

NoteMark Milestone 4 Requirements

This document outlines the requirements for Milestone #4 of the NoteMark app. Each milestone comes with a new requirements document and an updated Figma file. You can find all current milestones in the [members area](#).

The mockups show the exact look and colors of each UI element. The app uses a single theme, with no support for light or dark modes.

This document gives you a general idea of what the app should do. You are free to choose how to implement specific behaviors (e.g., how to display a loading indicator or handle errors).

You can find the mockups for **NoteMark here**:

<https://www.figma.com/design/3eLOJSbYh6HVwIYVcqR5YE/NoteMark?node-id=9511-15185>

Milestone #4 Goal

In this milestone, we transition the app to a **fully offline-first architecture**, add background syncing and conflict resolution, and expand the **Settings Screen** with more controls over sync behavior.

The goal is to give users a **reliable and transparent note-taking experience**, even when offline.

Icons

All icons for the app can be Material design icons or taken from the mockups as SVG (in case an equivalent Material icon doesn't exist).

Adaptive Layouts

All screens must look good and work well on phones and tablets, in both portrait and landscape. All UI requirements apply to each device, and unique differences will be explicitly mentioned.

Technical Requirements

▼ Note List Screen (updated from Milestone #3)

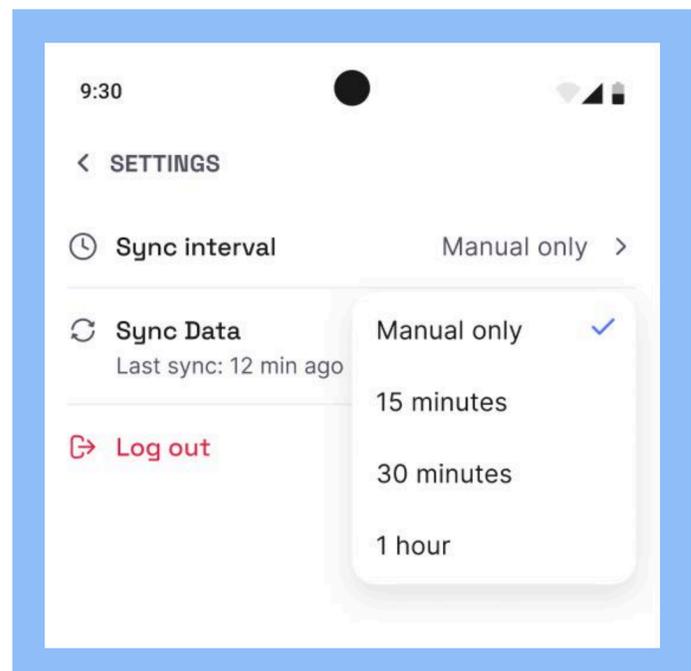
- The icon (e.g., cloud_off) is shown **only when the app is offline**
- Hidden when the device has internet access



▼ Settings Screen (extended)

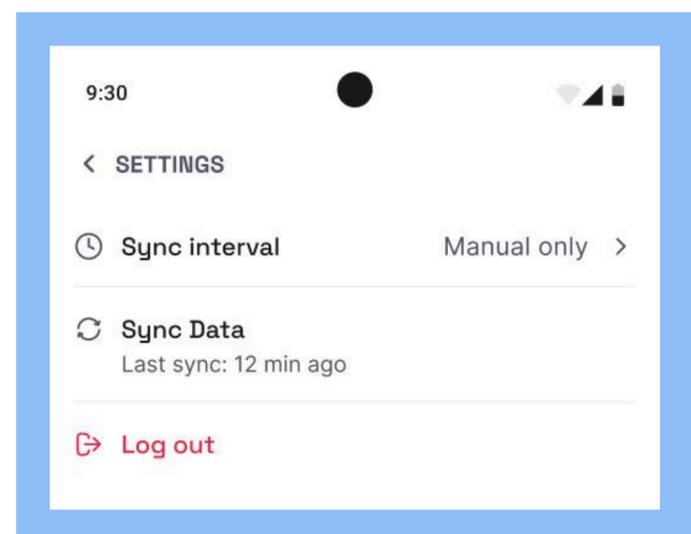
Sync Interval

- New settings item: **"Sync interval"**
- Current value (e.g., "Manual only") is shown on the right
- Tapping opens a dropdown with options:
 - "Manual only" (*default*)
 - "15 minutes"
 - "30 minutes"
 - "1 hour"
- The selected value determines the background sync interval
- Sync jobs must:
 - Run reliably in the background
 - Resume after device reboot, using a Broadcast Receiver



Sync Data

- Settings item: **"Sync Data"**
- This action performs a two-way sync between local and remote data:
 - It executes any **pending local changes** (e.g. created, edited, or deleted notes) and sends them to the backend
 - It also **fetches the latest remote data**, including changes or deletions that occurred on the server



- Below the title, show the last successful sync timestamp:
 - If no successful sync has occurred yet, display: “Never synced”
 - If the last sync was **less than 5 minutes ago**, show: “Just now”
 - If the sync occurred **within the past 7 days**, show a relative time:
 - e.g., “12 minutes ago”, “3 hours ago”, “2 days ago”
 - If the sync occurred **more than 7 days ago**, show the timestamp in full format: **dd MMM yyyy, HH:mm**
- Tapping the item triggers an immediate two-way sync
- While syncing, show a loading indicator
 - Recommended: *CircularProgressIndicator* centered on the screen

Logout

- Behavior matches Milestone 3, with the following enhancements:
 - Red icon + “Log out” text
 - If the device is **offline**, show a message such as: “*You need an internet connection to log out.*”
 - If the device is **online**, proceed with logout:
 - Perform a full two-way sync with the remote server.
 - Clear the local note database
 - Clear session tokens (e.g., shared prefs or data store)
 - Call **POST /api/auth/logout**
 - Clear the app backstack and return to the Login Screen

▼ Offline-First Architecture (new)

- All note changes (create, edit, delete) must:
 - Be performed **locally first**
 - Be added to a **sync queue**
- Sync must run in the background once the device is online again
- Sync behavior must ensure:
 - Reliable background execution
 - Automatic restart after device reboot (e.g., using a Broadcast Receiver)
 - Tracking and persistence of pending changes
 - Automatic retries if previous syncs failed

▼ Conflict Resolution: Last-Write-Wins (new)

- If both local and remote versions of a note have changed:
 - The version with the **newer lastEditedAt timestamp** wins
- No merge dialogs — conflict resolution is automatic

▼ Automatic Note Saving

- In **Edit Mode**, the “Save Note” button is removed
- Changes are saved **automatically while the user is typing**
 - To avoid excessive updates, auto-saving should be debounced — for example, 500–1000 ms after the user stops typing
- After the user finishes editing (i.e. when returning to View Mode):
 - the **lastEditedAt** field is updated to match the time editing was completed
- Display rules:
 - If the note was edited less than 5 minutes ago, display “Just now”
 - Otherwise, show the timestamp in standard format:
dd MMM yyyy, HH:mm

