

# Mehfooz Accounts App - One Page Repo Summary

## WHAT IT IS

Mehfooz Accounts App is a Flutter app for account/ledger operations across company datasets. It combines local SQLite workflows, optional remote sync, and in-app PDF reporting.

## WHO IT IS FOR

Primary persona in docs: Not found in repo.

Inferred from screens and queries: accounting staff managing pending balances and cash books.

## WHAT IT DOES

- Multi-step auth: email check, password login/set password, forgot password, saved account chooser.
- Imports .sqlite/.db files (share intent, picker, iOS Open-In/iCloud) into app-managed storage.
- Maintains per-user local databases and restores the active user DB on launch.
- Home dashboard shows cash-in-hand, AccID=1 summary, and pending amounts by currency/company.
- Transactions screen supports search, debit/credit/date filters, balance mode, and detail panels.
- Generates report PDFs: balance, credit, debit, pending, ledger, and last-credit summary.
- Optional sync pulls server batches, applies local upserts/deletes, then sends ack; supports auto-sync.

## HOW IT WORKS (REPO EVIDENCE)

- UI: Flutter screens in lib/ui; state managed with Provider + ChangeNotifier viewmodels.
- Data layer: repositories run Drift SQL over Acc\_Personal, Transactions\_P, AccType, Company, Db\_Info.
- Local persistence: DatabaseManager activates per-user SQLite files under app documents.
- Integrations: AuthService and SyncService call HTTPS endpoints at kheloaurjeeto.net.
- Flow: login/restore DB -> viewmodels watch repository streams -> UI updates; import/sync mutate DB.

## HOW TO RUN (MINIMAL)

- 1) Install Flutter SDK (repo pubspec requires Dart SDK ^3.10.0) and platform toolchains.
- 2) From repo root: flutter pub get
- 3) Start app: flutter run
- 4) Environment variable setup instructions: Not found in repo.
- 5) Full backend/local server setup guide: Not found in repo.