

PROJECT ABSTRACT

Peer-to-Peer Learning App using Flutter

Cross-platform Flutter study app with Firebase backend: study groups, realtime chat, notes sharing, flashcards and quizzes — built-to-order with source code, report, PPT and viva kit.

01. Title

Peer-to-Peer Learning App using Flutter

02. Introduction / Background

College students already study in informal peer groups — WhatsApp groups full of PDFs, scattered notes and last-minute doubt threads — but nothing in that workflow is organized around learning: no shared flashcard decks, no quizzes, no subject-wise grouping. A peer-to-peer learning app gives each subject its own study circle with realtime chat, shared notes and self-testing tools. Flutter builds this once for Android and iOS from a single codebase, and Firebase provides auth, the realtime database, file storage and push notifications without managing servers. This project ships that complete app.

03. Problem Statement

Study material in student groups is fragmented across chats, drives and screenshots, and there is no structured way to quiz each other before exams. The project addresses this with a dedicated learning app: students join subject-wise study groups, discuss in realtime chat, upload and browse shared notes, drill concepts with flashcards and test themselves with quizzes — all in one place, with push notifications keeping the group loop closed.

04. Objectives

- Build a Flutter app (single codebase, Android-first) with Firebase Auth (email/Google sign-in).
- Implement study groups backed by Firestore: create/join, member lists, subject tags.
- Implement realtime group chat, notes upload/download via Firebase Storage, flashcard decks with flip/shuffle, and quizzes with instant scoring.
- Add push notifications (FCM) for new messages and quiz events, and deliver the full source, setup guide, report, PPT and viva Q&A.

05. Proposed Methodology

The project follows the standard Flutter + Firebase app pattern. The UI is built as a widget tree with four main screens — Groups, Chat, Learn (flashcards + quiz) and Profile — managed with a simple state-management approach. Firebase Auth handles sign-in; Firestore collections store users, groups, messages, notes metadata and quiz results with security rules restricting reads/writes to group members; Firebase Storage holds the note PDFs and images; FCM delivers push notifications via a small Cloud Function triggered on new messages. The app is developed and tested on Android emulator and a physical device.

06. System Architecture / Working

The system has three layers. (1) Client: the Flutter app with bottom-tab navigation across Groups, Chat, Learn and Profile screens. (2) Backend: Firebase Auth, Firestore (users, groups, messages, decks, quiz attempts), Storage (notes) and FCM (notifications) — no custom server to maintain. (3) Realtime sync: Firestore snapshot listeners update chat and group screens live; optimistic UI keeps messaging snappy on slow networks. Quiz scoring runs client-side with results persisted per user for progress tracking.

07. Hardware / Software Requirements

- Flutter SDK (stable channel), Dart
- Firebase project: Authentication, Cloud Firestore, Storage, Cloud Messaging
- Android Studio / VS Code with Flutter extensions
- Android emulator or physical Android device for testing (iOS build is an optional add-on requiring macOS/Xcode)
- No ML dataset or model is involved in this project

08. Expected Outcomes

The student receives a working cross-platform app: users sign in, create and join study groups, exchange realtime messages, upload and download shared notes, flip through flashcard decks and take scored quizzes with instant feedback. Group screens update live for all members, and push notifications arrive for new chat activity. The report documents the Firestore data model, security rules and the test matrix (emulator + physical device).

09. Applications

- Subject-wise peer study groups in colleges
- Placement-preparation circles (DSA, aptitude quizzes)
- Class doubt-clearing channels moderated by class reps
- Reference Flutter + Firebase build for mobile-development coursework

10. Future Scope

- Video study rooms inside group screens
- Spaced-repetition scheduling for flashcards
- Leaderboards and study streaks for engagement
- iOS release build and web companion dashboard