

PROJECT ABSTRACT

Exam Syllabus Progress Tracker Android App

SyllabusTrack is an Android app that turns “how much syllabus is left?” from a gut feeling into a number: chapter-level progress with exam weightage tags, an exam countdown, study streaks and a revision-due queue.

Project type: Software (Android application)

Availability: Built to order

Suitable branches: Computer / IT, Android

Technologies: Kotlin, Android Studio, Room · SQLite, Material Design 3, MVVM, WorkManager, Chart rendering, Git

Prepared by Projectech · projectech.in — for academic project reference.

2. Introduction / Background

Students preparing for board exams and entrance tests face a planning problem disguised as a studying problem: the syllabus is large, time is fixed, and progress is usually tracked in the student's head or on a notebook page that goes stale within a week. High-weightage chapters are discovered unfinished days before the exam, and revision is piled onto already-full days.

SyllabusTrack makes preparation countable. Each subject holds its chapter list with High/Medium/Low weightage tags drawn from the exam blueprint; completing a chapter moves the subject's progress bar and the overall ring immediately. A dashboard pairs this with a live countdown to the exam date and a streak counter, while a stats screen charts weekly study hours and keeps a revision-due queue. All data lives on-device, keeping the app offline-first and fully demoable.

3. Problem Statement

Without a tracking system, students cannot answer the one question that matters — “am I on track?” — with anything better than a feeling. Weightage-blind studying gives equal time to low-marks chapters while high-marks chapters slip; finished chapters are never revised because no system resurfaces them; and daily consistency collapses because no counter rewards showing up.

Generic to-do apps fail here because they lack the exam-specific model: subjects, chapters, weightage tags, revision intervals and a countdown. This project builds an Android app around exactly that model, with

offline-first storage so it works in hostels with patchy connectivity and a stats screen that turns a weekly review into a three-number check.

4. Objectives

- Track syllabus progress at chapter level with instant subject and overall updates.
- Tag chapters with exam weightage (High/Medium/Low) so high-marks topics are prioritized.
- Show a live countdown to the exam date with the target score.
- Maintain a study streak with the longest streak recorded.
- Chart weekly study hours and surface a revision-due queue ordered by urgency.
- Work fully offline with all data in an on-device database.

5. Proposed Methodology

The exam-preparation workflow is modelled first: exam profile (name, date, target), subjects, chapters with weightage tags, study sessions and revision intervals per tag. This becomes a Room/SQLite schema with subjects, chapters, sessions and revision records.

The app is built in Kotlin following MVVM architecture — ViewModels exposing StateFlow/LiveData to Material Design 3 screens — with three main surfaces: dashboard (countdown, progress ring, subjects), subject detail (chapter checklist) and stats (hours chart, streak, revision queue). Revision reminders are scheduled with WorkManager as local notifications. Testing covers the progress-computation formula, reminder scheduling, database backup/restore and usability with real student chapter lists.

6. System Architecture / Working

The architecture is a single-device offline-first mobile app. The data layer is Room/SQLite holding the exam profile, subjects, chapters, study sessions and revision schedule. The domain layer (ViewModels) computes progress percentages, streaks and revision urgency from this data. The presentation layer is three Material Design 3 screens plus local notifications scheduled by WorkManager; chart rendering draws the weekly-hours bars.

Working flow: (1) the student creates an exam profile with date and target; (2) subjects and chapters are added with weightage tags from the exam blueprint; (3) chapters are tapped done as finished, updating progress instantly; (4) the dashboard shows countdown, overall ring and per-subject bars; (5) study sessions are logged or timed, feeding the hours chart and streak; (6) finished chapters resurface in the revision queue when their interval elapses, ordered by urgency; (7) the stats screen combines hours, streak and revision status for a weekly review.

7. Hardware / Software Requirements

- Device: Android 8.0 (API 26) or above; no server or internet required for the core loop.
- Development: Android Studio, Kotlin, Room, Material Design 3 components, Git.

- Notifications require the app installed with notification permission granted.

8. Expected Outcomes

- A working Android app that answers “how much is left?” as an exact percentage per subject and overall.
- Weightage-tagged chapter lists that visually prioritize high-marks topics.
- A dashboard with countdown, progress ring, streak and a stats screen with hours chart and revision queue.
- Reliable local revision reminders via WorkManager.
- Complete documentation: project report, presentation and viva Q&A.

9. Applications

- Board exam (Class 10/12) preparation tracking.
- Entrance test preparation (JEE, NEET, CET) with chapter weightage.
- University semester syllabus tracking.
- Competitive exam study planning with revision scheduling.

10. Future Scope

- Cloud sync and multi-device access with user accounts.
- Adaptive spaced-repetition intervals tuned from the student's recall pattern.
- Official syllabus import instead of manual chapter entry.
- Multi-exam dashboards managed in parallel.
- Study-group features: shared targets and accountability.