

Functional Requirements for Lightweight Music Equipment Cleaning Reminder App

CORE FUNCTIONAL REQUIREMENTS

1. Task Management

- Users can add, edit, and delete musical instruments or equipment.
- Each item can have a custom reminder interval (e.g. every 7 days).
- Users can press a "Task Completed" button for each instrument after cleaning.
- App logs each completion event with a timestamp.

2. Reminders

- App sends reminders only at user-defined intervals per item.
- Reminders use Android notification channels (silent or sound-based as preferred).
- No unnecessary background activity--runs only when reminders are due.

3. Data Logging

- App maintains an indefinite log of all cleaning activities.
- Log includes: instrument name, date/time cleaned, optional notes or photos.
- Users can view, filter, and sort this history within the app.

4. Data Export

- Users can export cleaning logs in CSV format.
- Optionally generate summary reports (e.g. weekly/monthly history).
- Export function is available entirely offline.

STORAGE AND SYNC

5. Local Storage

- All user data (equipment list, reminder intervals, cleaning logs, photos) is stored locally by default.
- Data is kept indefinitely unless explicitly deleted by the user.

6. Optional Cloud Sync

- Users can opt into cloud sync/backup (e.g. Google Drive or a simple cloud database).

- App continues to function fully offline without cloud features.

USER INTERFACE

7. GUI Design

- App includes a graphical interface with a minimalist, classy aesthetic.
- Clean layout with intuitive access to:
 - Task list
 - Completion button
 - Reminder settings
 - History log
 - Export options

8. User Input

- Users can:
 - Add/edit instrument details (name, category, optional photo)
 - Set custom reminder intervals per item
 - View/edit past task logs
 - Export data via a simple menu or button

PERFORMANCE AND POWER USE

9. Power Optimization

- App performs no background processing unless a reminder is scheduled.
- Avoids wake locks, polling, or frequent wakeups.
- All reminders and tasks use battery-friendly scheduling.

PRIVACY AND DATA HANDLING

10. Sensitive Data

- Users may optionally provide:
 - Basic demographic info
 - Photos of instruments

- No encryption or anonymization is applied.
- User data is stored in plain format, privately on device unless synced manually.

ARCHITECTURE AND MAINTAINABILITY

11. Modular Codebase

- App is structured for modular updates (e.g. adding export formats, themes, stats).
- Built to be stable and self-contained for low-maintenance.
- Designed for single-developer maintenance with clean, well-commented code.

12. Compatibility

- Minimum Android Version: 8.0 (API 26)
- Built with modern, stable Android libraries (e.g. Room for local DB, WorkManager for scheduled reminders).

DEPLOYMENT AND DISTRIBUTION

13. Distribution

- App will be distributed privately, via APK or sideloading.
- No Play Store publishing required.
- No integration with Google billing, ads, or compliance APIs.