

Jar

Product Requirements Document

Version 1.0.0 Release Specification

Document Version: 1.0

Target Release: v1.0.0

Author: Erich Benjamin Hubert

Last Updated: January 2026

Repository: github.com/Enrichment2/jar

Table of Contents

1. Executive Summary
2. Core Principles
3. Feature Requirements for v1.0.0
4. Technical Requirements
5. UI/UX Standards
6. Release Criteria
7. Out of Scope for v1.0.0
8. Post-1.0 Roadmap

1. Executive Summary

Jar is a local-first, privacy-focused productivity application designed to help users organize their thoughts, tasks, and personal information without compromising their data to cloud services or third-party providers.

The core metaphor of the app is a "jar" — a transparent container where users store their information. Unlike traditional productivity apps that fragment user data across multiple services, Jar provides a unified, local-first experience that puts users in complete control of their data.

Vision Statement

To create the most user-respecting productivity app that proves you don't need to sacrifice privacy for functionality. Jar will be the app users trust with their most personal thoughts and important tasks, knowing their data never leaves their device.

Target Users

- Privacy-conscious individuals who distrust cloud-based note apps
- Users seeking a unified productivity solution instead of multiple fragmented apps
- People who want full control and ownership of their personal data
- Open-source enthusiasts who value transparency in their tools

2. Core Principles

These principles are non-negotiable and must be upheld in every feature and design decision.

Local-First

All data is stored locally on the user's device. The app must function fully offline with zero internet requirement. No telemetry, analytics, or data collection of any kind.

Privacy by Design

User data never leaves the device unless explicitly exported by the user. No accounts, no sign-ups, no tracking. The app knows nothing about its users.

Open Source (AGPL-3.0)

Complete transparency in how the app works. Users can audit the code, contribute improvements, and fork if desired. No proprietary components or hidden functionality.

User Ownership

Users own their data completely. Export and import in open formats (JSON, Markdown). No lock-in, no walled gardens. Data portability is a first-class feature.

Simplicity Over Features

Do fewer things excellently rather than many things poorly. Every feature must earn its place through genuine utility. Complexity is the enemy.

Beautiful by Default

UI/UX quality is not optional. The app must look and feel premium. Material You integration ensures it feels native to each user's device.

3. Feature Requirements for v1.0.0

The following features must be fully implemented and polished before the v1.0.0 release.

3.1 Notes Module

Status: Partially Complete

Feature	Status	Description
Create, edit, and delete notes	Complete	Users can create notes with title and body, edit existing notes, and delete notes.
Rich text editor	Required	Bold, italic, bullet lists, numbered lists, headings. Markdown support preferred.
Search functionality	Complete	Real-time search across note titles and content, case-insensitive.
Tags system	Complete	Multiple tags per note, tag filtering, tag colors, inline tag creation.
Note organization	Required	Sort by date created, date modified, alphabetical. Pin important notes to top.
Note sharing	Required	Export individual notes as plain text, Markdown, or PDF.

3.2 Tasks / To-Do Module

Status: Not Started

The tasks module is a separate "jar" from notes, providing dedicated task management:

- Create tasks with title and optional description
- Mark tasks as complete/incomplete with checkbox
- Due dates with optional time
- Task priority levels (High, Medium, Low, None)
- Recurring tasks (daily, weekly, monthly, custom)
- Task lists/projects for grouping related tasks
- Overdue task highlighting
- Sort by due date, priority, creation date, or manual order
- Quick-add from main screen
- Completed tasks archive (with option to permanently delete)

3.3 Reminders Module

Status: Not Started

Standalone reminders system that integrates with Android's notification system:

- One-time reminders at specific date/time
- Recurring reminders (daily, weekly, monthly, yearly, custom)
- Location-based reminders (when arriving/leaving a place) — v1.1 consideration
- Notification customization (sound, vibration, priority)
- Snooze functionality (5 min, 15 min, 1 hour, custom)
- Link reminders to notes or tasks (optional)
- Reminder history/log
- Do Not Disturb schedule integration

3.4 Data Management

Status: Not Started

Complete data portability and backup system:

- Export all data to JSON (full backup)
- Export all data to ZIP (JSON + any attachments)
- Import from JSON backup
- Export individual modules (just notes, just tasks, etc.)
- Export notes to Markdown files
- Automatic local backup scheduling (daily/weekly)
- Backup to user-selected directory
- Clear all data option (with confirmation)
- Data statistics (note count, task count, storage used)

3.5 Settings & Customization

Status: Partial (Theme only)

- Theme: System default, Light, Dark
- Material You dynamic colors (Complete)
- Default view on app launch (Notes, Tasks, or last used)
- Date/time format preferences
- Notification preferences per module
- Default sort order preferences

- Confirm before delete toggle
- About page with version, licenses, links
- Quick tutorial / onboarding (skippable)

4. Technical Requirements

4.1 Platform & Compatibility

Platform	Android (primary), Linux desktop (future)
Minimum Android Version	Android 8.0 (API 26)
Target Android Version	Android 15 (API 35)
Architecture	MVVM with Repository pattern
Language	Kotlin 100%
Database	Room (SQLite)
UI Framework	Material Design 3 with Material You
Async Operations	Kotlin Coroutines + Flow
Dependency Injection	Consider Hilt for v1.0

4.2 Performance Requirements

- App cold start: < 1 second on mid-range devices
- Search results: < 100ms for 1000+ notes
- Database operations: Non-blocking, all on background threads
- Memory footprint: < 100MB in typical use
- APK size: < 15MB
- Battery: Minimal background drain (no services running when app closed)
- Smooth scrolling: 60fps in all lists

4.3 Security Requirements

- All data stored in app-private directory (not accessible to other apps)
- No network permissions requested
- No analytics or crash reporting SDKs
- Exported backups: Offer optional encryption with user-provided password
- Secure deletion: Overwrite data when permanently deleted
- No clipboard snooping or background data access

5. UI/UX Standards

5.1 Design Principles

Consistency: Same patterns and interactions throughout the app. A user who learns notes can immediately use tasks.

Clarity: Clear visual hierarchy. Important actions are obvious. No hidden functionality.

Responsiveness: Immediate feedback for all interactions. Loading states for any operation > 100ms.

Accessibility: Minimum touch target 48dp. Sufficient color contrast. Screen reader support.

Forgiveness: Undo for destructive actions. Confirm before permanent deletion. Easy data recovery.

5.2 Navigation Structure

Bottom navigation bar with three primary destinations:

- **Notes** — Note list with search and tag filter
- **Tasks** — Task list organized by due date or project
- **More** — Settings, Reminders, Data Management, About

5.3 Material You Integration

The app must feel native to each user's device by fully supporting Material You dynamic color theming. The color scheme should derive from the user's wallpaper (on supported devices) or fall back to the default purple theme. All interactive elements, accents, and surfaces must use the dynamic color tokens.

5.4 Empty States

Every list must have a thoughtful empty state that guides the user. Examples:

- Notes: "Your jar is empty. Tap + to add a note."
- Tasks: "All clear! Tap + to add a task."
- Search results: "No notes match your search."

6. Release Criteria for v1.0.0

All of the following must be true before v1.0.0 can be released:

6.1 Feature Complete

- Notes module: All features in section 3.1 implemented and tested
- Tasks module: All features in section 3.2 implemented and tested
- Reminders module: Core features implemented (location-based can wait)
- Data export/import: Full backup and restore working
- Settings: All preferences functional

6.2 Quality Bar

- Zero known crash bugs
- Zero data loss bugs
- All UI elements properly aligned and themed
- Performance requirements met (section 4.2)
- Tested on at least 3 different Android versions (10, 12, 14)
- Tested on at least 2 different screen sizes

6.3 Documentation

- README with clear description and screenshots
- Privacy policy (even though we collect nothing)
- License file (AGPL-3.0)
- Basic user guide or FAQ
- Changelog for all versions

6.4 Distribution

- Signed release APK available on GitHub Releases
- F-Droid submission prepared (metadata, fastlane structure)
- Play Store listing prepared (optional, user preference)

7. Out of Scope for v1.0.0

The following features have been explicitly deferred to future releases:

Cloud Sync

Contradicts core local-first principle. May consider optional encrypted P2P sync post-1.0.

AI Features

No AI auto-tagging, summarization, or smart suggestions. May reconsider with on-device models only.

Voice Notes

Audio recording and playback deferred to v1.1 or v1.2.

Budgeting Module

Will remain a separate app (Based Budgeting). Integration considered for v2.0.

Widgets

Home screen widgets deferred to v1.1.

Wear OS Support

Smartwatch companion app is a v2.0+ consideration.

iOS Version

Android-first. iOS port considered after Android reaches v1.0 stability.

Web Version

No web app planned. Desktop Linux version has higher priority.

Collaboration

Sharing/collaboration features contradict privacy-first approach.

Calendar Integration

System calendar sync deferred. Standalone reminders first.

8. Post-1.0 Roadmap

Tentative feature roadmap after v1.0.0 release:

Version 1.1 — "Polish"

- Home screen widgets (quick note, task list, upcoming reminders)
- Voice notes with transcription (on-device)
- Improved rich text editor
- Note templates
- App shortcuts for quick actions

Version 1.2 — "Power User"

- Advanced search with filters (date range, tags, has attachments)
- Bulk operations (multi-select delete, tag, move)
- Custom tag icons
- Task templates and checklists
- Keyboard shortcuts (for tablets/desktop)

Version 2.0 — "Desktop"

- Linux desktop application
- Local network sync between devices (encrypted P2P)
- Plugin system for community extensions
- Optional budgeting module integration
- Calendar view for tasks and reminders

This roadmap is tentative and subject to change based on user feedback and real-world usage patterns.