

# **MAPEAMENTO E ANÁLISE DA INTENSIDADE DE SINAIS WI-FI UTILIZANDO TÉCNICAS DE GEOLOCALIZAÇÃO E VISUALIZAÇÃO DE DADOS**

Nícolas RUZZA

[nic.ruzza@gmail.com](mailto:nic.ruzza@gmail.com)

Ciência da Computação, Unianchieta

Vinícius Eduardo ROVERI

[viroveri117@gmail.com](mailto:viroveri117@gmail.com)

Ciência da Computação, Unianchieta

Clayton Augusto VALDO

[clayton.valdo@anchieta.br](mailto:clayton.valdo@anchieta.br)

Ciência da Computação, Unianchieta

# **MAPEAMENTO E ANÁLISE DA INTENSIDADE DE SINAIS WI-FI UTILIZANDO TÉCNICAS DE GEOLOCALIZAÇÃO E VISUALIZAÇÃO DE DADOS**

## **WI-FI SIGNAL STRENGTH MAPPING AND ANALYSIS THROUGH GEOLOCATION AND DATA VISUALIZATION**

### **Resumo**

O Wi-Fi, baseado no padrão IEEE 802.11, consolidou-se como uma tecnologia fundamental na comunicação digital contemporânea, mas sua eficácia é frequentemente comprometida por interferências, atenuação de sinal e áreas de sombra, problemas cuja identificação é comumente realizada de forma empírica e imprecisa. Diante da massiva adoção de dispositivos móveis e de IoTs no Brasil, a otimização da cobertura de redes sem fio torna-se crucial. Este trabalho teve como objetivo principal investigar, aplicar e validar técnicas de análise e mapeamento da intensidade de sinais Wi-Fi, medida pelo Indicador de Força do Sinal Recebido (RSSI), através da sua correlação direta com dados de geolocalização. Para tanto, foi realizada uma pesquisa abrangente sobre os padrões IEEE 802.11 operando em 2,4 GHz e 5 GHz, as métricas de sinal e os fenômenos de propagação, como atenuação e reflexão. Foi explorada a utilização dos sistemas de navegação global (GNSS) e, especificamente do A-GPS para viabilizar uma localização rápida e precisa, mesmo em ambientes internos ou semi-internos. A plataforma Android foi escolhida como base de desenvolvimento, explorando sua arquitetura sobre o kernel Linux e suas APIs nativas do sistema operacional. O método de interpolação espacial de Estimção de Densidade por Kernel (KDE), foi utilizado como técnica estatística para transformar pontos de dados discretos em uma superfície contínua e visualmente interpretável. Com intuito de realizar uma avaliação prática, foi desenvolvido uma aplicação em Java, no qual permite ao usuário realizar a coleta de dados fornecidos pelo ponto de acesso sem fio e localização atual em tempo real, e renderizar um mapa de calor correlacionando essas métricas.

### **Palavras-Chave**

Wi-Fi; RSSI; Geolocalização; Kernel Density Estimation; Mapas de calor.

### **Abstract**

Wi-Fi, based on the IEEE 802.11 standard, has established itself as a fundamental technology in contemporary digital communication, but its effectiveness is often compromised by interference, signal attenuation, and shadow areas, problems whose identification is commonly carried out empirically and imprecisely. In view of the massive adoption of mobile devices and IoTs in Brazil, optimizing wireless network coverage becomes crucial. The main objective of this project was to investigate, apply, and validate techniques for analyzing and mapping the intensity of Wi-Fi signals, measured by the Received Signal Strength Indicator (RSSI), through its direct correlation with geolocation data. To this end, comprehensive research was conducted on the IEEE 802.11 standards operating at 2.4 GHz and 5 GHz, signal metrics, and propagation phenomena such as attenuation and reflection. The use of global navigation systems (GNSS), and specifically A-GPS, was explored to enable fast and accurate location, even in indoor or semi-indoor environments. The Android platform was chosen as the development base, exploring its architecture on the Linux kernel and its native operating system APIs. The spatial interpolation method of Kernel Density Estimation (KDE), used as a statistical technique to transform discrete data points into a continuous and visually interpretable surface. Aiming to carry out a practical evaluation, an application was developed in Java, which allows the user to collect data provided by the wireless access point and current location in real time, and render a heat map correlating these metrics.

### **Keywords**

Wi-Fi; RSSI; Geolocation; Kernel Density Estimation; Heat Maps.

## 1. INTRODUÇÃO

Desde os primeiros desenvolvimentos das redes sem fio baseadas no padrão IEEE 802.11 até as modernas infraestruturas que operam nas faixas de 2,4 GHz, 5 GHz e 6 GHz, o Wi-Fi consolidou-se como uma das tecnologias mais importantes da comunicação digital contemporânea.

Segundo a organização WI-FI ALLIANCE (2018), o Wi-Fi consolidou-se como elemento essencial para viabilizar a mobilidade, a alta largura de banda e a conectividade ubíqua, transformando a forma como indivíduos, empresas e instituições acessam e compartilham informações. Mais do que apenas permitir acesso à internet, essa tecnologia passou a sustentar aplicações avançadas, como realidade aumentada, realidade virtual, vídeo em alta definição, jogos colaborativos e Internet das Coisas, ampliando seu papel estratégico no cenário digital contemporâneo. Entretanto, junto a tais benefícios, permanecem limitações relacionadas à interferência entre dispositivos, à fragmentação de canais e às variações abruptas na qualidade do sinal, fatores que impactam diretamente a confiabilidade e a experiência de uso. Tais aspectos tornam imprescindível o monitoramento sistemático da intensidade do sinal e sua análise espacial como ferramentas para compreender e otimizar a cobertura das redes sem fio.

Nesse cenário, é importante destacar a dimensão do uso de dispositivos móveis no Brasil, que intensifica a dependência por redes sem fio de qualidade. Em 2024, o país contava com 167,5 milhões de pessoas com 10 anos ou mais de idade utilizando celular, o que correspondia a 88,9% da população nessa faixa etária (IBGE, 2024).

Em ambientes residenciais, empresariais e públicos, regiões com baixa intensidade de sinal, falhas de conectividade e áreas de sombra são problemas comuns, pois o concreto, metal e a topografia dificultam a propagação das ondas (Moraes et al., 2025). Na maioria das vezes, a identificação desses pontos críticos ocorre de forma empírica, por meio da percepção subjetiva, sem embasamento técnico consistente. Essa abordagem limita a capacidade de realizar diagnósticos precisos e, conseqüentemente, restringe o planejamento de soluções mais estruturadas para otimizar o desempenho das redes. Embora existam ferramentas para coleta e visualização de redes Wi-Fi, muitas são limitadas ou requerem um desembolso, e por vezes não integram os dados de geolocalização com precisão, o que dificulta análises mais detalhadas da infraestrutura.

Segundo Peres (2010, p. 12), os sinais de micro-ondas podem atravessar paredes e sofrer diferentes tipos de interferências, como atenuação, reflexão, refração, difração e dispersão, o que dificulta a definição precisa dos limites de alcance de uma rede sem fio.

O presente estudo propõe-se a investigar a intensidade de sinais Wi-Fi por meio da correlação entre valores de RSSI e dados de localização geográfica, de forma a produzir representações visuais como mapas de calor. Para tanto, serão desenvolvidos procedimentos de coleta automatizada que associam medições a coordenadas, bem como aplicadas técnicas de interpolação espacial, como o Kernel Density Estimation (KDE) explorado por Silverman (1986), visando representar com fidelidade as variações de sinal nos ambientes analisados.

Este trabalho tem como objetivo investigar e aplicar técnicas de análise e mapeamento da intensidade de sinais Wi-Fi, integrando informações de geolocalização para geração de representações visuais. Busca-se compreender os desafios técnicos envolvidos na coleta de dados de rede sem fio, no tratamento das medições de intensidade de sinal e na construção de mapas de calor que auxiliem a análise da cobertura e do desempenho de redes Wi-Fi.

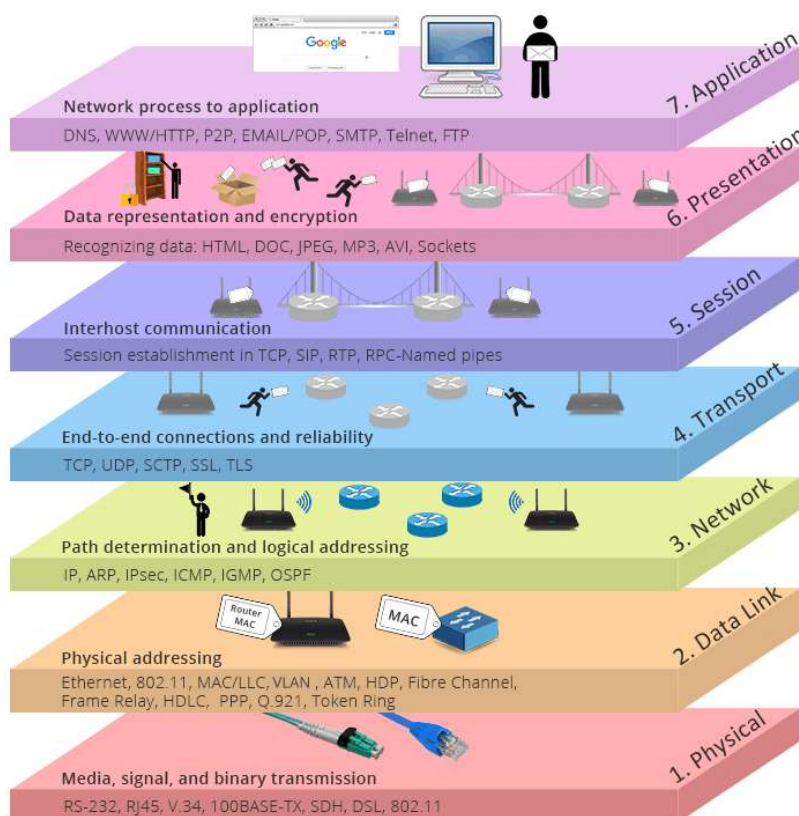
## 2. FUNDAMENTAÇÃO

Este capítulo detalha os conceitos essenciais que sustentam o desenvolvimento prático de uma aplicação que utiliza o celular para realização de mapeamento e análise de sinais Wi-Fi. A abordagem explora as tecnologias de redes sem fio, junto a sistemas de geolocalização, fingerprint, a plataforma Android e técnicas de interpolação de dados. Formando uma solução que contribui para novas oportunidades exploratórias a tecnologia Wi-Fi e os futuros desafios sobre a interferência deste tipo de comunicação sem fio. Conforme Tanenbaum, Feamster e Wetherall (2021) “Computadores móveis, como notebooks, tablets e smartphones, constituem um dos segmentos de mais rápido crescimento do setor de informática.”

## 2.1 Redes sem fio padrão IEEE 802.11

A comunicação de uma rede local sem fio (WLAN) é realizada por meio de sinais de micro-ondas, que ao se propagarem, podem sofrer fenômenos como atenuação, reflexão, refração e difração, tornando a delimitação da área de cobertura uma tarefa complexa, sendo necessário o uso de tecnologias para visualização do efeito causado por estes fenômenos. A antena dos celulares é responsável pela captura e envio dessas ondas (Moraes et al., 2025, p. 19). Para padronização dos equipamentos que se comunicam em uma rede sem fio foi criado o padrão IEEE 802.11, estabelecido em 1997, ele define as especificações para as camadas física e de enlace de dados do modelo OSI, conforme ilustrado na Figura 1, para redes sem fio. A Anatel é responsável por aprovar o uso deste padrão no Brasil, fazendo com que todos dispositivos aprovados legalmente utilizem ele. (Anatel, 2019)

**Figura 1.** Modelo OSI (Cisco, 2023)

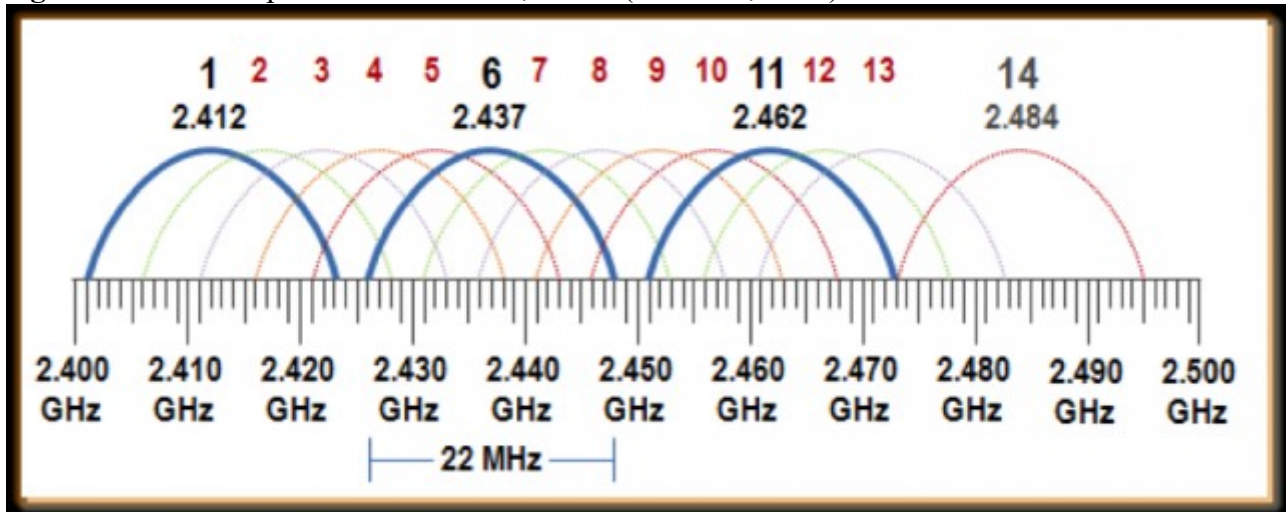


As tecnologias de rede sem fio padrão 802.11 podem operar em bandas de frequência distintas, como 2,4 GHz e 5 GHz. Cada uma possui vantagens e desvantagens relacionadas à interferência, alcance, capacidade de penetração em obstáculos e canais disponíveis, conforme ilustram a Figura 2 e Figura 3. Sobre essas características, Tanenbaum, Feamster e Wetherall (2021, p. 201) explicam que:

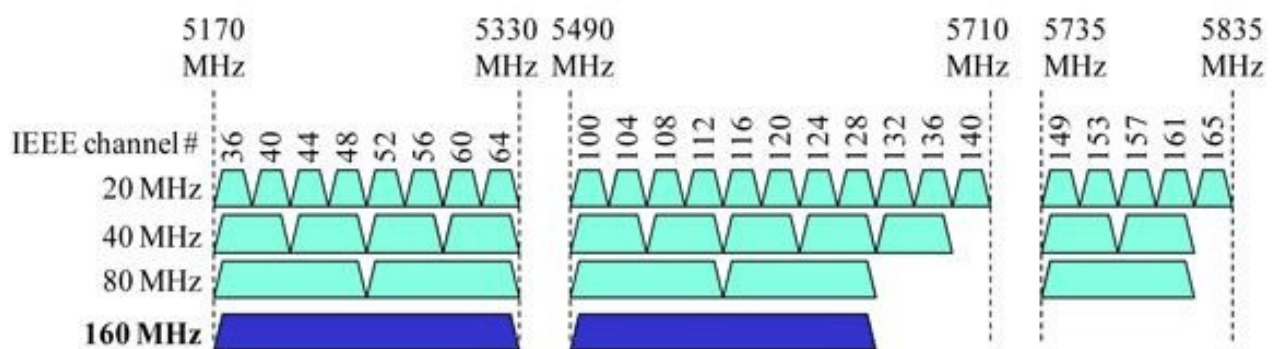
Todas as técnicas do 802.11 utilizam rádios de curto alcance para transmitir sinais nas bandas de frequência ISM de 2,4 GHz ou 5 GHz. Essas bandas têm a vantagem de não ser licenciadas e, portanto, estar disponíveis gratuitamente a qualquer transmissor que queira cumprir algumas restrições, como a potência irradiada de no máximo 1 W (embora 50 mW seja mais comum para rádios de LAN sem fio). Infelizmente, esse fato também é conhecido pelos fabricantes de aparelhos de abertura automática de garagem, telefones sem fio, fornos de micro-ondas e diversos outros dispositivos, todos competindo com os notebooks e smartphones pelo mesmo espectro. A banda de 2,4 GHz costuma ser mais sobrecarregada do que a de 5 GHz, de modo que esta pode ser melhor para algumas aplicações, embora tenha um alcance mais curto, em virtude da frequência mais alta. Infelizmente, as ondas de rádio mais curtas a 5 GHz não penetram em paredes tão bem quanto as de 2,4 GHz, de modo que 5 GHz não é uma vantagem definitiva.

Desde a sua criação, o padrão IEEE 802.11 evoluiu significativamente. A versão original (802.11) operava em 2,4 GHz, com taxas de até 2 Mbps. O padrão 802.11b introduziu o espalhamento espectral de alta taxa (HR-DSSS), alcançando 11 Mbps na mesma frequência. Posteriormente, os padrões 802.11a e 802.11g adotaram a multiplexação por divisão de frequência ortogonal (OFDM), permitindo taxas de até 54 Mbps. As gerações mais recentes, como o 802.11n (Wi-Fi 4), 802.11ac (Wi-Fi 5) e 802.11ax (Wi-Fi 6), continuaram a aprimorar a velocidade e a eficiência no uso do espectro. (IEEE, 2023)

**Figura 2.** Canais disponíveis na banda 2,4 GHz (Coleman, 2012)



**Figura 3.** Canais disponíveis banda 5 GHz (Vitanova, 2017)



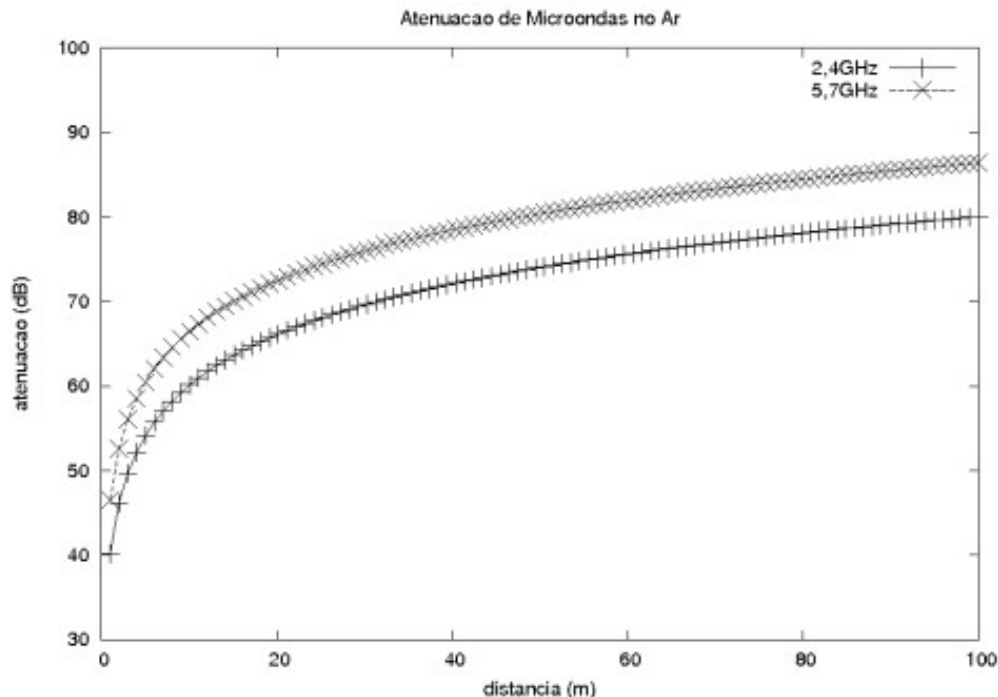
### 2.1.1 Potência e atenuação

Para analisar a qualidade de uma conexão sem fio, utilizam-se métricas como o Indicador de Força do Sinal Recebido (RSSI) e a Relação Sinal-Ruído (SNR). O RSSI representa a potência do sinal captado por um dispositivo, geralmente expressa em decibéis-miliwatts (dBm), conforme visível no índice de atenuação na Figura 4. Já o SNR compara a potência do sinal recebido com o nível de ruído de fundo do ambiente. No Android, não é possível obter diretamente o valor SNR, sendo o RSSI a principal ferramenta de análise para o mapeamento da intensidade e qualidade do sinal Wi-Fi. (Google, 2025d)

Como qualquer sinal enviado em um meio de comunicação, os sinais de microondas sofrem atenuação, ou seja, perda de potência de sinal devido à transmissão. No caso da propagação de microondas, dois fatores influenciam na perda desta potência: a distribuição do sinal no espaço durante a transmissão; e a capacidade da antena de recepção de captar o sinal. (Peres 2010, p. 20)

Sempre que o sinal é propagado, espalha-se no espaço conforme o ângulos de abertura da antena de transmissão. A potência do sinal é distribuída na área de propagação até atingir a antena de recepção. (Peres 2010, p. 20)

**Figura 4.** Índice de atenuação em dB por metro para micro-ondas a 2,4GHz e 5,7GHz (Peres, 2010)



## 2.2 Sistema Global de Navegação por Satélite (GNSS)

Conforme dito por Hofmann-Wellenhof, Lichtenegger e Wasle (2008), GNSS (Global Navigation Satellite System) é um termo definido como um sistema de posicionamento por rádio baseado no espaço, que pode incluir uma ou mais constelações de satélites (eventualmente com sistemas de aumento/auxílio), com a função de fornecer informações de posição tridimensional, velocidade e tempo, 24 horas por dia, a usuários adequadamente equipados em qualquer lugar da superfície terrestre.

O ponto central é que o GNSS não se limita a um único sistema, mas sim a um conjunto de sistemas globais de navegação por satélite. Exemplos incluem o GPS, o GLONASS, e o Galileo. (Hofmann-Wellenhof; Lichtenegger; Wasle. 2008)

### 2.2.1 Assistance - GPS (A-GPS)

O Assisted Global Positioning System (A-GPS) foi desenvolvido para superar limitações do GPS convencional, que apresenta tempo de inicialização elevado, em torno de um minuto, e baixa confiabilidade em aplicações civis que exigem posicionamento imediato, inclusive em ambientes internos. Tais restrições decorrem da fraca intensidade dos sinais de satélite e do tempo necessário para localizar e decodificar as informações orbitais de cada um deles. O A-GPS mitiga esse problema ao fornecer dados de assistência, como posições de satélite e frequências previstas, por meio de canais de comunicação rápidos, como redes celulares, o que não substitui a recepção dos sinais, mas acelera sua aquisição, reduzindo o tempo para a primeira fixação de minutos para segundos e aumentando a sensibilidade do sistema, possibilitando seu funcionamento em cenários mais desafiadores. (Diggelen, 2009)

Existem duas abordagens principais para o GPS assistido, conhecidas como GPS assistido por MS e GPS baseado em MS. "MS" significa estação móvel, ou seja, o receptor GPS. No GPS assistido por MS, a posição é calculada em um servidor, e a função do receptor GPS é apenas adquirir os sinais e enviar as medições para o servidor. No GPS baseado em MS, a posição é calculada pelo próprio receptor. Essa distinção primária leva a outras distinções. Se o receptor não for calcular a posição, ele não precisa necessariamente de dados de órbita de satélite (como um almanaque ou efemérides). É possível mantê-los no servidor, e o servidor pode calcular diretamente os dados de assistência de aquisição e enviá-los ao receptor. (Diggelen, 2009, p. 43, tradução própria "1")

### 2.2.2 Aplicação da geolocalização na coleta de dados espaciais

O método de posicionamento absoluto, no qual o receptor calcula suas coordenadas de forma autônoma a partir dos sinais recebidos de, no mínimo, quatro satélites da constelação GNSS. Conforme detalhado por Hofmann-Wellenhof, Lichtenegger e Wasle (2008), este princípio baseia-se na medição das pseudodistâncias aos satélites para determinar uma posição tridimensional na superfície terrestre.

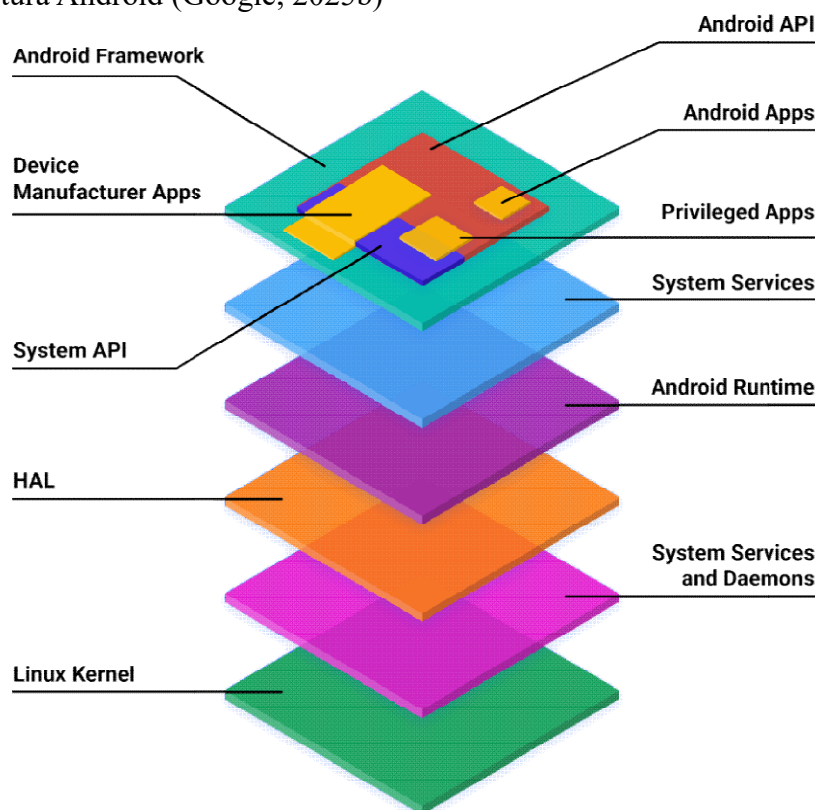
### 2.3 Arquitetura Android baseada no kernel Linux

O Android adota o kernel Linux como a camada fundamental de sua arquitetura de software, a qual pode ser observada na Figura 5, atuando como uma camada de abstração entre o hardware do dispositivo e o restante da pilha de softwares. Essa base Linux é responsável por gerenciar componentes essenciais do sistema, como gerenciamento de memória, processos, segurança e, crucialmente para este projeto, os drivers de rede, incluindo o da interface Wi-Fi. Acima do kernel, a Camada de Abstração de Hardware (HAL) fornece uma interface padrão que expõe os recursos de hardware para o framework de aplicativos, independentemente das particularidades do hardware. (Google, 2025a).

Conforme define a Google (2025a) na documentação oficial do Android:

Uma camada de abstração de hardware (HAL) é um tipo de camada de abstração com uma interface padrão para implementação por fornecedores de hardware. Um HAL permite que os fornecedores de hardware implementem recursos de nível mais baixo e específicos do dispositivo sem afetar ou modificar o código em camadas de nível mais alto.

**Figura 5.** Arquitetura Android (Google, 2025b)



Embora utilize o kernel Linux, o Android não é uma distribuição Linux tradicional. Desenvolvido pela Open Handset Alliance, liderada pelo Google, e mantido pelo Android Open Source Project (AOSP), o sistema foi otimizado para dispositivos móveis. Diferentemente das distribuições de desktop, o Android não incorpora o sistema X Window nem as bibliotecas GNU padrão, como a Glibc. Em seu lugar, utiliza sua própria biblioteca C, chamada Bionic, e um ambiente de execução otimizado (ART) para os aplicativos que são majoritariamente escritos em Java ou Kotlin. (Al-Rayes, 2012).



### 2.3.1 Interface Wi-Fi no Android

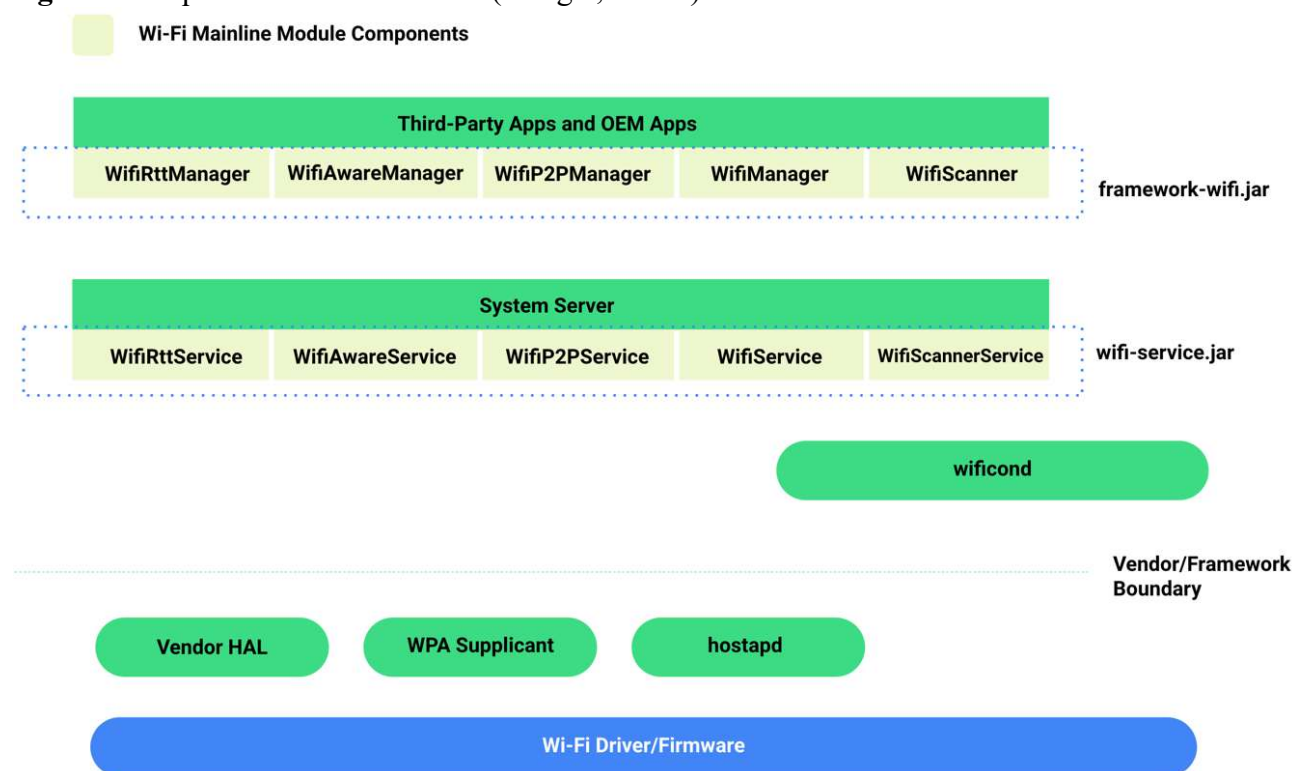
A base Linux do Android oferece vantagens significativas para tarefas que envolvem análise de rede, possibilitando acesso de baixo nível às interfaces de rede e fornecendo uma base flexível para o monitoramento e gerenciamento de conexões, a arquitetura e as camadas de acesso podem ser visualizadas na Figura 6 (Google, 2025b).

Isto é reforçado pelo modo como é estruturada a comunicação da API em Java com a camada em C/C++ e HAL, como descrito por Tanenbaum e Bos, (2016, p. 558)

Uma parte significativa do sistema operacional Android é escrita em uma linguagem de alto nível, a linguagem de programação Java. O núcleo e um grande número de bibliotecas de baixo nível são escritos em C e C++. No entanto, uma grande parte do sistema é escrita em Java e, com algumas poucas exceções, toda a API para aplicações é escrita e publicada em Java também. As partes do Android escritas em Java tendem a seguir um projeto bastante orientado a objetos como encorajado por aquela linguagem.

O framework de aplicativos do Android disponibiliza APIs ricas que permitem aos desenvolvedores acessar informações detalhadas sobre a conexão Wi-Fi, como o RSSI, SSID, BSSID, canal, largura do canal, método de segurança, geração do padrão IEEE, entre outras. (Google, 2025d).

**Figura 6.** Arquitetura Wi-Fi Android (Google, 2025c)



### 2.4 Interpolação de dados espaciais para estimativa de valores

Câmara, Davis e Monteiro (2001) definem a interpolação espacial como um conjunto de técnicas para estimar valores em locais não amostrados a partir de um conjunto de pontos conhecidos, o que é fundamental para a criação de superfícies contínuas.

A coleta de dados produz um conjunto de pontos discretos, onde cada ponto é definido por uma coordenada (x, y) e um valor de intensidade de sinal (z). A interpolação espacial é uma técnica estatística utilizada para estimar os valores de sinal em locais não amostrados, gerando uma superfície contínua a partir dos pontos discretos. Esse processo é vital para gerar uma visualização completa da cobertura de sinal em toda a área de interesse, e não apenas nos locais onde as medições foram realizadas. (Câmara; Davis; Monteiro, 2001)

Um aspecto central no uso da tecnologia de Geoprocessamento advém da característica lógico-matemática de sistemas de informação; para ser representada em ambiente computacional, temos de associar a cada tipo de informação geográfica uma escala de medida e de referência, que será utilizada pelo GIS para caracterizá-lo. (Câmara; Davis; Monteiro, 2001)



### 2.4.1 Método de estimação de densidade: Kernel Density Estimation (KDE)

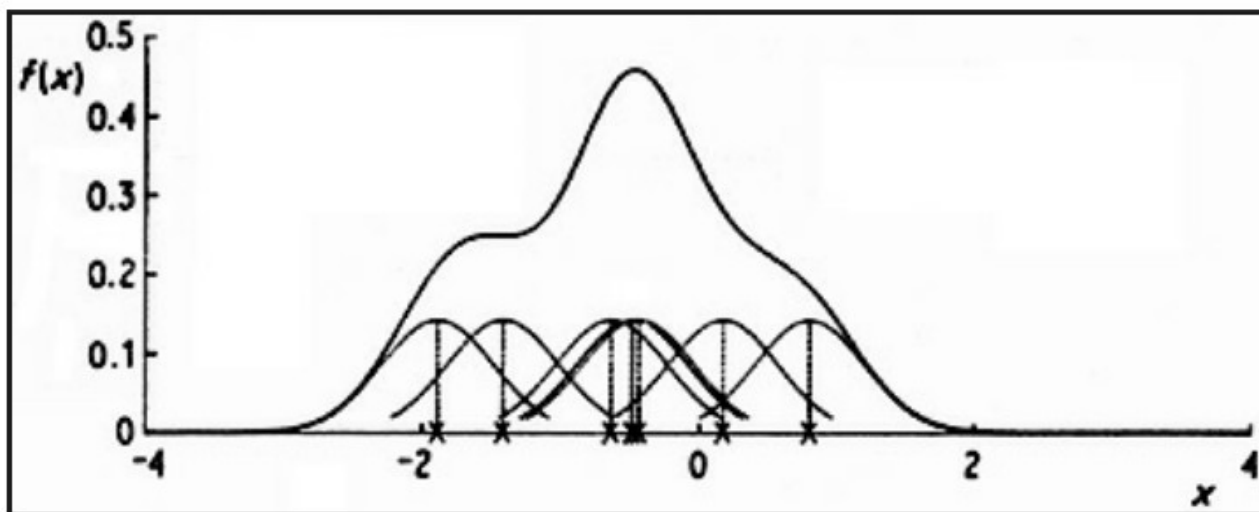
A Estimação de Densidade por Kernel (KDE), introduzida por Parzen (1962), é uma metodologia não-paramétrica consagrada para inferir a função de densidade de probabilidade de um conjunto de dados. Por ser não-paramétrica, essa abordagem não assume uma distribuição subjacente específica para os dados, baseando-se no uso de uma função kernel para construir uma estimativa suave e contínua. Conforme destacado por Turlach (1999), uma de suas aplicações mais importantes é a suavização de dados.

Assim como o estimador ingênuo pode ser considerado uma soma de 'caixas' centradas nas observações, o estimador kernel é uma soma de 'lombadas' posicionadas nas observações. A função kernel  $K$  determina o formato das lombadas, enquanto a largura da janela  $h$  determina a sua largura. [...], na qual são mostradas tanto as lombadas individuais  $n - 1h - 1K\{(x - X_i)/h\}$  quanto a estimativa final construída pela soma delas. (Silverman, 1986)

A Figura 7 ilustra as lombadas individuais descritas por Silverman (1986), onde o cálculo da somatória dos Kernels pode ser representada da seguinte maneira:

$$\hat{f}(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - X_i}{h}\right)$$

**Figura 7.** Estimativa do kernel mostrando kernels individuais (Silverman, 1986)



O KDE apresenta propriedades de suavidade superiores aos estimadores discretos, permitindo a análise de estruturas locais da distribuição com maior sensibilidade matemática. Conforme detalhado por Scott (2015), o uso de kernels diferenciáveis garante a obtenção de derivadas contínuas da densidade estimada, o que viabiliza aplicações mais avançadas, tais como análise de curvatura, identificação de máximos locais e estimativas do gradiente de densidade. Esses atributos tornam a técnica particularmente adequada para problemas em que a topologia da distribuição desempenha papel central, permitindo a refinação acerca de modos de concentração e regiões de rarefação de eventos observados.

Silverman (1986) demonstra que a utilização de kernels com suporte infinito, como o Gaussiano, tende a produzir estimativas com propriedades assintóticas desejáveis, convergindo pontualmente para a verdadeira densidade conforme o tamanho amostral cresce. Além disso, sob determinadas condições de suavidade, o KDE é consistente, ou seja, a probabilidade de a estimativa se afastar da densidade real converge para zero à medida que o número de observações aumenta (Wand; Jones, 1995). Esse comportamento evidencia a robustez do estimador, tornando-o aplicável tanto em contextos exploratórios quanto inferenciais.

#### 2.4.2 Mapas de calor como ferramenta de visualização

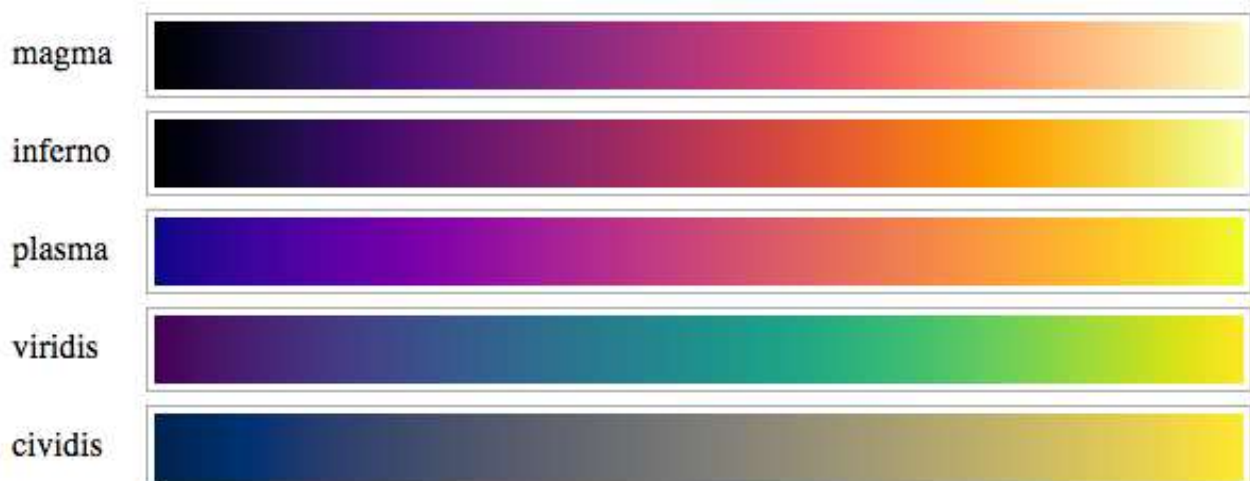
Mapas de calor são amplamente utilizados como ferramenta para representar a amplitude de valores em espaços bidimensionais, onde cores são empregadas para indicar intensidades relativas. Essa abordagem facilita a identificação de padrões, clusters e tendências em conjuntos de dados complexos, sendo útil em áreas como análise geoespacial, epidemiologia, astrofísica, engenharia e monitoramento ambiental. Ao traduzir grande volume de informação numérica em gradientes colorimétricos, o mapa de calor permite uma interpretação intuitiva, reduzindo a carga cognitiva do observador (Tufté, 2007).

Para geração dessa representação contínua, um dos métodos mais consolidados é a Estimção de Densidade por Kernel (KDE), responsável por suavizar a distribuição de pontos discretos em uma superfície rasterizada. O KDE calcula a densidade de valores em regiões específicas, reduzindo ruídos locais e destacando tendências de amplitude ao longo de uma superfície. Após a geração do raster, aplica-se uma paleta cromática para realçar visualmente variações, produzindo camadas que podem ser sobrepostas a mapas ou modelos espaciais, permitindo análises rápidas e consistentes (Silverman, 1986).

A escolha do mapa de calor exerce impacto significativo na interpretação dos resultados. Por décadas, gradientes do tipo rainbow, espalhados por diversos matizes, foram amplamente adotados, porém, estudos demonstram que suas transições abruptas e variação irregular de luminância podem introduzir erros interpretativos, induzindo o observador a perceber limites inexistentes ou hierarquias distorcidas (Borland; Ii, 2007). Entretanto, quando projetados com cuidados perceptuais, esses mapas de cores podem ser eficazes para tarefas específicas, como segmentação visual e detecção de fronteiras (Ware; Stone; Szafr, 2023).

Em resposta às limitações perceptuais do rainbow, foram propostos mapas de calor cientificamente calibrados, como o magma, inferno, plasma, viridis e cividis. Pode-se visualizar cada gradiente na Figura 8, desenvolvidos com luminância monotônica e ordenação perceptual estável. Esses gradientes são resultado de estudos em psicofísica e percepção humana, reduzindo ambiguidades entre níveis próximos e aumentando a acessibilidade para indivíduos com daltonismo (Nuñez; Anderton; Renslow, 2018). Sua construção é guiada por modelos perceptuais avançados, como o CIECAM02 proposto pela Comissão Internacional de Iluminação (CIE), que aproximam a visualização da resposta humana real (Tastl et al., 2005).

**Figura 8.** Gradiente de cores (Czaplicki, 2018)



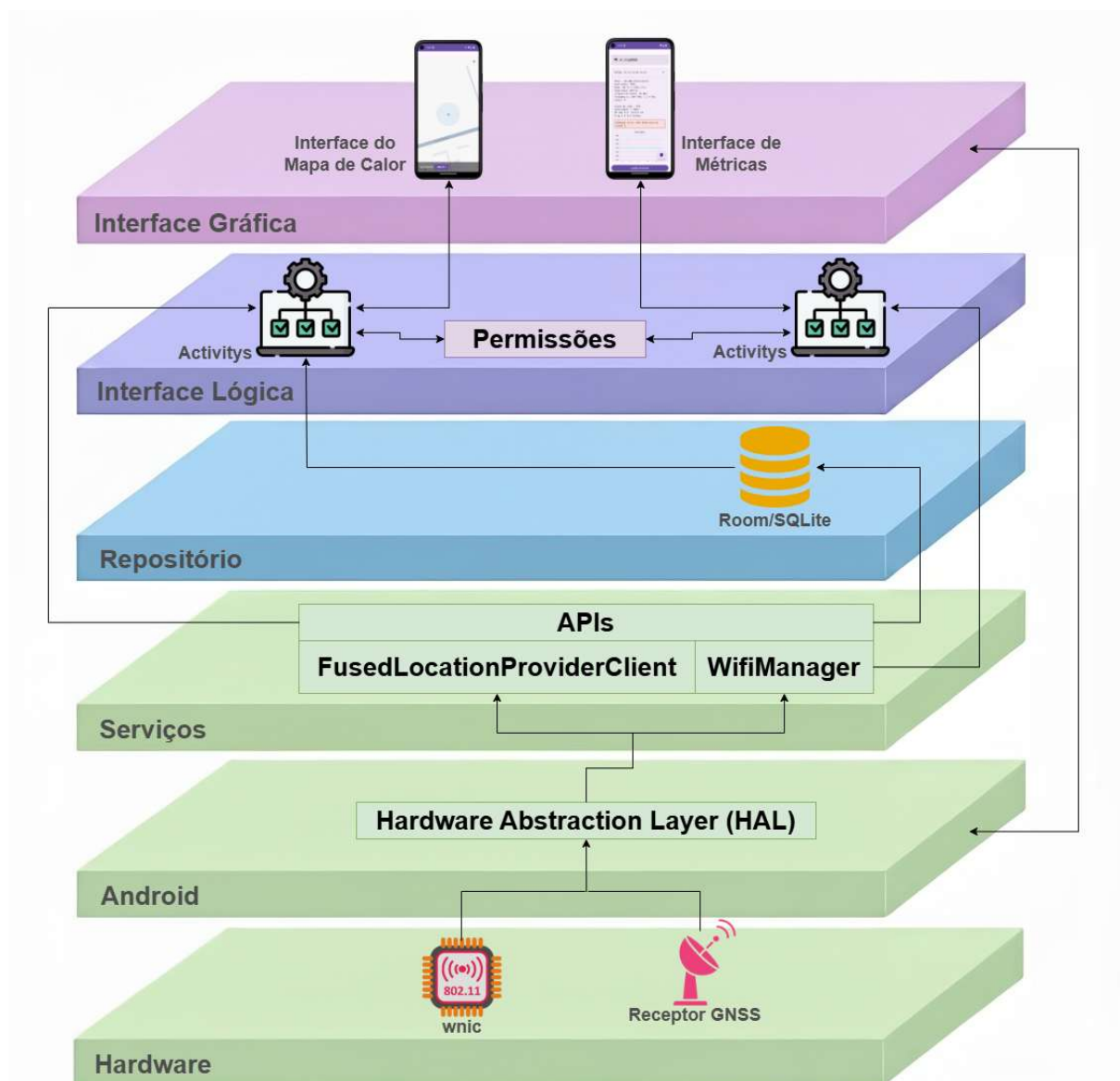
### 3. RESULTADOS E DISCUSSÃO

A fim de facilitar o diagnóstico da má qualidade de sinal para a população brasileira como um todo, foi desenvolvida uma aplicação que, ao utilizar recursos presentes em qualquer celular android, consegue exibir dados, gráficos e mapas de calor que podem ser facilmente analisados e transformados em informações precisas para detectar pontos de sombra e atenuação do sinal em ambientes residenciais, agropecuários ou industriais.

O sistema foi construído baseado nos princípios de separação de responsabilidade, conforme diagramado na Figura 9. As classes de serviço realizam requisições para o HAL, que retorna os dados às classes lógicas para processamento e, posteriormente, para a interface gráfica. Esta utiliza as informações tanto para gerar medições e gráficos quanto para plotá-las no mapa e produzir o mapa de calor. Paralelamente, os dados são armazenados no banco de dados Room para consultas futuras. Essa arquitetura modular possibilita que a aplicação execute duas funcionalidades distintas: um diagnóstico instantâneo das métricas e um mapeamento contínuo, de forma organizada e reaproveitando componentes essenciais. Dessa forma, é possível estruturar a aplicação em três camadas principais.

Para o desenvolvimento da aplicação, utilizou-se a IDE Android Studio. A linguagem de programação adotada foi o Java, selecionada devido à sua robustez e ampla consolidação internacional.

**Figura 9.** Arquitetura da aplicação



### 3.4 Interfaces

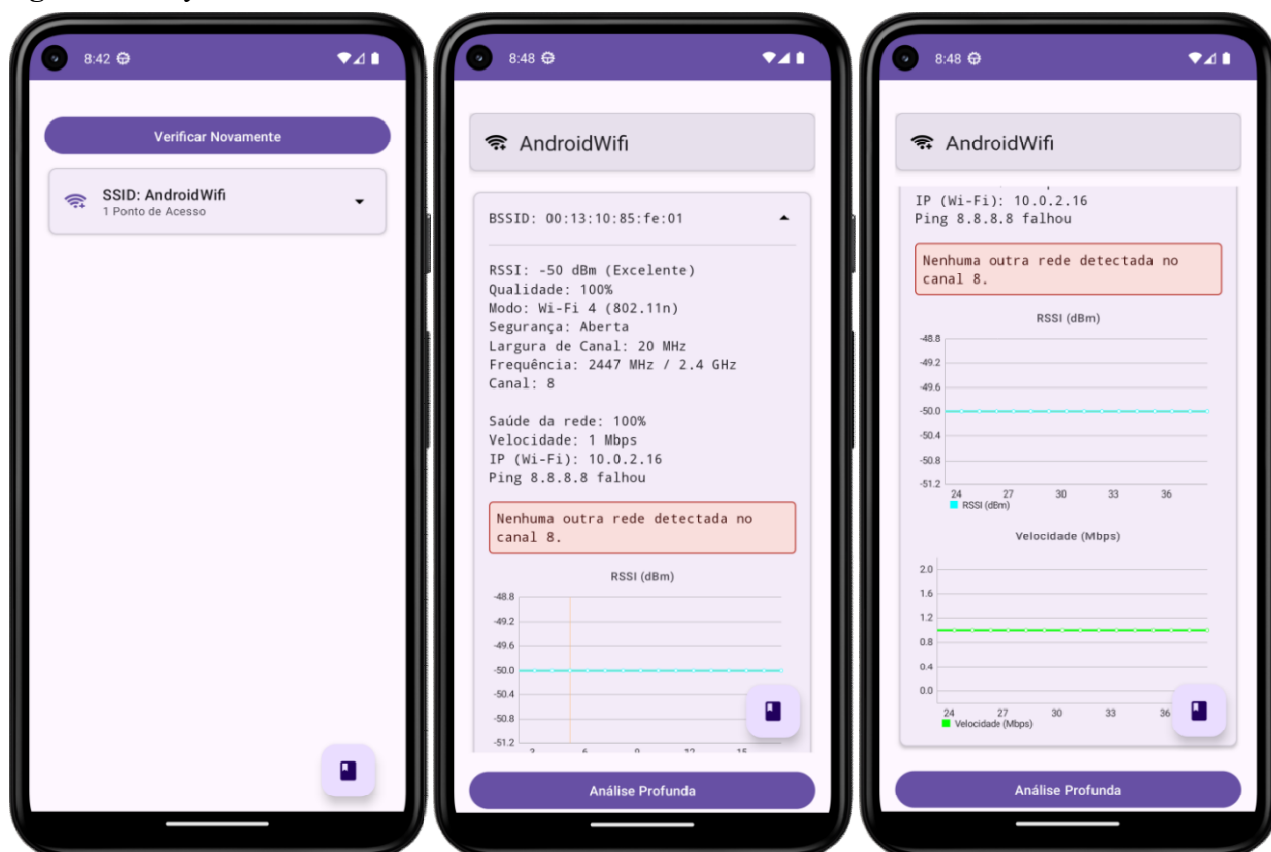
O aplicativo possui 2 interfaces principais, Métricas e Mapa de Calor, sendo imprescindivelmente necessário passar pela interface de métricas em sua inicialização, pois neste momento será avaliado qual SSID e BSSID estão próximos do aparelho celular e as informações que eles recebem provenientes dos pontos de acesso, sendo liberado a interface do mapa de calor apenas após esta validação.

#### 3.4.1 Interface de Métricas

A interface de Métricas oferece uma análise detalhada do ambiente de radiofrequência local. Ela permite ao usuário inspecionar redes vizinhas (SSIDs e BSSIDs) e avaliar profundamente a conexão atual. São exibidos parâmetros técnicos essenciais, como a força do sinal (RSSI), o modo Wi-Fi, o tipo de segurança, a frequência, o canal e sua largura. Além disso, a interface fornece métricas de desempenho, como a velocidade da rede (Mbps), e informações de diagnóstico de conectividade, incluindo o IP do dispositivo e um teste de verificação de acesso ao DNS do Google (8.8.8.8).

Através de fórmulas a aplicação calcula a Qualidade e Saúde da rede, e automaticamente analisa possíveis colisões de canais, é possível verificar de maneira dinâmica e gráfica a velocidade da rede e o RSSI. Isto tudo está organizado em lista, como exibido na Figura 10, para facilitar o uso do sistema.

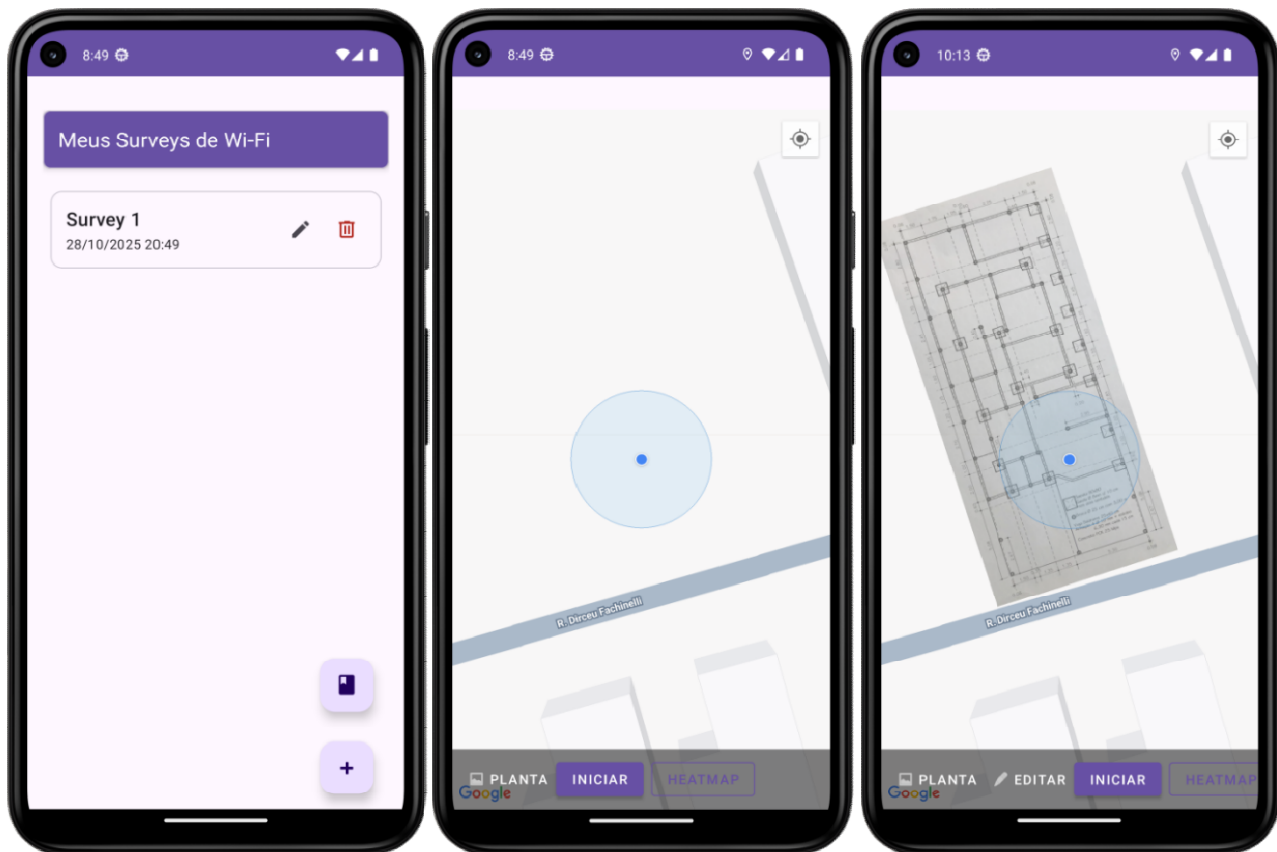
**Figura 10.** Layout interface de métricas



#### 3.4.2 Interface de Mapa de Calor

A interface de Mapa de Calor, a funcionalidade central deste trabalho, é o módulo de mapeamento, ilustrado na Figura 11. Esta tela é dominada por um componente interativo do Google Maps. A interface permite ao usuário carregar uma imagem da planta baixa, visualizar sua localização em tempo real via A-GPS e controlar a coleta de dados através dos botões "Iniciar" e "Parar". Após a coleta, o mapa de calor é renderizado e sobreposto ao mapa base e à planta baixa, fornecendo a visualização do mesmo.

**Figura 11.** Layout interface de mapa de calor



### 3.2 Camada Interface Lógica

A Camada de Interface Lógica orquestra o fluxo de dados entre a Interface Gráfica (View) e as camadas inferiores de Repositório e de Serviços. Esta camada é implementada seguindo o padrão de arquitetura MVVM (Model-View-ViewModel), que desacopla a lógica de negócio da interface do usuário, resultando em um código mais modular, testável e de fácil manutenção.

#### 3.2.1 Lógica do Mapa de Calor

As classes centrais são a SurveyActivity e o SurveyViewModel. A SurveyActivity desempenha o papel de controlador da Interface de Usuário (View), assumindo a responsabilidade por inflar o layout, gerenciar o ciclo de vida da atividade e renderizar o mapa do Google Maps. Além disso, ela captura as interações diretas do usuário.

Em contrapartida, o SurveyViewModel constitui o núcleo lógico da tela. Este componente é encarregado de manter o estado da aplicação, possuindo a capacidade de sobreviver a mudanças de configuração, como a rotação do dispositivo. É o SurveyViewModel que orquestra as operações de segunda plano, comunicando-se diretamente com o LocationProvider para a obtenção de coordenadas e com o WifiService para a leitura do RSSI, assegurando, assim, que a SurveyActivity permaneça estritamente focada na camada de apresentação dos dados.

#### 3.2.2 Lógica de Métricas

A MetricsActivity constitui o segundo pilar da Interface Lógica, atuando como a View, responsável por exibir as informações, enquanto a MetricsViewModel se encarrega de buscar e preparar os dados. A lógica de obtenção dos dados é centralizada na classe de serviço WifiService.

O fluxo de trabalho para a exibição das métricas inicia-se com a requisição de dados ao abrir a `MetricsActivity`, momento em que o `ViewModel` é instruído a carregar as informações de rede mais recentes. Para compilar um relatório de diagnóstico completo, o `ViewModel` invoca uma série de métodos específicos do `WifiService`. A chamada primária é feita através de `wifiService.getCurrentConnection()`, utilizada para obter um objeto `WifiInfo` que agrega dados essenciais das camadas física e de enlace, tais como a intensidade do sinal (RSSI), a velocidade do link, o SSID e o BSSID do ponto de acesso.

Por fim, a etapa de apresentação na interface gráfica consolida o padrão de arquitetura reativa. O `ViewModel` expõe as informações processadas através de objetos `LiveData`, os quais são observados continuamente pela `MetricsActivity`. Ao detectar qualquer alteração nos dados emitidos, a `Activity` atualiza os gráficos na tela.

### 3.2.3 Gerenciamento de Permissões

A funcionalidade central do aplicativo depende do acesso a dados sensíveis do dispositivo: a localização precisa e as informações da rede Wi-Fi. O sistema operacional Android, por questões de privacidade e segurança, exige que o usuário conceda permissão explícita em tempo de execução para esses acessos. As permissões necessárias são declaradas no `AndroidManifest.xml`, seguindo as diretrizes da arquitetura Android.

As permissões `ACCESS_FINE_LOCATION`, `ACCESS_WIFI_STATE`, `ACCESS_NETWORK_STATE` são necessárias para que o `FusedLocationProviderClient` acesse os dados do receptor GNSS, e que o `WifiManager`, em versões mais recentes do Android, possa retornar informações detalhadas como o RSSI da conexão atual.

O gerenciamento dessas permissões é tratado na `SurveyActivity` através de uma abordagem moderna utilizando `ActivityResultLauncher`. O método `checkLocationPermission()` verifica se a permissão já foi concedida. Caso contrário, ele lança o `requestPermissionLauncher`, que exibe a caixa de diálogo padrão do Android para o usuário.

### 3.2.4 Plotagem dos Dados

Os dados brutos coletados pelas APIs não possuem valor informativo direto até serem processados, correlacionados e organizados visualmente. O processo se inicia na coleta, onde a cada atualização de localização recebida do `FusedLocationProviderClient`, a camada lógica da interface gráfica executa duas ações imediatas: captura as coordenadas (latitude e longitude) do telemóvel e invoca o `WifiService` para obter o RSSI da rede conectada naquele exato instante.

Esses três dados, latitude, longitude e RSSI, são então encapsulados em uma entidade e enviados ao banco de dados Room/SQLite utilizando persistência assíncrona para não bloquear a execução do programa.

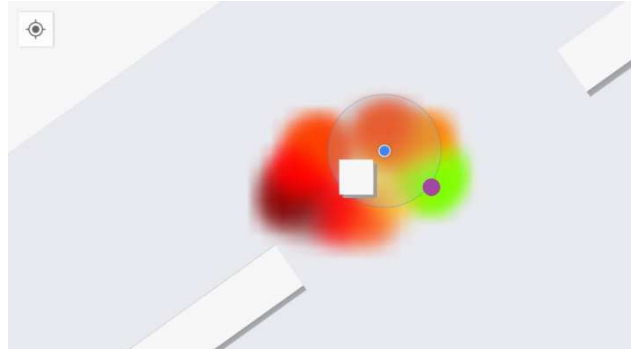
Após a finalização da coleta de dados, o processo de renderização visual é iniciado. A primeira etapa consiste em consultar a base de dados Room para recuperar a lista completa de todos os pontos de medição associados ao survey atual. A aplicação executa um algoritmo de interpolação espacial. Este método calcula os limites geográficos da coleta e cria uma grade de alta resolução. O valor de cada célula nesta grade é então determinado como uma média ponderada do RSSI dos pontos de dados reais mais próximos, conferindo maior peso aos mesmos.

Com a grade interpolada, é invocada uma função para traduzir essa matriz de valores em um bitmap pixelado, onde cada valor de RSSI é mapeado para uma cor ARGB específica. Para transformar esta grade de blocos em uma superfície suave e contínua, são aplicados os conceitos de Estimção de Densidade por Kernel (KDE). Esta etapa utiliza o `ScriptIntrinsicBlur` do Android para aplicar um filtro de kernel gaussiano sobre o bitmap, criando uma estimativa de densidade sobre a influência de cada ponto, assim suavizando as transições.



Observando a Figura 12, é possível visualizar o mapa de calor gerado pela aplicação. O círculo roxo indica a posição do ponto de acesso, servindo como referência para interpretação. Nota-se que o pilar ao lado atua como obstáculo físico, atenuando o sinal. Na área oposta ao ponto de acesso, o nível de potência aproximou-se de  $-80$  dBm, caracterizando baixa intensidade, enquanto nas regiões mais próximas da fonte os valores ficaram em torno de  $-30$  dBm, evidenciando forte recepção.

**Figura 12.** Plotagem mapa de calor



### 3.3 Camada de Repositório

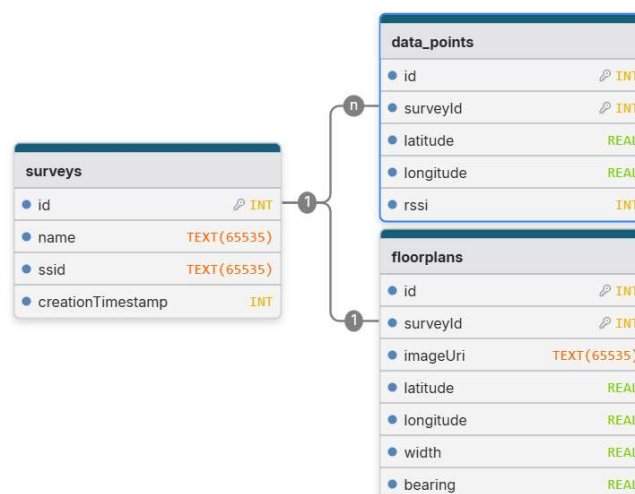
A Camada de Repositório, posicionada entre a Camada Lógica e a fonte de dados persistente, é a espinha dorsal da estratégia de gerenciamento de dados da aplicação. Ela implementa o padrão de projeto Repository, que atua como um mediador entre o domínio da aplicação e as fontes de dados. Conforme o diagrama de arquitetura, seu papel principal é abstrair o acesso ao banco de dados Room/SQLite, fornecendo uma interface de dados limpa e coesa para a Camada Lógica.

Ao fazer isso, o repositório estabelece o princípio da "Fonte Única de Verdade" (Single Source of Truth) para todos os dados persistidos. Isso significa que a Camada Lógica não precisa se preocupar com os detalhes de como os dados são armazenados, consultados ou atualizados, ela apenas se comunica com o repositório. A Figura 13 representa as tabelas e os relacionamentos utilizados para armazenar nossos dados.

A persistência de dados no projeto é gerenciada pela biblioteca Room, que é a camada de abstração recomendada pelo Google sobre o SQLite. O Room simplifica o trabalho com o banco de dados, fornecendo verificação de consultas em tempo de compilação e convertendo objetos Java em registros do banco de dados com o mínimo de código boilerplate.

A implementação da Camada de Repositório é materializada na classe SurveyRepository. Esta classe não interage diretamente com o SQLite, mas sim com os DAOs (Data Access Objects) definidos pela biblioteca Room. Os DAOs são interfaces que mapeiam chamadas de método para consultas SQL.

**Figura 13.** Diagrama de relacionamentos das tabelas do aplicativo



### 3.4 Camada de Serviços

A Camada de Serviços atua como uma interface direta para as APIs do sistema operacional e os serviços Google. Esta camada é a de mais baixo nível dentro da lógica da aplicação, sendo sua principal responsabilidade isolar o restante do código da complexidade de interagir com o hardware do dispositivo, como o módulo Wi-Fi (WNIC) e o receptor GNSS.

Toda interação com o hardware é mediada pelo sistema através da Camada de Abstração de Hardware (HAL). As duas APIs consumidas por esta camada são o `FusedLocationProviderClient` e a `WifiManager`.

#### 3.4.1 Abstração do Posicionamento Geográfico

A implementação desta funcionalidade está encapsulada na classe `LocationProvider`. Esta classe serve como um wrapper e simplificador para a complexa API de localização.

A inicialização do componente ocorre no construtor da classe, onde uma instância do `FusedLocationProviderClient` é obtida através de `LocationServices.getFusedLocationProviderClient()`. O comportamento dessa coleta é, então, finamente ajustado por meio de um objeto `LocationRequest`. No código, utiliza-se o método `new LocationRequest.Builder()` para definir parâmetros críticos ao levantamento, como o `Priority.PRIORITY_HIGH_ACCURACY`, que instrui o sistema a priorizar o GNSS para garantir a precisão necessária ao mapeamento de calor. Adicionalmente, são configurados filtros de otimização como o `setMinUpdateIntervalMillis()`, que evita a sobrecarga do sistema, e o `setMinUpdateDistanceMeters()`, um parâmetro crucial que restringe a entrega de novas coordenadas apenas quando o dispositivo se desloca uma distância mínima, eliminando redundâncias na coleta.

A operação principal reside na chamada `fusedLocationClient.requestLocationUpdates()`. Este método registra um `LocationCallback` junto ao sistema operacional, que passa a ser invocado de forma assíncrona sempre que uma nova localização satisfaz os critérios definidos no `LocationRequest`. Esse modelo garante que a aplicação não precise solicitar ativamente a posição, mas sim reagir a eventos do sistema.

#### 3.4.2 Interface com o Hardware Wi-Fi

A classe `WifiService` atua como uma fachada (Facade Pattern), provendo um conjunto de métodos diretos e coesos que ocultam os detalhes de implementação do `WifiManager`.

A inicialização da classe `WifiService` ocorre mediante a obtenção de uma instância do `WifiManager`, acessada através do serviço de sistema `context.getSystemService(Context.WIFI_SERVICE)`. Dentre as funcionalidades implementadas, destaca-se o método `getCurrentRssi()`, considerado crítico para a geração do mapa de calor. Internamente, este método executa a chamada síncrona `wifiManager.getConnectionInfo().getRssi()`, que acessa a Camada de Abstração de Hardware (HAL) para consultar diretamente o driver da interface WNIC. O retorno é a intensidade do sinal recebido (RSSI) em dBm (decibel-miliwatt) da rede conectada, sendo este procedimento invocado a cada nova coordenada geográfica registrada durante o levantamento.

Além da medição pura de sinal, o serviço fornece suporte essencial à interface de diagnósticos de rede por meio de métodos complementares. O `getCurrentConnection()`, utilizado primariamente pela interface de Métricas, retorna o objeto `WifiInfo` completo, um agregado de dados que inclui não apenas o RSSI, mas também a velocidade do link, o SSID, o BSSID o endereço IP do dispositivo e o endereço do gateway.

### 3.5 Camada Android

A Camada de Abstração de Hardware (HAL) representa o elo de conexão entre o código do sistema operacional e as especificidades proprietárias dos fabricantes, garantindo assim a portabilidade da aplicação. O Android, através do AOSP (Android Open Source Project), define apenas as interfaces padrão, conhecidas como HIDL ou AIDL, estabelecendo um contrato de software que determina quais funções de Wi-Fi e GPS devem ser expostas.

Cabe a cada fabricante de dispositivo ou fornecedor de chipset desenvolver a implementação concreta dessa HAL para seu respectivo hardware. É dentro dessas bibliotecas compartilhadas, construídas pelos fabricantes, que residem os comandos específicos para acionar os drivers proprietários do kernel Linux.

Essa arquitetura é fundamental, pois isola a lógica do aplicativo das variações de implementação de hardware. Embora a sensibilidade física da antena varie, a camada HAL assegura que a entrega dos objetos de dados para o framework Java siga estritamente o protocolo definido pelo Google. Isso permite que a aplicação execute as chamadas de API sem necessitar de adaptações de código para diferentes modelos de celular, confiando que a fabricante implementou corretamente as diretrizes da interface de varredura e localização do Android.

### **3.5.1 APIs Android**

A interação da camada lógica com os recursos do sistema ocorre por meio de APIs nativas, escolha que impactou diretamente a performance e a precisão dos dados visualizados.

Para a análise de redes, a utilização da classe WifiManager do android de forma nativa permitiu acesso direto aos objetos ScanResult. O funcionamento dessa API é assíncrono e sujeito às políticas de economia de energia do Android, uma limitação conhecida como Wi-Fi Scan Throttling. Isso significa que a atualização do mapa de calor não é contínua em tempo real, mas sim discreta, baseada na frequência de varreduras permitidas pelo sistema operacional. O aplicativo foi desenvolvido respeitando esses intervalos para garantir que cada ponto plotado no mapa representasse uma leitura de espectro válida e atualizada, evitando a repetição de dados em cache.

Quanto ao posicionamento, optou-se pelo FusedLocationProviderClient do Google Play Services, em detrimento do antigo LocationManager. Esta escolha técnica é a responsável pela viabilidade do uso da ferramenta em ambientes indoor. Enquanto o LocationManager tradicional depende quase exclusivamente da visibilidade direta de satélites, a API FusedLocationProviderClient implementa algoritmos de fusão de sensores. Ela combina os dados brutos do GNSS com sinais de radiofrequência (triangulação de antenas celulares e pontos Wi-Fi conhecidos) e sensores inerciais, como o acelerômetro e o giroscópio. O resultado é uma convergência de posição mais rápida e uma estabilidade maior do marcador no mapa, reduzindo o jitter da localização durante o deslocamento para a coleta das métricas, permitindo que o aplicativo registre coordenadas de latitude e longitude válidas mesmo em zonas de sombra de GPS.

### **3.6 Camada Hardware**

A aplicação desenvolvida não recorre a simulações estocásticas ou modelos preditivos teóricos para inferir a cobertura de sinal. Para que os resultados apresentados nas seções anteriores fossem gerados, houve uma interação contínua entre a camada de software e os componentes físicos do dispositivo móvel.

Todos os dados brutos de radiofrequência originam-se na Wireless Network Interface Controller (WNIC). A WNIC atua não apenas como interface de comunicação, mas também como um sensor de espectro. Ao receber a solicitação de scan via WifiManager, na camada de Serviços, o hardware alterna entre os canais das frequências de 2.4GHz e 5GHz definidos pelo padrão IEEE 802.11. Durante esse processo de varredura, a WNIC captura os Pacotes de Sinalização emitidos pelos Access Points. O valor de RSSI exibido no aplicativo é uma representação digital da energia de radiofrequência medida pelo circuito de rádio no preâmbulo do pacote recebido. É importante notar que a conversão dessa energia física para uma unidade inteira (por exemplo, -65 dBm) ocorre no firmware do chipset e é exposta ao Android via HAL, garantindo que o mapa de calor reflita a sensibilidade real da antena do dispositivo utilizado no levantamento.

Paralelamente, a captura de geolocalização envolve o uso de um receptor Global Navigation Satellite System (GNSS). Para a plotagem dos pontos no mapa, o receptor realiza cálculos de trilateração baseados no tempo de voo dos sinais de satélites. No entanto, ambientes indoor impõem barreiras físicas que atenuam o sinal de satélite.

É reconhecido que limitações inerentes ao hardware, como variações no design da antena, na sensibilidade do chipset (Rsens) entre diferentes fabricantes ou a degradação da precisão do GNSS em ambientes fechados, são fatores presentes que influenciam a coleta. Contudo, tais variações validam o propósito da ferramenta: aferir a experiência real do dispositivo cliente naquele ambiente específico, ao invés de projetar um cenário idealizado.

#### **4. CONCLUSÕES**

A problemática investigada foi desenvolvida com a aplicação de técnicas de mapeamento da intensidade de sinais Wi-Fi, integrando dados de geolocalização para gerar representações visuais que solucionassem a dificuldade do diagnóstico empírico de cobertura, através de um método proposto que correlacionasse valores de RSSI com coordenadas geográficas para produzir mapas de calor, substituindo a percepção subjetiva do usuário por uma análise técnica.

A aplicação demonstrou ser capaz de renderizar mapas de calor que identificaram com sucesso as zonas de sombra e a atenuação causada por obstáculos físicos, distinguindo áreas de recepção forte de áreas fracas. Resultando em uma ferramenta funcional que transforma a análise subjetiva da qualidade do sinal em um diagnóstico técnico, visual e georreferenciado. Também oferece também uma análise micro sobre o ambiente de radiofrequência, a qual permite ao usuário inspecionar o cenário local, exibindo redes vizinhas (SSIDs e BSSIDs), enquanto aprofunda a avaliação da conexão ativa através de parâmetros técnicos essenciais.

Para futuras versões do projeto, vislumbra-se um conjunto de melhorias focadas em aumentar a precisão tanto da geolocalização quanto da coleta de dados e do mapeamento ambiental. Considera-se a substituição da biblioteca de mapeamento, visto que o Google Maps se mostrou impreciso em certos pontos, uma alternativa viável seria a migração para a biblioteca osmdroid, uma solução open-source que utiliza o OpenStreetMap. Embora em testes iniciais tenha ofertado menos recursos, garante maior precisão e possibilita contribuições para sua melhoria. Além disso, para um enriquecimento significativo dos resultados, a incorporação de um sensor LiDAR, potencialmente integrada a bibliotecas de SLAM como o Google Cartographer, possibilitaria o mapeamento ao vivo da estrutura interna do ambiente, adicionando um grande nível de detalhe ao mapa e permitindo uma compreensão muito mais precisa de como a distribuição do sinal Wi-Fi é afetada por obstáculos físicos. Em paralelo, para a extração de dados de rede, avalia-se que, apesar da capacidade da arquitetura Android, ferramentas dedicadas como o Kali Linux ARM integrado a uma placa Raspberry Pi com módulos de antena devidamente configurados, entregaria uma medição precisa a nível de Jitter, conseguindo diagnosticar problemas de rede avançados e mantendo a mesma portabilidade que o aparelho celular no proporciona.

## REFERÊNCIAS BIBLIOGRÁFICAS

AGÊNCIA NACIONAL DE TELECOMUNICAÇÕES. Resolução nº 715, de 23 de outubro de 2019. Resolução nº 715, de 23 de outubro de 2019 Aprova o Regulamento de Avaliação da Conformidade e de Homologação de Produtos para Telecomunicações. Resolução nº 715, de 23 de outubro de 2019, Brasil, 24 out. 2019. Disponível em: <https://informacoes.anatel.gov.br/legislacao/resolucoes/2019/1350-resolucao-715>. Acesso em: 1 set. 2025.

AL-RAYES, Hadeel Tariq. Studying Main Differences between Android & Linux Operating Systems. International Journal of Electrical & Computer Sciences, [S. l.], v. 12, n. 05, p. 46-49, out. 2012. Disponível em: [https://www.researchgate.net/profile/Hadeel-Al-Rayes/publication/328125274\\_Studying\\_Main\\_Differences\\_between\\_Android\\_Linux\\_Operating\\_Systems/links/5c7424c8299bf1268d25999b/Studying-Main-Differences-between-Android-Linux-Operating-](https://www.researchgate.net/profile/Hadeel-Al-Rayes/publication/328125274_Studying_Main_Differences_between_Android_Linux_Operating_Systems/links/5c7424c8299bf1268d25999b/Studying-Main-Differences-between-Android-Linux-Operating-)

Systems.pdf?origin=publication\_detail&\_tp=eyJjb250ZXh0Ijp7ImZpcnN0UGFnZSI6InB1YmxpY2F0aW9uIiwicGFnZSI6InB1YmxpY2F0aW9uRG93bmxvYWQiLCJwcmV2aW91c1BhZ2UiOiJwZWJsaWNhdGlvbiJ9fQ&\_\_cf\_chl\_tk=ltpc88iNVj0PV8WFCFmqllI0pefmRLB2RDnVw4zjqNLI-1759286758-1.0.1.1-aOdq6WwRy9BTfdded8m8zGiDwIJTqdBLsDfinNHPO\_0. Acesso em: 23 set. 2025.

BORLAND, David; II, Russell M. Taylor. Rainbow Color Map (Still) Considered Harmful. IEEE Computer Graphics and Applications, [s. l.], v. 27, ed. 2, p. 14-17, maio-abril 2007. DOI 10.1109/MCG.2007.323435. Disponível em: <https://ieeexplore.ieee.org/abstract/document/4118486>. Acesso em: 27 out. 2025.

CÂMARA, Gilberto; DAVIS, Clodoveu. MONTEIRO, Antônio Miguel Vieira Introdução à ciência da geoinformação. Brasil: [s. n.], 2001. 345 p. Disponível em: <http://mtc-m12.sid.inpe.br/col/sid.inpe.br/sergio/2004/04.22.07.43/doc/publicacao.pdf>. Acesso em: 26 set. 2025.

CISCO. Protocolos básicos de rede na compreensão do modelo OSI. 2023. Disponível em: <https://community.cisco.com/t5/artigos-gerais/protocolos-básicos-de-rede-na-compreensão-do-modelo-osi/ta-p/4802414>. Acesso em: 26 sep. 2025.

COLEMAN, David. 2.4 GHz Channel Planning. 2012. Disponível em: <https://www.extremenetworks.com/resources/blogs/2-4-ghz-channel-planning>. Acesso em: 20 sep. 2025.

CZAPLICKI, Evan. Elm Colormaps. Elm Packages, 2018. Disponível em: <https://package.elm-lang.org/packages/2mol/elm-colormaps/latest>. Acesso em: 28 out. 2025.

DIGGELEN, Frank Van. A-GPS: Assisted GPS, GNSS, and SBAS. Boston: [s. n.], 2009. 380 p. ISBN 978-1-59693-374-3. Disponível em: [https://www.academia.edu/40813112/A\\_GPS\\_Assisted\\_GPS\\_GNSS\\_and\\_SBAS\\_By\\_Frank\\_van\\_Diggelen](https://www.academia.edu/40813112/A_GPS_Assisted_GPS_GNSS_and_SBAS_By_Frank_van_Diggelen). Acesso em: 1 out. 2025.

GOOGLE. Visão geral da camada de abstração de hardware (HAL). In: GOOGLE. AOSP. [S. l.], 27 jul. 2025a. Disponível em: <https://source.android.com/docs/core/architecture/hal?hl=pt-br>. Acesso em: 23 set. 2025.

GOOGLE. Visão geral da arquitetura. In: GOOGLE. AOSP. [S. l.], 2 out. 2025b. Disponível em: <https://source.android.com/docs/core/architecture?hl=pt-br>. Acesso em: 24 jul. 2025.

GOOGLE. Wi-Fi. In: GOOGLE. AOSP. [S. l.], 27 jul. 2025c. Disponível em: <https://source.android.com/docs/core/ota/modular-system/wifi?hl=pt-br>. Acesso em: 23 set. 2025.

GOOGLE. WifiManager. In: GOOGLE. Android Developers. [S. l.], 20 ago. 2025d. Disponível em: [https://developer.android.com/reference/android/net/wifi/WifiManager#ACTION\\_PICK\\_WIFI\\_NETWORK](https://developer.android.com/reference/android/net/wifi/WifiManager#ACTION_PICK_WIFI_NETWORK). Acesso em: 23 set. 2025.

GOV.BR (Brasil). IBGE; BELLO, Luiz. No Brasil, 88,9% da população de 10 anos ou mais tinha celular em 2024. Brasil: Luiz Bello, 24 jul. 2024. Disponível em: <https://agenciadenoticias.ibge.gov.br/agencia-noticias/2012-agencia-de-noticias/noticias/44032-no-brasil-88-9-da-populacao-de-10-anos-ou-mais-tinha-celular-em-2024>. Acesso em: 10 set. 2025.

HOFMANN-WELLENHOF, Bernhard. LICHTENEGGER, Herbert. WASLE, Elmar. GNSS – Global Navigation Satellite Systems: GPS, GLONASS, Galileo, and more. Áustria: [s. n.], 2008. 516 p. ISBN 978-3-211-73012-6. Disponível em: <https://sciarium.com/file/78994>. Acesso em: 25 set. 2025.

IEEE STANDARDS ASSOCIATION. The Evolution of Wi-Fi Technology and Standards. [S. l.], 16 maio 2023. Disponível em: <https://standards.ieee.org/beyond-standards/the-evolution-of-wi-fi-technology-and-standards>. Acesso em: 15 ago. 2025.

MORAES, Filipe; SILVA, Thiago; CEOLIN, Simone Regina; AZEVEDO, Renato Preigschadt. Análise e Otimização de Rede Wi-Fi num Ambiente Corporativo. RETEC - Revista de Tecnologias, Ourinhos-SP, ano 2024, v. 17, n. 2, p. 53-71, 10 mar. 2025. Disponível em: <https://retec.fatecourinhos.edu.br/index.php/retec/issue/view/42>. Acesso em: 11 set. 2025.

NUÑEZ, Jamie R.; ANDERTON, Christopher R.; RENSLOW, Ryan S. Optimizing Colormaps with Consideration of Color Vision Deficiency to Enable Accurate Interpretation of Scientific Data. PLOS One, [s. l.], v. 13, n. 7, 1 ago. 2018. DOI <https://doi.org/10.1371/journal.pone.0199239>. Disponível em: <https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0199239>. Acesso em: 10 dez. 2025.

PARZEN, Emanuel. On Estimation of a Probability Density Function and Mode. Project Euclid, [S. l.], v. 33, p. 1065-1076, 1 set. 1962. DOI 10.1214/aoms/1177704472. Disponível em: <https://projecteuclid.org/journals/annals-of-mathematical-statistics/volume-33/issue-3/On-Estimation-of-a-Probability-Density-Function-and-Mode/10.1214/aoms/1177704472.full?tab=ArticleFirstPage>. Acesso em: 16 set. 2025.

PERES, André. Mecanismo de Autenticação Baseado na Localização de Estações Sem Fios Padrão IEEE 802.11. Orientador: Raul Fernando Weber. 2010. 87 p. Tese (Doutorado em Ciência da Computação) - Universidade Federal do Rio Grande do Sul Instituto de Informática, Porto Alegre, 2010.

SCOTT, David. Multivariate Density Estimation: Theory, Practice, and Visualization. 2. ed. [S. l.]: Wiley, 2015. 305 p. ISBN 928-0-471-69755-8.

SILVERMAN, Bernard. Density Estimation for Statistics and Data Analysis. In: MONOGRAPHS on Statistics and Applied Probability. London: [s. n.], 1986. Disponível em: <https://ned.ipac.caltech.edu/level5/March02/Silverman/paper.pdf>. Acesso em: 30 set. 2025.

TANENBAUM, Andrew. BOS, Herbert. Sistemas operacionais modernos. 4. ed. Brasil: Pearson, 2016. 758 p. ISBN 978-85-4301-818-8. Disponível em: <https://www.kufunda.net/publicdocs/Sistemas%20Operacionais%20Modernos%20%28Andrew%20S.%20Tanenbaum,%20Herbert%20Bos%29.pdf>. Acesso em: 23 set. 2025.

TANENBAUM, Andrew. FEAMSTER, Nick. WETHERALL, David. Redes de computadores. 6. ed. Brasil: Bookman, 2021. 568 p. ISBN 9780135408001. Disponível em: <https://dn721905.ca.archive.org/0/items/tanenbaum-rede-de-computadores-6a/Tanenbaum%20Rede%20de%20Computadores%206a.pdf>. Acesso em: 30 set. 2025.

TASTL, Ingeborg; BHACHECH, Miheer; MORONEY, Nathan; HOLM, Jack. ICC Color Management and CIECAM02. Color and Imaging Conference, [s. l.], v. 13, ed. 1, p. 217-223, janeiro 2005. DOI 10.2352/CIC.2005.13.1.art00040. Disponível em: <https://library.imaging.org/cic/articles/13/1/art00040>. Acesso em: 23 out. 2025.

TUFTE, Edward. The visual display of quantitative information. 2. ed. Connecticut: Graphic Press, 2007. 197 p. Disponível em: <https://kyl.neocities.org/books/%5BTEC%20TUF%5D%20the%20visual%20display%20of%20quantitative%20information.pdf>. Acesso em: 2 out. 2025.

TURLACH, Berwin A. Bandwidth Selection in Kernel Density Estimation. In: HANDBOOK of Systemic Autoimmune Diseases. [S. l.: s. n.], 1999. Disponível em: [https://www.researchgate.net/publication/2316108\\_Bandwidth\\_Selection\\_in\\_Kernel\\_Density\\_Estimation\\_A\\_Review](https://www.researchgate.net/publication/2316108_Bandwidth_Selection_in_Kernel_Density_Estimation_A_Review). Acesso em: 28 set. 2025.



VITANOVA, Mirjana Kocaleva et al. Self-Organized Networks. 2017. Disponível em: [https://www.researchgate.net/publication/328064410\\_Self-Organized\\_Networks](https://www.researchgate.net/publication/328064410_Self-Organized_Networks). Acesso em: 1 out. 2025.

WAND, M. P.; JONES, M. C. Kernel Smoothing. 1. ed. Londres: [s. n.], 1995. Disponível em: <https://annas-archive.org/md5/112544f4c17e4eb6c717360d04c8a333>. Acesso em: 22 out. 2025.

WARE, Colin; STONE, Maureen; SZAFIR, Danielle Albers. Rainbow Colormaps Are Not All Bad. IEEE Computer Graphics and Applications, [s. l.], v. 43, ed. 3, p. 88-93, maio-junho 2023. DOI 10.1109/MCG.2023.3246111. Disponível em:

<https://ieeexplore.ieee.org/abstract/document/10128890>. Acesso em: 29 out. 2025.

WI-FI ALLIANCE (EUA). Wi-Fi® delivers the future of connectivity: Wi-Fi evolves, maintains core strengths for next generation use cases. Estados Unidos, Texas, Austin: Andrea Torres, 18 dez. 2018. Disponível em: <https://www.wi-fi.org/news-events/newsroom/wi-fi-delivers-the-future-of-connectivity>. Acesso em: 3 set. 2025.