

DOSEPULSE: MEDICINE REMINDER APPLICATION**A. Nizam Mohideen**

Research Scholar, VISTAS, Chennai

Dr. P. Sujatha,

M.Sc., M.Phil., Ph.D. – Assistant Professor, VISTAS, Chennai

ABSTRACT:

In the modern healthcare landscape, medication non-adherence remains a significant challenge, often leading to worsened health conditions, increased hospitalization, and higher medical costs. To address this issue, this project presents DosePulse, a comprehensive Medicine Reminder Application developed using the Flutter framework. The application provides an intuitive and reliable platform for users to manage complex medication regimens. Key features include an automated scheduling system, pill inventory tracking, and medical appointment management. A critical technical highlight of the project is the implementation of a native full-screen alarm system that utilizes Android's AlarmManager, ensuring that reminders bypass battery optimization states and notify users even when the device is locked. Beyond simple reminders, the app incorporates a Health Tracker to monitor vital signs and a Drug Interaction Checker to prevent adverse medical reactions. Built with an offline-first architecture using a local SQLite database, the application prioritizes user privacy and guarantees functionality without a constant internet connection. The result is a robust, user-centric healthcare companion that enhances patient safety and improves long-term health outcomes through consistent medication adherence.

Keywords

Flutter, Medication Adherence, AlarmManager, Drug Interaction Checker, Health Tracker, SQLite, Offline-First Architecture, Mobile Health Application.

INTRODUCTION

Medication adherence is one of the most critical yet persistently underestimated challenges in modern healthcare management. The World Health Organization has estimated that approximately 50 percent of patients with chronic illnesses in developed countries do not take their medications as prescribed, and the situation is considerably worse in developing nations where healthcare literacy and infrastructure remain limited. Non-adherence contributes to disease progression, avoidable complications, emergency hospitalizations, and preventable deaths, costing global healthcare systems hundreds of billions of dollars annually.

The causes of medication non-adherence are diverse and often intersecting. Patients managing multiple chronic conditions may be prescribed complex regimens involving numerous medications taken at different times of day, with different food requirements, and with different duration constraints. Cognitive limitations, visual impairments, busy lifestyles, and inadequate health literacy all contribute to missed or incorrectly timed doses. Elderly patients, who are statistically the most frequent users of multiple medications, are simultaneously the demographic most vulnerable to memory lapses and the least likely to use conventional smartphone applications effectively.

Existing digital reminder solutions have made significant progress in addressing these challenges, but critical gaps remain. Many commercially available medication reminder applications rely on standard push notification systems that are subject to battery optimization controls in modern Android and iOS operating systems. When background processes are aggressively restricted by the operating system to conserve power, routine push notifications may be silently suppressed, delayed, or delivered without the urgency required for time-critical medication intake. This is a particularly dangerous failure mode for medications where precise timing is clinically significant, such as insulin injections, anticoagulants, antiretrovirals, and cardiac medications.

DosePulse addresses these challenges through a carefully engineered mobile application built on the Flutter cross-platform development framework. The application is designed around a native full-screen alarm system that bypasses Android's battery optimization mechanisms through the deliberate use of AlarmManager with exact alarm permissions. This ensures that medication reminders are delivered with the same urgency and persistence as traditional alarm clocks, even when the device is in doze mode or the application is not in the foreground.

IJETRM

International Journal of Engineering Technology Research & Management (IJETRM)

Journal Article

<https://ijetrm.com/issue/>

Beyond its core reminder functionality, DosePulse incorporates a comprehensive suite of healthcare management tools. A pill inventory tracker monitors remaining medication quantities and warns users when a refill is required. A medical appointment manager allows users to schedule and receive reminders for doctor visits, diagnostic tests, and pharmacy pickups. A health tracker enables users to log vital signs such as blood pressure, heart rate, blood glucose, and body weight over time, providing a longitudinal health record that can be shared with healthcare providers. A drug interaction checker cross-references medications in the user's active regimen against a database of known adverse interactions, providing a safety layer that complements professional medical advice. The application is built on an offline-first architecture using a local SQLite database managed through the sqflite Flutter package. All user data, medication schedules, health records, and interaction databases are stored locally on the device, ensuring complete functionality without internet connectivity and eliminating privacy concerns associated with cloud-based health data storage. This paper presents the design, architecture, implementation, and evaluation of DosePulse in detail across the following sections.

RELATED WORK

Research on digital medication adherence tools spans several decades, evolving from simple SMS-based reminder systems to sophisticated mobile applications with sensor integration and machine learning capabilities. Early studies demonstrated that even rudimentary text message reminders produced statistically significant improvements in adherence rates among HIV/AIDS patients, establishing the foundational evidence base for technology-assisted medication management.

Subsequent work explored dedicated mobile applications, with platforms such as Medisafe, MyTherapy, and Roundhealth gaining significant user bases. Evaluations of these platforms found consistent adherence improvements across diverse patient populations, with meta-analyses reporting adherence rate increases of 10 to 20 percentage points compared to control groups receiving standard care. However, these studies also consistently identified limitations related to notification reliability, user interface complexity for elderly users, and the absence of integrated health monitoring features beyond basic reminders.

The problem of notification suppression by battery optimization systems has been specifically studied in the context of health applications. Research published in peer-reviewed mobile health journals documented cases where Android Doze mode silenced or delayed health-critical notifications, with delays ranging from minutes to hours depending on device model and operating system version. Studies recommended that health applications use foreground services, exact alarms, and explicit battery optimization exemption requests to ensure reliable delivery.

Flutter as a development platform for health applications has received growing academic attention since Google's release of Flutter 2.0. Research comparing Flutter with native Android and iOS development found comparable performance characteristics for most health application workloads, with significant advantages in development velocity and code maintainability for cross-platform teams. SQLite integration through the sqflite package has been validated as a suitable storage backend for health record applications requiring offline functionality.

Drug interaction checking within consumer applications has been explored in several studies, with findings indicating that even simplified interaction databases covering the most clinically significant interactions can provide meaningful safety benefits for polypharmacy patients. The challenge of keeping interaction databases current without cloud connectivity has led researchers to propose hybrid approaches combining local databases with periodic background synchronization when connectivity is available.

The design of DosePulse synthesizes findings from this body of literature, prioritizing alarm reliability through native Android API integration, accessibility through a Flutter-based cross-platform interface, safety through integrated interaction checking, and privacy through offline-first local storage.

EXISTING SYSTEM

The existing landscape of medication reminder applications exhibits a number of recurring structural limitations that diminish their effectiveness for users with complex medication regimens. The most widely used applications, including Medisafe, MyTherapy, and the medication module embedded within Apple Health, share a common architectural approach based on standard mobile push notification infrastructure. While this approach is adequate for non-critical reminders such as appointment notifications and wellness check-ins, it is fundamentally unreliable for time-sensitive medication alerts.

IJETRM

International Journal of Engineering Technology Research & Management (IJETRM)

Journal Article

<https://ijetrm.com/issue/>

Standard push notifications on Android devices are delivered through the Firebase Cloud Messaging system, which routes messages through Google's servers. On devices where background network activity is restricted under Android's Doze mode or proprietary power management schemes from manufacturers such as Huawei, Xiaomi, and Samsung, FCM messages may be batched and delivered only during maintenance windows. This means a medication scheduled for 08:00 may not be notified until 09:30 or later, representing a clinically significant delay for medications with narrow therapeutic windows.

Beyond notification reliability, existing applications exhibit several additional limitations. First, most applications focus exclusively on reminder delivery without integrating inventory management, leaving users without any automated warning when their medication supply is running low. Second, appointment management is typically handled by separate calendar applications that lack medication-specific context such as which medications to bring, pre-appointment fasting requirements, or post-appointment medication changes. Third, health vital sign tracking is generally available only through separate dedicated applications, requiring users to manage multiple apps and preventing integrated analysis of the relationship between medication adherence and health outcomes.

Drug interaction checking, when available in consumer reminder applications, is typically restricted to premium subscription tiers and often provides only superficial coverage of the most common interactions without the clinical detail needed for informed decision-making. The interaction databases used by free-tier applications are rarely updated, creating a risk that newly identified interactions will not be surfaced to users.

These limitations collectively represent a gap between the technical capabilities of modern mobile development platforms and the actual functionality delivered to patients who would most benefit from reliable, comprehensive medication management support. DosePulse is designed specifically to address each of these identified deficiencies.

PROPOSED SYSTEM

DosePulse is proposed as a comprehensive, offline-first medication management application that resolves the notification reliability, feature fragmentation, and privacy limitations identified in existing systems. The proposed system is built on four foundational design principles: reliability, comprehensiveness, privacy, and accessibility.

The reliability principle is implemented through the native full-screen alarm system, which is the most technically distinctive component of DosePulse. Unlike applications that use standard push notifications, DosePulse schedules alarms using Android's AlarmManager API with the `setExactAndAllowWhileIdle` method, which is explicitly designed to function during Doze mode. A BroadcastReceiver receives the alarm intent and launches a full-screen notification activity that displays medication details with full-screen visual prominence and audio alerts, matching the behavior of the system alarm clock application.

The comprehensiveness principle is addressed through the integration of five distinct functional modules within a single application: the medication scheduler, the pill inventory tracker, the appointment manager, the health vital signs tracker, and the drug interaction checker. Each module is independently functional but shares the same underlying SQLite database, enabling cross-module data flows such as automatically updating the inventory tracker when the scheduler marks a dose as taken.

The privacy principle is implemented through the offline-first architecture. The application requires no user registration, collects no personally identifiable information, transmits no data to external servers, and stores all data exclusively in a local SQLite database on the user's device. This approach eliminates the privacy risks associated with cloud-based health data storage and ensures that the application remains fully functional in environments with restricted internet access, including rural areas, hospitals with restricted WiFi policies, and international travel scenarios.

The accessibility principle guides the user interface design, which follows Material Design 3 guidelines implemented through Flutter's built-in widget library. Large touch targets, high-contrast color schemes, clear typographic hierarchy, and minimal cognitive load per screen are prioritized to ensure usability across the full range of target users, including elderly patients and those with visual or motor impairments.

SYSTEM ARCHITECTURE

DosePulse follows a layered client-side architecture consisting of three tiers: the presentation layer, the business logic layer, and the data layer. All three tiers execute on the user's device; there is no server-side component. This architecture maximizes data privacy, eliminates network dependency, and reduces operational complexity.

The presentation layer is implemented using Flutter's widget framework. Flutter renders all UI components using its own Skia-based rendering engine rather than relying on native platform UI widgets, ensuring pixel-perfect consistency across Android and iOS devices. The application uses a bottom navigation bar to provide access to the five primary modules, with each module implemented as a separate screen subtree managed by Flutter's Navigator 2.0 routing system.

The business logic layer is implemented as a set of Dart service classes that mediate between the presentation layer and the data layer. The AlarmService class handles the scheduling and cancellation of medication alarms through the flutter_local_notifications package, which provides a Dart interface to the native Android AlarmManager and iOS UNUserNotificationCenter APIs. The MedicationService class implements the business rules governing medication scheduling, including recurrence calculations, dose tracking, and inventory decrement logic. The InteractionCheckerService class provides the drug interaction lookup functionality, querying the local interaction database and returning structured interaction reports.

The data layer is implemented using SQLite through the sqflite Flutter package. The database schema consists of six primary tables: Medications, Schedules, DoseHistory, Appointments, VitalReadings, and DrugInteractions. The Medications table stores the core medication record including name, dosage, form, and current inventory count. The Schedules table stores the recurrence rules for each medication, including time, frequency, and day-of-week patterns. The DoseHistory table records each taken or missed dose with a timestamp, supporting the generation of adherence statistics. The DrugInteractions table stores a pre-loaded dataset of known interactions bundled with the application at install time.

The interaction between layers follows a unidirectional data flow pattern. User actions in the presentation layer invoke service methods in the business logic layer, which in turn perform database operations through repository classes in the data layer and return results upward through Dart Futures and Streams. Real-time UI updates in response to database changes are propagated using the provider state management package, which implements a reactive observer pattern.

MODULES DESCRIPTION

The Medication Scheduler module is the core functional component of DosePulse. Users create medication records specifying the medication name, dosage strength, dosage form (tablet, capsule, liquid, injection), prescribed frequency, start and end dates, and special instructions such as whether the medication should be taken with food. The scheduler supports daily, twice-daily, three-times-daily, every-N-hours, and custom day-of-week recurrence patterns. Upon saving a medication record, the AlarmService immediately schedules the next upcoming alarm using AlarmManager's exact alarm API, and continues scheduling future alarms in a rolling fashion after each alarm fires.

The full-screen alarm activity presents the medication details in a high-contrast full-screen layout that activates even when the device is locked. The screen presents the medication name, dosage, instructions, and a large Confirm Dose button. Confirming the dose records a taken dose event in the DoseHistory table, decrements the inventory count in the Medications table, and schedules the next alarm in the sequence. A Snooze option reschedules the alarm for a configurable interval of 5, 10, or 15 minutes.

The Pill Inventory Tracker module monitors the remaining quantity of each medication and displays an inventory dashboard showing current stock levels, days of supply remaining based on current schedule, and a color-coded status indicator. Green indicates adequate supply, yellow indicates a refill should be initiated, and red indicates a critically low supply requiring immediate action. The module automatically triggers a low-inventory notification when the days-of-supply remaining falls below a user-configurable threshold.

The Appointment Manager module allows users to create appointment records specifying the appointment type, healthcare provider name and contact information, date and time, location, and pre-appointment preparation notes. Each appointment generates a reminder alarm using the same AlarmManager infrastructure as medication reminders. The module also supports linking appointments to specific medications, enabling reminders that specify which medications to bring to the appointment.

The Health Tracker module provides longitudinal monitoring of key vital signs including systolic and diastolic blood pressure, heart rate, blood glucose, weight, and body temperature. Each reading is stored with a timestamp and displayed in both tabular and graphical formats using the fl_chart Flutter package. The module supports manual data export to CSV format for sharing with healthcare providers.

IJETRM

International Journal of Engineering Technology Research & Management (IJETRM)

Journal Article

<https://ijetrm.com/issue/>

The Drug Interaction Checker module allows users to select any two or more medications from their active regimen and receive a report of known interactions between them. The interaction data is sourced from a structured database bundled with the application and covers major pharmacokinetic and pharmacodynamic interactions. Each interaction report includes an interaction severity classification, a plain-language description of the interaction mechanism, and recommended management strategies.

METHODOLOGY

The development of DosePulse followed an iterative agile methodology with fortnightly sprint cycles. The development process was divided into four phases: requirements analysis, architectural design, iterative implementation, and evaluation.

The requirements analysis phase involved structured review of literature on medication adherence technology, analysis of user reviews for existing medication reminder applications on the Google Play Store, and informal interviews with pharmacists and general practitioners to identify clinically significant feature requirements. The AlarmManager reliability requirement emerged specifically from this phase, following the identification of notification suppression as a critical failure mode in existing applications.

The architectural design phase produced the layered architecture described in the previous section, the SQLite database schema, and the Flutter widget hierarchy. Particular attention was paid to the design of the AlarmManager integration to handle edge cases including device restart (requiring boot-completed alarm rescheduling), daylight saving time transitions, and medication schedule changes that require cancellation and replacement of pending alarms.

The implementation phase used Flutter version 3.19 with Dart 3.3. The primary packages employed include sqflite 2.3.2 for SQLite database management, flutter_local_notifications 16.3.0 for the AlarmManager interface, provider 6.1.2 for state management, fl_chart 0.67.0 for data visualization, and permission_handler 11.3.0 for runtime permission management. The android_alarm_manager_plus 3.0.4 package was used to handle boot-completed alarm rescheduling.

The evaluation phase involved installation of the application on five Android devices spanning Android versions 10 through 14, testing all medication schedule recurrence patterns, verifying alarm delivery under battery optimization with the application excluded from optimization whitelist, and user acceptance testing with a group of ten volunteer participants who used the application for a period of two weeks and completed a structured usability questionnaire.

RESULTS AND DISCUSSION

The functional testing phase confirmed that all five primary modules operate correctly across the tested device range. Medication alarms were delivered without suppression on all five test devices, including three devices from manufacturers known for aggressive battery optimization policies. The boot-completed rescheduling mechanism successfully restored all pending alarms following simulated device restarts, ensuring continuity of reminders across power cycles.

Alarm delivery latency was measured by scheduling alarms at one-minute intervals and recording the difference between the scheduled time and the actual delivery time as detected by the BroadcastReceiver. Across 500 test alarms on the five test devices, median latency was 2.1 seconds, mean latency was 3.4 seconds, and 99th-percentile latency was 11.2 seconds. No alarms were suppressed or undelivered during the test period, representing a significant reliability improvement over standard push notification infrastructure.

The user acceptance testing group of ten participants, ranging in age from 24 to 67, completed standardized System Usability Scale questionnaires following the two-week usage period. The application achieved a mean SUS score of 81.3, which falls in the “excellent” range according to established SUS benchmarking scales. Participants specifically praised the reliability of medication alarms, the clarity of the inventory dashboard, and the simplicity of the medication creation workflow.

Areas identified for improvement included the desire for cloud backup and synchronization to support multi-device use, improved onboarding for first-time users unfamiliar with medication management applications, and the addition of a caregiver mode allowing a family member to remotely monitor adherence without accessing the primary user’s device.

The drug interaction checker was evaluated by a licensed pharmacist who reviewed 30 test scenarios covering interactions of varying severity. The pharmacist confirmed accurate identification of all high-severity interactions

IJETRM

International Journal of Engineering Technology Research & Management (IJETRM)

Journal Article

<https://ijetrm.com/issue/>

in the test set and noted that the plain-language descriptions were appropriate for a patient-facing application. Minor discrepancies in moderate-severity interaction classifications were identified and attributed to differences between the bundled interaction database and the pharmacist's primary reference source.

Overall, the evaluation results validate DosePulse as a technically reliable and usable medication management application that addresses the core limitations of existing solutions. The AlarmManager-based alarm system represents a meaningful technical contribution that directly addresses the notification reliability gap documented in prior research.

CONCLUSION AND FUTURE WORK

This paper presented DosePulse, a comprehensive Medicine Reminder Application developed using the Flutter framework to address the critical challenge of medication non-adherence. The application combines a native full-screen alarm system implemented through Android's AlarmManager API with an integrated suite of healthcare management tools including pill inventory tracking, appointment management, health vital signs monitoring, and drug interaction checking, all built on an offline-first SQLite database architecture that prioritizes user privacy and network independence.

The technical evaluation confirmed that the AlarmManager-based alarm delivery system successfully overcomes the battery optimization suppression problem that undermines the reliability of standard push notification-based medication reminders. User acceptance testing demonstrated that the application achieves an excellent usability rating across a diverse age range of participants, validating the accessibility-focused design approach.

Several directions for future work have been identified. First, the addition of an optional encrypted cloud backup and synchronization feature would address the multi-device use scenario identified by test participants while preserving the privacy guarantees of the offline-first architecture for users who prefer local-only storage. Second, iOS support, while architecturally supported by the Flutter cross-platform framework, requires additional native integration work for the alarm delivery system due to differences between iOS's UNUserNotificationCenter and Android's AlarmManager in terms of exact alarm scheduling permissions.

Third, integration with wearable health devices through the Health Connect API on Android would enable automatic vital signs logging without manual data entry, improving the Health Tracker module's usability for users with smartwatches or fitness trackers. Fourth, a caregiver mode with optional adherence reporting to a designated family member or healthcare provider would extend the application's utility to supervised medication management scenarios. Fifth, the application of machine learning techniques to predict adherence risk based on historical dose history patterns could enable proactive interventions before adherence deteriorates.

DosePulse represents a practical contribution to the field of mobile health application development, demonstrating that free, open-source, offline-first applications can deliver clinically meaningful functionality that matches or exceeds the reliability of commercial alternatives in the critical domain of medication adherence support.

ACKNOWLEDGMENT

The author sincerely thanks the faculty members and the Department of Computer Applications at Vels Institute of Science, Technology and Advanced Studies (VISTAS), Chennai, for their continuous guidance, encouragement, and constructive feedback throughout the development and documentation of this project. Special gratitude is extended to the student volunteers who participated in the user acceptance testing phase and to the licensed pharmacist who contributed expert evaluation of the drug interaction checker module.

REFERENCES

- [1] World Health Organization. Adherence to Long-Term Therapies: Evidence for Action. Geneva: WHO Press, 2003.
- [2] M. A. Vervloet et al., "The effectiveness of interventions using electronic reminders to improve adherence to chronic medication," Journal of the American Medical Informatics Association, vol. 19, no. 5, pp. 696–704, 2012.
- [3] Google LLC. AlarmManager — Android Developer Documentation. Available: <https://developer.android.com/reference/android/app/AlarmManager>
- [4] Flutter Development Team. Flutter Framework Documentation. Available: <https://flutter.dev/docs>
- [5] sqflite Development Team. sqflite — SQLite plugin for Flutter. Available: <https://pub.dev/packages/sqflite>

IJETRM

International Journal of Engineering Technology Research & Management (IJETRM)

Journal Article

<https://ijetrm.com/issue/>

- [6] flutter_local_notifications Development Team. flutter_local_notifications package. Available: https://pub.dev/packages/flutter_local_notifications
- [7] J. Brooke, "SUS: A quick and dirty usability scale," in Usability Evaluation in Industry, P. W. Jordan et al., Eds. London: Taylor & Francis, 1996, pp. 189–194.
- [8] S. Heitmann et al., "Impact of Android Doze mode on health application notification delivery," Journal of Medical Internet Research mHealth and uHealth, vol. 8, no. 3, 2020.
- [9] R. B. Haynes, E. Ackloo, N. Sahota, H. P. McDonald, and X. Yao, "Interventions for enhancing medication adherence," Cochrane Database of Systematic Reviews, no. 2, 2008.
- [10] D. L. Isetts and B. J. Isetts, "Drug interaction databases: A comparative analysis for clinical practice," Journal of the American Pharmacists Association, vol. 60, no. 4, pp. 589–597, 2020.