

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL  
INSTITUTO DE INFORMÁTICA  
CURSO DE CIÊNCIA DA COMPUTAÇÃO

FÁBIO MALET PORTELA

**Mecanismos para análise de potência de sinal em dispositivos móveis**

Monografia apresentada como requisito parcial  
para a obtenção do grau de Bacharel em Ciência  
da Computação.

Orientador: Prof. Dr. Valter Roesler  
Coorientador: William Lautenschläger

Porto Alegre  
2015

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitor: Prof. Carlos Alexandre Netto

Vice-Reitor: Prof. Rui Vicente Oppermann

Pró-Reitor de Graduação: Prof. Sérgio Roberto Kieling Franco

Diretor do Instituto de Informática: Prof. Luís da Cunha Lamb

Coordenador do Curso de Ciência da Computação: Prof. Raul Fernando Weber

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Haro

## AGRADECIMENTOS

Em primeiro lugar, gostaria de agradecer aos meus pais, Ana Cristina e Napoleão, que fizeram o possível e o impossível para dar-me a melhor educação que esteve ao alcance deles. Sem o esforço, afeto e amor deles, nunca teria chegado tão longe assim. Este agradecimento especial estende-se também às minhas irmãs, Stéfani e Ana Caroline, que juntas dos meus pais, edificaram o conceito de família em minha vida.

Agradeço também aos meus avós, Rosa e Celso, que serviram de suporte aos meus pais nas horas de maior aperto, participando e auxiliando ativamente em boa parte da minha formação acadêmica.

Gostaria de agradecer também ao meu orientador, o Professor Doutor Valter Roesler, e ao meu coorientador, William Lautenschläger, que colaboraram bastante para este período de Trabalho de Conclusão ser o melhor possível.

Sou grato também a todos os familiares, paternos e maternos, que de alguma maneira ajudarem-me nesta longa caminhada.

Não posso deixar de dar um agradecimento especial a todos os amigos que formei durante o curso de graduação. Integrantes do Clube de Regatas Bem Na Humildade, colegas gremistas, parceiros de Olímpico e Arena e seus agregados(as), meu muito obrigado.

Não poderia deixar de lembrar também dos grandes amigos que fiz no Colégio Tiradentes, e seus agregados, que foram de fundamental importância para o período empregado neste curso.

Agradeço também a todos os professores que marcaram a minha caminhada durante o curso de graduação. Lembrar-me-ei para sempre dos bons mestres.

Por fim, agradeço a todos aqueles que, de alguma forma, contribuíram para me tornar bacharel no curso de Ciência da Computação.

## RESUMO

Este trabalho apresenta o estudo de mecanismos para análise de potência de sinal em dispositivos móveis e a elaboração de um aplicativo que auxilie na localização da maior potência de sinal possível entre as analisadas. Serão abordados conceitos técnicos acerca de redes de computadores e desenvolvimento Android. Utilizando-se abstrações de Engenharia de Software, será projetado e desenvolvido um aplicativo que resolva o problema central dessa monografia. Em seguida, serão apresentadas ideias de melhorias futuras para a ferramenta elaborada. Por fim, serão citados projetos semelhantes ao desenvolvido neste trabalho, relacionando-os e apontando vantagens e desvantagens destacáveis entre eles.

**Palavras-chave:** Análise de Sinais de Redes Móveis; Mecanismos de Localização; Redes de Computadores; Desenvolvimento Android; Engenharia de Software.

## **Mechanisms for signal strength analysis on mobile devices**

### **ABSTRACT**

This paper presents a study of mechanisms for signal strength analysis on mobile devices and the elaboration of an application to help locating the strongest signal possible among the analyzed networks. Technical concepts about computer networks and Android development will be discussed. Using Software Engineering abstractions, an app that solves the central problem of this work will be designed and developed. Then, ideas for future improvements on the created app will be presented. Finally, projects similar to the one developed in this work will be referred, showing relations and pointing advantages and disadvantages among them.

**Keywords:** Mobile Network Signals Analysis; Location Mechanisms; Computer Networks; Android Development; Software Engineering

## LISTA DE FIGURAS

Figura 2.1 – Localização das antenas do dispositivo Samsung Galaxy S5 .....	18
Figura 2.2 – Ciclo de vida de uma instância da classe <i>Activity</i> .....	27
Figura 3.1 – Tela principal do aplicativo Antennas .....	32
Figura 3.2 – Representação de todos os sinais de rede Wi-Fi alcançáveis pelo dispositivo móvel, fornecido pelo aplicativo <i>WiFi Analyzer</i> .....	33
Figura 3.3 – Mapa das torres de telefonia celular da região no aplicativo OpenSignal .	36
Figura 3.4 – Mapa de Calor dos sinais de telefonia móvel de Porto Alegre .....	37
Figura 4.1 – Diagrama de Casos de Uso .....	39
Figura 4.2 – Diagrama de Sequência da Tela Inicial do aplicativo .....	42
Figura 4.3 – Diagrama de Sequência da ação de Salvar Local.....	43
Figura 4.4 – Diagrama de Sequência das ações de Listar Locais Salvos e Visualizar Local .....	44
Figura 4.5 – Diagrama de Sequência das ações de Listar Redes Salvos e Visualizar Rede .....	45
Figura 4.6 – Diagrama de Classes referente à base de dados da aplicação .....	47
Figura 5.1 – Trecho de código utilizando código de API mais recente.....	50
Figura 6.1 – Tela inicial do aplicativo .....	54
Figura 6.2 – Menu principal exibido na tela inicial.....	55
Figura 6.3 – Tela que exibe opção de “Salvar Local” .....	56
Figura 6.4 – Tela de listagem de Locais Salvos .....	57
Figura 6.5 – Tela de detalhamento de um local salvo (“Quarto Fábio”).....	58
Figura 6.6 – Tela de listagem de Redes Salvos .....	59
Figura 6.7 – Tela de detalhamento de uma rede salva (Rede Móvel “Vivo”).....	60
Figura 6.8 - Tela de detalhamento de uma rede salva (Rede Wi-Fi “malets”).....	61
Figura 6.9 – Tela de informações do aplicativo .....	62
Figura 6.10 – Escolha de servidor de e-mail em opção de contato ao desenvolvedor ...	64

## LISTA DE ABREVIATURAS E SIGLAS

AMPS	<i>Advanced Mobile Phone System</i> (Sistema de Telefone Móvel Avançado)
API	<i>Application Programming Interface</i> (Interface de Programação de Aplicativos)
ASU	<i>Arbitrary Strength Unit</i> (Unidade de Potência Arbitrária)
AVD	<i>Android Virtual Device</i> (Dispositivo Virtual Android)
CCC	Central de Comutação Celular
CDMA	<i>Code Division Multiple Access</i> (Acesso Múltiplo por Divisão de Código)
dB	Decibel
dBm	Decibel-miliwatt
EDGE	<i>Enhanced Data rates for GSM Evolution</i> (Taxas de Dados Ampliadas para a Evolução do GSM)
EVDO	<i>EVolution-Data Optimized</i> (Evolução de Dados Otimizados)
ERB	Estação Rádio-Base
IDE	<i>Integrated Development Environment</i> (Ambiente de Desenvolvimento Integrado)
GSM	<i>Global System for Mobile Communications</i> (Sistema Global para Comunicações Móveis)
GPRS	<i>General Packet Radio Service</i> (Serviço de Rádio de Pacote Geral)
HSDPA	<i>High-Speed Downlink Packet Access</i> , em português, Pacotes Downlink com Alta Velocidade de Acesso
HSUPA	<i>High-Speed Uplink Packet Access</i> , em português, Pacotes Uplink com Alta Velocidade de Acesso
LTE	<i>Long-Term Evolution</i> (Evolução de Longo Prazo)

NMT	<i>Nordic Mobile Telephone</i> (Telefonia Móvel Nórdica)
SDK	<i>Software Development Kit</i> (Kit de Desenvolvimento de Software)
SIM	<i>Subscriber Identity Module</i> (Módulo de Identificação do Assinante)
TACS	<i>Total Access Communication System</i> (Sistema de Comunicação de Acesso Total)
TDMA	<i>Time Division Multiple Access</i> (Acesso Múltiplo por Divisão de Tempo)
UFRGS	Universidade Federal do Rio Grande do Sul
UMTS	<i>Universal Mobile Telecommunications System</i> (Sistema Universal de Telecomunicações Móveis)
WLAN	<i>Wireless Local Area Network</i> (Rede Sem-Fio Local)
XML	<i>eXtensible Markup Language</i> (Linguagem de Marcação Extensível)



# SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO .....</b>	<b>12</b>
<b>1.1</b>	<b>Objetivo .....</b>	<b>12</b>
<b>1.2</b>	<b>Justificativa .....</b>	<b>13</b>
<b>1.3</b>	<b>Visão Geral.....</b>	<b>13</b>
<b>2</b>	<b>CONCEITOS BÁSICOS .....</b>	<b>16</b>
<b>2.1</b>	<b>Conceitos de Redes de Computadores.....</b>	<b>16</b>
2.1.1	<i>Antenas.....</i>	<i>16</i>
2.1.2	<i>Ondas de Rádio.....</i>	<i>19</i>
2.1.3	<i>Decibel (dB).....</i>	<i>19</i>
2.1.4	<i>Decibel-miliwatt (dBm).....</i>	<i>20</i>
2.1.5	<i>Redes de Telefonia Móvel .....</i>	<i>22</i>
2.1.6	<i>Redes Wi-Fi.....</i>	<i>24</i>
<b>2.2</b>	<b>Conceitos de desenvolvimento <i>Android</i> .....</b>	<b>25</b>
2.2.1	<i>APIs Android e compatibilidade retroativa .....</i>	<i>25</i>
2.2.2	<i>Activity .....</i>	<i>26</i>
2.2.3	<i>View.....</i>	<i>28</i>
2.2.4	<i>Wi-Fi Manager.....</i>	<i>28</i>
2.2.5	<i>Telephony Manager .....</i>	<i>29</i>
2.2.6	<i>Intent .....</i>	<i>29</i>
2.2.7	<i>Android Manifest.....</i>	<i>29</i>
<b>3</b>	<b>TRABALHOS RELACIONADOS .....</b>	<b>31</b>
<b>3.1</b>	<b>Antennas.....</b>	<b>31</b>
<b>3.2</b>	<b>WiFi Analyzer .....</b>	<b>33</b>
<b>3.3</b>	<b>OpenSignal.....</b>	<b>34</b>
<b>4</b>	<b>PROJETO.....</b>	<b>38</b>
<b>4.1</b>	<b>Diagrama de Casos de Uso .....</b>	<b>38</b>

4.1.1	Ator.....	39
4.1.2	Visualizar potências de sinal de todas redes alcançáveis.....	39
4.1.3	Salvar Local.....	40
4.1.4	Visualizar Locais Salvos.....	40
4.1.5	Visualizar Local.....	40
4.1.6	Visualizar Redes Salvas.....	40
4.1.7	Visualizar Rede.....	40
4.1.8	Contatar Desenvolvedor.....	41
<b>4.2</b>	<b>Diagramas de Sequência.....</b>	<b>41</b>
4.2.1	Tela Inicial (Visualizar potências de sinal de todas redes alcançáveis).....	41
4.2.2	Salvar Local.....	42
4.2.3	Visualizar Locais Salvos / Visualizar (Detalhar) Local.....	43
4.2.4	Visualizar Redes Salvas / Visualizar (Detalhar) Rede.....	45
<b>4.3</b>	<b>Diagrama de Classe.....</b>	<b>46</b>
<b>5</b>	<b>IMPLEMENTAÇÃO.....</b>	<b>48</b>
<b>5.1</b>	<b>Ambiente de Desenvolvimento.....</b>	<b>48</b>
<b>5.2</b>	<b>API Android escolhida.....</b>	<b>49</b>
<b>5.3</b>	<b>Wi-Fi Manager.....</b>	<b>50</b>
<b>5.4</b>	<b>Telephony Manager.....</b>	<b>51</b>
<b>5.5</b>	<b>Intent.....</b>	<b>52</b>
<b>5.6</b>	<b>Base de dados.....</b>	<b>52</b>
<b>6</b>	<b>FUNCIONALIDADES DESENVOLVIDAS.....</b>	<b>53</b>
<b>6.1</b>	<b>Tela Inicial.....</b>	<b>53</b>
<b>6.2</b>	<b>Menu Principal.....</b>	<b>55</b>
<b>6.3</b>	<b>Salvar Local.....</b>	<b>55</b>
<b>6.4</b>	<b>Locais Salvos.....</b>	<b>57</b>
<b>6.5</b>	<b>Redes Salvas.....</b>	<b>59</b>

6.6	Sobre o Aplicativo .....	62
7	DESENVOLVIMENTO FUTURO .....	65
7.1	Varredura de Rede .....	65
7.1.1	Desenvolvimento .....	66
7.2	Mapa de Calor de Rede.....	66
7.2.1	Desenvolvimento .....	67
7.3	Teste de Velocidade de Rede .....	67
7.3.1	Desenvolvimento .....	68
8	CONCLUSÕES .....	69
	REFERÊNCIAS .....	70

# 1 INTRODUÇÃO

O escopo deste trabalho visa tratar inúmeras dificuldades cotidianas relacionadas a redes móveis. Essas adversidades incluem desde uma simples busca por um melhor sinal de telefonia móvel para realizar uma chamada de voz, até uma mais complexa análise do melhor local onde pode ser posicionado um ponto de acesso Wi-Fi, dentro de um determinado ambiente.

Por conta disso, a necessidade por uma ferramenta de diagnóstico e assistência em relação às redes alcançáveis por dispositivos móveis vem crescendo consideravelmente com a popularização dos mesmos.

Assim, este trabalho encaixa-se nesse contexto, já que visa reduzir ao máximo as dificuldades citadas anteriormente, constituindo uma útil aplicação de análise de sinais de redes em aparelhos móveis.

Nos parágrafos que seguem, será descrito um roteiro inicial do que será apresentado na monografia, introduzindo o que será desenvolvido nas seções posteriores e destrinchando finalidade, motivação e metodologia adotadas durante o progresso do trabalho.

## 1.1 Objetivo

O presente trabalho foi concebido com o intuito de auxiliar o usuário na apuração do local onde a potência de sinal em seu dispositivo móvel seja a maior possível.

Em outras palavras, a principal finalidade desta monografia é a de apresentar ao usuário uma maneira de quantificar e qualificar a potência dos sinais alcançáveis pelo seu dispositivo móvel, detalhando mais o nível de sinal dessas redes em relação ao que é mostrado pelo aparelho de telefonia móvel, e expondo de maneira clara as constantes mudanças e oscilações da rede durante seus deslocamentos. Além disso, objetiva-se dar maior flexibilidade ao usuário para que o mesmo possa armazenar essas informações de redes alcançáveis pelo dispositivo móvel, relacionando-as a locais nomeados pelo próprio, podendo futuramente compará-los e, assim, determinar o local em que melhor possa realizar determinada tarefa que dependa de alguma das redes móveis alcançáveis pelo aparelho móvel.

Para tanto, foram estudados alguns mecanismos que solucionassem o problema central do trabalho, analisando o comportamento das redes móveis e elaborando maneiras mais intuitivas de apresentação das mesmas para o usuário. Posteriormente, foi elaborada a implementação de um método que trouxesse à prática a resolução deste problema, relacionando tudo o que havia sido previamente analisado, elaborado e projetado, para então, entregar um produto proficiente ao utilizador final.

## **1.2 Justificativa**

Com a expansão do mercado de *smartphones* e *tablets* e seu uso massivo pelo mundo, ascendeu também a necessidade de crescimento das infraestruturas de redes móveis mundialmente, principalmente para que essas estruturas pudessem suportar o aumento exponencial no uso de dados pelos usuários destes dispositivos móveis.

Entretanto, o desenvolvimento das estruturas de redes ao longo do planeta, principalmente no Brasil, não conseguiu acompanhar este crescimento massivo no uso de dados. Em outras palavras, as redes móveis constituídas nos últimos anos não conseguem satisfazer adequadamente o aumento de demanda fomentado pela população. Por conta disso, há inúmeras situações corriqueiras do dia-a-dia que deixam o usuário de redes móveis extremamente desconfortável, tais como, inúmeras áreas e zonas em que não há redes móveis disponíveis, oscilações de qualidade e potência do sinal de redes móveis geradas por causas naturais (vento, chuva, luz solar, etc.), ou por número excessivo de usuários conectados a uma determinada antena ou ponto de acesso Wi-Fi, entre outras.

Dessa forma, o trabalho desenvolvido encaixa-se no cenário descrito anteriormente, pois, a partir de estudos, análise e desenvolvimento de uma ferramenta própria, consegue confrontar muitas das dificuldades acima descritas, auxiliando o usuário na localização da maior potência de sinal das redes móveis alcançáveis pelo seu dispositivo.

## **1.3 Visão Geral**

O trabalho elaborado, e apresentado a seguir, é estruturado de forma descentralizada, em capítulos e seções que detalham todos os elementos pertencentes ao processo de composição da monografia.

No capítulo 2, “Conceitos Básicos”, são apresentados alguns conceitos básicos referentes ao conhecimento teórico adquirido ao longo do curso de graduação. Todos aqueles elementos de aprendizagem que fundamentaram o trabalho desenvolvido são devidamente descritos e elucidados buscando tornar o entendimento da monografia muito mais claro para o leitor.

O capítulo 3, “Trabalhos Relacionados”, apresenta uma relação de projetos anteriores a este trabalho e que possuem finalidade e objetivo semelhantes. Esses projetos são superficialmente descritos e amplamente comparados ao objetivo traçado para esta monografia, destacando-se semelhanças entre as propostas e apontando funcionalidades importantes ausentes nos projetos relacionados e que motivaram o desenvolvimento deste trabalho.

O capítulo 4, “Projeto”, é aquele em que são apresentados elementos que ajudam a projetar os fragmentos que compõem a ferramenta desenvolvida como parte prática deste trabalho. Em outras palavras, munidos de alguns conceitos de projeto e engenharia de software, são elaborados, e esmiuçados, diagramas que facilitam o entendimento dos componentes posteriormente desenvolvidos na aplicação resultante da parte prática desta monografia.

Baseado no que foi elaborado no capítulo anterior, o capítulo 5, “Implementação”, detalha o processo de desenvolvimento do aplicativo resultante da parte prática deste trabalho. Neste capítulo são relatados o ambiente de desenvolvimento utilizados, a API escolhida como base para a composição do aplicativo, as peculiaridades contidas no código e a descrição do processo das principais classes utilizadas no desenvolvimento da ferramenta.

O capítulo 6, “Funcionalidades desenvolvidas”, mostra um apanhado geral de todas as funcionalidades desenvolvidas para a aplicação final, detalhando-as para melhor guiar o usuário no uso do aplicativo. Cada tela, menu, ou peculiaridade gráfica apresentada ao usuário da aplicação é devidamente descrita e contextualizada com o que foi definido na monografia.

Já no capítulo 7, “Desenvolvimento Futuro”, elenca-se um conjunto de ideias concebidas durante a elaboração do trabalho e que não puderam ser desenvolvidas. Tais concepções foram devidamente documentadas e definidas para que possam ser

empreendidas no futuro. Além disso, há uma projeção superficial do modo como poderia ser elaborada a ideia, apontando as dificuldades que a levaram a não ser desenvolvida.

No capítulo 8, “Conclusões”, é feita uma reflexão de tudo o que foi feito durante o desenvolvimento desta monografia, apontando objetivos alcançados, dificuldades encontradas e concluindo acerca do trabalho realizado.

## 2 CONCEITOS BÁSICOS

Neste capítulo serão apresentados todos os conceitos teóricos nos quais o processo de desenvolvimento deste trabalho baseou-se. Tais conhecimentos utilizados como suporte durante a elaboração desta monografia serão superficialmente explicados, para que o leitor possa adquirir uma certa base científica que o ajude no entendimento do que será exposto posteriormente.

Para esse fim, reuniu-se todos os conceitos teóricos utilizados durante a realização deste trabalho e dividiu-os em dois grandes grupos: o de conhecimentos relacionados à área de rede de computadores e o de conhecimentos relacionados ao desenvolvimento de aplicativos Android e suas peculiaridades.

### 2.1 Conceitos de Redes de Computadores

Nesta seção serão apresentados todos os conhecimentos teóricos envolvidos no desenvolvimento deste trabalho e que se relacionam com a área de redes de computadores.

Entre os assuntos a serem abordados, engloba-se os conceitos mais importantes relacionados à comunicação de dados entre dispositivos móveis e pontos de acesso, sejam eles de telefonia móvel ou Wi-Fi. Para aprofundar um pouco o entendimento dessa comunicação de dados, são utilizados alguns artifícios relacionados à Física para compreender-se conceitos como a propagação de sinais pelo meio e a quantificação da intensidade deste sinal. Por fim, procura-se relatar as distinções entre redes de telefonia móvel e Wi-Fi, mostrando o modo como esse sinal pode ser calculado e o significado do valor informado.

#### 2.1.1 Antenas

Antenas são aparatos imprescindíveis para a comunicação dos dispositivos móveis com o mundo. Sem a existência delas, o envio e recepção de dados nos utensílios móveis seria impossível. Desde as antigas chamadas telefônicas de voz e mensagens de texto às modernas comunicações multimídia do popular mensageiro *WhatsApp*, toda a troca de informação entre um dispositivo móvel e um outro remetente faz uso de uma ou mais antenas.



Uma antena é um dispositivo elétrico que converte energia elétrica em ondas de rádio, e vice-versa. É comumente utilizada com um transmissor, ou um receptor, de rádio. Em uma transmissão, um transmissor de rádio fornece uma corrente elétrica oscilante em radiofrequência (isto é, uma alta frequência de corrente alternada) para os terminais das antenas, e a antena irradia a energia da corrente em forma de ondas eletromagnéticas. Já na recepção, uma antena intercepta parte da energia de uma onda eletromagnética com o intuito de produzir uma pequena tensão elétrica em seus terminais, que é prontamente repassada a um receptor de rádio para ser amplificada (GRAF, 1999).

Contextualizando um pouco mais com o trabalho desenvolvido, o dispositivo móvel, o qual é alvo dos estudos desta monografia, possui alguma(s) antena(s) para realizar comunicações com os demais aparelhos externos. O mais comum, entre os aparelhos móveis mais modernos, é que ele possua um número considerável de antenas, espalhadas pelas bordas do dispositivo. Cada uma dessas antenas tem funções específicas e é responsável pela recepção ou transmissão de ondas de rádio de uma determinada frequência, ou sistema de comunicação móvel.

No caso do Samsung Galaxy S5, por exemplo, há espalhados pelo aparelho sete antenas, cada uma delas responsável por tecnologias e frequências diferentes, como pode ser visto na Figura 2.1. A antena 1, chamada de “Antena Principal”, é aquela que trata dos sinais de telefonia móvel mais comuns, como CDMA (*Code Division Multiple Access*, em português, Acesso Múltiplo por Divisão de Código), EVDO (*EVolution-Data Optimized*, em português, Evolução de Dados Otimizados), GSM (*Global System for Mobile Communications*, em português, Sistema Global para Comunicações Móveis), GPRS (*General Packet Radio Service*, em português, Serviço de Rádio de Pacote Geral), EDGE (*Enhanced Data rates for GSM Evolution*, em português, Taxas de Dados Ampliadas para a Evolução do GSM), UMTS (*Universal Mobile Telecommunications System*, em português, Sistema Universal de Telecomunicações Móveis) e LTE (*Long-Term Evolution*, em português, Evolução de Longo Prazo). As antenas 2 e 7, tratam de alguns sinais específicos da tecnologia LTE, responsável pelo sinal 4G dos celulares atuais. As antenas 3 e 4, abrangem os sinais do tipo WLAN (*Wireless Local Area Network*, em português, Rede Sem-Fio Local), ou seja, essas antenas tratam de todos os sinais Wi-Fi ao qual o dispositivo móvel esteja comunicando-se. Já o Tx e o Rx, expostos na Figura 2.1, representam que a antena tem um transmissor de rádio (Tx) e/ou um

receptor de rádio (Rx). São estes elementos que auxiliam no processo de transformação de energia elétrica em ondas de rádio e vice-versa.

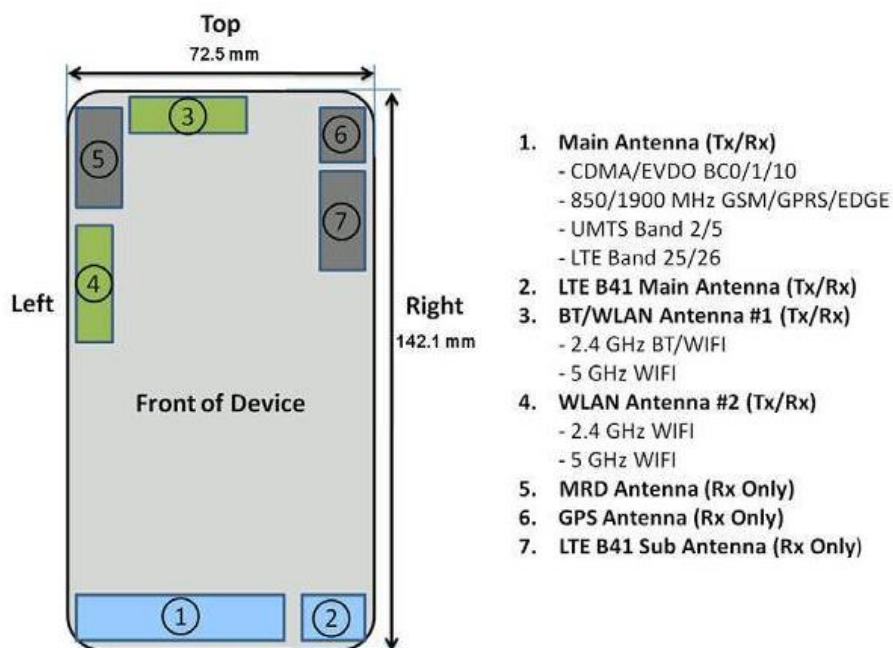


Figura 2.1 – Localização das antenas do dispositivo Samsung Galaxy S5  
(SHEPHERD, 2014)

Analogamente ao que fora exposto anteriormente em relação aos aparelhos móveis, os dispositivos com os quais eles se comunicam também possuem antenas. Como veremos no item que explica sobre Redes de Telefonia Móvel (ver item 2.1.5), o dispositivo móvel estabelece uma comunicação de dados móveis com uma torre de telefonia móvel. E, se no dispositivo móvel podemos ter uma ou mais antenas para tornar a comunicação de dados possível, o mesmo ocorre na torre de telefonia móvel, que pode ser superficialmente descrita como uma grande torre de concreto com uma ou mais antenas distribuídas em seu topo. Dessa forma, assim como a antena do dispositivo móvel tem a incumbência de receber ondas de rádio e transformá-las em correntes de energia elétrica, além de transmitir correntes de energia elétrica em forma de ondas de rádio, a antena da torre de telefonia móvel também executa essas tarefas para a transmissão e recepção dos dados aos quais é responsável.

Algo semelhante ocorre com as Redes Wi-Fi (assunto que será melhor abordado no item 2.1.6). Neste tipo de rede, a comunicação de dados é estabelecida entre um dispositivo móvel e um ponto de acesso Wi-Fi, e, assim como ocorre nas redes de

telefonia móvel, uma ou mais das antenas do dispositivo móvel responsável pela comunicação Wi-Fi comunica-se com uma ou mais antenas dos pontos de acesso Wi-Fi alcançáveis pelo dispositivo.

### 2.1.2 *Ondas de Rádio*

Ondas de rádio são um tipo de radiação eletromagnética com comprimento de onda suficientes para propagar informações em distâncias quilométricas (tipicamente atinge comprimentos de ondas entre 1 milímetro e 100 quilômetros). As distâncias atingidas dependem da frequência em que estas ondas estão propagando. As ondas de rádio tipicamente possuem frequências que variam entre 300 GHz e 3 kHz. E como todas as outras ondas eletromagnéticas, as ondas de rádio viajam na velocidade da luz (GRIFFITH, 2008).

Essas ondas de rádio são produzidas naturalmente por descargas elétricas de grande intensidade. Mas, o mais comum é que as ondas de rádio sejam geradas artificialmente, sendo utilizadas então por comunicação de telefonia fixa, telefonia móvel, radares, sistemas de navegação, comunicações via satélite, redes de computadores e em outras inúmeras situações.

No contexto de redes móveis, as ondas de rádio são responsáveis por transmitir informações entre dispositivos que não possuem ligação por fio. Oscilações de comprimento de onda ou na frequência são responsáveis por moldar os dados sendo transmitidos pelo ar entre dispositivos. Em outras palavras, como vimos anteriormente, as antenas transformam energia elétrica em ondas de rádio, e um conjunto dessas ondas de rádio produzidas pode ser responsável por carregar dados úteis (bits e bytes) entre dispositivos sem conexão com fio.

### 2.1.3 *Decibel (dB)*

O decibel, comumente representado pela sigla dB, é uma medida matemática muito mal compreendida e interpretada. Diferentemente de muitas outras representações que podem ser materializadas em unidades, como o quilograma, o centímetro ou o minuto, não podemos materializar 1 decibel. Isto ocorre devido ao fato de o decibel ser uma ordem de grandeza.

Em muitas áreas da tecnologia, precisamos comparar duas unidades de mesma grandeza para saber a variação quantitativa entre uma determinada variável em relação a um referencial de mesma grandeza. Para isso, a seguinte fórmula matemática exprime essa relação de forma simples:

$$V = 10 \log\left(\frac{x}{R}\right).$$

Na fórmula exposta acima,  $V$  é a variação resultante entre a variável  $x$  em relação ao referencial  $R$ ,  $x$  é a variável a ser comparada e  $R$  é o referencial utilizado para o cálculo.

Essa relação matemática é muito utilizada para variáveis de unidades de medida que variam excessivamente em ordens de grandeza. Em outras palavras, se há algum elemento largamente utilizado em alguma área de conhecimento e que possa ser encontrado em valores ínfimos, mas também encontrado em valores gigantescos, apresentá-lo em forma de decibéis facilita a visualização e a comparação, já que, tratar-se-á das potências de 10 destes valores.

Esses valores ínfimos possuem, por exemplo, ordens de grandeza equivalentes a mili, micro e nano que, em potência de 10, são expressos respectivamente por -3, -6 e -9. Essa mesma conversão acontece de maneira análoga para os valores gigantescos. Esses valores possuem, por exemplo, ordens de grandeza equivalentes a quilo, mega e giga, que, em potência de 10, são expressos respectivamente por 3, 6 e 9.

#### 2.1.4 Decibel-miliwatt (dBm)

O dBm, sigla largamente utilizada para representar o conceito de decibel-miliwatt, é uma abreviação para representar a taxa de potência em decibéis (dB) da potência medida referenciada a 1 (um) miliwatt (mW). Esta unidade de medida é utilizada em rádio, micro-ondas e redes de fibra ótica como uma medida conveniente de potência absoluta, por conta de sua capacidade de expressar tanto valores gigantescos quanto valores minúsculos em forma enxuta (HELD, 2002).

Utilizando a fórmula de decibel, apresentada na seção anterior, substituiremos:

- $V$  da variação resultante por  $N$  do nível de potência referido;
- $x$  do valor a ser comparado por  $P$  da potência de sinal bruta da amostra, em Watts;

- e o **R** do valor referencial de comparação por **1mW** (igual a  $10^{-3}$  W), valor referencial do dBm.

Desta forma, obtém-se a seguinte fórmula:

$$N = 10 \log\left(\frac{P}{1mW}\right).$$

Munidos desta fórmula, podemos reduzir a representação de importantes taxas de potência a simples valores de dBm, como veremos nos exemplos a seguir.

A potência de transmissão típica de uma estação de rádio FM com 50 quilômetros de distância é de  $100 \text{ kW} = 10^5 \text{ W}$  (SARKAR, SALAZAR-PALMA, & MOKOLE, 2008). Se mudarmos a representação desse valor para uma taxa de potência, em dBm, obteríamos um valor de 80 dBm, como demonstrado no cálculo a seguir:

$$N = 10 \log(10^5/10^{-3}) = 10 \log(10^8) = 10 \cdot 8 = 80.$$

A potência de sinal mínima recebida de uma rede sem fio (variantes do protocolo 802.11) é  $0.1 \text{ pW} = 10^{-13} \text{ W}$  (COLEMAN & WESTCOTT, 2012). Mudando a representação para uma taxa de potência em dBm, obteríamos após o cálculo um valor de -100 dBm.

Vale lembrar que, a relação do valor em decibel-miliwatt é diretamente proporcional a seu valor inteiro com sinal. Em outras palavras, quanto menor o valor absoluto, em dBm, pior ele é. Por exemplo, como veremos posteriormente no trabalho, um sinal Wi-Fi de -10 dBm é exorbitantemente melhor que um sinal Wi-Fi de -100 dBm (-10 dBm  $\gg$  -100 dBm).

Os exemplos citados anteriormente mostram o grande poder alcançado por essa mudança de representação. Valores muito altos resultam em valores positivos não muito grandes em módulo, em dBm, assim como valores muito baixos resultam em valores negativos não muito grandes em módulo, em dBm. E, desta forma, apenas uma ordem de grandeza de Watt é utilizada para quaisquer valores apresentados, tornando a comparação entre potências medidas mais simples e direta.

### 2.1.5 Redes de Telefonia Móvel

Uma rede de telefonia móvel, ou telefonia celular, é uma rede sem fio distribuída ao longo de áreas de terra chamadas células, cada uma delas servida por ao menos um transmissor de localização fixa, conhecido como estação base ou estação rádio-base (ERB). Em uma rede de telefonia móvel, cada célula usa um conjunto de frequências diferente de células vizinhas, para evitar interferência e fornecer largura de banda garantida no interior de cada célula (KHOSROW-POUR, 2006).

Juntas, estas células fornecem cobertura sobre uma enorme área territorial. Isto permite a um grande número de dispositivos portáteis (em geral, telefones celulares, tablets, modems 3G/4G, entre outros) comunicarem-se uns com os outros, com dispositivos fixos e com telefones, em qualquer lugar na rede, via estações rádio-base, até mesmo se algum dos dispositivos estiver movendo entre mais do que uma célula durante a transmissão.

Em uma rede de telefonia móvel, existem três elementos importantes que a constituem: Central de Comutação Celular (CCC), Estação Rádio-Base (ERB) e o terminal móvel.

A CCC é uma espécie de central telefônica digital que trata funções específicas para o funcionamento do sistema de telefonia móvel, como por exemplo o gerenciamento da comutação por circuito, alocação de frequência de sinal, entre outros.

Já a ERB é um sistema de rádio e antena de comunicação que permite a cobertura de uma área específica chamada de célula, como explicado anteriormente.

E o terminal móvel que se comunica através da rede é um dispositivo móvel utilizado pelo usuário final, aparecendo em forma de telefone celular, *smartphone*, *tablet*, *modem* 3G/4G, entre outros. Este dispositivo possui um sistema de comunicação sem fio particular e que evoluiu com o desenvolvimento da história da telefonia.

A primeira geração de sistemas de comunicação móvel (1G) apareceu no fim da década de 1970, principalmente com os sistemas AMPS (*Advanced Mobile Phone System*, em português, Sistema de Telefone Móvel Avançado) nos Estados Unidos, o TACS (*Total Access Communication System*, em português, Sistema de Comunicação de Acesso Total) na Inglaterra e o NMT (*Nordic Mobile Telephone*, em português, Telefonia

Móvel Nórdica) na Finlândia. Tratavam-se de sistemas analógicos e desta forma possuíam baixa capacidade, uma vez que permitiam apenas tráfego de voz e eram bastante vulneráveis à clonagem (KHOSROW-POUR, 2006).

A partir da segunda geração de sistemas de comunicação móvel (2G), a comunicação tornou-se digital, permitindo o tráfego de voz e dados pela rede. Nesta geração, destacam-se os sistemas TDMA (*Time Division Multiple Access*, em português, Acesso Múltiplo por Divisão de Tempo), CDMA e GSM. A principal diferença entre esses sistemas são as frequências em que operam – TDMA opera apenas em 850 MHz, CDMA em 850 e 1900 MHz, e GSM em 450, 850, 900, 1800 e 1900 MHz – e o fato do GSM introduzir o uso de Cartões SIM (*Subscriber Identity Module*, em português, Módulo de Identificação do Assinante) (os famosos “*chips*”) nos aparelhos (MISHRA, 2004).

Entre a segunda e terceira gerações, há uma geração intermediária, chamada de 2,5G, onde se destacam os sistemas GPRS e EDGE. Todos esses sistemas são evoluções sutis dos sistemas 2G, não mudando as frequências de operação utilizadas pelo sistema 2G mais moderno, o GSM. Mas, com algoritmos mais avançados e algumas técnicas modernas, as larguras de banda suportadas por essas tecnologias evoluíram um pouco em relação às anteriores. O sistema EDGE é considerado uma pequena evolução em relação ao GPRS.

Já na terceira geração (3G), mudou-se drasticamente a frequência de operação, utilizando-se a faixa de 2,1 GHz. Nesta geração, destacam-se os sistemas UMTS e HSDPA/HSUPA (*High-Speed Downlink/Uplink Packet Access*, em português, Pacotes Downlink/Uplink com Alta Velocidade de Acesso). Mesmo com frequência de operação diferente, o modelo UMTS é fortemente baseado no GSM, trazendo algumas inovações e otimizações interessantes que ajudaram a aumentar a largura de banda suportada pela tecnologia. Já o HSDPA/HSUPA é considerado uma evolução sutil do sistema UMTS, com relação semelhante à EDGE e GPRS.

Na quarta geração (4G), a frequência de operação foi modificada para a faixa de 2,5 GHz. Nesta geração, destaca-se o sistema LTE. A principal mudança nesse sistema é a utilização integral de tráfego de dados por comutação de pacotes, diferente de todos os sistemas anteriores que eram híbridos, e alternavam entre comutação por pacotes e

comutação por circuitos. Desta forma, o LTE torna a complexidade na infraestrutura de rede mais simples e ainda assim, mantém compatibilidade com todos os sistemas legados.

#### 2.1.5.1 Cálculo de potência de sinal de telefonia móvel

Um sinal de telefonia móvel é a potência de sinal (medida em dBm) recebida por um dispositivo móvel de uma rede de telefonia móvel. Dependendo de vários fatores, como a proximidade da torre de telefonia móvel, qualquer obstrução entre o dispositivo móvel e a torre, como prédios e árvores, entre outros, esta potência de sinal irá variar.

A potência do sinal medida pelo dispositivo móvel é geralmente apresentada em forma de um inteiro chamado ASU (*Arbitrary Strength Unit*, em português, Unidade de Potência Arbitrária). É possível calcular a potência de sinal real medida em dBm usando-se uma fórmula. Em redes GSM, onde o ASU é definido pelo RSSI (*Received Signal Strength Indicator*, em português, Indicador de Potência de Sinal Recebido), temos a seguinte fórmula para obter a potência do sinal em dBm:

$$P = 2 \cdot ASU - 113.$$

Na fórmula exposta anteriormente, P é a potência do sinal, medido em dBm, e ASU é a Unidade de Potência Arbitrária. O ASU é um número inteiro que está no intervalo entre 0 e 31, ou 99 para redes desconhecidas ou não detectáveis. Dessa forma, o intervalo de potências de sinal determináveis em dispositivos móveis é entre -51 dBm (ASU = 31) e -113 dBm (ASU = 0) (ETSI, 2008).

#### 2.1.6 Redes Wi-Fi

Wi-Fi é uma tecnologia de redes de computadores sem fio que permite aos dispositivos eletrônicos navegar na rede, principalmente usando frequências de rádio na faixa de 2,4 GHz e 5GHz (REID & SEIDE, 2003).

A Wi-Fi Alliance define Wi-Fi como qualquer produto WLAN baseado no padrão IEEE (Instituto de Engenheiros Elétricos e Eletrônicos) 802.11 (WI-FI ALLIANCE, 2000).

Muitos dispositivos podem usar Wi-Fi. Para tanto, esses necessitam conectar-se a um ponto de acesso de rede sem fio. Esses pontos de acesso possuem uma ou mais



antenas para comunicarem-se com os dispositivos conectados a ele. Toda informação recebida pela antena do ponto de acesso é processada e transmitida a cabo para um roteador conectado a este ponto de acesso. Analogamente, um ponto de acesso Wi-Fi envia informações recebidas de um roteador conectado a ele para um dispositivo móvel com o qual está comunicando-se. Essa troca de informações é feita por meio das antenas dos dispositivos envolvidos.

#### 2.1.6.1 Cálculo de potência de sinal Wi-Fi

Por ter uma estrutura mais simples, um único, e bem difundido padrão de implementação, e atender poucos usuários, a informação da potência de sinal da rede Wi-Fi é adquirida diretamente, sem necessidade de cálculo. Seu intervalo de valores possíveis vai de -10 dBm, a potência de sinal máxima em uma rede Wi-Fi à -100 dBm, a potência de sinal mínima em uma rede Wi-Fi (COLEMAN & WESTCOTT, 2012).

## 2.2 Conceitos de desenvolvimento *Android*

Nesta seção, serão apresentados os mais importantes conhecimentos da linguagem de programação Java, voltada para o desenvolvimento *Android*, e que ajudam no entendimento do que foi produzido na construção da ferramenta alvo da parte prática deste trabalho. Além disso, objetiva-se apresentar superficialmente elementos específicos da linguagem de programação, que são citados durante a monografia.

Entre os assuntos a serem abordados, destaca-se uma explicação superficial sobre as APIs (*Application Programming Interface*, em português, Interface de Programação de Aplicações) *Android*, suas implicações no desenvolvimento e a importância de uma correta escolha para o andamento do desenvolvimento da ferramenta. Serão versados também os conceitos das principais classes presentes no ambiente *Android* como *Activity*, *View*, *Intent*, além do arquivo de manifesto, o *Android Manifest*. Por fim, serão detalhados os *Android Managers* responsáveis pela coleta de informações dos sinais das redes de telefonia móvel e Wi-Fi.

### 2.2.1 APIs *Android* e compatibilidade retroativa

Uma API *Android* é, basicamente, o conjunto de bibliotecas, arquivos e padrões de programação definidos para o funcionamento de um aplicativo *Android*.

O Android divide as APIs desenvolvidas para seu sistema operacional em níveis, sendo que os níveis mais altos são mais recentes e incluem tudo do que foi desenvolvido para as APIs anteriores – ou quase tudo, já que alguns pouquíssimos métodos acabam tornando-se obsoletos ao longo do tempo. Cada API está também atrelada a uma versão diferente do sistema operacional Android. Atualmente, o Android tem como mais recente a API de nível 22, chamada de *Lollipop*, ou simplesmente, Android 5.1.

Cada modelo de celular possui uma versão de Android própria, podendo ser diferente de outros *smartphones*. Quanto mais antigo é um celular, mais antiga é sua versão de Android, menor é seu nível de API, e menos recursos do sistema operacional são disponibilizados para o usuário.

A escolha da API a ser utilizada no projeto de um aplicativo é algo a ser extensamente estudado antes do começo do desenvolvimento do mesmo. Como dito anteriormente, cada celular possui sua versão de Android, que traz uma API diferente. Se há modelos já atualizados para a última versão do Android, 5.1, com nível de API 22, há muitos celulares também com o Android 2.3.2, de nível de API 9. A versão de Android 2.3.2 é, inclusive, uma das mais populares no mundo atualmente, já que está presente na grande maioria dos *smartphones* de preços mais acessíveis.

Desta forma, quanto mais alto for o nível de API escolhido para desenvolver-se um aplicativo, menos usuários do sistema operacional serão atingidos pela ferramenta. Na contramão disso, quanto mais baixo for o nível de API utilizado, menos recursos disponíveis no Android poderão ser utilizados. Por isso que, como citado anteriormente, a escolha da API a ser utilizada deve ser criteriosa, bem estudada e tem grande importância no resultado final.

### 2.2.2 *Activity*

Uma Atividade (classe *Activity*) é uma ação única e focada que o usuário pode realizar durante a execução da aplicação. Quase todas atividades interagem com o usuário, então a classe *Activity* responsabiliza-se pela criação de uma janela onde é colocada a interface de usuário, descrita pelo desenvolvedor como um tipo de Visão (classe *View*, seção 2.2.3). Por estar diretamente relacionada com uma Visão, uma Atividade pode ser superficialmente descrita como o gerenciador do conjunto de ações do usuário em determinada tela.

Quando o aplicativo é iniciado pelo usuário, automaticamente é criada uma instância da atividade principal – aquela que está definida no Manifesto Android (seção 2.2.7). Esta atividade fica executando até que o usuário faça uma ação que a destrua (mude de atividade, feche o aplicativo, etc.) ou o próprio sistema operacional mate-a com o intuito de fornecer memória a algum aplicativo de maior prioridade.

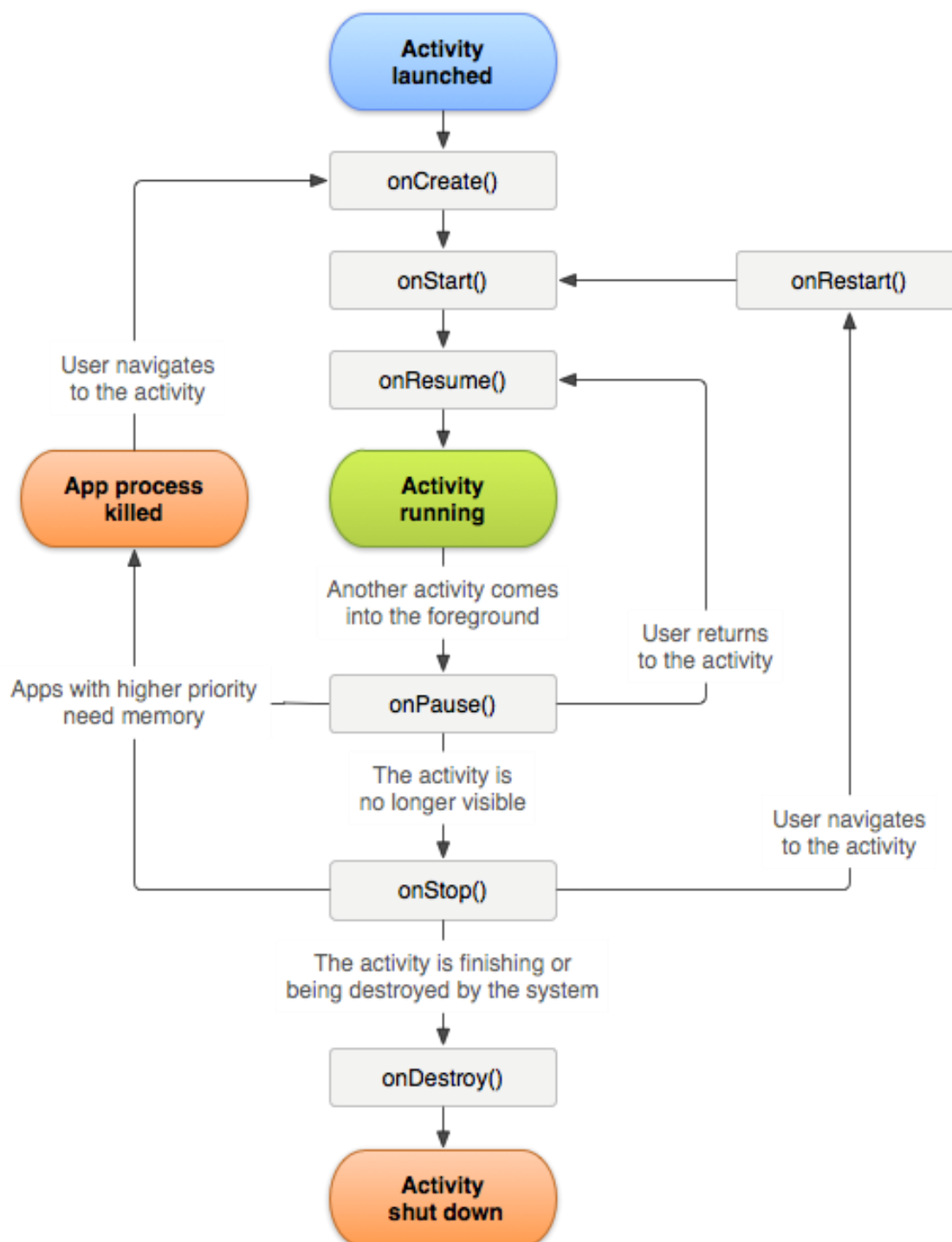


Figura 2.2 – Ciclo de vida de uma instância da classe *Activity*

(ANDROID, 2015)

A classe *Activity* é a principal classe de um aplicativo Android já que é ela quem gerencia a interação com o usuário, as mudanças na tela mostrada e a chamada de novas atividades. Por conta disso, a classe *Activity* tem um ciclo de vida bem definido e descrito na Figura 2.2.

O único método obrigatório desta classe, e por conta disso, o mais importante deles, é o método *onCreate()*. É nesse método que a Visão responsável pela interface do usuário é construída, as ações do usuário são tratadas, as opções do menu principal são criadas, novas atividades e *intents* (seção 2.2.6) são gerenciadas, entre outras ações possíveis.

### 2.2.3 *View*

Esta classe representa o bloco de construção básico de componentes de interface do usuário. Uma Visão ocupa uma área retangular na tela e é responsável pelo desenho da tela e pela manipulação de eventos. A Visão é a classe base para a criação de componentes de interface de usuário interativos (botões, caixas de texto, etc.).

A classe *View* é responsável por criar a tela de interface de usuário construída pelo desenvolvedor em um arquivo XML (*eXtensible Markup Language*, em português, Linguagem de Marcação Extensível) correspondente. No momento da criação de uma atividade, uma classe *View* é criada contendo o parâmetro do arquivo XML correspondente ao *layout* da tela desta atividade. A classe *View* então constrói todos elementos descritos no arquivo de *layout* fornecido, além de também construir os elementos definidos programaticamente.

A Visão também é responsável por gerenciar qualquer evento ocorrido na tela do aplicativo (clique de botão, mudança no estado de alguma entrada, etc.). Quando um desses eventos ocorre, a classe *View* trata o evento como definido pelo desenvolvedor na classe da Atividade em que ocorreu o evento.

### 2.2.4 *Wi-Fi Manager*

Esta classe é voltada para a realização de operações específicas de Wi-Fi. Quando criada uma instância dessa classe, é criada também uma *thread* executando em

segundo plano, que fica coletando informações sobre as redes Wi-Fi alcançáveis pelo dispositivo. Em havendo uma mudança nas informações coletadas pela *thread*, esta dispara um evento a ser tratado pela instância da classe. Desta forma, as informações coletadas pelo Wi-Fi Manager estão sempre atualizadas.

#### 2.2.5 *Telephony Manager*

Esta classe fornece acesso a informações sobre os serviços de telefonia presente no dispositivo. As aplicações podem usar os métodos nesta classe para determinar os estados e serviços de telefonia, assim como acessar alguns tipos de informações de assinaturas. As aplicações podem também registrar um *listener* para receber notificações de mudanças no estado da rede de telefonia. Se um *listener* for registrado, é criada uma *thread* em segundo plano que fica coletando informações sobre a rede de telefonia. Em caso de mudança no estado da rede, a *thread* envia um evento para o *listener* sinalizando que houve uma mudança no estado da rede de telefonia.

#### 2.2.6 *Intent*

Um *Intent* é uma descrição abstrata de uma operação a ser realizada. Como a própria tradução diz, essa classe é responsável por criar um elemento que informa ao sistema operacional uma “intenção” de realizar determinada tarefa, determinando o tipo desta tarefa e os argumentos da mesma. Essa classe pode ser usada para começar uma nova Atividade, para começar um serviço, para chamar um aplicativo que execute uma tarefa específica, entre outros.

#### 2.2.7 *Android Manifest*

O *Android Manifest* é um arquivo XML que define os parâmetros generalizadores do aplicativo. O tema utilizado, o esquema de cores do aplicativo, a versão de API mínima aceitável para executar a ferramenta, a versão de API indicada para o melhor desempenho do aplicativo, as permissões requeridas pelo aplicativo, ícone, título, e tantos outros parâmetros gerais são descritos neste arquivo. Funciona como uma espécie de “cabeçalho” do aplicativo.

Além disso, todas as atividades devem ser listadas no arquivo de Manifesto, pois nelas é que o usuário interagirá com o aplicativo. Na lista de atividades ainda devem ser definidos o título da atividade, o caminho onde encontra-se a classe que representa a

Atividade, os *intents* que podem chamá-la (ou seja, as “atividades-pai”), entre outros detalhes.

### 3 TRABALHOS RELACIONADOS

Neste capítulo serão apresentados alguns trabalhos que abordam assuntos semelhantes aos que fazem parte do escopo deste trabalho, mas que atacam o problema central do tópico versado de maneiras distintas.

Para cada trabalho apresentado, será feita uma breve descrição do objetivo do autor quando da concepção da monografia, serão descritas as principais funcionalidades do projeto e o mesmo será comparado ao que foi idealizado para este trabalho, a fim de apontar lacunas que venham a justificar o desenvolvimento desta monografia.

#### 3.1 Antennas

*Antennas* é um aplicativo simples que lê dados do gerenciador de redes de telefonia do aparelho móvel e mostra esses dados em um belo gráfico desenhado por cima do *Google Maps* (ANTENNAS, 2010).

O programa pode ser usado para dar uma ideia satisfatória das localizações das antenas em sua vizinhança, ajudando o usuário a comparar as redes de diferentes operadoras ou para verificar, em tempo real, como o dispositivo móvel seleciona a rede com a qual se conecta.

Como ilustrado na Figura 3.1, no ponto G, o aplicativo mostra a posição geográfica do dispositivo e mostra todas as antenas de rede de telefonia móvel alcançáveis pelo dispositivo, mostrando o valor do ASU (representação inteira positiva da potência do sinal, ver mais em 2.1.5.1) de cada uma dessas redes. A antena representada com a cor vermelha é a qual o dispositivo está conectado.



Figura 3.1 – Tela principal do aplicativo Antennas  
(ANTENNAS, 2010)

A principal vantagem do aplicativo é a de conseguir descobrir a localização e a potência do sinal em representação ASU, não só da antena conectada, mas também das antenas vizinhas alcançáveis pelo aplicativo.

Mas há também uma grande desvantagem: o aplicativo é praticamente obsoleto. O Antennas foi desenvolvido para o Android 1.0, primeira versão do sistema operacional da Google, utilizando-se uma API já ultrapassada e desatualizada. Por conta disso, o aplicativo só suporta algumas antigas redes de GSM, o que significa que funciona apenas para redes 2G. Outra limitação do aplicativo é a de não tratar redes Wi-Fi.



### 3.2 WiFi Analyzer

*WiFi Analyzer* é um aplicativo que lista visualmente ao usuário todas as redes Wi-Fi alcançáveis pelo seu dispositivo móvel, informando o nome da rede Wi-Fi, a potência do sinal dessa rede e o canal ao qual essa rede está associada.

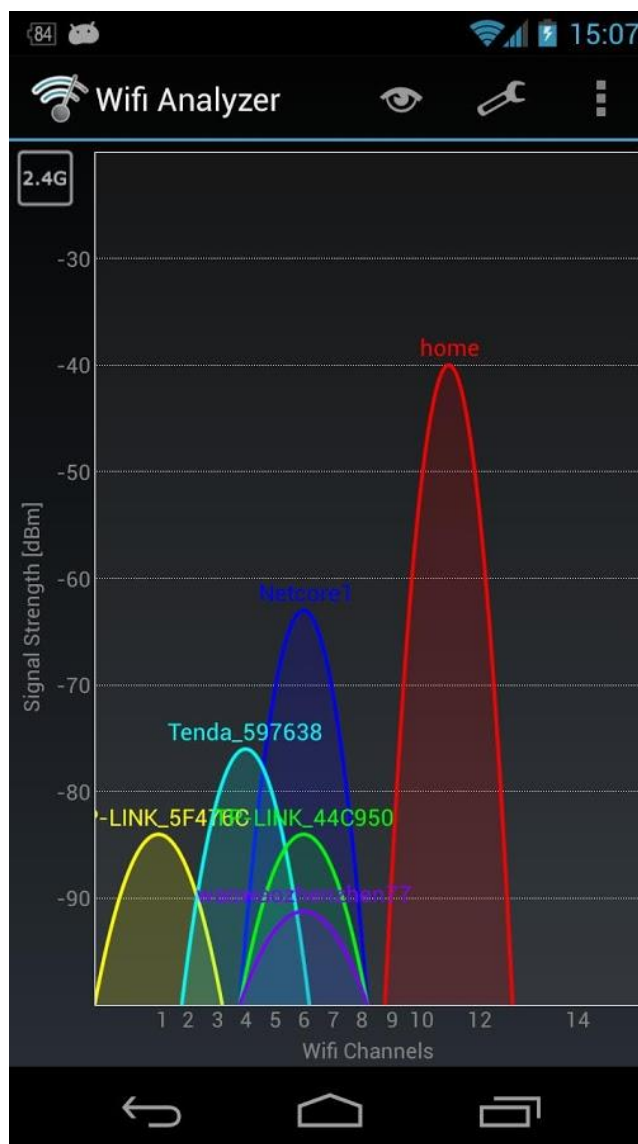


Figura 3.2 – Representação de todos os sinais de rede Wi-Fi alcançáveis pelo dispositivo móvel, fornecido pelo aplicativo *WiFi Analyzer* (WIFI ANALYZER, 2015)

Como pode ser observado na Figura 3.2, a tela do aplicativo está mostrando redes Wi-Fi com frequência de 2,4 GHz. O gráfico desenhado na tela é um plano cartesiano

positivo, onde no eixo das abscissas ficam os valores dos canais Wi-Fi possíveis e no eixo das ordenadas a potência do sinal Wi-Fi, em dBm.

Para inferir a representação do sinal Wi-Fi, você deve analisar o vértice da parábola formada e, assim, adquirir as informações referentes a rede Wi-Fi analisada.

No caso da rede “home” (parábola vermelha), ela está situada no canal Wi-Fi de número 11 e possui potência de sinal de -40 dBm.

A principal vantagem desse aplicativo é a de apresentar o canal onde se encontra a rede Wi-Fi analisada.

Em relação às desvantagens, podemos citar algumas. Primeiramente, o aplicativo peca em não trazer análise de redes de telefonia móvel nele. Outra coisa em que o aplicativo fica devendo é na representação difícil de ser interpretada. As parábolas possuem cores que não ajudam na visualização da representação e elas sobrepõem-se umas sobre as outras (em geral, se abrangerem o mesmo canal), impossibilitando a análise de algumas das redes.

### 3.3 OpenSignal

O *OpenSignal* é um aplicativo para o usuário mapear a cobertura de celular, encontrar pontos de acesso Wi-Fi e testar e melhorar a comunicação de dados do dispositivo. Ele ainda traz outras inúmeras funcionalidades como bússola de sinal apontando a direção de onde vem o sinal (auxiliando na busca por melhor sinal de determinada rede), mapa de redes Wi-Fi públicas alcançáveis, mapas de calor mostrando a cobertura da rede em determinada região (apresentado na Figura 3.4), mapa das torres de celular próximas (mostrado na Figura 3.3) e ranking de avaliação da melhor operadora disponível na região.

Para entregar todas essas funcionalidades, o aplicativo OpenSignal usa uma abordagem diferente para a avaliação da rede da região: faz uso de grande número de amostras dos usuários do aplicativo, que enviam informações, em *background*, para o banco de dados na nuvem do *OpenSignal*, a fim de contribuir com a melhora do mapeamento da região envolvida.

O aplicativo *OpenSignal* é muito robusto e completo. Uma das principais vantagens fica por conta do mapeamento de antenas em determinada região e da elaboração de mapas de calor de cobertura dessa mesma região.

Já as desvantagens ficam por conta da limitação do aplicativo em algumas situações. Para usar-se o *OpenSignal*, o usuário precisa estar conectado à Internet. Por conta disso, sua utilização fica impossibilitada em momentos que o dispositivo móvel não possui nenhuma conexão de dados estabelecida. E, em geral, esta situação é aquela em que o aplicativo mais poderia ser útil ao usuário.

Além disso, o aplicativo em questão é colaborativo já que necessita de dados de outros usuários para sua utilização. Ou seja, caso haja necessidade de usar o aplicativo em um local onde nenhum usuário contribuiu com o mapa colaborativo, o aplicativo perde sua utilidade.



Figura 3.3 – Mapa das torres de telefonia celular da região no aplicativo OpenSignal  
(OPEN SIGNAL, 2015)

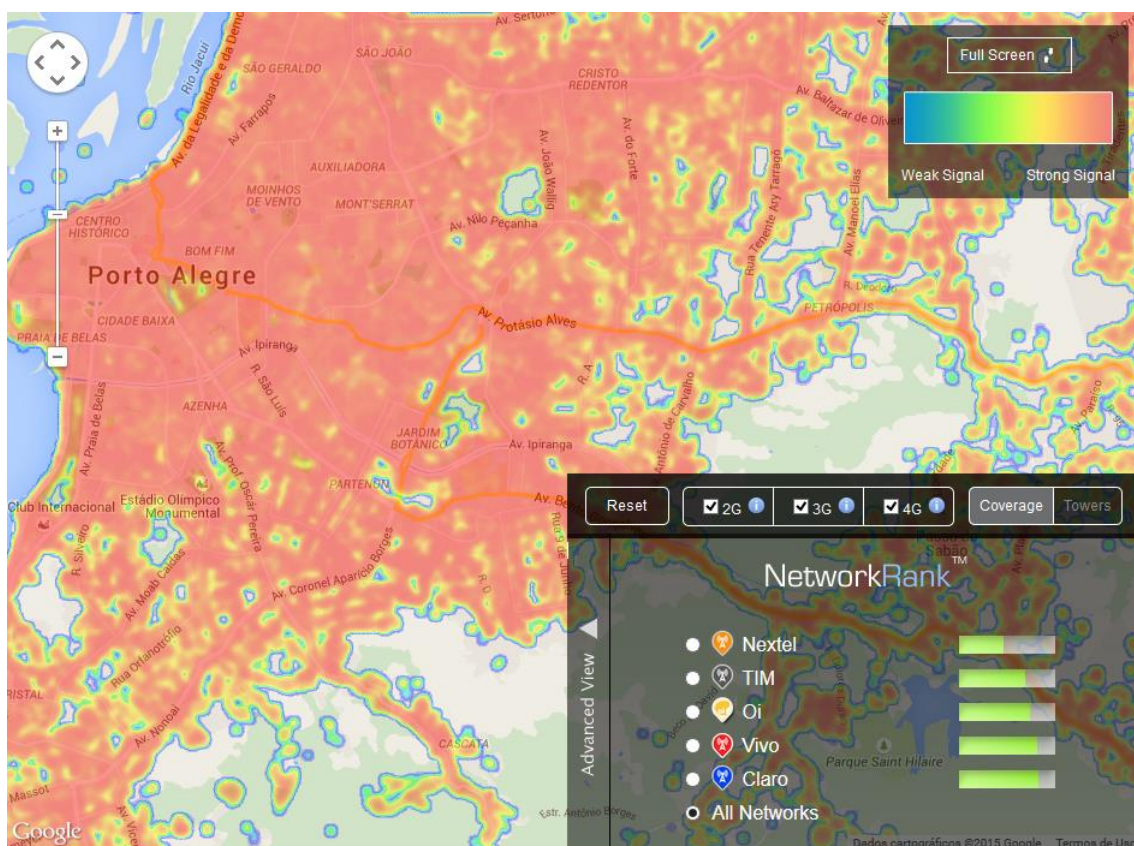


Figura 3.4 – Mapa de Calor dos sinais de telefonia móvel de Porto Alegre  
(OPEN SIGNAL, 2015)

## **4 PROJETO**

Conforme proposto no prefácio desta monografia, será apresentada uma implementação de um método para apuração do local onde a potência de sinal de um dispositivo móvel seja a maior possível, baseado no estudo de mecanismos que solucione o problema central do trabalho. Dessa forma, o aplicativo que compreende a parte prática desta monografia deve ser inicialmente projetado, antes de ser implementado, relacionando os elementos a serem explicitados aos conhecimentos adquiridos durante a graduação.

Assim, serão apresentados nas próximas seções alguns conceitos de Engenharia de Software que ajudarão a moldar o sistema final e permearão o posterior desenvolvimento da aplicação alvo.

### **4.1 Diagrama de Casos de Uso**

Para começar a definir o aplicativo a ser implementado neste trabalho, precisa-se definir todas as ações que serão disponibilizadas ao usuário no sistema final. Desta forma, foi elaborado o diagrama de casos de uso apresentado na Figura 4.1.

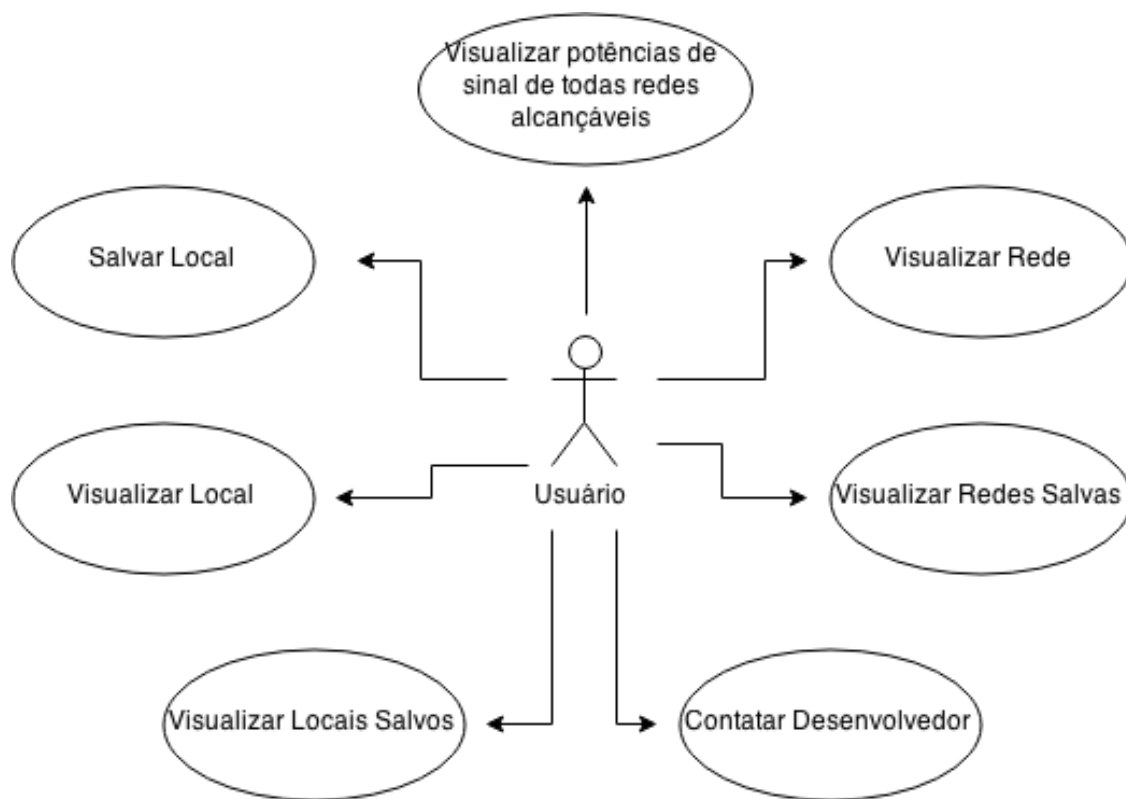


Figura 4.1 – Diagrama de Casos de Uso

#### 4.1.1 Ator

O único ator desse diagrama de casos de uso é o usuário da aplicação. Isso se deve ao fato de o usuário ter total controle das ações dentro do aplicativo, sem necessidade de qualquer outra intervenção externa. É o usuário quem inicia o aplicativo, lista e escolhe as opções do menu principal, interage com o aplicativo para visualizar informações, salva locais, e utiliza-se dos meios de comunicação com o desenvolvedor para sanar dúvidas e enviar críticas e/ou sugestões.

#### 4.1.2 Visualizar potências de sinal de todas redes alcançáveis

Também tratada como “Tela Inicial”, essa ação consiste da aquisição de informações referentes às redes móvel e Wi-Fi do usuário para mostrá-lo as informações adquiridas. Em outras palavras, o aplicativo busca o nome e a potência de sinal da rede móvel do usuário, bem como os nomes e as potências de sinal das redes Wi-Fi alcançáveis pelo dispositivo e disponibiliza-as para visualização na tela inicial.

#### 4.1.3 *Salvar Local*

Esta ação permite ao usuário salvar na base de dados do aplicativo todas as informações de rede contidas na tela inicial, no momento em que esta opção de menu foi acionada, associando estas redes apresentadas ao nome do local desejado pelo usuário.

#### 4.1.4 *Visualizar Locais Salvos*

Esta ação permite ao usuário visualizar todos os locais salvos por ele na base de dados do aplicativo, ordenados pela data de inclusão dos mesmos na base de dados, do mais antigo ao mais novo. Desta forma, o usuário pode explorar os locais salvos por ele no aplicativo, afim de comparar as redes salvas naquele local e inferir a localização do melhor sinal entre aqueles os quais têm interesse.

#### 4.1.5 *Visualizar Local*

Esta ação permite ao usuário visualizar todas as informações de rede disponíveis no local escolhido, dentre aqueles listados na opção de “Visualizar Locais Salvos” (brevemente descrita na seção 4.1.4). Estas informações de rede foram obtidas no momento em que o local foi salvo pelo usuário na base de dados do aplicativo. Assim, tais redes salvas são mostradas de maneira análoga ao que ocorre na tela principal do aplicativo, ordenadas por intensidade do sinal em dBm (exceto pelo sinal móvel, que sempre aparecerá no topo da listagem).

#### 4.1.6 *Visualizar Redes Salvas*

Esta ação permite ao usuário visualizar todas as redes salvas na base de dados do aplicativo, ordenadas pela data de inclusão das mesmas na base de dados, da mais antiga à mais nova. Desta forma, o usuário pode explorar as redes salvas no aplicativo, afim de comparar os locais nomeados por ele e inferir a localização da maior potência de sinal destas redes.

#### 4.1.7 *Visualizar Rede*

Esta ação permite ao usuário visualizar a lista detalhada de todos os locais salvos pelo usuário na base de dados do dispositivo em que a rede selecionada estava presente. Estes locais são listados de maneira análoga à tela inicial, mas neste caso, mostram na



coluna “Local” o nome do local, dado pelo usuário, onde essa amostra de rede foi obtida, e mostram na coluna “Potência”, a potência do sinal, em dBm, da amostra do sinal da rede selecionada, obtida naquele local.

#### 4.1.8 Contatar Desenvolvedor

Esta ação permite ao usuário visualizar informações relevantes sobre o desenvolvedor do aplicativo, além de dar a opção de contato para o mesmo. Na tela referente a esta ação, deve ser possível visualizar algumas informações do desenvolvedor, em formato de texto simples, além de opções interativas (na aplicação final representados por botões) que permitam o usuário entrar em contato com o desenvolvedor pelos meios disponibilizados pelo mesmo (neste caso, foi disponibilizado contato através da rede social *Facebook* ou através do e-mail da preferência do usuário).

## 4.2 Diagramas de Sequência

Um diagrama de sequência consiste de um conjunto de ações generalizadas que acabam por representar o diálogo entre elementos que se comunicam no decorrer do tempo da realização de uma determinada operação.

E para melhor ilustrar as ações e diálogos presentes no trabalho, foram elaborados diagramas de sequência para as principais ações a serem desenvolvidas posteriormente.

### 4.2.1 Tela Inicial (*Visualizar potências de sinal de todas redes alcançáveis*)

O Diagrama de Sequência que descreve a Tela Inicial do aplicativo está apresentado na Figura 4.2.

O fluxo começa quando o usuário inicia o aplicativo. Esta ação acarreta a criação de uma instância da classe *MainActivity*. Esta instância de classe, por sua vez, cria uma outra instância da classe *TelephonyManager*, com intuito de buscar sinal da rede de telefonia móvel. A instância da classe *TelephonyManager* faz todas as operações que lhe são pertinentes e ao final, retorna para a classe *MainActivity* o nome e a potência do sinal da rede de telefonia móvel conectada ao dispositivo. Assim, a classe *MainActivity* guarda estas informações para mostrar na tela inicial.

A mesma instância da classe *MainActivity* agora busca os sinais das redes Wi-Fi alcançáveis pelo dispositivo, e assim, cria uma instância da classe *WifiManager*. A instância da classe *WifiManager* faz todas as operações que lhe são pertinentes e ao final, retorna para a classe *MainActivity* uma lista contendo os nomes e as potências de todas as redes Wi-Fi alcançáveis pelo dispositivo, ordenadas por potência. Assim, a classe *MainActivity* junta estas informações com as anteriores retornadas de *TelephonyManager* e dispõe todas elas na tela para o usuário visualizar.

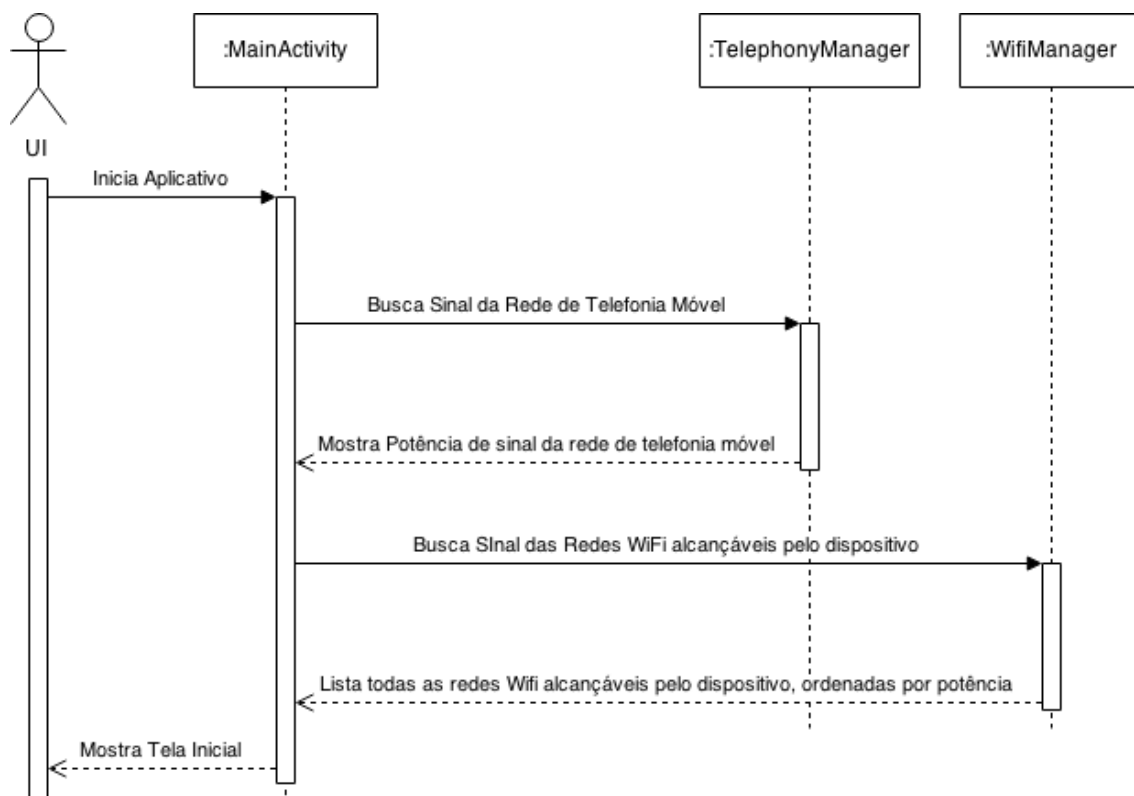


Figura 4.2 – Diagrama de Sequência da Tela Inicial do aplicativo

#### 4.2.2 Salvar Local

O Diagrama de Sequência da ação de Salvar Local está apresentado na Figura 4.3.

O processo começa quando o usuário inicia o aplicativo. Esta ação acarreta na criação de uma instância da classe *MainActivity*. Esta instância de classe executa tudo o que foi descrito na seção 4.2.1 e ao fim, mostra a Tela Inicial.

Na sequência, o usuário clica no botão menu, que aciona mais uma vez uma instância da classe *MainActivity*. Essa classe retorna então as opções de menu existentes para o usuário.

Após listadas as opções do menu, o usuário opta por clicar em Salvar Local. Esta ação acaba por criar uma instância da classe *SavePlace* que pede por um nome de local válido para o usuário. Em o usuário retornando um nome de local válido, a instância da classe *SavePlace* envia todos os dados atualmente mostrados pelo aplicativo, junto do nome do local à classe *DataHolder*, com o intuito de salvá-los. Assim, uma instância da classe *DataHolder* é criada. Esta instância de classe recebe os dados e salva-os na base de dados do aplicativo, retornando o estado da operação realizada. A instância da classe *SavePlace* então recebe o estado da operação no *DataHolder* e, caso todos os objetos tenham sido salvos com sucesso, ele então retorna o sucesso da operação para o usuário.

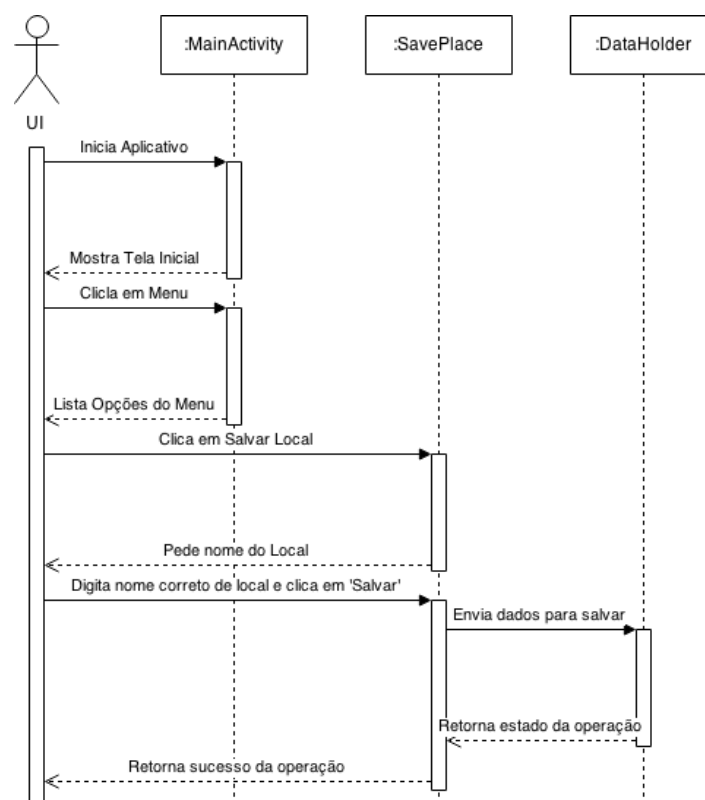


Figura 4.3 – Diagrama de Sequência da ação de Salvar Local

#### 4.2.3 Visualizar Locais Salvos / Visualizar (Detalhar) Local

O Diagrama de Sequência das ações de Visualizar Locais Salvos e Visualizar (Detalhar) Local são ilustrados na Figura 4.4.

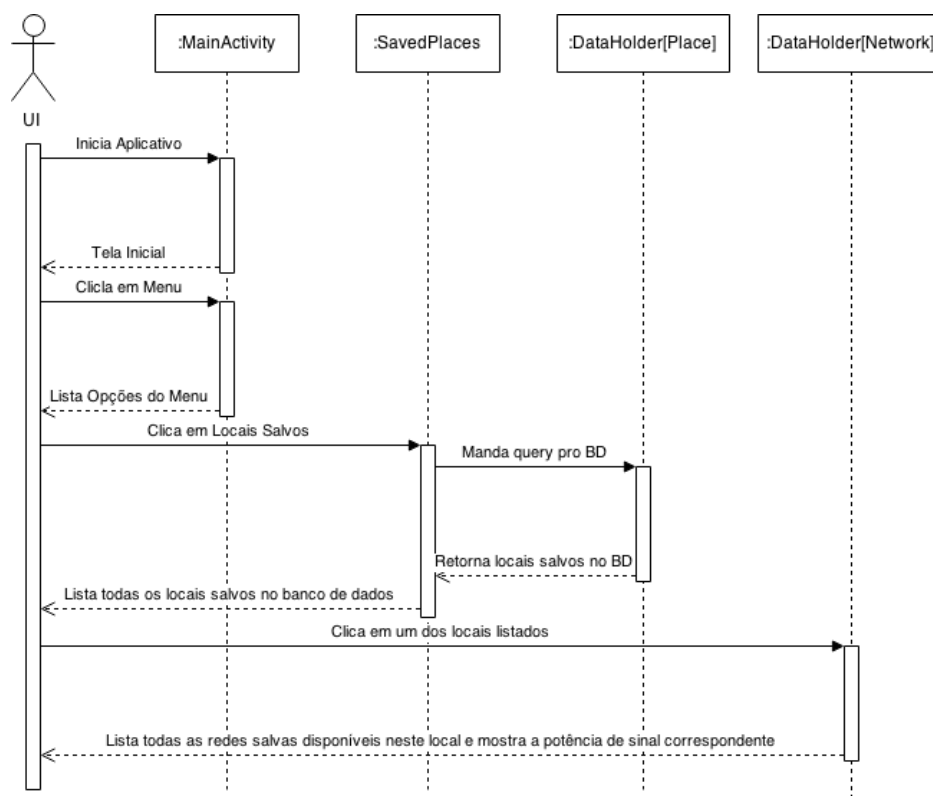


Figura 4.4 – Diagrama de Sequência das ações de Listar Locais Salvos e Visualizar Local

O processo começa quando o usuário inicia o aplicativo. Essa ação acarreta na criação de uma instância da classe *MainActivity*. Essa instância de classe executa tudo o que foi descrito na seção 4.2.1 e mostra a Tela Inicial.

Na sequência, o usuário clica no botão menu, que aciona mais uma vez uma instância da classe *MainActivity*. Essa classe retorna então as opções de menu existentes para o usuário.

Após listadas as opções do menu, o usuário opta por clicar em Locais Salvos. Esta ação acaba por criar uma instância da classe *SavedPlaces*. Esta classe por sua vez, cria uma instância da classe *DataHolder* procurando por todos os locais salvos na base de dados do aplicativo. A instância da classe *DataHolder* executa a busca e retorna todos os locais salvos na base de dados para a classe *SavedPlaces*, que, por sua vez, retorna a lista de locais salvos para mostrar ao usuário.

Após listados todos os locais salvos na base de dados do dispositivo, o usuário opta por clicar em um dos locais salvos. Esta ação acaba criando uma instância da classe

*DataHolder* para aquele local. Essa instância de classe, por sua vez, retorna todas as redes salvas para aquele local na base de dados e dispõe essas informações ao usuário de forma análoga à tela inicial.

#### 4.2.4 Visualizar Redes Salvas / Visualizar (Detalhar) Rede

O Diagrama de Sequência das ações de Visualizar Redes Salvas e Visualizar (Detalhar) Rede são ilustrados na Figura 4.5.

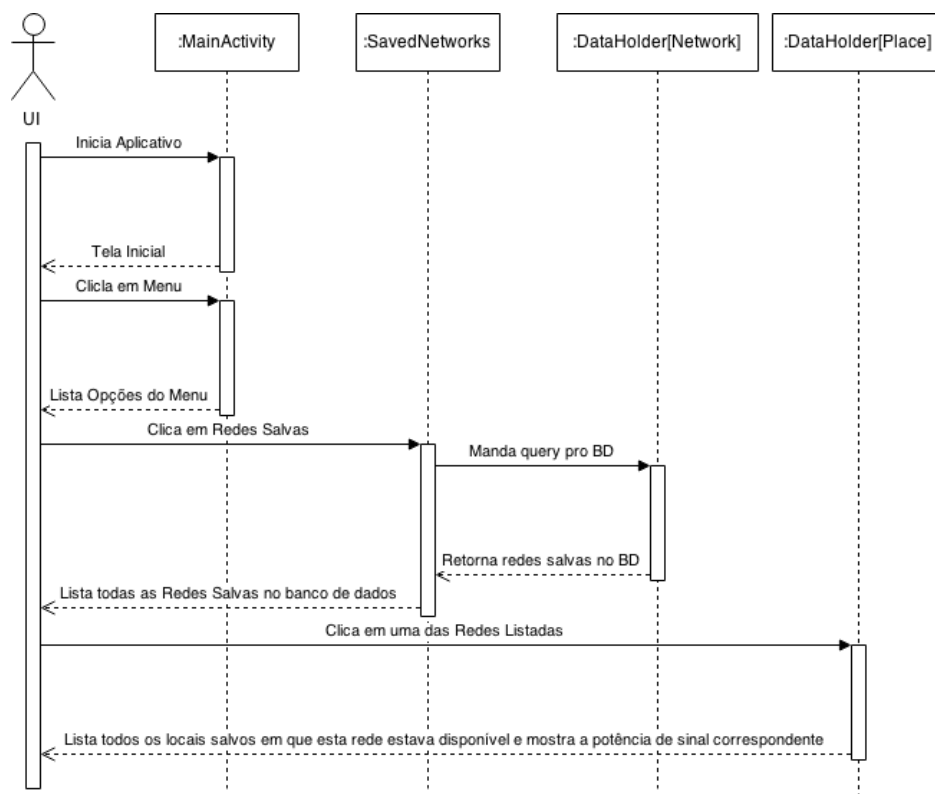


Figura 4.5 – Diagrama de Sequência das ações de Listar Redes Salvas e Visualizar Rede

O processo começa quando o usuário inicia o aplicativo. Essa ação acarreta na criação de uma instância da classe *MainActivity*. Essa instância de classe executa tudo o que foi descrito na seção 4.2.1 e mostra a Tela Inicial.

Na sequência, o usuário clica no botão menu, que aciona mais uma vez uma instância da classe *MainActivity*. Essa classe retorna então as opções de menu existentes para o usuário.

Após listadas as opções do menu, o usuário opta por clicar em Redes Salvas. Esta ação acaba por criar uma instância da classe *SavedNetworks*. Essa classe, por sua

vez, cria uma instância da classe *DataHolder* procurando por todas as redes salvas na base de dados do aplicativo. A instância da classe *DataHolder* executa a busca e retorna todas as redes salvas na base de dados para a classe *SavedNetworks*, que, por sua vez, retorna a lista de redes salvas para mostrar ao usuário.

Após listadas todas as redes salvas na base de dados do dispositivo, o usuário opta por clicar em uma das redes salvas. Essa ação acaba criando uma instância da classe *DataHolder* para aquela rede. Essa instância de classe, por sua vez, retorna todos os locais salvos para aquela rede na base de dados e dispõe essas informações ao usuário de forma análoga à tela inicial.

### 4.3 Diagrama de Classe

O diagrama de classe referente a base de dados constituída neste trabalho (ver sobre detalhes da implementação da base de dados na seção 5.6) está ilustrado na Figura 4.6. Neste diagrama, são expostas 3 classes: a classe *Place*, a classe *Network* e a classe *Signal*.

A classe *Place* guarda as informações referentes ao local salvo pelo usuário no aplicativo. Por conta disso, essa classe possui o parâmetro *name*, referente ao nome do local dado pelo usuário. O usuário pode salvar 0 ou mais lugares no aplicativo, então, essa classe pode ocorrer n vezes na aplicação.

A classe *Network* guarda as informações referentes à rede salva pelo usuário no aplicativo. Por conta disso, essa classe possui o parâmetro *name*, referente ao nome da rede como está registrada na antena de telefonia móvel ou no ponto de acesso Wi-Fi. O usuário pode salvar 0 ou mais redes no aplicativo, então, essa classe pode ocorrer n vezes na aplicação.

A classe *Signal* é uma classe associativa que relaciona as classes *Place* e *Network*. Como visto anteriormente, cada local pode ter 0 ou mais redes analisadas, assim como cada rede pode estar presente em 0 ou mais locais distintos. Dessa forma, constitui-se uma relação n para n, onde há a necessidade de uma classe associativa para relacioná-las. Essa classe guarda as informações de uma determinada rede em determinado local. Por conta disso, essa classe possui o parâmetro *strength*, referente à potência de sinal registrada por determinada rede no local onde o usuário deseja salvar as informações.

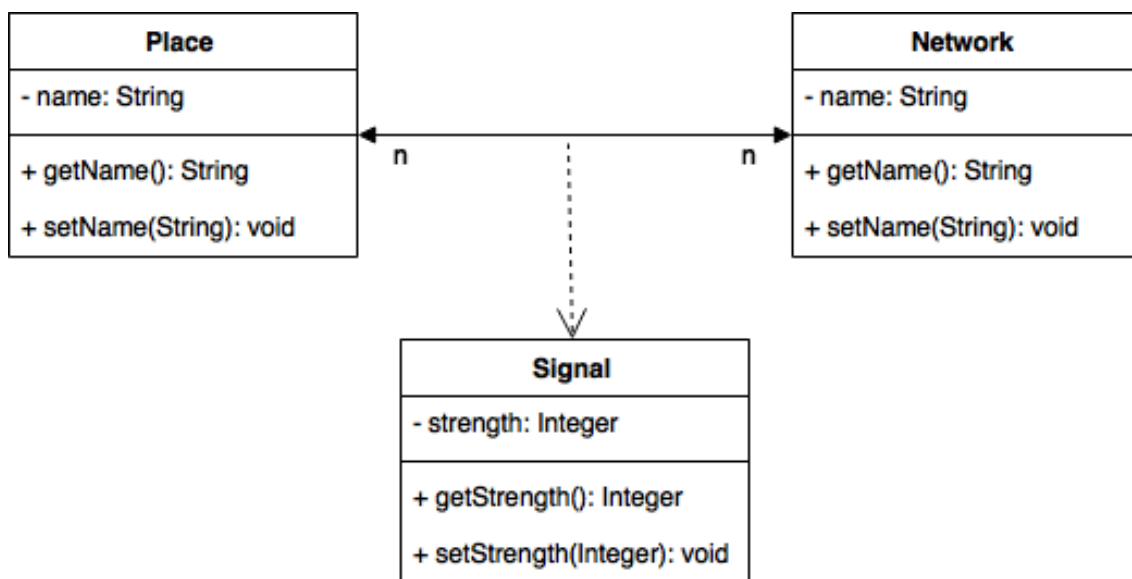


Figura 4.6 – Diagrama de Classes referente à base de dados da aplicação

## 5 IMPLEMENTAÇÃO

Neste capítulo será descrito, de maneira mais técnica, o processo de desenvolvimento da ferramenta. Será exposto o ambiente de desenvolvimento utilizado, os pré-requisitos necessários para o início do desenvolvimento da ferramenta, a descrição da API utilizada como base da aplicação, um aprofundamento em relação às classes que compõem a análise de estado da rede, bem como os parâmetros utilizados na listagem de redes da tela inicial do aplicativo, e a apresentação de algumas peculiaridades interessantes sobre o código desenvolvido.

### 5.1 Ambiente de Desenvolvimento

O ambiente de desenvolvimento utilizado e todo o ferramental necessário para a composição da parte prática deste trabalho está disponível no site do Android focado para desenvolvedores (<http://developer.android.com/sdk/index.html>). Nesta página, está disponibilizado para *download* um pacote, que inclui o SDK (*Software Development Kit*, em português, Kit de Desenvolvimento de Software) Android mais atualizado e o Android Studio, IDE (*Integrated Development Environment*, em português, Ambiente de Desenvolvimento Integrado) oficial para desenvolvimento Android.

O SDK disponibilizado no pacote descrito, traz boa parte dos subsídios necessários para iniciar-se no desenvolvimento Android. Ainda no SDK, há inúmeras ferramentas executáveis e que são de fundamental importância para definir o ambiente de desenvolvimento Android: o SDK Manager e o AVD (*Android Virtual Device*, em português, Dispositivo Virtual Android) Manager.

O SDK Manager é um gerenciador de todos os pacotes disponíveis para o desenvolvimento Android. Nele estão presentes todas as APIs para serem baixadas, ferramentas extras de desenvolvimento e até drivers de dispositivos móveis que possuem algum problema de sincronização para o desenvolvimento Android.

O AVD Manager, por sua vez, é um gerenciador de Dispositivos Virtuais Android que podem ser criados para que se possa testar os aplicativos desenvolvidos sem necessidade de um *smartphone* Android conectado. Neste gerenciados, pode-se criar dispositivos livremente, podendo configurar elementos como o tamanho da tela, a versão do Android e a memória RAM a ser disponibilizada, por exemplo.



Já o Android Studio é uma IDE construída baseada no *IntelliJ IDEA Community Edition*, uma das mais populares IDEs de Java, desenvolvida pela JetBrains (ANDROID). Trata-se então de uma IDE muito moderna, com editor de código inteligente e com inúmeros recursos que facilitam o desenvolvimento da aplicação. Alguns desses recursos são *templates* de códigos, integração com o GitHub, construção de *layouts* mais organizados, bonitos e modernos, construção do mesmo arquivo de *layout* para diferentes dispositivos, tratando as peculiaridades de cada um deles.

Para reproduzir resultados mais fidedignos, há a necessidade de utilizar-se um dispositivo móvel físico, ao invés de um dispositivo virtual presente no ADB. Para este trabalho, foi utilizado um *smartphone* Sony Xperia SP durante todo o desenvolvimento da ferramenta. Todas as imagens expostas no capítulo 6 são capturas de tela deste dispositivo móvel executando a ferramenta desenvolvida.

## 5.2 API Android escolhida

Como citado na seção 2.2.1, uma API Android é um conjunto de ferramentas e documentos que permitem o desenvolvimento de um aplicativo Android. Por conta do grande número de APIs disponível, a escolha da API a ser utilizada é uma importante definição a ser feita antes do começo do processo de desenvolvimento da aplicação.

No aplicativo desenvolvido neste trabalho, o nível de API utilizado foi o 8, que corresponde a versão Android 2.2, conhecida também como *Froyo*. Esta API foi escolhida pois cobre pouco mais de 99% dos *smartphones* utilizados no mundo (ANDROID, 2015). Por ser um aplicativo que usa poucos recursos mais modernos, a escolha foi pela máxima compatibilidade possível com o usuário final.

Mesmo assim, foram utilizados, no aplicativo desenvolvido, alguns recursos de APIs mais recentes. Para tanto, no começo de cada código que utiliza um recurso mais recente, é realizado um teste para saber se o dispositivo que está utilizando o aplicativo é compatível com o recurso ou não. Se é compatível, utiliza o recurso citado, caso contrário, utiliza um recurso semelhante, mas menos moderno. Além disso, deve-se usar uma marcação de “Nova API” no começo do método que utiliza o recurso em questão.

Como ilustrado na Figura 5.1, para utilizar-se um recurso recente, introduzido na API 11, de nome *Honeycomb*, também conhecida por Android 3.0, faz-se o teste se a

versão Android do dispositivo onde o aplicativo está rodando é maior ou igual a versão onde o recurso está disponível. Além disso, há a sinalização de utilização de “Nova API” no começo do método onde o recurso é utilizado.

```
@SuppressWarnings("NewApi")
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_place);

    if (android.os.Build.VERSION.SDK_INT >= android.os.Build.VERSION_CODES.HONEYCOMB) {
        getActionBar().setDisplayHomeAsUpEnabled(true);
    }
}
```

Figura 5.1 – Trecho de código utilizando código de API mais recente

### 5.3 Wi-Fi Manager

Como visto na seção 2.2.4, o Wi-Fi Manager é uma classe que constitui uma *thread* que colhe informações quanto ao estado de todas as redes Wi-Fi alcançáveis pelo dispositivo móvel. Quando há uma mudança em algumas dessas redes, um evento é chamado, contendo os dados coletados no último estado válido da classe.

Para coletar os dados referentes ao trabalho desenvolvido, usou-se o método *getScanResults()* desta classe. Este método tem por finalidade retornar uma lista dos últimos pontos de acesso examinados pela *thread* citada no parágrafo anterior. O retorno do método é uma lista de elementos da classe *ScanResult*.

A classe *ScanResult* foi projetada para descrever a informação de um ponto de acesso Wi-Fi detectado. Uma instância dessa classe possui como parâmetros alguns campos referentes a uma rede Wi-Fi. Destacamos os principais:

- *BSSID*: É um conjunto de caracteres que representa o endereço MAC do ponto de acesso Wi-Fi.
- *SSID*: É o conjunto de caracteres que representa o nome do ponto de acesso Wi-Fi.
- *Frequency*: É o inteiro que apresenta a frequência, em MHz, do canal ao qual o cliente está se comunicando com o ponto de acesso Wi-Fi.

- *Level*: É o inteiro que define o nível de sinal detectado, em dBm, do ponto de acesso Wi-Fi.

Para a listagem dos sinais das redes Wi-Fi destacáveis, usa-se o parâmetro SSID na coluna de nome de rede e o parâmetro *level* na coluna de potência de sinal, em dBm.

#### 5.4 Telephony Manager

Como visto na seção 2.2.5, o *Telephony Manager* é uma classe que constitui uma *thread* que colhe informações quanto ao estado da rede de telefonia móvel ao qual o dispositivo móvel está conectado. Quando há uma mudança no estado da rede de telefonia móvel, um evento é chamado, contendo os dados coletados no último estado válido da classe.

Para coletar informações úteis para o trabalho, primeiro adquirimos a referência do nome da rede móvel ao qual o dispositivo móvel está conectado. Para tanto, usamos a função *getNetworkOperatorName()*, que tem por finalidade retornar o nome alfabético do operador registrado na antena ao qual o dispositivo está conectado. Este nome fornecido pode variar de antena para antena. Em alguns testes, por exemplo, pode-se perceber redes com nome “Vivo”, outras com nome “Vivo RS 51” e ainda outras nomeadas como “VIVO”.

Já para coletar a informação da potência do sinal de telefonia móvel ao qual o dispositivo está conectado, é preciso registrar um *listener* para receber notificações de mudanças na potência do sinal da rede conectada. Esse *listener* é acionado a cada vez que uma mudança na rede de telefonia é detectada, como explicado no primeiro parágrafo desta seção. Assim, o *listener* recebe um evento contendo uma instância da classe *SignalStrength*.

A classe *SignalStrength* contém todos os parâmetros da potência do sinal da rede de telefonia móvel. Para coletar a potência de sinal informada pela instância da classe, basta chamar a função *getGsmSignalStrength()*. Esta função retorna o inteiro do ASU usado no cálculo demonstrado na seção 2.1.5.1.

Desta forma, estamos munidos das informações do nome da rede móvel ao qual o dispositivo está conectado e da potência do sinal da mesma, que é o necessário para a aplicação resultante desta monografia.

## 5.5 Intent

No aplicativo desenvolvido neste trabalho, a classe *Intent* foi utilizada para iniciar novas Atividades dentro de uma Atividade, enviando para a classe correspondente os parâmetros necessários para a execução perfeita da ação da Atividade seguinte.

A classe *Intent* também foi utilizada para executar ações em outros aplicativos. Quando o usuário do aplicativo feito neste trabalho precisa enviar um e-mail ao desenvolvedor ou visitar seu perfil na rede social *Facebook*, a ferramenta invoca, via *Intent*, os aplicativos mais indicados para a execução dessas tarefas.

No caso da *intent* que referencia o aplicativo *Facebook*, o sistema envia uma *intent* para o aplicativo do *Facebook*, contendo as informações do perfil a ser visitado na rede social. Caso o sistema não consiga enviar esta *intent* (usualmente quando o aplicativo do *Facebook* instalado no dispositivo do usuário não atende às especificações necessárias para a realização da operação, ou seja, o aplicativo está deveras desatualizado, ou quando não há aplicativo do *Facebook* instalado no dispositivo tratado), o sistema então envia uma *intent* para o navegador padrão do dispositivo do usuário contendo o link completo para o acesso ao perfil requerido.

Já no caso da *intent* de e-mail, o sistema envia uma *intent* do tipo e-mail para o sistema *Android* com a informação do destinatário. Assim, como pode ser visto no exemplo da Figura 6.10, o sistema *Android* mostra ao usuário todos os aplicativos instalados em seu dispositivo que podem tratar o tipo e-mail, e oportuniza o usuário a escolher aquele que mais lhe agrada para completar a ação.

## 5.6 Base de dados

Para manter a leveza e, conseqüentemente, a rapidez do aplicativo, usou-se um arquivo interno para guardar as informações salvas pelo aplicativo. O fato de o aplicativo salvar pouquíssimos dados em disco também facilitou a escolha.

Quando o usuário pretende salvar determinado estado do aplicativo, todas as informações referentes às classes de Lugar e Rede, e que devem ser guardadas em disco, são gravadas em um arquivo interno ao aplicativo, o qual apenas o aplicativo pode manipular.

## 6 FUNCIONALIDADES DESENVOLVIDAS

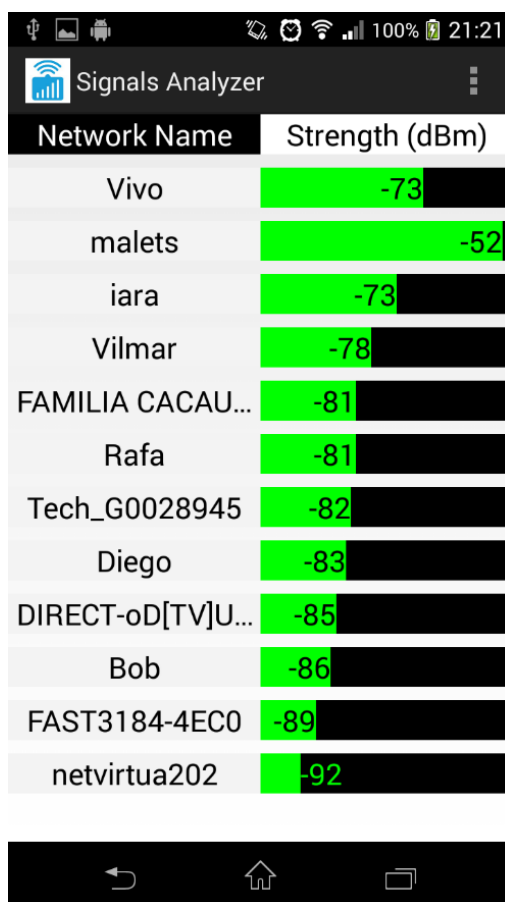
Baseado no projeto exposto no capítulo 4 e no desenvolvimento descrito no capítulo 5, a implementação resultante deste trabalho consiste de um aplicativo para o Sistema Operacional *Android*, de nome *Signals Analyzer*, que apresenta um método para apuração do local onde a potência de sinal de um dispositivo móvel seja a maior possível. Para tanto, a ferramenta desenvolvida será textualmente definida e minuciosamente esmiuçada, listando suas funções e opções existentes, descritas nos parágrafos que seguem.

O aplicativo desenvolvido pode ser instalado diretamente da Google Play Store no link: <https://play.google.com/store/apps/details?id=com.fabiomalet.mobsigalyzer>.

### 6.1 Tela Inicial

Como pode ser observado na Figura 6.1, a tela inicial do aplicativo informa ao usuário a potência, em dBm, dos sinais alcançáveis pelo dispositivo no lugar onde ele encontra-se no momento. Essas informações são referentes tanto ao sinal de telefonia móvel ao qual o dispositivo móvel está associado, quanto aos sinais de pontos de acesso Wi-Fi alcançáveis pelo aparelho. O sinal de telefonia móvel aparece sempre no topo da exibição dessa tela, com os sinais de pontos de acesso Wi-Fi aparecendo em seguida, ordenados pela potência do mesmo. Tanto o sinal de telefonia móvel, quanto os sinais de pontos de acesso Wi-Fi são rotulados conforme o nome de acesso que lhe são dados – “Vivo”, “malets”, “Tech\_G0028945” e “netvirtua202” são alguns exemplos.

Para o melhor entendimento dos dados informados, a aplicação dispõe de gráficos que auxiliam na visualização desses elementos, de forma a conseguir relacionar o número bruto da potência do sinal envolvido, criando intervalos de classificação que melhor amparem o usuário. Munidos dessa representação visual, e dos intervalos supracitados, o usuário da ferramenta pode qualificar o sinal analisado, inferindo sobre o quão bom, ou ruim, ele se apresenta e qual a proximidade deste sinal ao valor máximo e mínimo que ele poderia alcançar.



The screenshot shows the 'Signals Analyzer' app interface. At the top, there is a status bar with various icons and the time '21:21'. Below the status bar, the app title 'Signals Analyzer' is displayed. The main content is a table with two columns: 'Network Name' and 'Strength (dBm)'. The table lists several networks with their corresponding signal strengths, each represented by a green bar and a numerical value. At the bottom, there is a navigation bar with three icons: a back arrow, a home icon, and a folder icon.

Network Name	Strength (dBm)
Vivo	-73
malets	-52
iara	-73
Vilmar	-78
FAMILIA CACAU...	-81
Rafa	-81
Tech_G0028945	-82
Diego	-83
DIRECT-oD[TV]U...	-85
Bob	-86
FAST3184-4EC0	-89
netvirtua202	-92

Figura 6.1 – Tela inicial do aplicativo

Por conta de as informações referentes à potência de sinal oscilarem continuamente, a tela inicial é atualizada a cada mudança em um destes dados, de forma a mostrar ao usuário sempre o valor mais recente da potência de sinal no local onde o mesmo se encontra. Apoiado nas bibliotecas da API de *Android*, que tratam dos sinais de rede de dados do aparelho, o aplicativo possui *threads* executando em segundo plano para continuamente verificar as informações de potência dos sinais alcançáveis pelo dispositivo. Quando há mudança em alguma dessas informações, a *thread* responsável manda uma mensagem para a função principal da aplicação, que por sua vez, encaminha a informação para que a mesma seja atualizada.

## 6.2 Menu Principal

O menu principal do sistema desenvolvido consiste de um botão na barra de título do aplicativo que abre as opções inclusas no sistema, e que serão descritas nas seções que seguem.

A Figura 6.2 apresenta a tela inicial do aplicativo, como foi descrita na seção anterior, com o menu principal aberto, mostrando as opções que o aplicativo abrange. Cada opção do menu principal se relaciona a uma *Activity* (ver seção 2.2.2) correspondente, ou seja, quando uma opção listada é escolhida, sua respectiva *Activity* é iniciada.

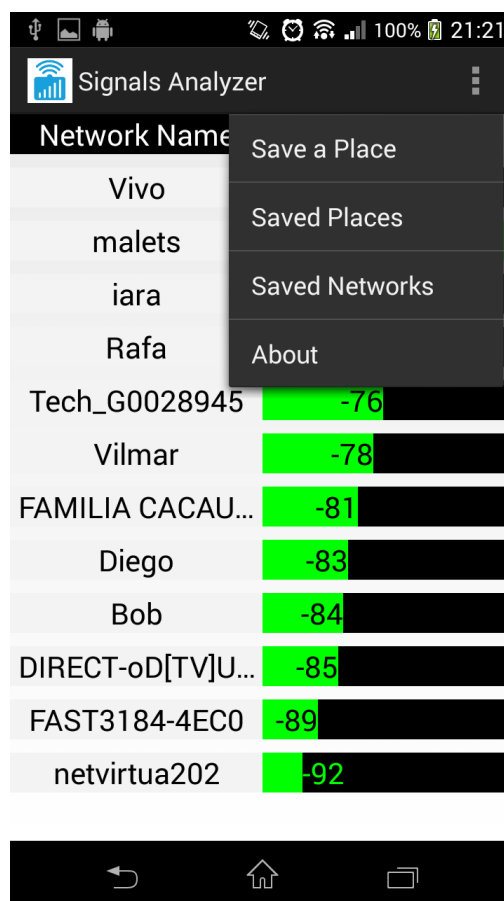


Figura 6.2 – Menu principal exibido na tela inicial

## 6.3 Salvar Local

A opção de “Salvar Local”, presente no menu principal da aplicação, adiciona a funcionalidade de poder salvar as informações contidas na tela inicial, rotulando-as com um texto a ser definido livremente pelo usuário.

Acessando o menu principal do aplicativo, o usuário verá listados alguns submenus. Dentre eles, há o de nome “Salvar Local”. Ao clicar neste submenu, aparecerá na tela do dispositivo uma caixa de texto a ser preenchida, indicando ao usuário que ele deve nomear o local a ser salvo, informando caracteres permitidos neste preenchimento. Ainda nesta caixa de preenchimento, haverá dois botões que definem a ação ocorrida na sequência: “Cancelar” e “Salvar”. Ao clicar “Cancelar”, o aplicativo volta à sua tela inicial. No caso de o usuário clicar no botão “Salvar”, o aplicativo verifica se foi digitada uma frase válida no campo de preenchimento. Caso o que foi preenchido seja legítimo, o aplicativo salva as informações atuais, rotulando-as com o nome fornecido. Do contrário, a ferramenta cancela o que foi feito e volta à tela inicial.

A Figura 6.3 exibida na sequência ajuda a elucidar a constituição da função descrita.

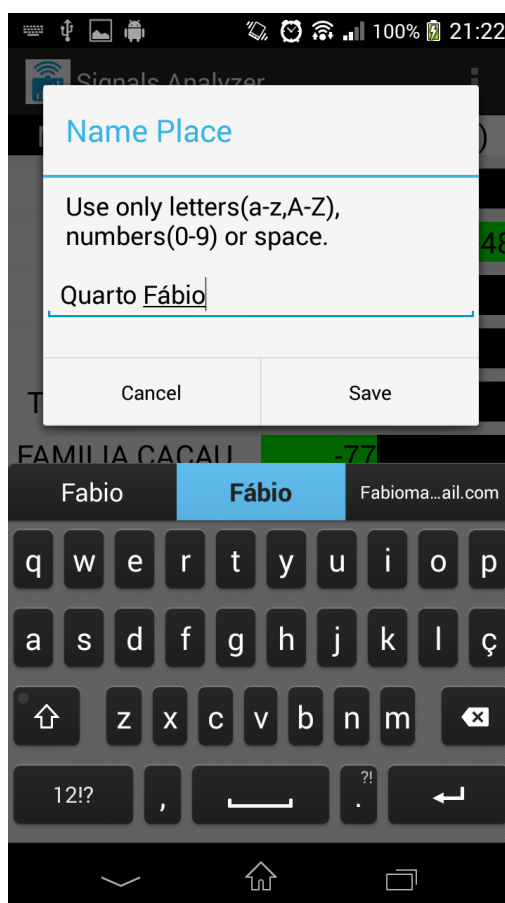


Figura 6.3 – Tela que exibe opção de “Salvar Local”

Para ficar mais claro, suponhamos que o usuário esteja usando o aplicativo em seu quarto. Dessa forma, ele pode salvar todas as informações contidas na tela de seu



aplicativo, rotulando-as com o nome “Quarto”. A efeito de comparação, o usuário dirige-se à sala de sua casa e faz o mesmo procedimento, mas desta vez rotulando a informação salva como “Sala”. Após isto, e como veremos posteriormente, o usuário poderá informar-se da intensidade dos sinais alcançáveis pelo dispositivo na sala de sua casa, em seu quarto e também compará-las, de modo a determinar o local onde há o melhor sinal dentre àqueles que investiga.

#### 6.4 Locais Salvos

Como pode ser visto na Figura 6.4, a opção de “Locais Salvos”, presente no menu principal da aplicação, propicia ao usuário, em nova tela, a listagem de todos os locais salvos pelo mesmo na aplicação, organizados pela ordem em que os mesmos foram armazenados. Dessa forma, os nomes dos locais guardados no arquivo de controle de dados da aplicação são dispostos em formato de lista de botões para que, quando clicados, detalhem o local selecionado.

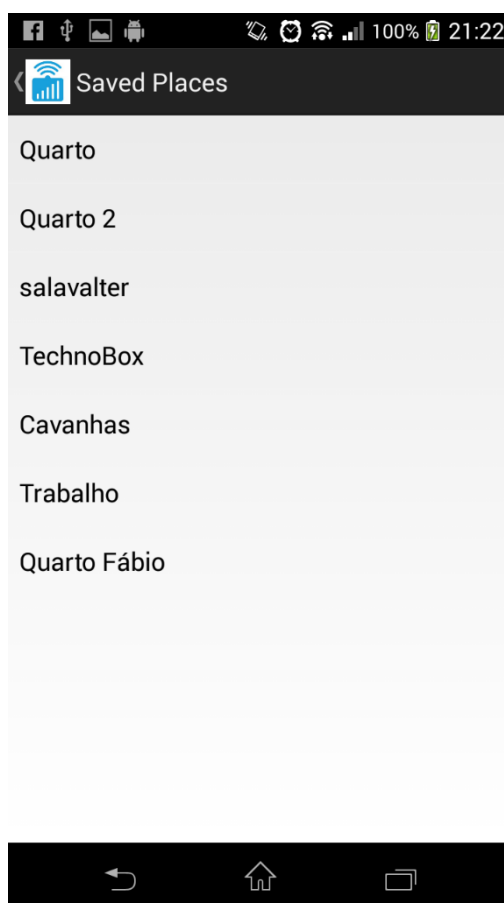
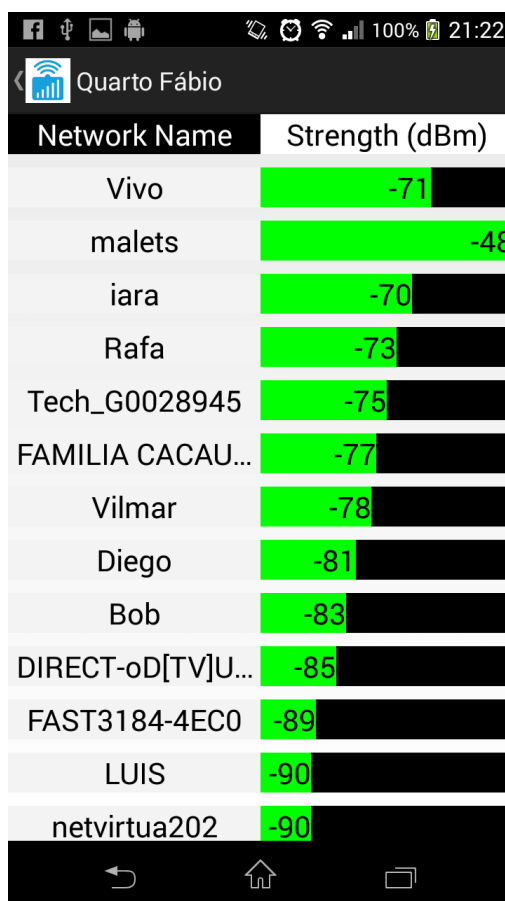


Figura 6.4 – Tela de listagem de Locais Salvos

Pegando como exemplo o cenário explicitado na seção 6.3, a opção de “Locais Salvos” levaria a uma nova tela, onde estariam listados dois lugares: “Quarto” e “Sala”. Ao clicarmos nestes locais listados, acessaríamos uma nova tela, análoga à tela inicial, mostrando as potências, em dBm, dos sinais salvos pela aplicação naquele local.

Na Figura 6.5 vemos o exemplo do detalhamento do local chamado “Quarto Fábio”. Nela observam-se todos os sinais salvos pela aplicação no local nomeado pelo usuário. A primeira linha desta listagem indica o sinal de telefonia móvel do dispositivo, que neste caso é representada pela operadora “Vivo”, nome presente na informação da antena a qual o dispositivo está conectado. Logo abaixo estão presentes todos os sinais de rede Wi-Fi alcançáveis pelo dispositivo no local, ordenados pela intensidade do mesmo.



Network Name	Strength (dBm)
Vivo	-71
malets	-48
iara	-70
Rafa	-73
Tech_G0028945	-75
FAMILIA CACAU...	-77
Vilmar	-78
Diego	-81
Bob	-83
DIRECT-oD[TV]U...	-85
FAST3184-4ECO	-89
LUIS	-90
netvirtua202	-90

Figura 6.5 – Tela de detalhamento de um local salvo (“Quarto Fábio”)

## 6.5 Redes Salvas

A opção de “Redes Salvas”, presente no menu principal da aplicação, propicia ao usuário, em nova tela, a listagem de todas as redes salvas pelo usuário na aplicação. Quando do salvamento de um local, o usuário também acaba por guardar as informações de um conjunto de redes pertencentes àquele local. Assim, as redes salvas são listadas na tela do submenu de “Redes Salvas”, rotuladas pelos seus respectivos nomes e em formato de lista de botões, análogo à tela de “Locais Salvos”, para que, quando clicados, detalhem mais a rede selecionada.

A listagem, brevemente descrita no parágrafo anterior e exemplificada na Figura 6.6, distingue a rede móvel para ser mostrada no topo da tela. Com isso, as redes Wi-Fi salvas no aplicativo são elencadas abaixo, organizadas pela ordem de inclusão da mesma na base de dados da aplicação.

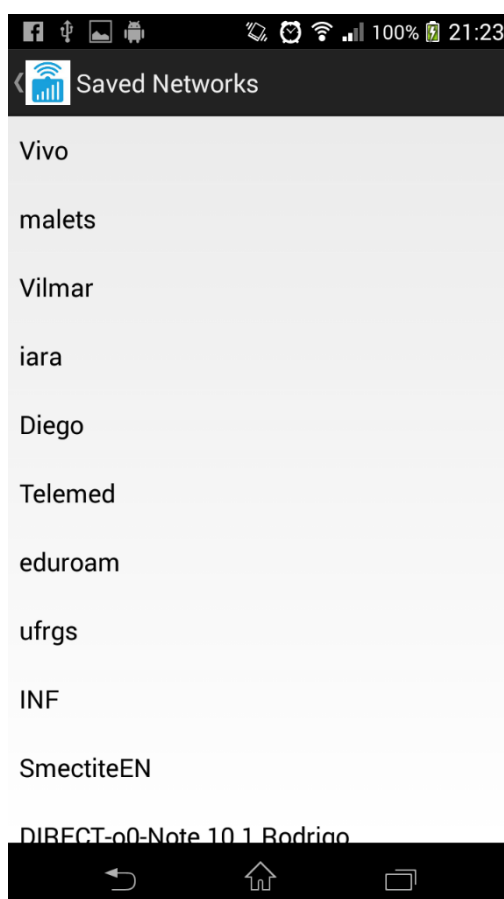
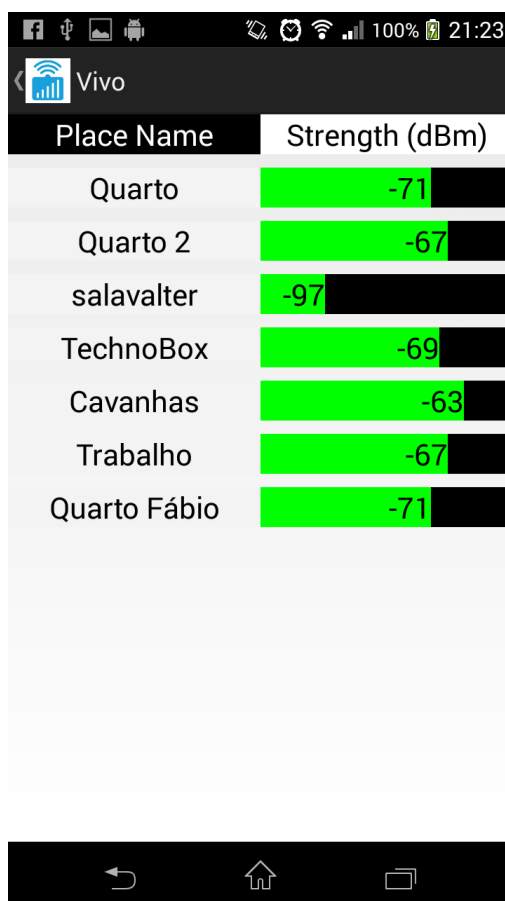


Figura 6.6 – Tela de listagem de Redes Salvas

Pegando novamente como exemplo o cenário descrito no item 6.3, a opção de “Redes Salvas” levaria a uma nova tela, onde estariam listadas todas as redes alcançáveis nos lugares salvos (“Quarto” e “Sala”). Suponhamos que ambos locais possuem listados a rede de telefonia móvel “Operadora RS 51” e o ponto de acesso Wi-Fi “AccessPoint”. Desta forma, a tela de “Redes Salvas” listaria todas as redes salvas, incluindo as redes “Operadora RS 51” e “AccessPoint”.

Ao clicarmos em uma destas redes salvas, acessaríamos uma nova tela, análoga à tela inicial, mostrando as potências, em dBm, dos sinais salvos pela aplicação naquele local, referente à rede correspondente. Ou seja, clicando-se em “Operadora RS 51” no menu de “Redes Salvas”, visualizar-se-ia uma lista com dois elementos: “Quarto” e “Sala”, mostrando as respectivas potências de sinal de “Operadora RS 51” naquele local. Desta forma, o usuário tem a possibilidade de informar-se sobre o local onde há o melhor sinal daquela rede investigada.

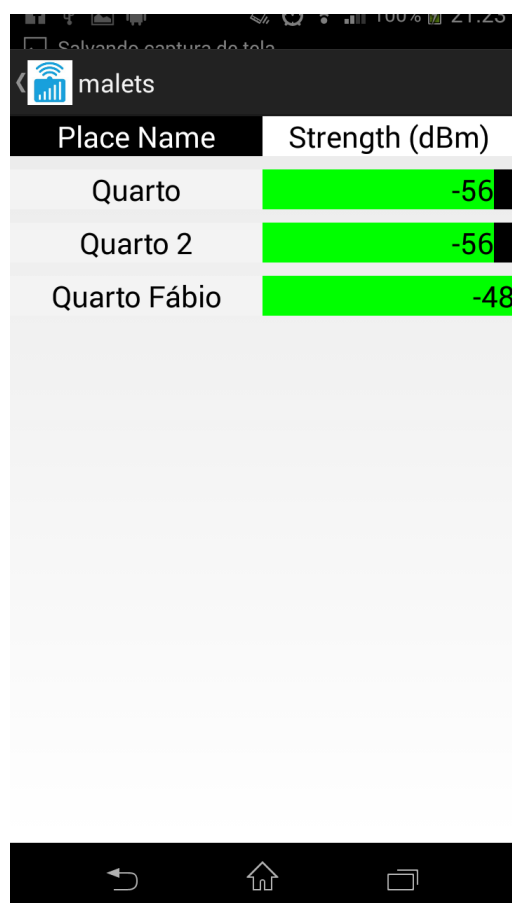


Place Name	Strength (dBm)
Quarto	-71
Quarto 2	-67
salavalter	-97
TechnoBox	-69
Cavanhas	-63
Trabalho	-67
Quarto Fábio	-71

Figura 6.7 – Tela de detalhamento de uma rede salva (Rede Móvel “Vivo”)

Nas figuras Figura 6.7 e Figura 6.8 pode-se visualizar o detalhamento de duas redes distintas. Na Figura 6.7, da rede móvel “Vivo”, observam-se todos os locais salvos pela aplicação em que essa rede estava alcançável pelo dispositivo, organizados pela ordem de inclusão do local na base de dados.

Supondo-se que os locais “Quarto” e “Quarto 2” sejam cômodos da mesma residência, na Figura 6.7, temos duas amostras distintas para a rede móvel “Vivo”. No local “Quarto” essa rede tem intensidade de sinal de valor -71 dBm. Já no local “Quarto 2” a intensidade de sinal dessa rede móvel melhora um pouco e chega a -67 dBm. Desta forma, o usuário morador dessa residência pode inferir que o melhor local para fazer uso da rede móvel “Vivo”, como por exemplo realizar uma chamada de voz, ou fazer uso do serviço de dados 4G, é no local denominado “Quarto 2”.



Place Name	Strength (dBm)
Quarto	-56
Quarto 2	-56
Quarto Fábio	-48

Figura 6.8 - Tela de detalhamento de uma rede salva (Rede Wi-Fi “malets”)

A Figura 6.8 mostra a tela de detalhamento da rede Wi-Fi “malets”. De maneira análoga à rede móvel “Vivo” (Figura 6.7), nessa imagem observam-se todos os locais

salvos pela aplicação em que a rede Wi-Fi “malets” estava alcançável pelo dispositivo, organizados pela ordem de inclusão do local na base de dados.

Supondo-se agora que os locais “Quarto” e “Quarto Fábio” sejam cômodos da mesma residência, na Figura 6.8 temos duas amostras distintas para a rede Wi-Fi “malets”. No local “Quarto” esta rede tem intensidade de sinal de valor -56 dBm. Já no local “Quarto Fábio” a intensidade de sinal dessa rede Wi-Fi melhora consideravelmente e chega a -48 dBm. Dessa forma, o usuário morador dessa residência pode inferir que o melhor local para fazer uso da rede Wi-Fi “malets”, como por exemplo uso de aplicações multimídia de uso massivo de dados da rede ou até mesmo a instalação de um repetidor de sinal Wi-Fi, é no local denominado “Quarto Fábio”.

## 6.6 Sobre o Aplicativo

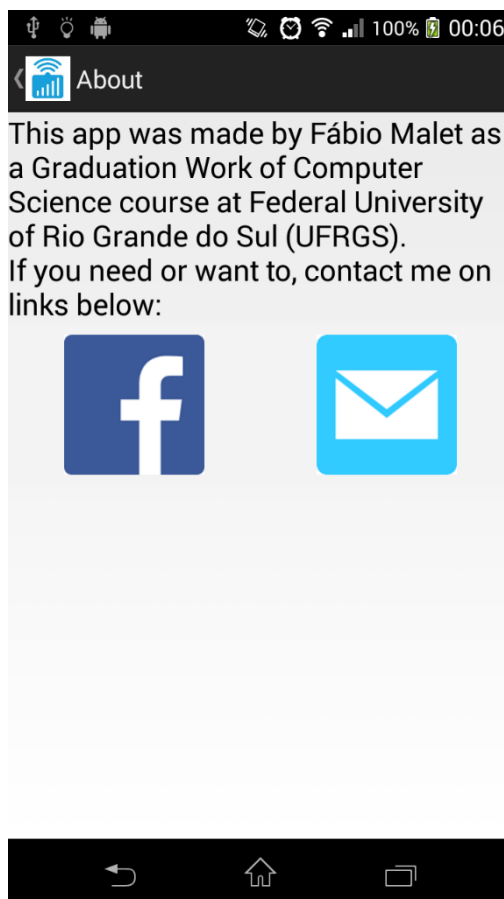


Figura 6.9 – Tela de informações do aplicativo

A opção “Sobre”, presente no menu principal da aplicação, propicia ao usuário, em nova tela, a visualização das informações referentes ao desenvolvedor da aplicação, bem como opções de contato com o mesmo.

Como pode ser visualizado na Figura 6.9, a tela da opção “Sobre” mostra informações sobre o desenvolvimento da aplicação, como seu autor e o objetivo do mesmo. Além disso, a tela traz a opção de contato com o desenvolvedor por meio dos dois botões visualizados na imagem.

O primeiro botão, que mostra o logotipo da rede social *Facebook*, redireciona o usuário para o perfil do desenvolvedor da aplicação na rede social. Para isso, quando o botão é clicado, o sistema envia uma *intent* para o aplicativo do *Facebook*, contendo as informações do perfil do desenvolvedor na rede social. Caso o sistema não consiga enviar esta *intent* (usualmente quando o aplicativo do *Facebook* instalado no dispositivo do usuário não atende às especificações necessárias para a realização da operação, ou seja, o aplicativo está deveras desatualizado, ou quando não há aplicativo do *Facebook* instalado no dispositivo tratado), o sistema então envia uma *intent* para o navegador padrão do dispositivo do usuário contendo o link completo para o acesso ao perfil do desenvolvedor da aplicação.

Já o segundo botão, com a imagem de um e-mail genérico, redireciona o usuário para a composição de uma nova mensagem em um aplicativo de e-mail à sua escolha e com o endereço de e-mail correspondente ao desenvolvedor da aplicação. Para isso, quando o botão é clicado, o sistema envia uma *intent* do tipo e-mail para o sistema *Android* com a informação do destinatário, no caso o e-mail do desenvolvedor da aplicação. Assim, como pode ser visto no exemplo da Figura 6.10, o sistema *Android* mostra ao usuário todos os aplicativos instalados em seu dispositivo que podem tratar o tipo e-mail, e oportuniza o usuário a escolher aquele que mais lhe agrada para completar a ação.

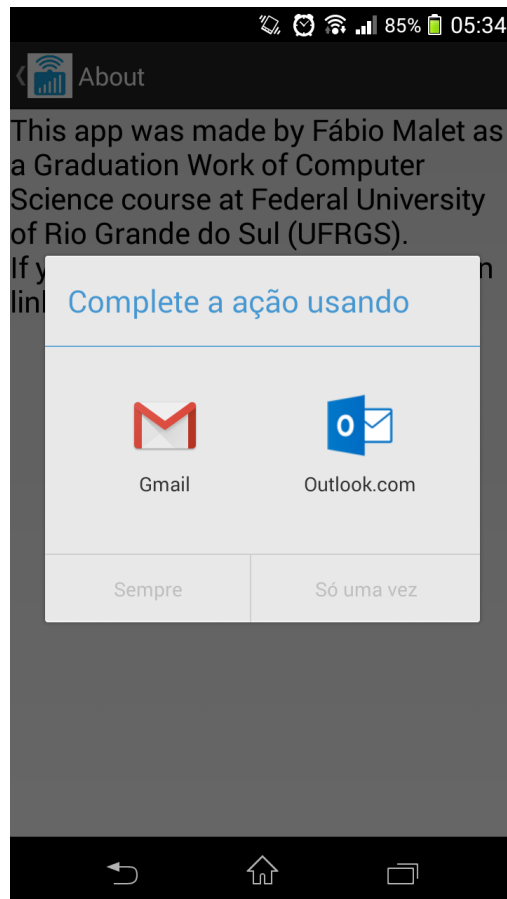


Figura 6.10 – Escolha de servidor de e-mail em opção de contato ao desenvolvedor



## 7 DESENVOLVIMENTO FUTURO

Nos parágrafos a seguir serão expostas algumas ideias importantes criadas no período de definição do escopo do trabalho a ser desenvolvido e que por motivos diversos, não puderam ser desenvolvidas. Essas abstrações proporcionariam uma maior robustez ao aplicativo e por conta disso, foram documentadas, conservadas e serão descritas a seguir, com o intuito de construir um guia de tarefas a serem realizadas no futuro.

### 7.1 Varredura de Rede

A ideia principal da opção de “Varredura de Rede” é a de automatizar a coleta de dados referente à potência, em dBm, dos sinais alcançáveis pelo dispositivo durante o deslocamento do usuário. Para isto, o usuário define o tempo que o aplicativo ficará continuamente coletando os dados e, ao final do processo, analisa os dados coletados em forma de mapa de calor, que será descrito posteriormente.

Quando o usuário escolher a opção de “Varredura de Rede”, aparecerá na tela do dispositivo uma caixa de texto a ser preenchida, permitindo apenas números e indicando ao usuário que ele deve informar o tempo total que o aplicativo ficará executando a varredura de rede em segundo plano. Ainda, nessa caixa de preenchimento, haverá dois botões que definem a ação ocorrida na sequência: “Cancelar” e “Iniciar Varredura”. Ao clicar “Cancelar”, o aplicativo volta à sua tela inicial. No caso de o usuário clicar no botão “Iniciar Varredura”, o aplicativo coleta o tempo total informado e começa a varredura de rede que terminará apenas quando a execução atingir o período indicado.

A varredura de rede consiste da execução, em segundo plano, de coletas de dados consecutivas, a cada intervalo de tempo pré-definido, durante o período decretado pelo usuário. Essa coleta de dados é composta pela definição do local onde encontra-se o usuário (definidos pela latitude e longitude do lugar), bem como a potência, em dBm, dos sinais alcançáveis pelo dispositivo nesse mesmo local.

Para exemplificar o poder que essa opção pode trazer a aplicação, suponhamos que o usuário defina uma varredura de 10 minutos e que o intervalo pré-definido para a coleta de dados seja de 5 segundos. Ao final da varredura de rede deste cenário, o usuário já teria mapeado 120 pontos de seu interesse!

### 7.1.1 *Desenvolvimento*

Em termos de código para execução da funcionalidade proposta, bem como dos elementos gráficos descritos, o desenvolvimento desta opção tem complexidade semelhante ao que já fora desenvolvido neste trabalho, sem nenhuma grande dificuldade latente deste ponto de vista.

Por outro lado, esta ação necessitaria de uma base de dados mais robusta. Em outras palavras, o aplicativo precisaria basear-se em um banco de dados puro, como por exemplo o MySQL, banco de dados comumente utilizado em grandes aplicações Android.

Mas o uso de um banco de dados robusto torna o aplicativo muito mais pesado. E, em se tratando de um aplicativo simples e desconhecido, o já baixo número de usuários interessados na utilização da ferramenta cairia drasticamente.

E, ainda, esta ação está diretamente ligada a funcionalidade que será descrita na seção 7.2. Sem o “Mapa de Calor de Rede”, a ação de “Varredura de Rede” não teria nenhuma aplicação sozinha.

## 7.2 **Mapa de Calor de Rede**

A opção de “Mapa de Calor de Rede” consiste de apanhar os dados coletados em varredura(s) de rede feita(s) pela aplicação, como descrito na seção anterior, e transformá-los em um mapa de calor, considerando intervalos de classificação de potência de sinal. Em outras palavras, a cada local registrado na varredura, associados a um par de latitude e longitude, assinala-se um ponto no mapa da região correspondendo à intensidade da potência do sinal registrada, conforme o esquema de cores do mapa de calor em questão.

Na Figura 3.4, podemos observar um exemplo de mapa de calor retirado do site OpenSignal.com (ver em detalhe sobre este projeto na seção 3.3). Nesse mapa de calor, podemos observar que os pontos mais próximos de vermelho possuem as maiores intensidades de potência de sinal das operadoras de telefonia móvel de Porto Alegre (listadas no canto inferior direito da imagem). Já os pontos mais próximos de azul, são aqueles que possuem menor intensidade de potência de sinal de telefonia móvel.

### 7.2.1 Desenvolvimento

Esta funcionalidade possui complexidade alta e depende diretamente da ação descrita na seção 7.1. Esta dificuldade dá-se por se tratar de uma ação puramente gráfica e que depende de utilização de recursos externos, como o Google Maps, por exemplo.

A parte gráfica do desenvolvimento Android possui uma curva de aprendizagem que demanda demasiado tempo para que todos conceitos sejam assimilados. O mesmo vale para recursos externos, que seriam necessários para a interação no mapa no qual o mapa de calor seria construído. Envolve não só a parte gráfica, como a definição da área selecionada, suas coordenadas e a definição dos dados coletados na região citada.

Por conta disso, o desenvolvimento dessa funcionalidade foi postergado e, como dito anteriormente, o desenvolvimento da seção 7.1 também o foi.

### 7.3 Teste de Velocidade de Rede

A opção de “Teste de Velocidade de Rede” consiste em dar possibilidade ao usuário de obter informações sobre o estado da rede em relação à taxa de *download* e *upload* de dados. Nem sempre a rede com maior potência de sinal, em dBm, provê a melhor conexão possível para o usuário do dispositivo móvel. Por conta disso, acrescentar-se-ia no menu principal da aplicação a opção de “Teste de Velocidade”. Quando o usuário selecionasse essa opção, abriria uma tela para que ele escolhesse quais das redes disponíveis na Tela Inicial do aplicativo ele gostaria de incluir no teste de velocidade, dando liberdade para que o usuário possa escolher quantas e quais redes achar necessário.

Após escolhidas as redes que o usuário quer realizar o Teste de Velocidade, o mesmo clica em “Iniciar Teste” e o teste de velocidade é realizado em todas as redes selecionadas, uma por uma.

No final da operação, as redes escolhidas seriam listadas de maneira análoga à tela inicial do aplicativo, substituindo-se apenas a coluna de “Potência do Sinal”, em dBm, por uma nova coluna chamada de “Taxa de Download” ou “Taxa de Upload”, o que o usuário escolher, em kbps.

### *7.3.1 Desenvolvimento*

Esta funcionalidade demanda um extenso processamento e por isso, não era indicado para uma versão inicial.

Para que esta funcionalidade seja realizada, o aplicativo deverá ter permissões de conectar-se e desconectar-se da rede, tanto de telefonia móvel, quanto Wi-Fi. O usuário deverá estar ciente de que taxas de navegação de dados podem ser cobradas. O aplicativo deverá ter permissões para enviar requisições externas, para ser possível medir-se o a velocidade da transmissão de dados em cada rede escolhida.

Devido a esse extenso processamento necessário, a quantidade de permissões indispensáveis e à complexidade do desenvolvimento desta funcionalidade, a mesma foi postergada para versões futuras.

## 8 CONCLUSÕES

Neste trabalho foi abordado o estudo de mecanismos para análise de potência de sinal em dispositivos móveis, e, baseado nestes estudos, foi elaborado um aplicativo que auxiliasse o usuário na localização da maior potência de sinal possível entre as redes analisadas. Enquanto a coleta de informações de nome e potência das redes alcançáveis pelo dispositivo são facilmente providas pelas bibliotecas da API Android utilizada, o mesmo não pode ser dito sobre a localização das antenas referidas. Essas antenas não possuem registro de localização geográfica disponível em lugar algum e, por conta disso, foi elaborado um método alternativo que tentasse suprir a necessidade do usuário em localizar a maior potência de sinal possível da rede desejada.

O aplicativo desenvolvido durante o trabalho atende algumas das situações hipotéticas levantadas na Introdução. Diferentemente de muitos outros aplicativos disponíveis no mercado, este aplicativo trabalha sem necessidade de conexão com a Internet, mostrando as mudanças de sinal em tempo real e fornecendo a possibilidade de poder salvar lugares personalizados, fato este que auxilia na utilização do usuário final.

Por fim, pode-se concluir que todos os objetivos propostos no princípio desta monografia foram cumpridos, incluindo o desenvolvimento de um aplicativo que resolve o problema que deu origem ao tema escolhido. Prova disso é a marca de mais de 100 *downloads* alcançadas pelo aplicativo em pouco mais de um mês de disponibilização na Google Play Store. Em se tratando de um aplicativo acadêmico e sem grande divulgação, este número de instalações da ferramenta desenvolvida é bem expressivo.

## REFERÊNCIAS

ANDROID. **<uses-sdk>**. Disponível em:

<http://developer.android.com/guide/topics/manifest/uses-sdk-element.html>. Acesso em: 4 jul. 2015.

ANDROID. **Activity**. Disponível em:

<http://developer.android.com/reference/android/app/Activity.html>. Acesso em: 4 jul. 2015.

ANDROID. **Dashboard**. Disponível em:

[https://developer.android.com/about/dashboards/index.html?utm\\_source=suzunone#Platform](https://developer.android.com/about/dashboards/index.html?utm_source=suzunone#Platform). Acesso em: 6 jul. 2015.

ANDROID. **Intent**. Disponível em:

<http://developer.android.com/reference/android/content/Intent.html>. Acesso em: 4 jul. 2015.

ANDROID. **NeighboringCellInfo**. Disponível em:

<http://developer.android.com/reference/android/telephony/NeighboringCellInfo.html>. Acesso em 5 jul. 2015.

ANDROID. **ScanResult**. Disponível em:

<http://developer.android.com/reference/android/net/wifi/ScanResult.html>. Acesso em: 4 jul. 2015.

ANDROID. **SignalStrength**. Disponível em:

<http://developer.android.com/reference/android/telephony/SignalStrength.html>. Acesso em: 4 jul. 2015.

ANDROID. **TelephonyManager**. Disponível em:

<http://developer.android.com/reference/android/telephony/TelephonyManager.html>. Acesso em: 4 jul. 2015.

ANDROID. **View**. Disponível em:

<http://developer.android.com/reference/android/view/View.html>. Acesso em 4 jul. 2015.

ANDROID. **WifiManager**. Disponível em: <http://developer.android.com/reference/android/net/wifi/WifiManager.html>. Acesso em: 4 jul. 2015.

ANTENNAS. **Antennas for Android 1.0**. Disponível em: <http://www.panix.com/~mpoly/android/antennas/r1.0/>. Acesso em: 5 jul. 2015.

COLEMAN, D. D; WESTCOTT, D. A. **CWNA: Certified Wireless Network Administrator Official Study Guide**. 3. ed. [S.l.]: John Wiley & Sons, 2012.

ETSI. **TS 127 007 V8.5.0**. Disponível em: [http://www.etsi.org/deliver/etsi\\_ts/127000\\_127099/127007/08.05.00\\_60/ts\\_127007v080500p.pdf](http://www.etsi.org/deliver/etsi_ts/127000_127099/127007/08.05.00_60/ts_127007v080500p.pdf). Acesso em 28 jun. 2015.

GRAF, R. F. **Modern Dictionary of Eletronics**. 7. ed. [S.l.]: Newnes, 1999.

GRIFFITH, W. T. **The Physics of Everyday Phenomena: A Conceptual Introduction to Physics**. 6. ed. [S.l.]: McGraw-Hill Higher Education, 2008.

HELD, G. **Building A Wireless Office**. [S.l.]: CRC Press, 2002.

KHOSROW-POUR, M. **Dictionary of Information Science and Technology**. [S.l.]: Idea Group Inc (IGI), 2006.

MISHRA, A. R. **Fundamentals of Cellular Network Planning and Optimisation: 2G/2.5G/3G... Evolution to 4G**. [S.l.]: John Wiley & Sons, 2004.

OPEN SIGNAL. **OpenSignal**. Disponível em: <http://pt.opensignal.com/>. Acesso em 5 jul. 2015.

REID, N. P; SEIDE, R. **802.11 (Wi-Fi): Networking Handbook**. [S.l.]: McGraw-Hill/Osborne, 2003.

SARKAR, T. K; SALAZAR-PALMA, M; MOKOLE, E. L. **Physics of Multiantenna Systems and Broadband Processing**. [S.l.]: John Wiley & Sons, 2008.

SHEPHERD, A. J. **Teaser: Samsung Galaxy S5 gets a boost via Wi-Fi but not carrier aggregation**. 10 mar. 2014. Disponível em:

<http://s4gru.com/index.php?/blog/1/entry-363-teaser-samsung-galaxy-s5-gets-a-boost-via-wi-fi-but-not-carrier-aggregation/>. Acesso em: 13 jun. 2015.

SILVEIRA, J. F. P. **O decibel, ou melhor: os decibéis**. Disponível em: <http://www.mat.ufrgs.br/~portosil/passa1e.html>. Acesso em: 18 jun. 2015.

TECHNOPEDIA. **What is cellular network?** Disponível em: <http://www.techopedia.com/definition/24962/cellular-network>. Acesso em: 2 jul. 2015.

WEBOPEDIA. **What is Wi-Fi (IEEE 802.11x)?** Disponível em: [http://www.webopedia.com/TERM/W/Wi\\_Fi.html](http://www.webopedia.com/TERM/W/Wi_Fi.html). Acesso em: 3 jul. 2015.

WI-FI ALLIANCE. **Wi-Fi**. Disponível em: <http://www.wi-fi.org>. Acesso em 5 jul. 2015.

WIFI ANALYZER. **WiFi Analyzer**. Disponível em: <http://a.farproc.com/wifi-analyzer>. Acesso em: 5 jul. 2015.

WIKIPEDIA. **Antenna (radio)**. Disponível em: [https://en.wikipedia.org/wiki/Antenna\\_\(radio\)](https://en.wikipedia.org/wiki/Antenna_(radio)). Acesso em: 13 jun. 2015.

WIKIPEDIA. **Cellular Network**. Disponível em: [https://en.wikipedia.org/wiki/Cellular\\_network](https://en.wikipedia.org/wiki/Cellular_network). Acesso em: 2 jul. 2015.

WIKIPEDIA. **dBm**. Disponível em: <https://en.wikipedia.org/wiki/DBm>. Acesso em: 19 jun. 2015.

WIKIPEDIA. **Mobile phone signal**. Disponível em: [https://en.wikipedia.org/wiki/Mobile\\_phone\\_signal](https://en.wikipedia.org/wiki/Mobile_phone_signal). Acesso em: 2 jul. 2015.

WIKIPEDIA. **Radio wave**. Disponível em: [https://en.wikipedia.org/wiki/Radio\\_wave](https://en.wikipedia.org/wiki/Radio_wave). Acesso em: 14 jun. 2015.

WIKIPEDIA. **Wi-Fi**. Disponível em: <https://en.wikipedia.org/wiki/Wi-Fi>. Acesso em 3 jul. 2015.