YACHECK SYSTEM DOCUMENT

Technology Stack & Scale Readiness

สรุปเทคโนโลยีทั้งระบบ สถาปัตยกรรมปัจจุบัน
และแนวทาง Index Sync สำหรับการขยายระบบ

4	13	5
ชั้นระบบหลัก Mobile / Admin / Platform / Prototype	ตารางฐานข้อมูลหลัก ครอบคลุม clinical, user, caregiver, admin	Supabase Edge Functions auth, recovery, delete, messaging

สรุปสั้นที่สุด	Production path หลักคือ Expo/React Native แบบ local-first เชื่อม Supabase โดยตรง ส่วนจุดที่ต้องเพิ่มก่อน scale คือ versioned catalog, incremental sync, pagination และ local search index

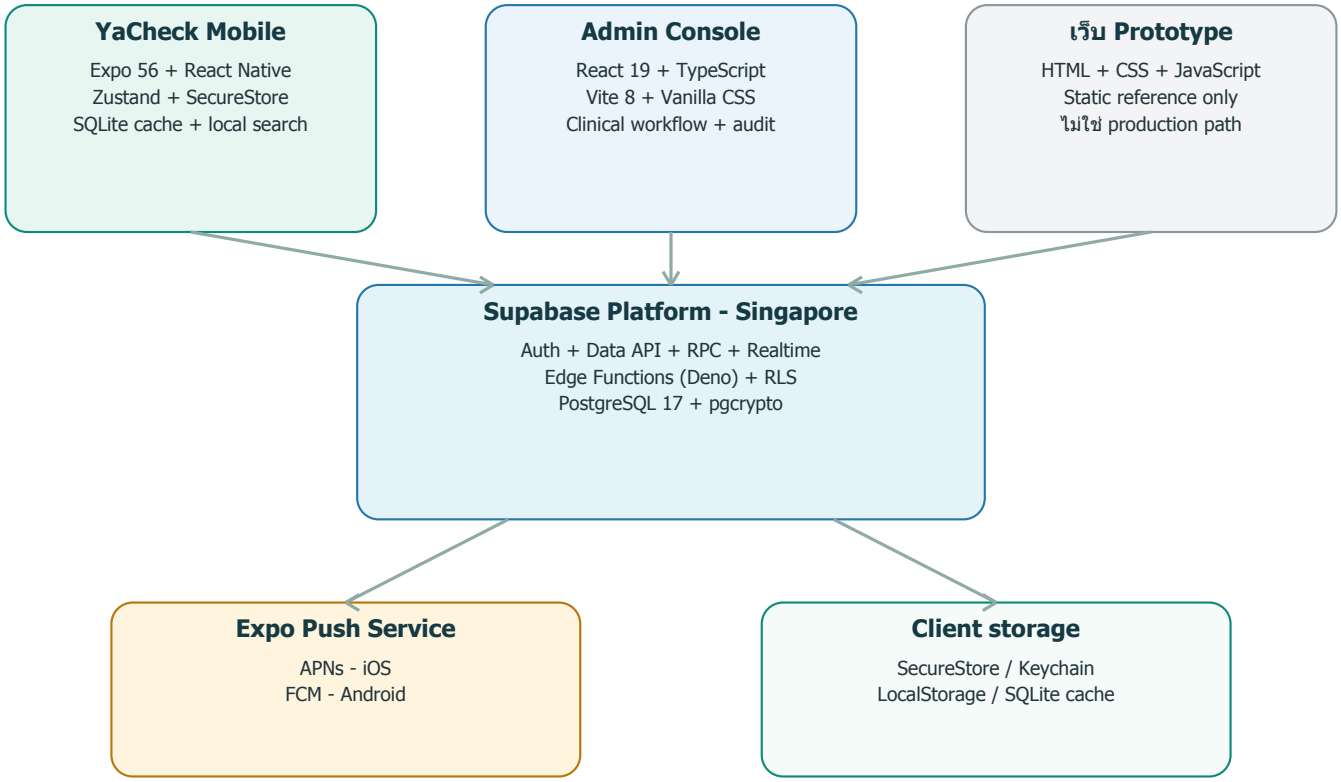
จัดทำจาก source code และ configuration ใน repository

สถานะ ณ วันที่ 20 กรกฎาคม 2026 | เอกสารภายในสำหรับการวางแผนระบบ

SECTION 01

ภาพรวมสถาปัตยกรรม

ระบบปัจจุบันเป็น client-to-Supabase architecture ไม่มี Express, NestJS หรือ application server แบบดั้งเดิมคั่นกลาง



จุดเด่น	การค้นหายาเกิดบนอุปกรณ์ จึงไม่ยิงฐานข้อมูลทุกครั้งที่ผู้ใช้พิมพ์ และยังทำงานจาก cache/fallback เมื่อเครือข่ายไม่พร้อม
ข้อจำกัด	เมื่อผู้ใช้เปิดแอป ระบบยังดึง clinical catalog ทั้งสามชุดใหม่ทั้งหมด ทำให้ภาระหลักอยู่ที่ startup sync ไม่ใช่จำนวนครั้งที่พิมพ์ค้นหา

SECTION 02

Mobile application stack

ฐานพัฒนาหลักของ YaCheck ใช้ Expo และ React Native พร้อม native projects สำหรับ iOS และ Android

หมวด	เทคโนโลยี
Framework	Expo 56.0.16, React Native 0.85.3, React 19.2.3
Language	TypeScript 6.0.3 แบบ strict และ TSX
Navigation	Expo Router 56.2.15, file-based routes, Stack และ Tabs
State	Zustand 5.0.14 พร้อม persisted application state
Runtime	Hermes, React Native New Architecture, Fabric, React Compiler
Backend client	Supabase JS 2.57.4 และ React Native URL polyfill
Web target	React Native Web 0.21.0 และ static web output

Expo modules และ native capability

Capability	Package	หน้าที่
Camera	expo-camera	สแกน barcode/QR และฉลากยา
Notification	expo-notifications	แจ้งเตือนกินยา, badge, push, deep link
Secure data	expo-secure-store	session และ health state บน native
Local cache	expo-sqlite	localStorage compatibility และ catalog JSON cache
Device UX	expo-haptics / speech	สั่นตอบสนองและอ่านข้อความ
Media/UI	expo-image / font	ภาพ, asset และ font integration
Linking	expo-linking	deep link และการโทรศัพท์

iOS

Swift AppDelegate, Xcode/CocoaPods, iOS 16.4+, Hermes, APNs entitlements และ schemes แยก Dev/Production

Android

Kotlin, Gradle, AndroidX, Hermes, New Architecture, Edge-to-edge, Fresco และ Camera permission

SECTION 03

Local data, backend และ database

ข้อมูลในเครื่องและ cloud แบ่งหน้าที่กันแบบ local-first โดย Supabase เป็น backend กลางของ YaCheck/MaCheck

Local state และ storage

- Zustand เก็บ profile, cabinet, dose history, water history, auth state และ archived medicine
- SecureStore แบ่งข้อมูลเป็น chunk 1,800 ตัวอักษร และ migrate ข้อมูลเก่าจาก AsyncStorage
- บน Web ใช้ browser localStorage; บน native ใช้ Keychain/Keystore ผ่าน SecureStore
- Clinical catalog cache ยังเป็น JSON ก้อนเดียว ไม่ใช่ relational SQLite tables หรือ FTS index

Supabase platform

บริการ	หน้าที่
Supabase Auth	JWT session, refresh token rotation และ password hash
PostgreSQL 17	ข้อมูล clinical, user health, caregiver, admin และ audit
Data API	PostgREST ผ่าน Supabase JS client
RPC	PL/pgSQL functions สำหรับสิทธิ์, workflow และ sensitive operations
Edge Functions	Deno TypeScript สำหรับ username auth, recovery, account deletion และ push
Realtime	Postgres Changes สำหรับ caregiver_nudges
Extension	pgcrypto สำหรับ UUID/cryptographic database support

กลุ่มข้อมูลหลัก

Schema/Domain	Tables
Private	account_handles, auth_rate_limits
Clinical	medications, drug_interactions, food_interactions
User health	app_profiles, patient_medications, dose_events, activity_logs
Caregiver	caregiver_links, caregiver_invitations, caregiver_nudges, user_push_tokens
Admin	admin_members, clinical_change_log

SECTION 04

Authentication, security และ messaging

ระบบใช้ username gateway เพื่อหลีกเลี่ยงการเก็บอีเมลหรือเบอร์โทรจริงในขั้นตอนสมัครบัญชี

ขั้น	Authentication flow
1	ผู้ใช้ส่ง username และ password ไป Edge Function
2	ระบบสร้าง internal random email และบัญชีใน Supabase Auth
3	private.account_handles เก็บ mapping ที่ client อ่านไม่ได้
4	Edge Function คืน access token, refresh token และ recovery code
5	Mobile เก็บ session ใน SecureStore และ refresh token อัตโนมัติ

Security controls

ชั้น	มาตรการ
Database	Row Level Security, auth.uid(), explicit grants และ foreign keys
Caregiver	Consent ก่อนอ่านข้อมูล, invitation expiry และ revoke ได้ทั้งสองฝ่าย
Clinical	Draft - Review - Published - Archived, source requirement และ dataset version
Admin	owner, clinical_editor, clinical_reviewer, auditor
Audit	Database trigger เขียน clinical_change_log
Auth protection	Rate limit, SHA-256, secret pepper และ constant-time comparison
Web	CSP, noindex, frame denial และ restrictive permissions policy

Messaging path

Caregiver -> Supabase RPC/Edge Function -> caregiver_nudges -> Realtime ขณะเปิดแอป และ Expo Push Service -> APNs/FCM เมื่อใช้ remote push

SECTION 05

Sync และ Search ปัจจุบัน

ระบบทำงานได้สำหรับ MVP แต่ยังใช้ full snapshot หลายจุด จึงต้องปรับก่อนรองรับข้อมูลหรือผู้ใช้จำนวนมาก

ส่วน	วิธีปัจจุบัน	ผลเมื่อ scale
Clinical pull	โหลด medications, drug_interactions และ food_interactions ทั้งหมดทุก session	เสียงโหลดซ้ำและเกิน API row limit
Catalog cache	อ่าน cache ก่อน แล้วเขียน JSON ทับทั้งก้อน	ไม่มี transaction/checksum ต่อ record
Medicine search	Normalize NFKC + includes + sort ใน memory	ต้นทุนเพิ่มตามจำนวนยา
Pair lookup	วน interactions.find ทุกครั้ง	ไม่มี O(1) pair index
User pull	ดึง cabinet และ dose_events ทั้งหมด	ประวัติโตตามอายุบัญชี
User push	Upsert profile, cabinet และ dose snapshot	เขียนข้อมูลเดิมซ้ำ
Admin list	โหลด clinical table แล้ว filter ฝั่ง browser	ไม่มี server pagination/search

Database indexes ที่มี

- patient_medications(user_id, updated_at desc)
- dose_events(user_id, event_date desc) และ activity_logs(user_id, occurred_at desc)
- clinical status/reviewed_at indexes และ audit indexes
- primary keys, unique constraints, caregiver invitation/push/message indexes

สิ่งที่ยังไม่มี Full-text/trigram search index, GIN index, SQLite FTS, catalog release manifest, sync cursor index, tombstone feed และ local interaction Map

SECTION 06

Target Stack สำหรับ Index Sync และ Scale

ต่อยอดจาก PostgreSQL/Supabase เดิมได้ โดยยังไม่จำเป็นต้องเพิ่ม Redis, Kafka, Elasticsearch หรือ Kubernetes ในระยะแรก

Capability	Stack	หน้าที่
Release manifest	clinical_catalog_releases	version, checksum, published_at และ snapshot URL
Change feed	clinical_catalog_changes	record key, version, operation และ tombstone
Server search	PostgreSQL pg_trgm + GIN	ค้นชื่อไทย/อังกฤษแบบ substring/fuzzy พร้อม limit
Pagination	Keyset cursor	ใช้ updated_at + stable ID แทน offset
Local database	SQLite tables	แยก medicine, interaction, food และ metadata
Local text index	SQLite FTS5	ค้นหาเร็วโดยไม่ scan array ทั้งหมด
Pair index	Map	ตรวจคู่ยาแบบ O(1)
Distribution	Supabase Storage/CDN	versioned compressed catalog snapshot
Validation	Checksum + transaction	apply สำเร็จทั้งชุดหรือ rollback

Target sync flow

1	2	3	4	5
เช็ก manifest	เทียบ version	โหลด delta/snapshot	verify checksum	SQLite transaction

ผลลัพธ์ ผู้ใช้ส่วนใหญ่เรียกเพียง manifest ขนาดเล็ก; ดาวน์โหลดข้อมูลเมื่อ version เปลี่ยน; ค้นหาบนอุปกรณ์ได้เร็ว; server ควบคุม pagination และรองรับ rollback

SECTION 07

Build, Deploy และ Operations

ระบบมี build/deploy path แล้ว แต่ automated quality gates และ observability ยังเป็นงานค้าง

ส่วน	Stack
Mobile build	npm, Expo CLI, EAS Build, EAS Submit, Expo Dev Client
iOS build	Xcode, CocoaPods, Dev/Production schemes และ signing entitlements
Android build	Gradle, Kotlin, AndroidX และ EAS credentials
Admin build	TypeScript project build + Vite 8
Admin hosting	Vercel SPA rewrite พร้อม CSP/security headers
Backend deploy	Supabase CLI, SQL migrations และ Edge Function deploy
Data tooling	Python extraction/audit, SQL dry-run และ Excel/PDF import artifacts
Quality tools	TypeScript strict, ESLint 9, Expo Doctor

ช่องว่างด้าน production operations

หมวด	สถานะ
Testing	ยังไม่มี unit, integration และ E2E test suite ใน repo
CI/CD gates	ยังไม่มี GitHub Actions หรือ pipeline บังคับ lint/typecheck/test/build
Observability	ยังไม่มี crash reporting, APM, structured log aggregation และ alerting
Load testing	ยังไม่มี benchmark สำหรับ catalog size, concurrent startup และ p95 latency
Disaster recovery	ยังต้องกำหนด automated backup, restore drill และ catalog rollback
Infrastructure	ไม่มี Docker, Terraform หรือ Kubernetes; ปัจจุบันยังไม่จำเป็นสำหรับ MVP

หมายเหตุสำคัญ Android release Gradle config ยังอ้าง debug signing; ก่อน public release ต้องใช้ production keystore/EAS signing credentials ขององค์กร

SECTION 08

ลำดับดำเนินงานแนะนำ

จัดลำดับจากสิ่งที่ลดความเสี่ยงข้อมูลไม่ครบและลดโหลดได้มากที่สุดก่อน

Priority	งาน	รายละเอียด	ผลลัพธ์
P0	Pagination safety	เพิ่ม keyset pagination ทุก clinical query และ user history query	กันข้อมูลถูกตัดโดยไม่รู้ตัว
P0	Catalog manifest	สร้าง global version/checksum และไม่ refresh ถ้า version เดิม	ลด request และ egress ทันที
P0	Local indexes	Medicine Map, pair Map และ SQLite tables	ลดเวลาค้นหา/ตรวจ interaction
P1	Incremental sync	Cursor, delta, tombstone และ transactional apply	รองรับ catalog โตและ update ถี
P1	User delta sync	push เฉพาะรายการเปลี่ยน, date range และ idempotency	ลด write amplification
P1	Server search	pg_trgm/GIN, debounce, minimum length และ result limit	รองรับ admin และ catalog ใหญ่
P1	Verification	unit/integration/E2E และ load tests	วัด correctness และ capacity
P1	Observability	structured logs, crash reporting, metrics และ alerts	ตรวจเหตุผิดปกติก่อนผู้ใช้แจ้ง
P2	CDN snapshot	signed/versioned compressed snapshot บน Storage/CDN	รองรับ concurrent startup จำนวนมาก
P2	Advanced search	ประเมิน dedicated search engine เมื่อ Postgres ไม่พอ	หลีกเลี่ยง complexity ก่อนจำเป็น

ข้อสรุป

Stack ปัจจุบันเหมาะกับ MVP และ pilot แบบควบคุมได้ จุดที่ควรลงทุนก่อนเพิ่ม infrastructure ใหม่คือความถูกต้องของ pagination, catalog versioning, incremental sync, local indexing และ observability

ไฟล์อ้างอิงหลักใน repository

- mobile/package.json, mobile/app.json, mobile/eas.json
- mobile/src/services/clinical-catalog.ts, sync.ts, supabase.ts และ secure-storage.ts
- mobile/src/data/medicine-db.ts และ mobile/src/utils/safety.ts
- admin/package.json, admin/src/lib/api.ts และ admin/vercel.json
- platform/supabase/config.toml, platform/README.md และ platform/supabase/migrations
- mobile/PRODUCTION_CHECKLIST.md

เอกสารนี้สะท้อน source code ณ วันที่จัดทำ ไม่ใช่การรับรองความพร้อมใช้งานเป็นผลิตภัณฑ์ทางการแพทย์