

Cosmiq App - One-Page Repo Summary

Evidence sources: README.md, index.html, src/App.tsx, src/hooks/*, src/pages/*, src/integrations/supabase/client.ts, supabase/function-manifest.json

What it is

Cosmiq is a Vite + React + TypeScript app described in repo metadata as a gamified self-improvement product for habits and quests with a celestial theme.

Repository code shows a web/PWA client plus a Capacitor iOS shell, backed by Supabase for auth, Postgres data, Edge Functions, and storage.

Who it is for

Primary persona: people who want a gamified personal planning and growth app with coach-like guidance (inferred from quest, mentor, companion, and pep-talk flows).

Explicit primary persona statement in repository docs: Not found in repo.

What it does

- Manages daily quests with inbox capture, scheduling, recurring tasks, subtasks, and streak handling.
- Provides mentor experiences: mentor selection, chat, daily pep talks, and reflection/recap screens.
- Runs a companion system with XP progression, evolution, story generation, postcards, and collections.
- Supports epics/campaigns/journeys planning with AI-assisted planning endpoints and orchestration.
- Integrates calendars across Google, Outlook, and Apple with send-only or full-sync options.
- Includes contacts and interaction logging, plus multi-tab navigation across mentor/inbox/journeys/companion.

How it works (architecture)

- Client: React app with React Router, React Query caching, and layered context providers in src/App.tsx.
- Data layer: supabase-js client handles auth/session and table access (for example daily_tasks, mentors, user_companion, contacts).
- Service layer: supabase.functions.invoke calls AI/service endpoints such as mentor-chat and ai-orchestrator.
- Backend surface: function-manifest allow-list contains 64 Edge Functions, including 3 scheduled functions.
- Realtime and runtime: RealtimeSyncProvider enables habits/epics/tasks sync; web uses service worker, iOS uses Capacitor plugins.
- Detailed production topology diagram and capacity/SLA metrics: Not found in repo.

How to run (minimal)

- Install dependencies: npm install
- Create env file: cp .env.example .env
- Set required vars: VITE_SUPABASE_URL, VITE_SUPABASE_PUBLISHABLE_KEY, VITE_SUPABASE_PROJECT_ID, VITE_NATIVE_REDIRECT_BASE
- Start local app: npm run dev