



**UNIVERSIDADE ESTADUAL DA PARAÍBA
CAMPUS CAMPINA GRANDE
CENTRO DE CIÊNCIAS E TECNOLOGIA
DEPARTAMENTO DE COMPUTAÇÃO
CURSO DE LICENCIATURA EM COMPUTAÇÃO**

VALDERÍ MEDEIROS DA SILVA

Transferência de dados via SMS utilizando a plataforma Android: uma alternativa para o envio de informações de prontuário médico

CAMPINA GRANDE – PB
2013

VALDERÍ MEDEIROS DA SILVA

Transferência de dados via SMS utilizando a plataforma Android: uma alternativa para o envio de informações de prontuário médico

Monografia apresentada ao Curso de Licenciatura em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de graduado.

Orientador: Prof. Dr. Robson Pequeno de Sousa

CAMPINA GRANDE – PB
2013

FICHA CATALOGRÁFICA ELABORADA PELA BIBLIOTECA CENTRAL-UEPB

S586t Silva, Valderí Medeiros da.
Transferência de dados via SMS utilizando a plataforma Android [manuscrito] : uma alternativa para o envio de informações de prontuário médico / Valderí Medeiros da Silva. – 2013.
55 f. : il. color.

Digitado
Trabalho de Conclusão de Curso (Graduação em Computação) – Universidade Estadual da Paraíba, Centro de Ciências e Tecnologia, 2013.
“Orientador: Prof. Dr. Robson Pequeno de Sousa, Departamento de Matemática, Estatística e Computação”.

1. Telemedicina. 2. Android. 3. SMS. 4. Aplicativos. 5. Prontuário médico. I. Título.

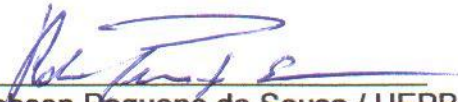
21. ed. CDD 004

VALDERÍ MEDEIROS DA SILVA

Transferência de dados via SMS utilizando a plataforma Android: uma alternativa para o envio de informações de prontuário médico

Monografia apresentada ao Curso de Licenciatura em Computação da Universidade Estadual da Paraíba, em cumprimento à exigência para obtenção do grau de graduado.

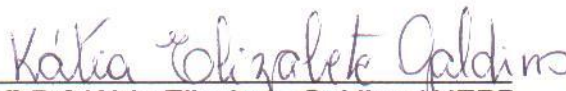
Aprovada em 08/08/2013.



Prof. Dr. Robson Pequeno de Sousa / UEPB
Orientador



Prof. Dr. Misael Moraes / UEPB
Examinador



Profª Drª Kátia Elizabete Galdino / UEPB
Examinadora

AGRADECIMENTOS

Antes de qualquer coisa, a Deus por tão grande amor que através de Jesus Cristo pode me alcançar, me trazendo vida e vida em abundância, além da certeza da salvação.

A minha querida esposa Tatiane, a qual esteve sempre ao meu lado, me dando todo incentivo e apoio necessários, se mostrando muito compreensiva nos momentos de ausência.

Ao meu pai Francisco e a minha mãe Lucia, os quais sempre desprenderam todo esforço e recursos necessários para que eu pudesse ter uma educação de qualidade.

Ao professor Dr. Robson Pequeno, por todo apoio e empenho, os quais fizeram o desenvolvimento desta pesquisa menos árduo.

Aos colegas pesquisadores, Luiz e Adson, os quais auxiliaram bastante neste trabalho.

Por fim, a todos aqueles colegas, amigos e parentes, que direta ou indiretamente contribuíram, não apenas no desenvolvimento desta pesquisa, mas na minha formação como pessoa, coisa que levarei para o resto da minha vida.

R E S U M O

Este trabalho de conclusão de curso tem como principal objetivo mostrar a viabilidade do uso do SMS na transferência de dados de prontuário médico, constituindo-se um aparato de grande valia, para um sistema de telemedicina de baixo custo e extremamente eficiente, e como proposto no presente trabalho, possibilitar a remoção de um paciente cardíaco atendido por uma equipe médica de urgência, em uma localidade bastante remota apenas quando realmente se fizer necessário. Uma vez, que um médico receberá os dados do prontuário médico do paciente, enviado através do sistema, enviará sua resposta com o diagnóstico de necessidade de remoção do paciente ou não, resposta essa que chegará até a equipe desta localidade remota. Para isso desenvolveu-se um protótipo de software, cuja arquitetura deste sistema proposto é composta por três subsistemas: Aplicativo da Equipe, Servidor SMS e aplicativo do médico. Sistema esse, que se mostra como sendo uma alternativa frente aos métodos mais convencionais de envio que se dão, em sua maioria, lançando mão de tecnologias teoricamente mais velozes como wifi, fibra Óptica, etc. Muito embora, estas não possibilitem uma mobilidade tal, requerida por alguns tipos de sistemas. Ou ainda tecnologias de acesso móveis como o GPRS. EDGE ou 3G, limitada ainda ou pelo valor cobrado e/ou pela cobertura disponibilizada.

.

PALAVRAS-CHAVE: Telemedicina. Prontuário Médico. SMS. Android.

A B S T R A C T

This course conclusion work has as main objective to show the feasibility of using SMS data transfer of medical records, constituting an apparatus of great value, for a telemedicine system for low cost and extremely efficient, and as proposed in this work, allowing the removal of a cardiac patient treated by a medical emergency in a fairly remote location only when you really do need. Once, a doctor will receive the data from the patient's medical record, sent through the system, send your answer with the diagnosis of the patient's need for removal or not, a response that the team will reach this remote location. To do this we developed a software prototype, whose architecture of this proposed system consists of three subsystems: Application Team, SMS Server and Application doctor. This system, shown as an alternative compared to more conventional methods of transmission that occur, mostly resorting to theoretically faster technologies such as WiFi, Fiber Optics, etc.. Although, they do not allow such mobility, required by some types of systems. Or even mobile access technologies such as GPRS. EDGE or 3G, or even limited by the amount charged and / or made available by coverage.

.

KEYWORDS: Telemedicine. Medical Record. SMS. Android.

LISTA DE FIGURAS

Figura 1 - Arquitetura da plataforma Android	Erro! Indicador não definido.
Figura 2 - Dados passados durante o envio de um SM.	Erro! Indicador não definido.
Figura 3 - Interface móvel do Pésinet.....	Erro! Indicador não definido.
Figura 4 - Interface Web do Pésinet.....	Erro! Indicador não definido.
Figura 5 - FrontLineSMS Mobile.....	Erro! Indicador não definido.
Figura 6 - FrontLineSMS para Computador	Erro! Indicador não definido.
Figura 7 - Tela do Ushadidi 1	Erro! Indicador não definido.
Figura 8 - Tela do Ushadidi 2	Erro! Indicador não definido.
Figura 9 - Desenho arquitetural do sistema	Erro! Indicador não definido.
Figura 10 - Diagrama de Componentes do Protótipo ...	Erro! Indicador não definido.
Figura 11 - Interface do Aplicativo da Equipe.....	Erro! Indicador não definido.
Figura 12 - Menus do aplicativo da equipe.....	Erro! Indicador não definido.5
Figura 13 - Botão gravar aplicativo da equipe	Erro! Indicador não definido.6
Figura 14 - Mensagem para Prontuário cadastrado com sucesso	Erro! Indicador não definido.6
Figura 15 - Ícones de status "Não Enviado", "Enviado" e "Resposta Recebida", respectivamente.	Erro! Indicador não definido.7
Figura 16 - Realização do envio de SMS.	Erro! Indicador não definido.7
Figura 17 - Registro com status de "Recebido"	39
Figura 18 - Opção "Ler Resposta" exibida após clicar no registro	39
Figura 19 - Visualização do retorno do especialista.	39
Figura 20 - Tela Inicial do aplicativo do Médico.....	40
Figura 22 - Tela de Resposta do aplicativo do médico.....	40
Figura 23 - Download do ADT Bundle	46
Figura 24 - Seleção do Sistema Operacional do ADT Bundle	46
Figura 25 - Site Oracle para download da JDK.....	47
Figura 26 - Tela de seleção da JDK.....	47
Figura 27 - Tela 1 da instalação da JDK	48
Figura 28 - Tela final da instalação da JDK	48
Figura 29 - Tela de criação de um device.....	49
Figura 30 - Processo de Instalação de um aplicativo no Android	50

LISTA DE SIGLAS

OHA	<i>Open Handset Alliance</i>
SMS	Short Message Service
SMSC	Short Message Service Center
CFM	Conselho Federal de Medicina
PEP	Prontuário Eletrônico do Paciente

SUMÁRIO

1. INTRODUÇÃO	10
1.1. Descrição do Problema.....	11
1.2. Motivação e Justificativa.....	11
1.3. Objetivos	11
1.3.1. Objetivo Geral.....	12
1.3.2. Objetivo Específico	12
1.3.3. Organização do documento	12
2. Fundamentação Teórica	13
2.1. ANDROID.....	13
2.1.1. Breve Histórico	13
2.1.2. A plataforma Android.....	14
2.1.3. Arquitetura.....	15
2.1.4. Kernel.....	15
2.1.5. Bibliotecas	16
2.1.6. <i>RunTime</i>	17
2.1.7. <i>Framework</i> de Aplicações	18
2.1.8. Aplicações no Android.....	18
2.1.9. Componentes de Aplicação.....	19
2.1.10. Desenvolvimento	20
2.1.10.1. Requisitos de Sistema	21
2.2. SMS	22
2.2.1. Aplicações referências no envio de dados via SMS	23
2.2.1.1. Pésinet	23
2.2.1.1.1. Funcionamento.....	23
2.2.1.2. FrontLineSMS	25
2.2.1.2.1. Funcionamento.....	25
2.2.1.3. Ushadidi	26
2.2.1.3.1. Funcionamento.....	26
2.2.1.4. Uso de SMS para coletar informações de hemorragia pós-parto na zona rural de Ghana	27
2.2.1.4.1. Funcionamento.....	27
2.3. Prontuário Médico.....	28
2.4. Prontuário Eletrônico do Paciente (PEP).....	30
3. METODOLOGIA	31
3.1. Instrumentos de Coleta de Dados.....	31
3.2. Equipamentos, Materiais e Ferramentas	32
4. PROTÓTIPO DO APLICATIVO.....	33
4.1. Protótipo de Aplicativo para Equipe Remota.....	34
4.2. Servidor SMS	39
4.3. Aplicativo do Médico	40
5. RESULTADOS E DISCUSSÕES.....	41
6. CONCLUSÃO E TRABALHOS FUTUROS	42
APÊNDICE I - Configurando ambiente de desenvolvimento Android – ADT	46
APÊNDICE II - Instalação de uma aplicação no Android	50
ANEXO I.....	51

1. INTRODUÇÃO

Ao passo que as tecnologias de uma maneira geral avançam, os dispositivos móveis, principalmente os smartphones e tablets, se tornam mais poderosos, em termos de poder de processamento, armazenamento, comunicação e ainda mais acessíveis para os diversos segmentos da sociedade. Estima-se que até o primeiro trimestre de 2012, existiam cerca de 6,1 bilhões de aparelhos celulares no mundo¹.

Se por um lado, usuários comuns desejam aparelhos com mais recursos como câmera, GPS, Bluetooth, internet, TV digital, ótima interface gráfica, jogos e muito mais, por outro lado, no mercado corporativo, o qual busca agilizar seus negócios e maximizar lucros, tem se buscado cada vez mais incorporar a mobilidade ao dia-a-dia para atingir tais objetivos (Lecheta, 2010).

Na área de saúde, mais do que nunca, se busca incorporar os avanços tecnológicos aos processos e procedimentos desta, visando ser mais efetivos no atendimento, já que se trabalha direta ou indiretamente com vidas.

Pensando nisso, o presente trabalho se propõe a mostrar a viabilidade prática do uso de SMS para transferências de dados de prontuários médicos, lançando mão da implementação de um protótipo de software funcional, com foco na plataforma Android.

No levantamento bibliográfico, serão abordados temas como: Android, Prontuário Médico, Prontuário Eletrônico, Funcionamento da tecnologia SMS, Aplicações referências no uso de SMS, isso tudo com a finalidade de embasar o desenvolvimento do trabalho.

Quanto à metodologia, trata-se de uma pesquisa aplicada, com perquisição bibliográfica, de natureza qualitativa. A Coleta de dados foi realizada através de entrevista não diretiva, que foram realizadas com o próprio orientador do trabalho, realizadas na coordenação do curso de computação da própria instituição, a fim de levantar os requisitos funcionais e não funcionais do sistema.

¹ Fonte: <http://www.teleco.com.br/pais/celular.asp>

1.1. Descrição do Problema

Atualmente, em termos de soluções relacionadas à telemedicina, tem ocorrido um grande avanço, e a cada dia surgem novas soluções, as quais são implementadas visando além de diminuir custos, agilizar o processo de atendimento ao paciente.

O problema consiste no fato de que, quando mais avançada uma solução voltada à telemedicina for, mais recursos essa irá requer, e o que se tem é que nem todas as cidades, principalmente as do interior, conseguem canalizar tantos recursos para este fim, e mesmo que estes recursos estejam disponíveis, o que muitas das vezes acontece é que inexistem as tecnologias que são pré-requisitos para que estas soluções sejam aplicadas. Assim, faz-se necessário o desenvolvimento de soluções eficiente, de baixo custo associado, e que preferencialmente faça uso de tecnologias já presentes nessas localidades.

1.2. Motivação e Justificativa

O professor Dr. Robson Pequeno de Sousa, com o qual já vinha desenvolvendo pesquisas de Iniciação Científica passou a indicar dentre outros assuntos, este que é tema deste TCC, o qual devido a importância que apresenta, principalmente no âmbito social e da saúde pública, se optou por pesquisar sobre a viabilidade de realizar a transferência de dados de prontuários eletrônicos fazendo uso de SMS.

O Trabalho de Conclusão de Curso se justifica porque além de se constituir como uma atividade científica da graduação, cuja exigência é um requisito essencial e obrigatório para integralização curricular, à comprovação de tal viabilidade e a possível implementação de um sistema destes, se constituirá sem sombra de dúvidas como um aparato de grande valia para que a utilização da telemedicina esteja presente também nos pequenos centros, reduzindo custos e possivelmente auxiliando os diversos sistemas de saúde no processo de salvamento de vidas

1.3. Objetivos

Nesta seção serão abordadas as intenções em relação ao problema da pesquisa, ou seja, serão definidos os objetivos gerais e específicos. Os objetivos

gerais definem o que se pretende alcançar com a realização desta pesquisa. Já os objetivos específicos abordam os passos percorridos para alcançar os objetivos gerais.

1.3.1. Objetivo Geral

O objetivo geral desse trabalho é verificar a viabilidade do uso de SMS para transferência de dados de Prontuário médico lançando mão do desenvolvimento de um protótipo de software funcional, abordando os diversos subsistemas necessários para a concepção deste.

1.3.2. Objetivo Específico

Os objetivos específicos do sistema para este TCC são:

- 1- Definir os requisitos do software, tanto funcionais como não funcionais;
- 2- Definir a arquitetura do software;
- 3- Implementar os subsistemas envolvidos no protótipo de software funcional.

1.3.3. Organização do documento

Este Trabalho de Conclusão de Curso está organizado da seguinte maneira: a Seção 1 introduz o tema da pesquisa. A Seção 2 apresenta a fundamentação teórica dos temas que norteiam a análise e projeto de software. A Seção 3 aborda a metodologia aplicada para a elaboração deste projeto. Na Seção 4, são apresentados especialmente os detalhes relacionados ao protótipo de software. Na Seção 5, são apresentados os resultados e discussão. Por fim, na Seção 6, são apresentados as considerações finais e trabalhos futuros.

2. Fundamentação Teórica

2.1. ANDROID

2.1.1. Breve Histórico

Visando atender tanto o anseio dos usuários comuns, sejam estes os mais exigentes ou não, do mercado corporativo e ainda facilitar o desenvolvimento de aplicações para dispositivos móveis, é que em 5 de novembro de 2007, o OHA anunciou a plataforma Android. A *Open Handset Alliance* é grupo de mais de 50 empresas líderes em tecnologias em geral, mas em especial tecnologias móveis, as quais compartilham das mesma visão, que é mudar a experiência de mobilidade dos consumidores (OHA, 2012a) e manter uma plataforma-padrão onde tenhamos todas as novas tendências do mercado em uma única solução (Lecheta, 2010).

Faz parte da aliança OHA, um grupo bastante diversificado de empresas, no qual incluem operadoras de telefonia móvel, fabricantes de aparelhos, companhia de semicondutores, companhias de softwares e empresas de comércio (OHA, 2012b). A seguir está a lista dessas empresas:

- Operadoras de Telefonia Móvel



- Fabricantes de Aparelhos



- Companhia de Semicondutores



- Companhia de Software



- Empresas de Comércio



2.1.2. A plataforma Android

Segundo a OHA (2012c), Android é uma pilha de software para dispositivos móveis, que inclui um sistema operacional, middleware² e aplicações chave³. Para Lecheta (2010), o Android é uma plataforma para smartphones, baseada no sistema operacional Linux, possui diversos componentes, com uma variada disponibilidade de bibliotecas e interface gráfica, além de disponibilizar ferramentas para a criação de aplicativos.

O Android permite a configuração de vários componentes da plataforma, inclusive o próprio sistema operacional, uma vez que sua arquitetura é muito flexível e possibilita a integração de aplicações nativas com aplicações de terceiros, ou ainda substituir qualquer aplicação nativa existente por uma aplicação desenvolvida, por quem quer que seja.

O fato de possuir um sistema operacional baseado no Linux é de fato mais um ponto forte dessa plataforma, uma vez que este se encarrega de gerenciar memória e os processos. Tal fato possibilita que diversas aplicações possam ser executadas ao mesmo tempo, permitindo que aplicações em segundo plano possam ser

² Também conhecido com mediador, é um software que faz a mediação entre software e as demais aplicações, independentemente de protocolos de comunicação, linguagem, etc.

³ Aplicativos mais comumente encontrados em dispositivos móveis como discador, lista de contatos galeria de imagens, tocador de música, etc.

executadas sem que o usuário perceba, enquanto ele está realizando qualquer outra tarefa (LECHETA, 2010).

2.1.3. Arquitetura

Silva (2009, p.11) afirma que é importante conhecer a arquitetura da plataforma, seus principais componentes, bibliotecas e subsistema, para poder identificar as limitações e dependências na criação de um aplicativo para dispositivo móvel.

A Figura 1 ilustra como está organizada a arquitetura da plataforma Android, e nas seções subseqüentes será dado um enfoque mais aprofundado de cada parte que compõe essa arquitetura:

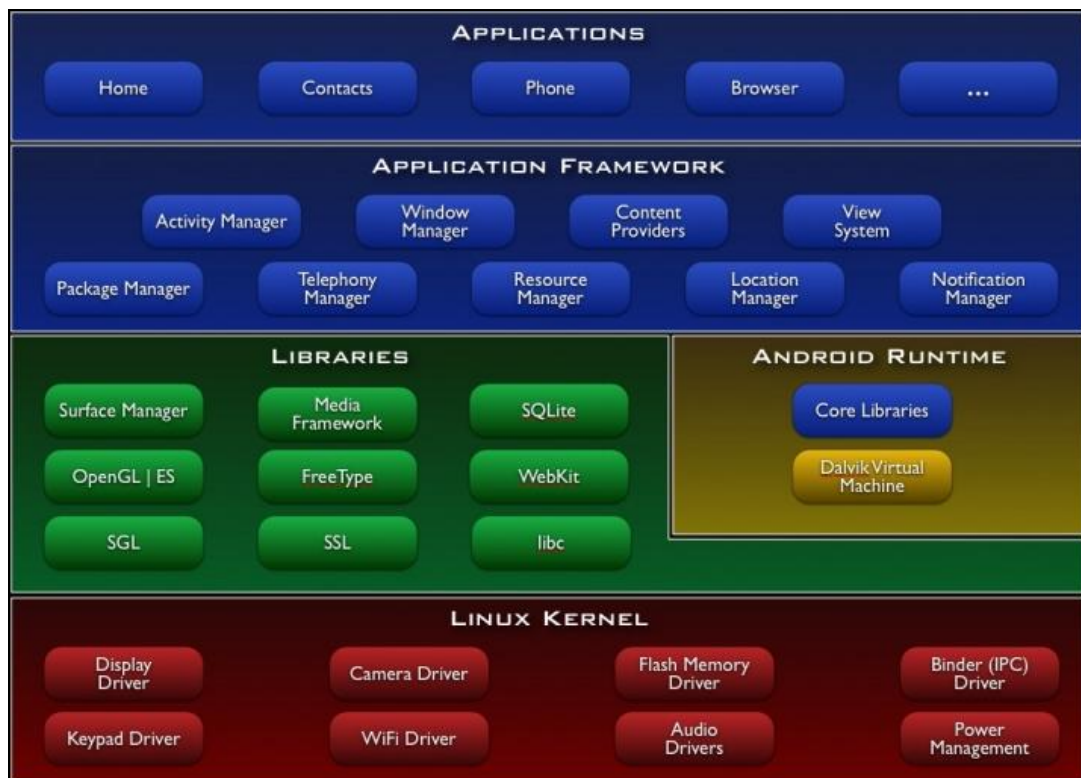


Figura 1 - Arquitetura da plataforma Android

2.1.4. Kernel

O kernel do Linux em sua versão 2.6 é o que compõe a base da arquitetura, esse como já foi falado (sessão 2.1.2), é responsável por gerenciar memória e processos, *threads*, segurança de arquivos e pastas, rede e *drivers*. A gerência de *drivers* se constitui um fato importantíssimo, uma vez que faz com que o desenvolvedor não necessite se preocupar em como acessar ou gerenciar dispositivos específicos do aparelho, proporcionando assim, uma espécie de abstração entre *hardware* e o *software*.

No Android, cada aplicativo dispara um processo no sistema operacional. Estes aplicativos possuem uma interface gráfica para o usuário ou não, neste último caso o aplicativo roda em segundo plano, podendo ficar nesse estado indefinidamente.

2.1.5. Bibliotecas

Assim como afirma Silva (2009, p.13), durante o desenvolvimento de programas, existem algumas rotinas básicas que muitas vezes são comuns na maioria dos *softwares*, como abrir arquivos, realizar cálculos aritméticos, que não são o foco principal do desenvolvedor, assim, visando evitar a reescrita dos códigos de funções e rotinas que se repetem com certa frequência ou possuem altas complexidades, existem as bibliotecas.

A plataforma Android inclui um conjunto de bibliotecas C/C++ utilizadas por vários componentes do sistema. Algumas das principais bibliotecas são listadas a seguir:

Surface Manager

Biblioteca que controla e gerencia o acesso ao subsistema de display. Compõe transparentemente camadas gráficas 2D e 3D a partir de aplicações de múltiplas aplicações.

3D libraries

Utiliza aceleração por *hardware* na medida do possível, além de renderização 3D via *software*, neste caso altamente otimizado. Esta biblioteca é baseada em *OpenGL ES 1.0*

Media libraries

Baseada em *OpenCORE*, essas bibliotecas suportam playback e gravação de muitos formatos de áudio e de vídeo, bem como imagens estáticas, incluindo MPEG4, H.264, MP3, AAC, AMR, JPG e PNG.

FreeType

É uma biblioteca gerenciadora de fonte, a qual facilita a renderização de fontes *TrueType*, otimizada para diversos tipos de plataforma.

SQLite

Biblioteca C que implementa um banco de dados SQL embarcado. SQLite não é uma biblioteca de cliente usada para conectar com um servidor de banco de dados. SQLite é o servidor. A biblioteca SQLite lê e escreve diretamente para e do arquivo do banco de dados no disco (AQUINO, 2007 p.6)

LibWebCore

Utilizada também no Safári, navegador da Apple, esta biblioteca é a base para o navegador do Android e visões da web.

System C Library

Uma vez que a plataforma Android é baseada no Linux, é necessário que essa possua uma biblioteca que defina as chamadas ao sistema.

2.1.6. RunTime

A linguagem base para o desenvolvimento de aplicações é Java, no entanto, no Android não existe uma máquina virtual Java (JVM). O que se tem é uma máquina virtual chamada Dalvik, a qual é otimizada para a execução em dispositivos

móveis. Assim, após a compilação os bytecode (.class) são convertidos para o formato .dex (Dalvik Executable), o qual representa a aplicação Android compilada, após isso todos os arquivos .dex e demais recursos da aplicação como imagens, etc., são compactados em um único arquivo com a extensão .apk (Android Package File), que é a aplicação final (LECHETA, 2010).

2.1.7. *Framework* de Aplicações

A arquitetura do Android foi projetada visando simplificar o reuso dos componentes, onde qualquer componente pode publicar suas capacidades e quaisquer outros componentes podem então fazer uso de tais capacidades, muito embora haja restrições de segurança reforçadas pelo *framework*, o que permite que componentes sejam modificados em tempo de desenvolvimento (AQUINO, 2007, p.7).

O *framework* de aplicações facilitam a implementação dos programas, ao passo que permite a criação de combos, botões, campos de textos, menus, etc. Dentre os principais componentes temos:

- Gerenciador de atividade (*Activity Manager*), o qual é responsável por gerenciar o ciclo de vida das aplicações e permitir sua execução em segundo plano.
- Gerenciador de notificações (*Notification Manager*), que permite que qualquer aplicação possa exibir alertas na barra de status, ou ainda reagir à notificações recebidas.
- Provedores de conteúdo (*Content Providers*), cuja função é permitir que aplicações acessem os dados de outras aplicações ou compartilhem os seus próprios dados.

2.1.8. Aplicações no Android

Após ter seus arquivos compilados pelo SDK⁴ do Android, todos os arquivos e recursos são compactados em outro arquivo com extensão .apk, o qual representa a

⁴ Sigla para **Software Development Kit**, ou seja, Kit de Desenvolvimento de Software.

própria aplicação a ser distribuída e instalada (ver Subseção 2.1.6), cada aplicativo é executado em um processo Linux, tendo cada um sua própria máquina virtual, nesse caso uma máquina virtual *Dalvik*.

O sistema operacional atribui em tempo de execução um ID único a cada aplicativo (isso é feito de maneira transparente), dessa maneira um processo não se apresenta, por padrão, visível para outros processos, no entanto, é possível fazer com que dois aplicativos possa compartilhar o mesmo ID, fazendo com que eles possam acessar arquivos e trocar dados entre si, além do mais pode ser que eles executem até na mesma máquina virtual no intuito de otimizar a utilização de recursos do sistema.

Cada aplicação, por padrão, tem acesso somente aos componentes que ela necessita para sua execução e nada mais. Isso cria um ambiente muito seguro em que um aplicativo não pode acessar partes do sistema para o qual não é dada a permissão, porém este pode solicitar permissão para acessar dados do dispositivo, como contatos do usuário, mensagens SMS, câmera, Bluetooth e muito mais, tal permissão deve ser concedida pelo usuário no momento da instalação (Google Inc. 2012).

2.1.9. Componentes de Aplicação

Para a construção de uma aplicação Android existem quatro componentes ou blocos principais que são Activity (Atividade), Intent Receiver (Receptor de Intenção), Service (Serviço) e Content Provider (Provedor de Conteúdo).

Segundo Aquino (2007), nem toda aplicação necessita fazer uso desses quatro componentes ou blocos, mas uma aplicação Android deverá ser escrita com um ou mais desses blocos de construção. Assim, uma vez decidido quais componentes são necessários à aplicação, é necessário listá-los em um arquivo chamado `AndroidManifest.xml`⁵.

⁵ Este arquivo apresenta informações essenciais sobre a aplicação e requisitos, para o sistema operacional, informações estas que o sistema deve receber antes de executar qualquer aplicação.

Activity (atividade) é um componente visual, usualmente uma tela ou janela em uma aplicação Android. Segundo Lecheta (2010) a Activity tem a responsabilidade de controlar os eventos da tela e definir qual View será responsável por desenhar a interface gráfica do usuário. São exemplos de Views: campos texto, barras de rolagem, combos, menus e muitos outros elementos, os quais podem ser desenhados em uma tela e estes agrupados podem constituir uma tela complexa.

Service (serviço) é um componente executado em background, que não dispõe de interface gráfica e pode se comunicar com uma aplicação através de uma interface de serviço.

Intent Receptor (receptor de intenção) é usado quando uma determinada aplicação deve reagir a um evento externo como, por exemplo, uma mensagem, uma chamada, etc. Este tipo de componente não apresenta uma interface gráfica, porém eles podem usar o gerenciador de notificações para informar ao usuário quando algum evento interessante ocorrer (AQUINO, 2007). Para que um *intent receptor* seja chamado, não é necessário que este esteja executando, uma vez que o sistema operacional se encarrega de chamá-lo e abrir aplicação correspondente.

Content Provider (provedor de conteúdo) é um tipo de componente responsável por fazer com que um conjunto de dados que estejam nas formas de arquivos, banco de dados do SQLite, estejam disponíveis para uma aplicação, assim possibilitando a abstração, por parte dos aplicativos, no que diz respeito a parte de armazenamento e gerenciamento de informações e dados (SILVA, 2009, p.19).

2.1.10. Desenvolvimento

Um *Framework* de desenvolvimento é uma estruturação de um conjunto de códigos, classes, funções e métodos, construída de uma maneira tal, que sirva de suporte à criação de uma dada linguagem ou plataforma.

O desenvolvimento de aplicações Android pode ser realizado a partir das mesmas ferramentas utilizados no desenvolvimento de aplicativos Java. O Android oferece no seu *Framework* de desenvolvimento, de um lado as principais bibliotecas já conhecidas e utilizadas no desenvolvimento de aplicativos Java, de outro as

bibliotecas específicas da plataforma para criação de aplicações móveis (SILVA, 2009, p.15).

2.1.10.1. Requisitos de Sistema

Segundo Google Inc. (2012a), os requisitos de sistema no desenvolvimento de aplicações para a plataforma Android são os que seguem:

Os sistemas operacionais suportados são:

- Windows XP (32-bit), Vista (32-bit ou 64-bit), ou Windows 7 (32- ou 64-bit)
- Mac OS X 10.5.8 ou superior (x86 apenas)
- Linux (testado no Ubuntu Linux, Lucid Lynx)
 - GNU C Library (glibc) 2.7 ou superior é requerida.
 - No Ubuntu Linux é requerida a versão 8.04 ou superior.

Os ambientes de desenvolvimento suportados são:

- Eclipse IDE⁶
 - Eclipse 3.6.2 (Helios) ou superior.
 - Eclipse JDT plugin (incluído na maioria dos pacotes de distribuição da IDE Eclipse).
 - JDK 6 (apenas a JRE não é suficiente)
 - Android Development Tools (ADT) plugin (recomendado)
 - Não compatível com Gnu Compiler for Java (gcj)
- Outros ambientes de desenvolvimento ou IDE
 - JDK 6 (apenas a JRE não é suficiente)

⁶ Um dos ambientes de desenvolvimento integrado (IDE) mais utilizados no mundo, tanto para programação Java como de outras linguagens de programação. Criado pela IBM em novembro de 2001 e suportado por conjunto de fornecedores de software (Eclipse Foundation, 2012).

- Apache Ant 1.8 ou superior
- Não compatível com Gnu Compiler for Java (gcj)

2.2. SMS

SMS (Short Message Service) ou serviço de mensagens curtas é um serviço provido pela rede de telefonia celular e que a cada dia que passa torna-se mais popular e utilizado.

Em sua versão tradicional, um SM (Short Message) é essencialmente uma mensagem de texto enviada de um aparelho celular ou de uma aplicação para outro aparelho celular. O número máximo de caracteres é em geral de 160 caracteres, porém, assim como afirma BROWN (2012), esse número diz respeito a um sistema que utiliza codificação com 7 bits, que é o caso dos caracteres latinos, já se tivermos caracteres não-latinos como Chineses e Arábicos, por exemplo, poderão ser utilizados 8 bits, o que faz como que se possa usar apenas 140 caracteres por mensagem, ou 16 bits onde teríamos apenas 70 caracteres.

Segundo DELALANDE (2012) no processo de envio de um SM são necessárias três entidades:

- **Fonte:** pode ser uma aplicação que está enviando a mensagem ou o próprio usuário através do aparelho celular.
- **Plataforma SMSC (Short Message Service Center):** o funcionamento dessa plataforma se dá através do modelo SFE (Store & Forward Engine), ou seja, armazenas os SMS em uma fila, verifica a disponibilidade de recebimento da mensagem pelo dispositivo de destino. Caso o aparelho de destino esteja disponível o SM é encaminhado, caso contrário, este é recolocado na fila para que seu envio seja re-tentado posteriormente.
- **Switch Celular:** é uma central responsável pelo processamento das chamadas e tem a função de entregar o SM e proceder com as informações necessárias para o SMSC.

Um exemplo dos dados que realmente são enviados, na maioria das vezes criptografados por medida de segurança ou por outra necessidade qualquer, está na figura :

Transfer protocol data unit: 07917283010010F5040BC87238880900F10000993092516195800AE8329BFD4697D9EC37	
Octet(s)	Description
07	Length of the SMSC information (in this case, 7 octets).
91	Type-of-address of the SMSC (91 means international format of the phone number).
72 83 01 00 10 F5	Service center number (in decimal semi-octets). The length of the phone number is odd (11), so a trailing F has been added to form proper octets. The phone number of this service center is "+27381000015."
04	First octet of this SMS-DELIVER message.
0B	Address length. Length of the sender number (0B hexadecimal = 11 decimal).
C8	Type-of-address of the sender number.
72 38 88 09 00 F1	Sender number (decimal semi-octets), with a trailing F ("+27838890001").
00	Protocol identifier (00 = SME-to-SME protocol—implicit).
00	Data coding field (00 = 7 bit, 01 = 8 bit, 10 = 16 bit, 11 = reserved).
99 30 92 51 61 95 80	Time stamp (semi-octets) in order (YY, MM, DD, HH, MM, SS, TIMEZONE in relation to GMT in units of 15 minutes). So, 0x99 0x30 0x92 0x51 0x61 0x95 0x80 means 29 Mar 1999 15:16:59 GMT+2.
0A	User data length: length of message. The data coding field indicated 7-bit data, so the length here is the number of septets (10). If the data coding field were set to indicate 8-bit data or Unicode, the length would be the number of octets (9).
E8329BFD4697D9EC37	Message "hellohello," 8-bit octets representing 7-bit data.

Figura 2 - Dados passados durante o envio de um SM.

2.2.1. Aplicações referências no envio de dados via SMS

2.2.1.1. Pésinet

2.2.1.1.1. Funcionamento

Agentes vão até a casa das famílias e realizam a pesagem das crianças. Os resultados são enviados via SMS, através de GPRS, pelo agente de pesagem para um médico local, que os recebe instantaneamente. Caso os dados apontem sinais de desnutrição o médico requisita uma visita para mãe e filho (a) e/ou recomenda um tratamento se assim for necessário.

Os dados também são armazenados nos aparelhos celulares através de uma aplicação Java, que foi pensada para ser fácil de baixar, instalar e executar, o fato de termos dados nos aparelhos celulares facilita o acesso e atualizações das informações dos pacientes.

A Figura 3 ilustra a interface móvel do Pésinet:



Figura 3 - Interface móvel do Pésinet

As informações coletadas são disponibilizadas através de uma interface web com duas características principais: a pré-seleção e sinalização de casos anormais e geração de registros médicos.

Coordenadores do programa podem administrar o serviço e acompanhar a coleta e o processo de inscrição, através da base de dados que armazena as informações trafegadas. Relatórios de atividade gerados automaticamente facilitam o acompanhamento remoto de cada nível de gestão. Os dados são agregados, e estatísticas exibidas podem ser filtradas em filtros específicos (por programa, por região, por país).

A Figura 4 ilustra o comportamento da interface Web do Pésinet:

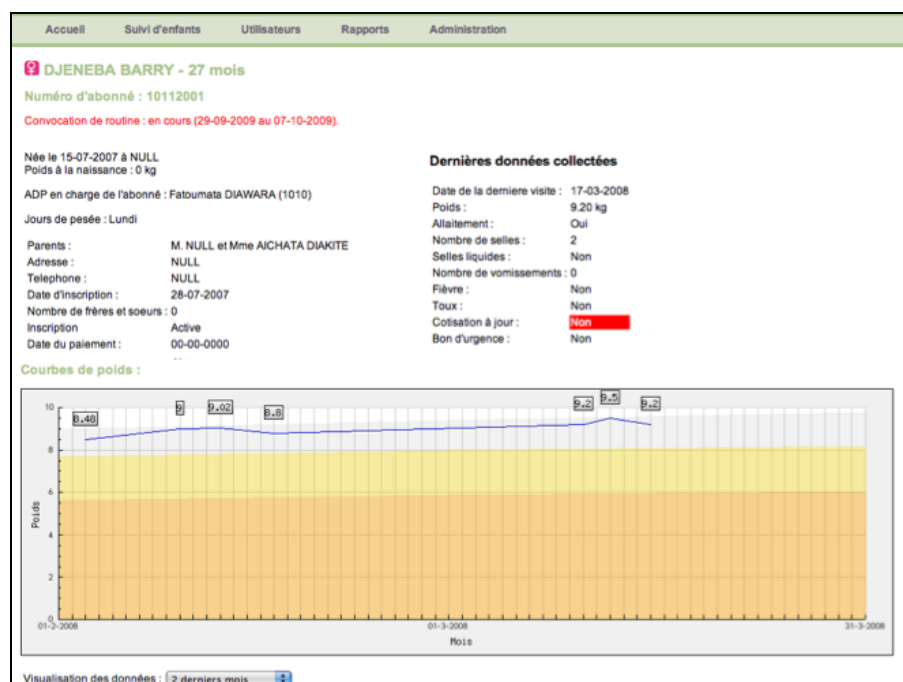


Figura 4 - Interface Web do Pésinet

2.2.1.2. FrontLineSMS

2.2.1.2.1. Funcionamento

O software pode ser usado para fins variados, mas todos com foco na coleta de dados e envio a distância. Por ser de fácil utilização, no caso aplicação Java para celulares, o sistema possibilita que pessoas sem tanta instrução possa fazer tal coleta.

A aplicação Java utiliza as bibliotecas básicas de aplicações móveis, fazendo com que aparelhos celulares simples possam rodar a aplicação. As principais funcionalidades dessa aplicação Java, é poder configurar um número de aparelho celular, para funcionar como o “servidor de dados”, o qual receberá os dados coletados e enviados.

É possível fazer o download de novos formulários que estejam disponíveis, permitindo que haja além de flexibilidade no modelo de formulário para cada fim específico, atualização dos mesmos, caso seja necessário. Os dados enviados também ficam armazenados no aparelho que as envia.

A Figura 5 mostra a tela do FrontLineSMS móbile:

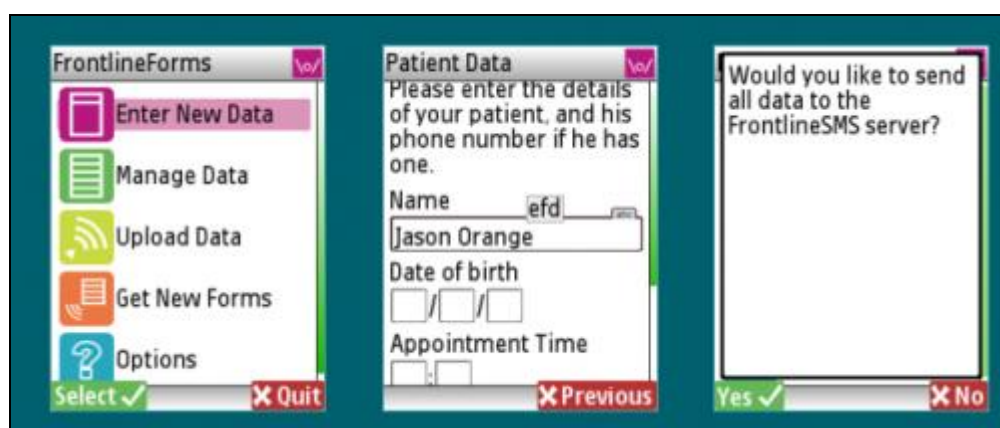


Figura 5 - FrontLineSMS Mobile

O FrontlineSMS dispõe de uma aplicação para PC's, a qual serve com o servidor e que é responsável por receber as mensagens SMS, e por fim envia as respostas quando assim for necessário.

O módulo conta com um gerenciador de contatos, banco de dados, gerenciador de email, possibilidade de se configurar uma resposta automática caso receba na mensagem SMS alguma das palavras-chave configuradas.

A aplicação para PC ainda dispõe de uma funcionalidade para criação de novos formulários. Atualmente está se fazendo a integração do FrontlineSMS com o openMRS⁷ que faz com que se tenha acesso aos dados dos pacientes, como histórico médicos.

A Figura 6 ilustra a versão do FrontLineSMS para computador:

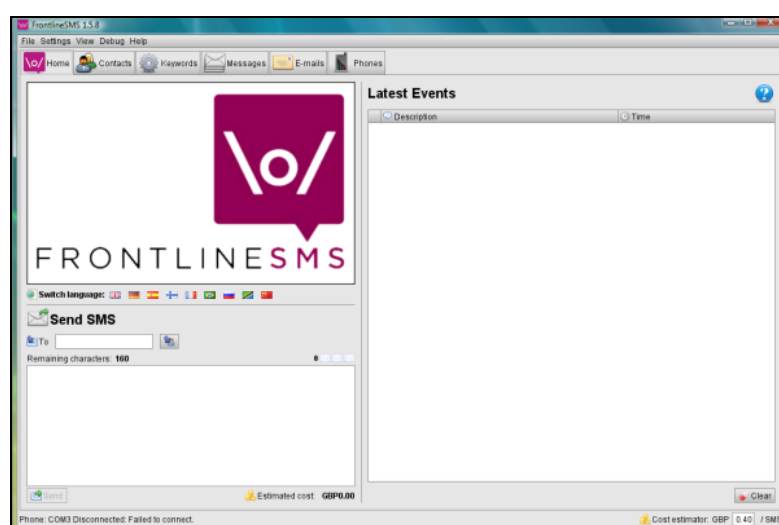


Figura 6 - FrontLineSMS para Computador

2.2.1.3. Ushadidi

2.2.1.3.1. Funcionamento

Diferentemente das outras aplicações, o Ushahidi muita embora possa ser usado para área médica, é uma aplicação de âmbito geral, e consiste basicamente na construção de um mapa de informações sobre a situação de uma determinada região nos contextos mais variados. Os dados são alimentados via SMS, twitter, emails e formulários que as pessoas queiram enviar.

⁷ É um projeto de código aberto colaborativo com o intuito de apoiar a prestação de serviços de saúde em países em desenvolvimento.

A plataforma pode ser usada, por exemplo, para traçar um mapa do estado de devastação após uma catástrofe, número de incidência de uma determinada epidemia em uma região, etc.

As Figura 7 e Figura 8 ilustram o Ushadidi:

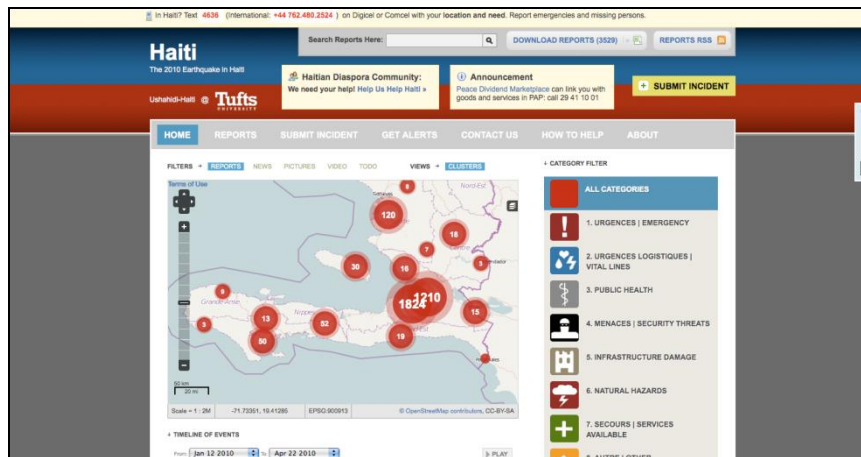


Figura 7 - Tela do Ushadidi 1

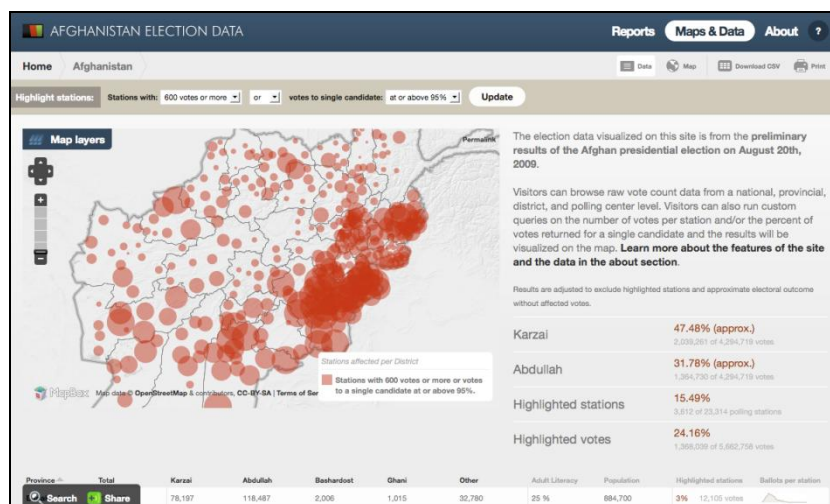


Figura 8 - Tela do Ushadidi 2

2.2.1.4. Uso de SMS para coletar informações de hemorragia pós-parto na zona rural de Ghana

2.2.1.4.1. Funcionamento

O início do projeto se deu quando foram treinadas parteiras para poderem reconhecer sintomas de hemorragia pós-parto e operar aparelhos GSM básicos,

reportando informações via SMS, com as respostas de um formulário predefinido, através de um “protocolo” específico.

Como exemplo de um SMS enviado pelas parteiras, reportando as informações seria:

0422110013

Onde:

04- Identificador do aparelho celular.

22- Idade da paciente.

1- Confirmação de hemorragia pós-parto.

1- Uso de compressão uterina bimanual.

0- Não houve morte no parto.

0- Não houve morte neonatal.

1- Foram feitos cuidados com pré-natal.

3- Foram realizadas três visitas durante o pré-natal.

O sistema usado além da facilidade de uso e eficácia para a que foi destinado, alguns fatores bastante importante é o baixo custo de funcionamento do sistema e abrangência (mobilidade), cobrindo até mesmo a zona rural.

Apesar das vantagens apresentadas acima, o sistema se porta bem frente às necessidades apresentadas, porém se mostra bastante limitado caso queira se trabalhar com informações um pouco mais complexas.

2.3. Prontuário Médico

De acordo com Mota (2005, *apud* Dias, 2007), a palavra *prontuarium* é a correspondente no latim para a palavra prontuário, e se remete ao lugar onde se guardam as coisas que podem ser necessárias a qualquer instante.

O Conselho Federal de Medicina – CFM define prontuário médico como:

[...]o documento único constituído de um conjunto de informações, sinais e imagens registradas, geradas a partir de fatos, acontecimentos e situações sobre a saúde do paciente e a assistência a ele prestada, de caráter legal, sigiloso e científico, que possibilita a comunicação entre membros da equipe multiprofissional e a continuidade da assistência prestada ao indivíduo (CFM, 2002).

Contrastando com a nomenclatura usada pelo CFM, Novaes (2003, *apud* Dias, 2007), o prontuário do “paciente” já vem sendo a nomenclatura usada há algumas décadas para denominar os prontuários, no intuito principal de fazer jus aos vários profissionais que nele inserem informações.

Assim, Cirqueira (2008) define o prontuário do paciente como:

Um registro de acontecimentos ocorridos com o paciente, mantendo toda a sua história referente aos seus cuidados de saúde, desde sua entrada em uma unidade hospitalar (recepção/internação), até a sua saída (alta/óbito), ou seja, desde o seu nascimento até a morte.

Levando em conta funções que o prontuário deveria desempenhar, a padronização destes, leva em consideração que:

- a) O prontuário é o registro legal das ações médicas;
- b) Deve servir de apoio ao processo de atenção à saúde, servindo de fonte de informação clínica e administrativa para tomada de decisão e meio de comunicação compartilhado entre todos os profissionais;
- c) Deve apoiar a pesquisa (estudos clínicos, epidemiológicos, avaliação da qualidade); e
- d) Deve promover o ensino e gerenciamento dos serviços, fornecendo dados para cobranças e reembolso, autorização dos seguros, suporte para aspectos organizacionais e gerenciamento do custo (MARIN, MASSAD E AZEVEDO NETO (2003 *apud* DIAS,2007)).

Cirqueira (2008) afirma que diversas modificações ocorreram nos prontuários do paciente, uma vez que por muitos séculos foram usados registros em papeis, o que facilitava a deterioração e perdas das informações, nos últimos anos com o advento da tecnologia, mais precisamente no setor de saúde, o prontuário tem se tornado mais objetivo e seguro.

2.4. Prontuário Eletrônico do Paciente (PEP)

Para IOM (1997, *apud* Dias, 2007), o prontuário eletrônico do paciente é o registro eletrônico inserido em um sistema projetado para apoiar os usuários, o qual fornece acesso a um completo conjunto de dados corretos, alertas, sistemas de apoio à decisão e outros recursos como links para bases de conhecimento médico.

Kmeteuk filho (2003, *apud* Cirqueira, 2008) afirma que o PEP surgiu na intenção de melhorar a eficiência e organização de guarda das informações de saúde, prometendo acima de tudo substituir os prontuários em forma de papel, e por intermédio de novos recursos e aplicações, elevar a qualidade da prestação de serviço.

As principais funcionalidades do sistema de prontuário eletrônico listadas por Zanetta, Wolynech e Madeira (2003, *apud* Dias, 2007) são:

- a. Os registros clínicos dos usuários podem ser mantidos on-line, em uma central de dados, que guarda todas as informações de todas as atividades realizadas do paciente;
- b. As informações incluem dados demográficos, diagnósticos, procedimentos, imunizações, encaminhamentos, medicamentos receitados e ministrados, internações, exames realizados, atendimentos médicos e multiprofissionais;
- c. Os dados dos pacientes podem ser localizados por diversas fontes, pelo nome, pela identificação do número do cartão SUS e outros dados de identificação;
- d. Apresentam resumos dos atendimentos guardados nos prontuários e relatórios; e. Mantém os arquivos de procedimentos realizados e do Código Internacional de Doenças (CID);
- f. As requisições dos exames são feitas on-line, os resultados recebidos on-line e registrados no prontuário do paciente;
- g. Registra as medicações indicadas e ministradas e envia as informações à farmácia, gerenciando os estoques;
- h. Gerência dados dos profissionais de saúde, que incluem o código de controle, nome, especialidades, identificação do registro do conselho ao qual pertence, a atividade que executa, endereço e número de telefone;
- i. Automatização do faturamento dos hospitais

Temos que em uma comparação com os prontuários em papel, os PEP:

[o Prontuário Eletrônico do Paciente] pode proporcionar inúmeras vantagens, entre elas: o acesso rápido; a disponibilidade remota; o uso simultâneo por várias pessoas; a legibilidade absoluta; a redução do espaço de armazenamento; maior confiabilidade, segurança e confidencialidade das informações por meio da utilização de senhas digitais; a extinção de pilhas de papéis que muitas vezes sofrem deteriorações, acarretando a perda de dados e informações devido às más condições de acondicionamento. (MOTA, 2005, *apud* DIAS, 2007).

3. METODOLOGIA

Este trabalho se caracteriza como sendo uma pesquisa aplicada, com perquisição bibliográfica, de natureza qualitativa, pois, visa à produção e implementação de um protótipo de software para transferência de dados de prontuário médico via SMS através da plataforma Android, o qual se prestará como um sistema de telemedicina.

Optou-se pela metodologia de desenvolvimento de protótipo, uma vez que, é possível "(...) demonstrar conceitos, experimentar opções de projetos e, geralmente, conhecer mais sobre os problemas e suas possíveis soluções." (SOMMERVILLE, 2007). O tipo de prototipagem utilizada foi o exploratório, na qual várias versões de um sistema podem ser criadas para validar requisitos e testar tecnologias (PRESSMAN, 2006).

Os objetivos dessa pesquisa foram atendidos, utilizando-se de um conjunto de procedimentos abaixo descritos.

3.1. Instrumentos de Coleta de Dados

Foi utilizado para a coleta dos dados o instrumento das entrevistas não diretivas (informais ou baseada no discurso livre do entrevistado), onde:

Pressupõe-se que o informante é competente para exprimir-se com clareza sobre questões da sua experiência e comunicar representações e análises suas, prestar informações fidedignas, manifestar em seus atos o significado que têm no contexto em que eles se realizam, revelando tanto a

singularidade quanto à historicidade dos atos, concepções e idéias. (TEIXEIRA, 2012).

A primeira parte das entrevistas não diretivas foi realizada no departamento de computação da UEPB, com o Professor Robson Pequeno de Sousa. Nestas entrevistas foram realizadas perguntas no sentido de elucidar os requisitos do protótipo de software a ser desenvolvido.

A segunda parte das entrevistas foi realizada no Hospital de Trauma de Campina Grande, com um especialista em problemas cardíacos. Estas tiveram como principal objetivo entender o panorama do atendimento a cardíacos no SUS da Paraíba, bem como agrupar algumas informações necessárias para constituir o prontuário médico do aplicativo da equipe, no sentido de mesmo sendo um protótipo, os campos a serem preenchidos possibilitem o médico que irá receber tais informações realizar um diagnóstico preciso do paciente.

3.2. Equipamentos, Materiais e Ferramentas

Foram utilizados para esta pesquisa, materiais, como: caneta, papel, computador, impressora, modem SMS, modelo MIN210 da Daruma, smartphone com Android, modelo Milestone 2 da Motorola e notebook.

Na Tabela 1 a seguir são listados os principais elementos de software utilizados para o desenvolvimento deste projeto.

Nome do Software	Utilidade	Versão	Arquitetural	Licença
Windows 7 Professional	Sistema Operacional	7	32 bits	OEM
Spring Source IDE (IDE Eclipse acrescida de funcionalidades)	Ambiente de Programação	2.9.2	32 bits	Open Source
Mysql	Bando de dados	5.2	32 bits	Free

SQLite	Bando de dados (Embarcado)	-	32 bits	Free
ADT Plugin	Plugin de Android	16.0.1	32 bits	Open Source
Android SDK	Kit de Desenvolvimento para a plataforma Android	Revision 16	32 bits	Free
Microsoft Word 2010	Editor de texto	2010	32 bits	Trial (60 dias)

4. PROTÓTIPO DO APLICATIVO

Com a finalidade de demonstrar a viabilidade técnica/operacional da transferência de dados de prontuário eletrônico fazendo uso da plataforma Android. O protótipo desenvolvido é composto por três subsistemas, como representado no desenho arquitetural presente na figura 9, os quais serão detalhados na sessões seguintes :

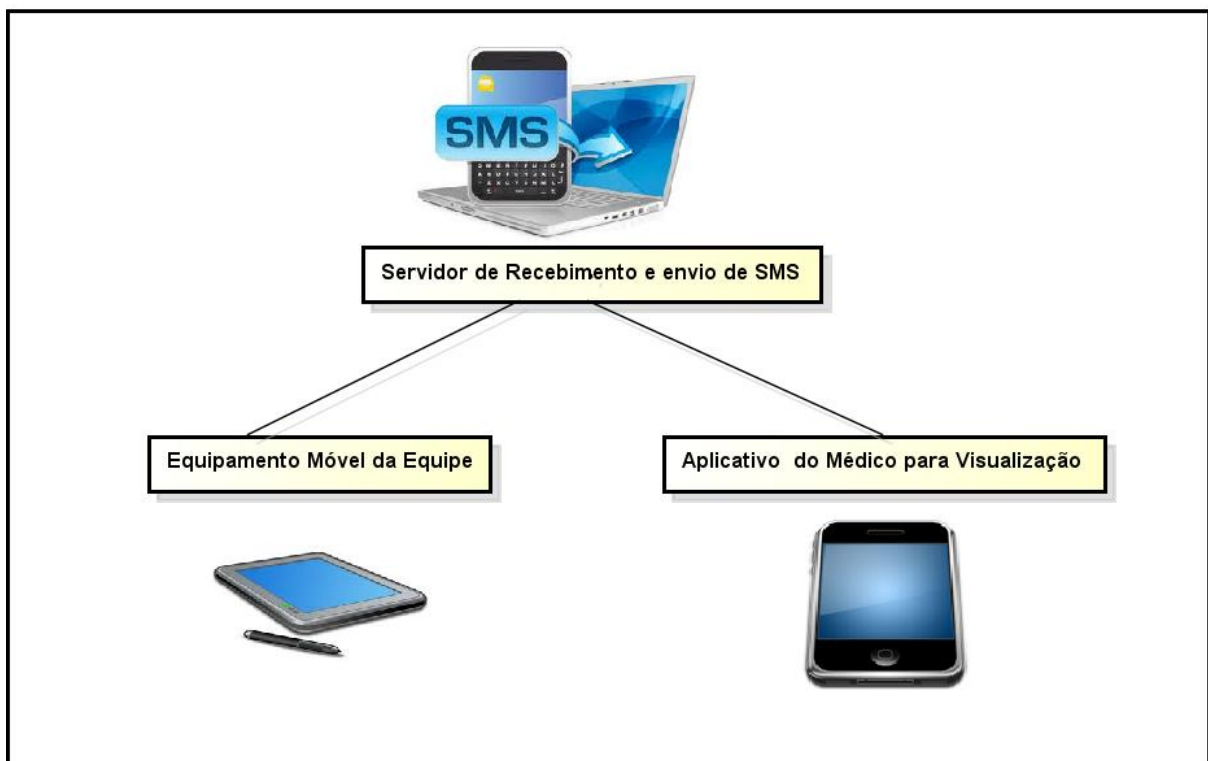


Figura 9 - Desenho arquitetural do sistema

O diagrama de componentes do protótipo é representado na Figura 10:

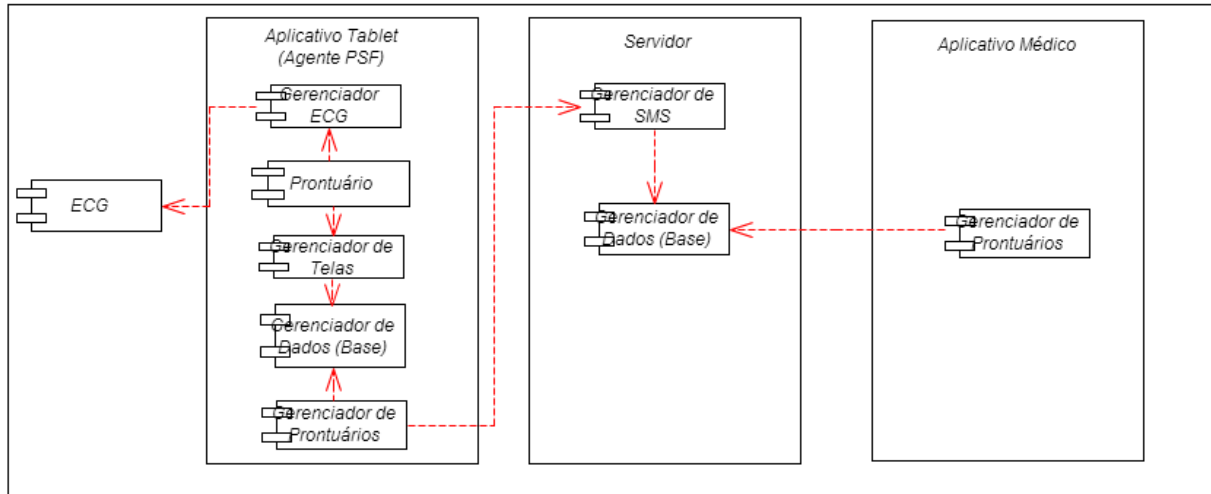
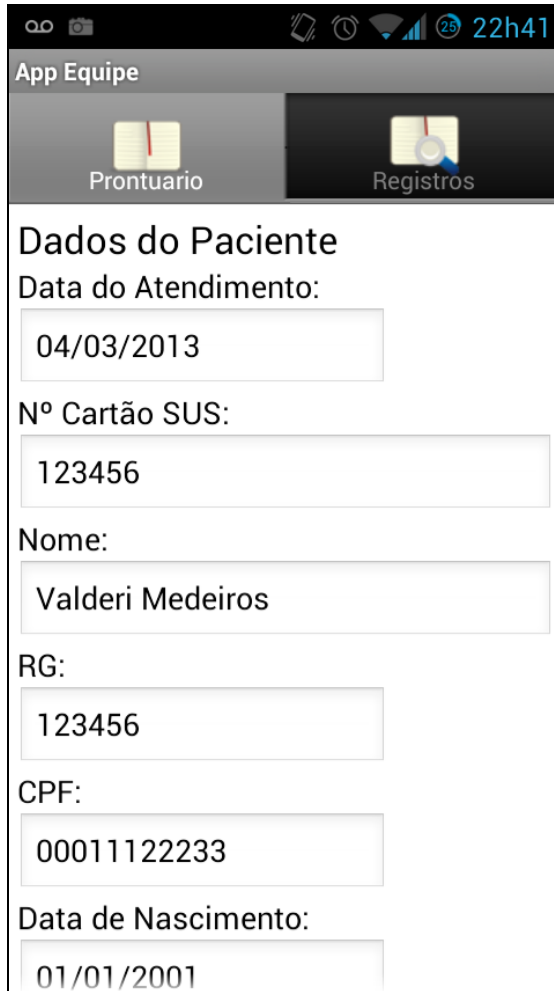


Figura 10 - Diagrama de Componentes do Protótipo

4.1. Protótipo de Aplicativo para Equipe Remota

Através do uso deste aplicativo, a equipe remota poderá realizar exames (neste caso Eletrocardiograma); preenchendo os dados básicos do paciente, o sistema coleta os dados provenientes do ECG, os quais de uma maneira geral constituem um prontuário eletrônico do paciente (PEP).

Na figura 11 é mostrada a interface do protótipo de aplicativo para a equipe remota (uma lista completa dos campos que fazem parte do prontuário, está presente no Anexo I):



App Equipe

Prontuario Registros

Dados do Paciente

Data do Atendimento:

Nº Cartão SUS:

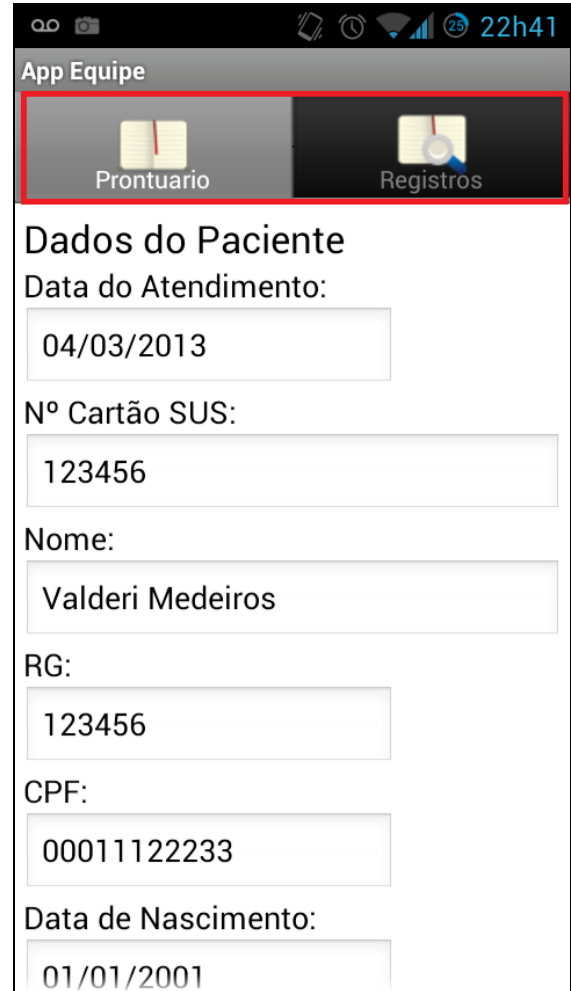
Nome:

RG:

CPF:

Data de Nascimento:

Figura 11 - Interface do Aplicativo da Equipe



App Equipe

Prontuario Registros

Dados do Paciente

Data do Atendimento:

Nº Cartão SUS:

Nome:

RG:

CPF:

Data de Nascimento:

Figura 12 - Menus do aplicativo da equipe

O protótipo é composto por duas abas, “Prontuário” e “Registros”, como pode ser visto na figura 12.

Na aba “Prontuário” os dados do paciente preenchidos pelo agente da equipe de saúde remota, são gravados através do botão “Gravar”, como visto na figura 13:

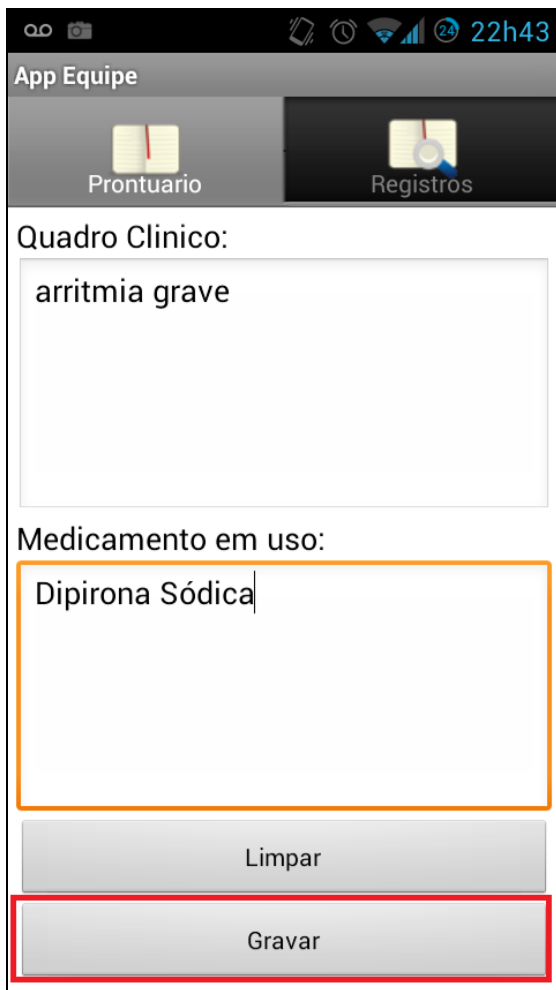


Figura 13 - Botão gravar aplicativo da equipe cadastrado

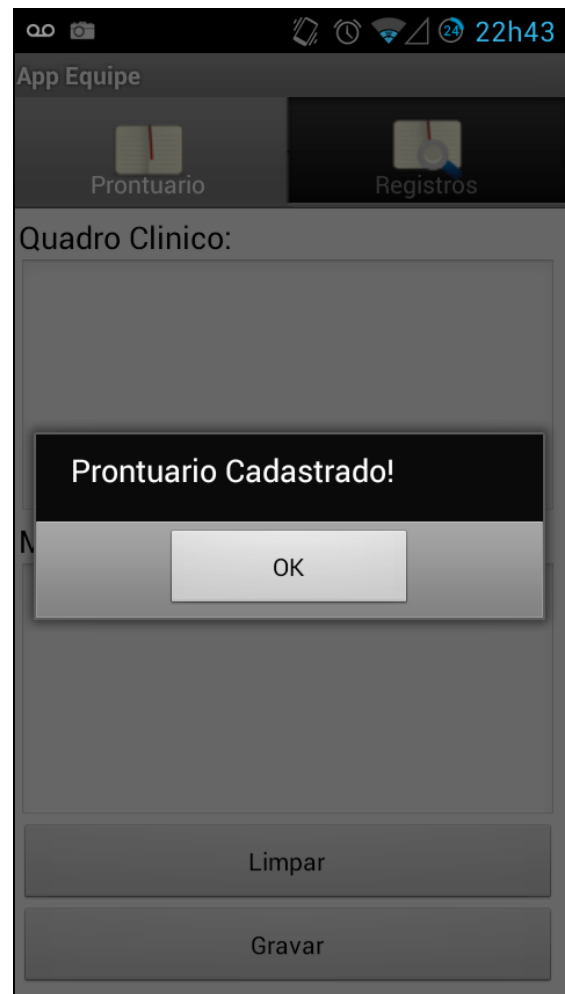


Figura 14 - Mensagem para Prontuário com sucesso

O botão “Limpar” da aba “Prontuário”, como o próprio nome diz, serve para limpar todos os dados digitados, caso queira-se que estes sejam descartados, não havendo possibilidade de recuperação.

Uma vez que os dados são gravados na aba “Prontuário”, na aba “Registros” os dados são exibidos e o seu respectivo status exibido. O status válidos de um registro podem ser: “Não Enviado”, “Enviado” e “Resposta Recebida”, onde cada um possuem um ícone correspondente no registro para tornar a identificação do status mais fácil de ser visualizada. A figura 15 contém os status válidos e os ícones que os representam:



Figura 15 - Ícones de status "Não Enviado", "Enviado" e "Resposta Recebida", respectivamente.

Para a realização do envio dos dados em si, o que se dá através de SMS, basta que o registro ainda não tenha sido enviado, ou seja, esteja com status “Não enviado” e o usuário clique por algum tempo sobre o registro que deseja enviar, aparecendo a opção na tela de “Enviar SMS” (conforme a figura 16), neste momento o aplicativo procede com o envio para o servidor SMS.

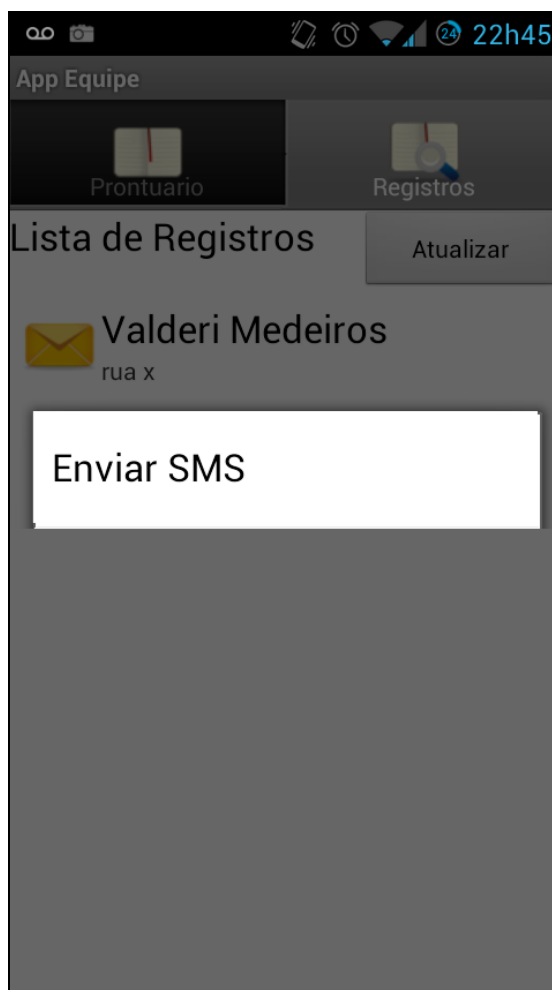


Figura 16 - Realização do envio de SMS.

Os dados enviados pelo aplicativo da equipe são formadas pela junção de todos os campos de dados separados por “|” (pipe). Em seguida, visando manter a segurança dos dados, os dados são codificados em uma string base64. Devido às limitações de caracteres que o uso de SMS acarreta (em geral 140 caracteres), os dados da mensagem são divididos em blocos de 136 caracteres. Cada bloco de 136 caracteres é acrescido na frente do número seqüencial da mensagem com três dígitos; acrescido no final do último bloco um “*” (asterisco) que funcionará como marcador de término da seqüência de mensagens. Segue uma demonstração:

Seguem os dados a serem enviados antes da encriptação:

Primeiro SMS:

```
04012013|12345|Valderi Medeiros|123456|00011122233|rua x|1111|Bairro
novo||cidade nova|paraíba|88888888|valderi@valderi.com|M|80|N1|S|S
```

Segundo SMS:

```
S|N|S|S|N|S|1|12|8|120|Apresenta arritmia
```

Seguem os dados a serem enviados após a encriptação:

Primeiro SMS:

```
001|MDQwMTIwMTN8MTIzNDV8VmFsZGVyaSBNZWRIaXJvc3wxMjM0NTZ8MDAw
MTExMjlyMzN8cnVhIHh8MTEyMjYwYXNjaWw8Y2kYWRlIG5vdmF8cGFy
YcOtYmF8ODg4ODg4ODh8dmFsZGVyaUB2YWxkZXJpLmNvbXxNfDgwfE4xfFN8U
w==
```

Segundo SMS:

```
002|MXxWYWxkZXJpIE1IZGVpcm9zIHxydWEgeHxydWEgeHxydWEgeA==*
```

Ao clicar em atualizar na aba “Registro”, caso o status do registro esteja como “Resposta Recebida”, basta que o usuário clique por alguns instantes no registro, até que seja exibida a opção “Ler Resposta”, então clique nesta para visualizar a resposta enviada pelo médico, o qual recebeu o prontuário enviado pelo aplicativo da equipe e foi repassado pelo servidor SMS, com a ação a ser realizada (por

exemplo: “Remover o paciente”, “Requer acompanhamento” ou “Quadro clínico normal”). As figuras 17, 18 e 19 ilustram esse processo:

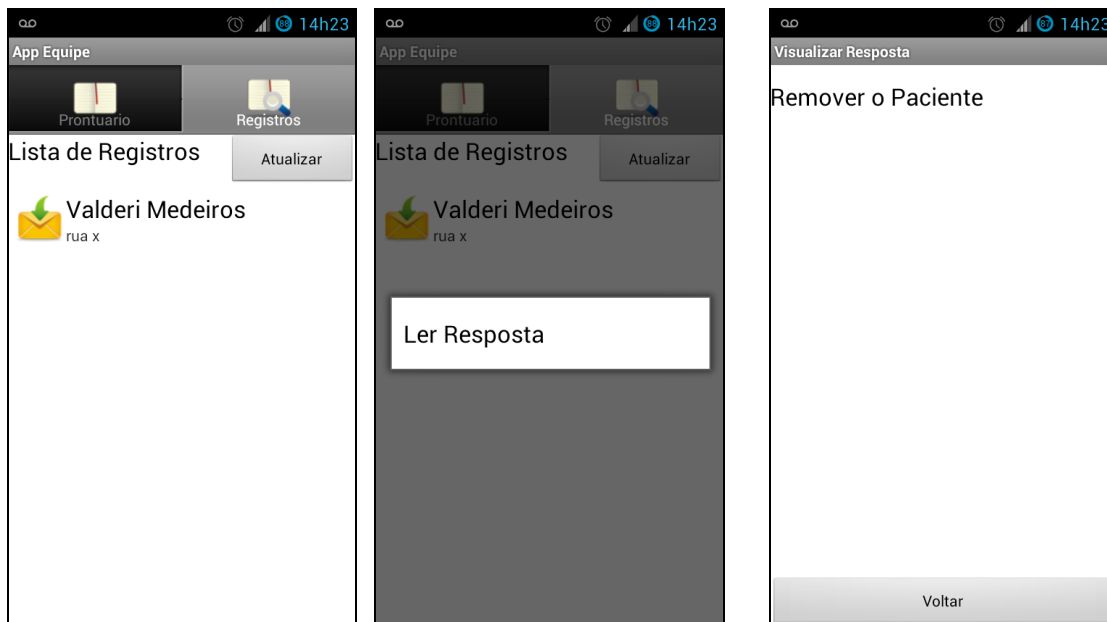


Figura 17 - Registro com status de Visualização do registro **Figura 18 - Opção "Ler Resposta" "Recebido" retorno do especialista.** **Figura 19 - exibida após clicar no**

4.2. Servidor SMS

O servidor SMS é o sistema responsável pelo recebimento do SMS contendo os dados do prontuário eletrônico do paciente, enviados pelo aplicativo da equipe.

Uma vez recebido a sequência de SMS contendo os dados do prontuário, o servidor SMS, vai recebendo cada mensagem e agrupando-as até que todas as mensagens tenham sido recebidas com sucesso (o que pode ser determinado pelo recebimento de uma mensagem contendo o caractere "*" no final da mesma). A chave de descryptografia é conhecida entre a aplicação da equipe e o servidor SMS, já que os dados são enviados criptografados na base 64. Após o processo de descryptografia dos dados, os campos enviados são identificados através do caractere "|" (pipe) e salvos na base de dados para serem acessados pelo aplicativo do médico.

No momento em que o servidor SMS identifica que houve uma resposta por parte do médico, o sistema se encarrega de enviar os dados também via SMS para o aparelho no qual os dados foram enviados, fazendo uso do mesmo processo de montagem/criptação usado na aplicação da equipe. Um exemplo de uma mensagem de retorno pode ser visto abaixo:

Seguem os dados a serem enviados antes da criptação:

Remover o Paciente

Seguem os dados a serem enviados após a criptação:

001|UmVtb3ZlciBvIFBhY2llbnRI*|

4.3. Aplicativo do Médico

Por intermédio do aplicativo do médico, o especialista pode visualizar as informações gravadas na base de dados pelo servidor SMS e assim enviar um diagnóstico baseado em sua análise, o que determinará se o paciente em questão deve ser removido imediatamente, requer acompanhamento ou apresenta um quadro clínico normal, por exemplo. A seguir temos um exemplo de uma resposta enviada pelo especialista e persistida na base de dados:

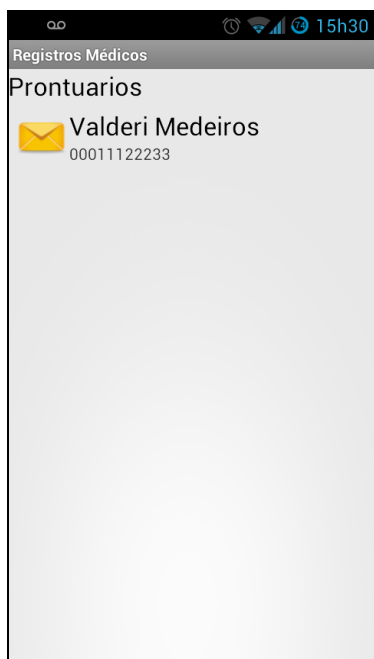


Figura 20 - Tela Inicial do aplicativo do médico

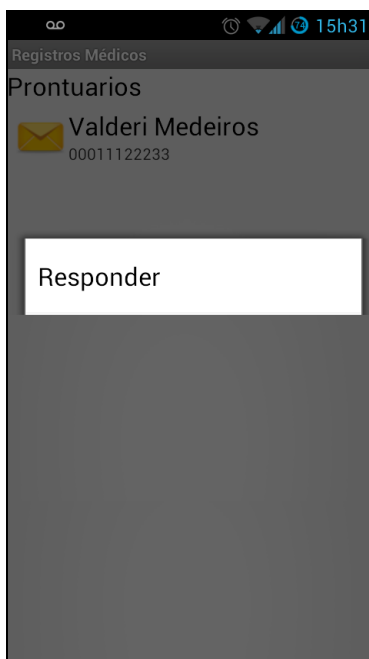


Figura 21 - Seleção da opção "Responder" aplicativo do Médico



Figura 22 - Tela de Resposta do aplicativo do médico

Tanto o acesso aos prontuários médicos enviados pelos aplicativos das equipes, ou o envio da resposta do médico especialista se dão através da conexão com a internet, visando tornar o processo o mais rápido possível.

5. RESULTADOS E DISCUSSÕES

Através das pesquisas realizadas, sobretudo no aporte teórico levantado, bem como no desenvolvimento do protótipo de software, temos como resultados que praticamente toda a cadeia de sistemas e/ou subsistemas (com exceção do envio dos dados do ECG, o qual é detalhado na sessão posterior) funcionou satisfatoriamente.

Foi possível verificar que enviando apenas os dados do prontuário no aplicativo da equipe, em média foram enviados 2 SMS por prontuário preenchido, o que facilitou em muito o processo.

No tocante ao servidor SMS, a performance deste foi um dos fatores que chamou mais atenção, uma vez que, em poucos segundos os dados enviados pelo

aplicativo da equipe eram processados e salvos no banco de dados para consulta do aplicativo do médico.

Com relação ao aplicativo do médico, assim como no aplicativo da equipe, um ponto importante a se destacar foi a facilidade de uso, já que estes não necessitam de menus ocultos, muitos botões ou quantidade de passos exagerados para realizar os procedimentos necessários.

6. CONCLUSÃO E TRABALHOS FUTUROS

Este TCC apresentou a proposta e implementação de um protótipo para envio de dados de prontuário médico utilizando SMS, os detalhes de sua arquitetura e exemplos do seu funcionamento.

O Android mostrou-se uma plataforma robusta, e o desenvolvimento de aplicações para este é em muito facilitado, uma vez que, ele este se dá através do uso da tecnologia Java, sendo uma das linguagens de programação mais utilizadas no mundo, ou seja, o fato de não ter que aprender uma linguagem nova, mesmo que utilizando um paradigma de programação já bastante difundido, faz com que o processo de desenvolvimento seja otimizado com um todo. Além do mais, a vasta documentação própria do Android, simplifica em muito o desenvolvimento e acima de tudo o entendimento do que está de implementando, sem contar os inúmeros exemplos de códigos para servirem como base para diversos segmentos de aplicações.

Ao longo da implementação do protótipo acima citado, o SDK do Android mostrou-se excelente, começando pela instalação que segue processo padrão de instalação de programas no Windows, restando apenas a conexão deste com o plugin ADT⁸ no Eclipse, o que também já é bastante familiar no meio dos usuários desta IDE. O emulador disponível para realização de testes de aplicativos, o qual vem acoplado ao SDK do Android, se mostrou bastante fiel, quando comparado com a execução em um dispositivo móvel real, seja ele um tablet ou smartphone.

⁸ Android Development Tools

A experiência adquirida ao longo do curso, em estudos extracurriculares e principalmente no desenvolvimento de sistemas para o mercado, facilitaram em muito a implementação do protótipo, assim diminuindo a complexidade da realização destas tarefas de uma maneira que pudéssemos ter um protótipo funcional no menor espaço de tempo possível.

Assim, o objetivo principal do presente TCC, foi alcançado, uma vez que temos protótipos funcionais de todos os subsistemas propostos na arquitetura do sistema. A próxima etapa é realizar a comunicação do aplicativo da equipe com o aparelho de Eletrocardiograma, o que não foi possível ao longo do desenvolvimento dos trabalhos, devido a falta de documentação e de casos de sucesso anteriores.

REFERÊNCIAS BIBLIOGRÁFICAS

AQUINO, Juliana França Santos. **Plataformas de Desenvolvimento para Dispositivos Móveis**. 2007. 14 f. Monografia (Graduação) – Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, 2007. p. 3-10.

BROWN Ron; SHIPMAN Bill; VETTER Ron. SMS: The Short Message Service. Disponível em < http://uncw.edu/itsd/documents/Computer-SMS_pdf.pdf >. Acesso em: 24 mai. 2012.

CFM - CONSELHO FEDERAL DE MEDICINA. **Resoluções 1638/2002**. Disponível em: <<http://www.portalmedico.org.br/>>. Acesso em: 29 mai. 2012.

DELALANDE, Christophe. *SMS – Short Message Service*. Disponível em: < http://www.wirelessbrasil.org/wirelessbr/colaboradores/christophe/chris_sms.html>. Acesso em: 24 mai. 2012.

DIAS, Juliana Lopes. **A UTILIZAÇÃO DO PRONTUÁRIO ELETRÔNICO DO PACIENTE PELOS HOSPITAIS DE BELO HORIZONTE**. 2007. 112 f. Dissertação (Mestrado) - Universidade Federal de Minas Gerais, Belo Horizonte, 2007. p. 39-49.

Eclipse Foundation, I. (2012). *About the eclipse foundation*. Disponível em: <<http://www.eclipse.org/org/>>. Acesso em: 17 mai. 2012. .

Google Inc. (2012). *Application fundamentals*. Disponível em: <<http://developer.android.com/guide/topics/fundamentals.html>>. Acesso em: 16 mai. 2012.

Google Inc. (2012a). *System Requirements*. Disponível em: < <http://developer.android.com/sdk/requirements.html>>. Acesso em: 17 mai. 2012.

LECHETA, Ricardo. **Google Android: aprenda a criar aplicações para dispositivos móveis com o Android SDK**. 2 ed. São Paulo: Novatec, 2010.

Microsoft Inc. (2012). *Qual é a plataforma do Microsoft Office que eu tenho?*. Disponível em: < <http://office.microsoft.com/pt-pt/word-help/qual-e-a-plataforma-do-microsoft-office-que-eu-tenho-HA010361023.aspx>>. Acesso em: 17 mai. 2012.

OHA. Open Handset Alliance. Disponível em < http://www.openhandsetalliance.com/oha_overview.html> Acesso em: 14 mai. 2012a.

OHA. Open Handset Alliance. Disponível em < http://www.openhandsetalliance.com/oha_members.html> Acesso em: 14 mai. 2012b.

OHA. Open Handset Alliance. Disponível em <<http://developer.android.com/guide/basics/what-is-android.html>> Acesso em: 14 mai. 2012c.

PRESSMAN, Roger S. **Engenharia de Software 6ª ed**. Makron Books, São Paulo, 2006

SILVA, Luciano Édipo Pereira da. **Utilização da plataforma Android no desenvolvimento de um aplicativo para o cálculo do Balanço Hídrico Climatológico**. 2009. 50 f. Monografia (Graduação) - Universidade Estadual do Mato Grosso do Sul, Dourados, 2009. p. 11-19.

SOMMEVILLE, Ian. **Engenharia de software 8ª edição**. Pearson Education do Brasil, São Paulo, 2007.

Teleco. Celulares no mundo. Disponível em <<http://www.teleco.com.br/pais/celular.asp>>. Acesso em: 14 mai. 2012.

TEIXEIRA, Gilberto. **O QUE É FAZER PESQUISA**. Disponível em <http://www.serprofessoruniversitario.pro.br/m%C3%B3dulos/metodologia-da-pesquisa/o-que-%C3%A9-fazer-pesquisa#.UOnc_uS5Plc>.

APÊNDICE I - Configurando ambiente de desenvolvimento Android – ADT

Realizar o download do ADT Bundle, acessando o site Android Developers⁹.

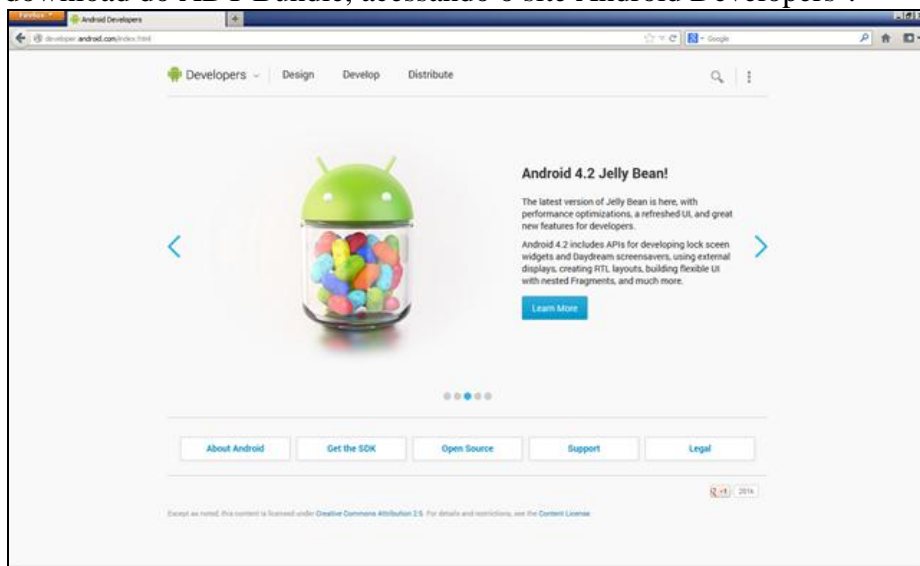


Figura 23 - Download do ADT Bundle

Acesse o link *Get the SDK*. Haverá o redirecionamento para a página de download do ADT Bundle. Clique no botão *Download the SDK*. Na próxima página aceite os termos do Android SDK, e escolha a versão do seu sistema operacional.

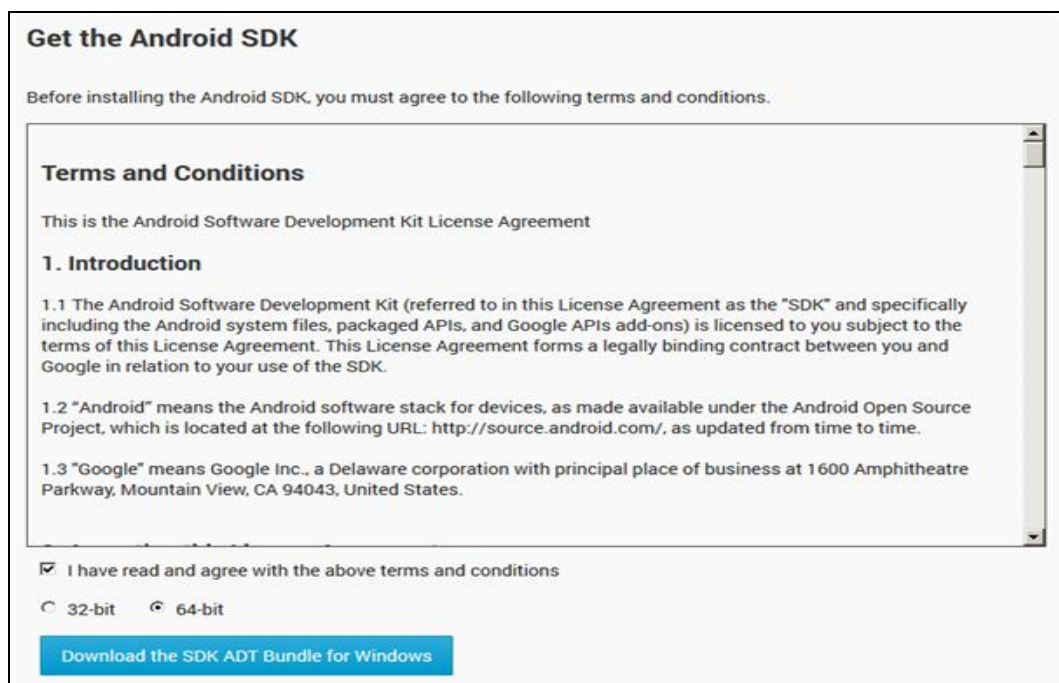


Figura 24 - Seleção do Sistema Operacional do ADT Bundle

O tamanho do arquivo de instalação é de aproximadamente 400 Mb. Realizar o download do JDK, acessando o site da Oracle.¹⁰

⁹ <http://developer.android.com/>

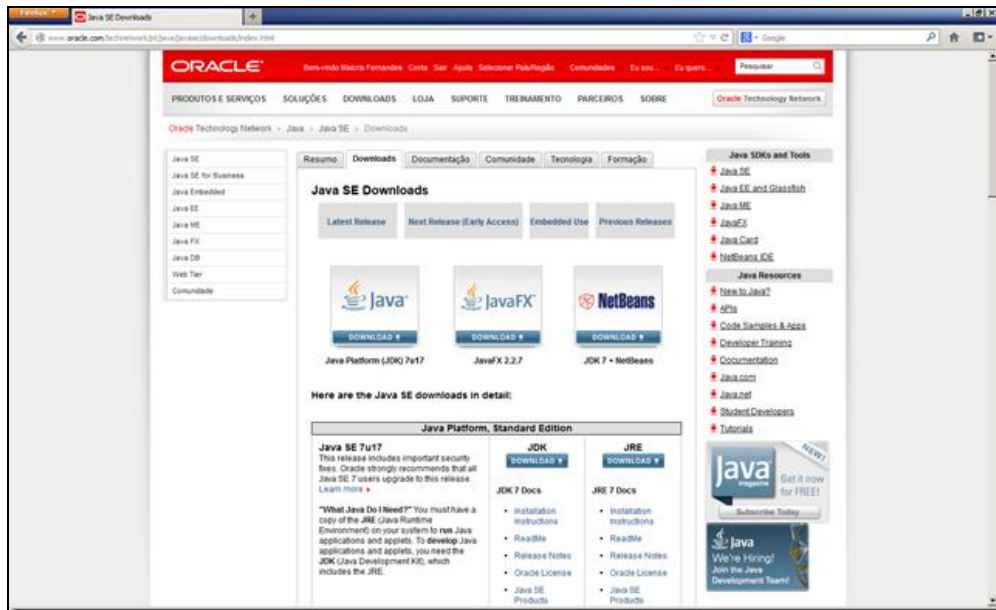


Figura 25 - Site Oracle para download da JDK

Clique no botão Download e selecione a JDK.

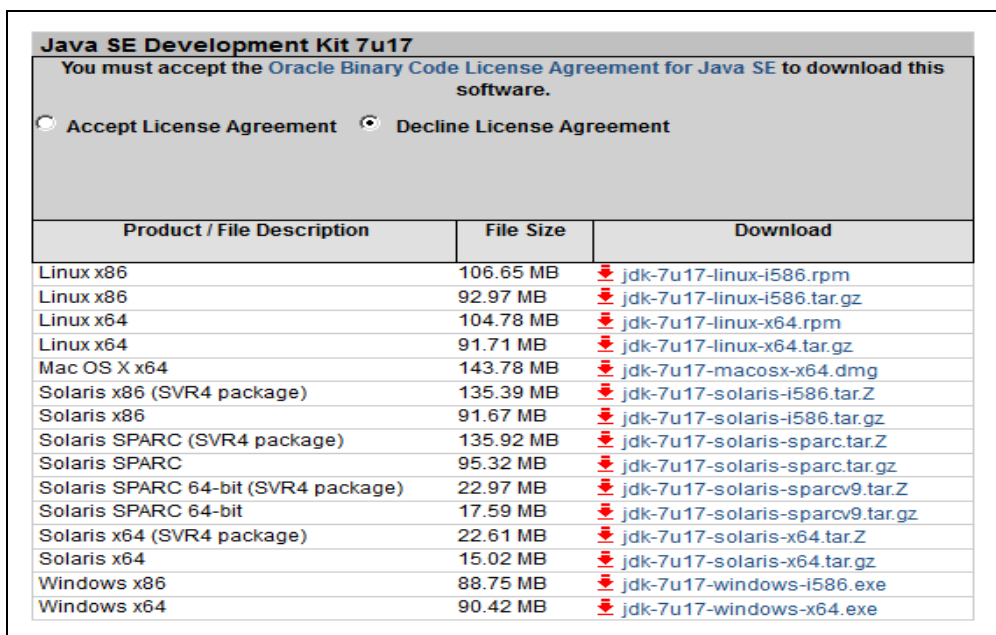


Figura 26 - Tela de seleção da JDK

Na próxima tela, aceite os termos da Oracle clique no link de download correspondente ao seu sistema operacional.

Instalação da JDK

Rode o arquivo de instalação do JDK. Nas três primeiras janelas clique em Next, e na última em Close.

¹⁰ <http://www.oracle.com/technetwork/pt/java/javase/downloads/index.html>

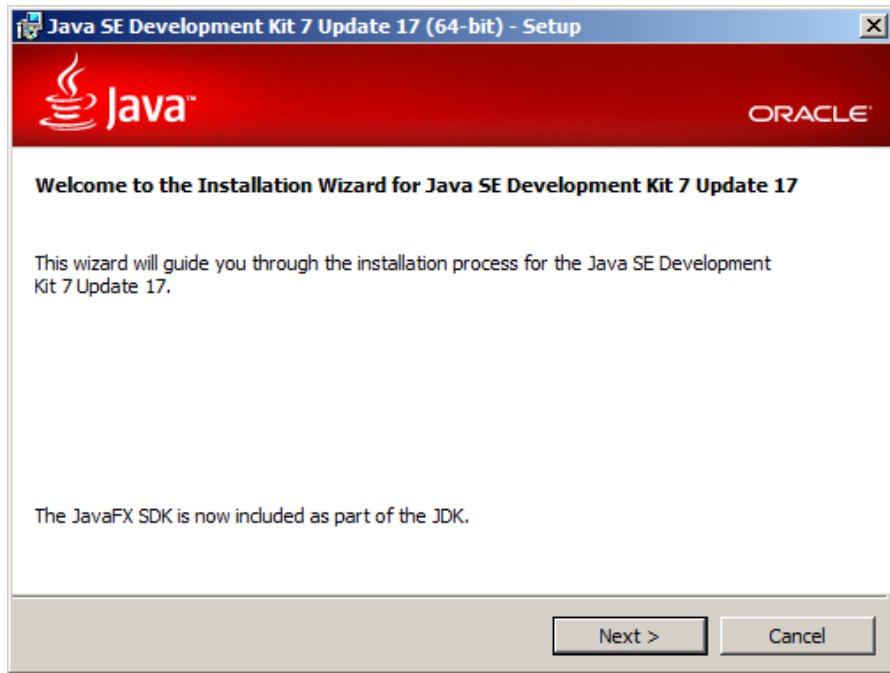


Figura 27 - Tela 1 da instalação da JDK



Figura 28 - Tela final da instalação da JDK

ADT Bundle

Para instalar, basta apenas descompactar a pasta do ADT Bundle no local desejado para instalação.

A pasta do ADT Bundle é composta por:

- Pasta Eclipse - contém os arquivos referentes ao software eclipse já configurado com o plugin ADT
- Pasta sdk - contém os arquivos do SDK Android e ferramentas adicionais
- Arquivo SDK Manager.exe - atualiza as versões do SDK do Android

Configuração

Android SDK Manager

Para realizar a atualização do Android, execute o arquivo SDK Manager.exe da pasta do ADT Bundle. Na janela Android SDK Manager que abrirá marque todas as atualizações, e clique em *install*.

Android Virtual Device Manager

Para criar um novo device (emulador) clique no ícone **Android Virtual Device Manager** na barra de ferramentas do Eclipse.

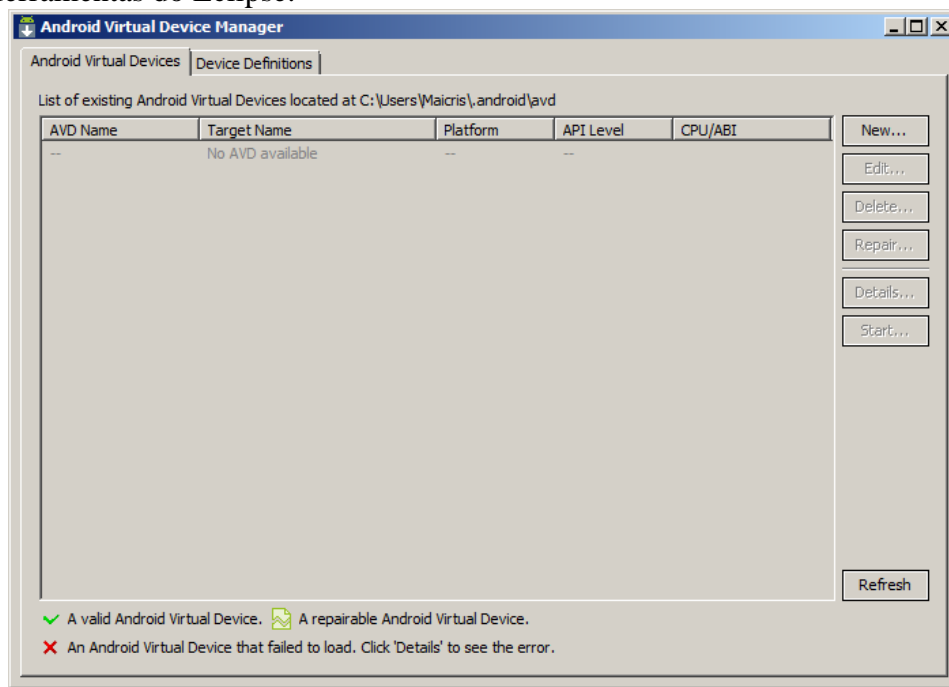


Figura 29 - Tela de criação de um device

Ao clicar em *New* na janela Android Virtual Device Manager, será aberta a janela Create New Virtual Device (AVD). Onde:

- **AVD Name:** é o nome que identifica o dispositivo.
- **Device:** define o tamanho para a tela do dispositivo.
- **Target:** define a versão da API Android.
- **CPU/ABI:** define o uso de versões diferentes de emulação, quando disponível.
- **SD Card size:** define o tamanho da memória para emulação de cartão SD do dispositivo.
- **Use Host GPU:** permite o uso de aceleração por hardware do dispositivo.

APÊNDICE II - Instalação de uma aplicação no Android

O processo de instalação de um aplicativo no Android é bastante simples, seguem os passos:

- Após ter copiado o arquivo .APK para o celular/tablet, abra o seu gerenciador de arquivos e localize a pasta com o .APK
- Clique duas vezes no .APK
- Será exibida a tela de confirmação da instalação mostrando quais alterações o aplicativo vai fazer e quais dados ele usará
- Clique em Instalar
- Caso nenhum erro aconteça, o aplicativo já está instalado, cliquem em Abrir caso queira abrir o aplicativo ou em Concluído para concluir a instalação.

A figura a seguir ilustra esse processo através da instalação de um aplicativo fictício:

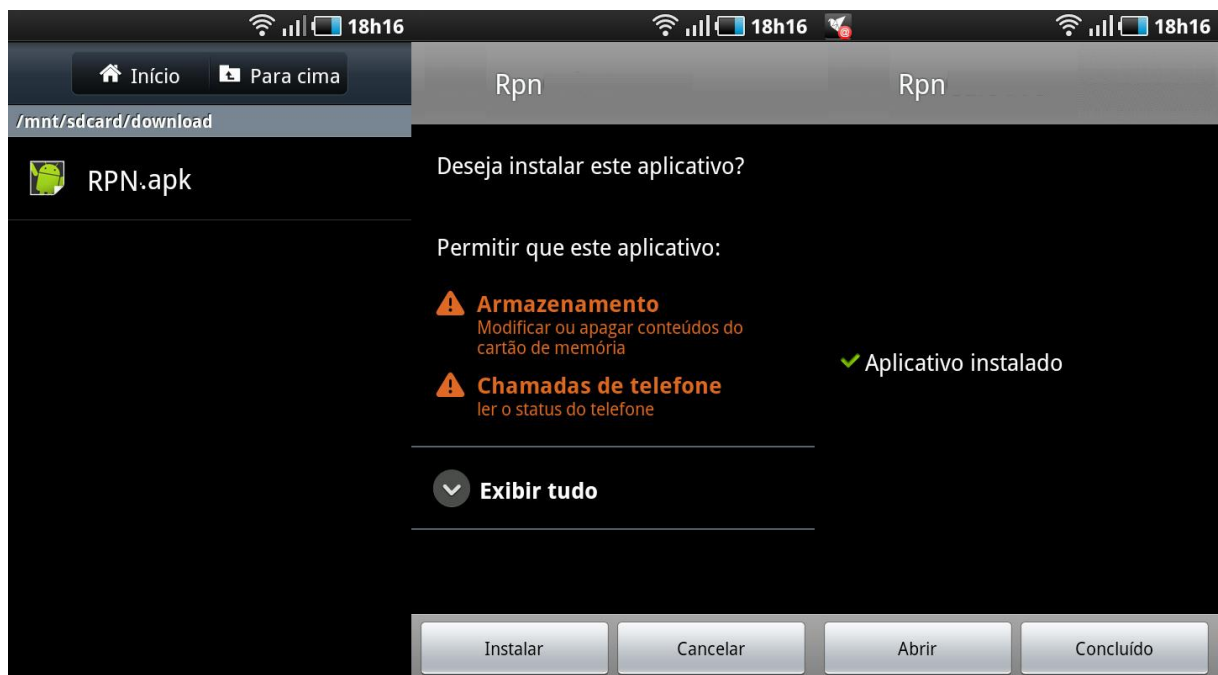


Figura 30 - Processo de Instalação de um aplicativo no Android

ANEXO I

Prontuário do aplicativo da equipe

Dados do Paciente

Data Atendimento

Nº Cartão do SUS

Nome

RG

CPF

Data Nascimento

Endereço Número

Bairro Complemento

Cidade Estado

Telefone Email

Sexo

Masculino

Feminino

Peso: Altura:

Dor no Peito

Sim

Não

Pontada

Quando Respira

Aumenta com Esforço

- Piora com a respiração
- Sudorese

Desmaio

- Sim
- Não

Falta de Ar

- Sim
- Não

Colesterol Elevado

- Sim
- Não

Diabetes

- Sim
- Não

Infarto Prévio

- Sim
- Não

Fumante

- Sim
- Não

Alcool

- Sim
- Não

Sedentarismo (Menos de 150 min semanal)

- Sim
- Não

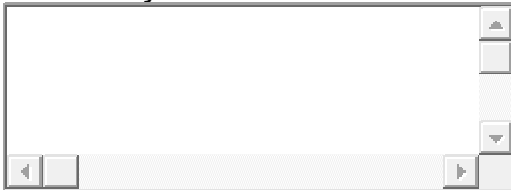
Medicamentos em uso

- ASS
- Estatina
- Antihipertensivo
- Hipoglicemiante

Pressão Arterial Diastólica

Pressão Arterial Sistólica

Frequência Cardíaca

ObservaçõesA rectangular text area with a light gray background and a thin border. It contains no text. On the right side, there are three small square buttons stacked vertically. On the bottom left, there are two small square buttons. On the bottom right, there are two small square buttons.

Dois minutos de ECG (144000 Amostras)