

What It Is

TestKra is a React Native + Expo mobile app for practice test series, timed mock exams, and performance tracking. It supports auth, cloud sync, question library management, and Android Pro purchases with backend verification.

Who It's For

- Primary user/persona: Not found in repo.
- Inferred from test categories and catalog data: students preparing for UPSC, SSC, Banking, Railways, and State PCS exams.

What It Does

- Sign in with email/password or Google OAuth using Supabase Auth.
- Browse/search/filter test series and gate premium access through Pro status.
- Run timed tests with question palette, mark-for-review, pause/resume, and timeout auto-submit.
- Calculate scores (+2 correct, -0.66 incorrect), then show result breakdown and solution review entry points.
- Save questions as saved/wrong/revision items, with local persistence and cloud sync.
- Track progress with charts for score trend, subject accuracy, study time, coverage, and weak topics.
- Process Android in-app purchases and grant Pro only after server-side Google Play verification.

How It Works (Repo Evidence)

- UI: ``src/screens/*`` + reusable ``src/components/*``; navigation in ``src/navigation/AppNavigator.tsx``; theming in ``src/contexts/ThemeContext.tsx``.
- State: Zustand stores (``authStore``, ``testStore``, ``toastStore``) persisted via AsyncStorage.
- Services: Supabase client in ``src/lib/supabase.ts``; runtime env validation in ``src/config/runtimeConfig.ts``.
- Data model: Supabase tables include ``profiles``, ``tests``, ``questions``, ``attempts``, ``bookmarks``, ``test_progress``, and ``purchase_receipts``.
- Data flow: app hydrates stores -> session check + sync -> screen fetches/tests/actions -> optimistic local updates -> Supabase writes/upserts.
- Payments flow: ``PricingScreen`` -> ``googlePlayBilling.ts`` -> Edge Function ``verify-google-play-purchase`` -> Google Play API verify -> receipt upsert + ``profiles.is_pro=true``.

How To Run (Minimal)

- ``npm install``
- Create ``.env`` from ``.env.example``, then set: ``EXPO_PUBLIC_SUPABASE_URL`` and ``EXPO_PUBLIC_SUPABASE_ANON_KEY`` (required).
- Optional envs: ``EXPO_PUBLIC_ANDROID_BILLING_PRODUCT_IDS``, ``EXPO_PUBLIC_ENABLE MOCK_FALLBACK``.
- ``npm run start`` (or ``npm run android`` / ``npm run ios``).
- Backend billing/release setup details are in ``docs/release-smoke-checklist.md`` (deploy edge function + migrations).
- Complete end-to-end local backend provisioning guide: Not found in repo.