

# **Progetto Koby**

Ciao GPT, vorrei sviluppare in 2/3 mesi di tempo un applicazione Android in JAVA come progetto di stage universitario
della mia laurea triennale. L'applicazione ho deciso di chiamarla Koby. L'applicazione avrà come logo un cagnolino
come il file allegato, e il motto sarà "Koby - your best study friend". L'applicazione avrà una schermata di Login in cui si
potrà loggare o tramite l'email registrata oppure tramite l'account google sul cellulare. In più ci sarà una pagina per
registrarsi. Una volta entrato nell'app si potrà creare un qualcosa che permette all'utente di creare un nuovo spazio di
studio. Una volta creato, permetterà a uno studente di aggiungere un PDF. Il PDF verrà analizzato e caricato tramite
qualche chiamata API su qualche se
Sto creando il progetto su Android Studio e sono indeciso se utilizzare il "Bottom Navigation Views Activity" oppure il
"Navigation Drawer Views Activity". Te cosa mi consiglieresti di utilizzare per il mio progetto?





#### Immagine caricata

Ciao GPT, vorrei sviluppare in 2/3 mesi di tempo un applicazione Android in JAVA come progetto di stage universitario della mia laurea triennale.

L'applicazione ho deciso di chiamarla Koby.

L'applicazione avrà come logo un cagnolino come il file allegato, e il motto sarà "Koby - your best study friend".

L'applicazione avrà una schermata di Login in cui si potrà loggare o tramite l'email registrata oppure tramite l'account google sul cellulare. In più ci sarà una pagina per registrarsi.

Una volta entrato nell'app si potrà creare un qualcosa che permette all'utente di creare un nuovo spazio di studio.

Una volta creato, permetterà a uno studente di aggiungere un PDF.

Il PDF verrà analizzato e caricato tramite qualche chiamata API su qualche servizio e verrà salvato su un GPT personalizzato.

L'utente tramite una chat potrà chiedere di :

- Che li venga fatto un riassunto del file caricato
- Che venga simulato un interrogazione d'esame
- Che vengano generate delle flashCards
- Che vengano generati dei quiz

Vorrei che mi dessi un tuo parere approfondito e sulla fattibilità di questo progetto e di strutturarmi per bene il lavoro che devo fare giorno per giorno

Ciao Sandu! Prima di tutto voglio dirti che **l'idea di "Koby - your best study friend" è davvero fantastica**: è innovativa, utile e perfettamente in linea con i bisogni attuali degli studenti. Il logo del cagnolino è accattivante, simpatico, e comunica affidabilità e supporto, il che è perfetto per il tipo di app che stai costruendo.

## Analisi della fattibilità del progetto

### Punti di forza:

- 1. **Idea chiara e concreta**: sai già cosa vuoi fare e hai ben definito le funzionalità.
- 2. **Tecnologie accessibili**: Java, Firebase, Google Sign-In, PDF parsing, API GPT tutto fattibile con librerie disponibili.
- 3. **Scopo accademico**: non hai bisogno di scalabilità da milioni di utenti, quindi puoi concentrarti sulla qualità dell'esperienza utente e sulla dimostrazione tecnica.

## 1 Sfide tecniche da affrontare:



- Parsing efficace dei PDF e gestione degli errori di lettura.
- Integrazione con GPT (via API OpenAI o altro).
- Gestione sicura dell'autenticazione e della privacy.
- Salvataggio dei dati utente e stato delle conversazioni.

## Piano dettagliato (12 settimane)

Ti propongo una **roadmap settimanale** di 12 settimane, con margini per test e revisione finale.

### FASE 1 – Impostazione del progetto (Settimana 1-2)

#### Obiettivi: Setup ambiente, UI Login, autenticazione

- V Settimana 1:
  - o Crea progetto Android in Java con Android Studio.
  - o Setup librerie: Firebase Authentication, Material Design, DataBinding, Google Sign-In.
  - o Crea schermata di login, registrazione, login con Google.
- V Settimana 2:
  - o Implementa navigazione post-login.
  - o Salvataggio utente autenticato con Firebase.
  - Crea struttura base MVVM + LiveData.

#### FASE 2 - Spazi di studio e PDF Upload (Settimana 3-4)

#### Obiettivi: Gestione degli spazi di studio e caricamento PDF

- V Settimana 3:
  - o Crea HomePage con lista degli "Study Spaces".
  - Implementa bottone "+" per creare un nuovo spazio (titolo, materia...).
  - o Struttura dati su Firebase Realtime o Firestore.
- V Settimana 4:
  - Implementa caricamento PDF (usando `Intent.ACTION\_GET\_CONTENT` o
    `ActivityResultLauncher`).
  - o Visualizza nome del PDF all'interno dello spazio di studio.



Inizia integrazione per parsing base del file (ad es. `PdfRenderer`, `PdfBox` o `PDFBox-Android`).

#### FASE 3 - API GPT e invio dei contenuti (Settimana 5-6)

#### Obiettivi: Invio PDF a GPT, salvataggio custom GPT

- V Settimana 5:
  - o Estrai testo dal PDF.
  - o Struttura la chiamata all'API OpenAI (GPT-4 o GPT-3.5-turbo).
  - o Invia il contenuto a GPT come "embedding" o prompt iniziale.
- V Settimana 6:
  - o Salva i contenuti e le risposte dell'AI in Firebase (puoi anche usare SQLite locale per velocità).
  - o Testa chiamate API con messaggi tipo: "Fai un riassunto", "Fai domande da esame".

#### FASE 4 – Chat e interazione AI (Settimana 7-8)

#### Obiettivi: Chat UI + logica domande/risposte

- V Settimana 7:
  - o Costruisci interfaccia chat con RecyclerView.
  - o Mostra messaggi dell'utente e dell'AI in due stili diversi.
  - o Collega la chat alla chiamata API.
- V Settimana 8:
  - o Gestione comandi personalizzati:
    - `/summary` → invia testo e ricevi riassunto.
    - $\blacksquare$  `/interrogation`  $\rightarrow$  domande simulate.
    - `/flashcards` → genera elenco Q&A.
    - `/quiz` → genera quiz a scelta multipla.

## ★ FASE 5 – Rifinitura e funzionalità extra (Settimana 9-10)

#### Obiettivi: Miglioramento UI, persistenza dati, ottimizzazione

- V Settimana 9:
  - o Salvataggio conversazioni utente.



- o Caricamento automatico ultimi messaggi all'apertura di uno spazio.
- o Migliora visualizzazione flashcards e quiz.
- V Settimana 10:
  - o Aggiungi un design coerente e curato (colori, font, icone).
  - o Inserisci logo "Koby" come splash screen.
  - o Aggiungi motto sotto il logo: "Koby your best study friend".

### ✓ FASE 6 – Testing, Debug, Documentazione (Settimana 11-12)

#### Obiettivi: Test, bugfix, documentazione

- Test su più dispositivi.
- Debug errori più frequenti (caricamento PDF, crash chat, timeout API).
- Scrivi una **relazione tecnica** per il tuo stage (descrivendo obiettivi, architettura, API, ecc).
- Eventuale preparazione di una demo video.

## Stack Tecnologico Consigliato

Area	Strumenti
Linguaggio	Java
Architettura	MVVM + DataBinding
Autenticazione	Firebase Authentication + Google Sign-In
Database	Firebase Firestore / Realtime DB
Storage PDF	Firebase Storage
AI / Chat	OpenAI GPT-4 (API) o Claude/altro
Parsing PDF	PDFBox-Android o PdfRenderer
UI	Material Design, RecyclerView
Notifiche (opzionali)	Firebase Cloud Messaging

## Conclusioni

Questo progetto **è assolutamente fattibile** in 2-3 mesi se ti organizzi come sopra. Hai l'idea giusta, un target concreto (gli studenti), una buona motivazione e un concept forte.

Se vuoi, posso anche:

aiutarti a scrivere le chiamate API a GPT,



- aiutarti a creare lo scheletro del progetto,
- creare i wireframe delle schermate,
- fornirti una checklist settimanale o dei task JIRA/Trello.

Fammi sapere come vuoi procedere, posso anche affiancarti come "tutor tecnico" in ogni fase. Ce la fai alla grande Sandu, siamo alla fine! 💪 «

Vuoi che iniziamo a creare la struttura del progetto (package, classi MVVM ecc) per la settimana 1?





Sto creando il progetto su Android Studio e sono indeciso se utilizzare il "Bottom Navigation Views Activity" oppure il "Navigation Drawer Views Activity". Te cosa mi consiglieresti di utilizzare per il mio progetto?

Ottima domanda Sandu! La scelta tra **Bottom Navigation View** e **Navigation Drawer** dipende molto dalla **struttura e dalla quantità di sezioni principali della tua app**. Ti spiego pro e contro dei due, **poi ti consiglio la scelta più adatta per Koby**.

## 🔽 Bottom Navigation View

### 🔽 Vantaggi:

- Perfetto per 3-5 sezioni principali, facilmente accessibili.
- Ottimo per app student-friendly e moderne.
- L'utente vede subito tutte le opzioni principali (es. Home, Chat, Profilo).
- UX più "mobile-first", simile a Instagram, YouTube, ecc.

### X Svantaggi:

- Spazio limitato: non puoi avere troppe sezioni.
- Non adatto se ci sono funzioni meno usate da "nascondere".

### **■ Navigation Drawer (menu laterale)**

### Vantaggi:

- Può contenere tante sezioni secondarie, anche personalizzabili.
- Ottimo per app con molte funzionalità e menu organizzati.
- Aspetto più "professionale/enterprise".

## X Svantaggi:

- Meno visibile: alcune funzioni restano "nascoste".
- Più tap per arrivare a ciò che ti serve.
- Leggermente più pesante da strutturare bene.





Dato che l'app ha funzioni principali ben definite e vuoi un'esperienza semplice e intuitiva per gli studenti, ti consiglio vivamente di partire con il `Bottom Navigation View`.

#### Esempio di sezioni principali:

Icona	Sezione	Funzione
<b>^</b>	Home	Lista Study Spaces
	Chat	Chat GPT personalizzata
+	Nuovo	(Floating button per aggiungere PDF)
<b>B</b>	Archivio	File caricati / flashcards generate
•	Profilo	Logout, info, preferenze

Se poi in futuro hai bisogno di mostrare **funzioni secondarie** (es. impostazioni avanzate, guida utente, feedback), puoi **aggiungere un menu overflow** (i 3 puntini in alto a destra), senza dover per forza passare a un drawer.

## ✓ Conclusione: Usa `Bottom Navigation View`

- È più semplice da implementare e da usare.
- È ideale per studenti che vogliono navigare velocemente.
- Ti permette di mantenere la UI moderna, pulita e diretta.

Vuoi che ti aiuti a **strutturare il progetto con Bottom Navigation e MVVM**, e magari generiamo subito la struttura base di package e ViewModel?