Airese TAS Screener — Consolidated Product Requirements Document

Version: 1.0 (Consolidated Master PRD)
Status: Living document
Last Updated: 2026-06-16
Owner: Airese Product Team
Audience: Mixed — Part I (Product) is written for non-technical stakeholders; Part II (Technical) is written for engineering.

> This document consolidates the product vision, market context, current state, roadmap, and technical architecture for the Airese TAS Screener app into a single master reference. It folds in the three in-flight feature specs — Mandarin localization, hybrid storage (MMKV + SQLite), and Explore article detail — and the Singapore launch plan. Individual specs remain authoritative for implementation detail; this document is the connective overview.
>
> **Source documents:** `docs/prd/features/*`, `docs/marketing/Airese-launch-plan-sg-2026.md`, `docs/ARCHITECTURE.md`, `docs/STATE.md`, `docs/NAVIGATION.md`, `docs/api/overview.md`, `docs/data/models.md`, `docs/deployment/*`, `docs/testing/strategy.md`, `CLAUDE.md`.

Part I — Product (Non-Technical)

1. Executive Summary

Airese is a mobile app (iOS + Android) that screens for snoring and sleep-disordered breathing using only the phone's microphone — no extra hardware required. A user places their phone by the bed, sleeps, and wakes up to an analysis report showing their snoring patterns, blood-oxygen (SpO2) readings, and a sleep risk indicator.

Airese is **co-developed by The Air Station (TAS) and Diametos**, combining TAS's regional retail/service footprint with Diametos's research-backed snoring analysis algorithm. It is positioned as a **consumer health-awareness tool, not a medical device** — it screens and informs; it does not diagnose or treat.

The product launches in **Singapore (soft launch August 2026)** on a **freemium model**: the first two sleep analyses are free, after which users convert to one of three paid tiers (Silver, Gold, Platinum). An optional BLE wearable band is supported but not required for the core experience.

Why now: Sleep-disordered breathing is massively under-screened. A Singapore community study found 30.5% moderate-to-severe prevalence in adults 21–79, with **91% of OSA cases previously undiagnosed**. Airese removes the two biggest barriers to screening — cost and clinic friction — by making the first screening free and hardware-free.

---

2. Problem & Market Opportunity

The problem

- ~40% of adults globally snore habitually; ~60% of men over 40 in Singapore snore.
- Most people who snore never get screened — they don't think it's "serious enough" for a clinic sleep study, which is expensive, slow, and inconvenient.
- Untreated obstructive sleep apnea (OSA) carries real cardiovascular and metabolic risk, yet the overwhelming majority of cases go undiagnosed.

The opportunity (Singapore beachhead)

Factor	Relevance
~88% smartphone penetration	High install potential, zero-hardware product fits
English-speaking, health-conscious	Low localization friction at launch; strong wellness-app receptivity
Dense HDB housing	Shared bedrooms amplify partner-complaint triggers
Mature digital ad market	Meta, TikTok, Google UAC all have proven SG audiences
Accessible private healthcare	GPs and ENT clinics are viable B2B referral partners

Globally, OSA affects an estimated **936 million adults aged 30–69**, peaking ages 36–60 — the same band as Airese's primary persona. Singapore is the validated beachhead; SEA expansion is the follow-

on.

---

3. Product Vision & Positioning

Positioning statement:

> For adults who suspect they snore but haven't taken action, Airese is a sleep screening app that gives real answers overnight — using only your phone. Unlike clinic-based sleep studies, Airese is instant, private, and free to start.

Messaging pillars:

Pillar	Message
Zero friction	"Just place your phone. Sleep. See your results."
Research-backed	"Built on certified, clinically-researched snoring analysis."
Free to start	"Your first two nights are on us."
Know before it's a problem	"Snoring can be more than an annoyance."

Compliance guardrails (hard rule across product, UX copy, and marketing):

- ✅ Say: "snoring analysis", "sleep screening", "research-backed", "understand your sleep patterns".
- ❌ Never say: "diagnose", "detect sleep apnea", "medical device", "treatment", or anything implying it replaces a clinical sleep study.

---

4. Target Users

Persona	Who	Trigger	Primary channels
The Self-Aware Snorer (primary)	Adults 36–60	Partner complaints, waking unrefreshed	TikTok, Instagram, Meta
The Health-Conscious Tracker (secondary)	Adults 30–45, wearable users	Wants complete sleep data	Google UAC, Instagram
The Concerned Partner (tertiary)	Spouse of a snorer	Disrupted sleep, worry for SO	Facebook, couples content
Clinic-Referred (B2B)	GP/ENT-referred patients	Doctor recommendation	Clinic referral codes

---

5. Commercial Model

Freemium funnel — the paywall lands **after** the user has experienced value twice:

```
Download → Auth (Patient ID) → 1st recording (free) → Report
                             → 2nd recording (free) → Report
                             → Paywall → Choose tier → Paying subscriber
```

Tier	Price (tentative)	Core features	TAS perks
Freemium	Free	Audio tracking, SpO2 charts, **no historical data**	None
Silver	SGD 8/mo	+ Historical data, trend tracking, gamification, loyalty card	5% off accessories
Gold	SGD 39/mo	+ GenAI coach, community, podcast, CPAP adoption pathway	15% off all branches
Platinum	TBD	All Gold features	15% off + free annual servicing + priority fast lane

> Pricing is tentative pending management confirmation. Tier concept and feature differentiation are approved for planning.

Conversion targets: Install→1st recording 60%+ · 1st→2nd 50%+ · 2nd→paid 8–15%.

---

6. Current Product State (as of 2026-06-16)

Implemented / working:

- User authentication (signup, login, password reset, social login)
- 6-step onboarding flow
- BLE device pairing and scanning (optional wearable band)
- Microphone permissions and audio recording
- On-device WAV→FLAC audio encoding (local Expo module)
- HealthKit integration (iOS — SpO2 reading)
- Tab navigation: **Sleep, Progress, Explore, Alerts, Account**
- i18n infrastructure (English + Simplified Chinese pilot — in progress)

In progress:

- Sleep tracking, morning report & insights
- Progress tracking and charts
- Mandarin localization pilot (`feature/mandarin-localization`)
- Hybrid storage migration (notification history → SQLite)
- Explore article-detail reading screen

---

7. Roadmap & In-Flight Initiatives

Three feature initiatives are active. Full specs are linked; summaries and the **key cross-feature dependency** are below.

Initiative	Spec	Status	One-line summary
Mandarin (zh-CN) localization	`features/mandarin-localization/mandarin-localization.md`	Draft — Phase 1	i18n infra + fully-translated pilot flow; first-launch language picker; Explore tab disabled in Phase 1
Hybrid storage (MMKV + SQLite)	`features/hybrid-storage-sqlite.md`	Draft	Move unbounded notification history from an MMKV JSON blob to a queryable SQLite table; keep MMKV everywhere it already works
Explore article detail	`features/explore-article-detail.md`	In progress	Tap an article card image/title to open a full reading screen

> ⚠️ **Reconciliation required before production.** Phase 1 of the Mandarin localization PRD **disables the Explore tab** (greyed out, non-interactive). The Explore article-detail story depends on a *working* Explore tab. These two cannot both ship to the same production build as-is — sequence them, gate by market/locale, or scope a compromise (e.g., Explore enabled in English builds, disabled in zh-CN pilot). **This decision is currently open.**

---

8. Singapore Launch (August 2026 Soft Launch)

- **Budget:** SGD 10,000 — Meta 35% / TikTok 25% / Google UAC 20% / Affiliate 15% / PR 5%.
- **Strategy:** Awareness-led, social-content-driven, zero-friction free tier.
- **Parallel B2B track:** GP/ENT clinic referral codes (extended free tier per code, monthly usage reports to partner clinics).
- **Launch-month KPIs:** 1,000+ installs · 60%+ first-recording completion · 4.0+ store rating · <SGD 5 blended CPI · 50–100 paid conversions.
- **Optimization KPIs (Sep–Oct):** D30 retention 20%+ · freemium→paid 8–15% · MAU 500+ · 100+ clinic referral installs.

Full plan, creative strategy, and channel playbooks: `docs/marketing/`.

> **Note:** `docs/marketing/Airese-launch-plan-sg-2026.md` is a separate working scaffold owned by the marketing team and still reflects an earlier July timeline. This PRD's **August 2026** soft-launch date is the current source of truth; the marketing scaffold is intentionally left untouched for the marketing team to maintain on their own cadence.

---

9. Product-Level Risks

Risk	Mitigation
Low install→recording conversion	Simplify onboarding; surface first-recording prompt immediately
App Store rejection (health claims)	Strict no-medical-claims copy; "screening tool, not diagnosis" framing
Low freemium→paid conversion	A/B test paywall; lead with "don't lose your history"
False positives → negative reviews	Set expectations in onboarding
AI medical translations inaccurate (zh-CN)	Human-review gate on medical-domain keys before release (see §15)
Explore-tab-disabled vs Explore-detail conflict	Resolve sequencing before any production tag (see §7)
Diametos branding alignment	Marketing to confirm co-development presentation & certification language

Part II — Technical

10. Architecture Overview

Pattern: File-based routing (Expo Router) with feature-grouped React Context providers and shared CVA-styled UI primitives. Dark-theme only (`#121212`).

Providers wrap the app in `app/_layout.tsx` (current order, outermost → innermost): Bluetooth → Keyboard → GestureHandlerRoot → BottomSheet → Stack. Auth and the new i18n/Database providers slot into this tree (see §15, §16).

Key entry points:

- `app/_layout.tsx` — root layout, providers, `<Stack>`, `<StatusBar>`
- `expo-router/entry` — app entry (`package.json`)
- `app/(tabs)/_layout.tsx` — bottom tab bar (5 tabs)

---

11. Technology Stack

Layer	Choice
Language	TypeScript 5.9.3
Framework	Expo SDK 54.0.22 · React Native 0.81.5 · React 19.1.0 (React Compiler experimental)
Routing	Expo Router 6.0.14 (file-based)
Styling	NativeWind 4.2.1 (Tailwind), dark-first
State	little-state-machine 5.0.1 (persisted), React Context, react-hook-form 7.66 + zod 4.1
BLE	react-native-ble-plx 3.5.0 (background enabled, patched)
Audio	expo-audio 1.0.14, react-native-audio-record 0.2.2, on-device WAV→FLAC local module
Storage	react-native-mmkv 4.2.0, expo-secure-store 15.0.8 (+ expo-sqlite incoming, see §16)
Animations	react-native-reanimated 4.1.3, rive-react-native 9.6.2
Testing	Jest 29.7 + @testing-library/react-native 12.9 (no tests authored yet)

Native modules requiring prebuild/EAS: ble-plx, mmkv, rive, reanimated, gesture-handler. Patches applied via `patch-package` (`postinstall`): `react-native-ble-plx`, `react-native-date-picker`.

---

12. App Structure & Navigation

```
app/
├── (tabs)/                      # tab screens
├── login/  register/  onboarding/
components/
├── ui/                          # primitives
├── sheets/                      # bottom-sheet modals
lib/
├── api/                         # fetch client + API modules
├── providers/                   # context providers
├── hooks/                       # custom hooks
├── contexts/  variants/  storage/
assets/  setup/  patches/  db/
```

Key directories:

Path	Contains
`app/`	Expo Router screens — file/folder names map to routes
`app/(tabs)/`	The 5 tabs: Sleep · Progress · Explore · Alerts · Account
`app/login/`, `register/`, `onboarding/`	Auth & onboarding flows — each scoped by a `_layout.tsx` Stack
`components/ui/`	Primitives: button, input, text, heading, checkbox, otp-input, divider
`components/sheets/`	`@gorhom/bottom-sheet` modals
`lib/api/`	Fetch client + typed API modules
`lib/providers/`	Bluetooth, BottomSheet, Auth (+ I18n, Database incoming)
`lib/hooks/`	use-auth, use-bluetooth, use-permission (+ use-language incoming)
`lib/contexts/`, `variants/`, `storage/`	Typed contexts, CVA style variants, MMKV/SQLite wrappers
`db/`	Drizzle schema + migrations (incoming — see §16)

- File/folder names map to routes; `_layout.tsx` defines Stack/Tabs containers; `(tabs)` is a route group that doesn't affect the URL.
- Declarative nav via `<Link href=...>`; imperative via `useRouter()`.
- Path alias `@/*` → project root.
- Conventions: functional components + TS only, named exports, `[ComponentName]Props`, kebab-case files, import order per `CLAUDE.md`.

---

13. State Management & Storage

Concern	Mechanism	Location
Services w/ lifecycle (BLE, bottom sheet)	React Context + provider	`lib/providers/`, `lib/contexts/`
Auth state	AuthContext	`lib/contexts/auth-context.ts`, `lib/providers/auth-provider.tsx`, `lib/hooks/use-auth.ts`
Multi-step form drafts (login/register/onboarding)	little-state-machine, persisted	`setup/storage.ts` (MMKV adapter)
Form state	react-hook-form + zod	per-screen
Fast KV / preferences / locale / article cache	react-native-mmkv	`lib/storage/*`, `lib/i18n/`
Secrets (sessionId)	expo-secure-store (AES-256)	`lib/secure-storage.ts`
Structured/queryable (notification history)	**expo-sqlite + drizzle-orm (incoming)**	`db/`, `lib/storage/alerts-storage.ts`

Storage rule of thumb (post-hybrid migration): MMKV for flat KV and anything written synchronously during active recording; SQLite for relational/queryable data. See §16.

---

14. API Integration & Data Models

Client: custom fetch-based, `lib/api/client.ts`. Base URL `https://backend.test-atlantic-snorefox.diametos.com/api/app`. `USE_MOCK_API` toggle for offline dev.

Auth: cookie sessions (`sessionId`, HttpOnly/Secure/SameSite=Strict) extracted from `Set-Cookie`, stored in SecureStore, auto-injected on requests; 401 → logout + session clear. 15s timeout. `APIError` carries `statusCode`, `message`, `error`, `reqId`.

Modules: `authAPI` (signup/login/logout/verify-email/password-reset/social-login), `userAPI` (profile, push token), `measurementAPI` (sleep measurements CRUD).

Core models (full defs in `lib/api/client.ts`, summarized in `docs/data/models.md`):

- `UserProfile` — id, email, `language: "EN" | "DE"` (zh-CN incoming, see §15), timezone, isEmailVerified, createdAt, consentPreferences.
- `Measurement` — status lifecycle `CREATED_WITH_DELAY_TIMER → RECORDING_AND_UPLOADING → RECORDING_FINISHED_UPLOADING → ANALYZING → ANALYZED`; optional `report` (summary, risk, apnea/snore charts, SpO2, sleep events), `deviceState` (battery, freeBytes, wifi/cellular flags).

---

15. In-Flight Spec — Mandarin (zh-CN) Localization

Goal: Add Simplified Chinese support so Mandarin-speaking users (SG, Greater China, Taiwan diaspora) can run the full screening flow natively. Phase 1 = i18n infrastructure + fully-translated **pilot flow**.

Stack: `i18next` + `react-i18next`, MMKV-persisted locale (`app.language` in a `language-storage` instance), Noto Sans SC font for CJK. `expo-localization` installed now for Phase 2 auto-detect (not used in Phase 1 logic).

Key modules: `lib/i18n/index.ts` (init, `getStoredLocale`, `setStoredLocale`), `lib/providers/i18n-provider.tsx` (hydrates from MMKV before render), `lib/hooks/use-language.ts`, `app/language-picker/` (first-launch picker), `locales/{en,zh-CN}.json` + `zh-CN.review.json`.

Behavior:

- First launch with no stored locale → language picker (English / 简体中文, no pre-selection, no skip). Subsequent launches skip it. Existing upgraders default to English silently.
- Account → Profile → Language row switches locale live (no restart) via `setStoredLocale()`.
- `I18nProvider` mounts **outermost** so even auth error strings resolve.
- Missing key → **always falls back to English**, never shows a raw key (dev-mode console warning).
- **Explore tab greyed out (30% opacity) and non-interactive in Phase 1** via a `disabled` prop on `components/tab-button.tsx`.

Quality gates: AI-drafted `zh-CN.json` flagged per-key in `zh-CN.review.json`; **CI fails if any medical-domain key** (`snoring*`, `apnea*`, `rei*`, `ahi*`, `spo2*`, `oxygen*`, `risk*`) is still `ai-drafted` at release-tag time. NFRs: <50ms init, <100KB gzipped bundle delta.

Delivery: 4 PRs — (1) infra, (2) picker + Explore-disabled, (3) string extraction + zh-CN translations, (4) font + visual QA. ~1.5 weeks.

> **Note the §7 conflict:** the Explore-disabled requirement collides with the Explore article-detail story.

---

16. In-Flight Spec — Hybrid Storage (MMKV + SQLite)

Problem: Notification history in `lib/storage/alerts-storage.ts` is an unbounded serialized JSON array in MMKV — every read deserializes the whole list, every mutation rewrites the full blob, and type-safe querying is impossible.

Decision: Hybrid. Add SQLite (`expo-sqlite` + `drizzle-orm`) for relational/queryable data; **keep MMKV everywhere it already works** (form state, sleep-storage synchronous recording writes, profile blob, locale, article cache, secure store). No MMKV stores removed.

Moves to SQLite: `NotificationEntry[]` history → `notifications` table (indexed on `timestamp DESC` and `read`). `AlertSettings` **stays in MMKV** (single object, no querying).

New structure: `db/schema.ts` (Drizzle), `db/migrations/`, `db/index.ts` (connection singleton), `lib/providers/database-provider.tsx` (runs `useMigrations` on startup, **outermost** provider). One-

time MMKV→SQLite migration runs once on first launch, guarded by a "migrated" flag + `onConflictDoNothing()`, deletes the old MMKV key after success.

Primary ripple effect: notification functions become **async** (SQLite is async). Audit all `alerts-storage` callers and add `await`; TypeScript surfaces every missing one. Requires an **EAS native rebuild** (new native module) — no OTA-only deploy.

Out of scope: sleep-session tables (next story), SQLCipher encryption, cloud sync.

---

17. In-Flight Spec — Explore Article Detail

Story: Tap an article card's image or title on the Explore tab → open a full reading screen.

- Target route: `/(tabs)/explore/[id]`; params `id`, `title`, `content`, `pictureUrl`, `publishedDate`, `readingTime`.
- Only the image and title `Pressable` areas navigate. Like / bookmark / share buttons fire independently and must **not** propagate to navigation.
- Article with no cover image → title tap still navigates (placeholder box is not a nav target).
- Out of scope: bookmarking, sharing, like-count persistence (separate stories).

> **Blocked-by:** depends on a working Explore tab — see the §7 reconciliation with Mandarin Phase 1.

---

18. CI/CD & Deployment

CI (`ci.yml`): runs on PR/push to `main`, `staging`, `build`, `development`. Steps: `npm ci` → `tsc --noEmit` → `expo lint` → `jest --coverage --ci`. (Jest passes trivially today — no test files yet, gate is wired and ready.)

CD: manually dispatched EAS cloud builds, one workflow per profile. **No auto-submit to stores** (manual upload), **no auto-trigger on merge**.

Profile	Distribution	Notes
`development`	internal	dev client, iOS simulator
`build` (QA) / `staging`	internal	device, Android APK
`production`	store	only `.aab`-producing profile
`preview`	internal	`npm run build:preview`
`preview-apk`	internal	orphaned — no workflow, local only

Branch → environment: `development` / `build` / `staging` / `main` map to like-named profiles; promote by manually dispatching the matching build after merge. Only secret needed: `EXPO_TOKEN`.

Platform config: iOS bundle / Android package `com.theairstation.airese`; HealthKit + Apple Auth (iOS); edge-to-edge (Android). API config currently hardcoded — future `EXPO_PUBLIC_API_BASE_URL` / `EXPO_PUBLIC_USE_MOCK_API`.

19. Testing Strategy

- **Framework:** Jest + `jest-expo`, `@testing-library/react-native`. Co-located `*.test.tsx`.
- **Target pyramid:** 70% unit / 20% integration / 10% E2E (E2E only on request).
- **Current state:** no tests authored yet — TDD to be established. The Mandarin spec defines the first concrete unit/integration suites (i18n init, persistence, fallback, language hook, disabled tab button). The hybrid-storage spec adds migration and async-caller coverage.
- **Manual:** `docs/TESTING.md`; preview-build OTP `1234`.

---

20. Platform & Permissions

Expo-first: prefer `app.json` / `app.config.*` for permissions and Info.plist keys; drop to native `ios/` / `android/` only for what config can't express, and document it.

- **BLE:** ble-plx + Android 12+ permission model; plugin block in `app.json`; native patch in `patches/`.
- **Microphone:** `RECORD_AUDIO` + expo-audio plugin prompt strings.
- **HealthKit (iOS):** SpO2 read keys.
- Permission flow: `lib/constants.ts` (Platform.select mappings), `lib/hooks/use-permission.ts` (requests + AppState re-check), onboarding screens (`2-allow-microphone`, `3-connect-band`), `lib/providers/bluetooth-provider.tsx` (guards BLE ops behind permission checks).

After native module/patch changes: `npm install` (runs `patch-package`) → `npx expo prebuild --clean` → rebuild on device.

---

21. Design System (reference)

Dark-only, `#121212` base. Full token tables in `docs/architecture/design-system.md` and `.claude/rules/figma-design-system.md` (Figma file `jjM5AP6ljMoUHWJ2RwcqgF`).

- **Never hardcode** hex/font sizes — use Tailwind tokens (`bg-gray-900`, `text-purple`, `text-4xl`, `font-montserrat-bold`).
- Text hierarchy: `text-white` → `text-gray-100` → `text-gray-300`. Accent: `bg-purple` (`#8400E5`).
- Status colors: green (excellent) / light-blue (fair) / yellow (low) / red (critical), each with a tint-background variant.
- CJK text (zh-CN) routes through a `<LocalizedText>` wrapper applying Noto Sans SC (see §15).
- Always check `components/ui/` before adding a primitive; never install new icon packages (icons live in `assets/icons/` as SVG).

---

22. Open Decisions

#	Decision	Owner	Blocking
1	**Explore tab disabled (zh-CN Phase 1) vs Explore article-detail** — sequence, gate-by-locale, or compromise	Product + Eng	Production tag for either feature
2	Final subscription pricing (Silver/Gold/Platinum)	Management	Paywall copy, store listing
3	Diametos branding alignment + certification language for marketing	Marketing	Consumer-facing materials
4	Move API config to env vars (`EXPO_PUBLIC_*`)	Eng	Multi-environment builds
5	Establish first test suites (currently zero)	Eng	CI coverage gating
6	Add `eas submit` / auto-trigger builds on merge	Eng	Release automation
7	**Backend ownership model** — Airese-owned backend (Jonas's proposal, for ZohoCRM lead capture) vs the contractual frontend-only boundary; dual-backend / shared-VPC architecture with Diametos. Not yet discussed with Diametos.	Jonas + Gary + Diametos	All non-audio features: lead capture, CRM sync, loyalty/gamification, B2B attribution, analytics, future products

> **Architecture note (Decision #7):** The Diametos contract is **Diametos = backend, Airese = frontend** — sound for the audio-algorithm core, but it means any feature that *doesn't* touch the audio algorithm currently has no Airese-owned server to run on. Jonas proposes standing up our own backend (starting with sign-up → ZohoCRM lead capture). The open technical question is how two backends coexist — likely a **shared/peered VPC** so the Airese and Diametos servers can communicate securely. This is a prerequisite for most post-launch tier features (see roadmap §2.6).

---

23. Glossary

Term	Meaning
TAS	The Air Station — co-developer, regional retail/service partner
Diametos	Technology partner; research-backed snoring analysis algorithm
OSA	Obstructive Sleep Apnea
SpO2	Blood-oxygen saturation (read via HealthKit on iOS)
REI / AHI	Respiratory Event Index / Apnea-Hypopnea Index — sleep-event severity measures
BLE	Bluetooth Low Energy — optional wearable band connectivity
MMKV	Fast native key-value storage
CPI / CPAP	Cost Per Install / Continuous Positive Airway Pressure (therapy pathway)
UAC	Universal App Campaigns (Google)
EAS	Expo Application Services (cloud build/submit)
i18n / zh-CN	Internationalization / Simplified Chinese locale

---

24. Source Documents

- **Features:** `docs/prd/features/mandarin-localization/mandarin-localization.md`, `docs/prd/features/hybrid-storage-sqlite.md`, `docs/prd/features/explore-article-detail.md`
- **Index:** `docs/prd/overview.md`
- **Marketing:** `docs/marketing/Airese-launch-plan-sg-2026.md` (+ `channels/`)
- **Architecture/technical:** `docs/ARCHITECTURE.md`, `docs/STATE.md`, `docs/NAVIGATION.md`, `docs/PLATFORM.md`, `docs/DEPENDENCIES.md`, `docs/api/overview.md`, `docs/data/models.md`
- **Ops:** `docs/deployment/ci-cd.md`, `docs/deployment/environments.md`, `docs/testing/strategy.md`, `docs/runbooks/incidents.md`
- **Design:** `docs/architecture/design-system.md`, `.claude/rules/figma-design-system.md`
- **Project guide:** `CLAUDE.md`

Version: 1.0 · **Last Updated**: 2026-06-16 · **Next review:** pre-launch (June 2026)