

**Mobile Application Development**

**PROJECT REPORT : MEDICATION REMINDER APP**

**Submitted to: MAM MARIAM**

**submitted by:**

**Ghulam hussain (21SW165)**

**MUHAMMad SHARIQUE (21SW148)**

**Kehfulwara ali (21SW012)**

**Medication Reminder App Report**

**1. Real World Problem Identification**

**Problem:**

Many people, particularly the elderly or those with chronic illnesses, often forget to take their medications on time, which can lead to serious health issues, hospitalizations, or ineffective treatment. Keeping track of medication schedules manually can be cumbersome and prone to errors, especially when there are multiple medications with different dosing times.

**2. Proposed Solution**

The **Medication Reminder App**  addresses this problem by allowing users to schedule and manage their medication routines effectively. The app ensures that users never miss a dose, helping them maintain their treatment plans properly. Key features include:

**- Scheduling Medication Reminders** : Users can easily input the name of the medication, dosage, and the time(s) they need to take it.

**- Recurring Alarms:** Users can set alarms that recur daily, weekly, or on custom schedules depending on their medication routine.

**- Tracking Medication History :** The app logs each time a user confirms that they’ve taken their medication, allowing them to track their adherence over time.

**3. Responsive User Interfaces (Screenshots of your app on different screens & platforms)**

The app is designed to work seamlessly on both small and large devices, including smartphones and tablets. By using Flutter, we ensure a consistent and responsive user experience across all platforms. The UI adjusts dynamically to different screen sizes, ensuring optimal usability for each user. Below are examples of how the app adapts:

**Small Screen (Smartphones):**

**- Home Screen:** Displays an overview of today's medication reminders.

**- Add Medication Screen:** Allows users to schedule new medications with dosage, frequency, and start time.

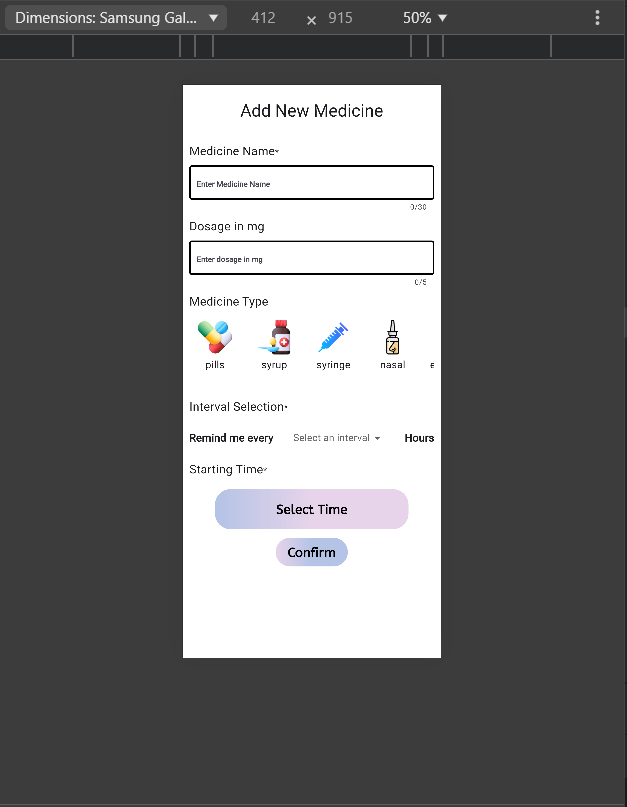
**- History Screen:** Shows a log of the user’s medication history (i.e., past doses taken or missed).

**Large Screen (Tablets):**

**- Home Screen**: The interface adapts by showing more information at once (e.g., a split view of current reminders and medication history).

**- Add Medication Screen:** Larger buttons and fields to take advantage of the additional screen real estate.

**Visual reference**: App running on different devices shown below

A screenshot of a phone

Description automatically generated

**4. Data Storage (With Justification for Using Shared Preferences)**

For the Medication Reminder app, **Shared Preferences** is used for local data storage, offering a lightweight, key-value mechanism suitable for storing simple data structures and configurations directly on the device.

**Shared Preferences (Local Storage):**

* **Justification**: Shared Preferences provides efficient local storage for structured data, like medication schedules, without needing an internet connection. This ensures that users can access medication information and receive timely reminders even in low-connectivity areas.
* **Usage**: Medication details and schedules are stored as JSON-encoded strings in Shared Preferences, ensuring quick access and data persistence across app sessions, providing reliable reminders for users.

**5. APIs/Packages/Plug-ins (With Justifications for Usage)**

Several Flutter packages enhance the app’s functionality and streamline development:

* **rxdart (Reactive Programming)**:
  + **Justification**: rxdart manages the app’s state reactively, updating the UI in real-time whenever the medication list changes.
  + **Usage**: The BehaviorSubject from rxdart is used to maintain the active medication list, notifying listeners of changes to ensure the UI displays the latest data.
* **shared\_preferences (Local Storage)**:
  + **Justification**: This package supports easy, persistent storage of basic data, ideal for medication schedules and settings, even across sessions.
  + **Usage**: Shared Preferences stores medication data locally as JSON, ensuring data reliability and accessibility whenever the user reopens the app.
* **flutter\_local\_notifications (Local Notifications)**:
  + **Justification**: This package enables daily reminders for users to take their medications, even when the app is inactive, using recurring notifications.
  + **Usage**: Unique notification IDs are assigned for each medication schedule, supporting daily reminders and improving adherence to schedules.

**6. Issues and Bugs Encountered and Resolved During Development**

* **Issue**: App Crashed on Scheduling Multiple Recurring Notifications.
  + **Cause**: Duplicate notification IDs led to conflicts.
  + **Solution**: Assigned unique IDs for each notification, preventing conflicts.
* **Issue**: Inconsistent Notification Alerts on Certain Android Devices.
  + **Cause**: Background restrictions on some devices prevented timely notifications.
  + **Solution**: Integrated a background service to ensure reliable notifications.
* **Issue**: App Layout Distortion on Larger Screens.
  + **Cause**: Hardcoded dimensions caused layout issues on larger screens.
  + **Solution**: Used MediaQuery and LayoutBuilder to adapt the UI to different screen sizes for responsive design.
* **Issue**: Slow Performance When Displaying Extensive Medication History.
  + **Cause**: Rendering all history at once led to performance lags.
  + **Solution**: Optimized rendering with ListView.builder, improving app speed and responsiveness.
* **Issue**: Input Validation and Entry Errors.
  + **Cause**: Invalid and duplicate entries allowed, especially for medication names.
  + **Solution**: Used EntryError enum for input validation, managing null, duplicate, and incorrect entries effectively.