zenity x

AI AGENT CO-WORKING SYSTEM · DAY 1

Basic CRM System with Vibe Coding

ออกแบบระบบ CRM ให้ตอบโจทย์การใช้งานของคุณ ด้วย AI Agent

เข้าใจการพัฒนาระบบ ออกแบบเป็น CLAUDE.md แล้วให้ AI Agent สร้างทีละเฟส — สมัคร ล็อกอิน แบ่งสิทธิ์ และจัดการลีดได้จริง

1 วัน · 10:00–17:00 น.

Claude Code / Codex · Supabase · Cloudflare



RUN OF SHOW

ตารางสอน 1 วัน

เวลาเรียน 10:00–17:00 | พักกลางวัน 12:30–13:30 | เบรก 15:00–15:15

ช่วงเช้า · เข้าใจ แล้วออกแบบ

10:00–10:45	**Session 1 · การพัฒนาระบบทำงานอย่างไร** SDLC · ระบบ 3 ชั้น · ตามรอย 1 คำขอ · Authentication / Authorization
10:45–11:30	**Session 2 · AI Agent ทำงานจริงอย่างไร** Main Agent / Subagent · ทำไมใช้ Main Agent ตัวเดียว · CLAUDE.md คืออะไร
11:30–12:30	**Workshop 1 · เตรียม Env + ออกแบบระบบ** **Cloudflare + Supabase** พร้อมใช้ · Use Case · ตารางสิทธิ์ · ได้ CLAUDE.md แบ่ง 3 เฟส

ช่วงบ่าย · สร้างทีละเฟสจาก CLAUDE.md

13:30–15:00	**Workshop 2 · Phase 1 สมัคร + ล็อกอิน + 3 บทบาท** **ด่าน 1** สมัครแล้วรอแอดมินอนุมัติ · สลับล็อกอิน 3 บทบาท เห็นเมนูต่างกัน
15:00–15:15	**เบรก**
15:15–16:30	**Workshop 3 · Phase 2 ลีด CRM** **ด่าน 2** ลีดเข้าระบบ · พนักงานขายเห็นเฉพาะลีดของตัวเอง
16:30–17:00	**สรุป + Phase 3 การบ้าน + ถาม-ตอบ** Dashboard ผู้บริหาร + ขึ้นเว็บ ทำต่อที่บ้านด้วย Prompt ที่ให้

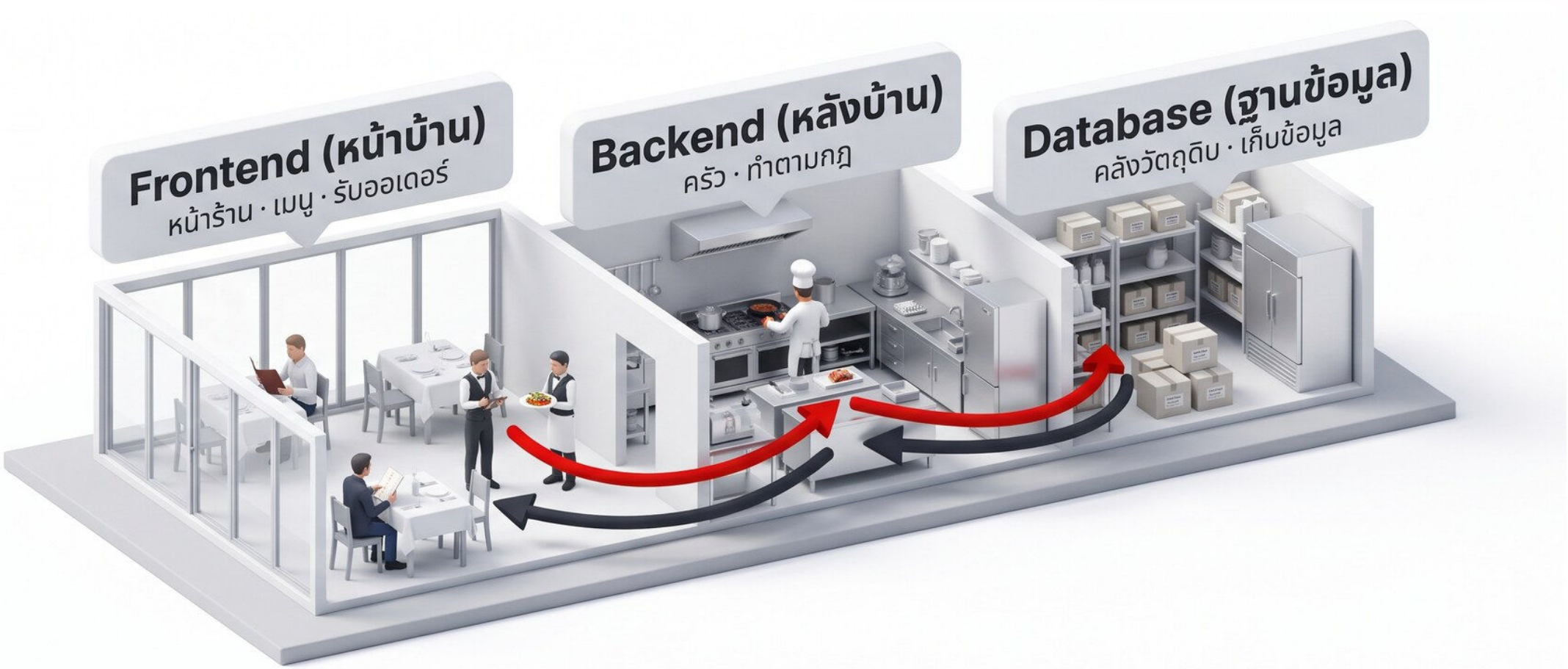
ตารางเวลาเป็นแนวทาง ผู้สอนอาจปรับเวลาแต่ละช่วงให้เหมาะกับความคืบหน้าของผู้เรียน โดยเนื้อหาและผลลัพธ์หลักยังครบถ้วน



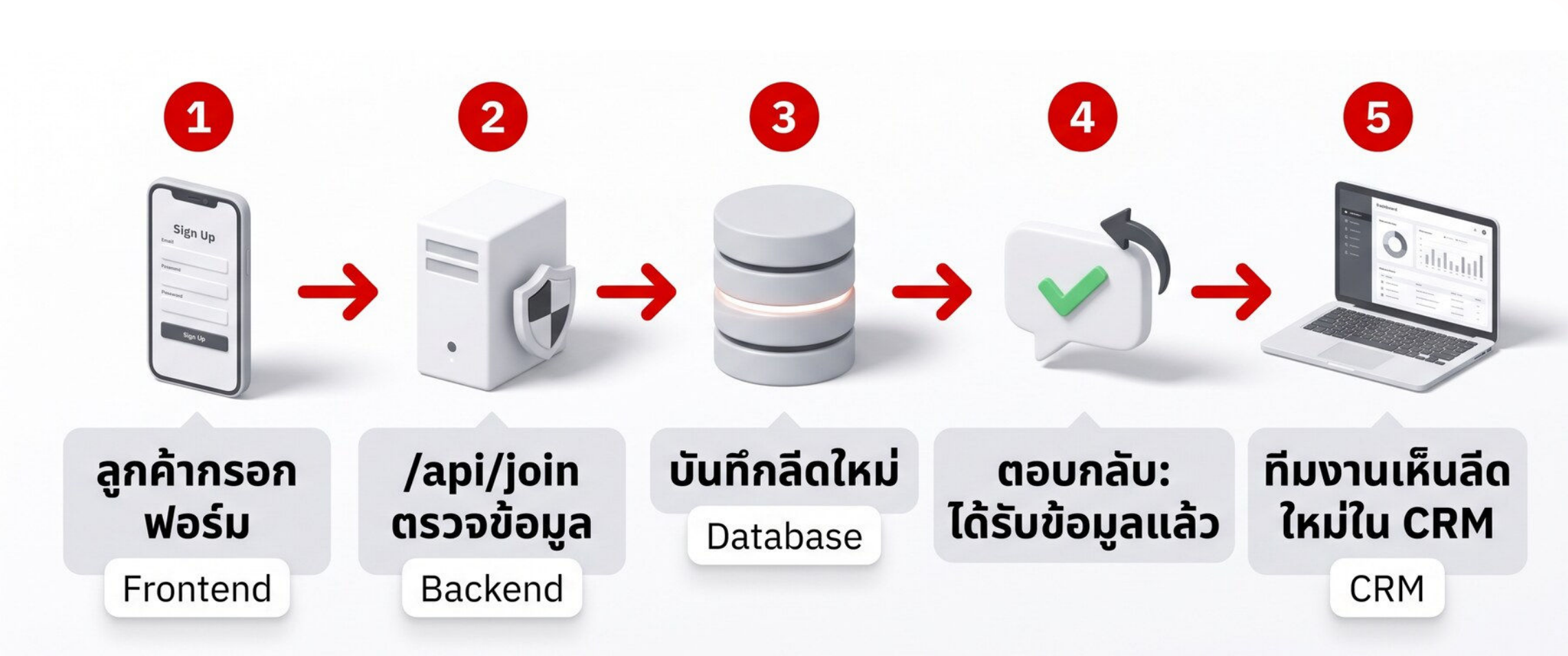
วันนี้: เช้า = **เก็บความต้องการ + วิเคราะห์ ออกแบบ** · บ่าย = **พัฒนา + ทดสอบ** · การบ้าน = ขึ้นระบบ

SESSION 1 · 3 ชั้นของระบบ

การทำงานของระบบ



ตัวอย่างการส่งข้อมูล



SESSION 1 · ล็อกอินทำงานอย่างไร

ล็อกอินมี 2 ด่าน: คุณคือใคร → คุณทำอะไรได้



Authentication · คุณคือใคร

ตรวจอีเมล + รหัสผ่าน (Supabase Auth ทำให้) ผ่านแล้วได้ **"บัตรผ่าน"** ติดตัวไปทุกคำขอ — เหมือนแตะบัตรเข้าตึก

Authorization · คุณทำอะไรได้

อ่านบทบาทจากตาราง `profiles` แล้วเปิดเมนูตามบทบาท + ล็อกที่ฐานข้อมูลด้วย **Row Level Security** — บัตรแต่ละใบเปิดได้แค่บางห้อง

ในระบบของเรา

สมัครแล้วแต่ยัง **รออนุมัติ** = ผ่านด่าน 1 แต่ไม่ผ่านด่าน 2 · พนักงานขายล็อกอินได้ แต่ **ลบลึดไม่ได้**

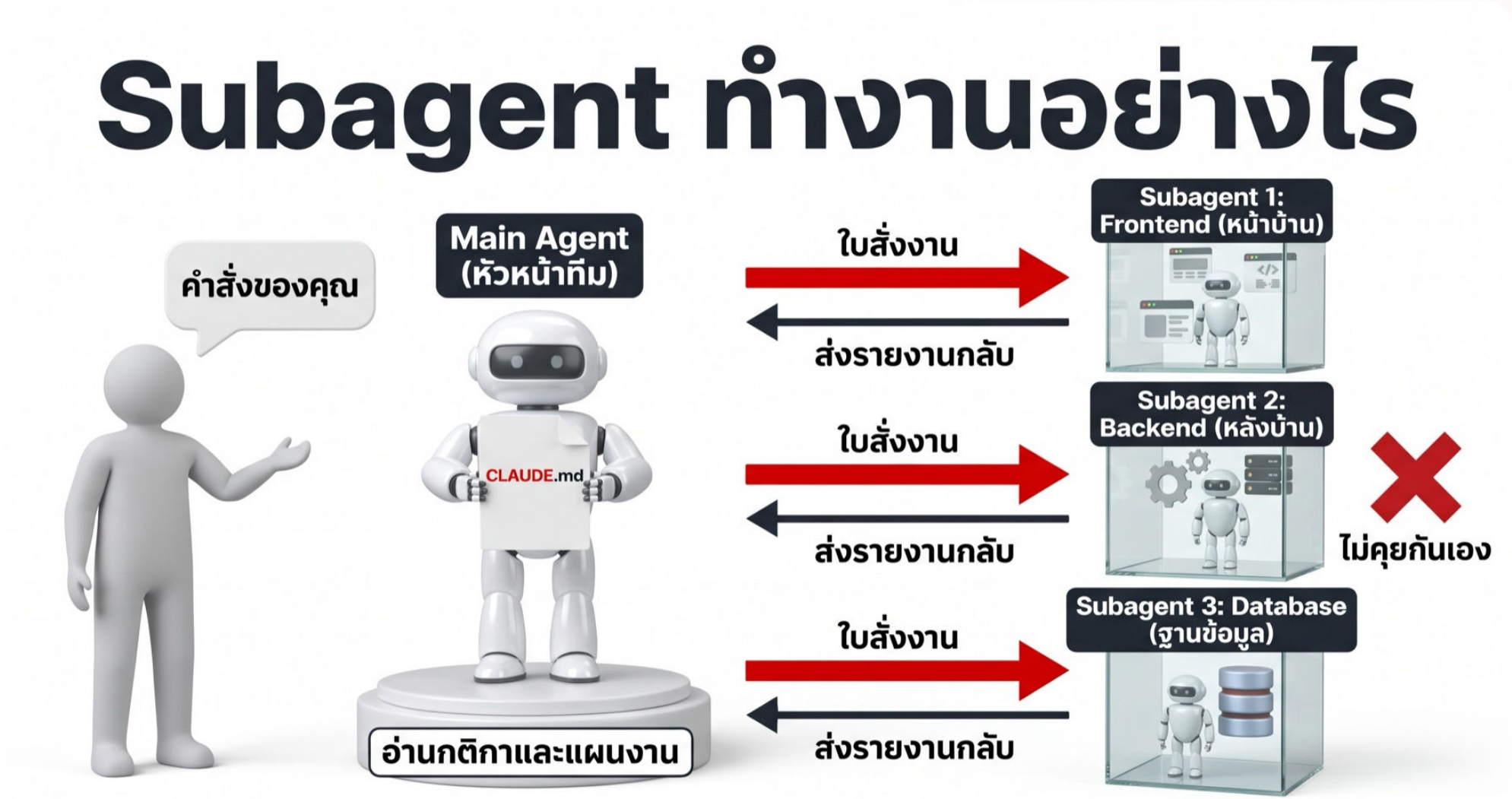
Agent คุยเองไหม?



Agent ไม่ได้คุยกันเอง — ทำงานผ่าน **ใบสั่งงาน · CLAUDE.md · ไฟล์ในโปรเจกต์**

Main Agent = หัวหน้าทีม เขียนใบสั่งงานให้แต่ละคน





Subagent เริ่มจากความจำว่าง รู้แค่ใบสั่งงาน + ไฟล์กติกา + ไฟล์ในโปรเจกต์



หลาย Agent ใช้โทเค็นราว **15 เท่า**ของแชตปกติ (ตัวเดียวราว 4 เท่า) · ที่แพงจริงคือ **จำนวนรอบที่ต้องสั่งแก้** · Subagent เหมาะกับงานอ่าน ค้น ตรวจ

อ้างอิง

Anthropic · Multi-agent research system
Anthropic · Building effective agents
Cognition · Don't Build Multi-Agents
Claude Code Docs · Subagents

SESSION 2 · AGENT 2 แบบ

Agent 2 แบบ



	ช่วยสร้างระบบ	ทำงานในระบบ
ตัวอย่าง	Claude Code / Codex	แชตบอตตอบลูกค้า · ผู้ช่วยสรุปลีด
ทำงานเมื่อไหร่	เมื่อเราสั่งในเครื่อง	ตลอด 24 ชม. บนเว็บที่ขึ้นแล้ว
อะไรสั่งให้เริ่ม	คนพิมพ์คำสั่ง	เหตุการณ์ เช่น ลีดใหม่เข้า ลูกค้าทักแชต
คนอยู่ตรงไหน	ตรวจและทดสอบงาน	กดอนุมัติก่อนส่งถึงลูกค้า

วันนี้ใช้แบบซ้ายสร้างระบบ · แบบขวาเป็นงานต่อยอดหลังระบบขึ้นเว็บแล้ว

SESSION 2 · CLAUDE.MD คืออะไร

CLAUDE.md = คู่มือโปรเจกต์ ที่ Agent อ่านเองทุกครั้ง

club-crm / CLAUDE.md

```
# CLAUDE.md — Harmony Club CRM
## 1. เป้าหมายระบบ
## 2. เครื่องมือ          Supabase · Cloudflare · Git
## 3. บทบาท + ตารางสิทธิ์
## 4. ตารางข้อมูล         profiles · leads · contact_logs
## 5. กติกาทีม            คีย์ลับห้ามอยู่หน้าเว็บ
## 6. แผนงาน
- [ ] Phase 1 สมัคร + ล็อกอิน + 3 บทบาท
- [ ] Phase 2 ลีด CRM
- [ ] Phase 3 Dashboard + ขึ้นเว็บ
```

Agent อ่านเองทุกครั้งที่เปิดงาน
ไม่ต้องสั่ง "อ่านไฟล์นี้" ซ้ำทุกรอบ

ไฟล์เดียวรวมทุกอย่าง
บทบาท สิทธิ์ ข้อมูล กติกา และแผนงาน อยู่ที่เดียว

สั่งงานสั้นลง
"อ่าน CLAUDE.md แล้วทำ Phase 1"

เขียนให้กระชับ (ไม่เกิน ~150 บรรทัด) · โปรเจกต์ใหญ่ขึ้นค่อยแยกรายละเอียดไปไว้ใน docs/ · ใช้ Codex = ไฟล์ AGENTS.md

อ้างอิง Claude Code Docs · Memory (CLAUDE.md)

SESSION 2 · CLAUDE.MD มีกี่ระดับ

CLAUDE.md มี 3 ระดับ — วันนี้เขียนระดับ Project

Global · ทุกโปรเจกต์

`~/.claude/CLAUDE.md`

นิสัยการทำงานของเราเอง เช่น ตอบเป็นภาษาไทย · ถามก่อนลบไฟล์ · สไตล์โค้ดที่ชอบ

Codex: `~/.codex/AGENTS.md`

Project · โปรเจกต์นี้

`club-crm/CLAUDE.md`

เป้าหมาย บทบาท ตารางสิทธิ์ ข้อมูล กติกา และแผน 3 เฟส — แชร์ให้ทีมผ่าน Git

Codex: `club-crm/AGENTS.md`

Subproject · โฟลเดอร์ย่อย

`club-crm/app/api/CLAUDE.md`

กติกาเฉพาะส่วน อ่านเมื่อ Agent ทำงานในโฟลเดอร์นั้น เช่น กติกาของ API

Codex: `AGENTS.md` ในโฟลเดอร์ย่อย

Global ✓ **Project** ✓ **Subproject** Agent อ่านซ้อนกันทุกระดับ · ระดับที่ใกล้งานกว่าละเอียดกว่า

อ้างอิง Claude Code Docs · Memory (CLAUDE.md)

W1

WORKSHOP 1 · 11:30-12:30

เตรียม Env + ออกแบบระบบ

เครื่องมือพร้อม → Use Case → หน้าจอตามบทบาท → ตาราง สิทธิ์ → ได้ CLAUDE.md แบ่ง 3 เฟส



WORKSHOP 1 · เครื่องมือที่ใช้วันนี้

เครื่องมือที่ใช้วันนี้

Claude Code
Main Agent ช่วยเขียนโค้ด

สมัคร Claude แบบรายเดือน ขั้นต่ำ **Pro $20/เดือน** · หรือใช้ Codex กับ ChatGPT Plus

Git
Version Control

บันทึก **จุดย้อนกลับ** ทุกครั้งที่ผ่านด่าน — พังเมื่อไหร่ ย้อนกลับได้

Node.js
ตัวรันโปรเจกต์

ใช้สร้างและรันเว็บบนเครื่องด้วย `npm run dev` · Agent ติดตั้งให้ได้

Next.js
Framework ของเว็บ

หน้าเว็บ + Backend ในโปรเจกต์เดียว · ต่อยอดเป็นงานจริงได้ · สร้างด้วย `create-next-app`

SETUP · ตรวจเครื่อง + สร้างโปรเจกต์ Next.js

```
ช่วยเตรียมเครื่องของฉันสำหรับทำระบบ CRM วันนี้:
1. ตรวจว่ามี Git และ Node.js (เวอร์ชัน 20 ขึ้นไป) หรือยัง ตัวไหนยังไม่มีให้ช่วยติดตั้ง (ถามก่อนทุกครั้งที่ต้องใช้รหัสผ่านเครื่อง)
2. สร้างโปรเจกต์ Next.js ชื่อ club-crm ด้วยคำสั่ง
   npx create-next-app@latest club-crm --ts --app --tailwind --eslint --no-src-dir --import-alias "@/*" --use-npm --yes
3. ติดตั้ง Supabase สำหรับ Next.js: npm install @supabase/supabase-js @supabase/ssr
4. ตรวจว่า .gitignore กันไฟล์ลับ .env* แล้ว และ Git เริ่มแล้ว
5. รัน npm run dev แล้วบอกให้ฉันเปิด http://localhost:3000 ดูว่าขึ้นหน้าเว็บ
สรุปผลเป็นตาราง: รายการ · สถานะ · ต้องทำอะไรต่อ
```

WORKSHOP 1 · เตรียม ENV

Supabase + Cloudflare ใช้ทำอะไร

Supabase

ฐานข้อมูล + ระบบล็อกอิน

เก็บข้อมูลเป็นตาราง · ตรวจว่าใครเป็นใคร · ล็อกสิทธิ์ที่ฐานข้อมูล (Row Level Security)

1. **New project ชื่อ `club-crm`**
   ตั้งรหัสผ่านฐานข้อมูล · Region Singapore
2. **Authentication → Sign In / Providers → Email**
   ปิด **Confirm email** (สำหรับคลาส)
3. **Project Settings → API Keys**
   จด **Project URL** + **publishable key** (secret key ใช้ Phase 2)

Cloudflare

ที่ฝากเว็บจริง

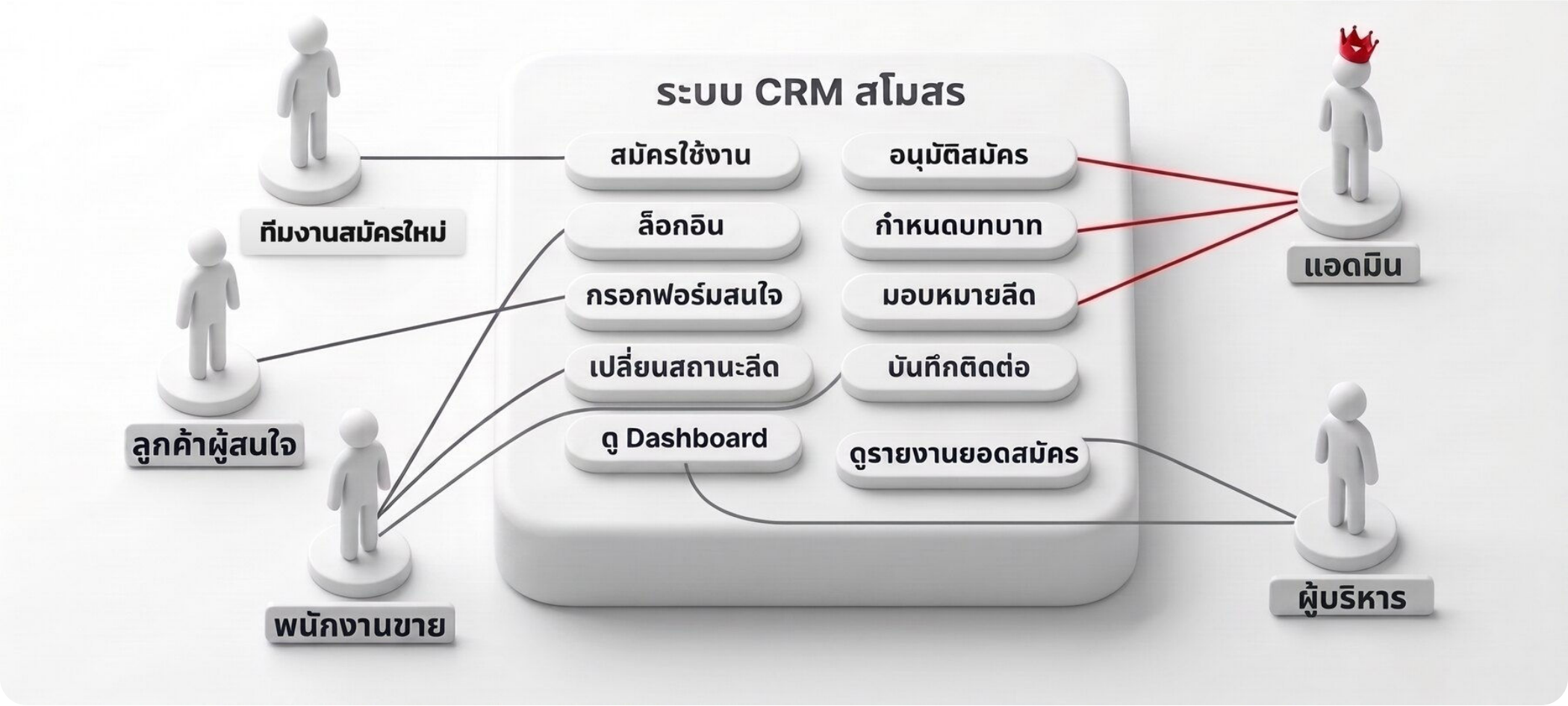
ระหว่างเรียนรันบนเครื่องด้วย `npm run dev` · Cloudflare ใช้ตอนเอาเว็บขึ้นจริงใน Phase 3

1. **สมัครฟรีที่ dash.cloudflare.com**
   ใช้อีเมลที่ใช้ประจำ · วันนี้สมัครไว้ก่อน
2. **Phase 3 ให้ Agent ตั้งค่าให้**
   Next.js + OpenNext adapter → `npm run deploy`
3. **ได้ URL เว็บจริง**
   เปิดบนมือถือแล้วล็อกอินได้

publishable key ใส่ `.env.local` ใช้ฝั่งหน้าเว็บได้

secret key = กุญแจห้องเก็บของ ใช้ฝั่ง server เท่านั้น ห้ามส่งในแชต

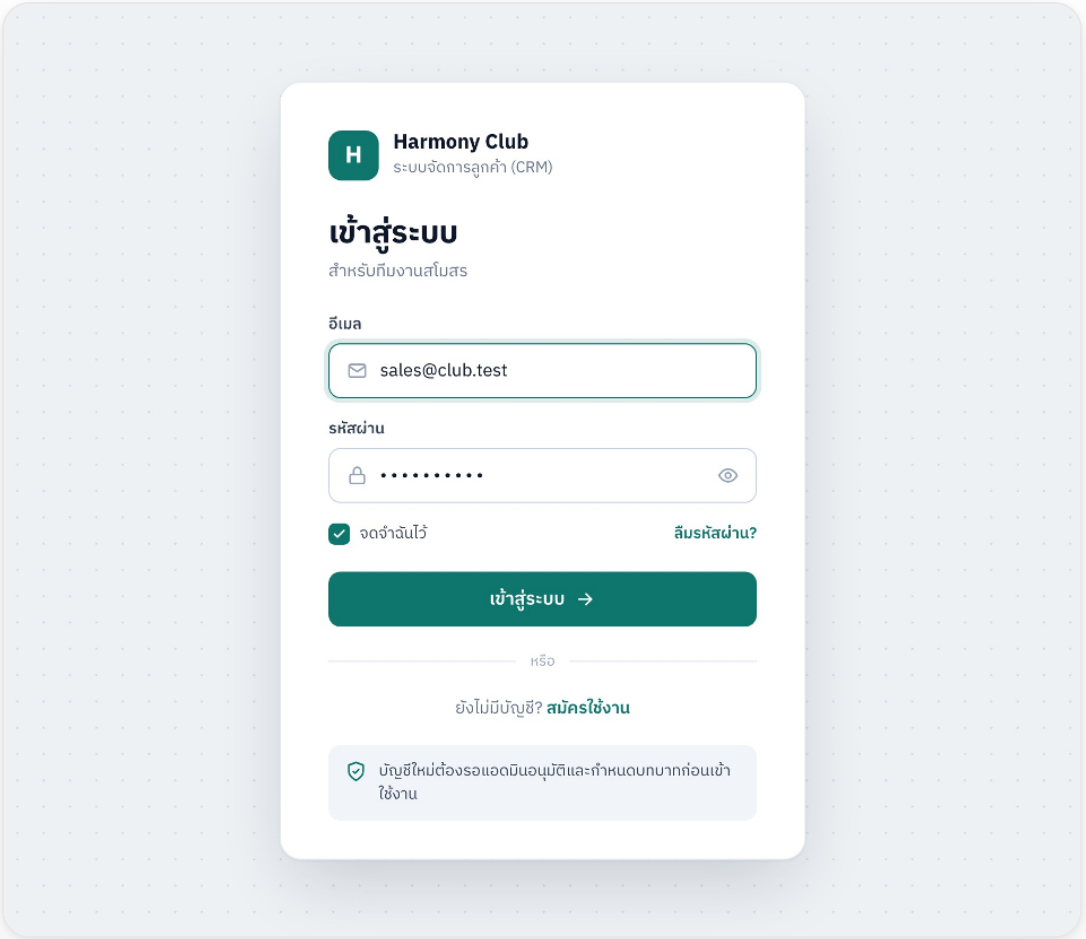
Use Case: ใครทำอะไรได้บ้าง



ทีมงานสมัครใหม่ = พนักงานที่ขอบัญชี รอแอดมินอนุมัติ · **ลูกค้าผู้สนใจ** = คนนอกที่กรอกฟอร์ม กลายเป็นลีด ไม่มีบัญชี · เส้นแดง = งานของแอดมิน

WORKSHOP 1 · หน้าจอตามบทบาท · ล็อกอิน

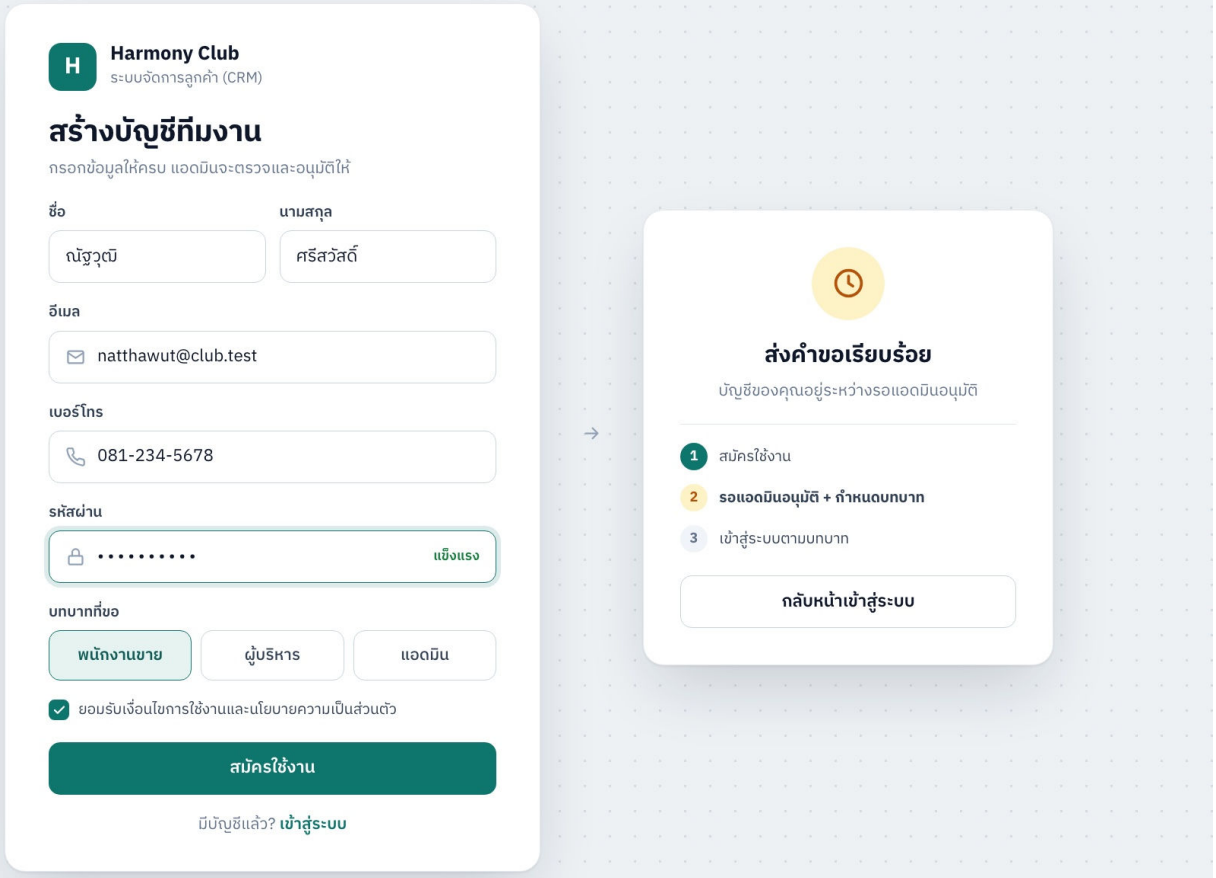
ล็อกอินหน้าเดียว → เปิดเมนูตามบทบาท



1. **กรอกอีเมล + รหัสผ่าน**
   หน้า /login
2. **ระบบตรวจว่าเป็นใคร**
   Authentication
3. **ยังรออนุมัติ → แจ้งเตือน ห้ามเข้า**
   สถานะ pending
4. **อ่านบทบาทจาก profiles → เปิดเมนู**
   Authorization · admin / sales / manager

WORKSHOP 1 · หน้าจอตามบทบาท · ทีมงานสมัครใหม่

สมัครใช้งาน → รอแอดมินอนุมัติ



ทำได้

- ✓ สมัครด้วยอีเมล + รหัสผ่าน
- ✓ เลือกบทบาทที่ขอ

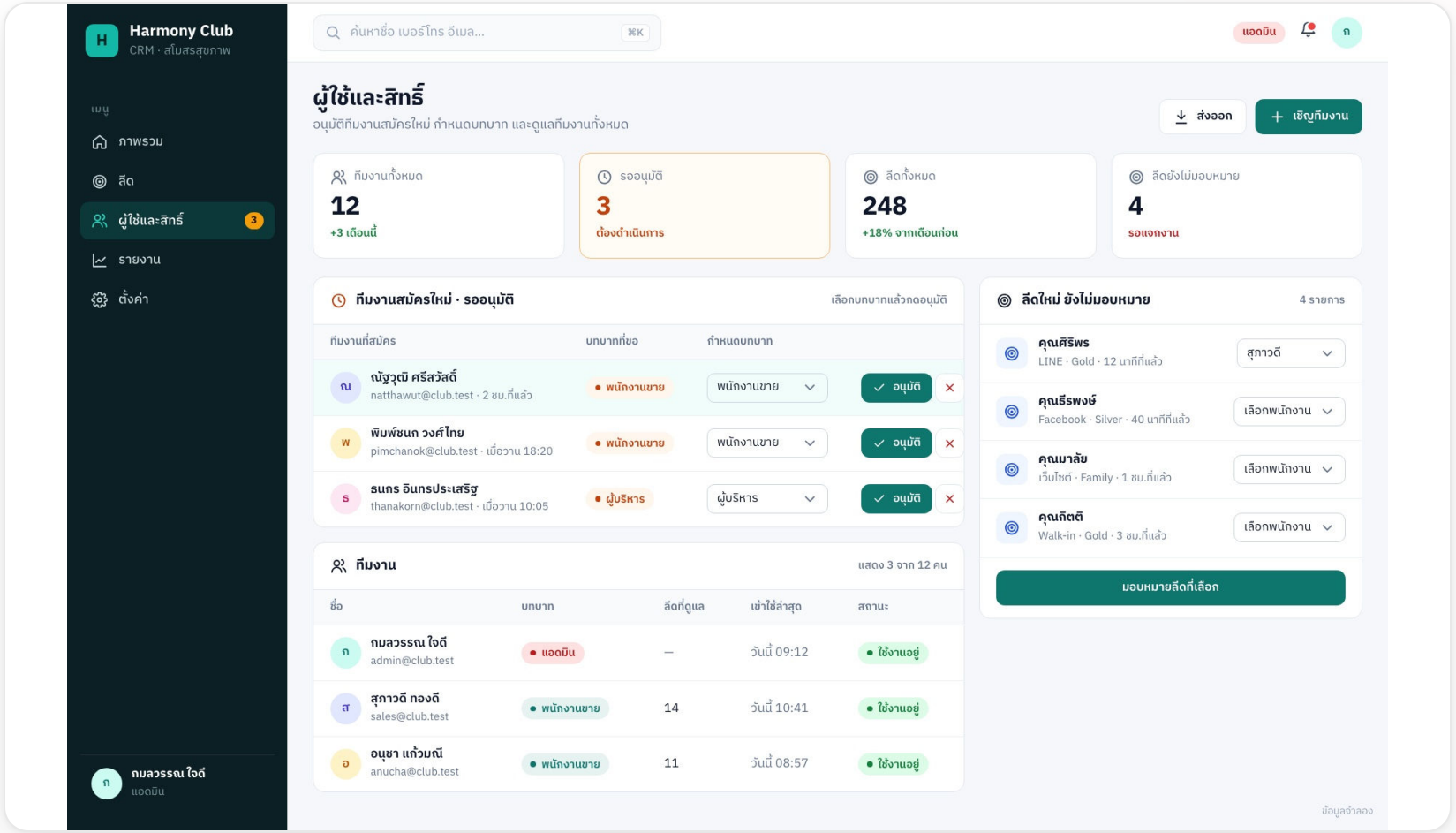
ทำไม่ได้

- ✗ เข้าระบบก่อนได้รับอนุมัติ
- ✗ ตั้งบทบาทให้ตัวเอง

หน้าจอตัวอย่างจาก mockups/roles.html · ข้อมูลจำลอง · กดที่ภาพเพื่อดูเต็มจอ

WORKSHOP 1 · หน้าจอตามบทบาท · แอดมิน

แอดมิน: อนุมัติทีมงานใหม่ และจัดการได้ครบ



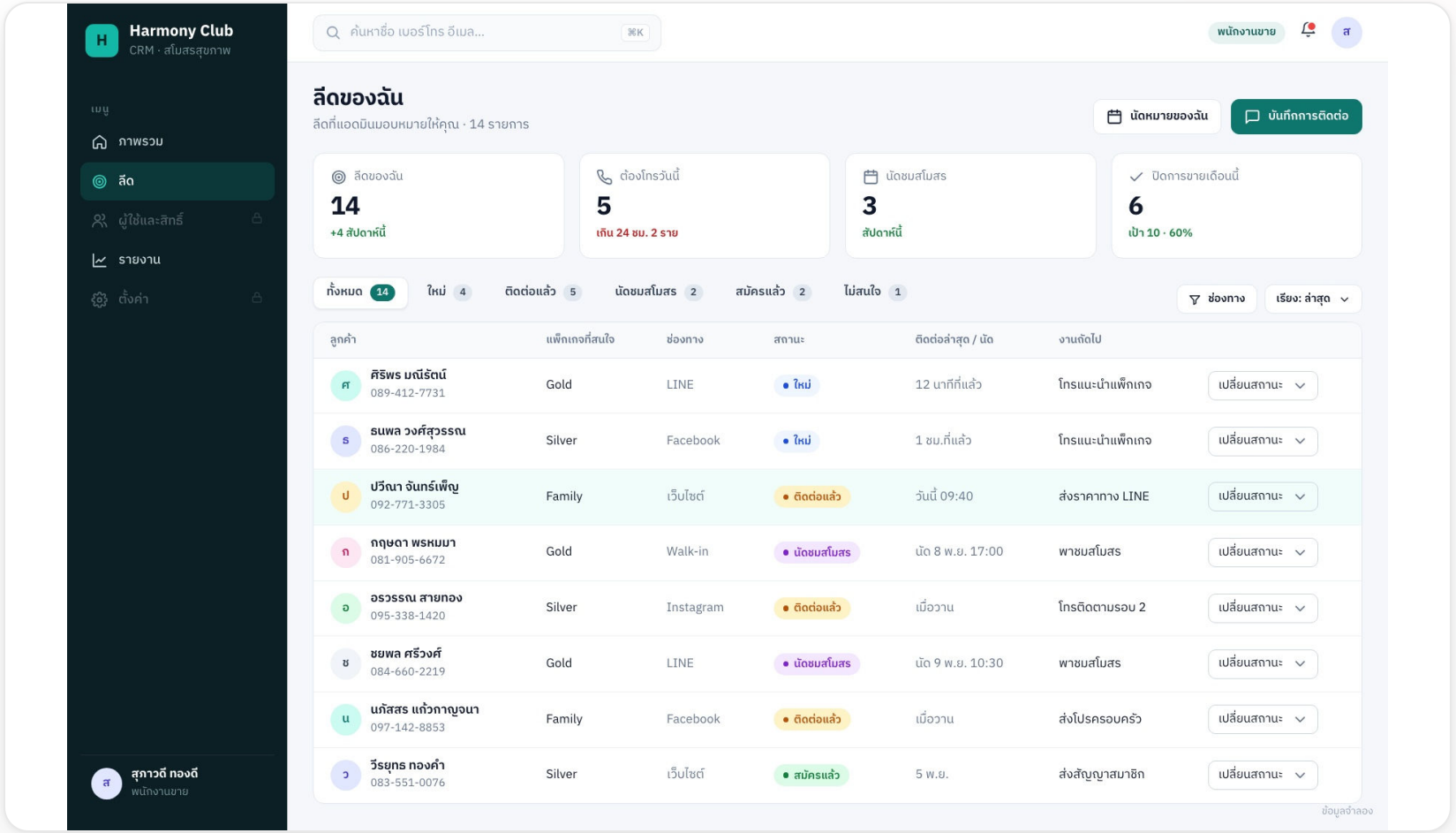
หน้าจอตัวอย่างจาก mockups/roles.html · ข้อมูลจำลอง · กดที่ภาพเพื่อดูเต็มจอ

ทำได้

- อนุมัติ / ปฏิเสธทีมงานสมัครใหม่
- กำหนดและเปลี่ยนบทบาท
- เห็นลีดทุกคน มอบหมาย แก้ ลบ

WORKSHOP 1 · หน้าจอตามบทบาท · พนักงานขาย

พนักงานขาย: ลีดที่ได้รับมอบหมาย



หน้าจอตัวอย่างจาก mockups/roles.html · ข้อมูลจำลอง · กดที่ภาพเพื่อดูเต็มจอ

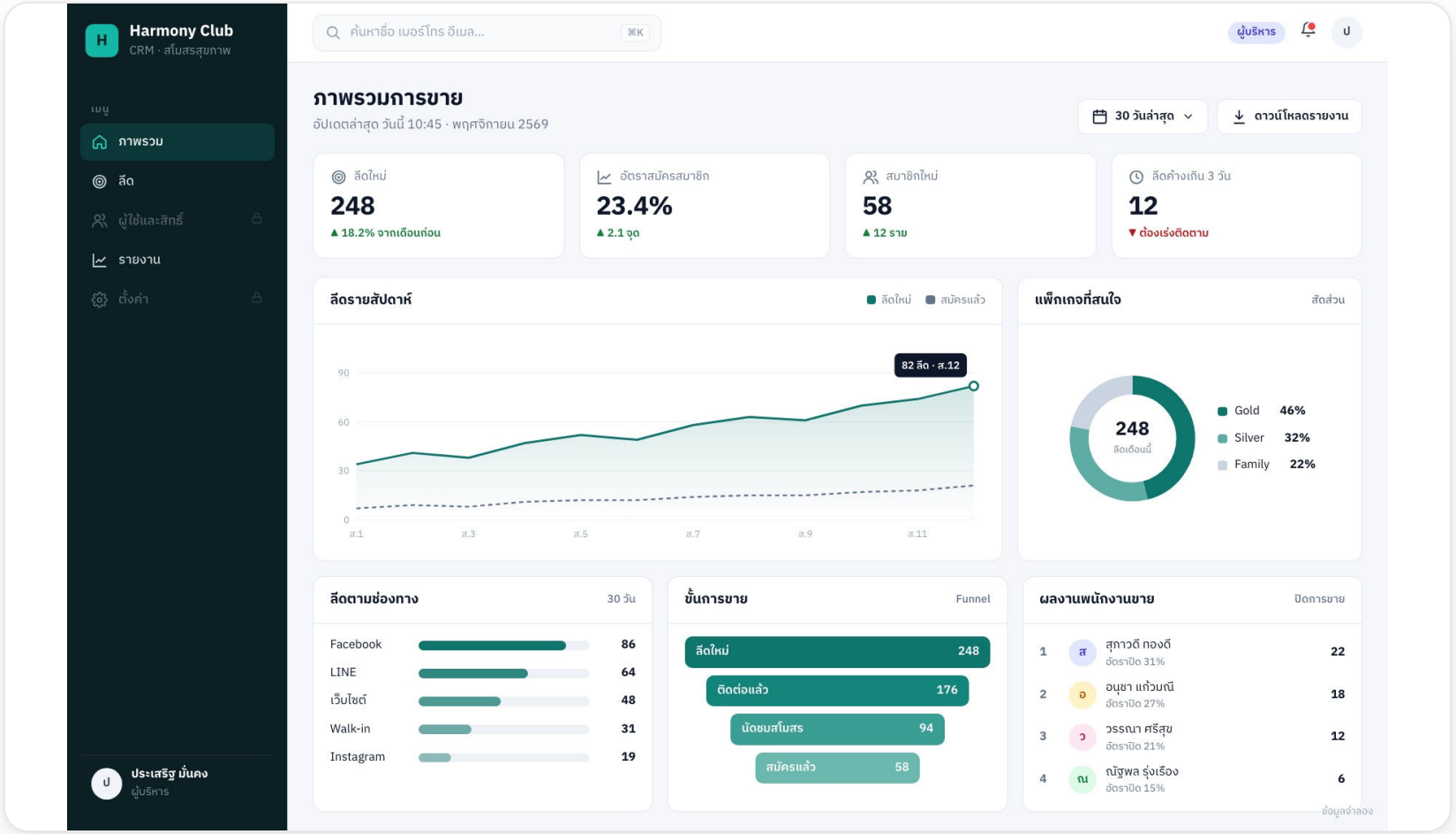
ทำได้

- ดูลีดของตัวเองทั้งหมด แยกตามสถานะ
- เปลี่ยนสถานะ · บันทึกการติดต่อ

ทำไม่ได้

- เห็นลีดของคนอื่น · ลบลีด
- อนุมัติผู้ใช้ · เปลี่ยนบทบาท

ผู้บริหาร: Dashboard ภาพรวม ดูอย่างเดียว



หน้าจอตัวอย่างจาก mockups/roles.html · ข้อมูลจำลอง · กดที่ภาพเพื่อดูเต็มจอ

ทำได้

- ดูตัวเลขสำคัญ กราฟแนวโน้ม
- ดูช่องทาง ขั้นการขาย ผลงานทีม

ทำไม่ได้

- แก้ไขหรือลบข้อมูล

WORKSHOP 1 · ตารางสิทธิ์ (PERMISSION MATRIX)

ใครทำอะไรได้ = ด่านทดสอบท้ายแต่ละเฟส

การกระทำ	ทีมงานสมัครใหม่ รออนุมัติ	แอดมิน admin	พนักงานขาย sales	ผู้บริหาร manager
สมัครใช้งาน / ล็อกอิน	✓	✓	✓	✓
อนุมัติทีมงานใหม่ · กำหนดบทบาท	×	✓	×	×
ดูรายชื่อลีด	×	✓	เฉพาะของตัวเอง	✓
มอบหมายลีด	×	✓	×	×
เปลี่ยนสถานะ · บันทึกการติดต่อ	×	✓	เฉพาะของตัวเอง	×
ลบลีด	×	✓	×	×
ดู Dashboard	×	✓	×	✓

นี่คือตัวอย่าง — ในสไลด์ถัดไปนักเรียนเติมบทบาทของระบบตัวเองลงใน Prompt · ลูกค้าผู้สนใจกรอกฟอร์มได้อย่างเดียว

WORKSHOP 1 · ออกแบบระบบ → CLAUDE.MD

เติมบทบาทของระบบเอง แล้วให้ Agent เขียน CLAUDE.md

ต้องเติมก่อนส่ง
บทบาทที่ 1–3
ชื่อ · ทำได้ · ทำไม่ได้ — ใช้ตารางสิทธิ์หน้าก่อนเป็นตัวอย่าง

ต้องเติมก่อนส่ง
ใครอนุมัติทีมงานใหม่
และข้อมูลลูกค้าที่จะเก็บจากฟอร์ม

Agent เขียนให้
CLAUDE.md 6 หัวข้อ + 3 เฟส
Phase 1 ล็อกอิน · Phase 2 ลีด · Phase 3 Dashboard + ขึ้นเว็บ

DESIGN · เขียน CLAUDE.md

เราจะสร้างระบบ CRM ของสโมสรสุขภาพ Harmony Club (ชื่อสมมติ · ใช้ข้อมูลจำลองเท่านั้น) ช่วยฉันออกแบบระบบ แล้วเขียนเป็นไฟล์ CLAUDE.md ที่ root ของโปรเจกต์ (ถ้าใช้ Codex ให้ตั้งชื่อ AGENTS.md)

ระบบของเรา (เติมในวงเล็บเหลี่ยมให้ครบก่อนส่ง):
- บทบาทที่ 1: [ชื่อบทบาท เช่น แอดมิน]
  ทำได้: [...]
  ทำไม่ได้: [...]
- บทบาทที่ 2: [ชื่อบทบาท เช่น พนักงานขาย]
  ทำได้: [...]
  ทำไม่ได้: [...]
- บทบาทที่ 3: [ชื่อบทบาท เช่น ผู้บริหาร]
  ทำได้: [...]
  ทำไม่ได้: [...]
- ทีมงานสมัครใหม่: สมัครเองได้ แต่สถานะ "รออนุมัติ" จนบทบาท [ใคร] อนุมัติและกำหนดบทบาทให้
- ลูกค้าผู้สนใจ (ไม่ต้องล็อกอิน): กรอกฟอร์มสนใจสมัคร เก็บข้อมูล [เช่น ชื่อ เบอร์โทร อีเมล แพ็กเกจที่สนใจ ช่องทางที่รู้จักเรา]

ถ้าข้อไหนฉันยังเว้นไว้หรือไม่ชัด ให้ถามฉันทีละข้อก่อนเขียน (ไม่เกิน 6 ข้อ)

CLAUDE.md มี 6 หัวข้อ เขียนกระชับ ไม่เกิน 150 บรรทัด:
1. เป้าหมายระบบ (2-3 บรรทัด)
2. เครื่องมือ: Next.js (App Router · TypeScript · Tailwind) + Supabase (Auth + Database + Row Level Security ผ่าน @supabase/ssr) + Git · ขึ้นเว็บบน Cloudflare ใน Phase 3
3. บทบาทและตารางสิทธิ์ (Permission Matrix เป็นตาราง) ตามที่ฉันเติมข้างบน
4. ตารางข้อมูล: profiles (ทีมงาน + บทบาท + สถานะอนุมัติ) · leads (ลูกค้าผู้สนใจ) · contact_logs (ประวัติการติดต่อ)
5. กติกาทีม: คีย์อยู่ใน .env.local เท่านั้น · secret key ใช้ฝั่ง server เท่านั้น · ใช้ข้อมูลจำลอง · ทุกตารางเปิด Row Level Security · ทำทีละเฟส เสร็จแล้วหยุดให้ฉันทดสอบ
6. แผนงาน 3 เฟส แต่ละเฟสมีเช็คลิสต์ "เสร็จเมื่อ…":
   - Phase 1 · สมัคร + ล็อกอิน + แบ่งสิทธิ์ตามบทบาท + อนุมัติทีมงานใหม่
   - Phase 2 · ลีด CRM: ฟอร์มสนใจ · มอบหมายลีด · แต่ละบทบาทเห็นลีดตามสิทธิ์
   - Phase 3 · Dashboard + ขึ้นเว็บ

ยังไม่ต้องเขียนโค้ด เขียนเสร็จแล้วสรุปให้ฉันตรวจ

12:30–13:30 · พักรับประทานอาหารกลางวัน

พักเที่ยง

ช่วงบ่ายเปิด CLAUDE.md แล้วสร้าง Phase 1 ทันที



W2

WORKSHOP 2-3 · 13:30-16:30

สร้างทีละเฟส จาก CLAUDE.md

Phase 1 สมัคร + ล็อกอิน + 3 บทบาท (13:30–15:00) · Phase 2 ลีด CRM (15:15–16:30) · ผ่านด่านแล้วบันทึกจุดย้อนกลับทุกครั้ง



PHASE 1 · ก่อนรัน PROMPT

คัดลอก Prompt เดียว ได้ระบบ สมัคร + ล็อกอิน + แบ่งสิทธิ์

ได้อะไรเมื่อรันเสร็จ

หน้า /register
สมัครแล้วสถานะ "รออนุมัติ"

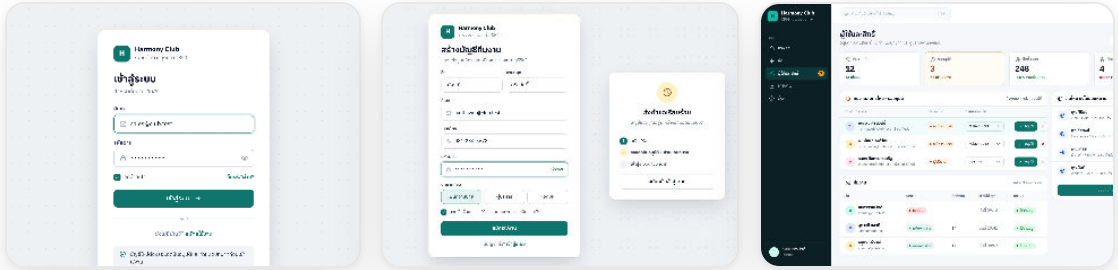
หน้า /login
ยังไม่อนุมัติ → เข้าไม่ได้

อนุมัติทีมงานใหม่
เลือกบทบาทตาม CLAUDE.md

เมนูตามบทบาท
+ ล็อกสิทธิ์ที่ฐานข้อมูล

ด่าน 1 สมัครใหม่แล้วเข้าไม่ได้จนอนุมัติ · แต่ละบทบาทเห็นเมนูต่างกัน · แก้บทบาทตัวเองไม่ได้

ก่อนกดส่ง · แนบหน้าจอตัวอย่าง 3 รูป



วาง **Project URL + publishable key** ใน `.env.local` เอง แล้วรัน `npm run dev`

ไม่ส่งคีย์ในแชต · ห้ามใช้ secret key ในหน้าเว็บ

PHASE 1 · PROMPT

อ่าน CLAUDE.md แล้วทำ Phase 1

P1-BUILD · สมัคร + ล็อกอิน + 3 บทบาท

```
อ่าน CLAUDE.md แล้วทำ Phase 1: สมัคร + ล็อกอิน + แบ่งสิทธิ์ตามบทบาทที่กำหนดใน CLAUDE.md

ฐานข้อมูล (เขียนไฟล์ supabase/phase1.sql ให้ฉันไปวางใน Supabase SQL Editor เอง)
- ตาราง profiles ผูกกับ auth.users: ชื่อ · อีเมล · เบอร์โทร · บทบาทที่ขอ · บทบาท (ตาม CLAUDE.md · ว่างได้) · สถานะ (pending / active / rejected)
- สมัครเสร็จ ให้ trigger สร้างแถว profiles อัตโนมัติ สถานะ pending
- Row Level Security ตามตารางสิทธิ์ใน CLAUDE.md: ทุกคนอ่านแถวของตัวเองได้ · เฉพาะบทบาทที่มีสิทธิ์อนุมัติเท่านั้นที่แก้บทบาท / สถานะได้ แม้เรียกฐานข้อมูลตรง

หน้าเว็บ (Next.js App Router + @supabase/ssr)
- /register สมัครใช้งาน: ชื่อ นามสกุล อีเมล เบอร์โทร รหัสผ่าน บทบาทที่ขอ → ขึ้น "รอการอนุมัติ"
- /login เข้าสู่ระบบ: ถ้ายังรออนุมัติ ให้แจ้งและห้ามเข้า
- /app หลังล็อกอิน แสดงเมนูตามบทบาทใน CLAUDE.md · หน้าอนุมัติทีมงานใหม่ (อนุมัติ / ปฏิเสธ + เลือกบทบาท) สำหรับบทบาทที่มีสิทธิ์ · หน้าของงานอื่นเว้นว่างไว้ทำใน Phase 2-3
- ดีไซน์สะอาด มืออาชีพ ตามหน้าจอตัวอย่างที่แนบ [แนบรูป mock-login · mock-register · mock-admin]

คีย์
- สร้างไฟล์ .env.local ที่มี NEXT_PUBLIC_SUPABASE_URL และ NEXT_PUBLIC_SUPABASE_PUBLISHABLE_KEY เว้นค่าว่างไว้ ฉันวางคีย์เอง
- ห้ามใช้ secret key ในโค้ดฝั่งหน้าเว็บ

ทำเสร็จแล้วบอก: วิธีรัน (npm run dev) · SQL 1 บรรทัดตั้งบัญชีแรกของฉันเป็นบทบาทที่มีสิทธิ์สูงสุด · เช็คลิสต์ทดสอบด่าน 1 แล้วหยุดรอฉันทดสอบ
```

15:00–15:15 · พักเบรก

เบรก

ต่อด้วย Phase 2 · ลีด CRM



PHASE 2 · ลีด CRM

Phase 2 สร้างทีละชั้น ทดสอบทุกชั้น

1 · Database

leads + contact_logs + ลีดจำลอง 30 ราย + RLS ตามตารางสิทธิ์ใน CLAUDE.md · **ทดสอบ:** เห็นตารางใน Table Editor

2 · Backend

app/api/join/route.ts รับฟอร์ม สร้างลีดด้วย secret key ใน .env.local · **ทดสอบ:** ส่งข้อมูลแล้วลีดเข้า

3 · Frontend

/join · หน้าลีดของแต่ละบทบาทตามสิทธิ์ · มอบหมาย · เปลี่ยนสถานะ · **ทดสอบ:** ด่าน 2

P2-BUILD · ลีด CRM

```
อ่าน CLAUDE.md แล้วทำ Phase 2: ลีด CRM (ต่อจาก Phase 1 · ห้ามแก้ระบบล็อกอินที่ผ่านแล้ว)
ทำทีละชั้น Database → Backend → Frontend เสร็จแต่ละชั้นหยุดให้ฉันทดสอบก่อน

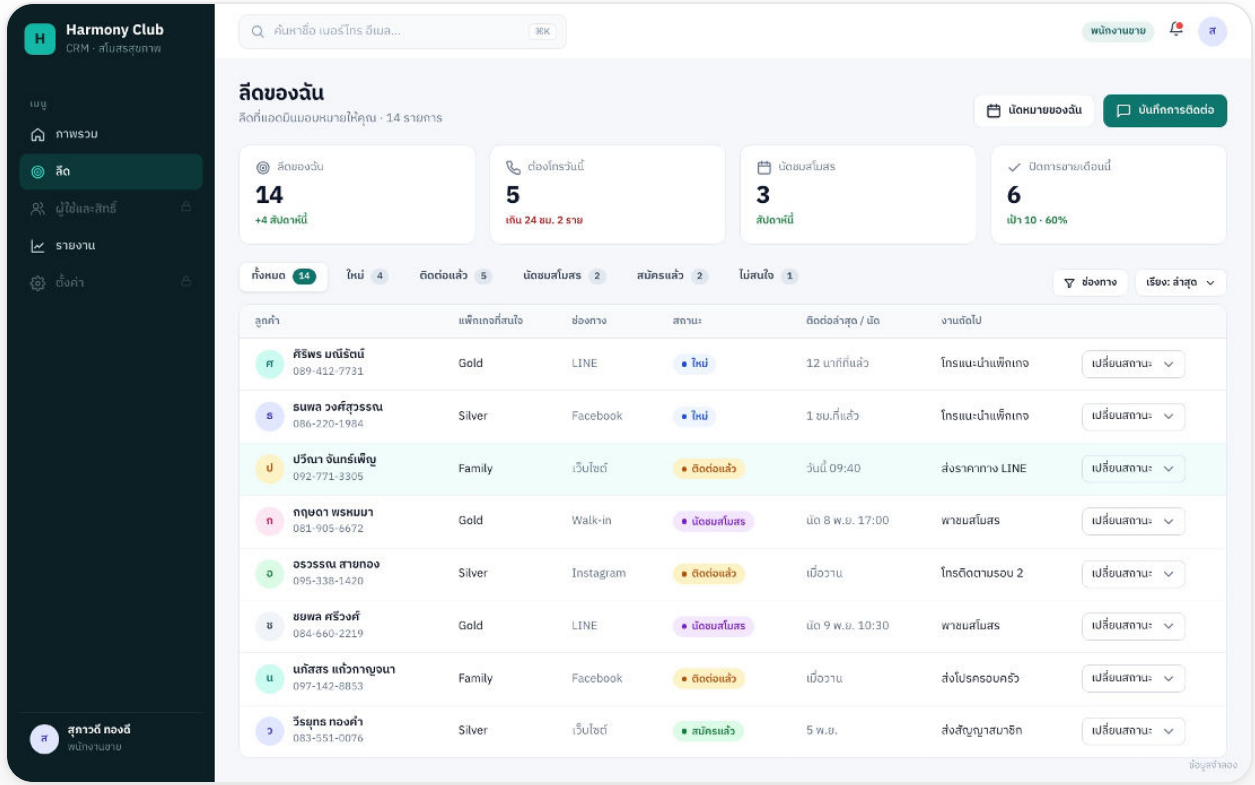
1. Database (เขียน supabase/phase2.sql ให้ฉันไปวางเอง)
- leads: ข้อมูลลูกค้าตามที่กำหนดใน CLAUDE.md + status (new / contacted / tour / member / lost) +
assigned_to (→ profiles) + created_at / updated_at
- contact_logs: lead_id · staff_id · note · created_at
- ข้อมูลจำลอง 30 ลีด กระจายให้ทีมงานที่มีในระบบ
- Row Level Security ตามตารางสิทธิ์ใน CLAUDE.md: ใครเห็นลีดไหน · ใครแก้ได้ · ใครมอบหมายได้ · ใครลบได้

2. Backend
- Route Handler app/api/join/route.ts รับฟอร์มสนใจ ตรวจข้อมูล แล้วสร้างลีดสถานะ new ด้วย
SUPABASE_SECRET_KEY ใน .env.local (ฉันวางคีย์เอง · ใช้ฝั่ง server เท่านั้น)

3. Frontend
- /join ฟอร์มสนใจสมัครสำหรับลูกค้า (ไม่ต้องล็อกอิน)
- หน้าลีดของแต่ละบทบาทตามตารางสิทธิ์ใน CLAUDE.md: ค้นหา กรอง · มอบหมาย · แท็บตามสถานะ · เปลี่ยนสถานะ · บันทึก
การติดต่อ · ไม่แสดงปุ่มที่บทบาทนั้นไม่มีสิทธิ์
- ดีไซน์เดียวกับ Phase 1 ตามหน้าจอตัวอย่างที่แนบ [แนบรูป mock-sales]
```

PHASE 2 · ทดสอบ · ด่าน 2

ลีดเข้าระบบ แล้วแต่ละบทบาทเห็นไม่เหมือนกัน



เป้าหมาย: หน้า "ลีดของฉัน" ของพนักงานขาย

ด่าน 2

- **ลูกค้ากรอก /join แล้วลีดเข้า**
  สถานะ new
- **มอบหมายลีดให้พนักงาน A**
  A เห็น · B ไม่เห็น (ตามสิทธิ์ใน CLAUDE.md)
- **A เปลี่ยนสถานะ + บันทึกการติดต่อได้**
  ลบไม่ได้
- **ผู้บริหารดูได้อย่างเดียว**
  แก้ไม่ได้

ผ่านแล้วบันทึกจุดย้อนกลับ

16:30 · สรุปวัน + การบ้าน

วันนี้ได้ระบบ Phase 1–2 · Phase 3 ทำต่อที่บ้าน

SDLC + ระบบ 3 ชั้น

เก็บความต้องการ วิเคราะห์ ออกแบบ ก่อนสั่งสร้าง · Next.js + Supabase

Main Agent + CLAUDE.md

บทบาทที่เราออกแบบเอง แบ่งเฟส สั่งทีละเฟส ทดสอบทีละด่าน

2 ด่านของการล็อกอิน

Authentication · Authorization + Row Level Security

P3-BUILD · การบ้าน · Dashboard + ขึ้นเว็บ

อ่าน CLAUDE.md แล้วทำ Phase 3: Dashboard + ขึ้นเว็บ
1. หน้า Dashboard สำหรับบทบาทที่ดูภาพรวมได้ตาม CLAUDE.md ตามหน้าจอตัวอย่างที่แนบ [แนบรูป mock-manager]: ตัวเลขสรุป (ลีดใหม่ · อัตราสมัคร · สมาชิกใหม่ · ลีดค้าง) · กราฟลีดรายสัปดาห์ · ลีดตามช่องทาง · ขั้นการขาย · ผลงานทีม
2. ขึ้นเว็บบน Cloudflare: ตั้งค่าโปรเจกต์ Next.js ด้วย OpenNext adapter (@opennextjs/cloudflare) ให้ deploy ได้ด้วย npm run deploy · ตั้งคีย์ฝั่ง Cloudflare (ห้ามใส่ในโค้ด) · บอกให้ฉันเพิ่ม URL เว็บใน Supabase → Authentication → URL Configuration
3. บอกวิธีทดสอบบนมือถือ: ล็อกอินได้ทุกบทบาท และเห็นข้อมูลตรงตารางสิทธิ์

เพื่อให้ผู้เรียนเข้าใจพื้นฐานอย่างแท้จริงภายใน 1 วัน คลาสนี้สอนเฉพาะแกนกลางของระบบ — โมดูลอื่นต่อยอดได้ด้วยวิธีเดียวกัน: เพิ่ม Phase 4, 5 ... ใน CLAUDE.md

zenityx

สรุปเนื้อหา · ถาม-ตอบ

Q&A

ขอบคุณที่ร่วมเรียนวันนี้

ทุกคำถามยินดีตอบ — ทั้งระบบที่สร้างวันนี้ และการต่อยอดในงานจริงของสโมสร

โทร
062-262-9644

อีเมล
admin@zenityxai.com

