Finance Manager - One Page App Summary

Evidence source: repository files (README.md, lib/main.dart, router, services, repositories, feature modules).

What It Is

- Finance Manager is a Flutter app for personal money management with local-first data storage and cloud sync support. It combines transaction, income, budget, category, recurring, analytics, and AI assistant flows in one mobile app.

Who It Is For

- Primary persona: an individual user who wants to track spending and income, monitor budgets, and review trends with optional AI guidance.

What It Does

- Email/password and Google sign-in via Firebase Auth.
- Tracks expenses and incomes with month filtering, search, and CRUD flows.
- Manages category budgets with spent calculations and warning/exceeded states.
- Initializes default categories and supports custom category management.
- Stores data locally in Hive, then syncs to Firestore with graceful fallback.
- Provides analytics screens/charts for balance and spending breakdowns.
- Includes an AI chat assistant (Gemini) for category suggestions and insights.

How It Works (Repo Evidence)

- App startup in lib/main.dart initializes Firebase + Hive, registers adapters, configures GetIt, and mounts MultiBlocProvider.
- Routing in lib/core/router/app_router.dart uses GoRouter with auth-based redirects.
- BLoCs (auth, transactions, budgets, categories, recurring, incomes, AI) orchestrate UI state and actions.
- Repositories in lib/data/repositories/ call HiveService first and FirebaseService second for offline-first behavior.
- FirebaseService persists documents in Firestore collections and uses FirebaseAuth for identity.
- AI path: AIChatBloc -> GeminiDatasource + GetFinancialContext(use case over transaction/income/budget repositories).

How To Run (Minimal)

- Install Flutter SDK (3.9.2+) and run: flutter pub get
- Configure Firebase for Android/iOS and keep lib/firebase_options.dart in place.
- Run code generation once: flutter pub run build_runner build --delete-conflicting-outputs
- Start app: flutter run

Not Found In Repo

- A secure runtime secret-management approach for the Gemini API key (current key is hardcoded in lib/features/ai/data/datasources/gemini_datasource.dart).
- Formal deployment/release pipeline documentation (CI/CD steps).