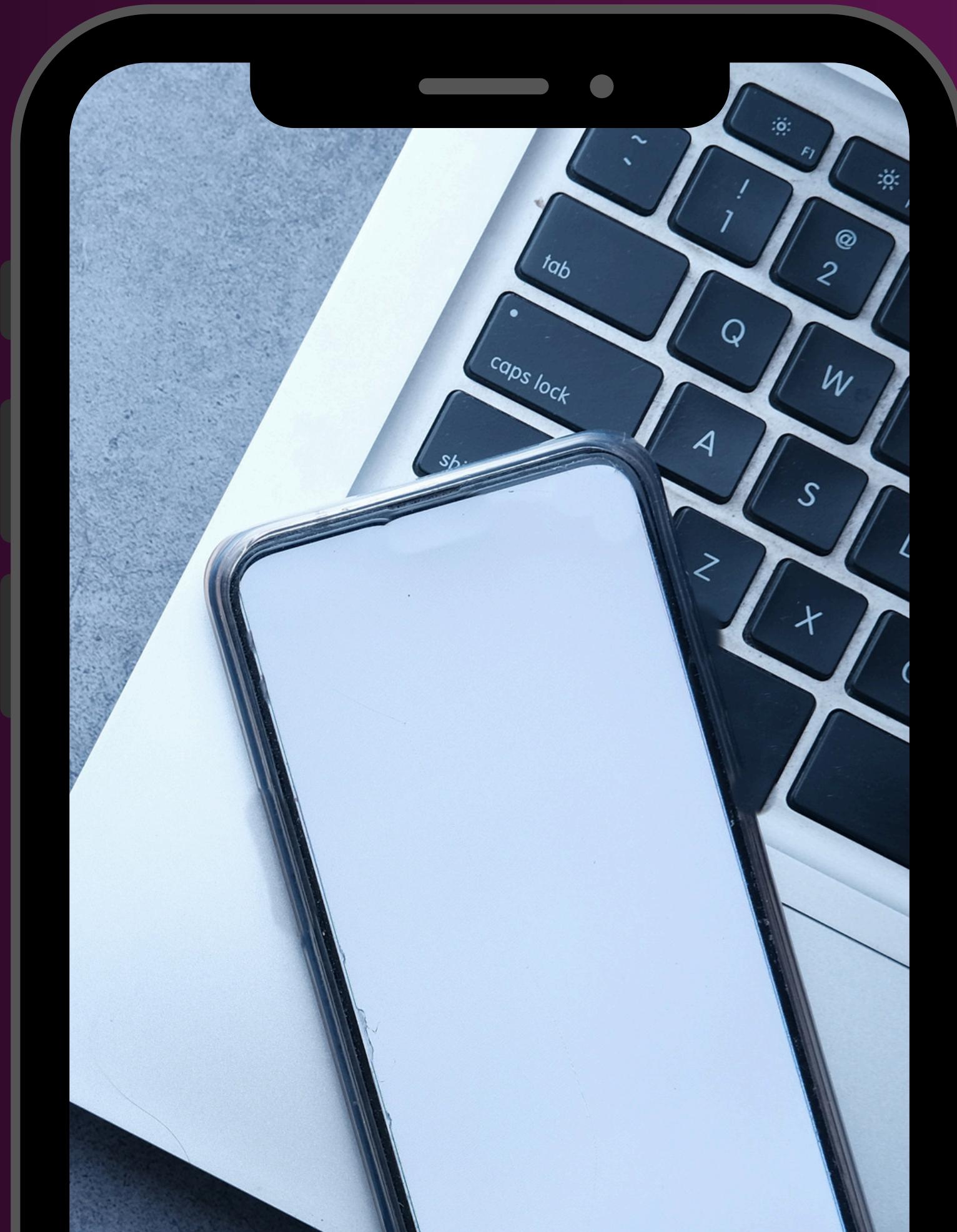


Version 1.1.0 (Final  
Release)

# BooksApp – Android Application

- Kotlin + Jetpack Compose
- MVVM + Repository Architecture
- Open Library API + Room (Offline Cache)
- Firebase Auth + Realtime Database

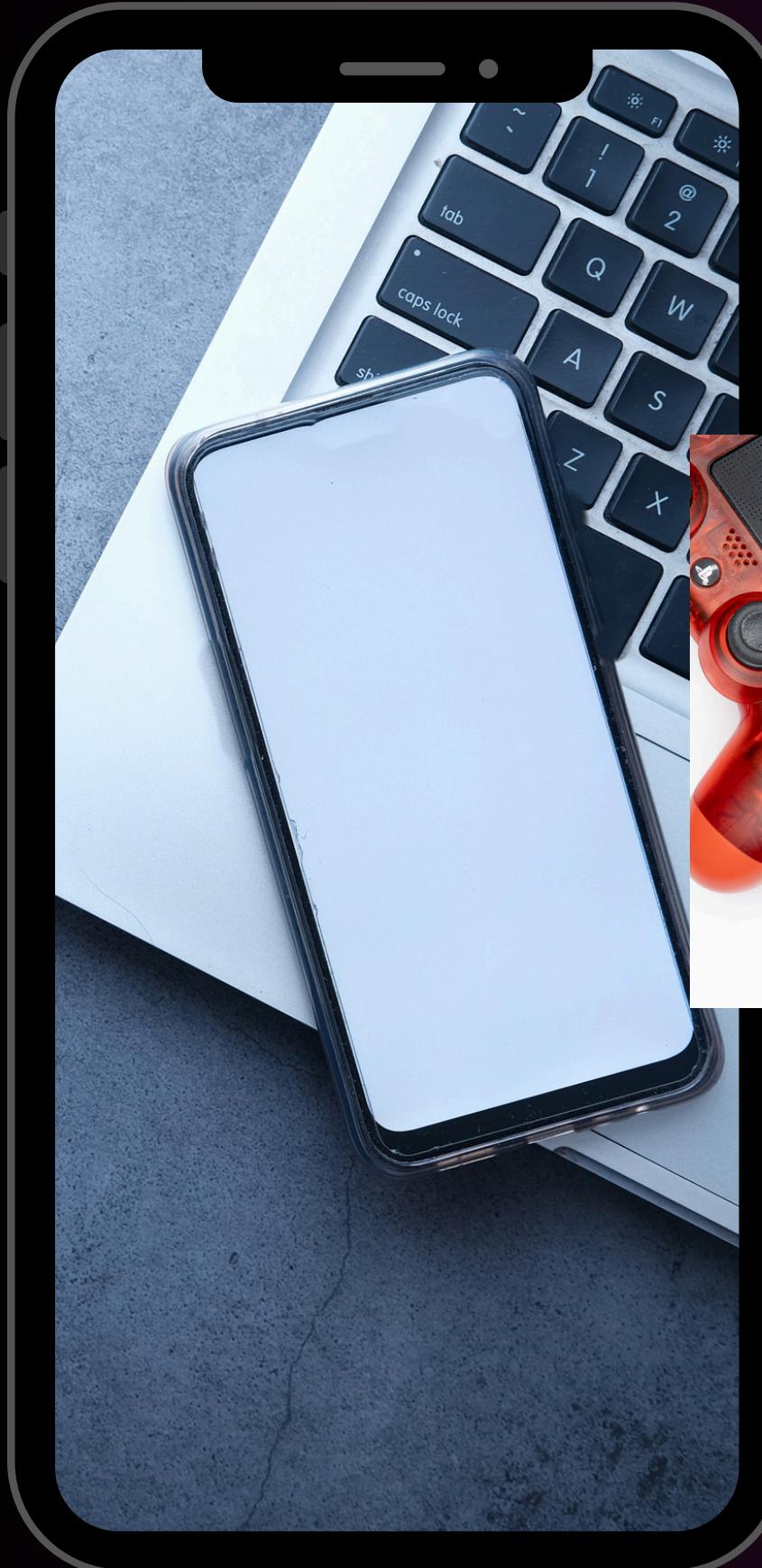
**Kassenov Abdulkarim,  
Yerdaulet Akhanov**



# Architecture Overview

- UI (Compose Screens)
- ViewModel (StateFlow, UI logic)
- Domain (UseCases, Business Logic)
- Repository (Single source of truth)
- Data (Retrofit API + Room + Firebase)
- Dependency Injection via Hilt





# ● Navigation & Screens



- Auth Screen (Login / Register)
- Feed Screen
- Search Screen
- Details Screen

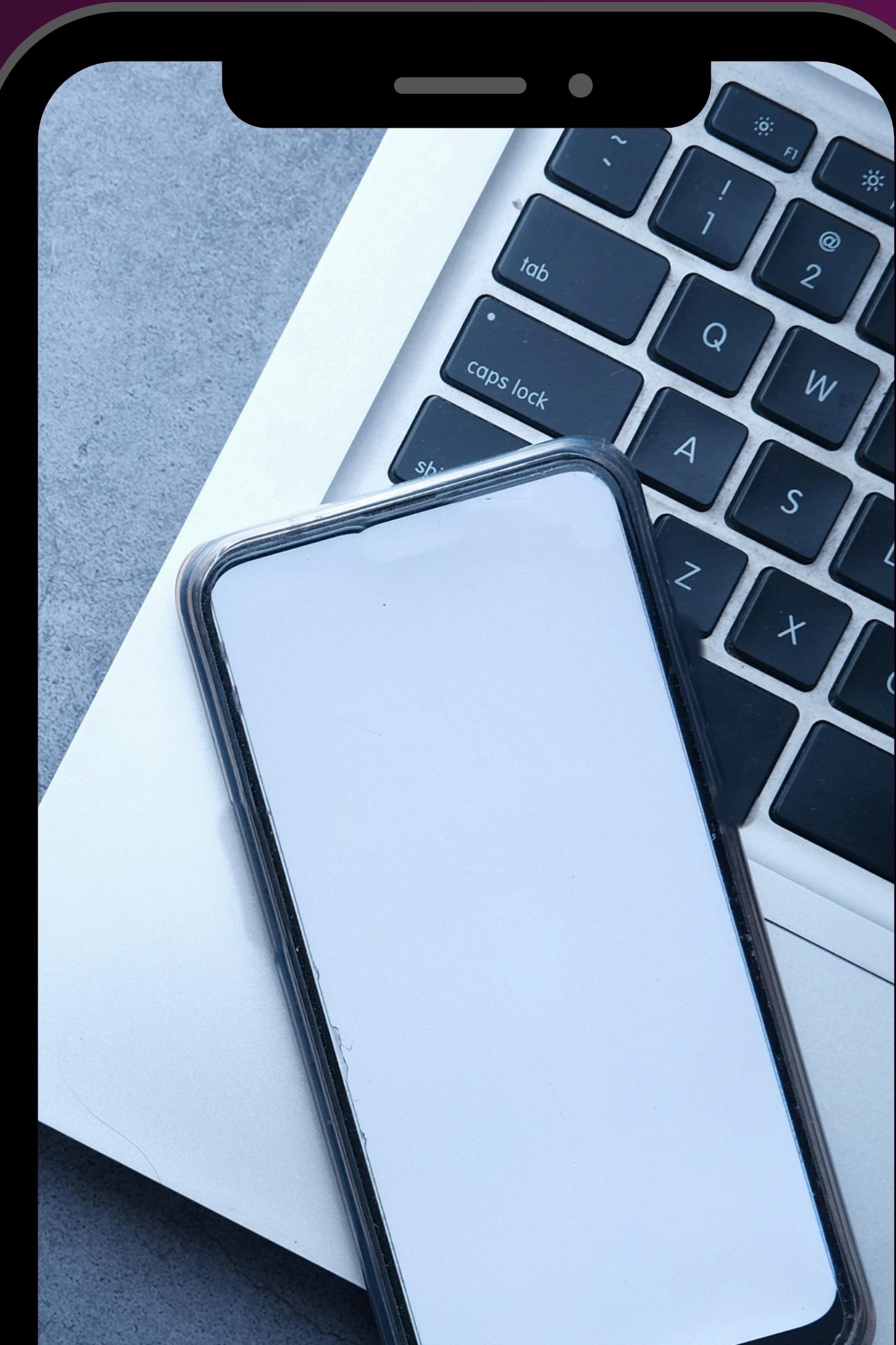
- Review Editor Screen
- Profile Screen
- Deep Navigation: Feed → Details → Review Editor

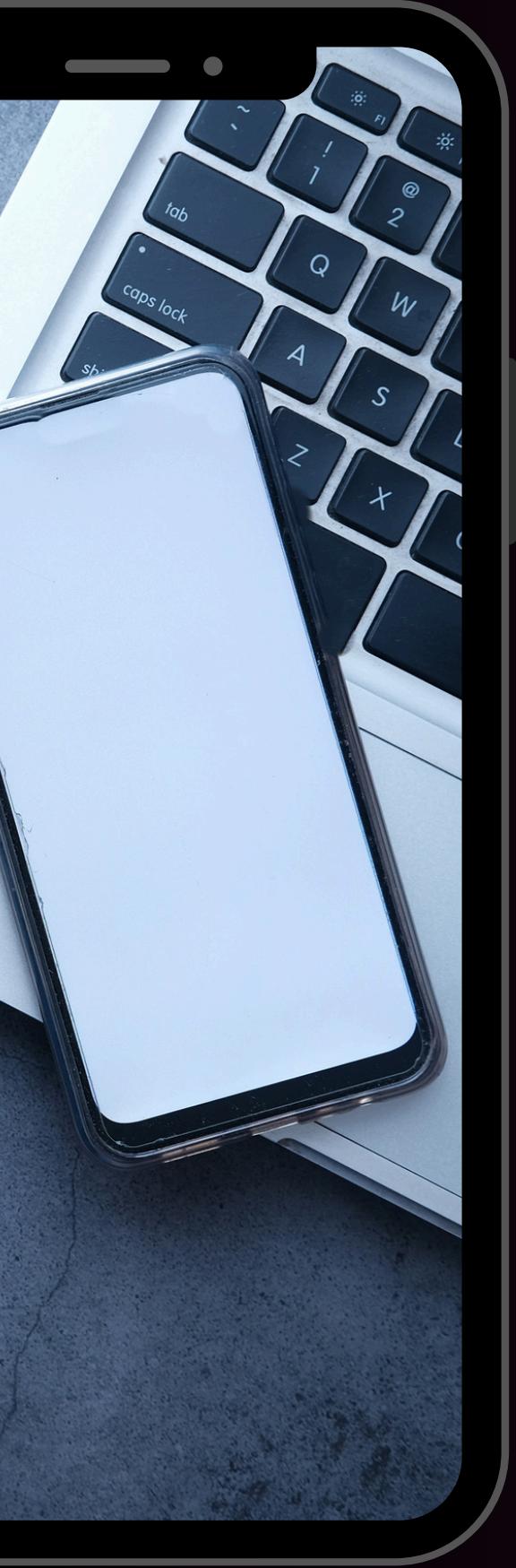
# Authentication & Session Management



The Smartest Innovations In Application Development

- Firebase Email/Password Authentication
- AuthStateListener for session persistence
- User remains logged after restart
- User-scoped data storage using UID





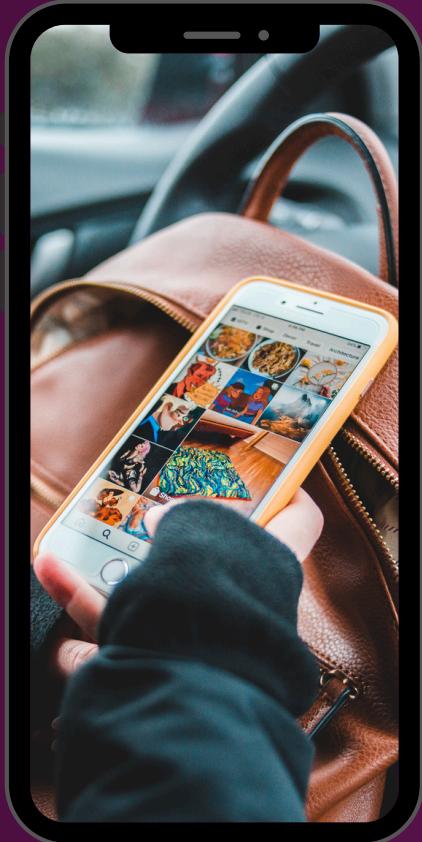
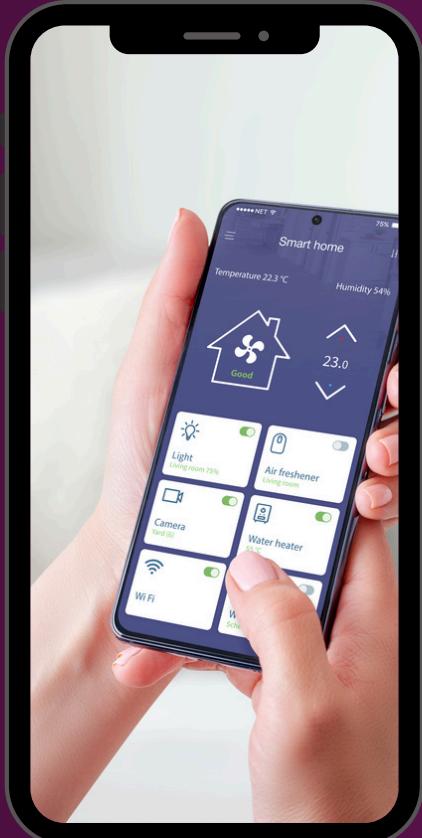
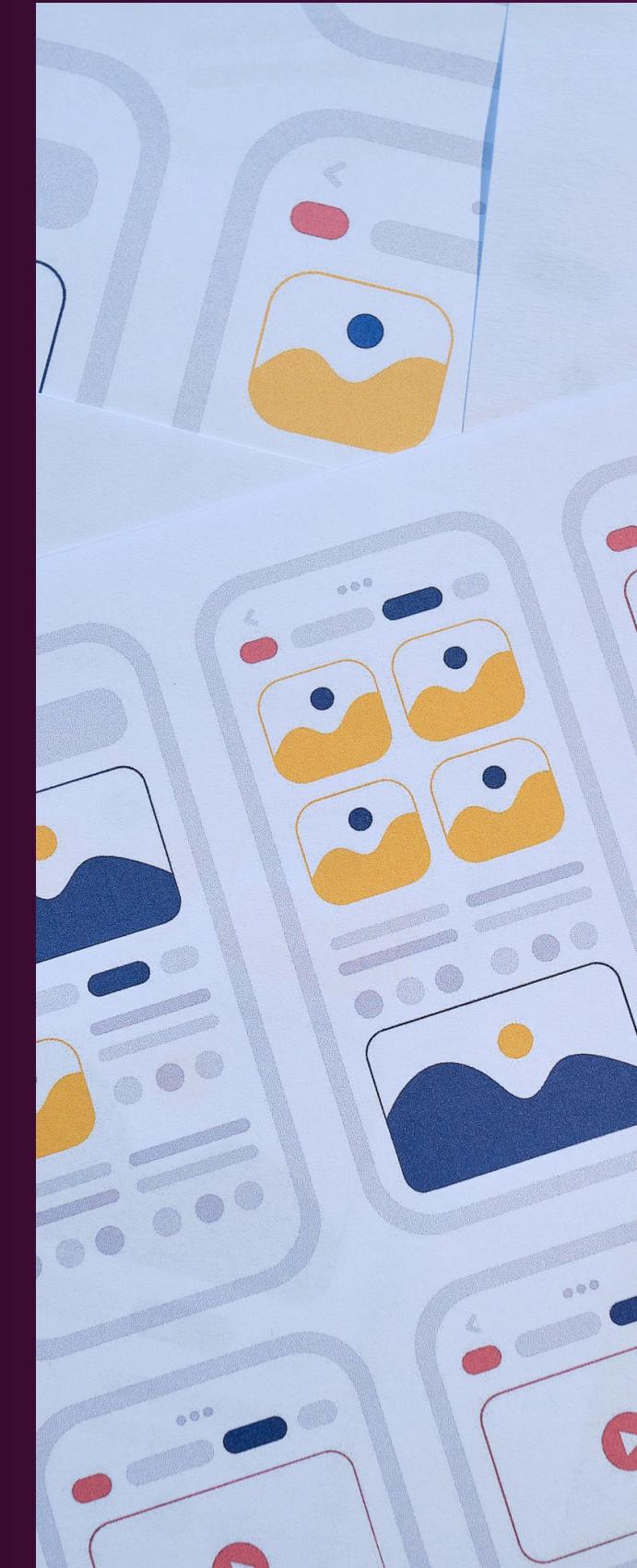
# Search + Debounce + Pagination

debounce(450ms) to prevent excessive API calls

distinctUntilChanged() to avoid duplicate queries

Pagination with loadMore()

Duplicate prevention via BookMergePolicy



# Mobile Security

- **Data Protection**

Encryption, secure connections

- **App Permissions**

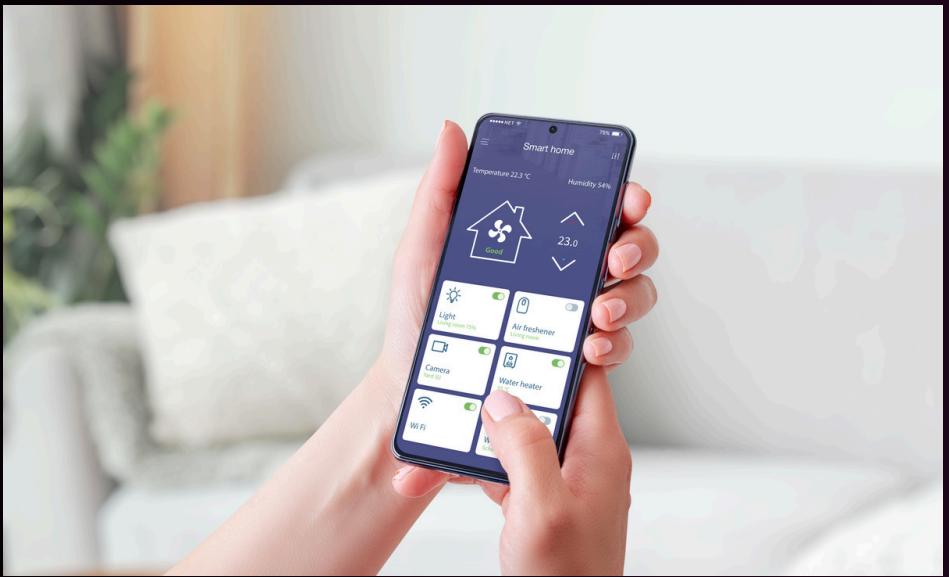
Control over access to personal data

- **Privacy Policies**

User consent and data collection practices



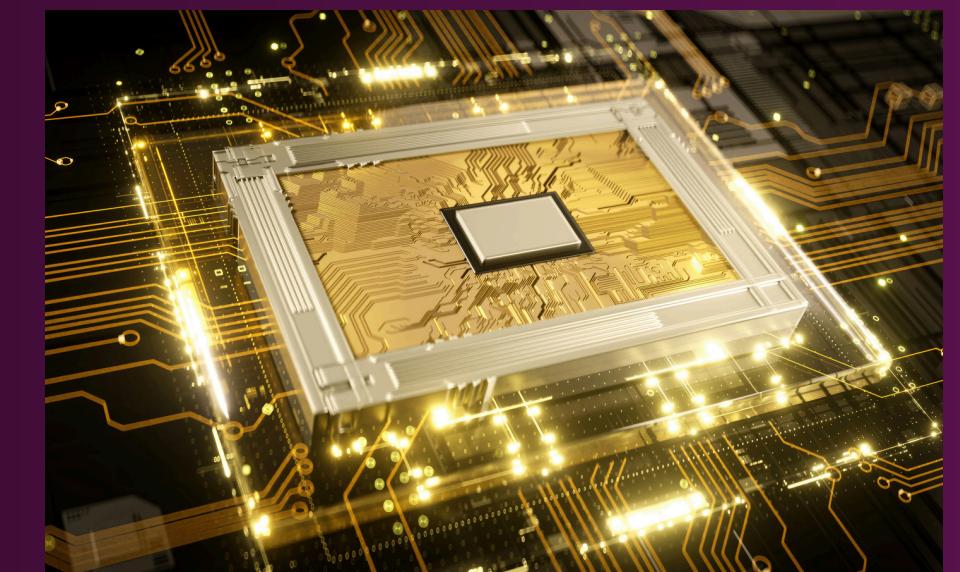
# Offline-First Strategy



- Data cached in Room database
- Cache freshness check via CachePolicy
- If cache is fresh → skip network call
- If network fails → fallback to cache
- Error only shown when no cache exists

# Realtime Firebase Features

- Favorites path:  
`/users/{uid}/favorites/{bookId}`
- Comments path:  
`/comments/{bookId}/{commentId}`
- Flow listeners for realtime updates
- Add / Edit / Delete without restart





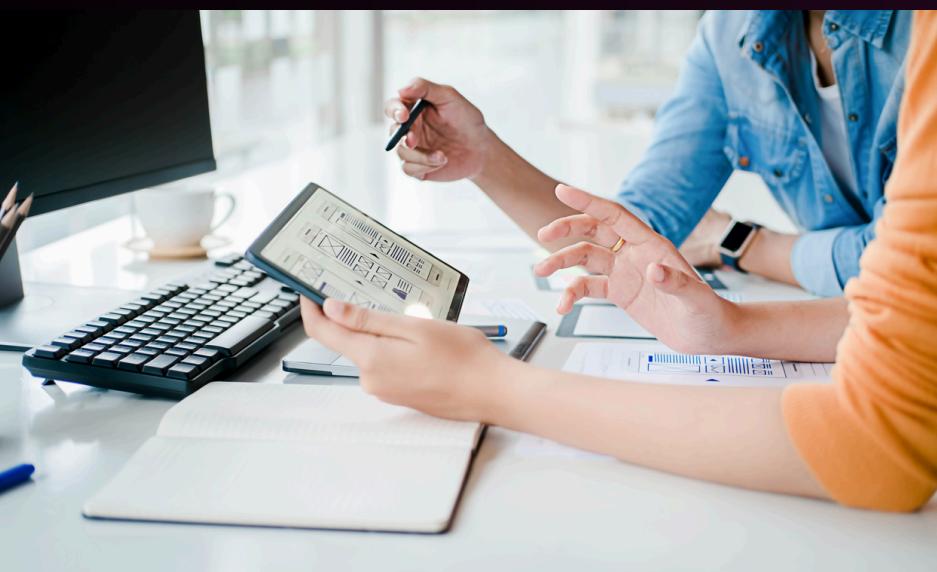
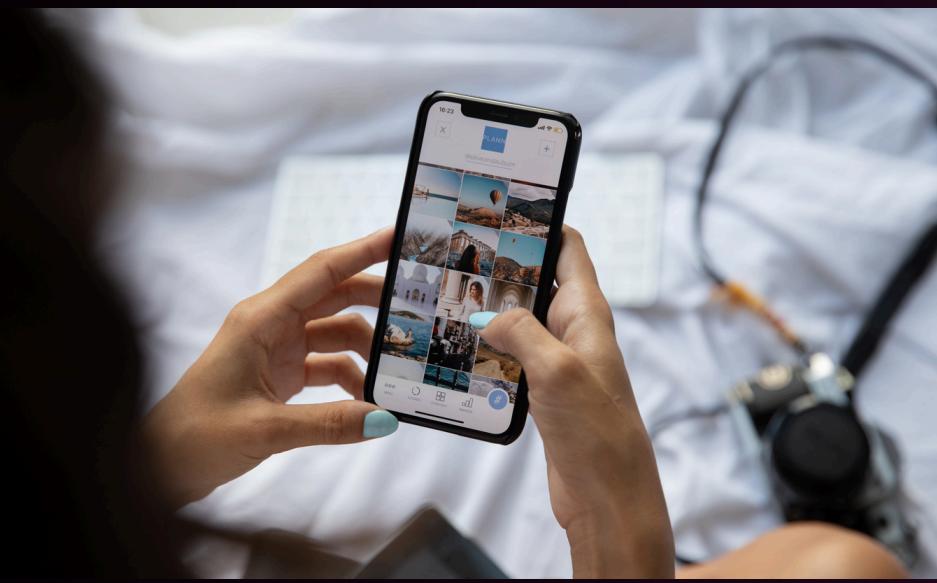
# Business Logic Layer

- Recommendation Scorer (recommendation calculation)
- BookMergePolicy (conflict resolution)
- CredentialsValidator (auth validation)
- ReviewValidator (review validation)
- QuerySanitizer (safe input handling)

# Security & Configuration

- No secrets stored in Git repository
- Keys loaded from local.properties
  - Firebase path sanitization
  - Input validation before saving data
- Verbose logs disabled in release build

# Final Improvements (Endterm → Final)



- versionCode = 2
- versionName = 1.1.0
- RetryPolicy implemented
- Performance improvements documented
- QA checklist completed (21 items)
- Release notes prepared



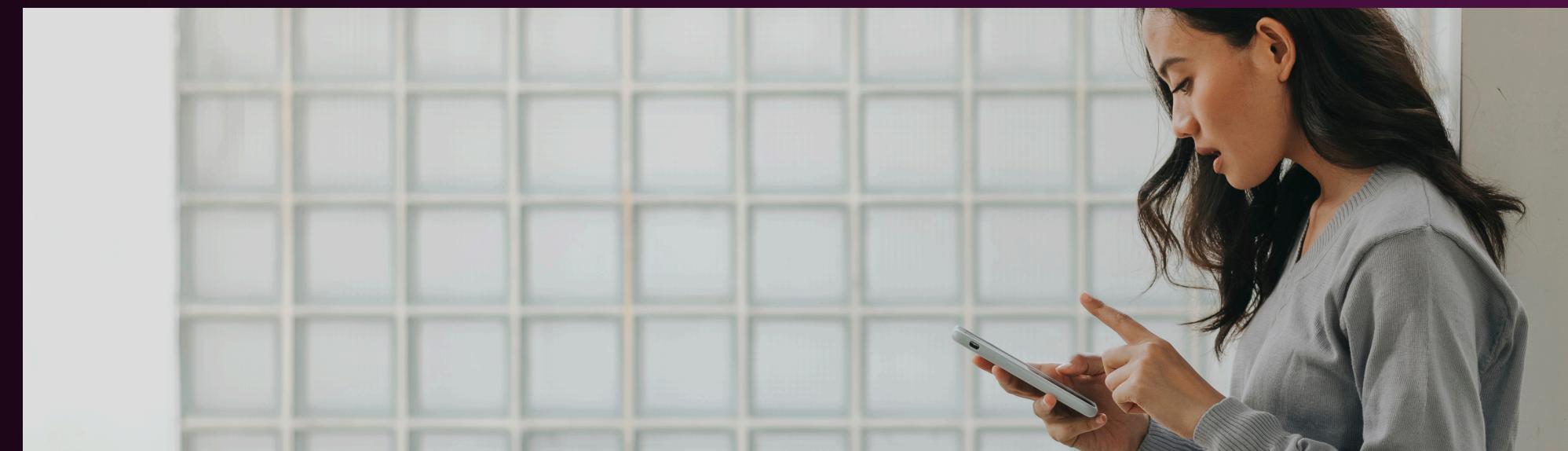
# Stability & Testability

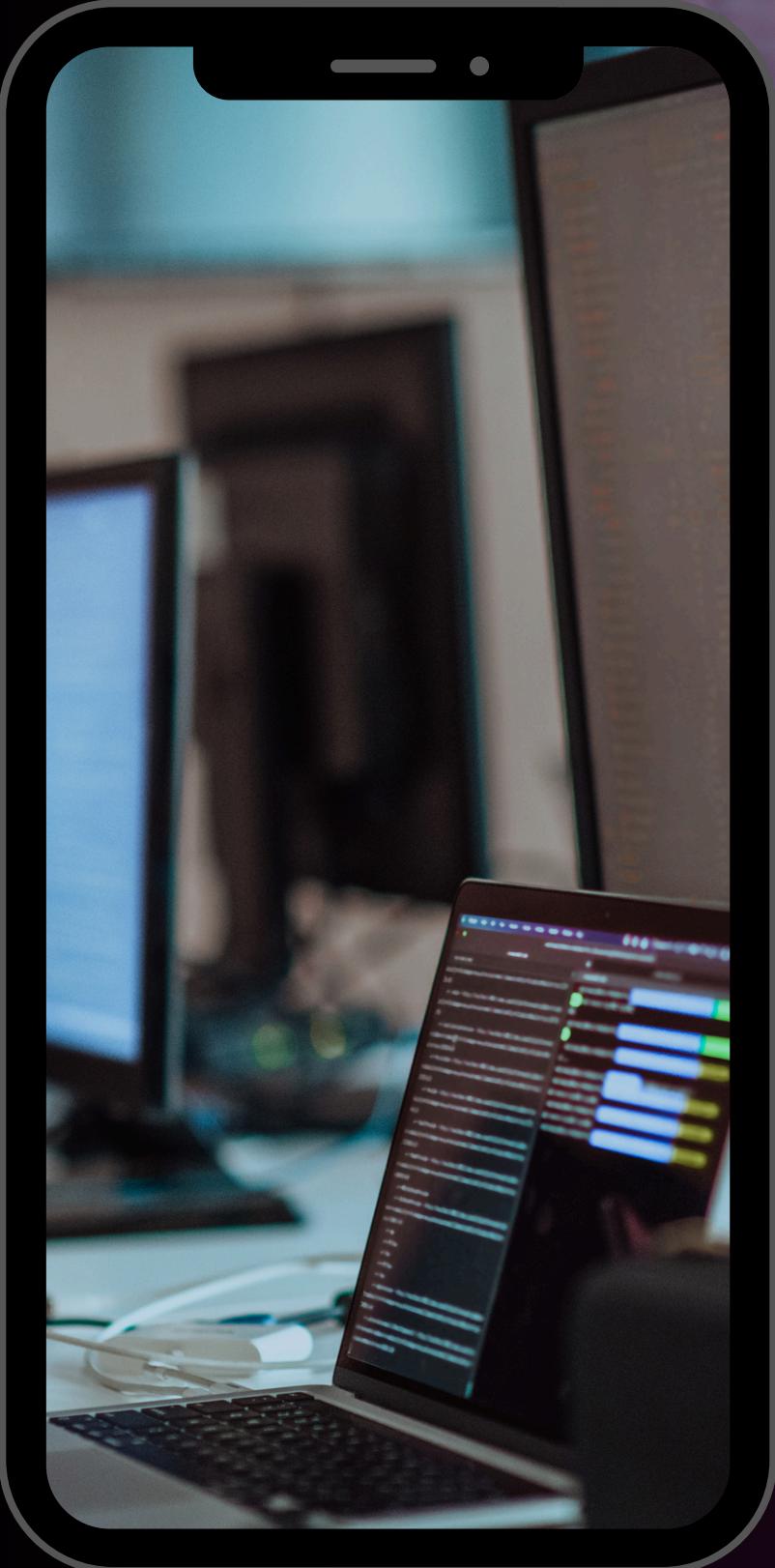
- MVVM improves separation of concerns
- Repository centralizes data sources
- RetryPolicy handles transient errors
- Manual retry in UI
- Offline fallback guarantees usability

# Demo Plan & Conclusion

1. Login / Registration
2. Browse Feed
3. Search with Debounce
4. Pagination
5. Offline mode demonstration
6. Realtime comment & favorite
7. Profile & Logout

Production-ready Android architecture





# Thank You

Thank you for exploring the fascinating world of mobile apps with us.  
Stay connected, stay app-savvy!

