

Centro Universitário Barão De Mauá
Curso de Bacharelado em Ciência da Computação

USO DE MOBILIDADE NO CONTROLE DE SAÚDE

Matheus Felipe Piscioneri

Ribeirão Preto
2014

Matheus Felipe Piscioneri

USO DE MOBILIDADE NO CONTROLE DE SAÚDE

Trabalho de conclusão de curso apresentado ao Curso de Ciência da Computação do Centro Universitário Barão de Mauá, como parte dos requisitos para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Me. Fábio Bento Takeda

Ribeirão Preto

2014

P755u

Piscioneri, Matheus Felipe

Uso de mobilidade no controle de saúde/ Matheus
Felipe Piscioneri – Ribeirão Preto, 2014.

58p.il

Trabalho de conclusão do curso Ciência da Computação
do Centro Universitário Barão de Mauá.

Orientador: Ms. Fábio Bento Takeda

Dedico aos meus pais pelo apoio e confiança,
aos familiares, a minha namorada, aos meus
amigos e a todos que durante esses anos me
apoiaram e me incentivaram.

Matheus Felipe Piscioneri

AGRADECIMENTOS

Agradeço em primeiro lugar a Deus, pela vida e por sempre estar no meu caminho, me iluminando a cada decisão a ser tomada.

A minha mãe Marta, ao meu padrasto Ricardo e meu irmão Otavio, por proporcionarem uma ótima estrutura, sacrificando seus objetivos para que eu pudesse realizar os meus e sempre estando comigo em momentos importantes e nas horas mais difíceis, dando apoio e confiança para a realização de meus objetivos.

Agradeço também a minha namorada Marcela que muito me incentivou e apoiou nessa trajetória.

Ao meu orientador Takeda, ao professor Rodrigo e todos os outros professores do curso de Ciência da Computação pelos ensinamentos transferidos ao longo dos anos.

Aos meus amigos e a todos que de alguma forma contribuíram diretamente ou indiretamente para a realização do trabalho.

*“Se falta de tempo realmente fosse uma justificativa para não realizar seus projetos,
somente os desocupados teriam sucesso.”*

Flávio Augusto

RESUMO

Uso de mobilidade no controle de saúde

PISCIONERI, M, F.

Com a popularização dos *smartphones* e o avanço das tecnologias, inclusive na área da saúde, o controle e monitoração de algumas medidas de saúde como glicose, pressão arterial, batimento cardíaco, peso, altura e IMC podem usufruir da mobilidade. Pensando nisso, foi desenvolvido um aplicativo para Android® em JAVA que auxilia o usuário a monitorar e controlar sua própria saúde, para a obtenção de uma melhor qualidade de vida. As informações sobre a saúde do usuário são armazenadas no dispositivo, com a disponibilidade de compartilhamento para diversos aplicativos, inclusive com a possibilidade de ter um acompanhamento médico. Estas informações também poderão ser enviadas a um *Web Service* onde é possível visualiza-las através de um painel Web. Os resultados presentes neste trabalho mostram que o aplicativo Controle de Saúde conquistou 83% das avaliações máximas e mais de mil *downloads* com apenas três meses de publicação no Google Play®, onde foi possível observar uma grande procura, até mesmo de outros países como Portugal, França, Angola, Argentina, Chile, Espanha e Estados Unidos, alcançando principalmente a faixa etária entre 30 e 50 anos, com o maior índice de preocupação no controle e monitoramento do diabetes.

Palavras-chave: Mobile, smartphone, Android, aplicativo, Web Service, saúde.

ABSTRACT

Use of mobility control in health

PISCIONERI, M, F.

With the popularity of smartphones and the advancement of technologies, including in the area of health, the control and monitoring of some health measures such as glucose, blood pressure, heart rate, weight, height and BMI can enjoy mobility. Thinking about it, software was developed in Java for Android[®] that helps the user to monitor and control their own health, to obtain a better quality of life. Information on the health of the user is stored in the device, with the availability sharing for various applications, including the possibility of having a medical monitoring. This information may also be sent to a Web service where you can view them via a Web dashboard. The results presented in this work show that the Health Control application won 83% of the maximum ratings and over a thousand downloads after only three months of publication on Google Play[®], which was observed in great demand, even from other countries such as Portugal, France, Angola, Argentina, Chile, Spain and the United States, mainly reaching the age group between 30 and 50 years with the highest level of concern in the control and management of diabetes.

Keywords: Mobile, Smartphone, Android, application, Web Service, Health

LISTA DE FIGURAS

Figura 1: SO instalado em <i>smartphones</i> no mundo em 2014	5
Figura 2 : Diagrama da arquitetura Android®	6
Figura 3: Ciclo de vida de uma <i>activity</i>	9
Figura 4: Hierarquia dos <i>widgets</i> no Android®	12
Figura 5: Aplicativo Controle de Saúde emulado na AVD	13
Figura 6: Console do desenvolvedor Google®: (A) Estatísticas de instalação, (B) Estatísticas de instalação por versão do Android®	14
Figura 7: Diagrama <i>Model-View-Controller</i>	15
Figura 8: Utilização de versões do Android®	18
Figura 9: Fluxograma do aplicativo Controle de Saúde.....	19
Figura 10: Tela de <i>login</i>	20
Figura 11: (A) Tela de recuperação de senha, (B) Notificação de envio de recuperação de <i>e-mail</i>	21
Figura 12: Tela de cadastro.....	22
Figura 13: Tela principal	23
Figura 14: Tela Detalhes	24
Figura 15: Tela Adicionar	24
Figura 16: Tela Gráfico.....	25
Figura 17: Tela Histórico: (A) Listagem dos registros, (B) Menu de opções	26
Figura 18: Tela Sobre.....	26
Figura 20: Tela de solicitação de acompanhamento médico: (A) Menu de acesso, (B) Inserção do <i>e-mail</i> do médico.....	28
Figura 21: <i>Logout</i> do aplicativo	28
Figura 22: Fluxograma do painel Web	30
Figura 23: Tela de <i>login</i> do painel Web.....	31
Figura 24: Tela principal do painel Web	32
Figura 25: Detalhes do módulo do painel Web.....	33
Figura 26: Seleção do tipo de usuário na tela de cadastro	34
Figura 27: Lista de pacientes	34
Figura 28: Relatório de <i>downloads</i> e instalações atuais do aplicativo.....	36
Figura 29: Instalações e desinstalações diárias	36
Figura 30: Países que realizaram o <i>download</i>	37
Figura 31: Classificação do aplicativo	37
Figura 32: Usuários cadastrados por mês.....	38
Figura 33: Sexo dos usuários.....	38
Figura 34: Idade e sexo dos usuários	39
Figura 35: Cadastros no perfil médico e paciente	40
Figura 36: Registros inseridos por mês.....	41
Figura 37: Gráfico com percentagem de utilização do sistema por funcionalidades	41
Figura 38: Versão do Android mais utilizada no aplicativo	42

LISTA DE QUADROS

Quadro 1: Estados do ciclo de vida de uma aplicação.....10

Quadro 2: *Status* do registro29

LISTA DE SIGLAS

ADT	<i>Android Development Tools</i>
API	<i>Application Program/Programming Interface</i>
AVD	<i>Android Virtual Device</i>
CSS	<i>Cascading Style Sheets</i>
GPS	<i>Global Positioning System</i>
HTML	<i>Hypertext Markup Language</i>
HTTP	<i>Hyper Text Transfer Protocol</i>
IBGE	Instituto Brasileiro de Geografia e Estatística
IDE	<i>Integrated Development Environment</i>
IMC	Índice de Massa Muscular
MVC	<i>Model View Controller</i>
PHP	<i>Hypertext Preprocessor</i>
SDK	<i>Software Development Kit</i>
SO	Sistema Operacional
SQL	<i>Structured Query Language</i>
URL	<i>Uniform Resource Locator</i>
XML	<i>Extensible Markup Language</i>

SUMÁRIO

1.	INTRODUÇÃO	1
2.	FUNDAMENTAÇÃO TEÓRICA	3
2.1.	Conceitos básicos sobre saúde	3
2.2.	Mobilidade	3
2.2.1.	<i>Smartphone</i>	4
2.2.2.	Uso da mobilidade na saúde	4
2.3.	Android®	5
2.4.	Ambiente de desenvolvimento	12
2.5.	Loja de aplicativos	13
2.6.	MVC.....	15
2.7.	<i>Web Service</i>	16
3.	DETALHAMENTO DO DESENVOLVIMENTO	18
3.1.	Sistema mobile	18
3.2.	<i>Web Service</i>	29
3.3.	Painel de visualização Web	30
4.	RESULTADOS	36
5.	CONCLUSÕES	43
6.	REFERÊNCIAS BIBLIOGRÁFICAS	45

1. INTRODUÇÃO

Segundo o estudo realizado pelo IBGE (2008) sobre acesso à Internet e posse de telefone móvel celular para uso pessoal, estimou que mais da metade (53,8%) da população de 10 anos ou mais de idade, ou seja, cerca de 86 milhões de pessoas, possuía telefone celular para uso pessoal. Destaca-se também que entre 2005 e 2008, enquanto a população de 10 anos ou mais de idade cresceu 5,4%, o contingente daqueles que possuíam celulares teve aumento de 54,9%.

Sendo assim, os celulares apesar de terem poder computacional diferenciado de um computador normal, a cada momento possuem uma nova forma, tamanho, aumento na capacidade de processamento, bem como novos aplicativos agregados e com isso o desenvolvimento de aplicativos para celulares vem tornando-se cada vez mais frequente nas empresas especializadas em produção de *software* (ROMEIRO, 2005).

Quando foram lançados, os celulares eram grandes e apenas pessoas com boas condições financeiras podiam possuir um. Com o tempo eles evoluíram, ficaram cada vez menores e mais populares. Assim surgiram os celulares inteligentes, *smartphones*, aparelhos que possuem múltiplas tarefas e funções diferentes de um simples de celular.

Em paralelo a essa realidade, o crescimento da tecnologia móvel relacionada à *smartphones*, *tablets* dentre outros dispositivos vem ganhando cada vez mais importância. E com o objetivo de suprir a demanda deste mercado, a empresa Google® desenvolveu um sistema operacional denominado Android® baseado em licenciamento de código aberto, o que possibilita a contribuição de qualquer desenvolvedor criar, personalizar ou até mesmo corrigir erros do sistema. O sistema operacional Android® é baseado no *Kernel* 2.6 do Linux e é o responsável por gerenciar a memória, os processos, os *threads* e a segurança do sistema no qual o mesmo está operando (SILVA *et al.*, 2013).

Também é possível mencionar que as aplicações executadas em um celular podem ter um comportamento de funcionamento *offline* ou *online*, sincronizando informações diretamente de um servidor confiável (LECHETA, 2013).

Com base nas afirmações de Lecheta (2013), o Android® possui muitos diferenciais interessantes e uma arquitetura flexível focada na integração de

aplicações, não existindo diferença entre uma aplicação nativa e uma construída pelo desenvolvedor.

Segundo Barra *et al.*, (2006) muitos avanços tecnológicos ocorreram, inclusive na área da saúde, com a introdução da informática e do surgimento de aparelhos modernos e sofisticados que trouxeram muitos benefícios e rapidez no controle e luta contra as doenças trazendo desta forma uma melhor qualidade de vida, bem estar e saúde que são termos cada vez mais presentes no nosso cotidiano e agora poderão receber auxílio desses avanços tecnológicos.

O objetivo deste trabalho é desenvolver um aplicativo para Android® que auxilie o usuário a monitorar e controlar sua saúde, para a obtenção de uma melhor qualidade de vida.

Com base neste propósito, este documento foi estruturado em: fundamentação teórica, onde serão apresentados os principais conceitos abordados neste trabalho, detalhamento do desenvolvimento responsável por apresentar as fases de desenvolvimento, resultados e conclusões deste trabalho.

2. FUNDAMENTAÇÃO TEÓRICA

Neste capítulo é apresentada a fundamentação teórica necessária para a melhor compreensão dos principais conceitos utilizados na elaboração dessa monografia.

Sua estruturação foi definida com base nas fases de desenvolvimento deste trabalho.

2.1. Conceitos básicos sobre saúde

A saúde pode ser considerada um estado de completo bem estar físico, mental e social, não apenas na ausência de doenças ou de enfermidade. Pode não se limitar somente ao corpo, mas também sobre a mente, emoções, relações sociais, satisfação no trabalho e condições de moradia (SILVA, 2014).

O desenvolvimento da ciência e da tecnologia vem acarretando transformações na sociedade contemporânea, refletindo em mudanças nos níveis econômicos, políticos, sociais e de saúde (PINHEIRO *et al.*, 2007).

Assim sendo, a utilização de novas tecnologias podem propiciar o acompanhamento ou monitoração da saúde pelo paciente em questão ou estreitar esta relação com o profissional de saúde mesmo que fisicamente separados. Outro fato que pode favorecer esta monitoração é a popularização de tecnologias, tais como *smartphones*, os quais, por estarem sempre próximos e assim auxiliam na mobilidade de monitoração da saúde.

2.2. Mobilidade

O conceito de mobilidade consiste no uso de dispositivos móveis portáteis fornecendo serviços para que usuários se conectem, obtenham dados e os forneçam a outros usuários e sistemas. Os dispositivos móveis podem ser definidos como equipamentos portáteis que é possível carregar facilmente e que se comunica com outros dispositivos através de uma tecnologia de comunicação sem fio. Exemplos de dispositivos móveis são os *laptops*, *smartphones*, *tablets*, dentre outros. Mesmo esses dispositivos possuindo funcionalidades capazes de fornecer serviços em

qualquer hora e lugar, ainda possuem restrições como: processamento, memória, duração da bateria limitada e *interfaces* de entrada de dados limitadas (LEE 2005).

O intuito da mobilidade é a simplicidade do uso e o acesso rápido às informações. Com isso, vem crescendo o uso de *smartphones* e *tablets*.

2.2.1. Smartphone

Nos últimos 10 anos, eram poucas pessoas que utilizavam celulares. Entretanto os avanços tecnológicos propiciaram uma maior acessibilidade e assim, tornou-se um recurso importante para as pessoas (OLIVEIRA *et al.*, 2011).

Os primeiros celulares comerciais foram lançados na década de 80, não sendo nada portáteis, com custo elevado e que possibilitavam apenas realizar ligações. No decorrer dos anos, os celulares além de diminuir de tamanho acabaram agregando novas funções. Atualmente, os celulares realizam uma grande variedade de funções, sendo assim caracterizados como *smartphones* (celulares inteligentes). Estes possuem um SO (Sistema Operacional) para gerenciamento de todos componentes do dispositivo móvel. (DIAS *et al.*, 2011).

Segundo OHA (2014) atualmente, ao redor do mundo aproximadamente 1 bilhão de pessoas utilizam a Internet e quase 3 bilhões de pessoas possuem um telefone celular, sendo este um dos produtos de consumo mais bem sucedidos.

Um dos fatores que propiciaram este aumento da utilização da Internet pode estar associado ao mercado de *smartphone*, que cresceu 110% no segundo trimestre de 2013 em comparação ao mesmo período de 2012 (EXAME, 2013).

2.2.2. Uso da mobilidade na saúde

É comum considerar a ciência e tecnologia como facilitadores para o progresso, que proporcionam não só o desenvolvimento do saber humano como também uma evolução para o mesmo. Desta forma, subtemde-se que ambas trarão benefícios à humanidade (PINHEIRO *et al.*, 2007).

A emergente aplicação da informática na monitoração da saúde tem proporcionado o surgimento de novas tecnologias e aplicações, assim produzindo muitas e variadas soluções para as necessidades do profissional da saúde armazenar e recuperar informações de pacientes (SIGULEM; SALOMÃO, 2004).

A tecnologia deve propiciar a integridade, disponibilidade e confidencialidade das informações, isso faz com que se substitua o papel de celulose por um “papel virtual”, em que tudo pode ser armazenado de maneira segura e de fácil organização (BREGA *et al.*, 2011).

Dentre suas diversas oportunidades, a tecnologia permite a população e os profissionais da saúde, o acesso às informações médicas, melhorando a eficiência dos cuidados a saúde em tempo útil. Com o surgimento dos sistemas móveis e a adoção generalizada dos *smartphones*, as aplicações móveis são de grande utilidade e estão tendo uma rápida expansão (KATZ *et al.*, 2005).

2.3. Android®

O Android® é, sob diversos ângulos, uma antítese do iPhone®. Enquanto a Apple® optou por manter um controle restrito sob sua plataforma, impondo restrições aos desenvolvedores e controlando a distribuição dos aplicativos, o Google® optou por seguir o caminho oposto, criando um sistema aberto e incentivando a criação de aplicativos para a plataforma, inclusive com prêmios em dinheiro. Assim como nos *desktops*, as plataformas de smartphones estão se consolidando e o mais importante passou a serem os aplicativos e não apenas o *hardware* ou as funções básicas do sistema, veja o resultado na Figura 1 (MORIMOTO, 2009).

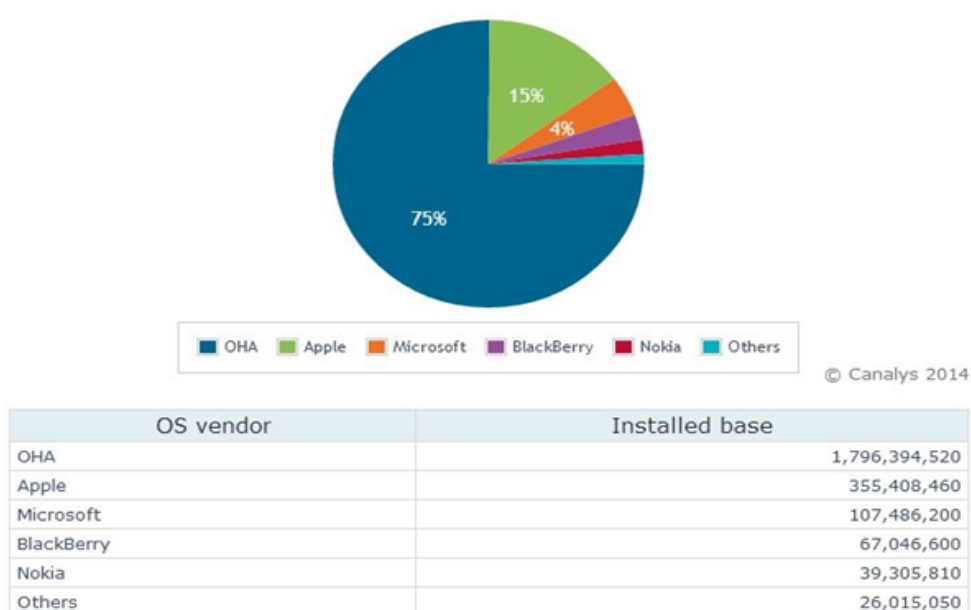


Figura 1: SO instalado em *smartphones* no mundo em 2014
Fonte - CANALYS (2014)

Observando a Figura 1, é possível concluir que o Android[®] segue disparado no número de instalações de SO em *smartphones*, somando as instalações de todos os outros concorrentes, não chegam à metade do número do Android[®].

Lançado em 2008, está disponível em 190 países ao redor do mundo e em centenas de milhões de dispositivos móveis, oferecendo um conjunto completo de *softwares* para dispositivos móveis (ANDROID, 2014).

O Android[®] foi construído sobre o *Kernel*¹ do Linux. Além disso, utiliza uma máquina virtual personalizada, Dalvik², que foi projetada para otimizar a memória e recursos de *hardware* em um ambiente móvel. Sua arquitetura é dividida em cinco camadas: *Kernel*, Bibliotecas, *Runtime*³, *Framework* de Aplicação e Aplicativos, conforme apresentado na Figura 2. Essas camadas têm suas características próprias e juntas dão sentido ao montante das aplicações para execução.

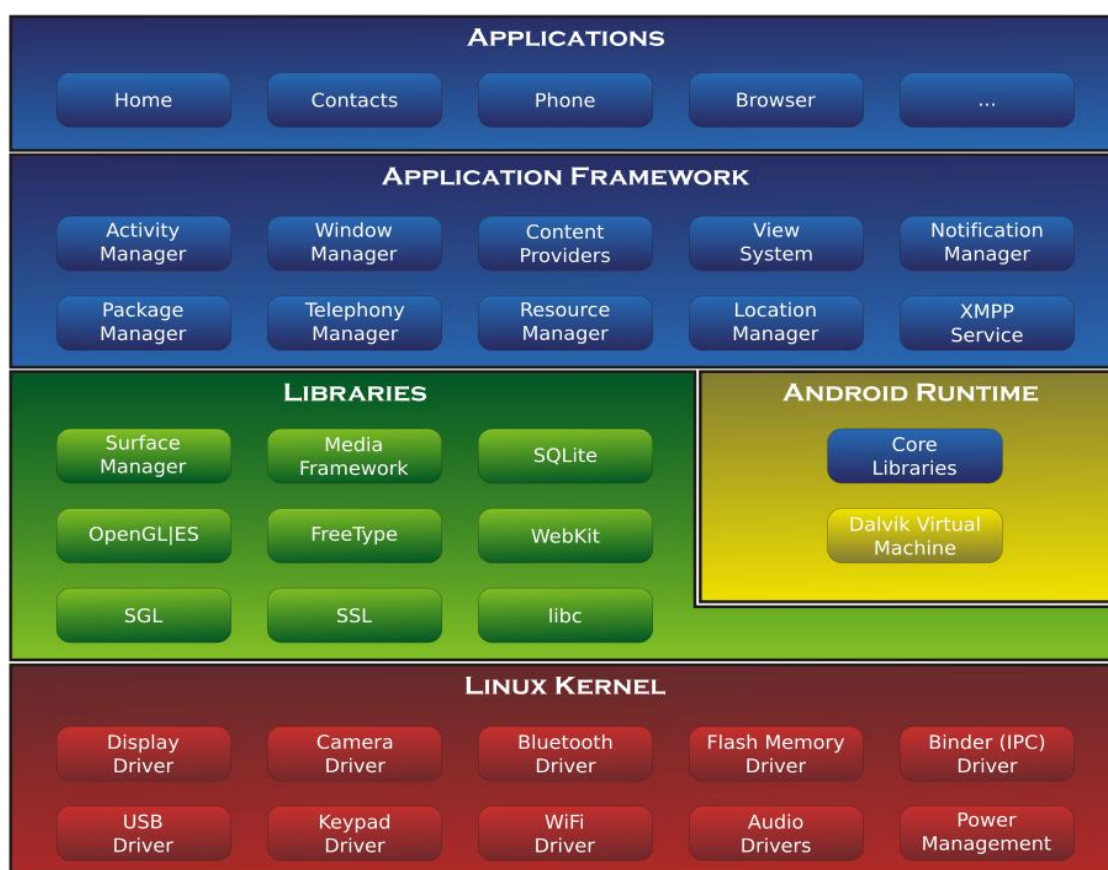


Figura 2 - Diagrama da arquitetura Android[®]
 Fonte - <http://pt.wikipedia.org/wiki/Android>

¹ É o núcleo do S.O., componente que faz a interação do *hardware* com o *software*.

² É uma máquina virtual otimizada para execução em dispositivos móveis, converte *bytecode* compilado (.class) em (.dex)

³ Tempo de execução

Segundo OLIVEIRA (2012), é possível descrever cada camada da seguinte forma:

- **Applications:** Esta é a camada mais alta da arquitetura, onde esta localizada uma lista de aplicações padrões, sendo capazes de realizar uma infinidade de atividades diretas com o usuário, como: calendário, navegador, despertador, *e-mail*, mapas, jogos e outros que serão desenvolvidos.
- **Framework Applications:** Nesta camada são fornecidas todas as funcionalidades necessárias para a construção de aplicativos, através das bibliotecas nativas. Ela é o elo entre a camada de aplicativos e a de bibliotecas, promovendo a facilidade de reutilização de procedimentos.
- **Libraries:** No Android® encontra-se um conjunto de bibliotecas C / C++ utilizadas por diversos componentes do sistema. Elas têm a capacidade de expor funcionalidades aos desenvolvedores através da estrutura de aplicativos Android®. Algumas das principais bibliotecas são: LibWebCore, SGL, Bibliotecas 3D, FreeType, SQLite.
- **Android Runtime:** Está no mesmo nível da camada de bibliotecas e é composto pela máquina virtual Dalvik, sendo responsável por permitir a execução de cada processo em sua própria instância virtual. A Dalvik é otimizada para utilizar pouca memória e permitir que tenha várias execuções de uma só vez.
- **Linux Kernel:** É a base para a execução das aplicações, onde acontece a abstração do *hardware* para o *software*. Foi baseado no *Kernel* do Linux versão 2.6, sendo responsável pelos serviços centrais do sistema, tais como segurança, gerenciamento de memória, gestão de processos, pilha de rede e modelo de *driver*.

O Android® é um sistema operacional baseado em um licenciamento *open source* podendo ser livremente estendido para incorporar novas tecnologias de ponta quando elas surgem. A plataforma continua a evoluir com a comunidade de desenvolvedores trabalhando em conjunto para construir aplicações móveis inovadoras (OHA, 2014).

Os desenvolvedores de *softwares* para *smartphones* seguem trabalhando com a ideia de tornar este equipamento essencial e necessário no dia a dia, sendo capaz de dispensar outros aparelhos de diversas funcionalidades. Atualmente os *smartphones* realizam tarefas de outros equipamentos, como por exemplo, os despertadores, o GPS (*Global Positioning System*), aparelhos de reprodução de música e vídeo, dentre outros. Ainda assim, existem ramos pouco explorados, como por exemplo, o uso na saúde. (OLIVEIRA *et al.*, 2011).

Um dos fatores importantes para os aplicativos refere-se aos mecanismos de armazenamento e recuperação de dados. Segundo Pereira e Silva (2009), o Android[®] oferece quatro mecanismos para armazenar e recuperar dados:

- Preferências compartilhadas: é um mecanismo simples para armazenar e recuperar pares de valores. Retorna o tipo e o valor do dado. Normalmente é usado para armazenar e recuperar as preferências do usuário. Estes dados irão persistir nas sessões do usuário, mesmo que o aplicativo seja finalizado.
- Arquivos: consiste no armazenamento de arquivos diretamente na memória interna do dispositivo ou em uma memória externa, o cartão SD.
- Base de Dados: o Android[®] oferece o suporte à criação e utilização de bases de dados SQLite[®]. É uma ferramenta simples e eficiente que permite armazenar os dados da aplicação em tabelas e manipular estes dados através de comandos SQL (*Structured Query Language*).
- Rede: é possível usar a rede (quando disponível) para armazenar e recuperar dados de servidores na Web, com a manipulação de endereços da Internet e solicitações HTTP (*Hypertext Transfer Protocol*).

Aplicações Android[®] tem uma característica de execução, compreender o ciclo de vida (Figura 3) de uma *activity*⁴ é muito importante, isto é, os possíveis estados em que elas se encontram, podendo ser: executando, temporariamente interrompida em segundo plano ou completamente destruída. O sistema operacional cuida desse ciclo de vida, mas ao desenvolver aplicações é importante levar cada estado possível em consideração para desenvolver uma aplicação mais robusta. O Android[®] fornece toda a estrutura necessária para

⁴ No Android[®] cada tela é representada por uma *activity*, que tem a finalidade de interagir com o usuário e tratar os eventos da tela.

isso, basta que se entenda o ciclo de vida de uma *activity* e implementá-la corretamente (LECHETA, 2013).

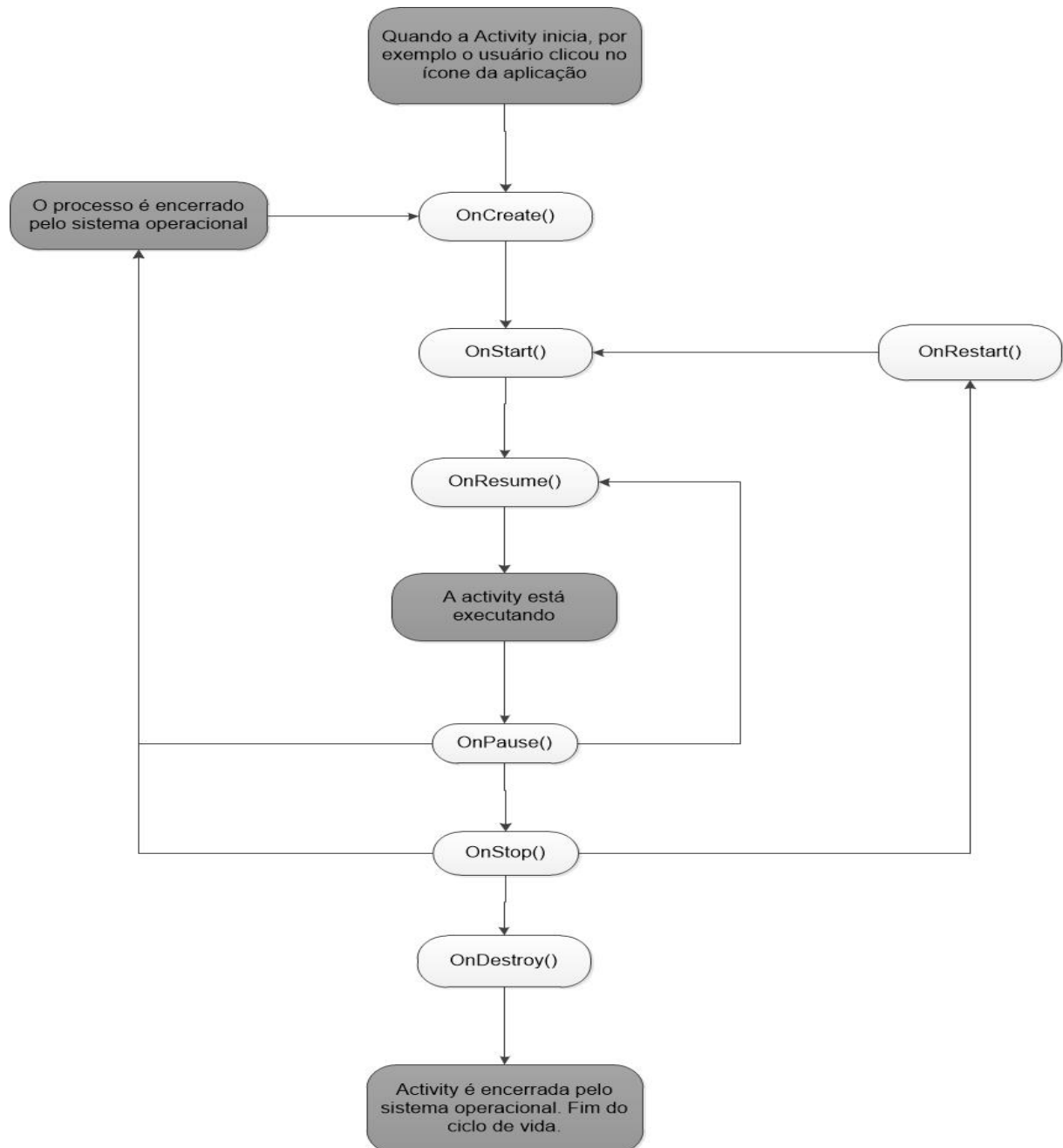


Figura 3: Ciclo de vida de uma *activity*
Fonte – LECHETA (2013)

A Figura 3 demonstra o ciclo de vida completo de uma *activity*, exibindo os estados possíveis e a chamada de cada método. A descrição detalhada destes ciclos é apresentada no Quadro 1: Estados do ciclo de vida de uma aplicação. (LECHETA, 2013).

Quadro 1: Estados do ciclo de vida de uma aplicação.

Método	Descrição
<i>onCreate()</i>	Método obrigatório e invocado uma única vez, é criado uma <i>view</i> (tela de apresentação) e chama o método <i>setContentView(view)</i> para que seja exibida a tela representada pela <i>view</i> . Depois de finalizado, o método <i>onStart()</i> é chamado para que se dê início ao ciclo de vida visível da <i>activity</i> .
<i>onStart()</i>	Método chamado quando a <i>activity</i> está ficando visível ao usuário, este método pode ser chamado logo após o <i>onCreate()</i> ou <i>onRestart()</i> .
<i>onRestart()</i>	Método chamado quando uma <i>activity</i> está parada temporariamente e esta sendo iniciada novamente, de forma automática chama o método <i>onStart()</i> .
<i>onResume()</i>	Método chamado quando a <i>activity</i> está no topo da pilha e começa a interagir com o usuário, representando o estado de execução da aplicação. É chamado sempre depois do método <i>onStart()</i> .
<i>onPause()</i>	Método chamado para salvar o estado da aplicação quando algum evento ocorre com o dispositivo, fazendo com que a <i>activity</i> encontrada no topo da pilha tenha de ser temporariamente interrompida, ficando invisível para o usuário. Mas é possível recuperar todas as informações necessárias no método <i>onResume()</i> .
<i>onStop()</i>	Método chamado quando a <i>activity</i> está sendo encerrada, e não está mais visível ao usuário. Se a <i>activity</i> for reiniciada, o método <i>onRestart()</i> é chamado ou no caso do Android® precisar de mais recursos, automaticamente o método <i>onDestroy()</i> é chamado para remover completamente a <i>activity</i> da pilha.
<i>onDestroy()</i>	Método que encerra a execução de uma <i>activity</i> , removendo completamente da pilha. Pode ser chamado automaticamente pelo sistema operacional, caso outras aplicações precisem de mais memória ou pelo método <i>finish()</i> da desta classe.

Outro fator importante a ser considerado no desenvolvimento para dispositivos móveis é a usabilidade, que pode ajudar a minimizar as dificuldades de interação com o usuários. Técnicas de avaliação de usabilidade específicas para aplicações móveis podem beneficiar o processo de desenvolvimento em tais dispositivos. Várias técnicas de avaliação de usabilidade têm sido propostas, como os *User Interface Guidelines* que são guias para auxiliar o desenvolvedor a aproveitar ao máximo os recursos de interface da plataforma, mas estas técnicas ainda não estão consolidadas (BONIFÁCIO *et al.*, 2010).

Segundo o Android (2014) as aplicações para este sistema operacional devem seguir três objetivos, são eles:

- Aplicativos Android® devem ser elegantes e ter *interfaces* agradáveis, as transições devem ser de fácil acesso, rápidas e claras, o *layout*, a tipografia e os ícones devem ser nítidos e significativos. A aplicação deve combinar beleza, simplicidade e criar uma experiência mágica, fácil e poderosa ao usuário.
- Ao utilizarem o aplicativo pela primeira vez, as pessoas precisam compreender as características mais importantes. Tarefas simples nunca devem exigir procedimentos complexos e tarefas complexas devem ser adaptadas para facilitar.
- Aplicativos Android® capacitam as pessoas a experimentar coisas novas e usá-las de novas maneiras, permitindo que os usuários combinem aplicações em novos fluxos de trabalho por meio de multitarefas, com outros aplicativos, notificações e compartilhamento.

O Android® oferece um sofisticado e poderoso modelo, baseado em componentes para construir sua *interface*, baseado no esquema de classes: *widgets* e *layouts* (LECHETA, 2010). Também é possível criar subclasses para personalizar algum componente ou até criar novas classes para personalizar totalmente os elementos gráficos.

Alguns exemplos de *widgets* e *layouts*: Button, TextView, EditText, ListView, CheckBox, RadioButton, Gallery, Spinner, ProgressBar, AnalogClock, ImageView, ImageButton, LinearLayout, FrameLayout, RelativeLayout, GridLayout, dentre outros, como mostrado na Figura 4.

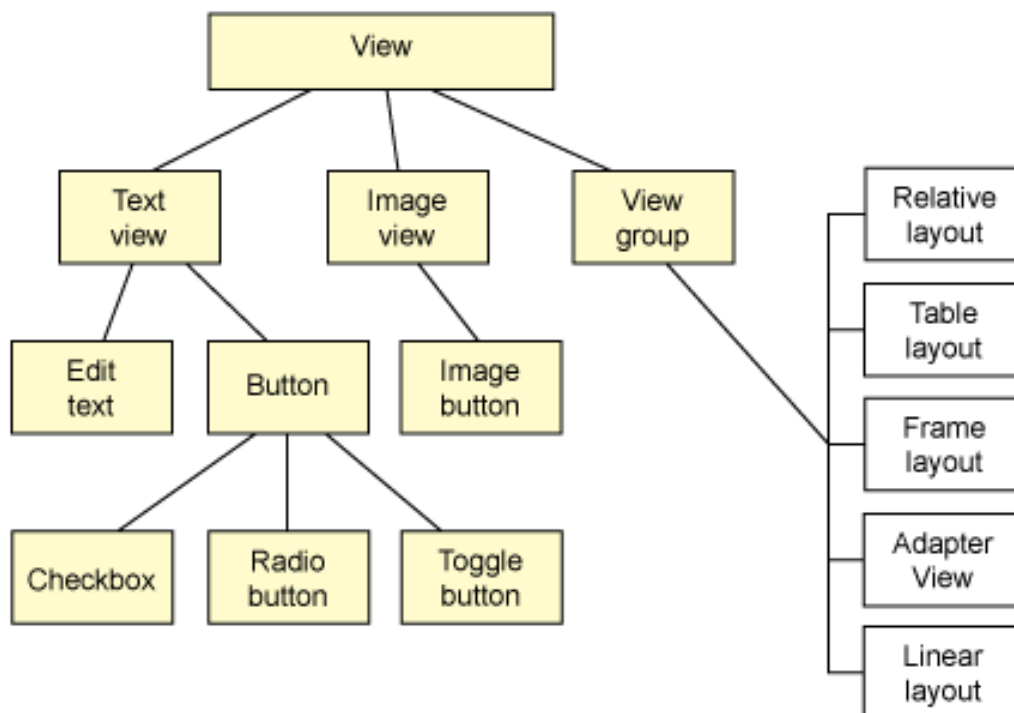


Figura 4: Hierarquia dos *widgets* no Android®
Fonte - IBM (2010)

2.4. Ambiente de desenvolvimento

A escolha do Sistema Operacional Android® foi feita primeiramente pelo conhecimento prévio que já possuía sobre a plataforma, por possuir conhecimento na linguagem JAVA e pelo ambiente de desenvolvimento sem custo.

As ferramentas utilizadas no desenvolvimento do aplicativo Controle de Saúde foram: o Android SDK (*Software Development Kit*), a IDE (*Integrated Development Environment*) Eclipse com o plugin ADT (*Android Development Tools*), a linguagem de programação Java e o banco de dados SQLite. Para desenvolver o *Web Service* e o painel foram utilizadas a IDE NetBeans, a linguagem de programação PHP e o banco de dados MySQL®. Essas ferramentas foram escolhidas pela familiaridade com as mesmas.

O Android SDK é o *kit* de desenvolvimento provido pela Google® que contém as ferramentas e bibliotecas necessárias para construir e testar aplicações para Android®. Junto com o SDK é disponibilizado o simulador de dispositivos Android® (Figura 5), o AVD (*Android Virtual Device*) que apesar das limitações, possibilita realizar a maior parte dos testes, com a vantagem de poder simular os diversos

tamanhos de tela, resoluções e versões do sistema, mas sua principal desvantagem é o tempo de simulação da aplicação. O Eclipse foi escolhido junto com o *plugin* ADT desenvolvido pelo Google®, é uma ferramenta que dispõe de diversos recursos para o desenvolvimento de aplicações para Android®. (MASCARENHAS *et al.*, 2013).

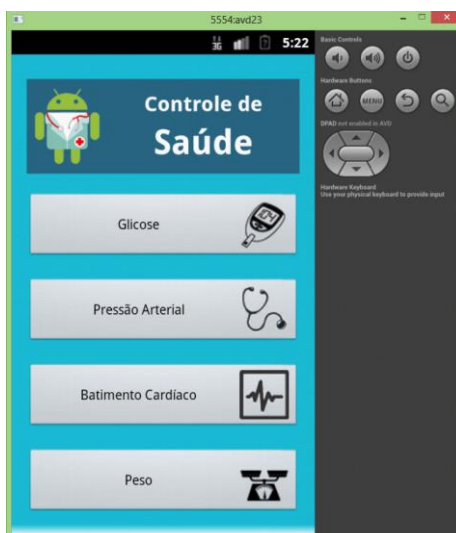


Figura 5: Aplicativo Controle de Saúde emulado na AVD
Fonte – Elaborada pelo autor

2.5. Loja de aplicativos

A flexibilidade do Android® tornou-se um dos pontos favoritos para os consumidores e desenvolvedores, contribuindo para o forte crescimento do consumo de aplicativos. Usuários desta plataforma realizam mais de 1,5 bilhão de *downloads* de aplicativos e jogos por mês na sua loja virtual, o Google Play® (ANDROID, 2014).

Ao publicar um aplicativo no Google Play®, o desenvolvedor alcança todos os usuários Android®, pois é o principal canal de distribuição de aplicativos para essa plataforma. Esse mercado de aplicativos permite ao desenvolvedor controlar a distribuição de seus produtos, podendo publicar quando quiser, quantas vezes quiser e para os clientes que deseja, se concentrando em segmentos específicos, de dispositivos, ou de recursos de *hardware* (ANDROID, 2014).

O Console de desenvolvedor do Google® (Figura 6) é o local que possui todos os dados de sua conta de desenvolvedor, permitindo a realização do *upload* e publicação de seus aplicativos, onde é possível controlar suas distribuições

gerenciando os países que seu aplicativo será distribuído, verificando a lista de dispositivos onde seu aplicativo está disponível com base no seu arquivo de manifesto, estabelecendo preço, analisando as falhas enviadas pelos usuários e possuindo acesso a diversas estatísticas, como instalações, *downloads*, comentários, classificações, atualizações, versões do Android®.

(A)



(B)

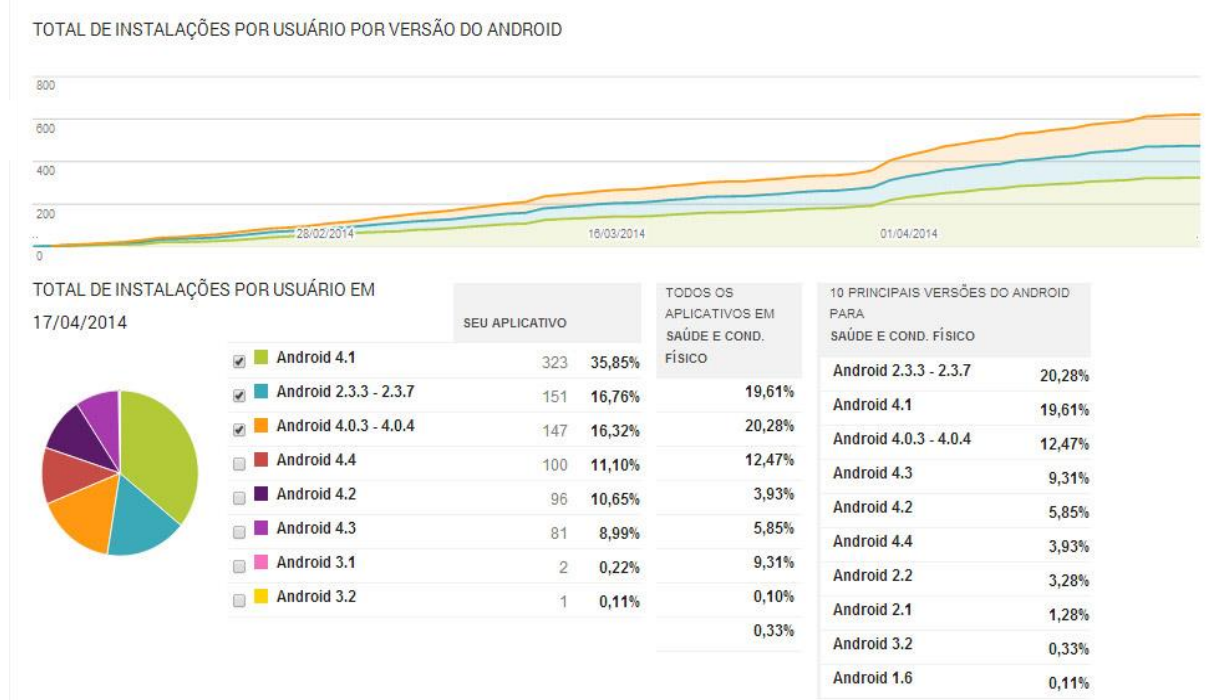


Figura 6: Console do desenvolvedor Google®: (A) Estatísticas de instalação, (B) Estatísticas de instalação por versão do Android®

Fonte – Elaborada pelo autor

As diversas estatísticas apresentadas na Figura 6 podem ajudar o desenvolvedor a conhecer melhor seu público alvo, a monitorar e priorizar os perfis de dispositivos com o objetivo de aperfeiçoar sua aplicação Android® (ANDROID, 2014).

2.6. MVC

Um fator importante a ser observado, trata-se do modelo em camadas baseado em MVC (*Model-View-Controller*) que o desenvolvimento Android® utiliza. O MVC é um padrão de arquitetura de *software* fundamental na separação dos dados (*Model*), do *layout* (*View*) e dos controladores (*Controller*). Desta forma, alterações feitas no *layout* não afetam a manipulação de dados nem os controladores, estes poderão ser reorganizados sem alterar o *layout* (MÄHLMANN; WAGNER, 2012).

Conforme explicado por Gonçalves (2008), o MVC (Figura 7) possui os seguintes componentes:

- **Model:** O *Model* (Modelo) é a camada que representa os seus dados, provendo meios de acesso à leitura e escrita. Essa camada não possui conhecimento específico dos controladores (*controller*) e das apresentações (*views*).
- **View:** É na camada *View* (Apresentação) que acontece a interação com os usuários, apresentando as entradas e saídas das informações obtidos pelo *Model*.
- **Controller:** O *Controller* (Controlador) é responsável por controlar todo o fluxo de informação que passa pela aplicação, servindo como uma camada intermediária entre a camada *view* e a *model*. Dirige as apresentações a serem exibidas e com as devidas mudanças de dados vindas da camada modelo.

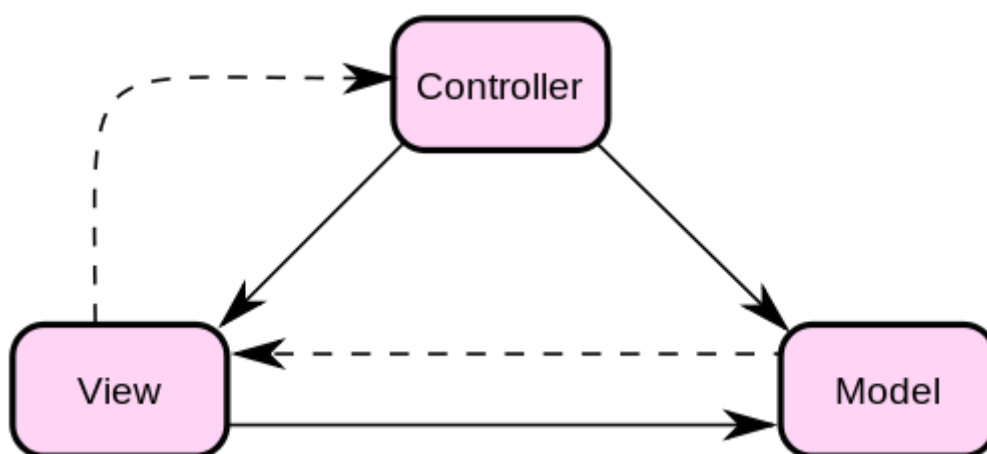


Figura 7: Diagrama *Model-View-Controller*
Fonte - <http://pt.wikipedia.org/wiki/MVC>

A Figura 7: Diagrama *Model-View-Controller* demonstra a relação entre *Model*, *View* e *Controller*. As linhas sólidas indicam associação direta e as tracejadas indicam associação indireta (WIKIPÉDIA, 2010).

Vários problemas podem surgir quando os *softwares* tem associação no código de acesso aos dados, controles e apresentação dos dados. Tais aplicações com os códigos desorganizados podem ser de difícil entendimento e manutenção, pois todos os componentes estão juntos, causando fortes efeitos em cascata sempre que uma alteração precisa ser feita em qualquer lugar. Com esse acoplamento se torna praticamente impossível à reutilização do código. (ORACLE, 2002).

A separação lógica da aplicação nestas partes assegura que a camada de modelo fique restrita a representar os componentes do problema. Igualmente, a camada de apresentação que está relacionada na exibição dos dados. Já o controlador, tem a responsabilidade de atuar como um intermediador (GONÇALVES, 2008).

2.7. Web Service

Os *Web Services* são aplicações modulares, auto descritivas, reutilizáveis, acessíveis facilmente através de uma URL (*Uniform Resource Locator*), geralmente via protocolo HTTP, independentes da plataforma de desenvolvimento e que permitem a interação entre aplicações sem intervenção humana. Apresentam-se como a solução para muitos problemas associados aos sistemas distribuídos (LOPES; RAMALHO, 2004).

Segundo QUEIROZ (2011) algumas características podem ser destacadas no MVC, como:

- Custo reduzido, pois o modelo permite um alto grau de reutilização de componentes que podem ser acessados via rede, reduzindo significativamente os custos de desenvolvimento.
- Alta confiabilidade e disponibilidade, pois os servidores e as aplicações podem ser disponibilizados em conjuntos redundantes, onde respondem de modo alternado a uma solicitação, dividindo a carga entre eles de modo que se um falhar haverá outros capazes de realizar a mesma solicitação.

- Extensibilidade, que pode ser executada através da capacidade dinâmica de configuração e reconfiguração de uma aplicação distribuída entre os recursos disponíveis na rede.
- Alta produtividade e curto ciclo de desenvolvimento, pois quebra uma grande aplicação em componentes pequenos que podem ser desenvolvidos de forma isolada.
- Reutilização, pois os componentes distribuídos podem ser executados por diversos serviços e utilizados por diversos clientes de diversas aplicações, economizando esforço repetitivo de desenvolvimento e aumentando a interoperabilidade entre componentes.

Por ser independente de plataforma, o *Web Service* facilita a comunicação entre sistemas diferentes. Sendo assim, o desenvolvimento e publicação de aplicativos em outras plataformas são favorecidos.

Como é o caso do painel de visualização Web do aplicativo Controle de Saúde. O painel foi desenvolvido em PHP (*Hypertext Preprocessor*) e acessa os dados cadastrados no aplicativo desenvolvido em Java para a plataforma Android®.

3. DETALHAMENTO DO DESENVOLVIMENTO

As etapas do desenvolvimento foram: design das telas, criação da *interface*, programação do aplicativo em JAVA, programação do *Web Service* em PHP, testes no emulador e em diversos dispositivos para verificar erros, publicação do aplicativo gratuitamente no Google Play®, tornando acessível a qualquer usuário da plataforma Android® com a versão 2.3.3 (Gingerbread) ou mais recente, correções de *bugs* e melhorias, criação do módulo de acompanhamento médico e por fim a criação do painel Web para visualização.

3.1. Sistema mobile

Analisando os dados disponibilizados pelo Google® (Figura 8), foi feita a escolha da versão 2.3.3 Gingerbread API (*Application Programming Interface*) 10 para o desenvolvimento do aplicativo Controle de Saúde, devido a um grande número de usuários que ainda utilizam esta versão, cerca de 20% de quem se conecta ao Google Play®.

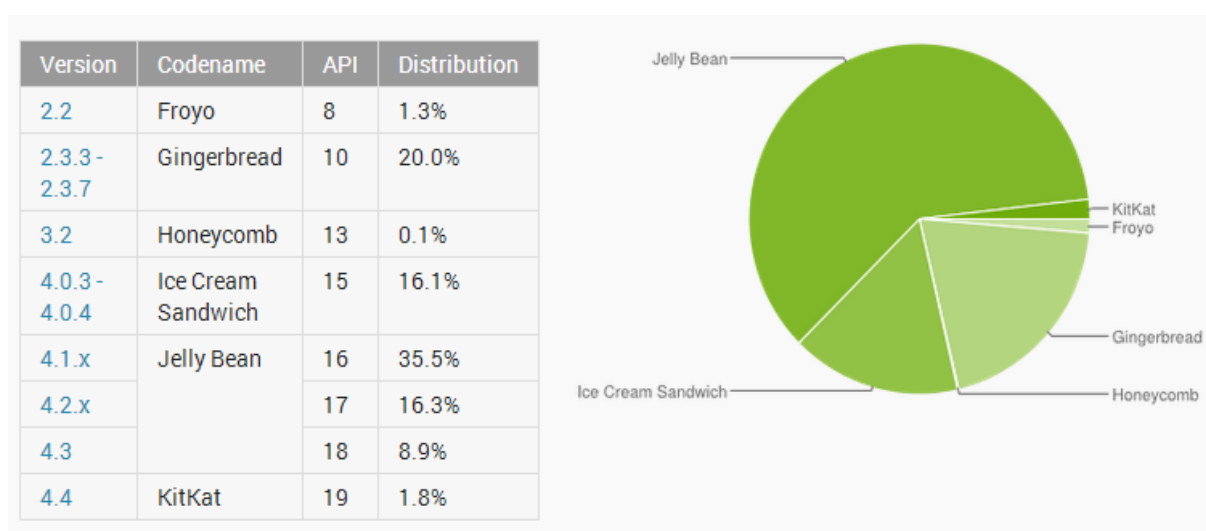


Figura 8: Utilização de versões do Android®
Fonte – ANDROID (2014)

Na Figura 9 é possível visualizar o fluxograma geral do desenvolvimento *mobile*.

tenha, o usuário é redirecionado para a tela de *login* (Figura 10). O usuário que já tenha cadastro poderá colocar seu *e-mail* e senha. Se existir conexão com a Internet, estas informações são enviadas para validação no *Web Service*, havendo sucesso o usuário poderá acessar no sistema e ficará salvo nas preferências compartilhadas do aplicativo que o usuário já está *logado*, assim não será necessário digitar o *e-mail* e senha novamente quando abrir o aplicativo. Caso não exista conexão com a Internet, será mostrada uma mensagem informando o usuário, “Você está desconectado”.



Figura 10: Tela de *login*
Fonte – Elaborada pelo autor

Caso não exista o *e-mail* e/ou senha na base de usuários cadastrados, será retornado um erro “Usuário ou Senha Inválidos!”.

Se o usuário já tem cadastro no aplicativo, mas não se recorda da senha que foi cadastrada, poderá clicar solicitar a recuperação da mesma, na opção “Esqueci minha senha” (Figura 11 A). Será necessário que o usuário preencha o *e-mail* que foi cadastrado e clique no botão “Recuperar”, a aplicação realizará uma verificação no *Web Service* para confirmar a existência do *e-mail*, se realmente existir, é enviado um *e-mail* para este usuário com a senha que foi cadastrada. A Figura 11 B representa o retorno quando o *e-mail* de senha é enviado com sucesso.

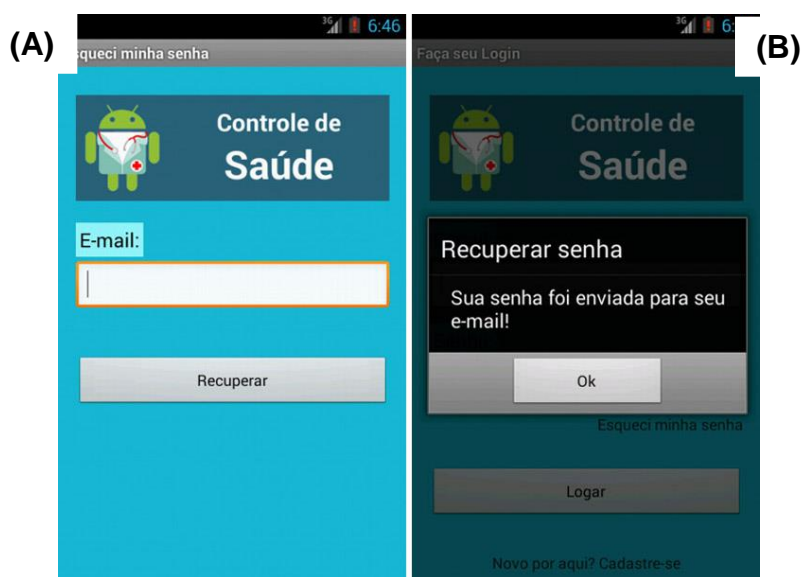


Figura 11: (A) Tela de recuperação de senha, (B) Notificação de envio de recuperação de *e-mail*
Fonte – Elaborada pelo autor

Lembrando que para o processo de recuperação de senha, é necessário ter conectividade com a Internet.

Caso seja um novo usuário, ele poderá realizar o cadastro para utilizar as funcionalidades do aplicativo. Clicando em “Novo por aqui? Cadastre-se” (Figura 12), será solicitado o preenchimento de algumas informações pessoais importantes sobre o usuário, como nome, *e-mail*, senha, sexo, data de nascimento e o tipo do usuário (paciente ou médico) que será de suma importância, pois apenas os usuários que se cadastraram com o perfil de médico, poderão ter acesso ao acompanhamento de pacientes.

A imagem mostra a tela de cadastro de um aplicativo móvel. No topo, há uma barra de status com o tempo 6:44 e ícones de rede e bateria. Abaixo, uma barra de título azul contém o texto 'Cadastre-se' e um ícone de um robô verde com uma cruz vermelha no peito. O título principal 'Controle de Saúde' está em uma caixa azul escura. Abaixo, há quatro campos de entrada brancos com rótulos azuis: 'Nome:', 'E-mail:', 'Senha:' e 'Confirmar senha:'. O campo 'Nome:' está destacado com uma borda laranja.

Figura 12: Tela de cadastro
Fonte – Elaborada pelo autor

Ao clicar em “Cadastrar” todos os dados são enviados para o *Web Service* e é feita uma verificação para validar se não existe um usuário cadastrado com o mesmo *e-mail* fornecido, se já existir, será mostrado um erro, informando ao usuário que o *e-mail* já foi cadastrado no aplicativo e solicitando que seja feito o *login*, redirecionando-o para a tela de *login*.

Se o *e-mail* não existir, os dados do usuário serão salvos no *Web Service* e ficará registrado nas preferências compartilhadas do aplicativo que o usuário está *logado*, sendo assim, não precisará inserir o *e-mail* e senha novamente quando abrir o aplicativo. Após este processo o usuário será redirecionado para a tela principal, podendo usufruir do aplicativo.

Na tela principal do aplicativo, contém a listagem dos módulos para o acompanhamento da saúde, Glicose, Pressão Arterial, Batimento Cardíaco, Peso, Altura e IMC (Índice de Massa Corporal). Esses itens foram escolhidos por serem sinais vitais e informações importantes sobre a condição de saúde do usuário, estes itens irão auxiliar os próprios pacientes e os profissionais da área de saúde sobre quais as ações necessárias para que o usuário tenha uma vida saudável e ativa (Figura 13).

No menu de contexto da tela principal do aplicativo tem algumas opções para acesso, tela Sobre, o módulo de solicitação de acompanhamento médico e a opção de *logout*.



Figura 13: Tela principal
Fonte – Elaborada pelo autor

Toda vez que a tela principal é chamada, a atividade de exportação de dados é acionada automaticamente. Esta função verifica se existe conexão com a Internet, através do *wifi* ou do 3g. Se o dispositivo estiver conectado, é iniciado o serviço de exportação de registros para o *Web Service*, este serviço cria uma nova *thread*⁵ que faz uma busca no banco de dados local por algum registro que foi inserido em um dos módulos e ainda não foi exportado, se existir é criado uma URL com todas as informações do registro e enviada ao *Web Service* pela biblioteca Apache HttpClient através do método GET⁶. Após este processo, o aplicativo recebe de volta uma mensagem informando se a exportação obteve sucesso ou não, se teve, o registro é atualizado localmente para saber que já foi exportado e não será mais necessário ser enviado ao *Web Service*. Como este processo pode ser demorado, pois depende da conexão com a Internet, ele é efetuado em *background* para não interferir e nem impedir que o usuário continue usando o aplicativo.

Todos os seis itens de monitoração apresentam o mesmo *layout*, (Figura 14), onde é exibida uma pequena descrição do módulo, exibido os maiores, menores e registros médios. Ao final da tela também são apresentados os últimos registros inseridos.

⁵ Thread é o encadeamento de execução, uma forma de um processo dividir a si mesmo em duas ou mais tarefas que podem ser executadas concorrentemente.

⁶ O método GET utiliza a própria URL para enviar dados ao servidor.



Figura 14: Tela Detalhes
Fonte – Elaborada pelo autor

O recurso *de* guias foi utilizado no aplicativo para deixar mais fácil e intuitivo para o usuário explorar e alternar entre as telas. Assim, o usuário não precisar ficar indo e voltando entre elas, pois as mesmas estão sempre visíveis e acessíveis, como pode ser visto no destaque da Figura 14.

A inclusão de um novo registro pode ser realizada pelo botão adicionar do menu de navegação. Ao clicar, uma nova janela é apresentada conforme a Figura 15. Ao salvar, os dados são armazenados no banco de dados interno do aplicativo e marcado para realizar a exportação ao *Web Service*, essa exportação ocorrerá quando o usuário voltar para a tela principal e estiver conectado com a Internet.



Figura 15: Tela Adicionar
Fonte – Elaborada pelo autor

Na tela Gráfico (Figura 16), é disponibilizado um gráfico de linha criado com a biblioteca AndroidPlot, que auxilia a criação de gráficos dinâmicos. Esta tela foi desenvolvida para facilitar o usuário a visualizar as últimas informações cadastradas, nela são mostradas até os últimos 14 registros inseridos, visando um melhor acompanhamento do usuário.

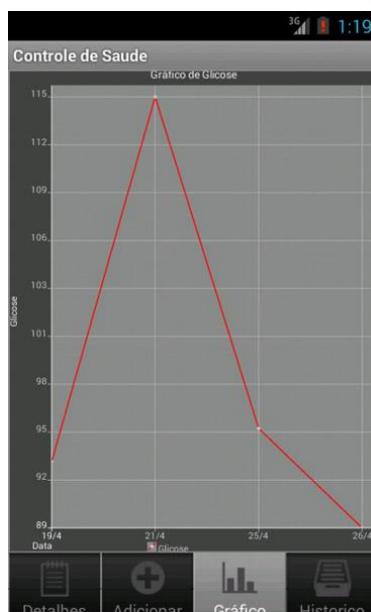


Figura 16: Tela Gráfico
Fonte – Elaborada pelo autor

Quando o usuário desejar visualizar todos os registros que já foram inseridos em cada módulo, pode ir até à aba Histórico (Figura 17 A). Nesta tela é possível editar e apagar cada registros. Para acionar estas duas opções, basta manter o registro pressionado até aparecer o menu de opções (Figura 17 B). Se o usuário clicar em editar, a aplicação é redirecionada para a tela de alteração e as informações já aparecerão preenchidas, bastando o usuário alterar apenas o que é necessário e depois salvar. O registro é atualizado localmente e assim que possível, à alteração será replicada no *Web Service* conforme descrito anteriormente. Na opção remover, o registro ficará oculto no aplicativo até ser sincronizado e deletado do *Web Service*, após essa operação, automaticamente o registro é removido no aplicativo também.

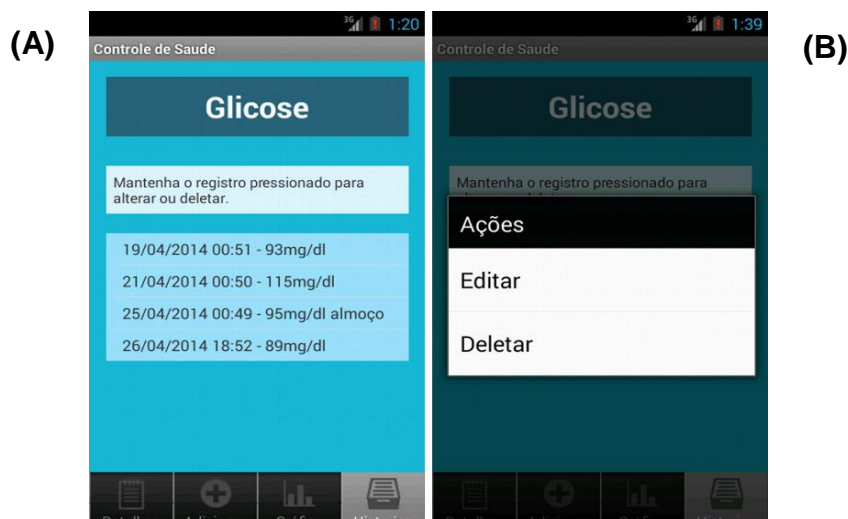


Figura 17: Tela Histórico: (A) Listagem dos registros, (B) Menu de opções
Fonte – Elaborada pelo autor

Na tela Detalhes do módulo, tem a opção compartilhar, que extrai todas as informações desta tela e mostra várias opções de aplicativos para realizar o compartilhamento, entre eles: *e-mail*, mensagem, WhatsApp®, *chat* do Google®, *chat* do Facebook®, Evernote® e bloco de notas.

Algumas informações sobre o aplicativo e o desenvolvedor, podem ser visualizadas na tela Sobre acessada pelo menu da tela principal, conforme visto na Figura 18.



Figura 18: Tela Sobre
Fonte – Elaborada pelo autor

O *link* disponibilizado visto na Figura 18, pra classificação do aplicativo tem o intuito de incentivar os usuários a avaliarem no Google Play®, através de estrelas e comentários. Essa avaliação é muito importante para o desenvolvedor ter um *feedback* de como o público esta recebendo seu aplicativo. Cada usuário pode classificar e comentar apenas uma vez cada aplicativo, mas a qualquer momento pode modificar sua avaliação.

Após o desenvolvimento foi observado que poderia ter um acompanhamento de um médico. Sendo assim, foi desenvolvida uma *interface* Web, a qual, se o paciente solicitar o acompanhamento médico, ele podem realizar o acompanhamento das monitorações realizadas.

Como não seria possível estimar o número de pacientes que um médico poderia ter, a *interface mobile* seria muito problemática neste caso, podendo atrapalhar a visualização no dispositivo. Por isso foi escolhido criar uma *interface* de monitoração do médico no painel Web.

O processo completo de solicitação de acompanhamento médico pode ser visualizado na Figura 19 em forma de fluxograma.

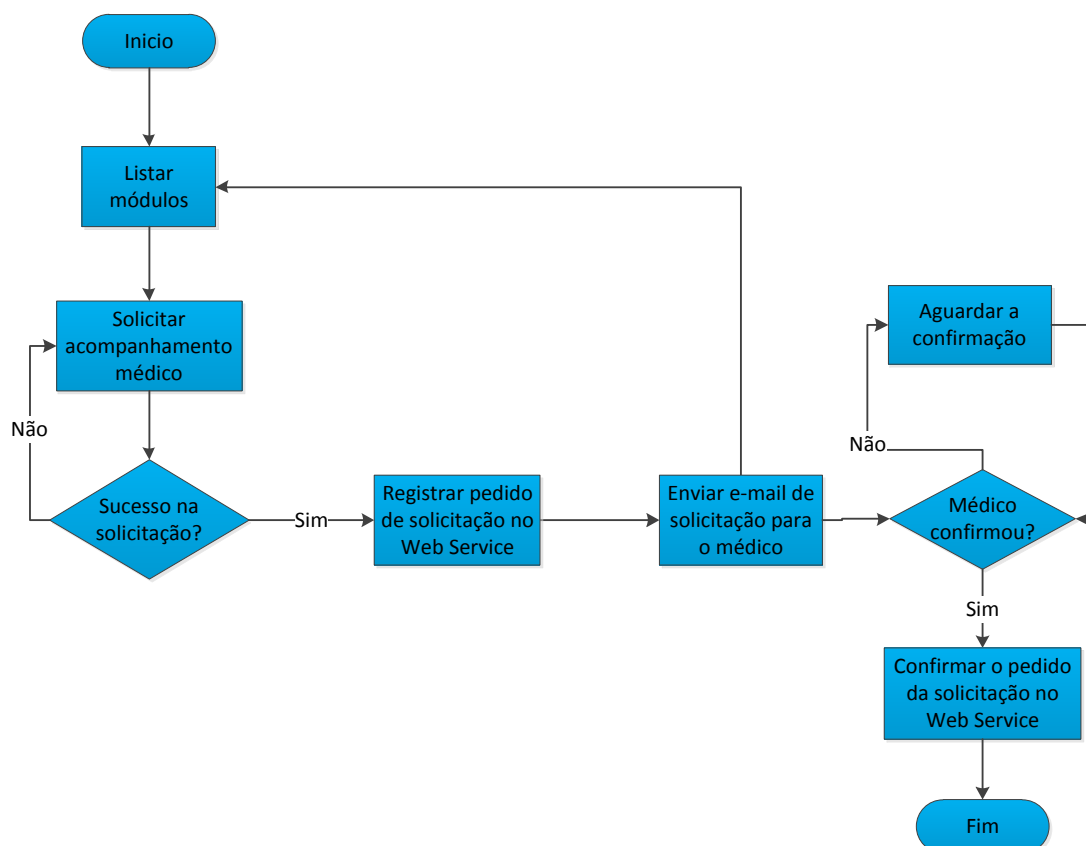


Figura 19: Fluxograma de acompanhamento médico
Fonte – Elaborada pelo autor

Com a disponibilidade do serviço de acompanhamento médico, o usuário que desejar que o médico tenha acesso às suas informações registradas no aplicativo, pode fazer uma solicitação a ele, esta opção pode ser encontrada acessando o menu da tela principal e entrando em “Acompanhamento médico” (Figura 20 A). Nesta tela (Figura 20 B), o usuário digita o *e-mail* que o médico se cadastrou, essa informação é enviada ao *Web Service* e é validado a existência do médico na base de dados. Se ele já estiver cadastrado, receberá um *e-mail* com um endereço, acessando este endereço é confirmado o acompanhamento. Após a confirmação o médico poderá acompanhar os registros do paciente através do painel Web. Caso o médico ainda não tenha cadastro, ou o *e-mail* digitado esteja incorreto, aparecerá uma mensagem de erro informando que o *e-mail* é inválido.

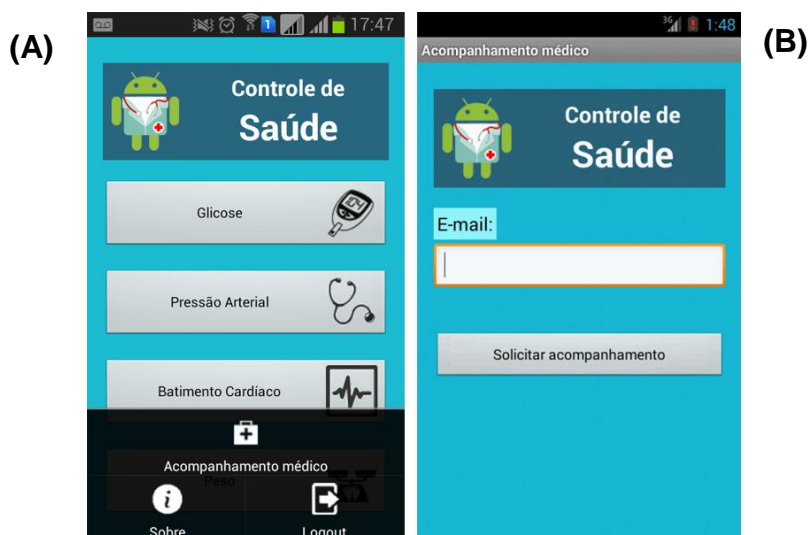


Figura 20: Tela de solicitação de acompanhamento médico: (A) Menu de acesso, (B) Inserção do *e-mail* do médico
Fonte – Elaborada pelo autor

Quando o usuário não desejar mais ser identificado automaticamente pelo aplicativo ao abri-lo, é possível realizar o *logout* (Figura 21).

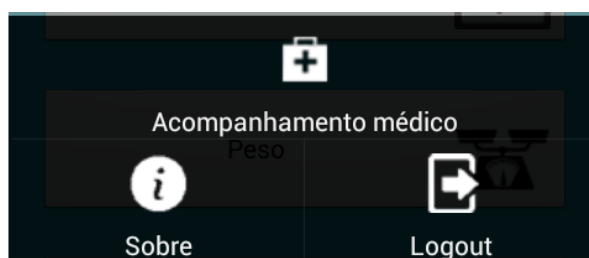


Figura 21: *Logout* do aplicativo
Fonte – Elaborada pelo autor

Ao acessar este método, o usuário *logado* é retirado das preferências compartilhadas do aplicativo e redirecionado para a tela de *login*. Sendo assim, será necessário que o usuário entre com seu *e-mail* e senha novamente para acessar suas informações no aplicativo.

3.2. Web Service

É uma aplicação Web que foi desenvolvida para disponibilizar serviços para gerenciamento e cadastramento das informações no banco de dados MySQL®, auxiliando o aplicativo Controle de Saúde e o painel Web.

Para que os registros sejam identificados no *Web Service*, foram adotadas duas validações. Data e hora da criação que é composta por ano, mês, dia, hora, minuto e segundo. E o *status*, onde é determinado o estado do registro, conforme o Quadro 2.

Quadro 2: *Status* do registro

ID	Estado	Detalhes
0	Exportado	O registro foi exportado para o <i>Web Service</i> e já é possível visualizá-lo no painel Web.
1	Novo	O registro foi inserido no aplicativo, mas ainda não foi exportado para o <i>Web Service</i> .
2	Editar	O registro foi atualizado no aplicativo e precisa ser atualizado no <i>Web Service</i> .
3	Excluir	Foi solicitado à exclusão do registro no aplicativo, ele ficará invisível até ser excluído no <i>Web Service</i> e depois será excluído no aplicativo.

As classes do *Web Service* apanham o que foi passado por GET na requisição enviada pelo aplicativo, realizam a inserção, consulta, atualização e deleção dos registros no banco de dados e depois retornam o resultado da operação.

3.3. Painel de visualização Web

O painel de visualização Web do Controle de Saúde foi desenvolvido para facilitar visualização de todos os dados inseridos pelo aplicativo, fornecendo assim outro canal para visualização das medidas cadastradas pelo aplicativo *mobile*.

Para facilitar a visualização dos processos incluídos no painel, foi criado um fluxograma (Figura 22).

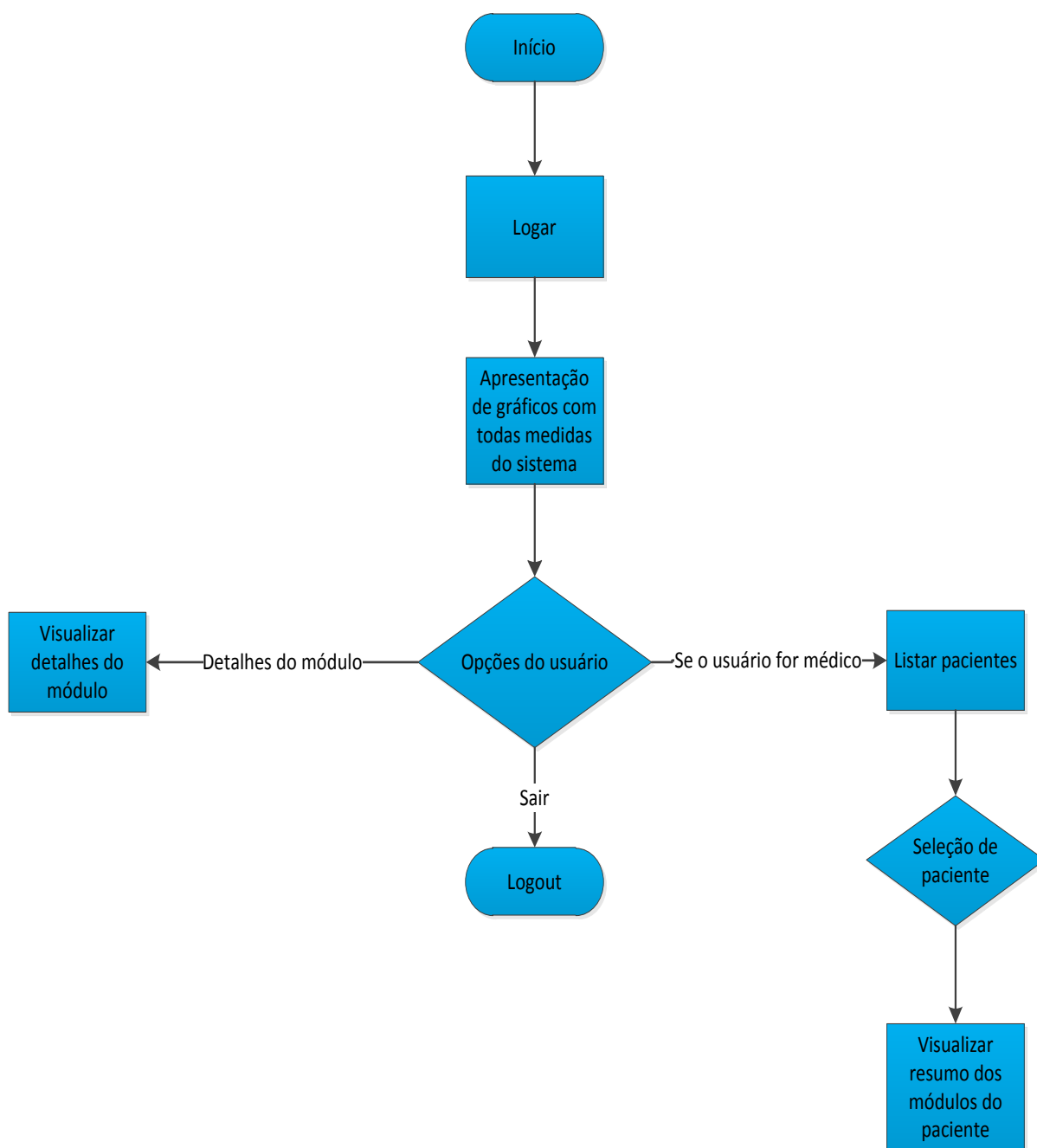


Figura 22: Fluxograma do painel Web
Fonte – Elaborada pelo autor

Este painel de acesso Web foi desenvolvido em PHP, JavaScript, HTML (*Hypertext Markup Language*) e CSS (*Cascading Style Sheets*).

O acesso ao painel é realizado pelas mesmas informações de cadastro do sistema *mobile* (Figura 23).

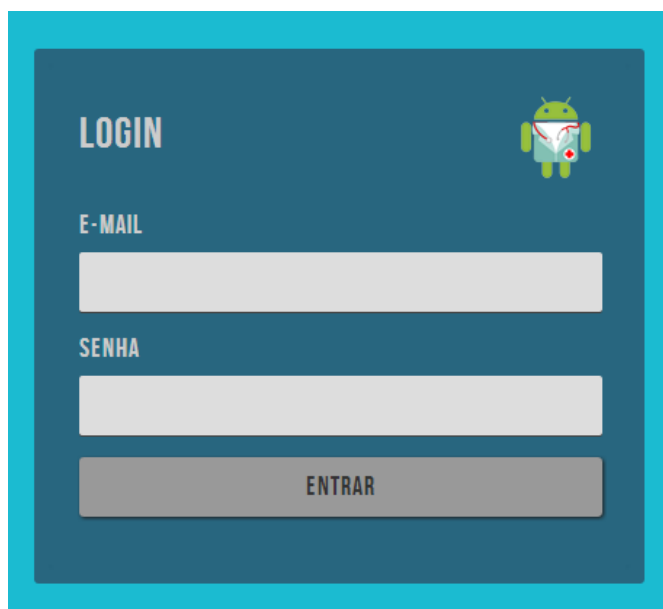
A imagem mostra uma interface de login com um fundo azul escuro e uma borda azul clara. No topo esquerdo, o texto "LOGIN" aparece em letras brancas maiúsculas. No topo direito, há um ícone de um robô verde com um estetoscópio. Abaixo do título, há dois campos de entrada de texto brancos: o primeiro é rotulado "E-MAIL" e o segundo "SENHA", ambos em letras brancas maiúsculas. No rodapé, há um botão cinza com o texto "ENTRAR" em letras brancas maiúsculas.

Figura 23: Tela de *login* do painel Web
Fonte – Elaborada pelo autor

Na tela principal, já é possível monitorar todos os módulos do Controle de Saúde com gráficos de linha (gerados com a biblioteca Google Charts) dos últimos 10 registros inseridos no aplicativo (Figura 24). Essa visualização foi projetada para facilitar o usuário no acompanhamento de suas medidas. Caso não existam registros em algum módulo ou eles ainda não foram exportados para o *Web Service*, no lugar do gráfico irá ser exibida uma mensagem “Informações não cadastradas”. No rodapé do painel é exibida a data e hora da última sincronização realizada entre o aplicativo e o *Web Service*.



Figura 24: Tela principal do painel Web

Fonte – Elaborada pelo autor

Na lateral esquerda (Figura 24) existe o menu de navegação para acessar cada módulo de controle do aplicativo e no canto superior direito tem o botão “Sair” para fazer o *logout* do painel e ser redirecionado para a tela de *login*.

Acessando um dos módulos de controle (Figura 25), na parte superior é apresentado um sumário das medidas realizadas, tais como, a maior, menor e a média dos valores cadastrados. Em seguida são apresentados os valores cadastrados, em forma de tabela, onde, o usuário poderá realizar buscas por qualquer informação do registro. Ao final da página é apresentado o gráfico com as medidas em maior escala para auxiliar na monitoração de suas medidas. Caso não tenha nenhum registro cadastrado no módulo de monitoração acessado, será exibida uma mensagem avisando o usuário.

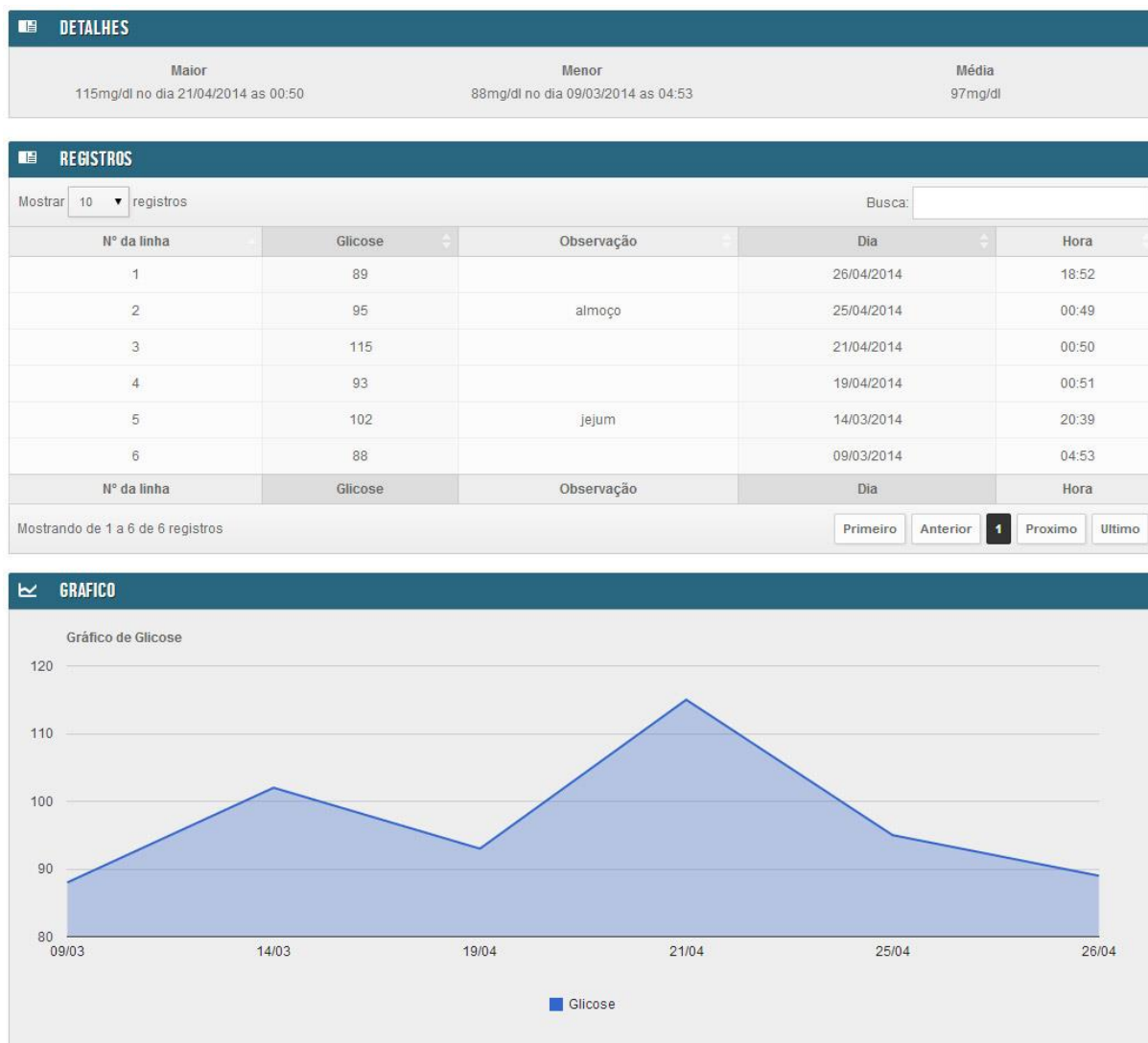


Figura 25: Detalhes do módulo do painel Web

Fonte – Elaborada pelo autor

O acompanhamento dos pacientes só estará disponível para os usuários que tenha o perfil de médico. Para ter este perfil é necessário que o usuário tenha selecionado a opção “Médico” no campo “Tipo de usuário” na hora do cadastro no aplicativo (Figura 26).

Figura 26: Seleção do tipo de usuário na tela de cadastro
Fonte – Elaborada pelo autor

Caso o usuário *logado* tenha o perfil de médico, poderá acessar a opção “Pacientes” no menu de navegação (Figura 27), serão listados todos os pacientes que solicitaram o acompanhamento deste médico e ele tenha confirmado por meio do *link* que foi enviado pelo *e-mail*.

Nº da linha	Nome	Email	Sexo	Data de nascimento
1	[Masked]	[Masked]	Feminino	22/02/1991
2	[Masked]	[Masked]	Feminino	22/02/1991

Figura 27: Lista de pacientes
Fonte – Elaborada pelo autor

As informações dos usuários na Figura 27, foram mascaradas para garantir a confiabilidade das informações, mas tem o intuito de mostrar a lista de pacientes, com informações básicas para que o médico possa realizar o acompanhamento.

O médico acessando um dos pacientes listados poderá monitorar os últimos registros inseridos pelo usuário em todos os módulos, através de gráficos iguais os da tela principal do painel.

4. RESULTADOS

O aplicativo Controle de Saúde ficou no Google Play® por três meses (Figura 28), nesse período alcançou 1196 *downloads*, das quais 244 instalações são atuais.

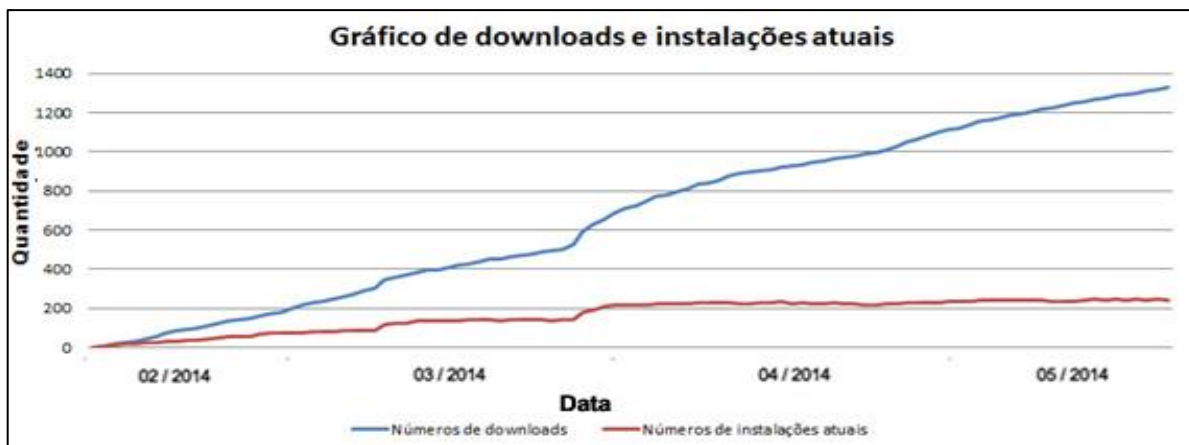


Figura 28: Relatório de *downloads* e instalações atuais do aplicativo.

Fonte – Elaborada pelo autor

Analisando a Figura 29, é possível observar as instalações e desinstalações diárias que ocorreram no aplicativo. A média diária de instalações é de 13,7 e a de desinstalações é de 10,2.

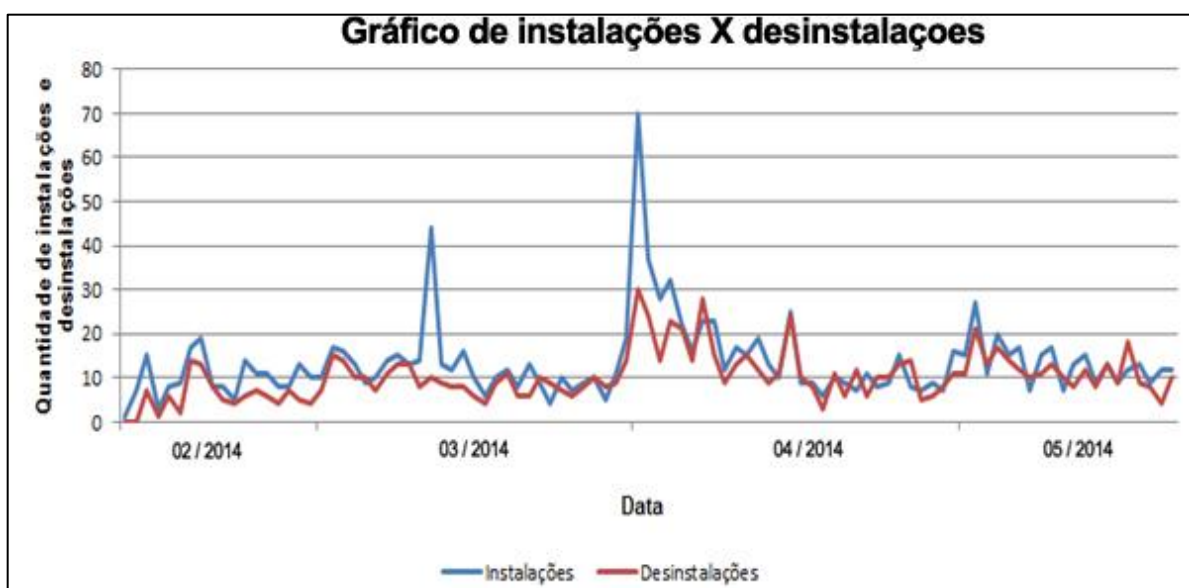


Figura 29: Instalações e desinstalações diárias

Fonte – Elaborada pelo autor

No Console do desenvolvedor é possível acompanhar os países onde os usuários estão utilizando o aplicativo (Figura 30). Neste caso, pode-se observar que a maior adesão foi do mercado brasileiro, entretanto houve interesse de 11,45% de outros países.

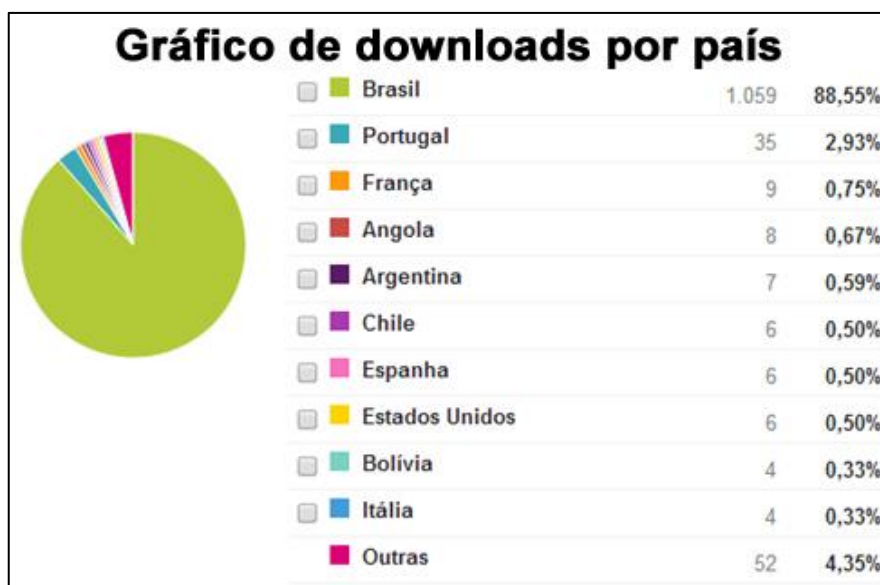


Figura 30: Países que realizaram o *download*
Fonte – Elaborada pelo autor

O Google Play® permite que os usuários classifiquem os aplicativos. Neste contexto, foram realizadas 47 avaliações (Figura 31) no aplicativo Controle de Saúde, obtendo 83% das notas máximas.



Figura 31: Classificação do aplicativo
Fonte – Elaborada pelo autor

O aplicativo conta com 729 usuários cadastrados, na Figura 32 é possível observar os cadastros separados por mês, com uma média de 243 cadastros.

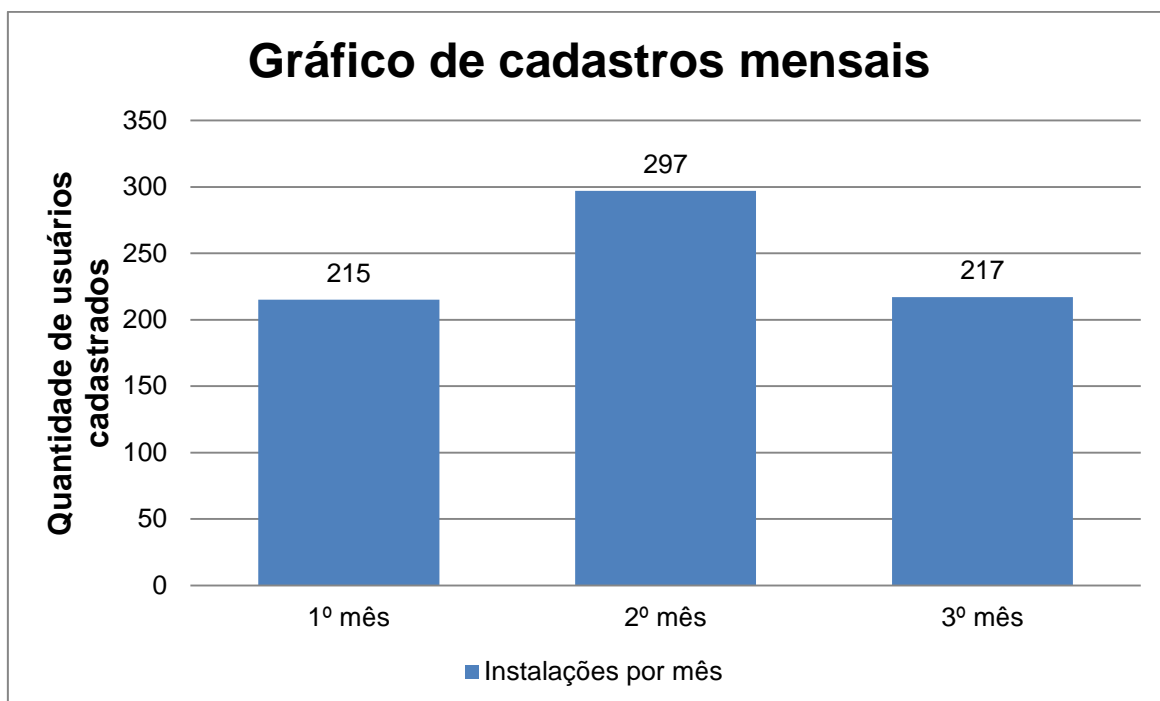


Figura 32: Usuários cadastrados por mês
Fonte – Elaborada pelo autor

Dentre os usuários do sistema, pode-se observar que existe uma equivalência na utilização do sistema pelos diferentes sexos, sendo 51% dos usuários do sexo feminino (Figura 33).

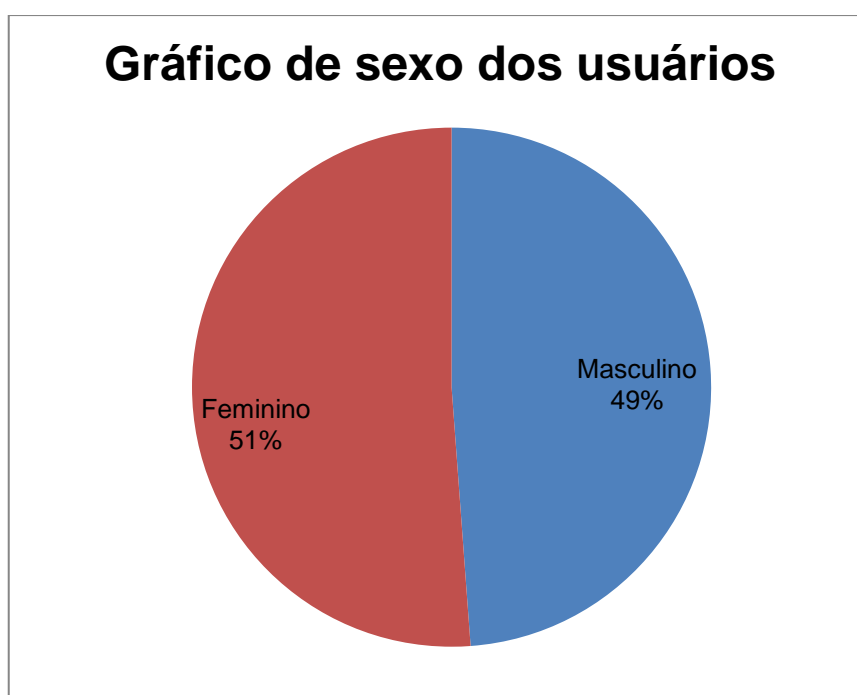


Figura 33: Sexo dos usuários
Fonte – Elaborada pelo autor

Na Figura 34, é possível observar que a faixa etária dos usuários que mais utilizam o aplicativo é entre 30 e 50. Observa-se também que a média geral dos usuários é de 26 anos.

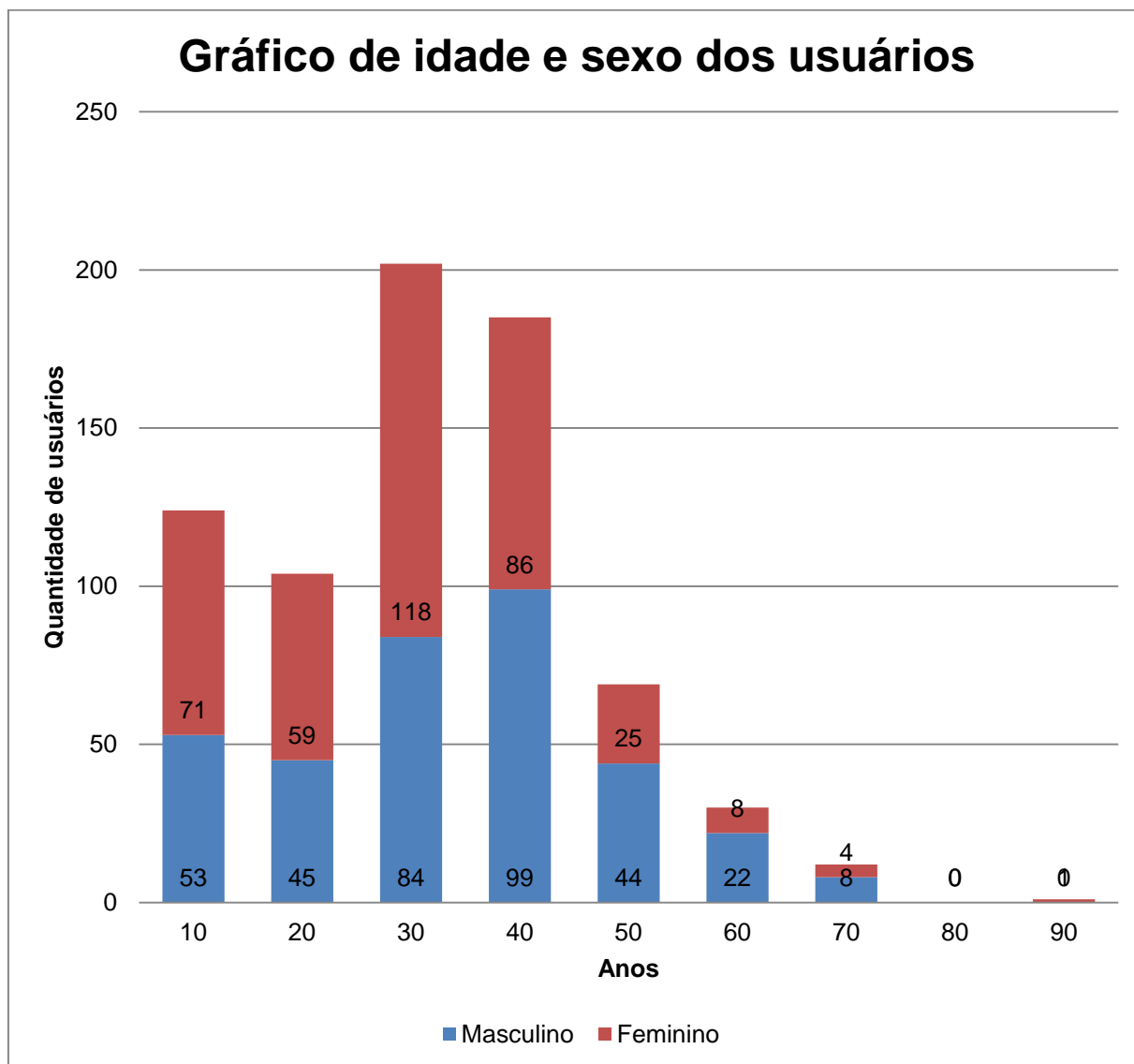


Figura 34: Idade e sexo dos usuários
Fonte – Elaborada pelo autor

Na Figura 34 existem apenas 727 usuários, 2 deles foram retirados desta contagem, pois se cadastraram com uma data de nascimento futura. Após ter constatado essa ação, optou-se por corrigir este erro e atualmente o sistema faz essa validação e não permite mais que um usuário se cadastre com uma data de nascimento inválida.

O sistema inicialmente foi pensado somente para o acompanhamento do usuário básico, entretanto com o passar do desenvolvimento foi sugerido à inclusão de um módulo adicional para o acompanhamento do médico, o qual é apresentado na Figura 35. Neste perfil se cadastraram 31 usuários, totalizando 4% dos usuários ativos.



Figura 35: Cadastros no perfil médico e paciente
Fonte – Elaborada pelo autor

Na base de dados, existem 1729 registros de saúde (Figura 36). Considerando os usuários ativos, temos uma média de 7,1 registros por usuário e uma média de 576,33 registros por mês. Lembrando que os registros de saúde contabilizados são apenas os que foram exportados para o *Web Service*.

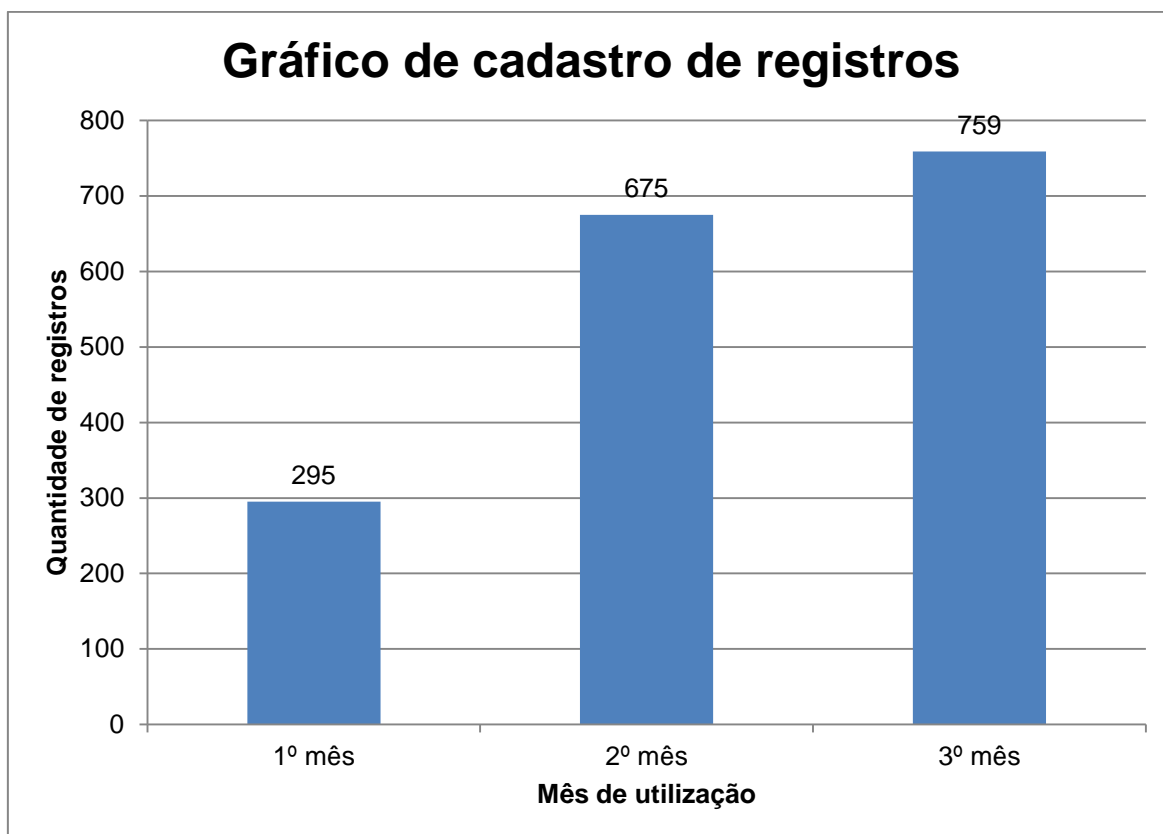


Figura 36: Registros inseridos por mês
Fonte – Elaborada pelo autor

Na Figura 37 é apresentado o gráfico sobre os percentuais de utilização por funcionalidade. O módulo mais usado do aplicativo foi o de Glicose com 25% de utilização.

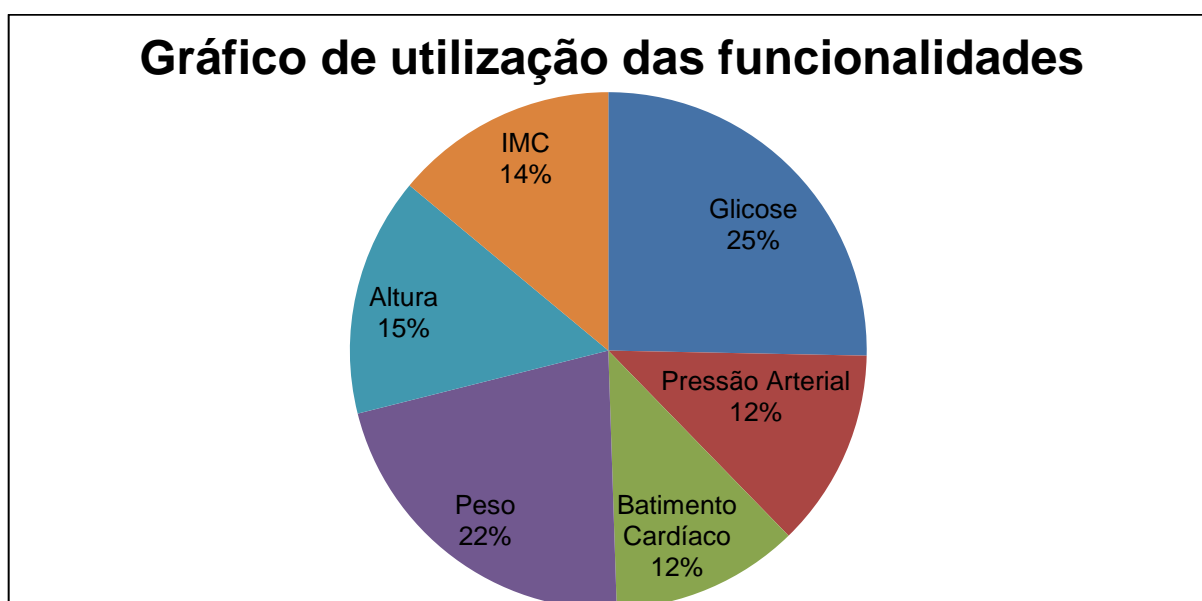


Figura 37: Gráfico com percentagem de utilização do sistema por funcionalidades
Fonte – Elaborada pelo autor

Segundo as informações obtidas pelo Console do desenvolvedor, a versão do Android® mais utilizada para fazer o *download* do aplicativo é a 4.1 com 34,15%, seguida da versão 2.3.3, como pode ser visto na Figura 38. Inclusive existe uma ampliação da menor faixa, onde é maximizado para exemplificar que nessa pequena porção, estão embutidas mais de três versões do Android®.

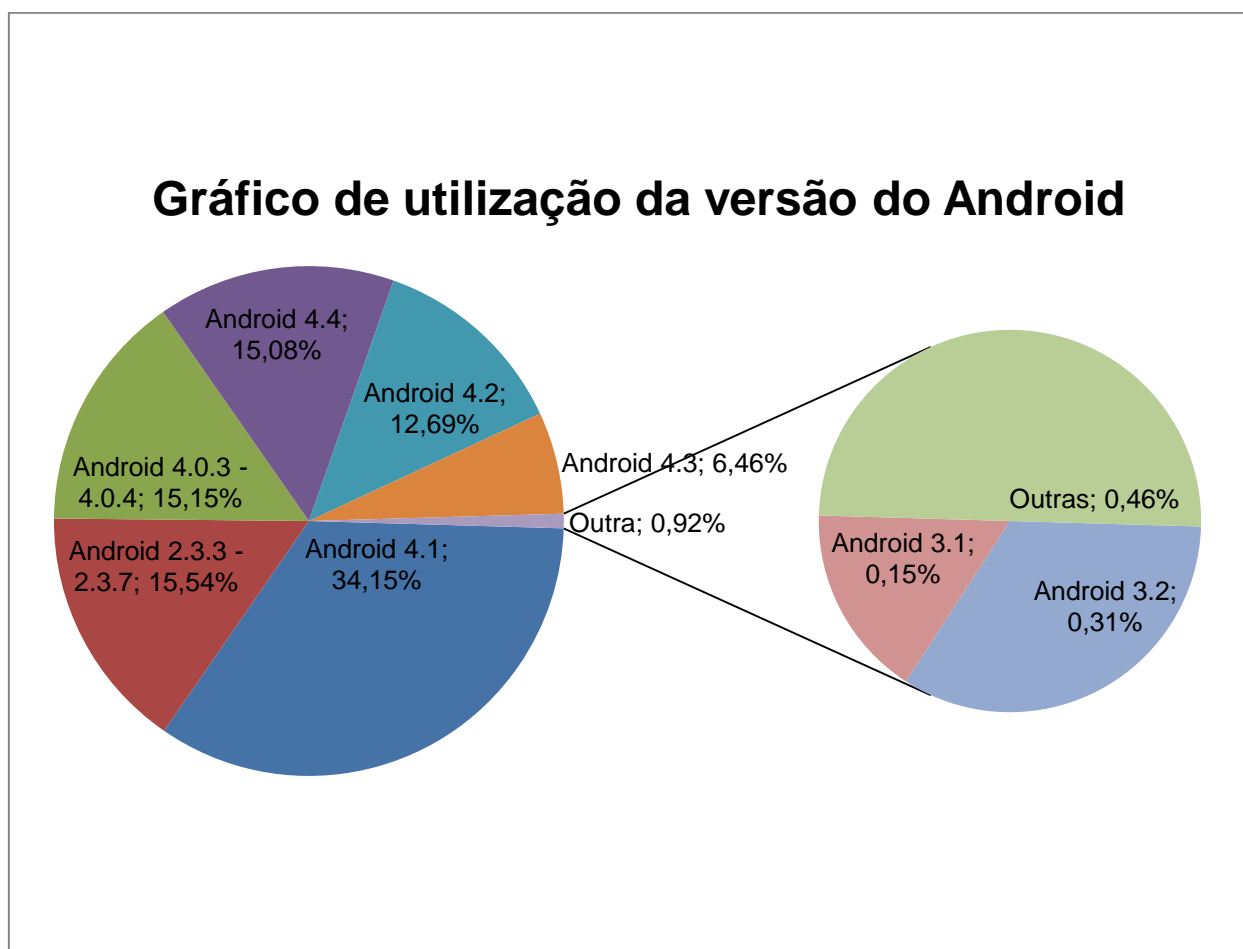


Figura 38: Gráfico da versão do Android® mais utilizada no aplicativo
Fonte – Elaborada pelo autor

O painel de visualização foi elaborado após a versão inicial de desenvolvimento do aplicativo *mobile*. Assim sendo, manteve-se em operação por aproximadamente 1 mês. Por mais que não tenha sido divulgado amplamente, onze usuários utilizaram o sistema Web.

5. CONCLUSÕES

O aplicativo Controle de Saúde foi disponibilizado para *download* gratuito no Google Play[®] através do endereço: <https://play.google.com/store/apps/details?id=br.com.matheuspiscioneri.controledesaude>.

O desenvolvimento do aplicativo, do *Web Service* e do painel Web foram rápidos, graças a grande quantidade de materiais disponibilizados em tutoriais, livros e fóruns especializados.

O aplicativo teve um grande número de *downloads*, mas na primeira impressão, não teve uma boa aceitação pelos usuários. Isso pode ser observado nas instalações atuais que não acompanharam o número de *downloads*, isso fez com que a média de instalações e desinstalações não tivesse uma grande diferença. Essa quantidade de desinstalações pode ter sido causada pelo fato do usuário não ter se adaptado com a *interface* gráfica. Inclusive em uma comunidade de desenvolvedores Android[®], foi sugerido melhoras em relação ao *layout* do aplicativo, para seguir o padrão mais atual de desenvolvimento de *interface* do Android[®], com o uso de *ActionBar*, *Navigation Tabs*, dentre outros.

Depois da utilização do sistema, foi observado que em diversos países como Portugal, França, Angola, Argentina, Chile, Espanha e Estados Unidos houve uma procura pelo aplicativo, o que na verdade não foi tão grande e poderia ter sido melhor se tivesse sido desenvolvido a internacionalização do sistema, para melhorar o entendimento dos usuários que falam outras línguas.

Analisando as estatísticas do Console do Desenvolvedor, somente 4% dos usuários classificaram o aplicativo, isso dificulta o processo de análise. Nas avaliações recebidas, algumas foram ruins, porém todos os comentários deixados pelos usuários foram de elogios e nenhum deles explicava o porquê de algumas más classificações. Isso pode ter ocorrido pelo fato de não existir uma divulgação ampla sobre a importância de classificar um aplicativo no Google Play[®].

Apenas 61% dos usuários que fizeram o *download*, realizaram o cadastro. Isso pode ter ocorrido devido à tela de cadastro, que pode intimidar o usuário no primeiro acesso. Uma solução que pode vir a ser desenvolvida é a realização do *login* por meio das contas do Facebook[®] ou do Google+, o que irá automatizar o preenchimento dos dados pessoais pelo usuário.

A percentagem do sexo dos usuários cadastrados foi semelhante, mas pode se perceber que até os 30 anos de idade tem uma concentração maior do sexo feminino e acima dos 30 anos, a predominância é do sexo masculino.

Dentre os módulos de controle oferecidos pelo sistema, o que teve a maior procura foi o de Glicose, o que mostra a grande quantidade de usuários que estão preocupados com o controle do diabetes.

Apesar de não ter sido feita a divulgação da ferramenta de acompanhamento médico, o aplicativo obteve 4% dos cadastros com este perfil, visando uma maior praticidade do profissional, poderá ser desenvolvido um método de monitoração diretamente no aplicativo.

A quantidade de registros exportados ao *Web Service* teve um crescimento contínuo, o que mostra que os usuários ativos estão utilizando o aplicativo para monitorar e controlar a saúde.

A versão Jelly Bean 4.1 do Android® é a que ocupa o primeiro lugar no número de *downloads* do aplicativo, registrando um total de 34,15% de aparelhos rodando esta versão. Enquanto isso, a Gingerbread 2.3.3 está em segundo lugar, alcançando uma porção de 15,54% dos usuários. Essas estatísticas são úteis para analisar melhor quais são as preferências de dispositivos e a compatibilidade de componentes e bibliotecas que poderão ser utilizadas.

O painel de visualização Web foi muito pouco acessado. Isso se deve a falta de divulgação dessa ferramenta, principalmente pelo fato de terem sido desenvolvidas depois da criação do aplicativo e sua propaganda não ter sido explorada corretamente. Mesmo com a falta de utilização dos usuários, essa ferramenta pode ter um grande potencial no controle e acompanhamento da saúde se for usado em conjunto com o aplicativo.

Com base nos resultados obtidos com o sistema, é possível concluir que há uma grande procura por aplicativos na área da saúde, o que mostra a preocupação das pessoas com o bem-estar e a qualidade de vida.

6. REFERÊNCIAS BIBLIOGRÁFICAS

ANDROID. Android Developers. Disponível em: <<http://developer.android.com/guide/index.html>>. Acesso em: 01 mar 2014.

ANDROID. Brand Guidelines. Disponível em: <<http://developer.android.com/distribute/googleplay/promote/brand.html>>. Acesso em: 04 mar 2014.

ANDROID. Creative Vision. Disponível em: <<http://developer.android.com/design/get-started/creative-vision.html>>. Acesso em: 04 mar 2014.

ANDROID. Dashboards. Disponível em: <<http://developer.android.com/about/dashboards/index.html>>. Acesso em: 04 mar 2014.

BARRA, D. C. C.; NASCIMENTO, E. R. P.; MARTINS, J. J.; ALBUQUERQUE, G. L.; ERDMANN, A. L. Evolução histórica e impacto da tecnologia na área da saúde e da enfermagem. **Rev. Eletr. Enf**, v. 8, n. 3, p. 422-430, 2006. Disponível em: <http://www.fen.ufg.br/revista/revista8_3/v8n3a13.htm>. Acesso em: 23 fev 2014.

BONIFÁCIO, B.; FERNANDES, P.; SANTOS, F.; OLIVEIRA, H. A. B. F.; CONTE, T. **Usabilidade de aplicações web móvel: avaliando uma nova abordagem de inspeção através de estudos experimentais**. 2010. Disponível em <http://cibse.inf.puc-rio.br/CIBSEpapers/artigos/artigos_CIBSE12/paper_38.pdf>. Acesso em: 04 mar 2014.

BREGA, J. R. F. B.; LAURIS, J. R. P.; MOREIRA, P. R. C.; PEREIRA, R. C. Levantamento Epidemiológico em Saúde Bucal Utilizando Ferramentas Móveis. Disponível em: <<http://dohs.codeplex.com>>. Acesso em: 03 mar 2014.

CANALYS. 2014. Smart phones, Worldwide, installed base by OS vendor, 2014. Disponível em: <<http://www.canalys.com/chart/index.html#display-98>>. Acesso em: 02 mar 2014.

DIAS, M. S.; BITTAR, T. J.; MENDONÇA, V. R. L. Um estudo dos Sistemas Operacionais Android e iOS para o desenvolvimento de aplicativos. Disponível em: <http://www.enacomp.com.br/2011/anais/trabalhos-aprovados/pdf/enacomp2011_submission_54.pdf> Acesso em: 02 mar 2014.

EXAME. 2013. Vendas de smartphones têm crescimento espetacular no Brasil. Disponível em <<http://exame.abril.com.br/tecnologia/noticias/8-3-milhoes-de-smartphones-sao-vendidos-no-segundo-trimestre>> Acesso em: 02 abr 2014.

GONÇALVES, E. Trabalhando com banco de dados. Desenvolvendo aplicações web com Netbeans IDE 6. Ciencia Moderna, 2008.

IBGE. 2008. Acesso a internet e posse de telefone móvel celular para uso pessoal. Disponível em: <<http://www.ibge.gov.br/home/estatistica/populacao/acesoainternet2008/internet.pdf>>. Acesso em: 23 fev 2014.

IBM. 2010. Introdução ao desenvolvimento de Android usando widgets Eclipse e Android. Disponível em: <<http://www.ibm.com/developerworks/br/opensource/tutorials/os-eclipse-androidwidget>>. Acesso em: 23 mar 2014.

KATZ, J. E.; RICE, R. E.; ACORD, S. Usos da Internet e de Tecnologias Móveis nos Sistemas de Saúde: abordagens sociais e organizacionais num contexto comparativo. Disponível em: <http://www.egov.ufsc.br/portal/sites/default/files/anexos/a_sociedade_em_rede_-_do_conhecimento_a_acao_politica.pdf#page=174>. Acesso em: 04 mar 2014.

LECHETA, R. R. Introdução ao Android. In: _____. Google Android. 3. ed. São Paulo: Novatec, 2013.

Lee, V.; Schneider, H. e Schelli, R. (2005) “Aplicações Móveis - Arquitetura, projetos e desenvolvimento”. São Paulo: Pearson Education do Brasil.

LOPES, C. J. F.; RAMALHO, J. C. *Web Services: Metodologias de Desenvolvimento*. Disponível em: <<http://repositorium.sdum.uminho.pt/bitstream/1822/559/1/LR04.pdf>> Acesso em: 03 mar 2014.

MÄHLMANN, F. G. G.; WAGNER, R. N. Sistema de gestão de produção de eventos. Disponível em: <<http://guaiba.ulbra.br/seminario/eventos/2012/artigos/sistemas/salao/1013.pdf>>. Acesso em: 02 abr 2014.

MASCARENHAS, M.; MARTINS, M.; BULÇÃO, L.; BRITO, J.; VIEIRA, V.; DURAN, A. Um estudo de caso com análise comparativa entre plataformas para aplicações móveis aberta e proprietária: Android e iOS. 2013. Disponível em: <<http://www.cin.ufpe.br/~ubibus/artigos/112186.pdf>>. Acesso em: 02 mar 2014.

MORIMOTO, C. E. Smartphones, Guia Prático. GDH Press, 2009.

OHA. Alliance Overview. Disponível em: <http://www.openhandsetalliance.com/android_overview.html>. Acesso em: 01 mar 2014.

OLIVEIRA, A. P.; FRAGOMENI, B. N.; NOGUEIRA, M. H.; IOSIMUTA, N.; VASCONCELLOS, V. H. **Smartcompras: desenvolvimento de um aplicativo para Celulares smartphone**. 2011. Disponível em: <<http://engenharia.anhembis.br/tcc-11/cc0-04.pdf>>. Acesso em: 01mar 2014.

OLIVEIRA, O. L. Estudo comparativo entre tecnologias de desenvolvimento Android e Java me. 2012. Disponível em: <<http://www.bibliotecadigital.ufmg.br/dspace/bitstream/handle/1843/BUOS-94MLQJ/eavndrolopesoliveira.pdf>>. Acesso em: 21 mai 2014.

ORACLE. Model-View-Controller. Disponível em: <<http://www.oracle.com/technetwork/java/mvc-140477.html>> Acesso em: 02 abr 2014.

PEREIRA, L. C. O.; SILVA, M. L. Android para Desenvolvedores. Rio de Janeiro: BRASPORT, 2009.

PHP. O que é PHP? Disponível em: <http://www.php.net/manual/pt_BR/intro-what-is.php> Acesso em: 03 mar 2014.

PINHEIRO, N. A. M.; SILVEIRA, R. M. C. F.; BAZZO, W. A. Ciência, tecnologia e sociedade: a relevância do enfoque CTs para o contexto do ensino médio. **Rev Ciência & Educação**, v.13, n.1, p.71-84, 2007. Disponível em: <<http://www.scielo.br/pdf/ciedu/v13n1/v13n1a05.pdf>>. Acesso em: 01 mar 2014.

QUEIROZ, A. H. B. Agenda de contatos em sistema web com sincronização via web service para celulares com sistema operacional Android. 2011. Disponível em: <http://repositorio.roca.utfpr.edu.br/jspui/bitstream/1/1196/1/CT_JAVA_VI_2010_02.PDF>. Acesso em: 03 mar 2014.

ROMEIRO, B. G. B. A. **Desenvolvimento de aplicativos para dispositivos móveis na plataforma J2ME**. 2005. Disponível em: <<http://tcc.ecomp.poli.br/BrunaRomeiro.pdf>>. Acesso em: 23 fev 2014.

RTC. Android Goes Beyond Google. Disponível em: <<http://www.rtcmagazine.com/articles/view/102586>>. Acesso em: 22 mar 2014.

SIGULEM, D.; SALOMÃO, P. Utilização do Computador de Mão Integrado à Telefonia Celular no Atendimento Médico: Desenvolvimento de Sistema e Avaliação. Disponível em: <<http://www.sbis.org.br/cbis9/arquivos/59.doc>>. Acesso em: 04 mar. 2014.

SILVA, C. G.; BORGES, I. S.; CHERUBINI, L. F.; MARTINS, R. M. **Aplicação do sistema operacional Android em robótica móvel**. 2013. Disponível em: <http://www.fadep.br/engenharia-eletrica/congresso/pdf/117889_1.pdf>. Acesso em: 23 fev 2014.

SILVA, P. F.; WAISSMANN, W. Normatização, o estado e a saúde: questões sobre a formalização do direito sanitário. *Ciênc. saúde coletiva*, v.10, n.1, p237-244, 2005. Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1413-81232005000100030&lng=pt&nrm=iso>. Acesso em: 01 mar. 2014.