



**INSTITUTO FEDERAL DO PIAUÍ – IFPI
PRÓ-REITORIA DE ENSINO
DIRETORIA DE ENSINO
CURSO SUPERIOR DE TECNOLOGIA EM ANÁLISE E DESENVOLVIMENTO DE
SISTEMAS**

RAFHAEL GASPAR DA SILVA

**LEMBREMÉDI: UMA SOLUÇÃO TECNOLÓGICA PARA O GERENCIAMENTO DE
MEDICAMENTOS**

**PICOS, PI
2026**

RAFHAEL GASPAR DA SILVA

LEMBREMÉDI: UMA SOLUÇÃO TECNOLÓGICA PARA O GERENCIAMENTO DE MEDICAMENTOS

Relatório Técnico de Software apresentado como exigência parcial para obtenção do diploma do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí, Campus Picos.

Orientador: Prof. Me. João Paulo Lima do Nascimento.

RAFHAEL GASPAR DA SILVA

LEMBREMÉDI: UMA SOLUÇÃO TECNOLÓGICA PARA O GERENCIAMENTO DE MEDICAMENTOS

Relatório Técnico de Software apresentado como exigência parcial para obtenção do diploma do Curso de Tecnologia em Análise e Desenvolvimento de Sistemas do Instituto Federal de Educação, Ciência e Tecnologia do Piauí, Campus Picos.

Aprovada em: 05/03/2026.

BANCA EXAMINADORA

Prof. Me. João Paulo Lima do Nascimento (Orientador)
Instituto Federal do Piauí (IFPI)

Prof. Me. Jader Anderson Oliveira de Abreu
Instituto Federal do Piauí (IFPI)

Prof. Dr. Rogerio Figueredo de Sousa
Instituto Federal do Piauí (IFPI)

AGRADECIMENTOS

Primeiramente, agradeço a Deus, por me conceder saúde, perseverança e sabedoria para superar os desafios ao longo desta jornada acadêmica.

Aos meus pais, Eronildo e Ivani, minha mais profunda e sincera gratidão. Sou grato por todo amor, dedicação e pelos ensinamentos que moldaram meu caráter. Agradeço por sempre acreditarem em mim, e por me apoiarem da melhor maneira que puderam. Esta conquista é tão minha quanto de vocês.

Às minhas irmãs, expresso minha gratidão pela presença constante e pelo apoio em todos os momentos da minha vida. Em especial, às minhas irmãs Azielly e Alliezy, que tiveram participação ativa e direta na realização deste trabalho. Agradeço imensamente pela ajuda fundamental na busca e organização dos voluntários que participaram da fase de testes, contribuindo de forma significativa para a validação do aplicativo LembreMédi.

A todos os voluntários que dedicaram seu tempo para testar o aplicativo e responder aos questionários, meu muito obrigado. A participação de vocês foi essencial para os resultados alcançados e para o aprimoramento da solução desenvolvida.

Aos professores do Instituto Federal do Piauí (IFPI) – Campus Picos, minha gratidão por todos os ensinamentos compartilhados ao longo do curso. Em especial, agradeço ao Prof. Me. Aislan Rafael, cujas orientações e percepções sobre o cotidiano me ensinaram que até mesmo os menores problemas do dia a dia podem se transformar em soluções tecnológicas importantes, inspiração que norteou a concepção deste projeto.

Por fim, um agradecimento especial ao Prof. Me. João Paulo Lima do Nascimento. Sou imensamente grato por sua orientação e dedicação, paciência e valiosas contribuições, não apenas na condução deste trabalho, mas também durante o projeto de pesquisa que o antecedeu. Obrigado por sempre me guiar na direção certa, incentivando-me a fazer o melhor trabalho possível e a crescer como pesquisador e profissional.

Vi veri veniversum vivus vici

“Pelo poder da verdade, eu, enquanto vivo, conquistei o universo.”

Aleister Crowley, 1904

RESUMO

O presente Relatório Técnico de Software descreve o desenvolvimento do aplicativo LembreMédi, uma solução nativa para a plataforma Android voltada ao gerenciamento de alarmes de medicamentos. O sistema tem como objetivo auxiliar pacientes e cuidadores na organização da rotina de medicação, contribuindo para a adesão aos tratamentos. A metodologia adotada compreendeu a análise de aplicativos existentes, o levantamento de requisitos funcionais e não funcionais, a modelagem do sistema por meio de diagramas UML e a definição de uma arquitetura baseada no padrão MVVM. A implementação foi realizada em Kotlin, utilizando o banco de dados Room para persistência local e o AlarmManager para o agendamento de alarmes. Para validação, o aplicativo foi submetido a testes com usuários reais, incluindo 31 participantes na avaliação de usabilidade por meio da System Usability Scale (SUS), que resultou em pontuação média de 88,31 pontos, e 12 participantes na avaliação de efetividade, na qual os usuários relataram melhora na lembrança dos horários e redução de esquecimentos.

Palavras-chave: Aplicativo móvel; Android; Kotlin; Lembrete de medicamentos.

ABSTRACT

This Software Technical Report describes the development of the LembreMédi application, a native solution for the Android platform focused on managing medication alarms. The system aims to assist patients and caregivers in organizing their medication routine, contributing to adherence to treatments. The methodology adopted included the analysis of existing applications, the gathering of functional and non-functional requirements, the modeling of the system using UML diagrams, and the definition of an architecture based on the MVVM pattern. The implementation was carried out in Kotlin, using the Room database for local persistence and AlarmManager for alarm scheduling. For validation, the application was submitted to tests with real users, including 31 participants in the usability evaluation using the System Usability Scale (SUS), which resulted in an average score of 88.31 points, and 12 participants in the effectiveness evaluation, in which users reported improved memory of schedules and reduced forgetfulness.

Keywords: Mobile application; Android; Kotlin; Medication reminder.

LISTA DE ILUSTRAÇÕES

Figura 1 – Diagrama de Casos de Uso do aplicativo LembreMédi.....	24
Figura 2 – Diagrama de Classes simplificado do sistema.....	26
Figura 3 – Arquitetura do sistema baseada em MVVM.....	28
Figura 4 – Tela Inicial (Lista de Alarmes).....	30
Figura 5 – Tela de Detalhes do Alarme.....	32
Figura 6 – Tela de Cadastro de Alarme.....	34
Figura 7 – Tela de Edição de Alarme.....	35
Figura 8 – Tela de Configurações do Aplicativo.....	36
Figura 9 – Tela de Alarme Tocando.....	38
Figura 10 – Percepção de efetividade do aplicativo.....	47
Figura 11 – Percepção de efetividade do aplicativo.....	48
Figura 12 – Confiabilidade dos alarmes.....	48

LISTA DE QUADROS E TABELAS

Quadro 1 – Requisitos funcionais.....	21
Quadro 2 – Requisitos não funcionais.....	22
Quadro 3 – Quadro Comparativa de Funcionalidades Percebidas pelo Usuário.....	40
Quadro 4 – Questionário SUS aplicado aos usuários do aplicativo LembreMédi....	42
Tabela 1 – Pontuação SUS individual dos participantes.....	43

LISTA DE ABREVIATURAS E SIGLAS

API	Application Programming Interface
DAO	Data Access Object
eHealth	Electronic Health
IDE	Integrated Development Environment
MVP	Minimum Viable Product
MVVM	Model-View-ViewModel
RF	Requisito Funcional
RNF	Requisito Não Funcional
RTS	Relatório Técnico de Software
SDK	Software Development Kit
SUS	System Usability Scale
UML	Unified Modeling Language
UI	User Interface
XML	Extensible Markup Language

SUMÁRIO

1. INTRODUÇÃO	12
1.1. OBJETIVOS	13
1.1.1. Geral	13
1.1.2. Específicos	13
2. TECNOLOGIAS ENVOLVIDAS	14
2.1. Plataforma Android	14
2.2. Linguagem de Programação – Kotlin	14
2.3. Persistência de Dados – Room Database	15
2.4. Gerenciamento de Alarmes – AlarmManager	16
2.5. Componentes Android para Execução do Alarme	17
2.6. Gerenciamento de Áudio e Vibração	18
2.7. Gerenciamento de Permissões Especiais	18
2.8. Gerenciamento de Preferências do Usuário	19
2.9. Ferramentas de Desenvolvimento	19
3. MODELAGEM DO PROJETO	20
3.1. Levantamento de Requisitos	20
3.1.1. Requisitos Funcionais (RF)	21
3.1.2. Requisitos Não Funcionais (RNF)	22
3.2. Diagramas de Casos de Uso	23
3.3. Diagramas de Classes (UML)	25
3.4. Arquitetura do Sistema	26
3.5. Interface e Navegação	28
3.5.1. Fluxo Geral de Navegação	29
3.5.2. Tela Inicial – Lista de Alarmes	29
3.5.3. Tela de Detalhes do Alarme	31
3.5.4. Tela de Cadastro de Alarme	32
3.5.5. Tela de Edição de Alarme	34
3.5.6. Tela de Configurações	36
3.5.7. Tela de Alarme Tocando	37
4. SOFTWARE	38
4.1. Implantação	38
4.2. Disponibilização do Software	39
4.3. Análise Comparativa com Aplicativos Existentes	40
4.4. Testes	41
4.4.1. Avaliação de Usabilidade – System Usability Scale (SUS)	41
4.4.2. Avaliação da Efetividade do Sistema (Pós-Uso)	45
5. CONSIDERAÇÕES FINAIS	50
REFERÊNCIAS	52

1. INTRODUÇÃO

A administração correta de medicamentos é um fator determinante para a eficácia dos tratamentos de saúde, especialmente em pacientes com condições crônicas ou que fazem uso contínuo de múltiplos fármacos. A via oral é amplamente predominante nos protocolos clínicos, sendo responsável pela maior parte das administrações medicamentosas. Um estudo realizado em um hospital universitário brasileiro identificou que 97,3% das administrações de medicamentos ocorreram por via oral, evidenciando a centralidade desse método na prática clínica (Moreira et al., 2018).

Apesar de sua ampla utilização, a eficácia do tratamento medicamentoso está diretamente relacionada à adesão adequada aos horários prescritos. A não adesão ou o uso incorreto de medicamentos pode comprometer os resultados terapêuticos, aumentar riscos à saúde e gerar custos adicionais ao sistema de saúde. Estudos apontam que até 50% dos pacientes não seguem corretamente as prescrições médicas, seja por esquecimento, confusão entre medicamentos ou falta de organização na rotina diária (Sabaté, 2003).

O horário de administração dos medicamentos é um fator crítico, pois pode influenciar diretamente sua absorção, eficácia e até a ocorrência de efeitos adversos. Conforme destacado por Vieira (2022), o momento do dia em que o medicamento é ingerido pode impactar tanto seus benefícios quanto seus potenciais prejuízos à saúde. Nesse contexto, métodos tradicionais de controle, como anotações em papel, calendários físicos ou alarmes genéricos em dispositivos móveis, frequentemente se mostram insuficientes, principalmente para usuários que gerenciam múltiplos medicamentos ao longo do dia.

Com o avanço das tecnologias digitais e a popularização dos smartphones, surgiram diversas soluções no campo da *eHealth* voltadas para o apoio à gestão da saúde. Aplicativos móveis de gerenciamento de medicamentos representam uma alternativa promissora, oferecendo lembretes personalizados, notificações em tempo real e maior organização das rotinas de medicação. Estudos indicam que tecnologias assistivas desse tipo podem contribuir significativamente para a melhoria

da adesão ao tratamento e para a redução de erros de administração (Hamine et al., 2015; Aitkten, 2013).

Entretanto, análises de aplicativos existentes, como *Medisafe*¹ e *MyTherapy*², realizadas durante a pesquisa de produtos semelhantes, apontaram limitações importantes, especialmente no que diz respeito à personalização dos alarmes e ao nível de destaque das notificações, fatores essenciais para garantir que o usuário perceba e responda ao alerta no momento correto. Essas lacunas motivaram o desenvolvimento do aplicativo LembreMédi, objeto central deste Relatório Técnico de Software.

1.1. OBJETIVOS

1.1.1. Geral

Desenvolver e documentar tecnicamente um aplicativo *Android* para gerenciamento de alarmes de medicamentos, descrevendo sua concepção, implementação, arquitetura e funcionamento, de modo a permitir a avaliação de sua efetividade no apoio à adesão ao tratamento medicamentoso.

1.1.2. Específicos

- Realizar o levantamento de requisitos e a modelagem do sistema, incluindo a documentação técnica do software por meio da definição de requisitos, elaboração de diagramas UML e descrição da arquitetura do aplicativo.
- Desenvolver o aplicativo móvel para a plataforma *Android*, utilizando tecnologias e componentes nativos do ecossistema *Android* para implementação das funcionalidades de gerenciamento de alarmes de medicamentos.
- Avaliar a usabilidade e a efetividade do aplicativo, por meio de testes com usuários reais e aplicação de instrumentos de avaliação, analisando os resultados obtidos quanto à experiência de uso e ao apoio à rotina de medicação.

¹ <https://medisafeapp.com/>

² <https://www.mytherapyapp.com/>

2. TECNOLOGIAS ENVOLVIDAS

Esta seção apresenta, de forma detalhada, as tecnologias, ferramentas e linguagens utilizados no desenvolvimento do aplicativo LembreMédi, justificando a escolha de cada um deles com base nos objetivos do projeto, nas restrições da plataforma *Android* e nas boas práticas de engenharia de *software* aplicáveis a um Relatório Técnico de Software (RTS).

O conjunto tecnológico adotado foi definido visando garantir confiabilidade na execução dos alarmes, boa experiência de uso, facilidade de manutenção do código, aderência às diretrizes do sistema *Android* e escalabilidade futura do aplicativo.

2.1. Plataforma *Android*

O aplicativo LembreMédi foi desenvolvido exclusivamente para a plataforma *Android*³, adotando como base a API de nível 35 (*Android* 15), considerando sua ampla base de usuários e a flexibilidade para utilização de recursos nativos essenciais ao funcionamento do sistema, como gerenciamento de alarmes em segundo plano, serviços em primeiro plano (*foreground services*), notificações persistentes e permissões especiais do sistema.

A escolha pela plataforma *Android* justifica-se, ainda, pela necessidade de controle detalhado sobre o ciclo de vida da aplicação, aspecto fundamental para garantir o disparo confiável de alarmes mesmo em cenários de economia de bateria, tela bloqueada ou aplicativo encerrado pelo sistema operacional (Android Developers, 2025a).

2.2. Linguagem de Programação – *Kotlin*

A linguagem *Kotlin*⁴ (versão 1.9.x) foi adotada como linguagem principal de desenvolvimento do projeto, por ser oficialmente recomendada pelo *Google* para o desenvolvimento de aplicações *Android* e por oferecer recursos modernos que aumentam a segurança e a produtividade do código (Android Developers, 2025e; JETBRAINS, 2025).

³ <https://developer.android.com/about/versions/15?hl=pt-br>

⁴ <https://kotlinlang.org/docs/whatsnew19.html>

Entre os principais benefícios da utilização do *Kotlin* no contexto deste projeto, destacam-se:

- Maior segurança contra erros comuns, como *NullPointerException*;
- Sintaxe mais concisa e legível, facilitando a manutenção e evolução do código;
- Total interoperabilidade com bibliotecas Java já consolidadas no ecossistema *Android*;
- Melhor suporte a programação assíncrona por meio de *coroutines*⁵.

Essas características contribuíram diretamente para a organização da arquitetura do aplicativo e para a implementação de fluxos complexos, como o gerenciamento de alarmes e a execução de serviços em segundo plano, garantindo maior robustez e confiabilidade ao sistema.

2.3. Persistência de Dados – Room *Database*

Para o armazenamento local dos dados, foi utilizada a biblioteca Room (versão 2.6.x), integrante do conjunto de componentes *Android Jetpack*, que fornece uma camada de abstração robusta sobre o banco de dados *SQLite*. Essa biblioteca simplifica o acesso e a manipulação de dados persistentes, promovendo maior organização e segurança no gerenciamento das informações da aplicação.

O uso do Room permite:

- Mapeamento de entidades do banco de dados por meio de classes *Kotlin* anotadas, seguindo o paradigma de persistência orientada a objetos;
- Verificação de consultas SQL em tempo de compilação, reduzindo erros de execução e inconsistências de dados;
- Integração nativa com componentes arquiteturais como *LiveData* e *ViewModel*, favorecendo a reatividade da interface;
- Maior padronização e segurança no acesso aos dados, com separação clara entre camadas de persistência e lógica.

⁵ **Coroutines** são blocos de código assíncronos e "cooperativos" que podem suspender sua execução (pausar) sem bloquear a thread principal, permitindo que outros processos funcionem, e retomar de onde pararam.

<https://minacriativa.com.br/glossario/o-que-e-kotlin-coroutines-corrotinas-kotlin/>

No contexto do aplicativo LembreMédi, o Room é empregado para armazenar localmente as informações relacionadas aos alarmes de medicamentos, incluindo horário programado, nome do medicamento, dosagem, unidade de medida, estado do tratamento e demais configurações associadas. Essa abordagem garante persistência confiável dos dados mesmo em cenários de encerramento do aplicativo ou reinicialização do dispositivo, contribuindo para a continuidade e consistência do funcionamento do sistema (Android Developers, 2025h).

Como medida adicional de segurança, o armazenamento local pode ser protegido por mecanismos de criptografia disponíveis no ecossistema Android, como bibliotecas compatíveis com o *SQLite* utilizadas pelo Room. Esse mecanismo permite que todo o banco de dados seja armazenado de forma criptografada no dispositivo, impedindo o acesso direto às informações por aplicações externas ou usuários sem autorização. Como o aplicativo opera integralmente de forma *offline*, sem envio de dados para servidores externos, os riscos relacionados à transmissão de informações sensíveis pela rede são eliminados.

2.4. Gerenciamento de Alarmes – *AlarmManager*

O *AlarmManager* é o componente central do sistema responsável pelo agendamento e disparo dos alarmes do aplicativo, possibilitando a execução de eventos em horários previamente definidos, mesmo quando a aplicação não está em execução ativa. Trata-se de uma API nativa do *Android*, utilizada conforme as diretrizes das versões recentes do sistema operacional (*Android* API Level 23+), que introduziram melhorias relacionadas à precisão e ao funcionamento em modos de economia de energia.

No presente projeto, foi empregada a configuração `setExactAndAllowWhileIdle`, que permite o disparo preciso de alarmes mesmo quando o dispositivo se encontra em estado de inatividade (*Doze Mode*) ou sob restrições de economia de bateria. Essa abordagem assegura maior confiabilidade temporal na execução dos lembretes de medicação.

A utilização desse método é fundamental para o objetivo do aplicativo LembreMédi, uma vez que atrasos ou imprecisões no disparo dos alarmes podem

comprometer a adesão adequada ao tratamento medicamentoso. Assim, o *AlarmManager* contribui diretamente para a efetividade funcional do sistema, garantindo que os lembretes sejam acionados no momento correto, independentemente do estado do aplicativo ou do dispositivo (Android Developers, 2026b).

2.5. Componentes *Android* para Execução do Alarme

Para assegurar que o alarme seja exibido corretamente em diferentes estados do sistema operacional, o aplicativo utiliza uma combinação de componentes nativos do *Android*, conforme as diretrizes das versões recentes da plataforma (*Android* API Level 26+ para serviços em primeiro plano e API Level 21+ para notificações avançadas)(Android Developers, 2025a):

- ***BroadcastReceiver***: responsável por interceptar o evento disparado pelo *AlarmManager*, permitindo a reação imediata do sistema ao horário programado (Android Developers, 2025c);
- ***ForegroundService***: executado com prioridade elevada e notificação obrigatória (API 26+), garantindo a continuidade da execução do alarme mesmo sob restrições de segundo plano (Android Developers, 2025f);
- ***Activity* em tela cheia**: utilizada para exibir a interface do alarme ao usuário, inclusive sobre a tela bloqueada, conforme as permissões e *flags* de janela disponíveis nas APIs recentes do *Android* (Android Developers, 2026a);
- **Notificações persistentes**: mantêm o alarme visível enquanto ativo, assegurando *feedback* contínuo ao usuário e conformidade com as políticas de execução em primeiro plano (Android Developers, 2025g).

A adoção integrada desses componentes foi necessária para contornar as limitações progressivamente impostas pelas versões mais recentes do *Android* em relação à execução de tarefas em segundo plano e ao gerenciamento de energia, garantindo que os lembretes de medicação sejam acionados e exibidos de forma confiável independentemente do estado do aplicativo ou do dispositivo (Android Developers, 2025a; Android Developers, 2026b).

2.6. Gerenciamento de Áudio e Vibração

O gerenciamento de áudio do alarme é realizado por meio do *MediaPlayer* em conjunto com o mecanismo de *AudioFocus* do *Android* (via *AudioManager* e *AudioFocusRequest*, API Level 26+), garantindo que o som do alarme seja reproduzido com prioridade adequada, mesmo quando outros aplicativos estiverem utilizando recursos de áudio do dispositivo (Android Developers, 2026c; Android Developers, 2025b).

Além disso, o sistema permite a seleção de sons padrão do *Android* por meio do *RingtoneManager*, ampliando a flexibilidade de personalização do alarme pelo usuário (Android Developers, 2026d).

A vibração é controlada de forma opcional utilizando as APIs *Vibrator* e *VibrationEffect* (API Level 26+), sendo ativada conforme as preferências definidas nas configurações do aplicativo. Essa abordagem assegura conformidade com as diretrizes modernas de gerenciamento de energia e experiência do usuário, mantendo a efetividade dos alertas mesmo em diferentes estados do dispositivo (Android Developers, 2026f).

2.7. Gerenciamento de Permissões Especiais

O funcionamento adequado do aplicativo depende de permissões específicas do sistema *Android*, entre elas:

- Permissão para desenhar sobre outras aplicações (*SYSTEM_ALERT_WINDOW*), disponível a partir do Android 6.0 (API 23);
- Permissão para agendamento exato de alarmes (*SCHEDULE_EXACT_ALARM*), introduzida no Android 12 (API 31).

Essas permissões são gerenciadas por meio de fluxos controlados utilizando a *Activity Result* API, disponibilizada pela biblioteca *AndroidX Activity* (versão 1.2.0 ou superior), garantindo transparência ao usuário e conformidade com as diretrizes oficiais da plataforma *Android* (Android Developers, 2025a; Android Developers, 2026b).

2.8. Gerenciamento de Preferências do Usuário

As preferências do usuário, como tema visual, som do alarme e vibração, são armazenadas utilizando o mecanismo `SharedPreferences`, disponível desde as primeiras versões do *Android* (API 1) e mantido como parte do SDK oficial da plataforma (Android Developers, 2026e). No projeto, sua utilização ocorre por meio das APIs padrão do Android, garantindo compatibilidade com versões modernas do sistema operacional e persistência eficiente das configurações definidas pelo usuário (Android Developers, 2025d; Android Developers, 2026e).

Esse mecanismo permite a persistência simples e eficiente das configurações escolhidas pelo usuário, possibilitando uma experiência personalizada sem impactar significativamente o desempenho do aplicativo, uma vez que os dados são armazenados localmente em formato chave-valor, com baixo custo de leitura e escrita (Android Developers, 2025d).

2.9. Ferramentas de Desenvolvimento

Durante o desenvolvimento do projeto, foram utilizadas as seguintes ferramentas:

- **Android Studio**⁶ (versão 2024.x ou superior): Ambiente de desenvolvimento integrado (IDE) oficial para aplicações *Android*;
- **Git**⁷ (versão 2.x): Controle de versionamento do código-fonte;
- **GitHub**⁸: Plataforma de hospedagem de repositórios Git utilizada para armazenamento remoto e gerenciamento do código-fonte do aplicativo;
- **Emuladores e dispositivos físicos**: Utilizados para testes funcionais e de usabilidade, permitindo validar o comportamento do aplicativo em diferentes versões do sistema operacional e configurações de *hardware*.

Essas ferramentas foram fundamentais para garantir organização, rastreabilidade das alterações e estabilidade ao longo do processo de desenvolvimento do *software*.

⁶ <https://developer.android.com/studio?hl=pt-br>

⁷ <https://git-scm.com/docs>

⁸ <https://github.com/about?locale=pt-BR>

3. MODELAGEM DO PROJETO

A modelagem do projeto tem como objetivo representar, de forma estruturada e compreensível, o funcionamento do sistema desenvolvido, descrevendo seus requisitos, comportamentos, arquitetura e interface. Nesse sentido, a modelagem pode ser compreendida como uma abstração do sistema que permite a visualização de suas propriedades sob diferentes perspectivas, facilitando a compreensão de problemas complexos antes da construção propriamente dita (Sommerville, 2011). Essa etapa é fundamental para garantir que o *software* atenda aos objetivos propostos, além de servir como base para implementação, manutenção e avaliação do sistema.

No contexto deste trabalho, a modelagem do aplicativo LembreMédi foi conduzida a partir de uma abordagem prática e incremental, baseada em necessidades reais de uso, uma vez que o sistema surgiu inicialmente para solucionar um problema pessoal do autor relacionado à organização e adesão a tratamentos medicamentosos. Posteriormente, essa modelagem foi formalizada por meio de documentos técnicos e diagramas, alinhando o projeto às boas práticas de engenharia de *software*.

As subseções a seguir apresentam detalhadamente o levantamento de requisitos, os diagramas de casos de uso, os diagramas estruturais, a arquitetura do sistema e a modelagem da interface do aplicativo.

3.1. Levantamento de Requisitos

O levantamento de requisitos do sistema foi realizado a partir da análise do problema a ser resolvido: a dificuldade de usuários em organizar e seguir corretamente horários de medicação. Essa etapa é crucial, pois consiste na compreensão das necessidades dos usuários e do negócio, estabelecendo as bases de design e construção que garantem que o produto final realmente solucione o problema identificado (Pressman; Maxim, 2021).

Diferentemente de projetos tradicionais que iniciam com entrevistas e pesquisas com usuários, neste trabalho os requisitos foram definidos com base em:

- Necessidade prática identificada pelo autor;
- Análise de aplicativos similares (estado da arte);
- Restrições técnicas da plataforma Android.

Os requisitos foram organizados em Requisitos Funcionais (RF) e Requisitos Não Funcionais (RNF), garantindo uma visão clara tanto das funcionalidades quanto das qualidades esperadas do sistema.

3.1.1. Requisitos Funcionais (RF)

Os Requisitos Funcionais (RF) descrevem as funcionalidades que o sistema deve oferecer aos usuários, ou seja, as ações que o aplicativo é capaz de executar para atender ao seu propósito principal. Conforme explica Sommerville (2011), esses requisitos detalham os serviços que o software deve fornecer, como ele deve reagir a entradas específicas e como deve se comportar em determinadas situações. Sua importância reside no fato de que eles compõem a base para o desenvolvimento técnico, garantindo que as necessidades operacionais dos usuários sejam plenamente satisfeitas pelo sistema.

No contexto do LembreMédi, os requisitos funcionais estão diretamente relacionados ao gerenciamento de alarmes de medicamentos e ao controle básico do tratamento medicamentoso. O Quadro 1 apresenta os Requisitos Funcionais (RF) identificados para o sistema, descrevendo as funcionalidades que o aplicativo deve oferecer aos usuários.

Quadro 1 - Requisitos funcionais

Código	Requisito	Descrição
RF-01	Cadastrar alarme de medicamento	O sistema deve permitir que o usuário cadastre um novo alarme de medicamento informando os dados necessários para agendamento e exibição.
RF-02	Validar dados obrigatórios no cadastro	O sistema deve validar os dados inseridos no formulário de cadastro antes de permitir salvar o alarme.
RF-03	Listar alarmes cadastrados	O sistema deve exibir ao usuário uma lista contendo todos os alarmes cadastrados.

RF-04	Acessar detalhes do alarme	O sistema deve permitir que o usuário acesse a tela de detalhes de um alarme selecionado na lista.
RF-05	Editar alarme cadastrado	O sistema deve permitir que o usuário edite um alarme existente.
RF-06	Excluir alarme cadastrado	O sistema deve permitir que o usuário exclua um alarme existente.
RF-07	Exibir tela de alarme ao usuário	O sistema deve abrir uma tela específica para avisar o usuário quando um alarme estiver tocando

Fonte: Elaborado pelo autor

3.1.2. Requisitos Não Funcionais (RNF)

Os Requisitos Não Funcionais (RNF) especificam os critérios de qualidade, restrições técnicas e características operacionais que o sistema deve atender, mas que não estão diretamente relacionadas a funcionalidades visíveis ao usuário. Segundo Pressman e Maxim (2021), esses requisitos são fundamentais, pois definem propriedades do sistema como desempenho, segurança, disponibilidade e usabilidade. A importância dos RNFs reside no fato de que, se não forem devidamente atendidos, o sistema pode ser funcionalmente correto, mas falhar em satisfazer as expectativas dos usuários ou as necessidades do ambiente operacional.

No LembreMédi, os RNFs possuem papel fundamental, uma vez que a confiabilidade do sistema está diretamente associada à sua capacidade de:

- Executar alarmes com precisão;
- Manter funcionamento mesmo sob restrições do sistema Android;
- Garantir desempenho adequado em dispositivos variados.

Os Requisitos Não Funcionais, que especificam os critérios de qualidade e restrições técnicas do sistema, estão detalhados no Quadro 2.

Quadro 2 - Requisitos não funcionais

Código	Requisito	Descrição
--------	-----------	-----------

RNF-01	Tempo de inicialização do aplicativo	O aplicativo deve iniciar e apresentar a tela principal em até 2 segundos em condições normais de uso.
RNF-02	Tempo de carregamento da lista de alarmes	A listagem de alarmes deve ser carregada e exibida em até 1 segundo após a abertura da tela.
RNF-03	Uso eficiente de memória	O aplicativo deve manter o consumo de memória inferior a 150 MB de RAM durante o uso normal.
RNF-04	Confiabilidade no disparo de alarmes	O sistema deve disparar os alarmes programados com precisão máxima de ± 5 segundos em relação ao horário configurado.
RNF-05	Continuidade de execução em segundo plano	Quando um alarme for disparado, o sistema deve continuar executando o processo de alerta mesmo se o app estiver em segundo plano.
RNF-06	Tratamento de múltiplos alarmes simultâneos	O sistema deve lidar corretamente com alarmes disparados em sequência ou simultaneamente.
RNF-07	Persistência e recuperação após reinício	Após a reinicialização do dispositivo, o sistema deve restaurar automaticamente 100% dos alarmes previamente cadastrados.
RNF-08	Interface simples e intuitiva	A interface deve permitir que usuários realizem as principais tarefas (cadastro e confirmação de alarmes) sem treinamento prévio, alcançando pontuação mínima de 70 pontos na escala SUS.
RNF-09	Armazenamento local e privacidade	O aplicativo deve armazenar dados localmente, preservando a privacidade.
RNF-10	Proteção e criptografia de dados	Os dados sensíveis do usuário devem ser armazenados localmente utilizando mecanismos de criptografia fornecidos pela plataforma Android.

Fonte: Elaborado pelo autor

3.2. Diagramas de Casos de Uso

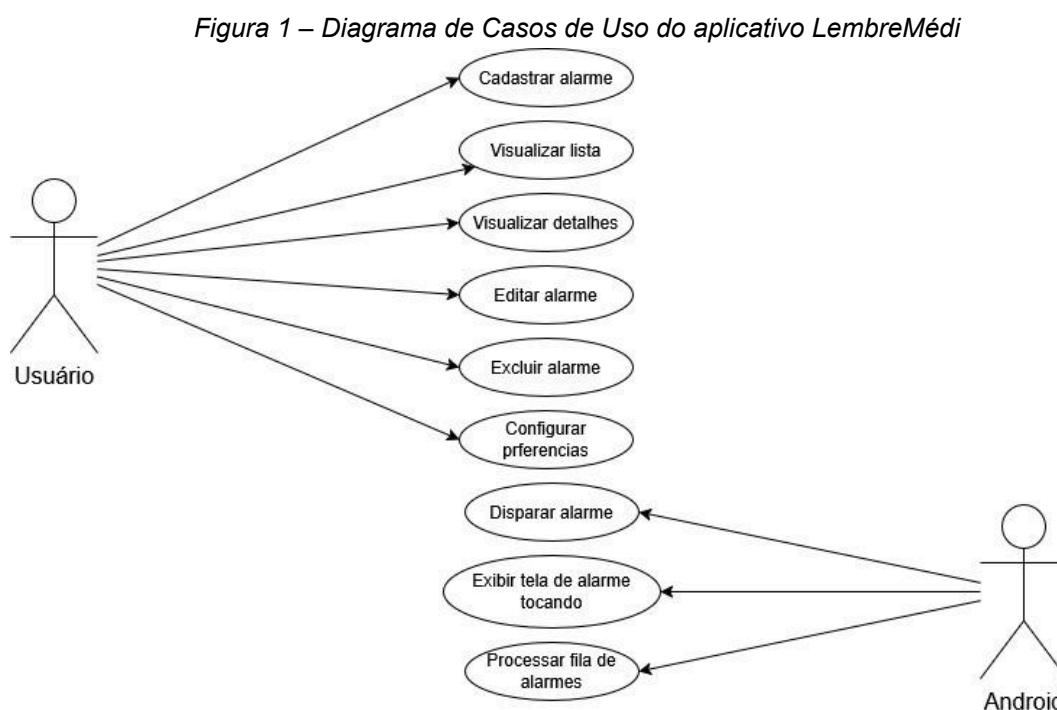
Os diagramas de casos de uso têm como finalidade representar, de forma visual e simplificada, as interações possíveis entre o usuário e o sistema LembreMédi. Esses diagramas auxiliam na compreensão das funcionalidades

oferecidas pelo aplicativo, independentemente de detalhes técnicos de implementação. De acordo com Guedes (2018), essa técnica de modelagem é essencial para descrever o comportamento do sistema sob o ponto de vista do usuário, permitindo que as funcionalidades sejam validadas e compreendidas sem a necessidade de conhecimento sobre a codificação interna.

O principal ator identificado no sistema é o usuário, responsável por interagir com todas as funcionalidades disponíveis no LembreMédi. Entre os principais casos de uso modelados, destacam-se:

- Cadastrar alarme de medicamento;
- Editar alarme existente;
- Remover alarme;
- Visualizar lista de alarmes;
- Confirmar uso do medicamento;
- Configurar preferências do aplicativo.

A Figura 1 apresenta o Diagrama de Casos de Uso do aplicativo LembreMédi, ilustrando as principais interações entre o usuário e as funcionalidades disponíveis no sistema.



Fonte: Elaborado pelo autor

3.3. Diagramas de Classes (UML)

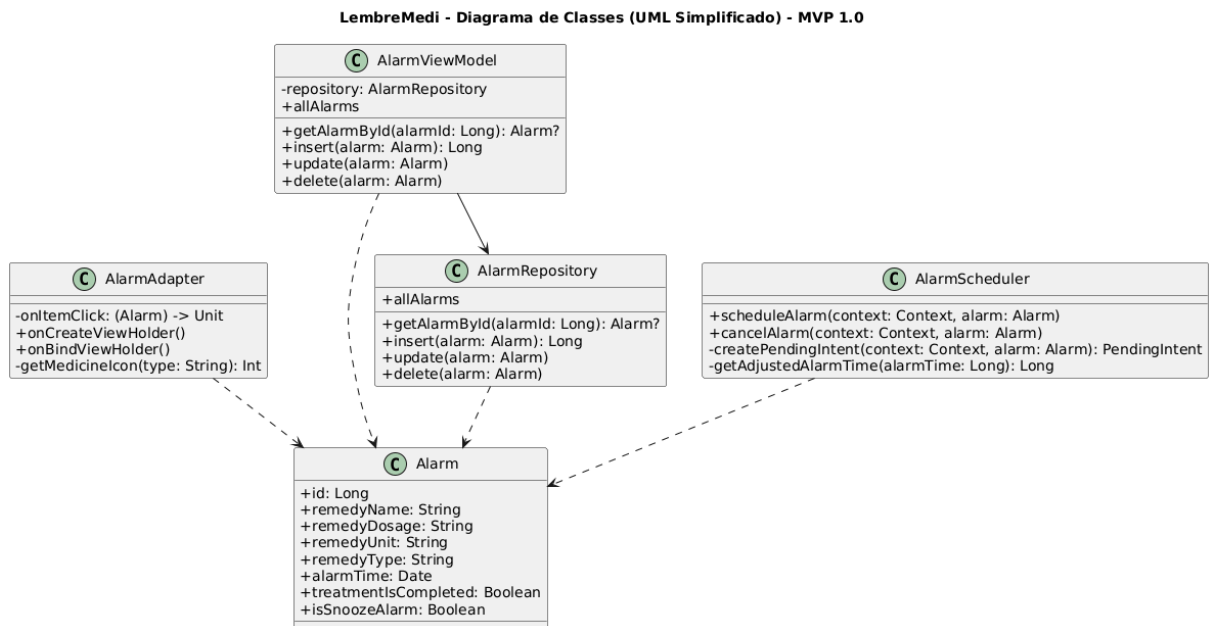
Os diagramas de classes representam a estrutura estática do sistema, evidenciando as principais entidades, suas propriedades, métodos e relacionamentos. No LembreMédi, esse diagrama foi elaborado de forma simplificada, focando nas classes centrais que compõem a lógica do sistema. Segundo Guedes (2018), o diagrama de classes é considerado o mais importante da UML, pois serve de base para a construção da maioria dos outros diagramas e descreve de forma precisa como os objetos serão estruturados e como eles se relacionam. Sua importância reside na capacidade de mapear o domínio do problema para uma linguagem técnica, facilitando a implementação do código e garantindo a consistência da arquitetura de dados.

Entre as principais classes modeladas estão:

- **Alarm**: entidade principal que representa um alarme de medicamento;
- **AlarmDao**: responsável pelo acesso aos dados persistidos;
- **AlarmRepository**: camada intermediária entre banco de dados e *ViewModel*;
- **AlarmViewModel**: responsável pela lógica de apresentação;
- **AlarmScheduler**: gerenciador de agendamento e cancelamento de alarmes;
- **AlarmReceiver** e **AlarmForegroundService**: responsáveis pelo disparo e execução dos alarmes;
- **AlarmRingingActivity**: interface exibida durante o toque do alarme.

O Diagrama de Classes simplificado do sistema pode ser visualizado na Figura 2, que demonstra a estrutura estática das principais entidades e seus relacionamentos.

Figura 2 – Diagrama de Classes simplificado do sistema



Fonte: Elaborado pelo autor

3.4. Arquitetura do Sistema

A arquitetura do sistema foi definida com base no padrão MVVM (Model–View–ViewModel), aliado a uma organização em camadas bem definidas. Essa abordagem promove separação de responsabilidades, maior testabilidade e facilidade de manutenção. Nesse contexto, a arquitetura de software é compreendida como o design crítico que organiza os componentes de um sistema e suas inter-relações, servindo como uma ponte entre os requisitos e a implementação. Segundo Sommerville (2011), uma arquitetura bem estruturada é vital para garantir que o sistema atenda a atributos de qualidade essenciais, como desempenho e robustez, além de facilitar a evolução do software ao longo do tempo.

O padrão MVVM, especificamente, atua separando a interface do usuário (View) da lógica de negócios e dos dados (Model), utilizando o ViewModel como um mediador que gerencia o estado da interface. De acordo com a documentação do Android Developers (2025i), essa arquitetura permite que as atividades e fragmentos fiquem isolados de mudanças nos dados, tornando o código mais limpo e modular. No contexto do LembreMédi, a escolha pelo MVVM justifica-se pela necessidade de gerenciar dados sensíveis e eventos em tempo real, como os alarmes de medicação, garantindo que a interface permaneça responsiva e que os dados de

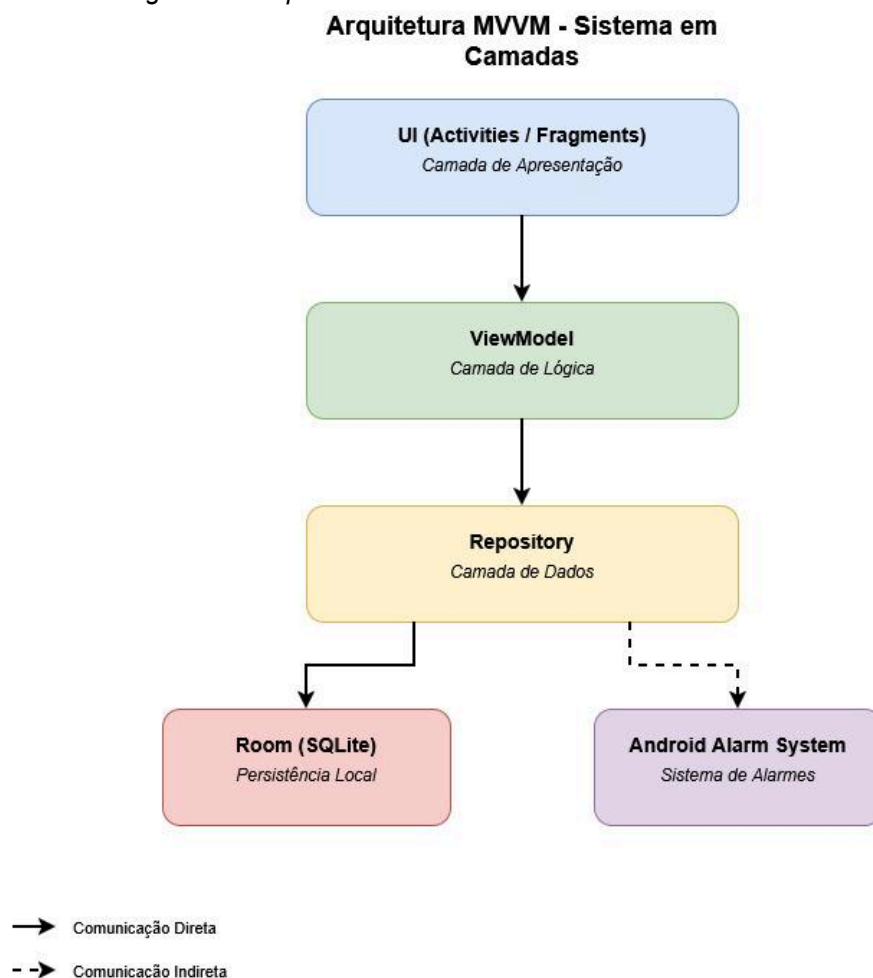
horários persistidos via biblioteca Room sejam atualizados de forma consistente sem sobrecarregar a interação do usuário.

O sistema foi dividido nas seguintes camadas:

- **Camada de Interface (UI):** *Activities, layouts XML e adapters;*
- **Camada de *ViewModel*:** gerenciamento de estado e lógica de apresentação;
- **Camada de Repositório:** abstração de acesso a dados;
- **Camada de Persistência:** banco de dados local utilizando Room;
- **Camada de Sistema *Android*:** *AlarmManager, Services, Receivers* e permissões.

Além disso, o projeto conta com componentes específicos para o fluxo crítico de alarmes, garantindo execução confiável mesmo em condições adversas do sistema operacional. A arquitetura do sistema baseada no padrão MVVM está representada na Figura 3, evidenciando a separação em camadas e o fluxo de dados entre os componentes.

Figura 3 – Arquitetura do sistema baseada em MVVM



Fonte: Elaborado pelo autor

3.5. Interface e Navegação

A interface do sistema foi projetada com foco na simplicidade, clareza visual e facilidade de uso, considerando que o aplicativo será utilizado em contextos sensíveis, como lembretes de medicação, onde rapidez e compreensão imediata são essenciais.

A modelagem da interface do aplicativo LembreMédi foi realizada a partir da definição de um fluxo de navegação linear e intuitivo, minimizando o número de etapas necessárias para executar ações críticas, como visualizar alarmes ou confirmar o uso de um medicamento. Essa abordagem busca reduzir erros de uso e aumentar a adesão ao tratamento.

A seguir, são apresentadas as telas que compõem o sistema, bem como o fluxo de navegação entre elas.

3.5.1. Fluxo Geral de Navegação

O fluxo de navegação do aplicativo inicia-se na Tela Inicial (Lista de Alarmes), que atua como ponto central do sistema. A partir dessa tela, o usuário pode:

- Visualizar todos os alarmes cadastrados;
- Acessar os detalhes de um alarme específico;
- Criar um novo alarme;
- Navegar para as configurações do aplicativo.

O fluxo principal pode ser resumido da seguinte forma:

1. O usuário acessa a tela inicial, onde visualiza a lista de alarmes cadastrados;
2. A partir da tela inicial, o usuário pode:
 - Criar um novo alarme;
 - Acessar a tela de detalhes onde pode editar ou excluir um alarme;
 - Acessar a tela de configurações do aplicativo.
3. Quando um alarme é disparado, o sistema interrompe o fluxo atual e apresenta a tela de alarme tocando, que possui prioridade máxima;
4. Após a confirmação ou adiamento do alarme, o usuário retorna ao fluxo normal do aplicativo.

Esse fluxo foi definido para garantir que ações críticas, como exclusão de alarmes, não ocorram de forma acidental, sempre passando por uma tela intermediária de confirmação e visualização de dados. E o evento crítico, o disparo do alarme, seja tratado de forma imediata e destacada, sem comprometer a usabilidade do sistema.

3.5.2. Tela Inicial – Lista de Alarmes

A Tela Inicial é a primeira tela exibida ao usuário ao abrir o aplicativo. Ela apresenta uma lista com todos os alarmes cadastrados, ordenados de acordo com o horário da próxima dose.

Cada item da lista exibe informações essenciais, como:

- Nome do medicamento;

- Dosagem e unidade;
- Horário da próxima dose ou status de tratamento concluído;
- Ícone representativo do tipo de medicamento.

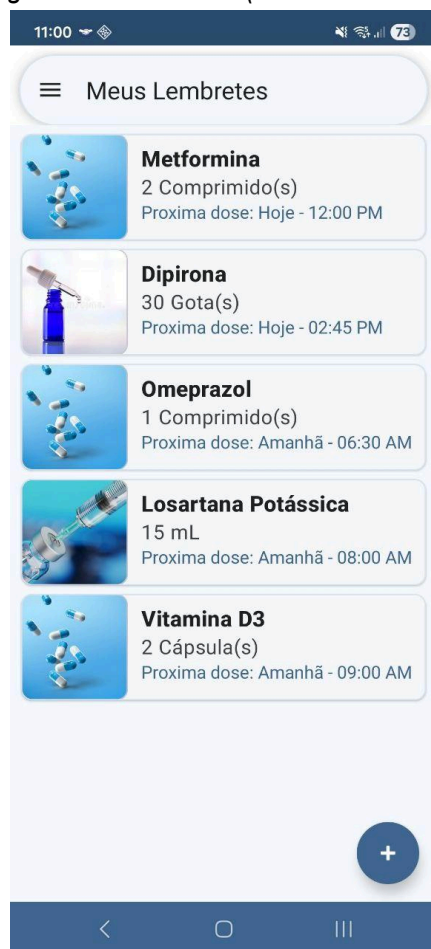
Essa tela permite uma visão rápida e objetiva da rotina de medicamentos do usuário, funcionando como o painel principal de controle do aplicativo.

Ações disponíveis nesta tela:

- Selecionar um alarme para visualizar seus detalhes;
- Acessar a tela de criação de um novo alarme;
- Acessar as configurações do aplicativo.

A Tela Inicial do aplicativo, que exibe a lista de alarmes cadastrados, é mostrada na Figura 4, funcionando como o painel principal de controle do sistema.

Figura 4 – Tela Inicial (Lista de Alarmes)



Fonte: Elaborado pelo autor

3.5.3. Tela de Detalhes do Alarme

A Tela de Detalhes do Alarme atua como uma tela intermediária entre a visualização e as ações de edição ou exclusão de um alarme. Seu objetivo é permitir que o usuário revise todas as informações do alarme antes de realizar qualquer alteração.

Nesta tela, são exibidos de forma detalhada:

- Nome e descrição do medicamento;
- Tipo de medicamento;
- Dosagem e unidade;
- Horário configurado;
- Intervalo entre doses;
- Duração do tratamento;
- Quantidade de doses já tomadas;
- *Status* do tratamento (ativo ou concluído).

Essa abordagem reduz a possibilidade de alterações ou exclusões acidentais, garantindo maior segurança na gestão dos alarmes.

Ações disponíveis nesta tela:

- Navegar para a tela de edição do alarme;
- Excluir o alarme;
- Retornar à tela inicial.

Conforme ilustrado na Figura 5, a Tela de Detalhes do Alarme permite ao usuário revisar todas as informações do medicamento antes de realizar alterações.

Figura 5 – Tela de Detalhes do Alarme



Fonte: Elaborado pelo autor

3.5.4. Tela de Cadastro de Alarme

A Tela de Cadastro de Alarme é utilizada para criar um novo alarme de medicamento. Nessa tela, o usuário informa todos os dados necessários para o funcionamento correto do lembrete.

Os campos disponíveis incluem:

- Nome do medicamento;
- Tipo de medicamento;
- Dosagem e unidade;
- Observação (opcional);
- Horário do alarme;
- Duração do tratamento;
- Intervalo entre doses,

A organização dos campos segue uma ordem lógica, facilitando o preenchimento e reduzindo erros por parte do usuário. Após o preenchimento, o usuário pode confirmar o cadastro, momento em que o sistema realiza o agendamento do alarme junto ao sistema operacional *Android*.

Ações disponíveis nesta tela:

- Salvar o novo alarme;
- Cancelar o cadastro e retornar à tela anterior.

A Tela de Cadastro de Alarme, apresentada na Figura 6, contém todos os campos necessários para a criação de um novo lembrete de medicamento.

Figura 6 – Tela de Cadastro de Alarme

9:37 4G 87

← Novo lembrete

Detalhes do medicamento

Nome do medicamento
Dipirona
Ex.: Dipirona, Amoxicilina

Tipo
Gotas
Selecione o tipo do medicamento

Dosagem

Quantidade
30
Ex.: 1, 2

Unidade
Gota(s)
Ex.: gotas, ml, mg

Observações (opcional)
Remédio para dores em geral
Ex.: Instruções especiais, efeitos, etc.

Agendamento

Primeira dose

Data
27/01/2026

Hora
09:36

Repetição do tratamento

Duração
1 dias

Intervalo
6 h

Como funciona?

- Primeira dose: data e hora do primeiro lembrete
- Duração: quantos dias tomará o medicamento
- Intervalo: a cada quantas horas (ex.: 8 = de 8 em 8h)

Salvar lembrete

Cancelar

Fonte: Elaborado pelo autor

3.5.5. Tela de Edição de Alarme

A Tela de Edição de Alarme reutiliza a estrutura da tela de cadastro, porém com os campos previamente preenchidos com os dados do alarme selecionado. Essa tela permite que o usuário ajuste informações do alarme, como horário, dosagem ou duração do tratamento, sem a necessidade de recriar o alarme do zero.

Ações disponíveis nesta tela:

- Salvar alterações realizadas;
- Cancelar a edição e retornar à tela de detalhes do alarme.

A Tela de Edição de Alarme, demonstrada na Figura 7, reutiliza a estrutura da tela de cadastro com os campos previamente preenchidos.

Figura 7 – Tela de Edição de Alarme

9:37



 4G 87

 Editar lembrete

 **Detalhes do medicamento**

Nome do medicamento

Dipirona

Ex.: Dipirona, Amoxicilina

Tipo

Gotas

Selecione o tipo do medicamento

Dosagem

Quantidade

30

Ex.: 1, 2

Unidade

Gota(s)

Ex.: gotas, ml, mg

Observações (opcional)

Remédio para dores em geral

Ex.: Instruções especiais, efeitos, etc.

 **Agendamento**

Primeira dose

Data

27/01/2026



Hora

09:40



Repetição do tratamento

Duração

1 dias

Intervalo

6 h

Como funciona?

• Primeira dose: data e hora do primeiro lembrete

• Duração: quantos dias tomará o medicamento

• Intervalo: a cada quantas horas (ex.: 8 = de 8 em 8h)

Salvar Alterações

Cancelar

Fonte: Elaborado pelo autor

3.5.6. Tela de Configurações

A Tela de Configurações permite ao usuário personalizar o comportamento do aplicativo de acordo com suas preferências.

Entre as opções disponíveis estão:

- Ativação ou desativação do modo escuro;
- Ativação ou desativação do modo silencioso;
- Configuração de som do alarme;
- Ativação ou desativação de vibração.

Essa tela contribui para uma experiência mais personalizada e adequada às necessidades individuais de cada usuário. As opções de personalização do aplicativo estão disponíveis na Tela de Configurações, conforme exibido na Figura 8.

Figura 8 – Tela de Configurações do Aplicativo



Fonte: Elaborado pelo autor

3.5.7. Tela de Alarme Tocando

A Tela de Alarme Tocando é exibida automaticamente quando um alarme é disparado pelo sistema, mesmo que o dispositivo esteja bloqueado ou com outro aplicativo em uso.

Essa tela é apresentada em modo de alta prioridade, garantindo que o usuário perceba o alerta. Nela, são exibidas informações essenciais para a tomada do medicamento, como:

- Nome do medicamento;
- Dosagem e unidade;
- Observações do remédio (se houver).

Ações disponíveis nesta tela:

- Confirmar que o medicamento foi tomado;
- Adiar o alarme (soneca), quando disponível.

Após a interação do usuário, o sistema registra a ação e direciona o fluxo para o próximo alarme, quando aplicável. Quando um alarme é disparado, o sistema apresenta a Tela de Alarme Tocando (Figura 9), que possui prioridade máxima de exibição.

Figura 9 – Tela de Alarme Tocando



Fonte: Elaborado pelo autor

4. SOFTWARE

Esta seção descreve os aspectos relacionados à implantação do *software* desenvolvido, bem como a abordagem metodológica adotada para os testes e validação do sistema. O objetivo é apresentar como o aplicativo foi preparado para execução em ambiente real e quais estratégias foram definidas para avaliar sua qualidade, usabilidade e efetividade após o uso por usuários finais.

4.1. Implantação

A implantação do aplicativo LembreMédi foi realizada considerando o ambiente *Android* como plataforma-alvo, utilizando desenvolvimento nativo em *Kotlin* e as bibliotecas oficiais do ecossistema *Android*. O aplicativo foi projetado para funcionar de forma autônoma no dispositivo do usuário, sem dependência de servidores externos, uma vez que o armazenamento e processamento das informações ocorrem localmente por meio do banco de dados Room.

Durante a fase de implantação, foram considerados os seguintes aspectos técnicos:

- **Compatibilidade com versões do *Android*:** O sistema foi desenvolvido visando compatibilidade com versões modernas do *Android*, respeitando as políticas de permissões e restrições introduzidas a partir do *Android* 12 (API 31), especialmente no que se refere ao agendamento de alarmes exatos e execução de serviços em primeiro plano.
- **Gerenciamento de permissões:** O aplicativo solicita permissões críticas, como:
 - Agendamento de alarmes exatos (`SCHEDULE_EXACT_ALARM`);
 - Execução de serviços em primeiro plano (`FOREGROUND_SERVICE`);
 - Sobreposição de tela (`SYSTEM_ALERT_WINDOW`);
 - Vibração e controle de energia (`WAKE_LOCK`).

Essas permissões são solicitadas de forma explícita ao usuário, seguindo as diretrizes de boas práticas da plataforma Android.

- **Persistência de dados:** Todos os dados relacionados aos alarmes, configurações do usuário e estado do tratamento são armazenados localmente no dispositivo, garantindo funcionamento offline e maior privacidade das informações do usuário.

A implantação do sistema foi concebida para refletir um cenário real de uso, possibilitando que os usuários finais utilizem o aplicativo em seu cotidiano sem a necessidade de configurações técnicas adicionais.

4.2. Disponibilização do Software

Para fins de desenvolvimento e controle de versões, o código-fonte do aplicativo foi mantido em um repositório privado na plataforma GitHub. A opção por manter o repositório privado deve-se à intenção de continuidade do desenvolvimento do sistema e à possibilidade de exploração comercial futura da aplicação.

Durante a fase de avaliação do sistema, o aplicativo foi disponibilizado aos usuários participantes por meio da instalação direta do pacote APK nos dispositivos

Android. A publicação oficial do aplicativo na *Google Play Store* encontra-se em fase de preparação, sendo prevista sua disponibilização pública após a conclusão das etapas finais de validação e eventuais ajustes identificados durante o processo de avaliação descrito neste relatório.

4.3. Análise Comparativa com Aplicativos Existentes

Com o objetivo de contextualizar o aplicativo desenvolvido neste trabalho em relação a soluções já disponíveis no mercado, foi realizada uma análise comparativa com dois aplicativos amplamente utilizados no gerenciamento de medicamentos: *Medisafe* e *MyTherapy*. A comparação foi realizada considerando funcionalidades perceptíveis ao usuário final, especialmente aquelas relacionadas ao gerenciamento de alarmes, interação com notificações e organização da rotina de medicamentos. O Quadro 3 apresenta uma síntese das principais características observadas.

Quadro 3 – Quadro Comparativa de Funcionalidades Percebidas pelo Usuário

Funcionalidade	Medisafe	MyTherapy	LembreMédi (MVP)
Cadastro simples de medicamentos	Sim	Sim	Sim
Configuração de alarmes de medicamentos	Sim	Sim	Sim
Notificação de lembrete no horário programado	Sim	Sim	Sim
Tela dedicada para confirmação da dose	Parcial	Parcial	Sim
Alarmes com som e vibração	Sim	Sim	Sim
Funcionamento sem conexão com internet	Parcial	Parcial	Sim
Armazenamento local dos dados	Parcial	Parcial	Sim
Interface focada apenas na rotina de medicamentos	Não	Não	Sim
Simplicidade de uso para usuários iniciantes	Média	Média	Alta
Dependência de serviços externos	Sim	Sim	Não

Privacidade dos dados (sem envio para servidores)	Parcial	Parcial	Sim
---	---------	---------	-----

Fonte: Elaborado pelo autor

A partir da comparação apresentada, observa-se que os aplicativos comerciais analisados oferecem uma ampla variedade de funcionalidades voltadas ao acompanhamento geral da saúde. Entretanto, o LembreMédi apresenta características relevantes no contexto da proposta deste trabalho, destacando-se pelo funcionamento totalmente offline, pelo armazenamento local das informações e pela interface simplificada voltada especificamente para o gerenciamento de alarmes de medicamentos. Essa abordagem prioriza a confiabilidade do sistema de alertas e a facilidade de uso, aspectos considerados fundamentais para usuários que necessitam de lembretes consistentes para a administração de tratamentos.

4.4. Testes

A fase de testes do aplicativo LembreMédi foi planejada com foco na avaliação da qualidade percebida pelo usuário final, considerando principalmente aspectos de usabilidade e efetividade do sistema no apoio à rotina de administração de medicamentos.

Diferentemente de testes automatizados ou testes unitários tradicionais, esta etapa do trabalho prioriza métodos de avaliação centrados no usuário, alinhados aos objetivos do trabalho e à proposta do aplicativo como ferramenta de apoio à saúde.

4.4.1. Avaliação de Usabilidade – *System Usability Scale* (SUS)

Para a avaliação da usabilidade do sistema, foi utilizada a *System Usability Scale* (SUS), uma escala padronizada amplamente adotada para medir a percepção de usabilidade de sistemas interativos. A escala SUS consiste em um questionário com 10 afirmações, avaliadas pelo usuário em uma escala numérica de cinco pontos, variando de “discordo totalmente” a “concordo totalmente”.

De acordo com Brooke (1996), a SUS é uma ferramenta de baixo custo que permite avaliar de forma global a usabilidade de um sistema, sendo particularmente eficaz por ser independente da tecnologia utilizada e por fornecer resultados robustos mesmo em amostras pequenas. O autor ressalta que essa métrica captura

a percepção subjetiva do usuário sobre a eficácia, eficiência e satisfação no uso da interface. O resultado final gera uma pontuação que permite classificar a usabilidade do sistema em níveis como inadequado, aceitável ou excelente.

A escolha da SUS se justifica pelos seguintes fatores:

- Facilidade de aplicação;
- Reconhecimento acadêmico e industrial;
- Capacidade de fornecer uma métrica quantitativa clara sobre a experiência do usuário;
- Adequação para amostras pequenas de usuários.

Os testes do aplicativo foram realizados com 31 voluntários recrutados por conveniência, compostos majoritariamente por conhecidos do autor e de familiares. Não foram coletados dados sociodemográficos no questionário, com o objetivo de preservar a privacidade dos participantes. Contudo, com base em informações previamente conhecidas, e obtidas de forma não estruturada e posterior à aplicação do questionário, a maioria dos voluntários se encontrava na faixa etária entre 18 e 28 anos. Todos possuíam, no mínimo, ensino médio completo, sendo parte deles estudantes do ensino superior. Quanto à localização geográfica, os participantes estavam concentrados nas cidades de Picos, no estado do Piauí, e Afrânio, no estado de Pernambuco, sendo esta última responsável pela maior parte da amostra. Os voluntários, que utilizaram o aplicativo LembreMédi e, em seguida, responderam ao questionário, atribuindo notas de 1 a 5 para cada afirmação, conforme seu nível de concordância (1 = Discordo totalmente; 5 = Concordo totalmente).

O Quadro 4 apresenta as dez afirmações que compõem o questionário *System Usability Scale* (SUS) aplicado aos participantes da avaliação de usabilidade.

Quadro 4 – Questionário SUS aplicado aos usuários do aplicativo LembreMédi

Item	Afirmação
1	Eu acho que gostaria de usar o aplicativo LembreMédi com frequência.
2	Eu achei o aplicativo LembreMédi desnecessariamente complexo.

Item	Afirmação
3	Eu achei o aplicativo LembreMédi fácil de usar.
4	Eu acho que precisaria de ajuda de alguém com conhecimento técnico para conseguir usar o LembreMédi.
5	Eu achei que as funcionalidades do LembreMédi são bem integradas entre si.
6	Eu achei que houve muita inconsistência no funcionamento do LembreMédi.
7	Eu imagino que a maioria das pessoas aprenderia a usar o LembreMédi rapidamente.
8	Eu achei o aplicativo LembreMédi complicado de usar.
9	Eu me senti confiante ao usar o LembreMédi.
10	Eu precisei aprender muitas coisas antes de conseguir usar bem o LembreMédi.

Fonte: Elaborado pelo autor

O cálculo da pontuação SUS seguiu o método padronizado por Brooke (1996), conforme as seguintes etapas:

1. Para as afirmações ímpares (itens 1, 3, 5, 7, 9): subtrai-se 1 da nota atribuída pelo usuário.
2. Para as afirmações pares (itens 2, 4, 6, 8, 10): subtrai-se a nota atribuída de 5.
3. Soma-se os valores ajustados de todas as dez questões.
4. Multiplica-se o total por 2,5 para obter a pontuação final em uma escala de 0 a 100.

A Tabela 1 apresenta as pontuações individuais de cada participante, calculadas a partir das respostas fornecidas.

Tabela 1 – Pontuação SUS individual dos participantes

Participante	Pontuação SUS	Participante	Pontuação SUS
1	77,5	17	95,0
2	62,5	18	82,5
3	100,0	19	95,0
4	82,5	20	100,0

5	65,0	21	97,5
6	95,0	22	95,0
7	100,0	23	87,5
8	97,5	24	97,5
9	92,5	25	72,5
10	77,5	26	100,0
11	100,0	27	77,5
12	95,0	28	80,0
13	97,5	29	95,0
14	97,5	30	62,5
15	97,5	31	70,0
16	92,5		

Fonte: Elaborado pelo autor

Após o cálculo individual, obteve-se a pontuação SUS média do aplicativo LembreMédi:

- **Média das pontuações:** 88,31 pontos
- **Desvio padrão:** 12,17
- **Pontuação mínima:** 62,5
- **Pontuação máxima:** 100,0

Conforme a escala de avaliações adjetivas proposta por Bangor, Kortum e Miller (2009), pontuações SUS entre 53 e 73 são classificadas como “Boas”, entre 74 e 85 são classificadas como “Excelentes”, enquanto valores entre 86 e 100 são considerados “O melhor imaginável”. Além disso, na escala de notas (grades), pontuações entre 70 e 79 correspondem ao conceito de “C”, entre 80 e 89 correspondem ao conceito “B”, e valores entre 90 e 100 ao conceito “A”. Dessa forma, a média obtida de 88,31 pontos posiciona o aplicativo LembreMédi na categoria “O melhor imaginável”, com nota B, indicando um nível muito elevado de usabilidade e uma experiência altamente positiva para a maioria dos usuários.

Observa-se que não houve pontuações abaixo de 52 pontos, o que indica que nenhum participante classificou a usabilidade do sistema nas categorias “Pobre” ou “O pior imaginável”, segundo a escala adjetiva proposta por Bangor, Kortum e Miller (2009). As menores pontuações registradas foram 62,5 (participantes 2 e 30), enquadradas na faixa “Boas” (53–73), evidenciando avaliações ainda positivas do aplicativo. A maioria dos participantes apresentou pontuações superiores a 80 pontos, concentrando-se principalmente nas faixas “Excelentes” (74–85) e “O melhor imaginável” (86–100), sendo que diversos usuários alcançaram valores iguais ou superiores a 95 pontos, o que reforça a percepção de alta qualidade de usabilidade do aplicativo.

A dispersão dos resultados, indicada pelo desvio padrão de 12,17, revela uma variabilidade moderada entre as avaliações. Isso sugere que, embora a percepção geral tenha sido bastante positiva, alguns usuários podem ter encontrado pequenas dificuldades, o que impactou suas pontuações individuais.

De modo geral, os resultados obtidos demonstram que o aplicativo LembreMédi atende de forma satisfatória aos critérios de usabilidade estabelecidos, proporcionando uma experiência intuitiva, eficiente e bem aceita pela maioria dos usuários avaliados..

4.4.2. Avaliação da Efetividade do Sistema (Pós-Uso)

Além da usabilidade, foi realizada uma avaliação da efetividade do aplicativo, com o objetivo de verificar se o LembreMédi cumpre sua proposta principal: auxiliar o usuário na organização e gerenciamento da rotina de medicamentos.

Para essa avaliação, foi adotado o seguinte procedimento metodológico:

- Seleção de um grupo de usuários voluntários;
- Utilização do aplicativo por um período contínuo, estimado entre 7 e 14 dias;
- Uso do aplicativo em contexto real, integrado à rotina diária dos participantes;
- Aplicação de um questionário pós-uso, contendo perguntas relacionadas à percepção de utilidade, facilidade de uso, impacto na rotina e confiança no sistema.

O perfil dos participantes desta etapa constituiu um subproduto da amostra utilizada na avaliação de usabilidade. Inicialmente, os 31 voluntários participaram do teste de usabilidade e, posteriormente, foi selecionado um subconjunto de 12 usuários pertencentes a esse mesmo grupo, que relataram estar em uso atual de algum tratamento medicamentoso oral, seja por prescrição médica formal (por exemplo, tratamentos obtidos mediante receita) ou por iniciativa própria, como o uso regular de suplementos vitamínicos. Esse critério resultou em uma subamostra correspondente a aproximadamente um terço dos participantes iniciais, os quais foram então incluídos na avaliação de efetividade.

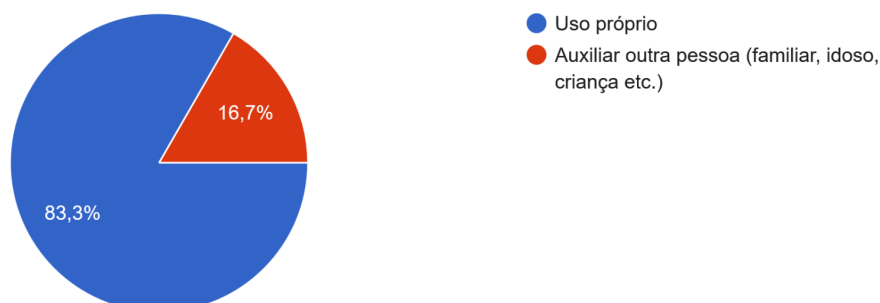
O período de uso do aplicativo foi definido entre 7 e 14 dias por representar, de forma aproximada, a duração mais comum de tratamentos medicamentosos orais de caráter agudo, tais como terapias antibióticas e anti-inflamatórias. Esse intervalo foi considerado suficiente para que o usuário vivenciasse o ciclo completo ou parcial do tratamento, permitindo avaliar a efetividade do sistema no apoio à organização, adesão às doses e gerenciamento de alarmes.

Os 12 participantes da avaliação utilizaram o aplicativo por períodos variados, com a maioria na faixa entre 7 e 10 dias, correspondente a 66,7% da amostra. Quanto à finalidade de uso, 83,3% dos participantes utilizaram o aplicativo para uso próprio, enquanto 16,7% o fizeram para auxiliar outra pessoa, como familiar, idoso ou criança. Esse resultado indica que o aplicativo atende tanto usuários diretos quanto cuidadores informais, ampliando seu escopo de aplicação. Conforme demonstrado na figura 10.

Figura 10 – Utilização do aplicativo

2. Você utilizou o aplicativo principalmente para:

12 respostas



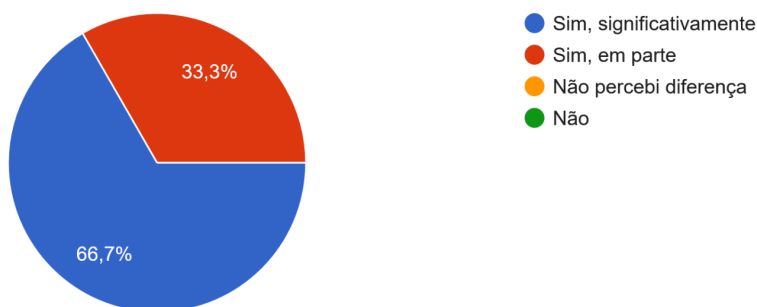
Fonte: Elaborado pelo autor

Os resultados obtidos demonstram uma avaliação amplamente positiva quanto à efetividade do aplicativo. Todos os participantes afirmaram que o aplicativo ajudou a lembrar de tomar os medicamentos nos horários corretos, sendo que 66,7% responderam "ajudou muito" e 33,3% "ajudou bastante". Da mesma forma, a totalidade dos usuários considerou que passou a esquecer ou atrasar menos doses após a utilização do aplicativo, com 66,7% indicando redução significativa e 33,3% redução parcial. No que tange à organização da rotina de medicação, todos os participantes declararam que o aplicativo contribuiu para esse aspecto, sendo que metade da amostra 50% assinalou "contribuiu completamente" e a outra metade 50% "contribuiu bastante". A Figura 11 sintetiza os resultados da percepção dos usuários quanto à efetividade do aplicativo no apoio à rotina de medicação.

Figura 11 – Percepção de efetividade do aplicativo

4. Após utilizar o aplicativo, você considera que esqueceu ou atrasou menos doses de medicamentos?

12 respostas



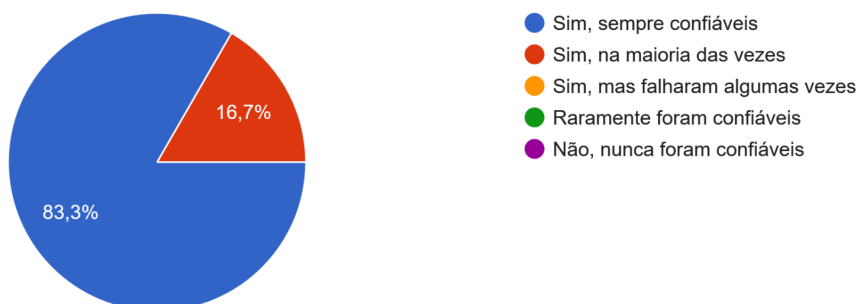
Fonte: Elaborado pelo autor

No que se refere à confiabilidade dos alarmes, 83,3% dos usuários relataram que estes foram sempre confiáveis, enquanto os 16,7% restantes indicaram que foram confiáveis na maioria das vezes. Não houve relatos de falhas frequentes ou ausência de alarmes ao longo do período de teste. Os dados relativos à confiabilidade dos alarmes percebida pelos usuários estão apresentados na Figura 12.

Figura 12 – Confiabilidade dos alarmes

6. Os alarmes foram confiáveis (tocaram no horário esperado)?

12 respostas



Fonte: Elaborado pelo autor

Quanto à clareza das informações exibidas durante o alarme, 91,7% dos participantes consideraram-nas "claras e suficientes" ou "muito claras e completas".

Apenas um participante (8,3%) classificou as informações como "mais ou menos / razoáveis", não havendo avaliações negativas quanto a esse aspecto.

Em relação à segurança percebida, 91,7% dos usuários sentiram-se "bastante seguros" ou "totalmente seguros" ao utilizar o aplicativo como apoio à medicação. Quanto ao atendimento das necessidades, todos os participantes 100% afirmaram que o aplicativo atendeu suas demandas para lembrar e gerenciar medicamentos, sendo que 41,7% responderam "atendeu completamente" e 58,3% "atendeu a maioria".

Os indicadores de satisfação e adesão revelaram-se majoritariamente positivos. Do total de participantes, 83,3% afirmaram que continuariam utilizando o aplicativo no dia a dia, enquanto os 16,7% restantes manifestaram que talvez continuariam utilizando. De forma análoga, 83,3% declararam que recomendariam o aplicativo a outra pessoa que faz uso contínuo de medicamentos, e 16,7% indicaram que talvez o recomendariam.

Quando questionados sobre a principal contribuição do aplicativo durante o período de uso, as respostas mais frequentes foram: "lembrar de tomar os remédios na hora certa", "não esquecer os medicamentos" e "saber o remédio e dose corretos". Esses relatos reforçam que o objetivo central do sistema foi alcançado.

As principais dificuldades apontadas pelos usuários incluíram: falta de um histórico de medicamentos tomados (mencionado por 50% dos participantes), ilustrações dos medicamentos muito semelhantes entre si (41,7%) e ausência de opção para registrar doses não tomadas (um participante). Como contrapartida, as sugestões de melhoria mais recorrentes foram: implementação de um histórico detalhado de medicamentos tomados, com registro de datas e horários; maior diferenciação visual entre os medicamentos; e inclusão de funcionalidade para registro de doses ignoradas ou não tomadas. Esses *feedbacks* constituem insumos relevantes para orientar o aprimoramento contínuo do aplicativo.

5. CONSIDERAÇÕES FINAIS

O presente Relatório Técnico de *Software* descreveu o desenvolvimento do aplicativo LembreMédi, uma solução nativa para a plataforma *Android* destinada ao gerenciamento de alarmes de medicamentos, contemplando as etapas de concepção, definição de requisitos, modelagem, seleção tecnológica, implementação e avaliação com usuários reais.

Os resultados indicam que o aplicativo atendeu de forma satisfatória aos objetivos propostos, evidenciando robustez técnica na implementação do sistema de alarmes para gerenciamento de medicamentos. A utilização de componentes nativos do *Android*, como *AlarmManager*, *BroadcastReceiver*, *ForegroundService* e telas de sobreposição, garantiu o disparo confiável dos alertas mesmo sob restrições energéticas do dispositivo. Além disso, a arquitetura baseada no padrão MVVM, com organização em camadas (UI, *ViewModel*, *Repository* e persistência com Room), contribuiu para a estabilidade, previsibilidade e manutenção do comportamento do sistema.

No que se refere à usabilidade, a avaliação com 31 participantes resultou em média de 88,31 pontos na escala SUS, indicando nível “melhor imaginável” de acordo com a escala de avaliações adjetivas proposta por Bangor, Kortum e Mille (2009) e demonstrando facilidade de uso e compreensão dos fluxos de navegação e gerenciamento de alarmes. Quanto à efetividade, o subconjunto de 12 usuários que utilizaram o aplicativo em contexto real relatou melhora na lembrança dos horários e redução de esquecimentos, confirmando o apoio à adesão medicamentosa. Ademais, não foram observados relatos recorrentes de falhas durante os testes, reforçando a confiabilidade operacional e a adequação das tecnologias empregadas.

Apesar dos resultados amplamente favoráveis, a avaliação com usuários reais também permitiu identificar oportunidades de aprimoramento para versões futuras do aplicativo. Entre as principais sugestões destacam-se a implementação de um histórico detalhado de medicamentos e registros de uso, o aprimoramento da diferenciação visual entre medicamentos cadastrados, o registro de doses ignoradas ou não tomadas e a ampliação das opções de personalização dos alarmes. Tais propostas indicam caminhos para a evolução contínua do sistema, contribuindo para

o aumento da funcionalidade, da rastreabilidade das informações e da adequação às necessidades individuais dos usuários.

Dessa forma, conclui-se que o LembreMedi apresenta-se como uma solução tecnicamente sólida, com elevada aceitação e potencial de impacto positivo na adesão medicamentosa, ao mesmo tempo em que mantém espaço para aperfeiçoamentos que poderão ampliar ainda mais sua utilidade e abrangência em contextos reais de uso.

REFERÊNCIAS

AITKEN, M. **Patient Apps for Improved Healthcare: Novelty to Mainstream**. [S. l.: s. n.], 2013. Disponível em: https://ignacioriesgo.es/wp-content/uploads/2014/03/iihi_patient_apps_report_editora_39_2_1.pdf. Acesso em: 29 jan. 2026.

ANDROID DEVELOPERS. **Activities and Tasks**. 2026a. Disponível em: <https://developer.android.com/guide/components/activities/intro-activities>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **AlarmManager**. 2026b. Disponível em: <https://developer.android.com/reference/android/app/AlarmManager>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Android API reference and platform guides**. 2025a. Disponível em: <https://developer.android.com/about/versions/15?hl=pt-br>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Audio focus**. 2025b. Disponível em: <https://developer.android.com/guide/topics/media-apps/audio-focus>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **BroadcastReceiver**. 2025c. Disponível em: <https://developer.android.com/reference/android/content/BroadcastReceiver>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Data and file storage overview**. 2025d. Disponível em: <https://developer.android.com/training/data-storage/shared-preferences>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Develop Android apps with Kotlin**. 2025e. Disponível em: <https://developer.android.com/kotlin>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Foreground services overview**. 2025f. Disponível em: <https://developer.android.com/guide/components/foreground-services>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Guide to app architecture**. 2025i. Disponível em: <https://developer.android.com/topic/architecture?hl=pt-br>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **MediaPlayer**. 2026c. Disponível em: <https://developer.android.com/reference/android/media/MediaPlayer>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Notifications overview**. 2025g. Disponível em: <https://developer.android.com/develop/ui/views/notifications>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **RingtoneManager**. 2026d. Disponível em:

<https://developer.android.com/reference/android/media/RingtoneManager>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Room Persistence Library**. 2025h. Disponível em: <https://developer.android.com/training/data-storage/room?hl=pt-br>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **SharedPreferences Overview**. 2026e. Disponível em: <https://developer.android.com/reference/android/content/SharedPreferences>. Acesso em: 16 fev. 2026.

ANDROID DEVELOPERS. **Vibrator e VibrationEffect**. 2026f. Disponível em: <https://developer.android.com/reference/android/os/VibrationEffect>. Acesso em: 16 fev. 2026.

BANGOR, Aaron; KORTUM, Philip; MILLER, James. **Determining what individual SUS scores mean**: adding an adjective rating scale. *Journal of Usability Studies*, v. 4, n. 3, p. 114-123, 2009.

BROOKE, John. **SUS: a "quick and dirty" usability scale**. *Usability evaluation in industry*, v. 189, n. 194, p. 4-7, 1996. Disponível em: <https://hell.meiert.org/core/pdf/sus.pdf>. Acesso em: 21 fev. 2026.

GUEDES, Gildo de Araújo. **UML 2.5: Guia Prático**. 3. ed. São Paulo: Érica, 2018.

HAMINE, S. et al. Impact of mobile health technology on medication adherence: a systematic review and meta-analysis. **Journal of Medical Internet Research**, v. 17, n. 2, p. e52, 2015. Disponível em: <https://www.jmir.org/2015/2/e52/>. Acesso em: 29 jan. 2026.

JETBRAINS. **Kotlin Programming Language Documentation**. 2025. Disponível em: <https://kotlinlang.org/docs/home.html>. Acesso em: 16 fev. 2026.

MOREIRA, M. A. DE J. et al. Perfil dos medicamentos utilizados via oral e por sonda gastroenteral em um Serviço de Pronto Atendimento. **Revista da Escola de Enfermagem da USP**, v. 52, p. e03385, 2018.

PRESSMAN, Roger S.; MAXIM, Bruce R. **Engenharia de Software: uma abordagem profissional**. 9. ed. Porto Alegre: AMGH, 2021.

SABATÉ, E. **Adherence to long-term therapies**: Evidence for action. Geneva, Switzerland: World Health Organization, 2003.

SOMMERVILLE, Ian. **Engenharia de Software**. 9. ed. São Paulo: Pearson, 2011.

VIEIRA, N. **Estudo ressalta perigos de tomar remédio na hora errada**. Canaltech, 2022. Disponível em: <https://canaltech.com.br/saude/estudo-ressalta-perigos-de-tomar-remedio-na-hora-errada-233182/>. Acesso em: 29 jan. 2026.