PROJECT TRANSFORMATION SUMMARY

YaCheck × MaCheck

จากต้นแบบ สู่ระบบที่ใช้งานจริง

สรุปสิ่งที่ปรับปรุง ระบบกลาง ความปลอดภัย ผลทดสอบ
และงานที่ต้องปิดก่อนเปิดให้ผู้ใช้ทั่วไป

18 กรกฎาคม 2026

MINIMAL / CLEAR / PRODUCTION-MINDED

สถานะโดยสรุป

วันนี้ระบบหลักทำงานจริงแล้ว

2

แอป native
YaCheck และ MaCheck

1

ฐานข้อมูลกลาง
Supabase โครงการเดียว

4

บริการบัญชี
สมัคร / เข้า / กู้คืน / ลบ

ข้อเท็จจริงสำคัญ

โครงสร้างเทคนิคพร้อมสำหรับการทดสอบแบบควบคุม
แต่ยังไม่ควรประกาศว่าเป็นบริการทางการแพทย์พร้อมใช้สาธารณะ จนกว่าจะผ่านการรับรองข้อมูล
กฎหมาย และการปฏิบัติการ

สิ่งที่เปลี่ยน ไม่ใช่แค่หน้าตาแอป

ก่อนปรับ

ต้นแบบ / เดโม

- YaCheck ยังเป็นเว็บต้นแบบ
- MaCheck มีโหมดจำลองและ LAN backend
- ข้อมูลแยกกันและเชื่อถือข้ามเครื่องไม่ได้

หลังปรับ

ระบบจริงระดับ technical foundation

- Expo / React Native สำหรับ iOS และ Android
- ฐานข้อมูลกลางพร้อม auth, sync และ RLS
- ทำงานแบบ local-first พร้อม fallback เมื่อออฟไลน์

สองแอปใช้แหล่งข้อมูลกลางเดียวกัน

YaCheck

กล้อง / แจ้งเตือน
ตู้ยา / บันทึกกินยา

SUPABASE CENTRAL

**บัญชี + ข้อมูลผู้ใช้
ฐานยา +
กฎความปลอดภัย**

RLS คุมสิทธิ์ทุกแถว

MaCheck

กล้อง / ตรวจสอบยา
ข้อมูลผู้ดูแล /
การติดตาม

แต่ละเครื่องเก็บ session และ cache ไว้ใน SecureStore | เมื่อออนไลน์จึงซิงก์กับฐานกลาง

YaCheck ถูกยกระดับจากเว็บเป็น native app

ไม่ใช่การห่อเว็บไว้ในกรอบ แต่เปลี่ยนเป็นโครงสร้าง Expo / React Native
ที่เข้าถึงความสามารถของโทรศัพท์จริง

01 กล้องจริง

สแกน barcode / QR
และค้นหายาด้วยชื่อ

02 แจ้งเตือนในเครื่อง

ตั้งเวลาตามตารางการใช้ยาโดยไ
ม่พึ่งหน้าเว็บ

03 ข้อมูลคงอยู่

ตู้ยา ตาราง
และประวัติไม่หายเมื่อปิดแอป

รหัสแอป **com.yacheck.app**

deep link **yacheck://**

ฟังก์ชันของ YaCheck เชื่อมเป็นวงจรการใช้ยา

01
เพิ่มยา
สแกนหรือค้นชื่อยา

02
ตั้งตาราง
เลือกเวลาและแจ้งเตือน

03
บันทึกการกิน
กินแล้ว / ข้าม / เลื่อน

04
ติดตามผล
ดู adherence และประวัติ

ฟังก์ชันเสริม

ตรวจยาตีกัน • อาหาร/โรคที่ควรระวัง • ประวัติแพ้ยา • ติดต่อฉุกเฉิน • การตั้งค่าการเข้าถึง

MaCheck เลิกพฤติกรรมแบบเดโม

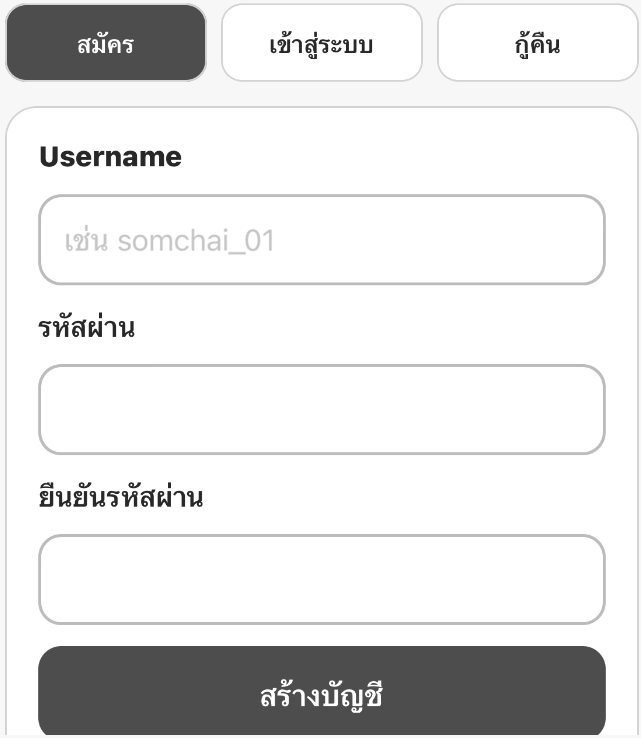
ถอดออก

- OCR / AI ปลอมที่ให้ผลเหมือนรู้จักยา
- alert simulator, ตัวเร่งเวลา และเคสตัวอย่าง
- คำแนะนำเว้นยาแบบตายตัวและ LAN backend

แทนที่ด้วย

- **กล้องจริง แล้วให้ผู้ใช้ยืนยันชื่อยาจาก catalogue**
- **ยาที่เพิ่มเองถูกระบุว่า “ยังไม่ผ่านการตรวจทาน”**

และแพ้ยายังซิงก์ข้าม YaCheck/MaCheck ได้



ภาพด้านขวา: build ปัจจุบันที่เปิดบน iOS Simulator

SUPABASE SCHEMA

ฐานข้อมูลกลางมีโครงสร้างรองรับงานจริง

บัญชี

account handles
app profiles
auth rate limits

ข้อมูลสุขภาพ

ตู้ยา
เหตุการณ์กินยา
activity logs

ข้อมูลทางคลินิก

รายการยา
ยาตีกัน
อาหาร/โรค

ผู้ดูแล

caregiver links
caregiver nudges
สถานะการเชื่อมโยง

ข้อมูลตั้งต้นบนระบบจริง

33 รายการยา **17 คู่ยาตีกัน** **6 ความสัมพันธ์อาหาร/โรค**

สมัครด้วย username เพื่อลดข้อมูลระบุตัวตนโดยตรง

ไม่อัปโหลด

ชื่อจริง
อีเมลจริง
เบอร์โทร
วันเกิด

ยังต้องเก็บ

username / UUID
ข้อมูลยา โรค แพ้ยา
ประวัติการใช้ยา

ข้อมูลสุขภาพยังเป็นข้อมูลอ่อนไหว แม้ไม่มีชื่อจริง

ข้อมูลที่จำเป็นเฉพาะเครื่อง เช่น เบอร์ติดต่อฉุกเฉิน เก็บใน SecureStore และไม่ส่งขึ้น cloud

AUTHENTICATION

วงจรบัญชีครบตั้งแต่สมัครจนถึงลบ

1

สมัคร

username + รหัสผ่าน
ระบบสร้าง mapping
ภายใน

2

เข้าสู่ระบบ

ตรวจรหัสผ่านผ่าน
Supabase Auth

3

กู้คืน

ใช้ recovery code
เพื่อกำหนดรหัสใหม่

4

ลบบัญชี

ยืนยันตัวตนแล้วลบ
ข้อมูลลูกแบบ cascade

บริการ Edge Functions 4 ตัว | มี rate limiting | รหัสผ่านขั้นต่ำ 10 ตัวอักษร | service-role key ไม่อยู่ในแอป

สิทธิ์ข้อมูลถูกบังคับที่ฐานข้อมูล

01 RLS	ทุกตารางข้อมูลผู้ใช้เปิด Row Level Security
02 auth.uid()	ผู้ใช้เห็นและแก้ได้เฉพาะข้อมูลที่เป็นเจ้าของ หรือได้รับสิทธิ์ผู้ดูแล
03 explicit grants	ปิดการเปิดตารางอัตโนมัติ แล้วอนุญาตเฉพาะคำสั่งที่แอปต้องใช้
04 private schema	mapping ระหว่าง username และบัญชี auth ไม่เปิดให้อุปกรณ์อ่าน

ทดสอบจริงด้วยบัญชีสองราย: อ่านข้ามบัญชีไม่ได้ และเขียน UUID ของผู้อื่นถูกปฏิเสธ

แอปยังทำงานต่อได้เมื่อเน็ตไม่เสถียร

ในเครื่อง

session token
health cache
ตารางและประวัติ

เมื่อออนไลน์

ซิงก์ profile
ตู้ยา
dose events

เมื่อออฟไลน์

ใช้ข้อมูลยา
ที่ bundle มากับแอป
เป็น fallback

ข้อจำกัดปัจจุบัน **ยังไม่มี offline queue และกติกาแก้ข้อมูลชนกันหลายเครื่องที่สมบูรณ์**

ฐานข้อมูลทางคลินิกมีจริง แต่ยังต้องรับรอง

33
รายการยา

17
คู่ยาตีกัน

6
อาหาร / โรค

หลักการที่ใช้

ไม่สรุปว่า “ปลอดภัย” ถ้ายังไม่มีข้อมูลรองรับ

ยาที่ผู้ใช้เพิ่มเอง =
ยังไม่ผ่านการตรวจทาน

ก่อน pilot ต้องเพิ่ม แหล่งอ้างอิง · ผู้ตรวจ · วันที่ตรวจ · เวอร์ชันข้อมูล · การลงนามจากเภสัชกร

Supabase Free Plan เหมาะกับการพัฒนาและ pilot

ใช้โครงการกลางเดียวในภูมิภาค Singapore โดยไม่ต้องจ่ายค่าบริการในช่วงเริ่มต้น

01 ได้ของจำเป็นครบ

Database, Auth, Edge Functions และ security rules อยู่ในระบบเดียว

02 ควบคุมต้นทุน

เหมาะกับการทดสอบภายในและ pilot ขนาดเล็กก่อนมีรายได้

03 มีภาระดูแล

โครงการอาจ pause เมื่อไม่มีการใช้งาน ต้องเปิด dashboard และสำรองข้อมูล

คำแนะนำ: ใช้ Free ต่อได้ แต่กำหนดเจ้าของระบบ ตรวจ quota และทำ manual backup เป็นรอบ

VERIFICATION EVIDENCE

ผ่านการทดสอบกับระบบจริง ไม่ใช่แค่ผ่าน build

BACKEND LIVE TESTS

- สมัคร / login / recovery / เปลี่ยนรหัส / ลบบัญชี
- เขียน profile, ตู้ยา และ dose event สำเร็จ
- RLS แยกข้อมูลสองบัญชีได้จริง
- Database lint ผ่าน และบัญชีทดสอบถูกล้างแล้ว

APP BUILDS

- YaCheck: typecheck, lint, Expo Doctor 21/21
- YaCheck: export iOS / Android / Web สำเร็จ
- MaCheck: typecheck, lint และ export สำเร็จ
- MaCheck: native iOS Simulator build เปิดใช้งานจริง

4 migrations ตรงกับ remote · **4 Edge Functions ACTIVE** · **ไม่มี schema error**

READINESS SNAPSHOT

พร้อมด้านเทคนิค แต่ยังมี release gates

พร้อมแล้ว

- Backend กลางและบัญชีทำงานจริง
- สองแอปเชื่อมข้อมูลชุดเดียวกัน
- RLS, SecureStore และ account deletion
- build/export หลายแพลตฟอร์มผ่าน

ยังต้องปิด

- ทดสอบ iOS / Android บนอุปกรณ์จริง
- clinical review และข้อความเตือนฉุกเฉิน
- PDPA, consent, retention และ data export
- automated tests, monitoring และ store signing

RECOMMENDED ROADMAP

เดินหน้าเป็น 3 ระยะ โดยเริ่มจาก pilot

A	**Pilot readiness**	ทดสอบข้ามอุปกรณ์ • caregiver invite • privacy/consent • clinical sign-off • monitoring
B	**Public release**	unit/integration/E2E • CI • accessibility • store assets/signing • rollback plan
C	**Scale and reliability**	incremental sync • conflict resolution • versioned clinical dataset • performance metrics

CONCLUSION

โปรเจกต์หลุดจากสถานะ mockup แล้ว

ตอนนี้ YaCheck และ MaCheck มีฐานเทคนิคที่ใช้งานจริงร่วมกัน
แต่เส้นชัยถัดไปคือ “pilot ที่ปลอดภัย” ไม่ใช่รีบขึ้นสโตร์

ข้อเสนอให้ตัดสินใจ

1 ปิดงาน P0 ให้ครบ

2 ทดลองกับกลุ่มเล็กแบบควบคุม

3 เก็บหลักฐานและแก้จุดเสี่ยง

4 จึงเตรียม public release