ao máximo a inserção de dados por parte do usuário e fazer com que tenha uma experiência mais agradável na utilização do aplicativo. A Figura 12 representa o uso dos respectivos componentes.

Figura 12 - Uso dos componentes: RadioButton, Checkbox e Spinner

The screenshot shows a mobile application interface titled 'FichaA-Mobile'. At the top, there are four icons representing different categories: 'Especial', 'Adulto', 'Criança', and 'Moradia'. Below these are two text input fields for 'Nascimento:' and 'Idade:'. The 'Sexo:' section has two radio buttons: 'Feminino' (unselected) and 'Masculino' (selected). The 'Alfabetizado:' section has two radio buttons: 'Sim' (selected) and 'Não' (unselected). There is a 'Ocupação:' dropdown menu currently showing 'Autonomo'. Below that is a 'Doença ou Condição de Saúde:' section with several checkboxes: 'ALC', 'CHA', 'DEF', 'DIA' (checked), 'EPI', 'GES', 'HA' (checked), 'TB', 'HAN', and 'MAL'. At the bottom of the form are four buttons: 'NOVO', 'Anterior', 'Proximo', and 'Excluir'.

Fonte: (AVD – Emulador 4.0)

Ainda na tela de cadastro das fichas é possível acessar as opções de menu, acionando o botão MENU do dispositivo, conforme a Figura 13.

Figura 13 - Opções de Menu

The screenshot shows the 'FichaA-Mobile' application with the 'CONTATO' (Contact) form. At the top, there are four icons: 'Familia', 'Adulto', 'Criança', and 'Moradia'. The form has a title 'CONTATO' and several input fields: 'Endereço:', 'Número:', 'Bairro:', 'Município:' (with a dropdown menu showing 'Areia Branca'), and 'Telefone:'. At the bottom of the screen is a navigation bar with four buttons: 'Novo' (orange), 'Salvar' (blue), 'Excluir' (red with an 'X' icon), and 'Voltar' (blue with a circular arrow icon).

Fonte: (AVD – Emulador 4.0)

3.6 PROPOSTAS DE SINCRONIZAÇÃO DOS DADOS

Um dos problemas identificados durante o desenvolvimento do trabalho era a respeito da sincronização dos dados, de qual forma esses dados serão transferidos do dispositivo móvel para o SIABMUN. A fim de solucionar este problema são explicitados três possibilidades de sincronização dos dados: sincronização através do *JavaDBF*, através de um SGBD e de um servidor web.

3.6.1 SINCRONIZAÇÃO ATRAVÉS DO *JAVADB*

A base de dados do SIAB (versão desktop municipal) foi desenvolvida utilizando o Sistema Gerenciador de Banco de dados *Dbase*, criando seus arquivos com extensão *DBF* (*Data Base File*), como se pode observar no dicionário de dados do SIABMUN presente no Apêndice A. Partindo dessa informação, a sincronização dos dados armazenados localmente no *FichaA-Mobile Android* com o SIABMUN (versão desktop) poderá ser feita através da biblioteca *JavaDBF*.

O *JavaDBF* é uma biblioteca do Java, que permite a leitura e escrita em arquivos com extensão *.dbf* (ABRAHAM; ANIL, 2003). Os dados poderão ser transferidos através de uma conexão via *USB* ou *Bluetooth*, onde a biblioteca permitirá a escrita direta nos arquivos da base de dados armazenados no diretório de instalação do SIABMUN.

Um ponto crítico dessa opção de sincronização dos dados é o fato de alterar de forma direta os dados armazenados na base de dados, onde em caso de erros durante a sincronização, poderá afetar a estrutura dos dados e tornar difícil a sua recuperação.

Uma solução para este problema seria fazer uma cópia temporária das tabelas do banco, fazer a inserção dos dados nestas tabelas e caso não tenham ocorrido erros, estas tabelas irão substituir as tabelas originais do banco.

3.6.2 SINCRONIZAÇÃO DOS DADOS ATRAVÉS DE UM SGBD

Na sincronização dos dados através de um Sistema Gerenciador de Banco de Dados (SGBD) os dados serão transferidos através de uma conexão *USB* ou *Bluetooth* para uma base de dados intermediária, utilizando SGBD como *postgreSQL* ou *MySQL*, por exemplo. A partir dos dados armazenados, serão criados arquivos com extensão *.dbf* na mesma estrutura dos arquivos da base de dados do SIAB (versão desktop municipal) que posteriormente irão substituir os arquivos presentes no diretório de instalação do SIABMUN.

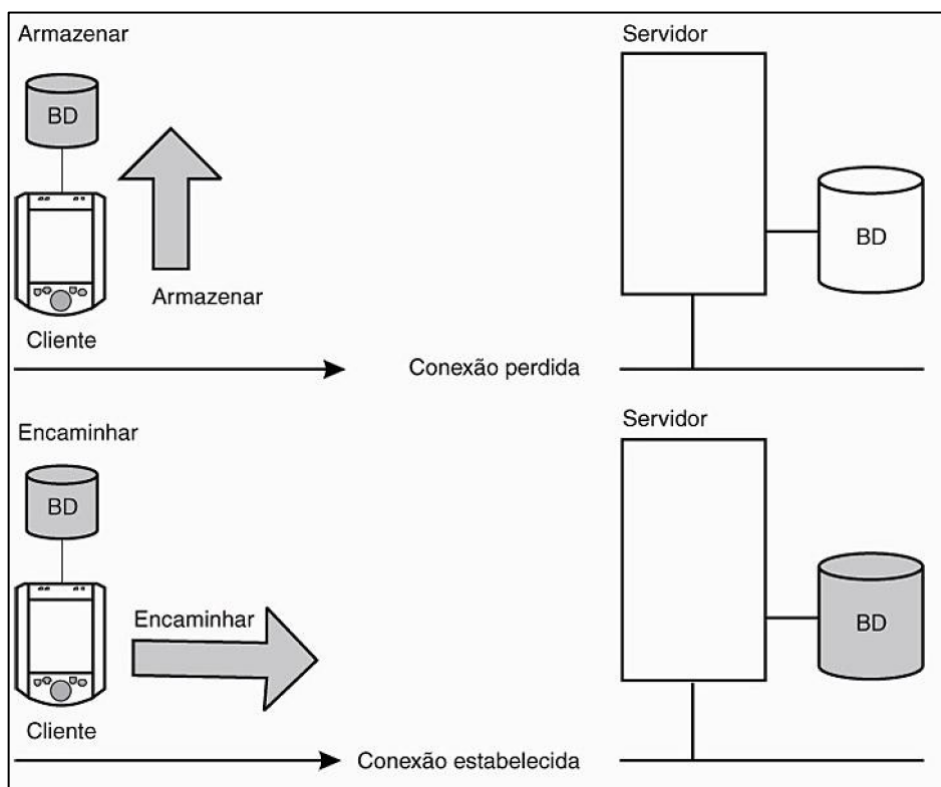
Uma vantagem importante desta opção de sincronização, é que não seria necessário alterar de forma direta os dados, evitando eventuais erros durante a sincronização, e tornando mais fácil a recuperação dos dados em caso de erros. Além das vantagens pertinentes dos SGBDs.

3.6.3 SINCRONIZAÇÃO ATRAVÉS DE UM SERVIDOR WEB

A sincronização dos dados através de um servidor *web* é semelhante à opção de sincronização dos dados através de um SGBD. A diferença é que os dados serão transferidos através de conexão com a *internet* e armazenados em uma base de dados presente no servidor *web*. A sincronização pode ser feita através do método chamado “armazenar e encaminhar”, conforme a Figura 14. No caso de não haver conexão com a *internet* no momento da sincronização, os dados ficam armazenados no banco dos dispositivos móveis, e após a conexão ser estabelecida os dados podem ser enviados para o servidor *web*.

De acordo com *LEE et al* (2005), um fator importante a ser observado nesse método, é assegurar a integridade dos dados, visto que, outros usuários poderão estar modificando dados que possivelmente podem estar em conflito com os dos dispositivos móveis. Esse fator levantado pelo autor, no caso do *FichaA-Mobile Android* não seria um problema, pelo fato de que durante a sincronização, os ACS só podem alterar os dados referentes à sua Micro Área, não sendo permitido à modificação de dados de outros ACS.

Figura 14 - Armazenar e Encaminhar



Fonte: (LEE et all, 2005, pg. 36)

Analisando as propostas de sincronização, é notório que a melhor opção seria a sincronização dos dados através de um servidor *web*, onde traria mais comodidade para o ACS, visto que, os dados poderiam ser enviados de qualquer lugar, necessitando apenas de conexão com a *internet*. Porém, esta opção é mais viável para municípios de grande porte, onde, os ACS teriam conexão com a *internet* de forma contínua e de qualidade. Como no município em questão mais de 50% dos ACS trabalham e residem na zona rural, dificilmente terão *internet* de forma contínua e de qualidade, portanto a sincronização dos dados através de um SGBD se torna mais interessante para o município.

4 CONCLUSÃO

Este trabalho teve como fruto a modelagem do sistema proposto, bem como, a modelagem da base de dados, os diagramas de casos de uso e de classes, e por fim, a implementação de algumas telas do sistema. A modelagem do sistema só foi possível graças ao conhecimento adquirido durante todo o curso de ciência da computação, e também as pesquisas realizadas sobre a tecnologia *Android*, Usabilidade em dispositivos móveis, e práticas para implementação de interface com o usuário.

A fundamentação teórica demonstrou a constante evolução da tecnologia móvel, assim como, os principais aspectos sobre o SO *Android*, e a importância da usabilidade no processo de desenvolvimento de um sistema, e evidência que a usabilidade está diretamente ligada à aceitação do sistema por parte do usuário.

Uma das problemáticas identificadas no desenvolvimento deste trabalho era encontrar uma solução viável para inserção das tantas informações presentes na ficha A impressa (Apêndice B) em um dispositivo de tela reduzida e que não viesse a comprometer a produtividade por parte do usuário. A solução encontrada foi dividir o conteúdo em abas, conforme explicitado anteriormente. Outra problemática é a sincronização dos dados, no presente trabalho foram mencionadas três possíveis soluções, ficando evidenciado que não há uma solução padronizada, e sim, que irá depender de caso a caso.

Concluído o desenvolvimento da modelagem do sistema, e sabendo que este servirá como referencial teórico para uma futura implantação do mesmo, a fim de agregar a mobilidade ao processo de cadastramento das fichas A do SIAB, pode-se sentir a emoção de um pequeno trabalho concluído, sentido ainda a sensação de incompletude diante de tantas possibilidades de continuidade deste trabalho.

5 PERSPECTIVAS DE TRABALHOS FUTUROS

Uma perspectiva de trabalho futuro é dar continuidade ao desenvolvimento do aplicativo para uma possível implantação do mesmo no município de Santo Antônio-RN. Fazer uma avaliação de usabilidade, possivelmente através de *checklists* com os ACS do município, para posteriormente fazer melhorias na usabilidade do aplicativo *FichaA-Mobile Android*.

Desenvolver um servidor web com o objetivo de permitir a sincronização dos dados através do servidor e atender as necessidades dos municípios de grande porte, como mencionado anteriormente.

O *FichaA-Mobile Android* ainda pode possuir muitas funcionalidades, que podem ser incorporadas em trabalhos futuros, tais funcionalidades podem ser úteis para o aperfeiçoamento do aplicativo e melhora da interação do usuário com o aplicativo, tais como: Inclusão de outras fichas utilizadas pelo ACS (B-GES, B-HA, B-DIA, dentre outras); controle de visitas domiciliares utilizando o *GPS* do dispositivo, para verificar se o ACS está visitando as famílias de sua área; dentre outras aplicações.

6 REFERÊNCIAS BIBLIOGRÁFICAS

ANDROID DEVELOPERS (2011a). **Get the Android SDK**. Disponível em: <<http://developer.android.com/sdk/index.html>>. Acessado em: 20 de dezembro de 2011.

ANDROID DEVELOPERS (2011b). **ADT Plug-in**. Disponível em: <<http://developer.android.com/sdk/eclipse-adt.html>>. Acessado em: 20 de dezembro de 2011.

ANDROID DEVELOPERS (2012a). **What is Android?**. Disponível em: <<http://code.google.com/android/what-is-android.html>>. Acessado em: 10 de janeiro de 2012.

ANDROID DEVELOPERS. (2012b). **Platform Versions**. Disponível em: <<http://developer.android.com/resources/dashboard/platform-versions.html>>. Acessado em 20 de janeiro de 2012.

ANDROID DEVELOPERS. (2012c). **Android 2.2 APIs**. Disponível em: <<http://developer.android.com/about/versions/android-2.2.html>>. Acessado em 20 de janeiro de 2012.

ABRAHAM, K. J; ANIL, K. K. **JavaDBF**. 2003. Disponível em: <<http://sarovar.org/projects/javadb/>>. Acessado em: 21 de setembro de 2012.

BEZERRA, E. **Princípios de análise e projeto de sistemas com UML**. Rio de Janeiro: Elsevier, 2007.

BRASIL. Ministério da Saúde, Secretaria de Atenção à Saúde, Departamento de Atenção Básica. **SIAB: Manual do sistema de Informação de Atenção Básica**. Brasília – DF: Ministério da Saúde, 2003.

DEITEL, H. M; DEITEL, H. M. **Java: como programar**. São Paulo: Pearson Prentice Hall, 2005.

ECLIPSE. **Eclipse Downloads**. Disponível em: <<http://www.eclipse.org/downloads/>>. Acessado em: 20 de dezembro de 2011.

ELMASRI, R; SHAMKANT, B. N. **Sistema de banco de dados**. São Paulo: Pearson Addison Wesley, 2005.

GOOGLE Inc. **Google play**. Disponível em: <<https://play.google.com/about/>>. Acessado em: 08 de março de 2012.

LECHETA, R. R. **Google Android – Aprenda a criar aplicações para dispositivos móveis com o Android SDK**. São Paulo: Novatec Editora, 2010.

LEE, V.; SCHNEIDER, H.; SCHELL, R. **Aplicações Móveis: arquitetura, projeto e desenvolvimento**. São Paulo: Pearson, 2005.

NIELSEN, J. **Usability engineering**. San Francisco: Morgan Kaufmann, 1993.

NBR ISO 9242-11. Requisitos Ergonômicos para Trabalho de Escritórios com Computadores Parte 11 – **Orientações sobre Usabilidade**. 2002.

OPEN HANDSET ALLIANCE (2012a). **Members**. Disponível em: <http://www.openhandsetalliance.com/oha_members.html>. Acessado em: 09 de janeiro de 2012.

OPEN HANDSET ALLIANCE (2012b). **Overview**. Disponível em: <http://www.openhandsetalliance.com/oha_overview.html>. Acessado em: 09 de janeiro de 2012.

PAULA FILHO; W. P. **Engenharia de software: fundamentos, métodos e padrões**. Rio de Janeiro: LTC, 2009.

SOMMERVILLE, I. **Engenharia de software**. São Paulo: Pearson Addison-Wesley, 2007.

SIAB. **Apresentação do SIAB**. Disponível em: <<http://www2.datasus.gov.br/SIAB/index.php?area=01>>. Acessado em: 16 de dezembro de 2011.

SILBERSCHATZ, A; HENRY, F. K.; SUBARSHAN, S. **Sistema de banco de dados**. Rio de Janeiro: Elsevier, 2006.

SQLITE (2012a). **About SQLite**. Disponível em: <<http://www.sqlite.org/about.html>>. Acessado em: 27 de fevereiro de 2012.

SQLITE (2012b). **Appropriate Uses For SQLite**. Disponível em: <<http://www.sqlite.org/whentouse.html>>. Acessado em: 27 de fevereiro de 2012.

SQLITE (2012c). **Distinctive Features of SQLite**. Disponível em: <<http://www.sqlite.org/different.html>>. Acessado em: 27 de fevereiro de 2012.

APÊNDICE A

DESCRIÇÃO DOS ARQUIVOS DO SIAB – MÓDULO MUNICIPAL

DESCRIÇÃO DOS ARQUIVOS DO SIAB – MÓDULO MUNICIPAL

ADULTO__.DBF – Corresponde às informações das pessoas maiores de 14 anos

CRIANC__.DBF – Corresponde as informações das pessoas menores de 15 anos

SANMUN__.DBF – Corresponde as informações da situação da moradia, saneamento e outras informações da família.

Os três arquivos citados acima são os arquivos que guardam as informações do cadastramento familiar, isto é a Ficha A do agente. Para cada registro de família no arquivo sanmun__.dbf podem existir vários registros de adultos e vários registros de crianças nos arquivos adulto__.dbf e crianc__.dbf, isso vai depender do número de pessoas de cada família.

Arquivo ADULTO__.DBF (os dois espaços, correspondem ao ano).

NOME	TIPO	TAMANHO	DESCRIÇÃO	PREENCHIMENTO
COD_SEG	C	2	Código do segmento	Obrigatório, com zeros à esquerda
COD_AREA	C	4	Código da área/equipe	Obrigatório, com zeros à esquerda
COD_MICRO A	C	2	Código da microárea	Obrigatório, com zeros à esquerda
NFAMILIA	C	3	Número da família	Obrigatório, com zeros à esquerda
DTNASC	C	10	Data de nascimento	Opcional
IDADE	C	3	Idade	Obrigatório
SEXO	C	1	Indica o sexo	Obrigatório, sendo "F" feminino e "M" masculino
ALFA	C	1	Indica se alfabetizado	Obrigatório, sendo "S" alfabetizado e "N" não alfabetizado
COD_OCUP	C	3	Código da ocupação	Conforme Tabela de Ocupação
NOME_OCUP	C	45	Nome da ocupação	Conforme Tabela de Ocupação
ALC	C	1	Indica Alcoolismo	"X" ou ""
CHA	C	1	Indica Doença de Chagas	"X" ou ""
DEF	C	1	Indica Deficiência Física	"X" ou ""
DIA	C	1	Indica Diabetes	"X" ou ""
DME	C	1	Indica Distúrbio mental	"X" ou ""
EPI	C	1	Indica Epilepsia	"X" ou ""
GES	C	1	Indica se Gestante	"X" ou ""
HAN	C	1	Indica Hanseníase	"X" ou ""
HÁ	C	1	Indica Hipertensão arterial	"X" ou ""
MAL	C	1	Indica Malária	"X" ou ""
TBC	C	1	Indica Tuberculose	"X" ou ""

Arquivo CRIANC___.DBF (os dois espaços, correspondem ao ano).

NOME	TIPO	TAMANHO	DESCRIÇÃO	PREENCHIMENTO
COD_SEG	C	2	Código do segmento	Obrigatório, com zeros à esquerda
COD_AREA	C	4	Código da área/equipe	Obrigatório, com zeros à esquerda
COD_MICROA	C	2	Código da microárea	Obrigatório, com zeros à esquerda
NFAMILIA	C	3	Número da família	Obrigatório, com zeros à esquerda
DTNASC	C	10	Data de nascimento	Opcional
IDADE	C	2	Idade	Obrigatório
SEXO	C	1	Indica o sexo	Obrigatório, sendo "F" Feminino e "M" Masculino
ESCOLA	C	1	Indica se está na escola	Obrigatório, sendo "S" na escola e "N" não na escola
COD_OCUP	C	3	Código da ocupação	Conforme Tabela de Ocupação
NOME_OCUP	C	45	Nome da ocupação	Conforme Tabela de Ocupação
ALC	C	1	Indica alcoolismo	"X" ou " "
CHA	C	1	Indica Doença de Chagas	"X" ou " "
DEF	C	1	Indica Deficiência Física	"X" ou " "
DIA	C	1	Indica Diabetes	"X" ou " "
DME	C	1	Indica Distúrbio Mental	"X" ou " "
EPI	C	1	Indica Epilepsia	"X" ou " "
GES	C	1	Indica se Gestante	"X" ou " "
HAN	C	1	Indica Hanseníase	"X" ou " "
HÁ	C	1	Indica Hipertensão arterial	"X" ou " "
MAL	C	1	Indica Malária	"X" ou " "
TBC	C	1	Indica Tuberculose	"X" ou " "

Arquivo SANMUN___.DBF(os dois espaços, correspondem ao ano).

NOME	TIPO	TAMANHO	DESCRIÇÃO	PREENCHIMENTO
COD_SEG	C	2	Código do segmento	Obrigatório, com zeros à esquerda
COD_AREA	C	4	Código da área	Obrigatório, com zeros à esquerda
COD_MICROA	C	2	Código da microárea	Obrigatório, com zeros à esquerda
COD_ZONA	C	2	Código da zona	Obrigatório, "01" – Urbana "02" – Rural
TIPO_EQP	C	2	Identificador do Tipo de Equipe	Obrigatório "01" – ESF "02" – ESF com SB1 "03" – ESF com SB2 "04" – EACS "05" – Outros "10" – EACS com SB1 "11" – EACS com SB2 "12" – ESFPR "13" – ESFPR com SB "14" – ESFF "15" – ESFF com SB
ID_MODELO	C	1	Identificador do modelo	Obrigatório, "1" – PACS "2" – PSF "3" – OUTROS
NFAMILIA	C	3	Número da família	Obrigatório, com zeros à esquerda
NPESSOAS	C	2	Número de pessoas da família	Obrigatório, com zeros à esquerda
FEM1	C	2	Número de pessoas do sexo feminino < 1 ano	
FEM2	C	2	Número de pessoas do sexo feminino de 1 a 4 anos	
FEM3	C	2	Número de pessoas do sexo feminino de 5 a 6 anos	
FEM4	C	2	Número de pessoas do sexo feminino de 7 a 9 anos	
FEM5	C	2	Número de pessoas do sexo feminino de 10 a 14 anos	
FEM6	C	2	Número de pessoas do sexo feminino de 15 a 19 anos	
FEM7	C	2	Número de pessoas do sexo feminino de 20 a 39 anos	
FEM8	C	2	Número de pessoas do sexo feminino de 40 a 49 anos	
FEM9	C	2	Número de pessoas do sexo feminino de 50 a 59 anos	
FEM10	C	2	Número de pessoas do sexo feminino >= 60 anos	
MASC1	C	2	Número de pessoas do sexo masculino < 1 ano	
MASC2	C	2	Número de pessoas do sexo masculino de 1 a 4 anos	
MASC3	C	2	Número de pessoas do sexo masculino de 5 a 6 anos	
MASC4	C	2	Número de pessoas do sexo masculino de 7 a 9 anos	
MASC5	C	2	Número de pessoas do sexo masculino de 10 a 14 anos	
MASC6	C	2	Número de pessoas do sexo masculino de 15 a 19 anos	

Arquivo SANMUN___.DBF (continuação)

NOME	TIPO	TAMANHO	DESCRIÇÃO	PREENCHIMENTO
MASC7	C	2	Número de pessoas do sexo masculino de 20 a 39 anos	
MASC8	C	2	Número de pessoas do sexo masculino de 40 a 49 anos	
MASC9	C	2	Número de pessoas do sexo masculino de 50 a 59 anos	
MASC10	C	2	Número de pessoas do sexo masculino >= 60 anos	
ESCOLA	C	2	Número de crianças de 7 a 14 anos na escola	
ALFABE	C	2	Número de pessoas de 15 anos e mais alfabetizados	
ALC1	C	2	Número de alcóolicos de 0 a 14 anos	
ALC2	C	2	Número de alcóolicos com 15 anos e mais	
CHA1	C	2	Número de pessoas com chagas de 0 a 14 anos	
CHA2	C	2	Número de pessoas com chagas com 15 anos e mais	
DEF1	C	2	Número de pessoas com deficiência física de 0 a 14 anos	
DEF2	C	2	Número de pessoas com deficiência física com 15 anos e mais	
DIA1	C	2	Número de pessoas com diabetes de 0 a 14 anos	
DIA2	C	2	Número de pessoas com diabetes com 15 anos e mais	
DME1	C	2	Número de pessoas com distúrbio mental de 0 a 14 anos	
DME2	C	2	Número de pessoas com distúrbio mental com 15 anos e mais	
EPI1	C	2	Número de pessoas com epilepsia de 0 a 14 anos	
EPI2	C	2	Número de pessoas com epilepsia com 15 anos e mais	
GES1	C	2	Número de gestantes de 10 a 19 anos	
GES2	C	2	Número de gestantes com 20 anos e mais	
HAN1	C	2	Número de pessoas com hanseníase de 0 a 14 anos	
HAN2	C	2	Número de pessoas com hanseníase com 15 anos e mais	
HA1	C	2	Número de pessoas com hipertensão arterial de 0 a 14 anos	
HA2	C	2	Número de pessoas com hipertensão arterial com 15 anos e mais	
MAL1	C	2	Número de pessoas com malária de 0 a 14 anos	
MAL2	C	2	Número de pessoas com malária com 15 anos e mais	
TBC1	C	2	Número de pessoas com tuberculose de 0 a 14 anos	
TBC2	C	2	Número de pessoas com tuberculose com 15 anos e mais	

Arquivo SANMUN___.DBF (continuação)

NOME	TIPO	TAMANHO	DESCRIÇÃO	PREENCHIMENTO
ID_TIPO	C	1	Identificador do tipo de casa	Obrigatório, "1" - Tijolo/Adobe "2" - Taipa revestida "3" - Taipa não revestida "4" - Madeira "5" - Material Aproveitado "6" - Outros
ID_ELET	C	1	Identificador da energia elétrica	Obrigatório, "1" ou " "
QTD_COMO	C	2	Quantidade de cômodos	
ID_LIXO	C	1	Identificador do destino do lixo	Obrigatório, "1" - Coletado "2" - Queimado/Enterrado "3" - Céu Aberto
ID_TRATA	C	1	Identificador do tratamento da água no domicílio:	Obrigatório, "1" - Filtração "2" - Fervura "3" - Cloração "4" - Sem tratamento
ID_AGUA	C	1	Identificador da água utilizada	Obrigatório, "1" - Rede Pública "2" - Poço ou Nascente "3" - Outros
ID_URINA	C	1	Identificador de fezes e urina	Obrigatório, "1" - Sistema de esgoto "2" - Fossa "3" - Céu Aberto
ID_DOEN	C	5	Identificador para em caso de doença	"1" - Hospital "2" - Unidade de Saúde "3" - Benzedeira "4" - Farmácia "5" - Outros
ID_MEIO	C	3	Identificador do meio de transporte	"1" - Rádio "2" - Televisão "3" - Outros
ID_GRUPO	C	4	Identificador de participação em grupos comunitários	"1" - Cooperativa "2" - Grupo Religioso "3" - Associações "4" - Outros
ID_TRAN	C	5	Identificador dos meios de transportes que mais utiliza	"1" - Ônibus "2" - Caminhão "3" - Carro "4" - Carroça "5" - Outros
COBERTUR A	C	2	Número de pessoas cobertas pór plano de saúde	
SIGAB	C	1	Identifica se as informações vieram do SIGAB	"*"
BF_RECEB E	C	1	Indica se a família recebe bolsa família	"S", "N" ou " "
BF_ELEGIV	C	1	Indica se a família está inscrita no Cad_Único	"S", "N" ou " "
NIS	C	11	Número do NIS do responsável	

APÊNDICE B

FICHA DE CADASTRAMENTO FAMILIAR – FICHAA

FICHA A

FICHA DE CADASTRAMENTO FAMILIAR

A **Ficha A** é preenchida nas primeiras visitas que o Agente Comunitário de Saúde (ACS) faz às famílias de sua comunidade. Deve ser preenchida uma ficha por família.

As informações recolhidas - identificação da família, cadastro de todos os seus membros, situação de moradia e outras informações adicionais - permitem à equipe de saúde conhecer as condições de vida das pessoas da sua área de abrangência e melhor planejar suas intervenções.

Todos os dados desta ficha devem ser atualizados sempre que houver alteração. O ACS deve estar atento para registrar, todo mês, a ocorrência de nascimentos, mortes e mudanças de atividade profissional (ocupação) dos membros da família e as condições de moradia e saneamento.

FICHA A		SECRETARIA MUNICIPAL DE SAÚDE SISTEMA DE INFORMAÇÃO DE ATENÇÃO BÁSICA					UF __ __	
ENDEREÇO				NUMERO _ _ _ _ _	BAIRRO		CEP _ _ _ _ _ - _ _	
MUNICÍPIO _ _ _ _ _ _ _ _ _	SEGMENTO _ _	ÁREA _ _ _	MICROÁREA _ _	FAMÍLIA _ _ _	DATA _ _ - _ - _ _ _ _			

CADASTRO DA FAMÍLIA							
PESSOAS COM 15 ANOS E MAIS NOME	DATA NASC.	IDADE	SEXO	ALFABETIZADO		OCUPAÇÃO	DOENÇA OU CONDIÇÃO REFERIDA (sigla)
				sim	não		

PESSOAS DE 0 A 14 ANOS NOME	DATA NASC.	IDADE	SEXO	FREQUÊNTA A ESCOLA		OCUPAÇÃO	DOENÇA OU CONDIÇÃO REFERIDA (sigla)
				sim	não		

Siglas para a indicação das doenças e/ou condições referidas

ALC - Alcoolismo
CHA - Chagas
DEF - Deficiência
DIA - Diabetes

EPI - Epilepsia
GES - Gestação
HA - Hipertensão Arterial
TB - Tuberculose

HAN - Hanseníase
MAL - Malária

SITUAÇÃO DA MORADIA E SANEAMENTO

TIPO DE CASA	
Tijolo/Adobe	
Taipa revestida	
Taipa não revestida	
Madeira	
Material aproveitado	
Outro - Especificar:	
Número de cômodos / peças	
Energia elétrica	
DESTINO DO LIXO	
Coletado	
Queimado / Enterrado	
Céu aberto	

TRATAMENTO DA ÁGUA NO DOMICÍLIO	
Filtração	
Fervura	
Cloração	
Sem tratamento	
ABASTECIMENTO DE ÁGUA	
Rede pública	
Poço ou nascente	
Outros	
DESTINO DE FEZES E URINA	
Sistema de esgoto (rede geral)	
Fossa	
Céu aberto	

OUTRAS INFORMAÇÕES

Alguém da família possui Plano de Saúde?		Número de pessoas cobertas por Plano de Saúde	
Nome do Plano de Saúde _____			

EM CASO DE DOENÇA PROCURA	
Hospital	
Unidade de Saúde	
Benedeira	
Farmácia	
Outros - Especificar:	
MEIOS DE COMUNICAÇÃO QUE MAIS UTILIZA	
Rádio	
Televisão	
Outros - Especificar:	

PARTICIPA DE GRUPOS COMUNITÁRIOS	
Cooperativa	
Grupo religioso	
Associações	
Outros - Especificar:	
MEIOS DE TRANSPORTE QUE MAIS UTILIZA	
Ônibus	
Caminhão	
Carro	
Carroça	
Outros - Especificar:	

A família é beneficiária do Programa Bolsa Família?		NIS do Responsável	_____
A família está inscrita no Cadastro Único de Programas Sociais do Governo Federal (CAD-Único)?			

OBSERVAÇÕES

ANEXO A

INSTALAÇÃO DO *ANDROID SDK* E *PLUG-IN ADT* NO *ECLIPSE*

INSTALAÇÃO DO ANDROID SDK E PLUG-IN ADT NO ECLIPSE

Para desenvolver aplicações para o *Android*, se faz necessário a instalação dos seguintes *softwares*: o *Android SDK (Software Development Kit)*, um ambiente de desenvolvimento *Java (Eclipse ou NetBeans)* e do *plug-in Android Development Tools (ADT)*.

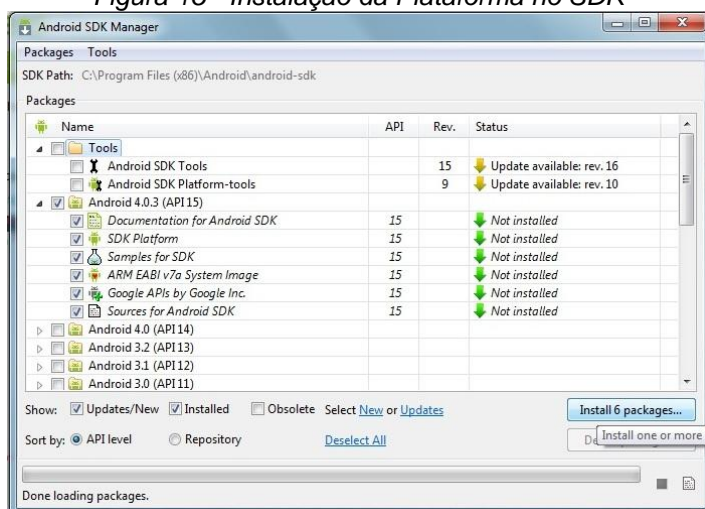
O *Android SDK* fornece as ferramentas necessárias para o desenvolvimento de aplicações na plataforma *Android* usando a linguagem de programação *Java* e possui um emulador para simular o dispositivo físico e testar as aplicações. O *SDK* é suportado nos sistemas operacionais *Windows, Linux e Mac OS*.

O *Plugin ADT* é o responsável pela integração do ambiente de desenvolvimento *Java* com o emulador, onde se torna possível a inicialização do emulador dentro do próprio ambiente de desenvolvimento *JAVA*, instalação automática da aplicação no emulador e depurar o código-fonte como qualquer outra aplicação *Java*.

Instalação do SDK

Para efetuar a instalação do *Android* é necessário primeiro verificar se já existe o *JDK (Java Development Kit)* instalado no computador, visto que o *Android* é compatível com no mínimo o *JDK* versão 5. Após a verificação e instalação do *JDK* (se necessário), deve-se efetuar o *download* do *SDK (ANDROID DEVELOPERS, 2011a)*, em seguida executar o arquivo baixado de acordo com o seu sistema operacional e efetuar a instalação. Ao término da instalação o arquivo *SDK Manager* (dentro da pasta do *Android*) deverá ser executado para instalação da Plataforma de Desenvolvimento. Quando iniciado o *SDK Manager* irá atualizar a sua fonte de dados para disponibilizar as plataformas para instalação. Após a atualização, deve-se selecionar a(s) plataforma(s) e clicar no botão “*Instala packages*” conforme a Figura 15, aceitar os termos de licença e em seguida aguarda o término da instalação. No presente trabalho, o *SDK* foi instalado no sistema operacional *Windows 7 Home Premium* e a plataforma instalada foi a 4.0 (*Ice Cream Sanduiche*) por ser a mais recente plataforma própria para *tablets*.

Figura 15 - Instalação da Plataforma no SDK



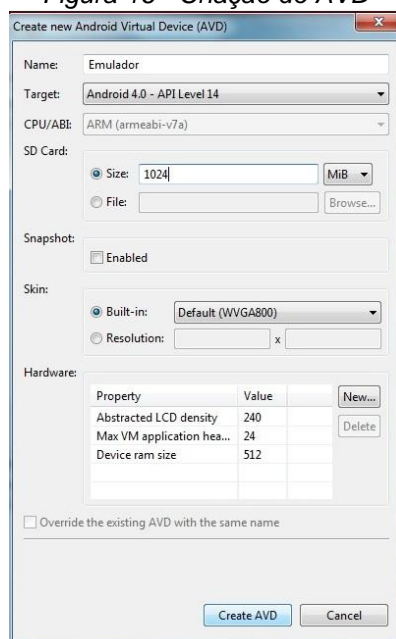
Fonte: (Android SDK Manager)

Criação de um *Android Virtual Device (AVD)*

Para simular a configuração virtual de um dispositivo real é necessário criar um *AVD*, onde é possível simular a aparência de um dispositivo físico com o SO *Android*, aplicações chaves, cartão de memória e entre outros.

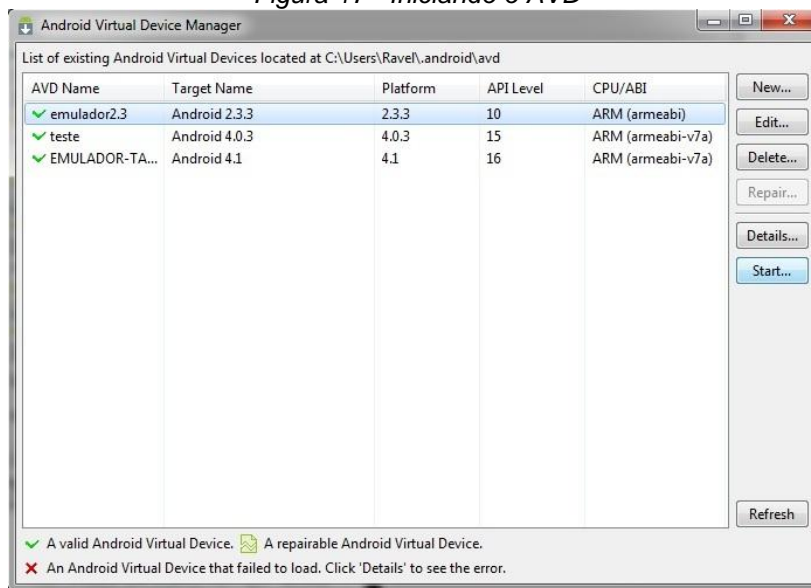
O arquivo *AVD Manager* (dentro da pasta do *Android*) deverá ser executado para iniciar a criação do *AVD*, após executado o arquivo, deve-se clicar no botão “*New*” e preencher os dados conforme a Figura 16 e clicar no botão “*Create AVD*”. Para executar o emulador basta selecioná-lo e clicar no botão “*start*” (Figura 17).

Figura 16 - Criação do AVD



Fonte: (AVD Manager)

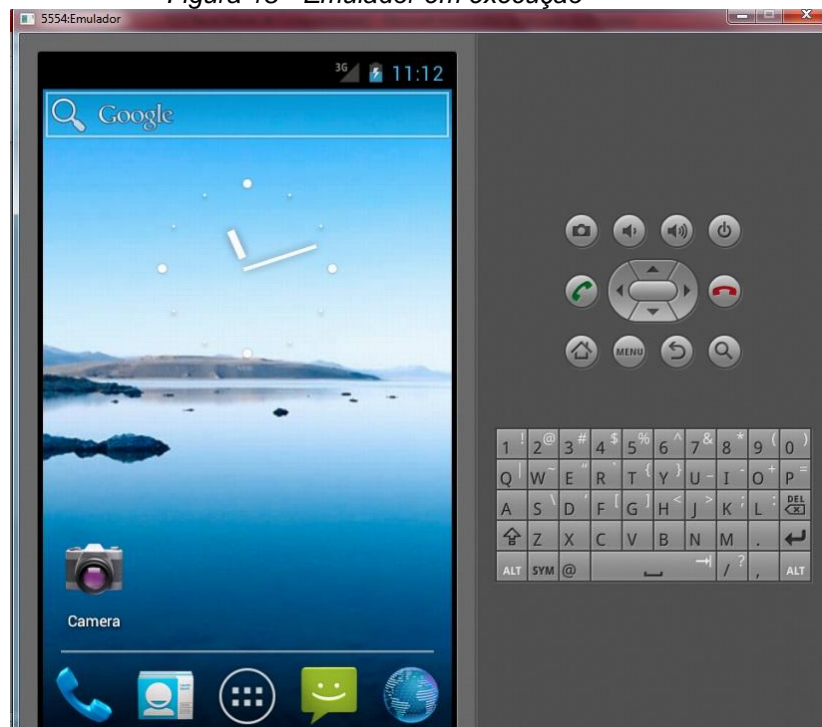
Figura 17 - Iniciando o AVD



Fonte: (AVD Manager)

Se todos os procedimentos anteriores foram executados corretamente o emulador será carregado e exibido conforme a Figura 18, para a versão 4.0 do SO Android.

Figura 18 - Emulador em execução



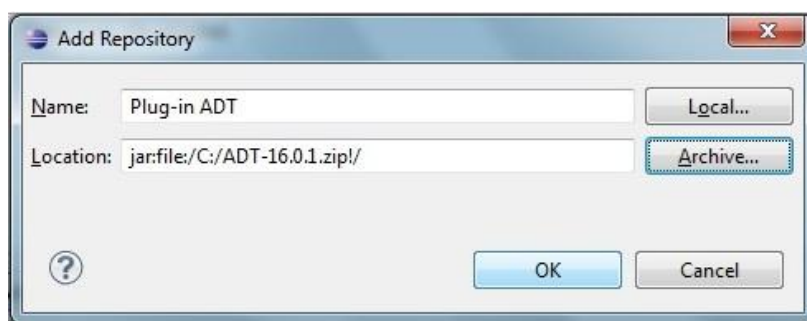
Fonte: (AVD – Emulador 4.0)

Instalação do Plug-in ADT

No desenvolvimento do presente trabalho, foi utilizado o Ambiente de Desenvolvimento *Java Eclipse 3.7* (ECLIPSE, 2011) e o *Plug-in do Android para Eclipse “ADT-16.0.1”* (ANDROID DEVELOPERS, 2011b).

Para se dar início à instalação, o *plug-in* deve ser copiado para a unidade “C:\” e com o *Eclipse* aberto clicar na opção “*Install New Software*” dentro do menu “*Help*”. Após esse procedimento clicar no botão “*Add*”, selecionar o *plug-in* na opção “*Archive*” e preencher os dados conforme a Figura 19 e clicar no botão “*OK*”.

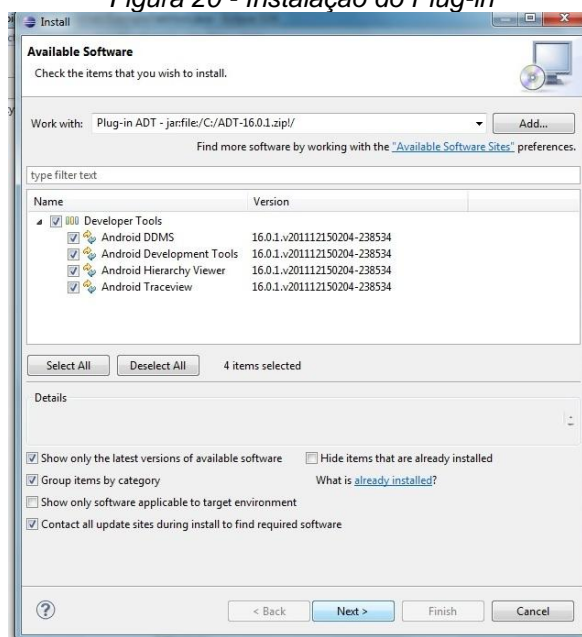
Figura 19 - Selecionando plug-in ADT



Fonte: (Eclipse 3.7)

Na tela seguinte, deve-se marca todas as opções dentro do item “*Developer Tools*”, clicar no botão “*Next*” conforme a Figura 20, logo depois no botão “*Finish*”, aguarde o processo de instalação e em seguida reiniciar o *Eclipse*.

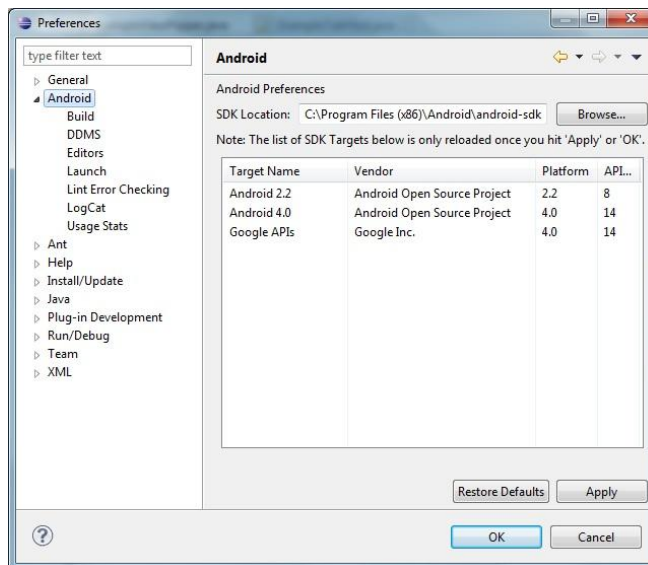
Figura 20 - Instalação do Plug-in



Fonte: (Eclipse 3.7)

Depois de instalado, é necessário configurar o *plug-in* para que o *Eclipse* possa encontrar o emulador e as plataformas instaladas no *SDK*. Para configurar, basta entrar no menu **Window>Preferences** e clicar na opção **Android**. Inserir o caminho da instalação do *SDK*, conforme a Figura 21 e clicar em **OK**.

Figura 21 - Configuração das preferências do Eclipse

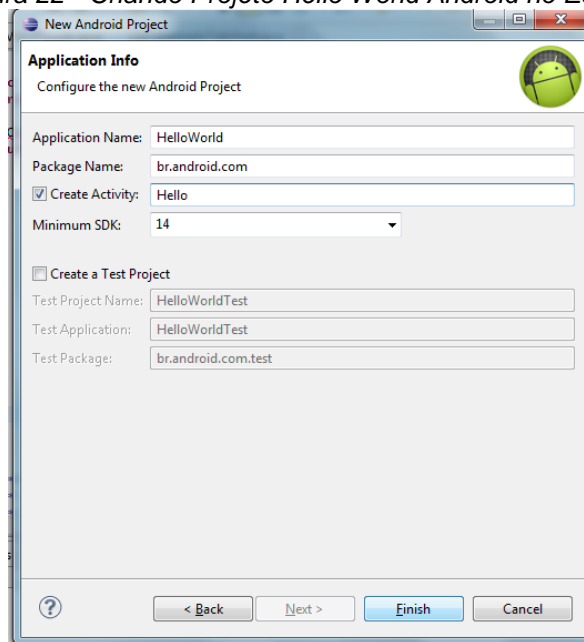


Fonte: (Eclipse 3.7)

Criando e executando o primeiro projeto *Android* no *Eclipse*

Para criar um projeto no *Eclipse* basta entrar no menu **File > New > Project**, selecionar “*Android Project*” e clicar em *Next*. Na tela seguinte preencher o “*Project Name*” com o nome do projeto, como exemplo, “*HelloWorld*”, novamente clicar em *Next*, selecionar a plataforma e na próxima tela preencher os dados, conforme a Figura 22 e clicar em **Finish**.

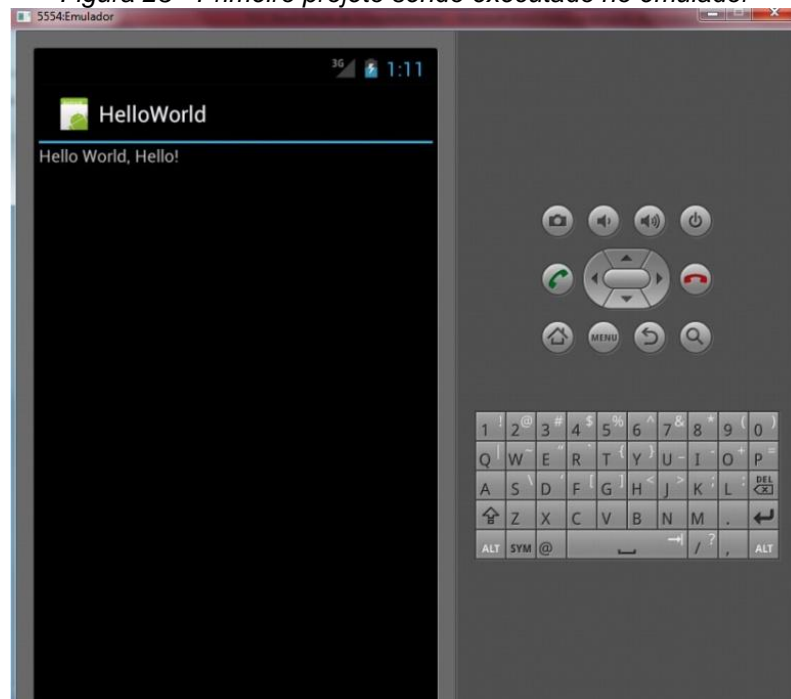
Figura 22 - Criando Projeto Hello World Android no Eclipse



Fonte: (Eclipse 3.7)

Para executar o projeto, basta clicar com o botão direito do mouse, selecionar o menu **Run as > Android Application** e aguardar a execução do projeto. A Figura 23 mostra a aplicação sendo executada no emulador.

Figura 23 - Primeiro projeto sendo executado no emulador



Fonte: (AVD – Emulador 4.0)