

TITLE: TIMETUNE

Introduction:

This system enables teachers to swap lectures dynamically with students receiving real-time updates through the app and push notifications.

Current Software Architecture:

The system follows client-server model, with the Android app as the client and Firebase service managing the backend (Authentication, messaging)

Proposed Software Architecture:

Follows the MVC pattern.

Persistent Data Management: Data is stored in JSON like-format for lectures, while user data is managed by Firebase Authentication.

Access Control and Security: Firebase Authentication manages access control with encrypted data storage.

Global Software Control: Firebase handles real-time updates and concurrency. No explicit threading is needed.

Boundary Conditions: App checks authentication at startup, terminates Firebase listeners at shutdown, and handles errors.

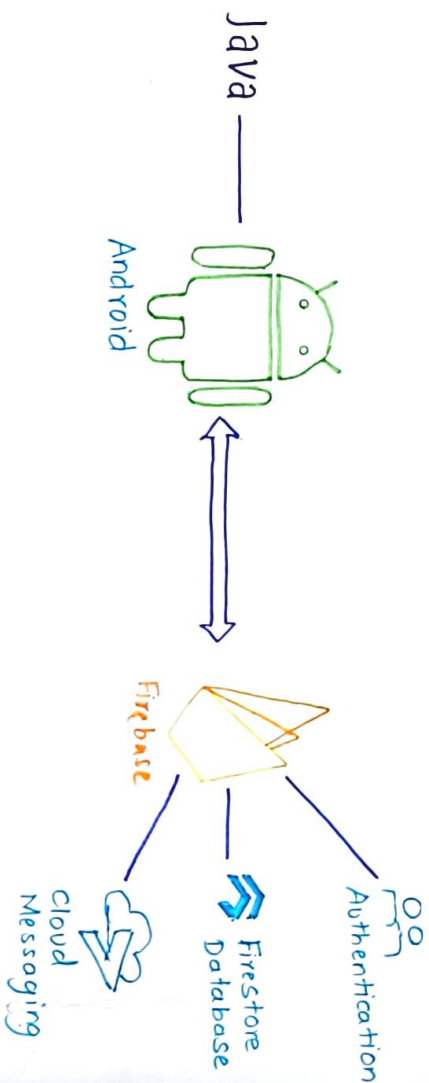
Subsystem Services:

User Management: APIs for User creation login and logout.

Lecture Management: To retrieve and swap lectures.

Notification: To send notifications using Firebase Messaging.

SYSTEM ARCHITECTURE



FLOWCHART

