1SHARING100 CLUB · DAY 1

Full-Stack Vibe Coding

ระบบจองห้องพักตามสิทธิ์สมาชิก

Day 1: Login · บทบาท · สิทธิ์ตามระดับสมาชิก

วันพุธที่ 25 พฤศจิกายน 2569 | 10.00–17.00 น. | Claude Code



STEP 01
เข้าใจโจทย์
Vibe Coding · SDLC · Role ≠ Tier

STEP 02
Phase 1
Login + 2 บทบาท

STEP 03
Phase 2
นำเข้าสมาชิกจาก CSV

STEP 04
Phase 3
สิทธิ์ตามระดับ + ขึ้นเว็บ

GETTING STARTED

วิธีใช้คู่มือเล่มนี้

คู่มือเรียงตามลำดับของวันเรียน ช่วงเช้าทำความเข้าใจโจทย์ของสโมสรและออกแบบระบบ ช่วงบ่ายสั่ง Claude Code สร้างทีละเฟส ทั้งวันใช้แค่ 7 พรอมต์ (เช้า 3 · บ่าย 4) ทุกพรอมต์มีคำอธิบายภาษาง่าย ปุ่มคัดลอก และภาพหน้าจอจริงที่ควรเห็นเมื่อทำเสร็จ

ปุ่มคัดลอก
ในคู่มือฉบับเว็บ ทุกกล่อง Prompt และไฟล์ SQL มีปุ่มคัดลอกที่มุมขวาบน

[ ] ตัวแดงในวงเล็บ
ข้อความที่เปลี่ยนเป็นของตัวเองก่อนกดส่ง

ด่านทดสอบ
สิ่งที่ต้องเห็นบนจอจริง ก่อนไปเฟสถัดไป

เมื่อรัน + ภาพจริง
ใต้ทุก Prompt บอกว่ารันเพื่ออะไร ได้อะไร และภาพหน้าจอจริงจากเดโม

LEARNING OUTCOMES

ผลลัพธ์หลังเรียน Day 1

จบวันนี้ ทีมสถานีแบ่งปันจะได้ระบบจองห้องพักต้นแบบที่รู้ว่าใครเป็นใคร ถือสิทธิ์ระดับไหน และแสดงเฉพาะห้องที่จองได้ — ขึ้นเว็บจริง เปิดบนมือถือได้

- ✓ **เว็บระบบจองห้องพักบน Cloudflare** ลิงก์จริง เปิดบนมือถือได้
- ✓ **ฐานข้อมูล Supabase** พร้อมกฎ 5 ระดับ ที่พัก 6 แห่ง ราคา และ Row Level Security
- ✓ **ระบบ Login + สมัครสมาชิก + 2 บทบาท** แอดมิน / สมาชิก
- ✓ **หลังบ้านแอดมิน 5 เมนู** ภาพรวม · สมาชิก · นำเข้า CSV · กฎสิทธิ์ · การจอง
- ✓ **หน้าสมาชิก** การ์ดสิทธิ์ + ห้องพักที่จองได้หรือถูกล็อกพร้อมเหตุผล + จองห้องจริง + แนบรูปสลิป + ประวัติการจอง
- ✓ **แอดมินแก้กฎสิทธิ์ได้เอง** โดยไม่ต้องแก้โค้ด · ตรวจสลิป ยืนยัน/ยกเลิกการจองได้
- ✓ **CLAUDE.md แผนทั้งระบบ** + 7 พรอมต์ + Git จุดย้อนกลับทุกด่าน

ขอบเขตของวันนี้

ครบในวันเดียว: Login · สมัครสมาชิก · 2 บทบาท · หลังบ้านแอดมินทั้งหมด (สมาชิก · นำเข้า CSV · กฎสิทธิ์ · การจอง) · ห้องพักและราคาตามสิทธิ์ · จองห้อง · แจ้งชำระด้วยรูปสลิป · ประวัติการจอง · ขึ้นเว็บ

นอกขอบเขตคอร์ส: ระบบชำระเงินออนไลน์จริง · ส่งอีเมลจริง · ใช้ข้อมูลส่วนบุคคลจริง · ผูกโดเมน

DAY OVERVIEW

ตารางเวลา

วันพุธที่ 25 พฤศจิกายน 2569 · 10.00–17.00 น. · เช้าเข้าใจโจทย์และเตรียมเครื่อง · บ่ายสร้างทีละเฟสจนขึ้นเว็บจริง

เวลา	ช่วง	เนื้อหา
10.00–10.30	**Session 1 · วิวัฒนาการ Vibe Coding**	4 ยุค · AI ลงมือ เราเป็นคนกำกับ · Vibe Coding ไม่ใช่การเดาสุ่ม
10.30–11.10	**Session 2 · ออกแบบระบบก่อนสั่ง AI**	ฟังโจทย์ลูกค้า · บทบาท ≠ ระดับ · 5 ระดับสมาชิก · ด่านตรวจสิทธิ์ · ใครทำอะไรได้ · ตารางทดสอบ
11.10–12.30	**Workshop 1 · เตรียมเครื่อง + วางแผน**	เชื่อม MCP Supabase + Cloudflare · พรอมต์ 1 เตรียมเครื่อง · วางคีย์ลับเอง · พรอมต์ 2 เว็บ "สวัสดี" · พรอมต์ 3 CLAUDE.md
12.30–13.30	พักกลางวัน	
13.30–14.10	**Workshop 2 · Phase 1**	พรอมต์ 4 · Login + สมัครสมาชิก + บทบาท · แอดมินเข้าได้ · สมัครแล้วรอแอดมินอนุมัติ
14.10–15.00	**Workshop 3 · Phase 2**	พรอมต์ 5 · หลังบ้านแอดมินทั้งหมด · นักเรียนทดสอบนำเข้า CSV เอง → สมาชิกครบ 5 ระดับ
15.00–15.15	พักเบรก	
15.15–16.30	**Workshop 4 · Phase 3**	พรอมต์ 6 · สิทธิ์ตามระดับ + จองห้อง + แจ้งชำระด้วยสลิป + ประวัติ · ต้องจ่ายก่อนจองใหม่ · ห้ามจองซ้อนวัน
16.30–16.50	**ขึ้นเว็บจริง + ทดสอบบนมือถือ**	พรอมต์ 7 · ล็อกอินครบ 5 ระดับ + จองจริง 1 ครั้งบนมือถือ
16.50–17.00	**สรุปวัน + ถาม-ตอบ**	ระบบจองห้องพักครบในวันเดียว

> **หมายเหตุถึงผู้เรียน**
> ตารางเวลาเป็นแนวทาง ผู้สอนอาจปรับเวลาแต่ละช่วงให้เหมาะกับความคืบหน้าของผู้เรียน โดยเนื้อหาและผลลัพธ์หลักยังครบถ้วน

BEFORE CLASS

ก่อนวันเรียน

เครื่องส่วนใหญ่พร้อมแล้วจากคลาส Basic CRM System — เตรียมบัญชีให้ครบ ที่เหลือให้ Agent ช่วยใน Workshop 1

- ✓ **โน้ตบุ๊กที่ใช้ Claude Code ได้**
  บัญชี Claude Pro ขึ้นไป + Git + Node.js 20 ขึ้นไป (ติดตั้งแล้วจากคลาส CRM)
- ✓ **บัญชี GitHub · Supabase · Cloudflare**
  สมัครฟรีไว้ก่อนวันเรียน
- ✓ **มือถือสำหรับทดสอบเว็บจริง**
  ใช้ช่วงท้ายวัน ล็อกอินครบ 5 ระดับ
- ✓ **ชุดไฟล์ starter (SQL + CSV)**
  ดาวน์โหลดจาก Hub หรือปุ่มในหัวข้อ 1.0 เก็บไว้ในโฟลเดอร์ Downloads

กติกา 3 ข้อของห้องเรียน

ใช้ข้อมูลจำลอง
สมาชิก อีเมล และเบอร์โทรในคลาสเป็นข้อมูลจำลองทั้งหมด (PDPA) · ไม่ใช้รายชื่อจริงจากฟอร์ม Join Now

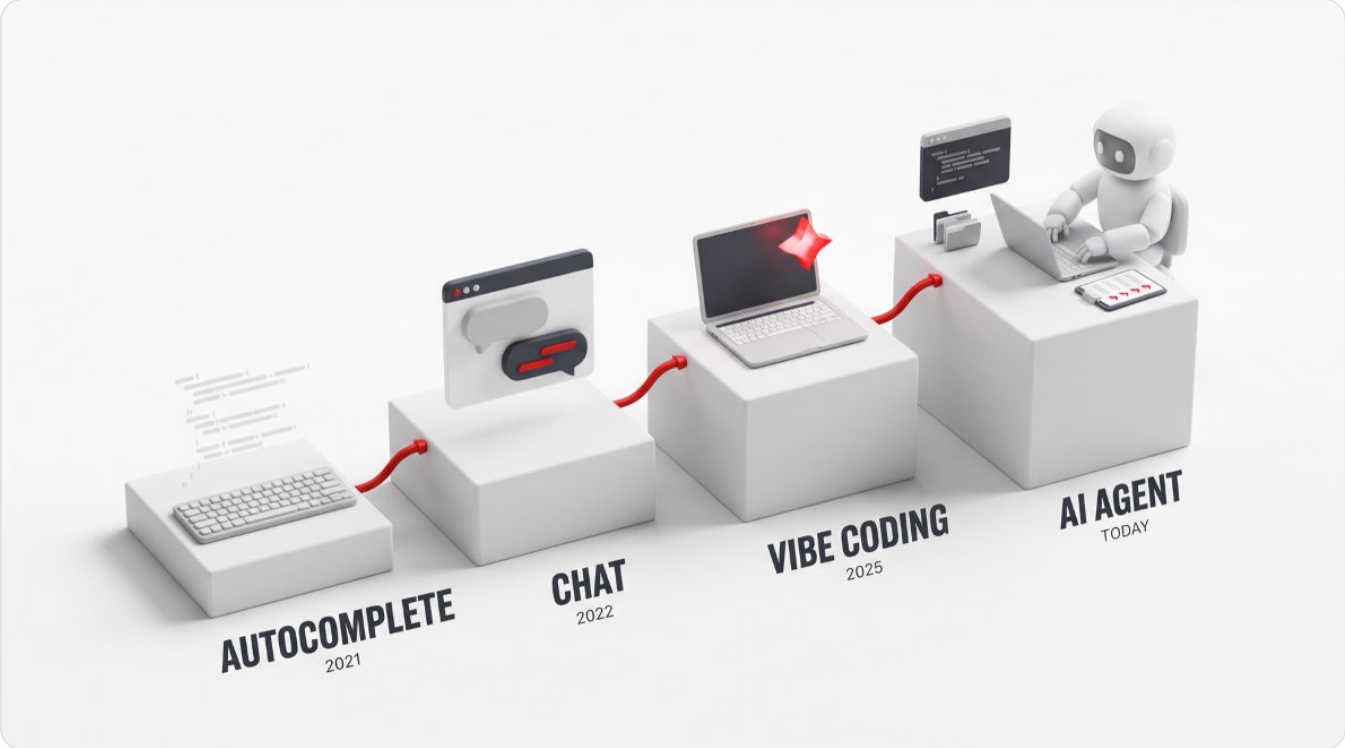
คีย์ลับอยู่ใน .env.local
วางคีย์เองในไฟล์ `.env.local` ห้ามวางในแชต ห้ามอยู่ในโค้ดหรือ Git

คนตรวจทุกด่าน
Agent เขียนโค้ด คนออกแบบ ทดสอบ ตามด่าน และตัดสินใจ

SESSION 1 · 10.00–10.30

S1 วิวัฒนาการ Vibe Coding

เครื่องมือเปลี่ยนไปมากใน 4–5 ปี จาก AI ที่ช่วยเดาบรรทัดถัดไป จนเป็น Agent ที่ลงมือทำในเครื่องเราเอง — **บทบาทของคนจึงเปลี่ยนตาม**



ยุค 1 · ราว 2021
AI เติมโค้ด
เดาบรรทัดถัดไป คนยังเขียนเองทั้งหมด (เช่น GitHub Copilot)

ยุค 2 · ปลาย 2022
แชตถาม-ตอบ
ถามแล้วคัดลอกโค้ดไปวางเอง ต้องรู้ว่าวางตรงไหน (เช่น ChatGPT, Claude)

ยุค 3 · 2025
Vibe Coding
อธิบายสิ่งที่อยากได้ด้วยภาษาคน แล้วให้ AI เขียนทั้งก้อน · คำนี้เริ่มใช้โดย Andrej Karpathy ต้นปี 2025

ยุค 4 · 2025 → วันนี้
AI Agent ในเครื่อง
อ่านทั้งโปรเจกต์ รันคำสั่ง ทดสอบ แก้เอง (เช่น Claude Code)

บทบาทคนเปลี่ยน · สิ่งที่ไม่เปลี่ยน

คนเปลี่ยนบทเป็น

ผู้ออกแบบ · ผู้ตรวจ · ผู้ตัดสินใจ

Agent ลงมือเขียนและรันโค้ด ส่วนเราบอกว่าต้องการอะไร ตรวจผลบนจอจริง และเลือกทางเมื่อมีหลายทาง

สิ่งที่ไม่เปลี่ยน

- requirement ต้องชัด
- ต้องทดสอบ
- ความปลอดภัยเป็นหน้าที่ของคน

> **"Vibe Coding ไม่ใช่การเดาสุ่ม — โฟกัสที่ไอเดียและเหตุผล แต่ต้องมีด่านตรวจ"**

ทบทวนจากคลาส CRM

Main Agent ตัวเดียว

ทุกชั้นของระบบพึ่งกัน ให้ Agent ตัวเดียวสร้างทั้งระบบ ผลออกมาเข้ากัน

CLAUDE.md

คู่มือโปรเจกต์ที่ Agent อ่านเองทุกครั้งที่เปิดงาน สั่งงานสั้นลงเหลือ "อ่าน CLAUDE.md แล้วทำ Phase 1"

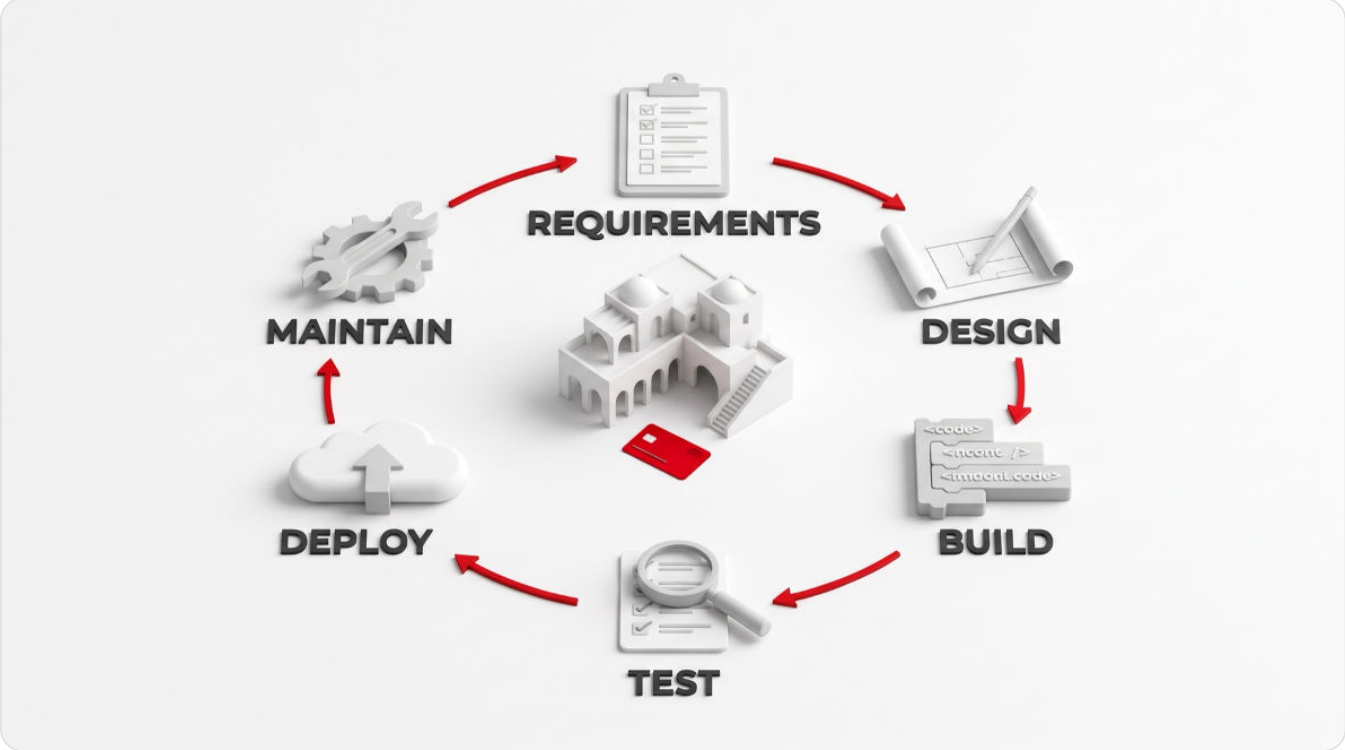
ทีละเฟส + Git

ทำทีละเฟส ทดสอบผ่านด่านแล้วบันทึกจุดย้อนกลับด้วย Git ทุกครั้ง

SESSION 2 · 10.30–11.15

S2 SDLC + ระบบ 3 ชั้น กับโจทย์จริง

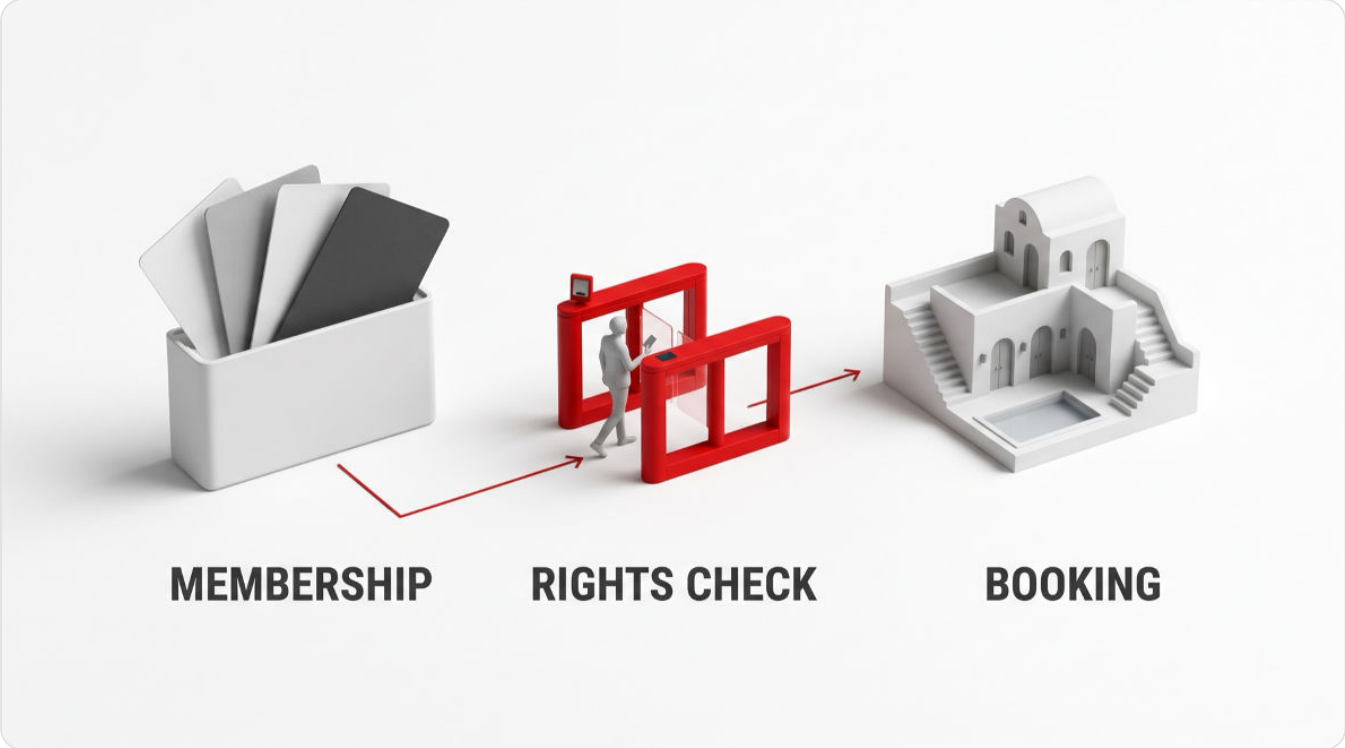
ใช้ 6 ขั้นของการพัฒนาระบบ (SDLC) กับโจทย์ของสโมสรจริง ตั้งแต่ฟังเสียงลูกค้า จนแอดมินดูแลระบบได้เอง



#	ขั้น	ในระบบจองห้องพักนี้
1	**Requirement**	เสียงลูกค้า → รายการเงื่อนไข
2	**วิเคราะห์ + ออกแบบ**	Role ≠ Tier · กฎเป็นข้อมูล · ตาราง · หน้าจอ
3	**พัฒนา**	Phase 1–3 ครบในวันนี้ (รวมการจองและชำระเงิน)
4	**ทดสอบ**	บัญชีทดสอบ 5 ระดับ + Test Matrix
5	**ขึ้นระบบ**	Cloudflare (ลิงก์จริง เปิดบนมือถือได้)
6	**ดูแล**	แอดมินปรับตัวเลขเอง ไม่ต้องแก้โค้ด

โจทย์จากเสียงลูกค้า

ระบบมี 2 ส่วนที่ต้องเชื่อมกัน



ส่วนที่ 1

ระบบสิทธิ์สมาชิก

ลงทุนมากได้สิทธิ์มาก (เหมือนถือหุ้น) · มีสิทธิ์แล้วจะใช้หรือไม่ใช้ก็ได้

ส่วนที่ 2

ระบบจองห้องพัก

เหมือนจองโรงแรม จองแล้วชำระเงิน

ตัวเชื่อม

ด่านตรวจสิทธิ์

รู้ว่าเป็นใคร ถือสิทธิ์ระดับไหน แล้วแสดงเฉพาะสิ่งที่จองได้ — "เหมือนเจนรูปแล้วเด้งว่าเครดิตไม่พอ"

เงื่อนไขเพิ่มเติม (ทำใน Phase 3 วันนี้): โควตาวันต่อปี · จ่ายครั้งก่อนก่อนจองครั้งถัดไป (กันจองกักห้อง) · ห้องเต็มห้ามจองเกิน · สิทธิ์อายุ 1 ปี / เปลี่ยนระดับได้ปีละ 1–2 ครั้ง · ต่อการจองได้ถ้าโควตาเหลือ · ประวัติการจอง/ใช้สิทธิ์ · ทุกตัวเลขแก้ได้ในหลังบ้าน

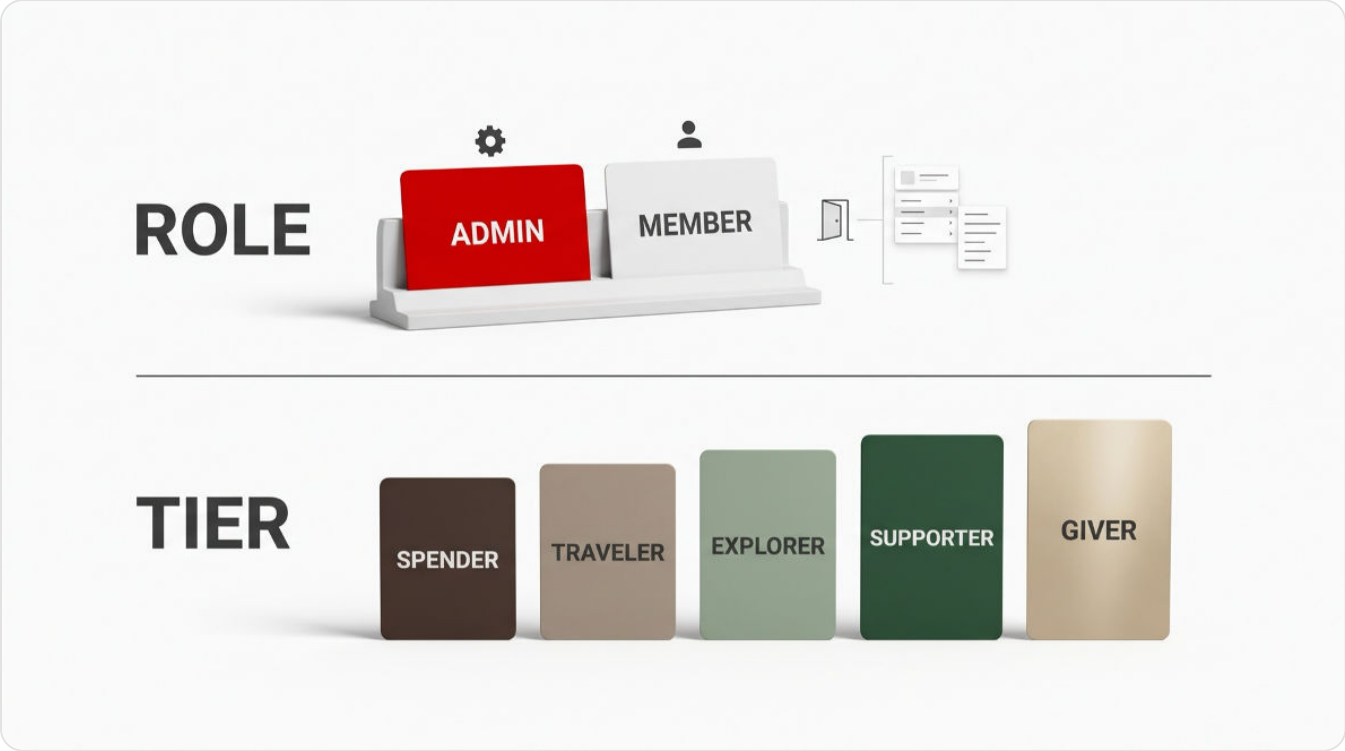
แปลงเสียงลูกค้าเป็น requirement

ทุกประโยคที่ลูกค้าเล่า ต้องตอบได้ว่า "เก็บไว้ที่ไหนในระบบ" และ "ทำวันไหน"

ลูกค้าเล่าว่า	กลายเป็นในระบบ	ทำเมื่อ
TRAVELER พักได้ไม่เกิน 90 วันต่อปี	`tiers.nights_per_year = 90`	วันนี้ · เก็บกฎ
ห้องเต็มแล้วห้ามจองเพิ่ม	`branch_rooms.total_rooms` + ตรวจตอนจอง	Phase 3
ตัวเลขพวกนี้ยังเปลี่ยนได้	หน้าแก้กฎของแอดมิน `/admin/tiers`	วันนี้ · Phase 3

ROLE ≠ TIER

บทบาท กับ ระดับสมาชิก เป็นคนละเรื่องกัน



บทบาท (role)

admin / member = กุญแจเข้าเมนู · ตัดสินว่าเข้าหน้าไหนได้ เก็บใน `profiles.role`

ระดับสมาชิก (tier)

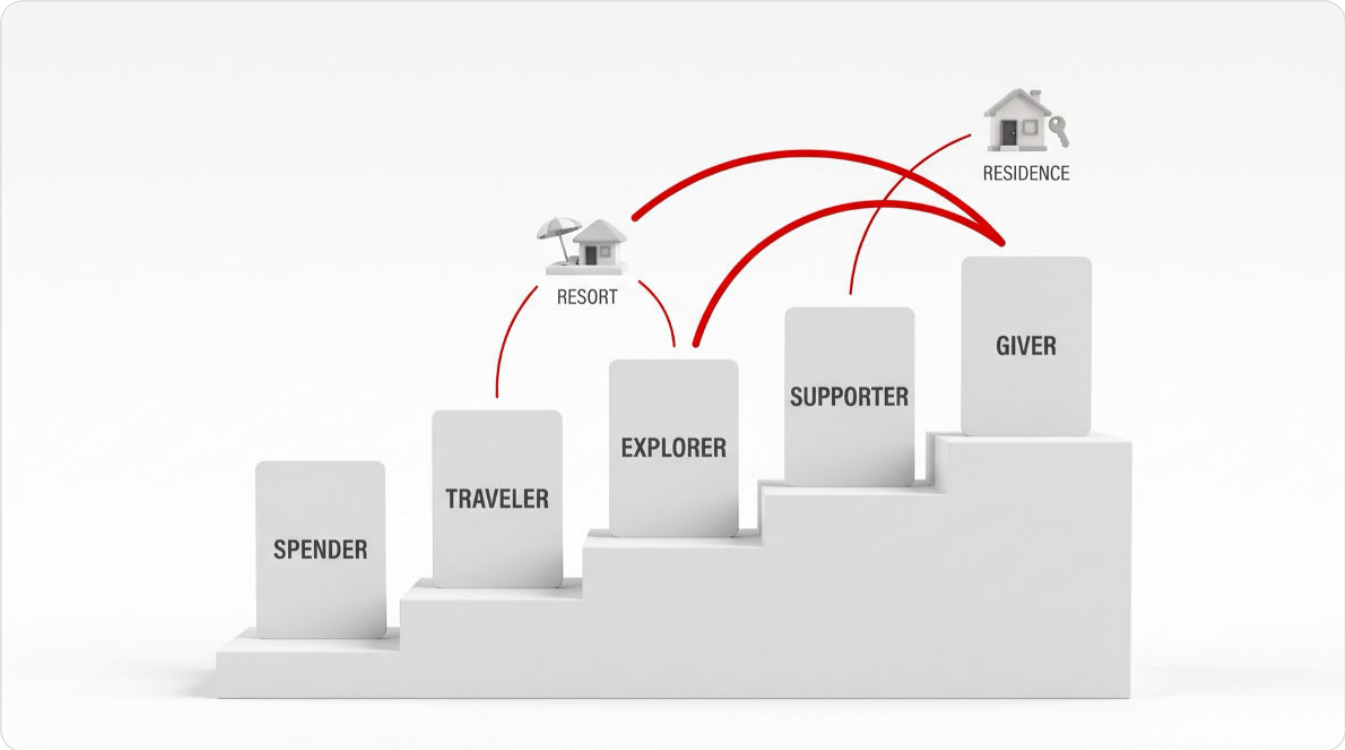
5 ระดับ = สิทธิ์จองห้อง · ตัดสินว่าจองห้องอะไรได้ ราคาเท่าไร เก็บใน `profiles.tier_code`

สมาชิกไม่เลือกระดับเอง

แอดมินกำหนดระดับจากข้อมูลการลงทุน · สมาชิกแก้ระดับของตัวเองไม่ได้ แม้เรียกฐานข้อมูลตรง (ล็อกด้วย Row Level Security)

ระดับสมาชิก 5 ระดับ

กฎสิทธิ์จากคู่มือสมาชิก (ฉบับร่าง)



ภาพประกอบแนวคิดเท่านั้น เส้นในภาพเป็นการตกแต่ง — สิทธิ์จริงดูจากตารางด้านล่าง

Tier	มูลค่า	ห้องพักประจำ	ห้องพักตากอากาศ	วัน/ปี	จองล่วงหน้า	อายุสิทธิ์	ราคา/คืน ทั่วไป · ดูแลสุขภาพ
SPENDER	1,000,000 บาท/ปี	ได้ · พักจริงขั้นต่ำ 6 เดือน	ได้ · ≤ 90 วัน/แห่ง	365	3 เดือน	365 วัน	1,750 · 2,500
TRAVELER	2,000,000 บาท / 200 หุ้น	ไม่ได้	ได้ · ≤ 30 วัน/แห่ง	90	6 เดือน	ตลอดชีพ	450 · 1,850
EXPLORER	3,880,000 บาท / 388 หุ้น	ไม่ได้	ได้ · ≤ 90 วัน/แห่ง	365	6 เดือน	ตลอดชีพ	450 · 1,850
SUPPORTER	3,880,000 บาท / 388 หุ้น	ได้ · > 90 วัน · พักจริง ≥ 50%	ไม่ได้	365	8 เดือน	ตลอดชีพ	450 · 1,850
GIVER	10,000,000 บาท / 1,000 หุ้น	ได้	ได้ · ≤ 90 วัน/แห่ง	365	1 ปี	ตลอดชีพ	450 · 1,850

SPENDER · one_of
ได้ห้องพักประจำ **หรือ** ห้องพักตากอากาศ อย่างใดอย่างหนึ่ง

GIVER · all
ได้ห้องพักประจำ **และ** ห้องพักตากอากาศ ทั้งคู่

ที่พัก 6 แห่ง 5 จังหวัด: เชียงใหม่ · บึงกาฬ · หัวหิน · หนองคาย (พักผ่อนทั่วไป) · หนองคาย · ดูแลสุขภาพ · พุทธมณฑล · ดูแลสุขภาพ — จำนวนห้องสมมติ ห้องประจำ 20 · ตากอากาศ 30 ต่อแห่ง (แก้ได้) · ระบุชื่อผู้มีสิทธิ์เข้าพักได้สูงสุด 5 ท่าน และชำระเต็มจำนวน 100% ณ วันจอง (ใช้ใน Phase 3)

จุดที่คู่มือฉบับร่างขัดกันเอง — ยืนยันกับทีมก่อนใช้งานจริง

1) หน้า 2 บอกว่า SUPPORTER ไม่มีสิทธิ์ห้องพักตากอากาศ แต่หน้า 4 มีราคา "SUPPORTER (ที่พักตากอากาศ)"
2) คู่มือเขียนอายุสิทธิ์ "ตลอดชีพ" แต่ในการประชุมเล่าว่า "1 ปีสมัครใหม่"
ระบบเก็บทุกค่าเป็นข้อมูลที่แอดมินแก้ได้ ยืนยันแล้วแก้ในหน้าแอดมินได้ทันที

กฎเป็นข้อมูล ไม่ใช่โค้ด

ตัวเลขสิทธิ์อยู่ในตาราง แอดมินแก้เองได้



✕ ฝังในโค้ด

```
if tier == 'TRAVELER' then 90
```

ตัวเลขเปลี่ยนเมื่อไร ต้องเรียกโปรแกรมเมอร์มาแก้โค้ด

✓ เก็บเป็นข้อมูล

```
tiers.nights_per_year = 90
```

ตาราง `tiers` + หน้าแก้กฎของแอดมิน เปลี่ยนแล้วมีผลทันที

ระบบ 3 ชั้น · รีสอร์ต

หน้าบ้าน · หลังบ้าน · ฐานข้อมูล



Frontend

เคาน์เตอร์ต้อนรับ — หน้าเว็บที่สมาชิกเห็นและกด เช่น หน้า Login การ์ดสิทธิ์ หน้าห้องพัก

Backend

ห้องหลังที่ถือสมุดกฎ — รับคำขอ ตรวจสิทธิ์ ใช้คีย์ลับ เช่น นำเข้าสมาชิกผ่าน `/api/admin/import`

Database

ห้องเก็บแฟ้มสมาชิกและกฎ — ตาราง Supabase + ฟังก์ชันด่านตรวจสิทธิ์ `my_room_options()`

AUTHENTICATION · AUTHORIZATION

คุณคือใคร กับ คุณทำอะไรได้

Authentication

คุณคือใคร

อีเมล + รหัสผ่าน (Supabase Auth ตรวจให้) ผ่านแล้วได้ "บัตรผ่าน" (session) ติดไปทุกคำขอ

Authorization

คุณทำอะไรได้

role (เข้าเมนูไหน) + **tier** (จองห้องอะไร) + **RLS** (ฐานข้อมูลล็อกแถวที่ไม่ใช่ของเรา)

ตัวอย่างในระบบสโมสร

traveler1 ล็อกอินได้ (Authentication ผ่าน) แต่เปิด `/admin` ไม่ได้ และเห็นห้องพักประจำเป็น "ล็อก" (Authorization ไม่ให้)

คีย์ 2 ดอกของ Supabase

publishable key

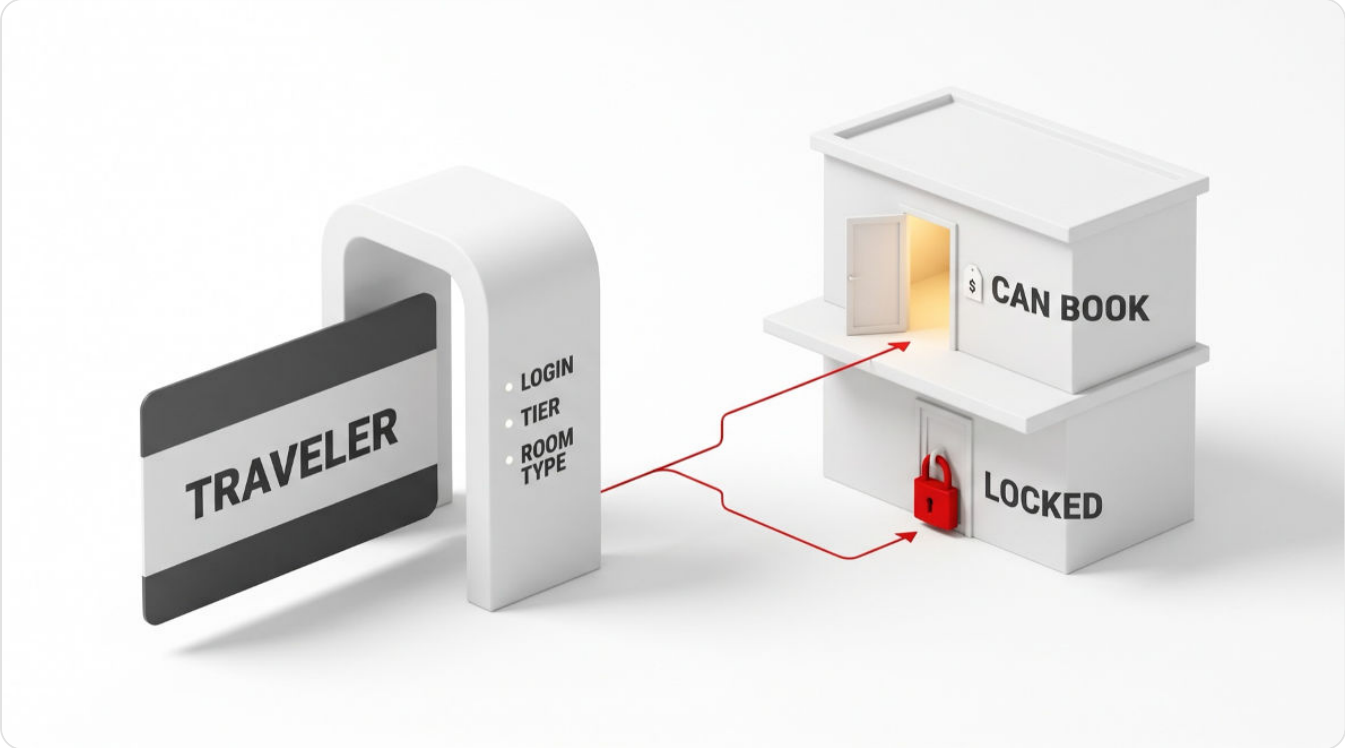
ใส่หน้าเว็บได้ · `NEXT_PUBLIC_SUPABASE_PUBLISHABLE_KEY` · ข้อมูลยังปลอดภัยเพราะ RLS เฝ้าอยู่

secret key

= กุญแจห้องเก็บของ **ใช้ฝั่ง server เท่านั้น** · `SUPABASE_SECRET_KEY` · ใช้สร้างบัญชีสมาชิกใน Phase 2

ตามรอย 1 คำขอ

TRAVELER กดดูห้องหนองคาย



1. TRAVELER กดดูห้องหนองคาย **(Frontend)**
2. หน้าเว็บส่งคำขอพร้อมบัตรผ่าน (session)
3. ฐานข้อมูลอ่าน tier ของคนที่ล็อกอินอยู่เอง — **ส่ง tier ปลอมมาไม่ได้ (Database)**
4. เทียบกฎใน `tier_room_rules` + `tier_prices` แล้วตอบกลับ: **ตากอากาศ จองได้ 450 บาท/คืน ≤ 30 วัน** · ห้องประจำ ล็อก "ระดับ TRAVELER ไม่มีสิทธิ์ห้องพักประจำ"
5. หน้าเว็บแสดงการ์ดตามผลที่ได้ — ไม่คำนวณสิทธิ์เอง

ผังตัดสินใจของด่านตรวจสิทธิ์

6 คำถามที่ระบบถามทุกครั้ง

#	ระบบถาม	ถ้าไม่ผ่าน / ผลที่ได้
1	**ล็อกอินหรือยัง**	ยัง = เห็นแค่หน้า Login
2	**บทบาทอะไร**	admin = เมนูจัดการ · member = เมนูสมาชิก
3	**มีระดับสมาชิกไหม**	ไม่มี = "ยังไม่มีระดับสมาชิก ติดต่อแอดมิน"
4	**ระดับนี้จองห้องประเภทนี้ได้ไหม**	ดู `tier_room_rules.allowed` · ไม่ได้ = ล็อก + เหตุผล
5	**ราคาเท่าไร · กี่คืน**	`tier_prices` × ประเภทที่พัก · เงื่อนไขจำนวนคืน (min/max)
6	**จองได้จริงไหม (Phase 3)**	โควตาวันเหลือไหม · ห้องว่างไหม · จ่ายครั้งก่อนแล้วหรือยัง · จองล่วงหน้าเกินกำหนดไหม

ข้อ 1–5 อยู่ในฟังก์ชัน `room_options(tier)` และ `my_room_options()` ในไฟล์ `00-setup.sql` แล้ว · ข้อ 6 อยู่ใน check_booking() ใช้ใน Phase 3

PERMISSION MATRIX

ตารางสิทธิ์ — ใครทำอะไรได้

การกระทำ	ไม่ล็อกอิน	Member	Admin
เข้าหน้า Login	ได้	ได้	ได้
ดูการ์ดสิทธิ์ของตัวเอง	ไม่ได้	ได้	ได้
ดูห้องพัก + ราคาตามสิทธิ์	ไม่ได้	เฉพาะระดับตัวเอง	ดูแทนได้ทุกระดับ
ดูรายชื่อสมาชิกทั้งหมด	ไม่ได้	ไม่ได้	ได้
นำเข้าสมาชิกจาก CSV	ไม่ได้	ไม่ได้	ได้
เปลี่ยนระดับสมาชิก	ไม่ได้	ไม่ได้ (แม้ของตัวเอง)	ได้
แก้กฎระดับ · ราคา · จำนวนห้อง	ไม่ได้	ไม่ได้	ได้

TEST MATRIX

ผลที่ต้องเห็นเมื่อทดสอบ

ใช้ตารางนี้ตรวจ Phase 3 และตอนทดสอบบนมือถือ — ล็อกอินทีละบัญชี แล้วเทียบกับหน้า "ห้องพัก"

ล็อกอินเป็น	ห้องพักประจำ	ห้องพักตากอากาศ	ราคา/คืน
`spender1`	**จองได้ · ≥ 180 คืน**	**จองได้ · ≤ 90 คืน/แห่ง**	1,750 · 2,500
`traveler1`	ล็อก	**จองได้ · ≤ 30 คืน/แห่ง**	450 · 1,850
`explorer1`	ล็อก	**จองได้ · ≤ 90 คืน/แห่ง**	450 · 1,850
`supporter1`	**จองได้ · ≥ 91 คืน**	ล็อก	450 · 1,850
`giver1`	**จองได้**	**จองได้**	450 · 1,850
แอดมินแก้ราคา TRAVELER เป็น 500		**traveler1 เห็น 500 ทันที**	—

ราคา/คืน = ที่พักทั่วไป · ที่พักดูแลสุขภาพ (หนองคาย · ดูแลสุขภาพ และ พุทธมณฑล) · รหัสผ่านสมาชิกทุกบัญชี `Member@2569` · บัญชีลงท้าย `@club.test`

WORKSHOP 1 · 11.10–12.30

W1 เตรียมเครื่อง + วางแผนระบบ

เป้าหมาย: เชื่อม MCP 2 ตัว · พรอมต์ 1 ได้โปรเจกต์ + ฐานข้อมูล + ไฟล์เก็บคีย์ · พรอมต์ 2 ได้ลิงก์เว็บ "สวัสดี" · พรอมต์ 3 ได้ `CLAUDE.md` — ครบก่อนพักเที่ยง



00 01 02 03

เครื่องมือ วันนี้ใช้อะไรบ้าง

Claude Code
ผู้ช่วยที่ลงมือทำ · อ่านทั้งโปรเจกต์ เขียนโค้ด รันคำสั่ง

Next.js
หน้าเว็บ + หลังบ้าน · เปิดดูที่ localhost:3000

Supabase
ฐานข้อมูล + ระบบล็อกอิน · เชื่อมผ่าน MCP

Cloudflare
ที่ฝากเว็บจริง · เชื่อมผ่าน MCP

Git
จุดย้อนกลับทุกครั้งที่ผ่านด่าน

Node.js
ให้โปรเจกต์รันบนเครื่องได้

MCP คืออะไร
สายเชื่อมที่ให้ Claude Code ใช้ Supabase และ Cloudflare ได้เอง — สร้างฐานข้อมูล ใส่ข้อมูล ดึงลิงก์ ตรวจเว็บ โดยเราไม่ต้องเข้าเว็บไปคัดลอกเอง · สิ่งเดียวที่ยังทำเองคือวางคีย์ลับ 1 ดอก

1.0 ดาวน์โหลด starter + เชื่อม MCP

ดาวน์โหลดไฟล์ zip เก็บไว้ในโฟลเดอร์ Downloads — ไม่ต้องแตกเอง พรอมต์ 1 จะหยิบไปใช้

ดาวน์โหลดชุดไฟล์ STARTER

club-booking-starter.zip

SQL 6 ไฟล์ · CSV 2 ไฟล์ · หน้าจอตัวอย่าง 6 ภาพ — เก็บไว้ในโฟลเดอร์ Downloads ไม่ต้องแตกเอง พรอมต์ 1 จะหยิบไปใช้ (ดาวน์โหลดจาก Hub ของคอร์สได้เช่นกัน)

ดาวน์โหลด .zip

ในไฟล์ .zip	ใช้ทำอะไร
`supabase/00-setup.sql`	ระดับ 5 ระดับ · ที่พัก 6 แห่ง · ราคา · ด่านตรวจสิทธิ์
`supabase/02 · 03 · 04 · 05`	การจอง · ตัวเลือกที่พักตอนสมัคร · ปิดช่องโหว่ · ที่เก็บรูปสลิป
`supabase/01-make-admin.sql`	สำรอง: ตั้งแอดมินด้วยมือ ถ้าสคริปต์สร้างแอดมินมีปัญหา
`data/members-demo.csv`	สมาชิกสมมติ 10 คน ระดับละ 2 คน (Phase 2)
`data/members-with-errors.csv`	ไฟล์ทดสอบการตรวจ ผ่าน 1 แถว ไม่ผ่าน 3 แถว (Phase 2)
`mockups/*.jpg`	หน้าจอตัวอย่างที่พรอมต์อ้างถึง

1. สมัคร/ล็อกอิน `supabase.com` และ `dash.cloudflare.com` ไว้ก่อน (ฟรี)
2. เปิด Terminal แล้ววางคำสั่งด้านล่าง (2 บรรทัด) · เชื่อมครั้งเดียวใช้ได้ทุกโปรเจกต์
3. พิมพ์ `claude` เปิด Claude Code → พิมพ์ `/mcp` → เลือก supabase → Authenticate → กดอนุญาตในเบราว์เซอร์ · ทำซ้ำกับ cloudflare

TERMINAL — ก่อนพรอมต์ 1 · เชื่อม MCP 2 ตัว (พิมพ์ใน Terminal)

```
claude mcp add --scope user --transport http supabase https://mcp.supabase.com/mcp
claude mcp add --scope user --transport http cloudflare https://mcp.cloudflare.com/mcp
```

เช้า · ขั้น 0 · รันเพื่อ

ให้ Claude Code คุยกับ Supabase และ Cloudflare ได้เอง ไม่ต้องคัดลอกไปมา

ทำตามนี้

1. **วาง 2 บรรทัดใน Terminal**
   เพิ่ม MCP ของ Supabase และ Cloudflare
2. **เปิด Claude Code แล้วพิมพ์ /mcp**
   เลือกทีละตัว แล้วกด Authenticate
3. **กดอนุญาตในเบราว์เซอร์**
   ล็อกอินบัญชีของตัวเอง

ได้อะไรกลับมา

✓ **Claude Code ทำงานกับฐานข้อมูลและเว็บของเราได้**

เช็คว่าผ่าน /mcp ขึ้นว่าเชื่อมแล้วทั้ง 2 ตัว

1.1 พรอมต์ 1 · เตรียมเครื่องครบในรอบเดียว

พรอมต์เดียวได้ทั้งโปรเจกต์ ฐานข้อมูล และไฟล์เก็บคีย์ · ถ้า Claude Code ถามเรื่องสร้างโปรเจกต์ Supabase ให้อ่านค่าใช้จ่ายก่อนตอบ (แพ็กเกจฟรี = 0 บาท)

PROMPT 1 — พรอมต์ 1 · เตรียมเครื่องครบในรอบเดียว

```
ช่วยเตรียมเครื่องสำหรับระบบจองห้องพัก 1SHARING100 CLUB ให้ครบในรอบเดียว ทำทีละข้อ แล้วสรุปเป็นตาราง: รายการ · สถานะ · ฉันต้องทำอะไรต่อ

1. เครื่องมือ: ตรวจว่ามี Git และ Node.js 20 ขึ้นไป ถ้าไม่มีให้บอกวิธีติดตั้ง (ถามก่อนทุกครั้งที่ต้องใช้รหัสผ่านเครื่อง)
2. โปรเจกต์: สร้าง Next.js ชื่อ club-booking ในโฟลเดอร์ Documents ด้วยคำสั่ง
   npx create-next-app@latest club-booking --ts --app --tailwind --eslint --no-src-dir --import-alias "@/*" --use-npm --yes
   แล้วติดตั้ง npm install @supabase/supabase-js @supabase/ssr papaparse และ npm install -D @types/papaparse
3. ไฟล์ตั้งต้น: แตก club-booking-starter.zip จากโฟลเดอร์ Downloads ให้ได้โฟลเดอร์ supabase/ data/ mockups/ ที่ root ของโปรเจกต์
4. MCP: ตรวจว่าเชื่อม MCP ของ Supabase และ Cloudflare แล้ว ถ้ายังไม่ได้ ให้หยุดแล้วบอกฉันว่าต้องทำอะไร
5. ฐานข้อมูล (ผ่าน MCP Supabase):
   - ถ้ายังไม่มีโปรเจกต์ชื่อ club-booking ให้สร้างใหม่ที่ Region Singapore แพ็กเกจฟรี (บอกค่าใช้จ่ายและถามฉันก่อนยืนยัน)
   - รันไฟล์ SQL ในโฟลเดอร์ supabase/ ตามลำดับ 00 → 02 → 03 → 04 → 05 (ข้ามไฟล์ 01)
   - ตรวจผล: ตาราง tiers ต้องมี 5 แถว · branches ต้องมี 6 แถว
6. ไฟล์เก็บคีย์ .env.local ที่ root ของโปรเจกต์ (ห้ามขอคีย์ลับจากฉันในแชต):
   NEXT_PUBLIC_SUPABASE_URL=                ← ดึงจาก MCP Supabase
   NEXT_PUBLIC_SUPABASE_PUBLISHABLE_KEY= ← ดึงจาก MCP Supabase
   SUPABASE_SECRET_KEY=                     ← เว้นว่าง ฉันวางเอง
   IMPORT_DEFAULT_PASSWORD=Member@2569
   ADMIN_EMAIL=admin@club.test
   ADMIN_PASSWORD=Admin@2569
   แล้วบอกฉันว่าไปคัดลอก secret key จากหน้าไหนของ Supabase
7. Git: ตรวจว่า .gitignore กันไฟล์ .env* แล้ว · เริ่ม Git · commit แรก
8. รัน npm run dev แล้วบอกให้ฉันเปิด http://localhost:3000
```

เช้า · พรอมต์ 1 · รันเพื่อ

รันพรอมต์เดียว ได้ทั้งโปรเจกต์ ฐานข้อมูล และไฟล์เก็บคีย์

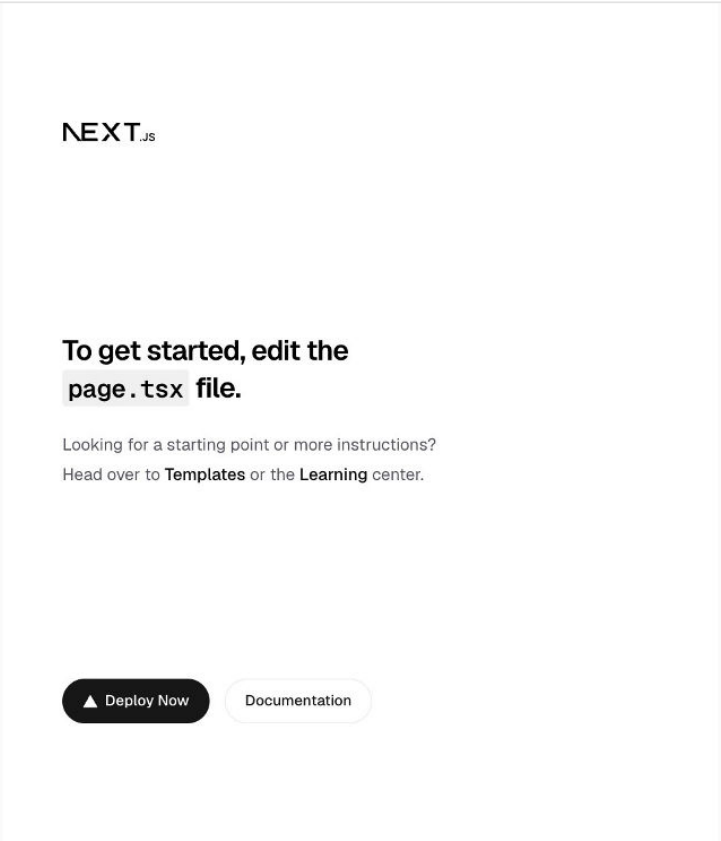
Claude Code จะทำให้

1. **สร้างโปรเจกต์**
   โฟลเดอร์ club-booking + ไฟล์ตั้งต้นจาก starter
2. **สร้างฐานข้อมูล**
   ผ่าน MCP Supabase แล้วใส่กฎ 5 ระดับ ที่พัก 6 แห่ง
3. **ทำไฟล์เก็บคีย์**
   .env.local เติมให้เกือบครบ เหลือคีย์ลับ 1 ช่อง
4. **เปิดเว็บในเครื่อง**
   localhost:3000

ได้อะไรกลับมา

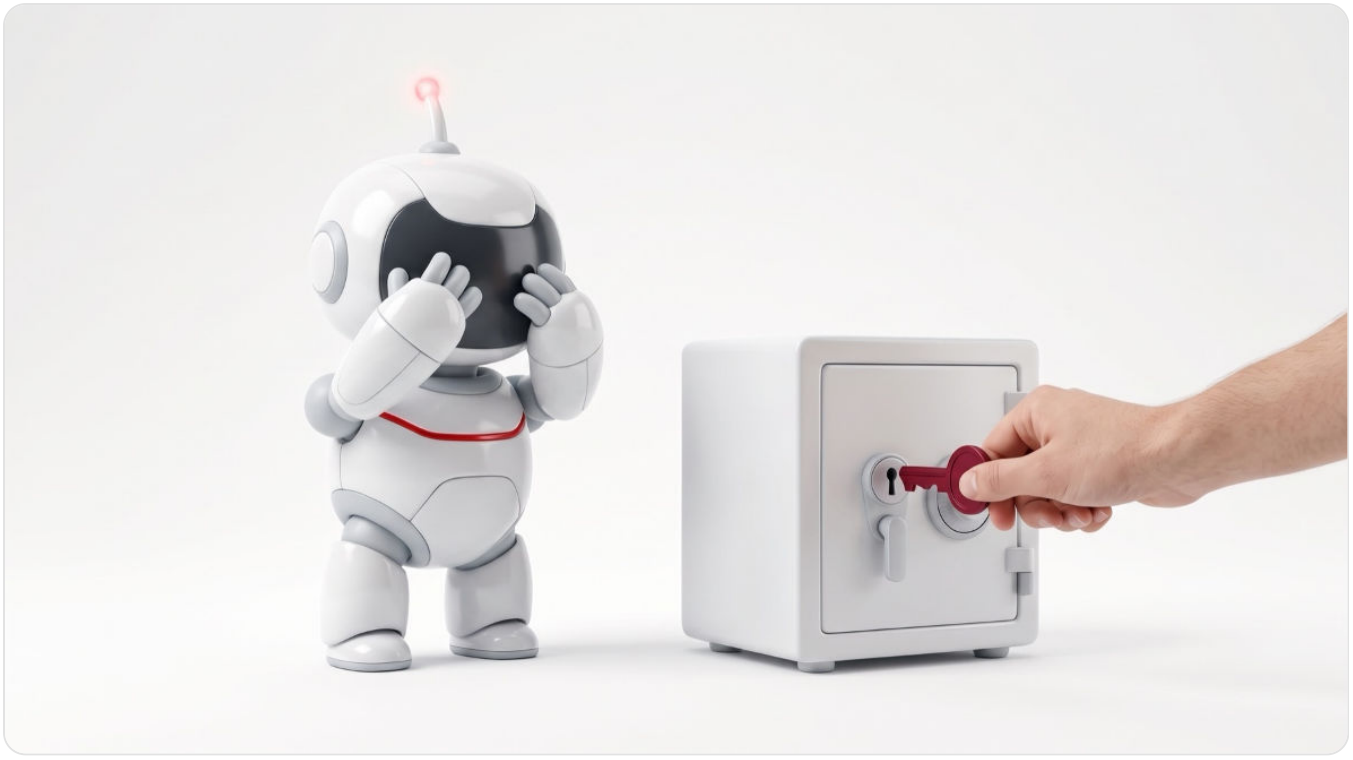
- ✓ **โฟลเดอร์โปรเจกต์พร้อมใช้**
- ✓ **ฐานข้อมูลมีระดับสมาชิก 5 ระดับ**
- ✓ **ไฟล์ .env.local รอเราวางคีย์ลับ**

เช็คว่าผ่าน เปิด localhost:3000 ได้ | ตาราง tiers มี 5 แถว | มีไฟล์ .env.local



ภาพจริง หน้าเริ่มต้นของ Next.js ที่ http://localhost:3000 — ยังไม่ใช่ระบบของเรา แค่ยืนยันว่าเครื่องพร้อม

1.2 วางคีย์ลับด้วยมือตัวเอง



1. เปิด Supabase → โปรเจกต์ club-booking → **Project Settings → API Keys**
2. คัดลอก **secret key** (กดรูปตาแล้วกดคัดลอก)
3. เปิดไฟล์ `.env.local` ในโปรเจกต์ วางต่อจาก `SUPABASE_SECRET_KEY=` แล้วบันทึก
4. บอก Claude Code ว่า "วางคีย์แล้ว" — **ไม่ต้องบอกค่าคีย์ ห้ามวางในแชต**

ตัวแปรใน .env.local	ใครเติม	ใช้ที่
`NEXT_PUBLIC_SUPABASE_URL`	Claude Code (ผ่าน MCP)	หน้าเว็บได้
`NEXT_PUBLIC_SUPABASE_PUBLISHABLE_KEY`	Claude Code (ผ่าน MCP)	หน้าเว็บได้
`SUPABASE_SECRET_KEY`	**เราวางเอง**	**server เท่านั้น**
`IMPORT_DEFAULT_PASSWORD` · `ADMIN_EMAIL` · `ADMIN_PASSWORD`	พรอมต์ 1 (ค่าของคลาส)	**server เท่านั้น**

1.3 พรอมต์ 2 · หน้าเว็บ "สวัสดี" ขึ้นเว็บจริง

เปลี่ยน **[ชื่อเล่น...]** เป็นชื่อเล่นภาษาอังกฤษของตัวเอง หรือปล่อยไว้ให้ Claude Code ถาม · ตอนหน้าต่างเบราว์เซอร์เด้งขึ้นมา ให้กดอนุญาต (Allow) เอง

PROMPT 2

พรอมต์ 2 · หน้าเว็บ "สวัสดี" ขึ้นเว็บจริง

ทำหน้าแรกของเว็บให้เรียบง่าย แล้วขึ้นเว็บจริงบน Cloudflare:
1. หน้าแรก (app/page.tsx) เหลือแค่: ชื่อ 1SHARING100 CLUB ตัวเล็กด้านบน · หัวข้อใหญ่ "สวัสดี" · ข้อความ "นี่คือสถานีแบ่งปัน สำหรับการจองห้องพัก" — พื้นขาว ตัวหนังสือเข้ม จัดกลางจอ อ่านง่ายบนมือถือ ไม่ใช้อีโมจิ
2. ตั้งค่าให้ขึ้น Cloudflare ด้วย OpenNext (@opennextjs/cloudflare) ตามวิธีล่าสุดในเอกสารทางการ ให้ขึ้นเว็บได้ด้วยคำสั่ง npm run deploy
3. ล็อกอิน Cloudflare ด้วย npx wrangler login (ฉันกดอนุญาตในเบราว์เซอร์เอง)
4. ตั้งชื่อ Worker ว่า club-booking-[ชื่อเล่นภาษาอังกฤษของฉัน] ถ้าไม่รู้ชื่อเล่นให้ถามฉันก่อน
5. deploy แล้วใช้ MCP Cloudflare ตรวจว่ามี Worker นี้แล้ว จากนั้นส่งลิงก์ .workers.dev ให้ฉันเปิดบนมือถือ
6. commit ด้วย Git
ยังไม่ต้องใส่คีย์ Supabase ฝั่ง Cloudflare (ทำตอนท้ายวัน)

เช้า · พรอมต์ 2 · รันเพื่อ

ได้ลิงก์เว็บจริงที่เปิดบนมือถือได้ ตั้งแต่ก่อนพักเที่ยง

Claude Code จะทำให้

1. **ทำหน้าแรก**
   "สวัสดี นี่คือสถานีแบ่งปัน สำหรับการจองห้องพัก"
2. **ตั้งค่าขึ้น Cloudflare**
   เราแค่กดอนุญาตในเบราว์เซอร์
3. **ขึ้นเว็บ แล้วตรวจผ่าน MCP**
   ส่งลิงก์ .workers.dev มาให้

ได้อะไรกลับมา

- ✓ **ลิงก์เว็บจริงของตัวเอง**

เช็คว่าผ่าน เปิดลิงก์บนมือถือได้ | เห็นคำว่า "สวัสดี"



ภาพจริง หน้าแรกบนคอมพิวเตอร์



ภาพจริง ลิงก์เดียวกันบนมือถือ

1.4 พรอมต์ 3 · สร้าง CLAUDE.md แผนทั้งระบบ

ผู้สอนเขียนแผนไว้ให้ครบแล้วในพรอมต์ — ไม่ต้องวิเคราะห์เอง · หน้าที่เราคือ**อ่านแล้วตรวจ**ว่าตรงกับที่ลูกค้าต้องการไหม (โดยเฉพาะหัวข้อ 3 ใครทำอะไรได้ และหัวข้อ 4 สิทธิ์ 5 ระดับ)

PROMPT 3

พรอมต์ 3 · สร้าง CLAUDE.md (แผนทั้งระบบ เขียนไว้ให้แล้ว)

สร้างไฟล์ CLAUDE.md ที่ root ของโปรเจกต์ ตามเนื้อหาด้านล่างทุกบรรทัด (ยังไม่ต้องเขียนโค้ด)
จากนั้นเปิดไฟล์ในโฟลเดอร์ supabase/ ดูว่าชื่อตารางและฟังก์ชันตรงกับที่เขียนไว้ไหม ถ้าไม่ตรงให้บอกฉัน ห้ามแก้เอง แล้ว commit

```
# CLAUDE.md — ระบบจองห้องพักตามสิทธิ์สมาชิก 1SHARING100 CLUB

## 1. เป้าหมาย
สมาชิกจองห้องพักได้ตามสิทธิ์ของระดับสมาชิก · ลงทุนมากได้สิทธิ์มาก · ใช้ข้อมูลจำลองทั้งหมด

## 2. เครื่องมือ
Next.js (App Router · TypeScript · Tailwind) · Supabase (Auth + Database + Row Level Security + Storage ผ่าน @supabase/ssr) · papaparse (อ่าน CSV) · Git · Cloudflare (OpenNext) · MCP Supabase + MCP Cloudflare

## 3. ผู้ใช้ 3 แบบ ทำอะไรได้บ้าง
### ยังไม่ล็อกอิน
- เห็นหน้าแรก · เข้าสู่ระบบ · สมัครสมาชิก
- สมัครด้วยอีเมลอะไรก็ได้ **ไม่ต้องยืนยันอีเมล** สมัครแล้วเข้าสู่ระบบได้ทันที
- สมัครเองแล้ว = สมาชิก "รอแอดมินอนุมัติ" (ยังไม่มีระดับ) ดูห้องและจองไม่ได้ จนแอดมินกำหนดระดับให้
### แอดมิน (role = admin)
- ภาพรวม: จำนวนสมาชิกแต่ละระดับ · สมาชิกรออนุมัติ · การจองที่รอตรวจสลิป
- สมาชิก: ดูรายชื่อ ค้นหา กรองตามระดับ · อนุมัติสมาชิกที่สมัครเองด้วยการกำหนดระดับ · เปลี่ยนระดับ (ปีละไม่เกินจำนวนที่ตั้งไว้)
- นำเข้าสมาชิกจากไฟล์ CSV: ตรวจทุกแถวก่อน แล้วสร้างบัญชีให้
- กฎสิทธิ์: แก้กฎแต่ละระดับ · ราคาต่อคืน · จำนวนห้องของแต่ละที่พัก · ตั้งค่าการจอง · ดูหน้าห้องพักแทนระดับใดก็ได้
- การจอง: ดูการจองและประวัติทั้งหมด · ดูรูปสลิป · ยืนยัน/ตีกลับการชำระ · ยกเลิกการจอง
### สมาชิก (role = member)
- ดูการ์ดสิทธิ์ของตัวเอง
- ดูห้องพักและราคาเฉพาะที่ระดับตัวเองจองได้ · ห้องที่ไม่มีสิทธิ์แสดงเป็นล็อกพร้อมเหตุผล
- จองห้องตามสิทธิ์ · แจ้งชำระด้วยรูปสลิป · ยกเลิกก่อนวันเช็คอิน · ดูประวัติการจอง · ต่อการจอง
- เลือกหรือแก้ระดับของตัวเองไม่ได้ · เห็นข้อมูลของคนอื่นไม่ได้

## 4. สิทธิ์ 5 ระดับ (ค่าเริ่มต้นจากคู่มือฉบับร่าง · เก็บในฐานข้อมูล แอดมินแก้ได้)
| ระดับ | ห้องพักประจำ | ห้องพักตากอากาศ | วันพัก/ปี | จองล่วงหน้า | อายุสิทธิ์ | ราคา/คืน ทั่วไป · ดูแลสุขภาพ |
|---|---|---|---|---|---|---|
| SPENDER | ได้ · อย่างน้อย 180 คืน | ได้ · ไม่เกิน 90 คืน/แห่ง | 365 | 3 เดือน | 365 วัน | 1,750 · 2,500 |
| TRAVELER | ไม่ได้ | ได้ · ไม่เกิน 30 คืน/แห่ง | 90 | 6 เดือน | ตลอดชีพ | 450 · 1,850 |
| EXPLORER | ไม่ได้ | ได้ · ไม่เกิน 90 คืน/แห่ง | 365 | 6 เดือน | ตลอดชีพ | 450 · 1,850 |
| SUPPORTER | ได้ · อย่างน้อย 91 คืน | ไม่ได้ | 365 | 8 เดือน | ตลอดชีพ | 450 · 1,850 |
| GIVER | ได้ | ได้ · ไม่เกิน 90 คืน/แห่ง | 365 | 1 ปี | ตลอดชีพ | 450 · 1,850 |
- SPENDER เลือกได้อย่างเดียว: ห้องประจำ หรือ ตากอากาศ · GIVER ได้ทั้งสองแบบ

## 5. กฎการจอง (ฐานข้อมูลตรวจให้ทุกข้อด้วย check_booking() ก่อนบันทึก)
1. มีระดับสมาชิกแล้ว (รออนุมัติ = จองไม่ได้)
2. ระดับนี้จองประเภทห้องนี้ได้
3. วันที่ถูกต้อง · จองล่วงหน้าไม่เกินที่ระดับกำหนด · สิทธิ์ยังไม่หมดอายุ
4. จำนวนคืนตามเงื่อนไขของระดับ · ระบุชื่อผู้เข้าพักไม่เกิน 5 ท่าน
5. **ต้องชำระการจองครั้งก่อนให้เสร็จก่อน** — ยังมีการจองที่รอชำระหรือรอตรวจสลิป = จองใหม่ไม่ได้
6. **ห้ามจองซ้อนวัน** กับการจองของตัวเองที่ยังไม่ยกเลิก
7. SPENDER ใช้ได้ห้องประจำหรือตากอากาศอย่างเดียวในปีนั้น
8. วันพักรวมทั้งปีไม่เกินโควตาของระดับ
9. ห้องว่างพอ (ห้องเต็มห้ามจองเกิน)
สถานะการจอง: รอชำระ → แนบรูปสลิป = รอตรวจ → แอดมินยืนยัน = ยืนยันแล้ว (แอดมินตีกลับ = กลับไปรอชำระ) · ไม่ชำระภายในเวลาที่ตั้งไว้ = หมดอายุ คืนห้อง · ยกเลิกได้ก่อนวันเช็คอิน · ชำระเต็มจำนวน 100%

## 6. ข้อมูล (โฟลเดอร์ supabase/)
- 00-setup.sql: ระดับ · ประเภทห้อง · กฎ · ที่พัก 6 แห่ง · ราคา · profiles (role + tier_code) · ด่านตรวจ
```

```
สิทธิ์ my_room_options() และ room_options(ระดับ)
- 02-day2-booking.sql: ตั้งค่าการจอง · การจอง · ประวัติเปลี่ยนระดับ · check_booking · quote_booking · create_booking · cancel_booking · my_quota · admin_confirm_payment · admin_reject_payment
- 03 · 04 · 05: ที่พักให้เลือกตอนสมัคร register_branch_options() · ปิดช่องโหว่ · ที่เก็บรูปสลิป payment-slips + submit_payment(การจอง, ที่อยู่รูปสลิป, เลขอ้างอิง)
- role (admin / member) = เข้าเมนูไหนได้ · tier_code (5 ระดับ) = จองห้องอะไรได้ ราคาเท่าไร — คนละเรื่องกัน

## 7. กติกาทีม
- กฎทุกข้อ (ระดับ ราคา จำนวนคืน จำนวนห้อง) อ่านจากฐานข้อมูล ห้ามเขียนตัวเลขหรือชื่อระดับฝังในโค้ด
- หน้าเว็บแสดงสิทธิ์จาก my_room_options() และตรวจการจองด้วย quote_booking() เท่านั้น ห้ามคิดเงื่อนไขเองในหน้าเว็บ
- สร้าง แจ้งชำระ ยืนยัน ตีกลับ ยกเลิกการจอง ผ่านฟังก์ชันในฐานข้อมูลเท่านั้น
- คีย์อยู่ใน .env.local · SUPABASE_SECRET_KEY ใช้ฝั่ง server เท่านั้น · ห้ามใส่คีย์ในโค้ดหรือ Git
- ห้ามแก้ไฟล์ใน supabase/ เอง ถ้าต้องเปลี่ยนโครงฐานข้อมูลให้เขียนไฟล์ใหม่ (เลขถัดไป) แล้วถามฉันก่อนรัน
- หน้าเว็บภาษาไทย · ไม่ใช้อีโมจิ ใช้ไอคอน SVG · ดีไซน์ตามรูปใน mockups/ · ใช้ได้ทั้งมือถือและคอมพิวเตอร์
- ทำทีละเฟส เสร็จแล้วรัน npx tsc --noEmit และ npm run lint ให้ผ่าน บอกวิธีทดสอบ แล้วหยุดรอฉัน
- เมื่อฉันบอกว่า "ผ่านแล้ว" ให้ตรวจว่าไม่มีไฟล์ลับ แล้ว commit และติ๊กเฟสนั้นในหัวข้อ 8

## 8. แผนงาน (ครบในวันเดียว)
- [ ] Phase 1 · Login + สมัครสมาชิก + บทบาท — เสร็จเมื่อ: มีบัญชีแอดมิน ล็อกอินแล้วเห็นเมนูจัดการ · รหัสผิดขึ้นข้อความไทย · ไม่ล็อกอินเข้า /app /admin ไม่ได้ · สมัครด้วยอีเมลใหม่แล้วเข้าระบบได้ทันที (ไม่ต้องยืนยันอีเมล) และเห็น "รอแอดมินอนุมัติ" · ออกจากระบบได้
- [ ] Phase 2 · หลังบ้านแอดมิน — เสร็จเมื่อ: ฉันนำเข้า data/members-demo.csv ได้ครบ 10 คน (ระดับละ 2) · data/members-with-errors.csv ผ่าน 1 แถว อีก 3 แถวขึ้นเหตุผล · อนุมัติสมาชิกที่สมัครเองได้ · เปลี่ยนระดับได้ · แก้ราคา/กฎได้ · หน้าการจองเปิดได้ · สมาชิกเข้า /admin ไม่ได้
- [ ] Phase 3 · สิทธิ์ตามระดับ + จองห้อง + ชำระเงิน + ประวัติ — เสร็จเมื่อ: ผลตรงหัวข้อ 9 ครบทุกข้อ

## 9. ตารางทดสอบ
### ดูห้องตามสิทธิ์ (ล็อกอินเป็น → ต้องเห็น)
| บัญชี | ห้องพักประจำ | ห้องพักตากอากาศ | ราคา/คืน |
|---|---|---|---|
| spender1@club.test | จองได้ · อย่างน้อย 180 คืน | จองได้ · ไม่เกิน 90 คืน | 1,750 · 2,500 |
| traveler1@club.test | ล็อก | จองได้ · ไม่เกิน 30 คืน | 450 · 1,850 |
| explorer1@club.test | ล็อก | จองได้ · ไม่เกิน 90 คืน | 450 · 1,850 |
| supporter1@club.test | จองได้ · อย่างน้อย 91 คืน | ล็อก | 450 · 1,850 |
| giver1@club.test | จองได้ | จองได้ | 450 · 1,850 |
- แอดมินแก้ราคา TRAVELER เป็น 500 → traveler1 เห็น 500 ทันที (ทดสอบแล้วตั้งกลับ 450)
### จองห้อง (ใช้ traveler1)
1. จองห้องพักตากอากาศ 3 คืน → สำเร็จ สถานะ "รอชำระ"
2. จองอีกครั้งทั้งที่ยังไม่ชำระ → ไม่ได้ ขึ้นเหตุผลให้ชำระครั้งก่อนก่อน
3. แนบรูปสลิป → "รอตรวจ" · แอดมินเห็นการจองและรูปสลิป แล้วกดยืนยัน → "ยืนยันแล้ว"
4. จองครั้งใหม่ช่วงวันอื่น → ได้ · จองช่วงวันซ้อนกับการจองเดิม → ไม่ได้
5. จองห้องพักประจำ → ไม่ได้ (ระดับไม่มีสิทธิ์)
6. ประวัติการจองขึ้นทั้งหน้า "การจองของฉัน" และหน้า "การจอง" ของแอดมิน
บัญชีทดสอบ: แอดมิน = ADMIN_EMAIL / ADMIN_PASSWORD ใน .env.local · สมาชิกที่นำเข้า = รหัส Member@2569
```

เช้า · พรอมต์ 3 · รันเพื่อ

ให้ Claude Code รู้ก่อนว่าระบบนี้ใครทำอะไรได้ ทุกเฟสหลังจากนี้อ่านไฟล์นี้

Claude Code จะทำให้

1. **เขียนไฟล์ CLAUDE.md**
   ตามเนื้อหาที่เขียนไว้ให้ครบแล้ว
2. **เทียบกับฐานข้อมูล**
   ชื่อตารางตรงไหม ไม่ตรงให้บอกเรา
3. **บันทึกด้วย Git**
   จุดย้อนกลับแรกของวัน

ได้อะไรกลับมา

- ✓ **ไฟล์ CLAUDE.md 9 หัวข้อ**
- ✓ **กฎการจอง 9 ข้อ + แผน 3 เฟส ที่จบในวันเดียว**

เช็คว่าผ่าน เปิด CLAUDE.md อ่านแล้วตรงกับที่ตกลง | ส่วนที่ไม่ตรง จดไว้ถามผู้สอน

ผลที่ได้ · CLAUDE.md 9 หัวข้อ

#	หัวข้อ	มีอะไร
1–2	**เป้าหมาย · เครื่องมือ**	ระบบจองห้องพักตามสิทธิ์ · Next.js · Supabase · Cloudflare · MCP
3	**ผู้ใช้ 3 แบบ**	ยังไม่ล็อกอิน · แอดมิน · สมาชิก — ใครทำอะไรได้เป็นข้อ ๆ
4	**สิทธิ์ 5 ระดับ**	ห้องประจำ · ตากอากาศ · วันพัก · จองล่วงหน้า · อายุสิทธิ์ · ราคา
5	**กฎการจอง**	9 ข้อที่ฐานข้อมูลตรวจให้ · **ค้างชำระจองใหม่ไม่ได้ · ห้ามจองซ้อนวัน** · โควตา · ห้องเต็ม · สถานะ รอชำระ → รอตรวจ → ยืนยันแล้ว
6	**ข้อมูล**	ไฟล์ SQL แต่ละไฟล์มีอะไร · บทบาท ≠ ระดับ
7	**กติกาทีม**	กฎอ่านจากฐานข้อมูล · คีย์อยู่ใน .env.local · ทำทีละเฟส · "ผ่านแล้ว" = commit
8	**แผนงาน**	Phase 1–3 ครบในวันเดียว · ทุกเฟสมี "เสร็จเมื่อ..."
9	**ตารางทดสอบ**	ดูห้องครบ 5 ระดับ + จองห้อง 6 ขั้น

ด่านเช้า

- พิมพ์ `/mcp` แล้วเห็นว่าเชื่อมแล้วทั้ง 2 ตัว
- เปิด `http://localhost:3000` ได้ · ตาราง `tiers` มี 5 แถว · วางคีย์ลับแล้ว
- ลิงก์เว็บ "สวัสดี" (`.workers.dev`) เปิดบนมือถือได้
- มีไฟล์ `CLAUDE.md` อ่านแล้วตรงกับที่ตกลง

LUNCH BREAK

พักกลางวัน 12.30–13.30 น.



AFTERNOON · 13.30–16.50

ช่วงบ่าย · 4 พรอมต์ สร้างทีละเฟส

พรอมต์ 4 Login → พรอมต์ 5 หลังบ้านแอดมิน → พรอมต์ 6 สิทธิ์ตามระดับ → พรอมต์ 7 ขึ้นเว็บจริง · ทุกเฟสวนรอบเดียวกัน 4 ขั้น



01	02	03	04
คัดลอกพรอมต์ของเฟสนั้น	เทียบหน้าจอกับภาพในคู่มือ	ทดสอบตาม "เช็คว่าผ่าน"	พิมพ์ "ผ่านแล้ว" → Git

WORKSHOP 2 · 13.30–14.10

P1 พรอมต์ 4 · Login + สมัครสมาชิก

เป้าหมาย: หน้าเข้าสู่ระบบ + หน้าสมัคร · แยกเมนูตามบทบาท · กันหน้า · **จบเฟสนี้: แอดมินเข้าสู่ระบบได้ · คนนอกสมัครได้แล้วรอแอดมินอนุมัติ** · สมาชิกตามระดับมาใน Phase 2



PROMPT 4

พรอมต์ 4 · Phase 1 · Login + สมัครสมาชิก + บทบาท

```
อ่าน CLAUDE.md แล้วทำ Phase 1: Login + สมัครสมาชิก + บทบาท

1. หน้าเข้าสู่ระบบ /login: อีเมล + รหัสผ่าน · รหัสผิดขึ้นข้อความภาษาไทยที่เข้าใจง่าย · เข้าได้แล้วพาไปตามบทบาท (แอดมิน → /admin · สมาชิก → /app)
2. หน้าสมัครสมาชิก /register: ชื่อ · นามสกุล · เบอร์โทร · อีเมล · รหัสผ่าน · ที่พักที่สนใจ (ดึงจาก register_branch_options())
   - ไม่ต้องยืนยันอีเมล ใช้อีเมลอะไรก็สมัครได้: สร้างบัญชีฝั่ง server ด้วย SUPABASE_SECRET_KEY (auth.admin.createUser · email_confirm: true · ใส่ชื่อ นามสกุล เบอร์ ที่พักใน user_metadata) แล้วล็อกอินให้ทันที ไม่ต้องพึ่งการตั้งค่า Confirm email ของ Supabase
   - ตรวจข้อมูลฝั่ง server ด้วย: อีเมลถูกรูปแบบ · อีเมลซ้ำให้บอกว่ามีบัญชีแล้ว · รหัสผ่านอย่างน้อย 8 ตัว
   - สมัครเสร็จแล้วเป็นสมาชิก "รอแอดมินอนุมัติ" (ยังไม่มีระดับ)
3. กันหน้า: ยังไม่ล็อกอินเข้า /app หรือ /admin ไม่ได้ (เด้งไป /login) · สมาชิกเข้า /admin ไม่ได้ · อ่านบทบาทจากตาราง profiles ของคนที่ล็อกอิน (ใช้ @supabase/ssr ทั้งฝั่ง server และ browser)
4. โครงหน้าหลังล็อกอิน: เมนูด้านซ้ายตามบทบาท + ชื่อผู้ใช้ + ระดับ (ถ้ามี) + ปุ่มออกจากระบบ
   - แอดมิน: ภาพรวม · สมาชิก · นำเข้าสมาชิก · กฎสิทธิ์ · การจอง (เว้นว่างไว้ ทำใน Phase 2)
   - สมาชิก: สิทธิ์ของฉัน · ห้องพัก · การจองของฉัน (เว้นว่างไว้ ทำใน Phase 3) · ถ้ายังรออนุมัติ แสดง "บัญชีนี้รอแอดมินอนุมัติ"
5. บัญชีแอดมินคนแรก: เขียนสคริปต์ scripts/create-admin.mjs ที่ใช้ SUPABASE_SECRET_KEY สร้างบัญชีจาก ADMIN_EMAIL + ADMIN_PASSWORD ใน .env.local (ยืนยันอีเมลให้เลย) แล้วตั้ง role = admin · รันซ้ำได้ไม่พัง · รันให้ฉัน
6. ดีไซน์ตามรูป mockups/mock-login.jpg · สะอาด มืออาชีพ · ใช้ได้ทั้งมือถือและคอมพิวเตอร์

เสร็จแล้วบอกวิธีทดสอบตาม "เสร็จเมื่อ" ของ Phase 1 แล้วหยุดรอฉันทดสอบ
```

ง่าย · พรอมต์ 4 · Phase 1 · รันเพื่อ

ได้ประตูหน้าเว็บ และบัญชีแอดมินคนแรก

Claude Code จะทำให้

1. **หน้าเข้าสู่ระบบ + หน้าสมัคร**
   รหัสผิดขึ้นข้อความไทย
2. **สมัครได้ทันที ไม่ต้องยืนยันอีเมล**
   ใช้อีเมลอะไรก็ได้ → รอแอดมินอนุมัติ
3. **แยกเมนู + กันหน้า**
   แอดมินเห็นเมนูจัดการ · สมาชิกเข้าหน้าแอดมินไม่ได้
4. **สร้างบัญชีแอดมิน**
   admin@club.test พร้อมใช้

ได้อะไรกลับมา

- ✓ **แอดมินเข้าสู่ระบบได้**
- ✓ **คนนอกสมัครแล้วเข้าได้ทันที แล้วรอแอดมินอนุมัติ**

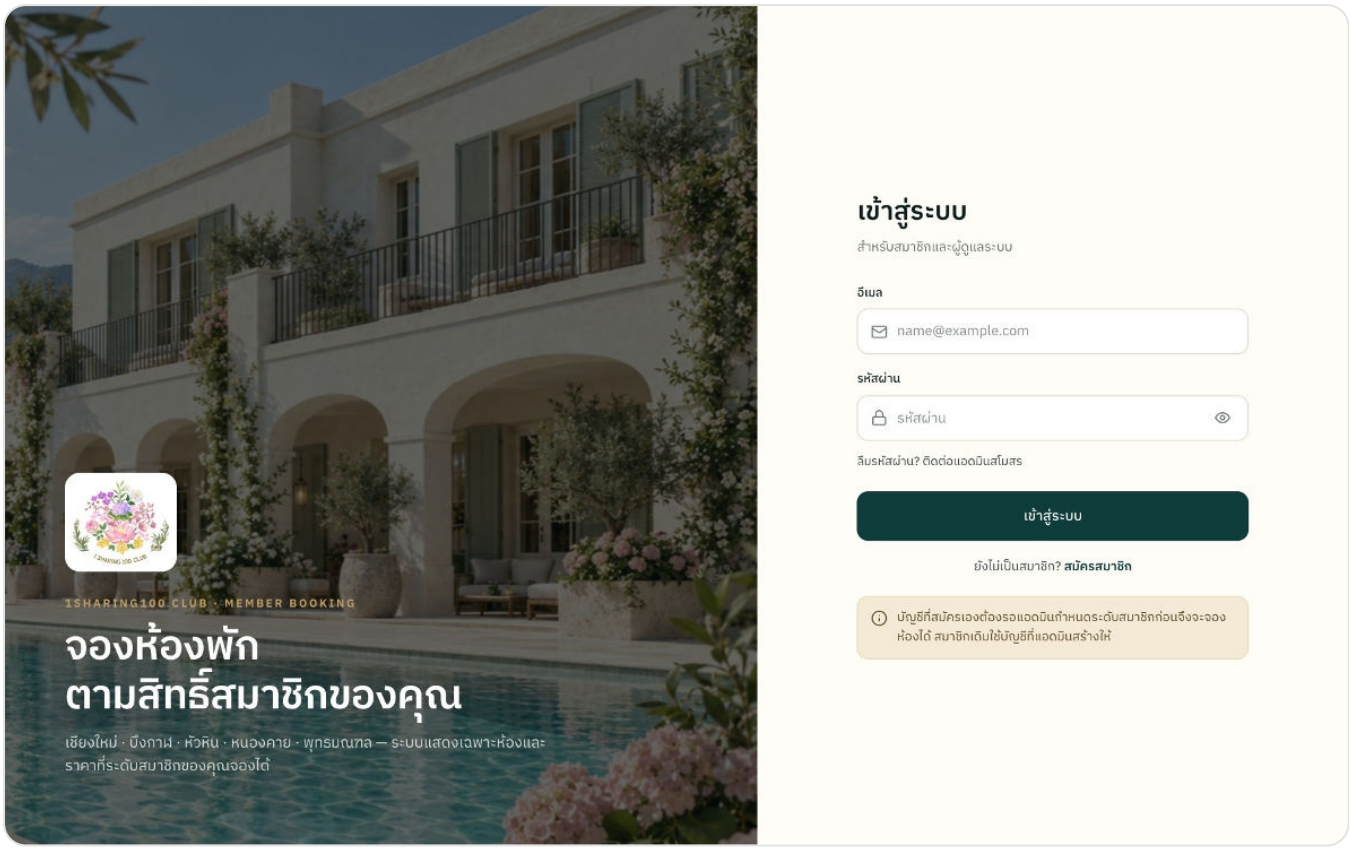
เช็คว่าผ่าน

- แอดมินล็อกอินแล้วเห็นเมนูจัดการ
- รหัสผิดขึ้นข้อความไทย
- ไม่ล็อกอินเปิด /admin แล้วเด้งไปหน้า Login
- สมัครด้วยอีเมลใหม่ → เข้าได้ทันที ไม่ต้องยืนยันอีเมล → เห็น "รอแอดมินอนุมัติ"

Claude Code ทำเสร็จแล้วเปิด `http://localhost:3000/login` → ล็อกอินด้วย `admin@club.test` / `Admin@2569`

ผลจริง · เข้าสู่ระบบ

`/login · Phase 1`

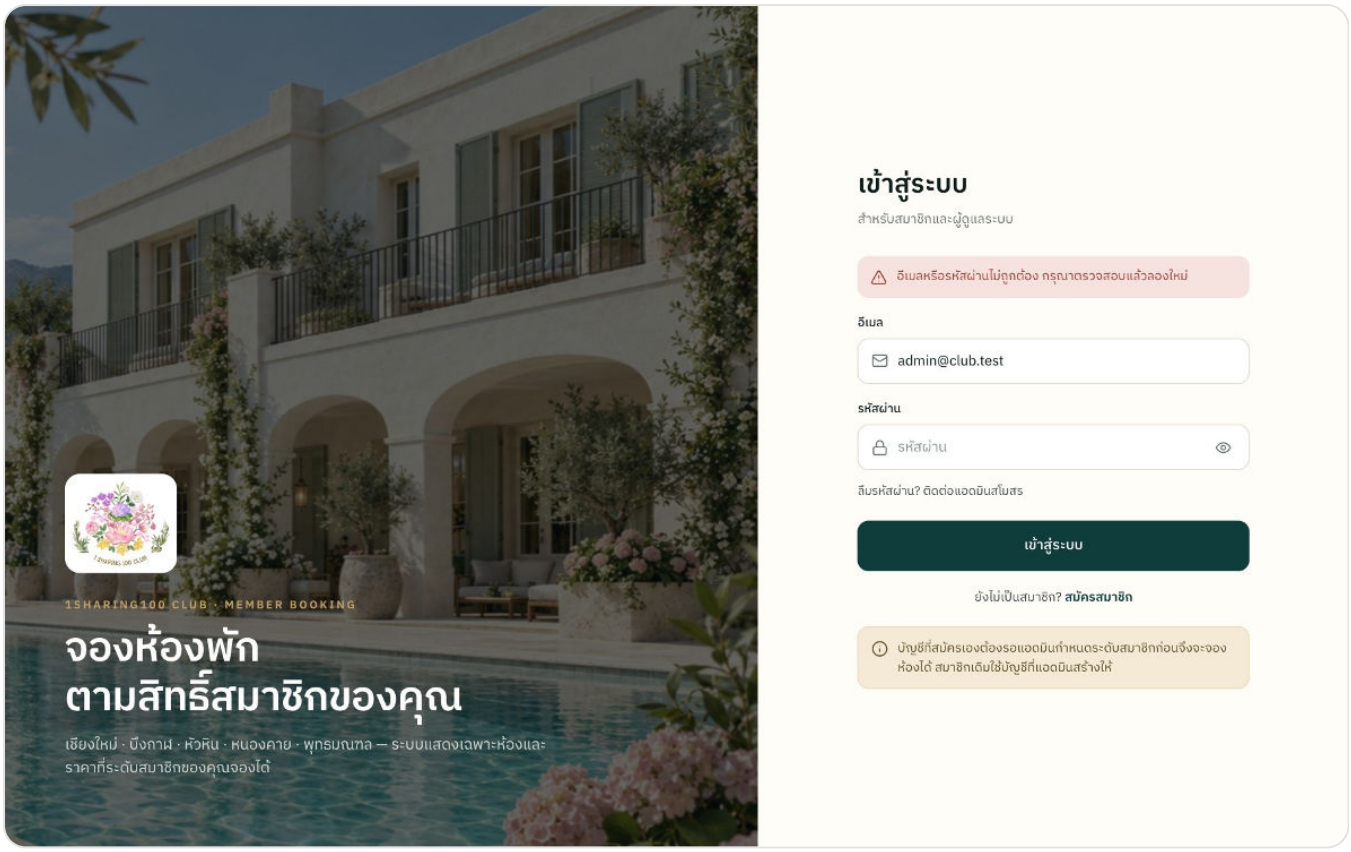


ภาพจริง หน้าเข้าสู่ระบบ · ภาพรีสอร์ตด้านซ้าย · ช่องอีเมล + รหัสผ่านด้านขวา · ข้อมูลจำลอง

> ภาพจริงจากระบบเดโมที่สร้างด้วยพรอมต์ชุดนี้ · ข้อมูลจำลอง · หน้าตาของแต่ละคนอาจต่างกันเล็กน้อย

ผลจริง · ใส่รหัสผิด

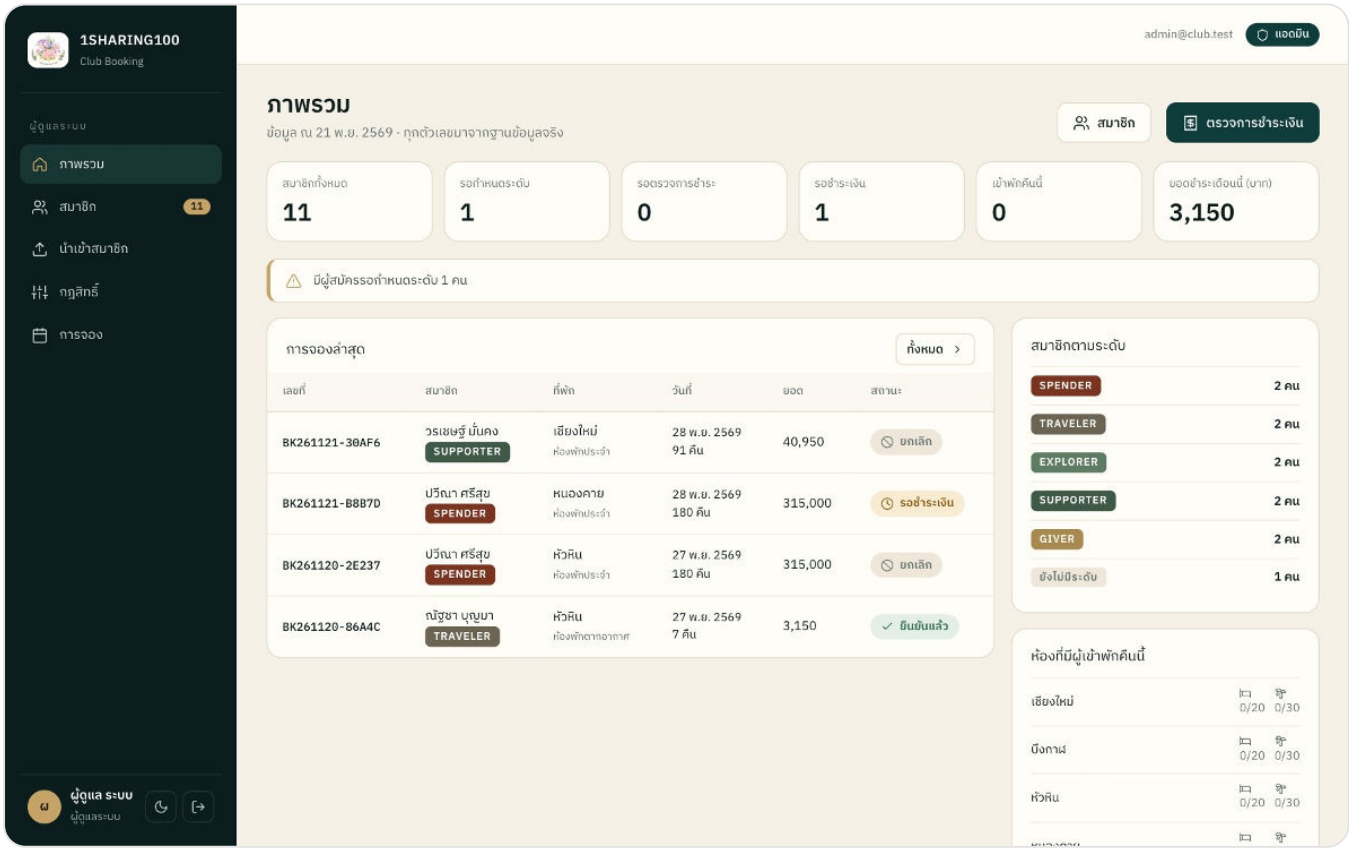
`/login`



ภาพจริง ใส่รหัสผิด → ข้อความภาษาไทยที่เข้าใจง่าย อยู่เหนือช่องอีเมล

ผลจริง · แอดมินล็อกอิน

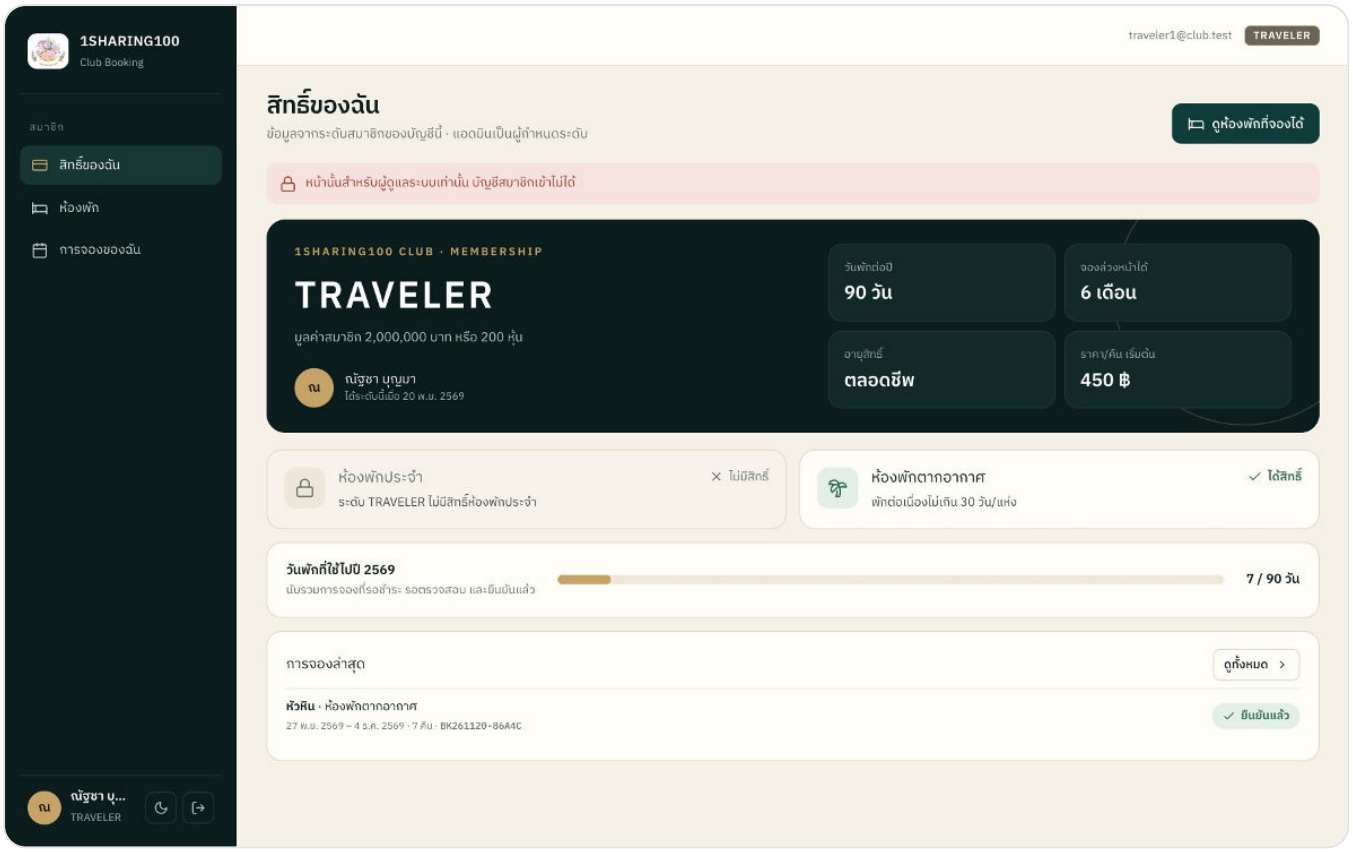
`/admin · admin`



ภาพจริง แอดมินเห็นเมนูจัดการด้านซ้าย · หน้าที่เหลือจะเติมใน Phase 2 · ตัวเลขการจองในภาพมาหลัง Phase 3

ผลจริง · สมาชิกพิมพ์ /admin เอง

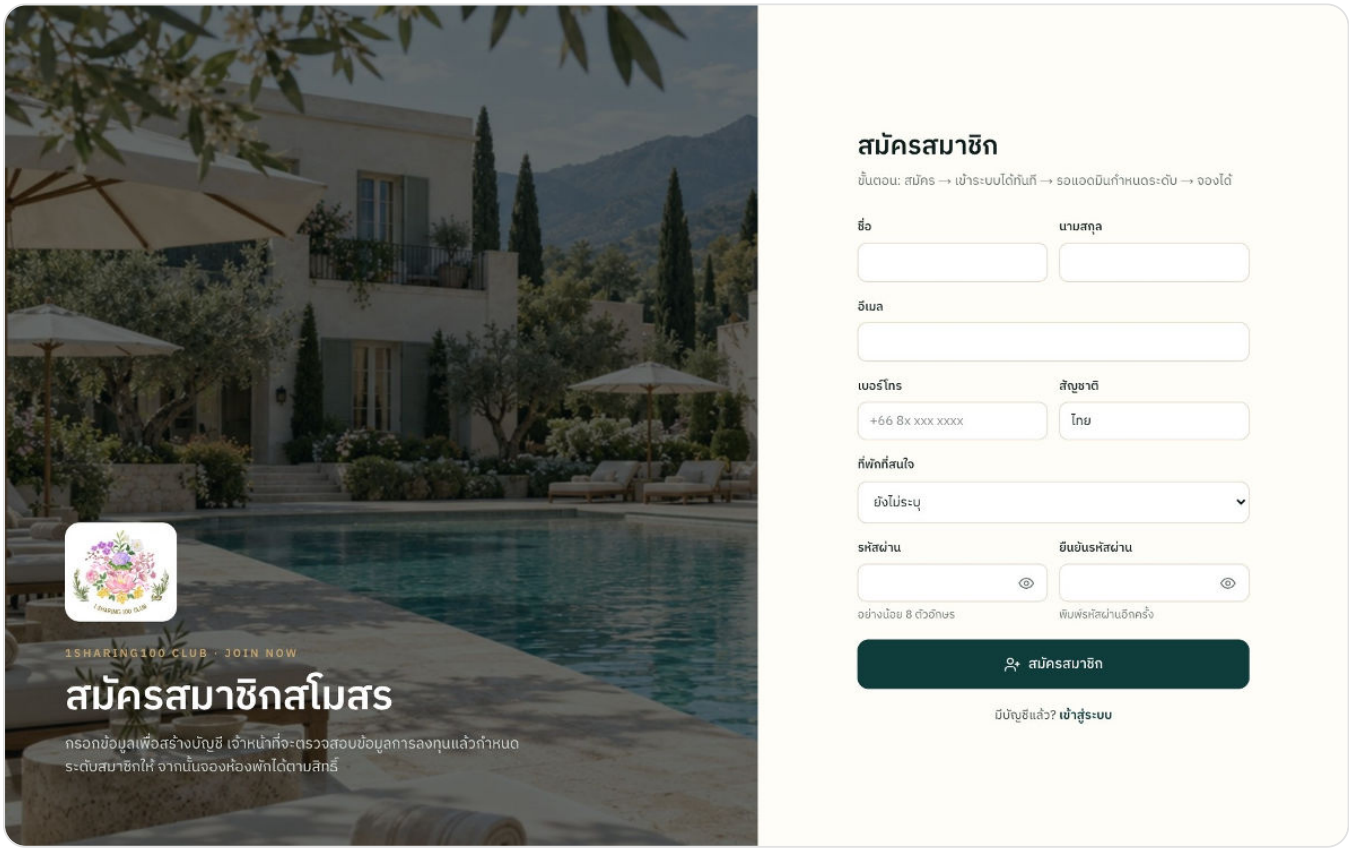
`traveler1`



ภาพจริง สมาชิกเข้า /admin ไม่ได้ → ถูกพากลับหน้า "สิทธิ์ของฉัน" พร้อมข้อความเตือนสีแดงด้านบน

ผลจริง · สมัครสมาชิก

`/register`



ภาพจริง หน้าสมัครสมาชิก · ใช้อีเมลอะไรก็ได้ ไม่ต้องยืนยันอีเมล สมัครแล้วเข้าระบบได้ทันที แต่ขึ้น "รอแอดมินอนุมัติ" ดูห้องและจองไม่ได้จนแอดมินกำหนดระดับ (หน้าสมาชิกใน Phase 2)

ด่าน 1

- แอดมินล็อกอินแล้วเห็นเมนูจัดการ
- ใส่รหัสผิด → ขึ้นข้อความภาษาไทยที่เข้าใจง่าย
- ไม่ล็อกอินแล้วเปิด `/admin` → เด้งไป `/login`
- สมัครด้วยอีเมลใหม่ → เข้าได้ทันทีโดยไม่ต้องยืนยันอีเมล → เห็น "รอแอดมินอนุมัติ" · กดออกจากระบบได้

ทดสอบผ่านแล้ว พิมพ์บอก Claude Code ว่า **"ผ่านแล้ว"** → ตรวจไฟล์ลับ · commit · ติ๊กเฟสใน CLAUDE.md ให้เอง

WORKSHOP 3 · 14.10–15.00

P2 พรอมต์ 5 · หลังบ้านแอดมินทั้งหมด

เป้าหมาย: หลังบ้าน 5 เมนู (ภาพรวม · สมาชิก · นำเข้าสมาชิก · กฎสิทธิ์ · การจอง) · **พรอมต์ทำแค่ระบบ — นักเรียนเป็นคนทดสอบนำเข้า CSV เอง** จนได้สมาชิกครบ 5 ระดับ พร้อมใช้ใน Phase 3



/admin/members · /admin/import

สมาชิก + นำเข้า CSV

อนุมัติคนที่สมัครเอง · ค้นหา กรอง เปลี่ยนระดับ · ตรวจทุกแถวก่อน แล้วสร้างบัญชีฝั่ง server ด้วย secret key

/admin/tiers

กฎสิทธิ์

แก้กฎ ราคา จำนวนห้อง ตั้งค่าการจอง · "ดูแบบระดับ..." เห็นผลทันที

/admin/bookings

การจอง

ประวัติการจองทั้งหมด · ดูสลิป ยืนยัน/ตีกลับ ยกเลิก · ยังว่างจนถึง Phase 3

PROMPT 5

พรอมต์ 5 · Phase 2 · หลังบ้านแอดมินทั้งหมด

```
อ่าน CLAUDE.md แล้วทำ Phase 2: หลังบ้านแอดมินทั้งหมด (ห้ามแก้ระบบ Login ที่ผ่านแล้ว)
ทำแค่ระบบ ยังไม่ต้องนำเข้าไฟล์ CSV เอง ฉันจะทดสอบนำเข้าเองหลังทำเสร็จ

1. ภาพรวม /admin: จำนวนสมาชิกแต่ละระดับ · สมาชิกรออนุมัติ · การจองที่รอตรวจสลิป
2. สมาชิก /admin/members: สมาชิกที่สมัครเองและรออนุมัติขึ้นด้านบน พร้อมปุ่มเลือกระดับเพื่ออนุมัติ · ตาราง ชื่อ · อีเมล · ระดับ (ป้ายสี) · ที่พักที่สนใจ · วันที่ได้ระดับ · ค้นหา · กรองตามระดับ · เปลี่ยนระดับจาก dropdown (เกินจำนวนครั้งต่อปีให้ขึ้นเหตุผล)
3. นำเข้าสมาชิก /admin/import:
   - อัปโหลดไฟล์ CSV (UTF-8) คอลัมน์ตาม data/members-demo.csv แล้วอ่านด้วย papaparse
   - ตรวจทุกแถวก่อนบันทึก: อีเมลถูกรูปแบบ · ไม่ซ้ำในไฟล์ · ไม่ซ้ำกับบัญชีเดิม · ระดับมีจริง · ที่พักมีจริง (เว้นว่างได้)
   - ตารางตัวอย่าง: แถวผ่าน (เขียว) · แถวไม่ผ่านพร้อมเหตุผลภาษาไทย (แดง) · ปุ่ม "นำเข้า N แถวที่ผ่าน"
   - สร้างบัญชีที่ app/api/admin/import/route.ts ฝั่ง server ด้วย SUPABASE_SECRET_KEY: ตรวจก่อนว่าผู้เรียกเป็นแอดมิน (ไม่ใช่ตอบ 403) · รหัสผ่าน = IMPORT_DEFAULT_PASSWORD · ตรวจซ้ำอีกรอบฝั่ง server · ส่งผลกลับรายแถว
4. กฎสิทธิ์ /admin/tiers (แบ่งแท็บ): กฎแต่ละระดับ · ราคาต่อคืน · จำนวนห้องแต่ละที่พัก · ตั้งค่าการจอง (booking_settings) · ตัวเลือก "ดูแบบระดับ…" แสดงผลจาก room_options('<ระดับ>') ทันทีหลังแก้
5. การจอง /admin/bookings: ตารางการจองและประวัติทั้งหมด กรองตามสถานะ ค้นหา · ดูรูปสลิปผ่านลิงก์ชั่วคราว (กดดูเต็มจอ) · ยืนยัน/ตีกลับการชำระ (ใส่เหตุผล) · ยกเลิกการจอง — ทุกปุ่มเรียกฟังก์ชันในฐานข้อมูล (admin_confirm_payment · admin_reject_payment · cancel_booking) · ถ้ายังไม่มีการจองให้แสดง "ยังไม่มีการจอง"
6. ห้ามเขียนชื่อระดับ ตัวเลข หรือราคาลงในโค้ด · ดีไซน์ตามรูป mockups/mock-admin-members.jpg · mock-admin-import.jpg · mock-admin-tiers.jpg

เสร็จแล้วบอกวิธีทดสอบ (ไฟล์ดี data/members-demo.csv และไฟล์มีข้อผิดพลาด data/members-with-errors.csv) แล้วหยุดรอฉันทดสอบ
```

บ่าย · พรอมต์ 5 · Phase 2 · รันเพื่อ

สร้างหลังบ้านแอดมินให้ครบ — ส่วนการนำเข้า CSV นักเรียนเป็นคนทดสอบเอง

Claude Code จะทำให้

1. **หน้าสมาชิก**
   อนุมัติคนที่สมัครเอง · ค้นหา · เปลี่ยนระดับ
2. **หน้านำเข้า CSV**
   ทำระบบตรวจทุกแถว แล้วสร้างบัญชีให้
3. **หน้ากฎสิทธิ์**
   แก้ราคา กฎ จำนวนห้อง
4. **หน้าการจอง**
   ดูประวัติ ดูสลิป ยืนยัน/ตีกลับ ยกเลิก

ได้อะไรกลับมา

- ✓ **หลังบ้าน 5 เมนู**
- ✓ **นักเรียนนำเข้าไฟล์ทดสอบเอง → สมาชิก 10 คน ครบ 5 ระดับ**

เช็คว่าผ่าน ไฟล์ดี: นำเข้าได้ 10/10 · ไฟล์ผิด: ผ่าน 1 แถว อีก 3 แถวขึ้นเหตุผล · อนุมัติสมาชิกที่สมัครเองได้ · สมาชิกเปิด /admin ไม่ได้

คอลัมน์ในไฟล์ CSV

คอลัมน์	ความหมาย · กฎตรวจ	ตัวอย่าง
`email`	อีเมลถูกรูปแบบ · ไม่ซ้ำในไฟล์ · ไม่ซ้ำกับบัญชีที่มีอยู่	`traveler1@club.test`
`first_name`	ชื่อ	ณัฐชา
`last_name`	นามสกุล	บุญมา
`phone`	เบอร์โทร (จำลอง)	`+66 81 000 2001`
`nationality`	สัญชาติ	ไทย
`tier`	ต้องมีอยู่ในตาราง `tiers`	`TRAVELER`
`preferred_branch`	ที่พักที่สนใจ ต้องมีในตาราง `branches` (เว้นว่างได้)	`buengkan`

รหัสที่พัก: `chiangmai` เชียงใหม่ · `buengkan` บึงกาฬ · `huahin` หัวหิน · `nongkhai` หนองคาย · `nongkhai-care` หนองคาย · ดูแลสุขภาพ · `phutthamonthon` พุทธมณฑล · ดูแลสุขภาพ — ไฟล์ต้องบันทึกแบบ UTF-8 (Excel → "CSV UTF-8")

ทดสอบด้วยไฟล์ 2 ไฟล์ — นำเข้าตามลำดับนี้

ไฟล์ที่ 1 · ไฟล์ดี

`data/members-demo.csv`

สมาชิกสมมติ 10 คน ระดับละ 2 คน (spender1/2 · traveler1/2 · explorer1/2 · supporter1/2 · giver1/2) → **ผ่าน 10/10**

ไฟล์ที่ 2 · มีข้อผิดพลาด

`data/members-with-errors.csv`

4 แถว → **ผ่าน 1 แถว** อีก 3 แถวขึ้นเหตุผล (ต้องนำเข้าไฟล์ที่ 1 ก่อน แถว traveler1 จึงซ้ำ)

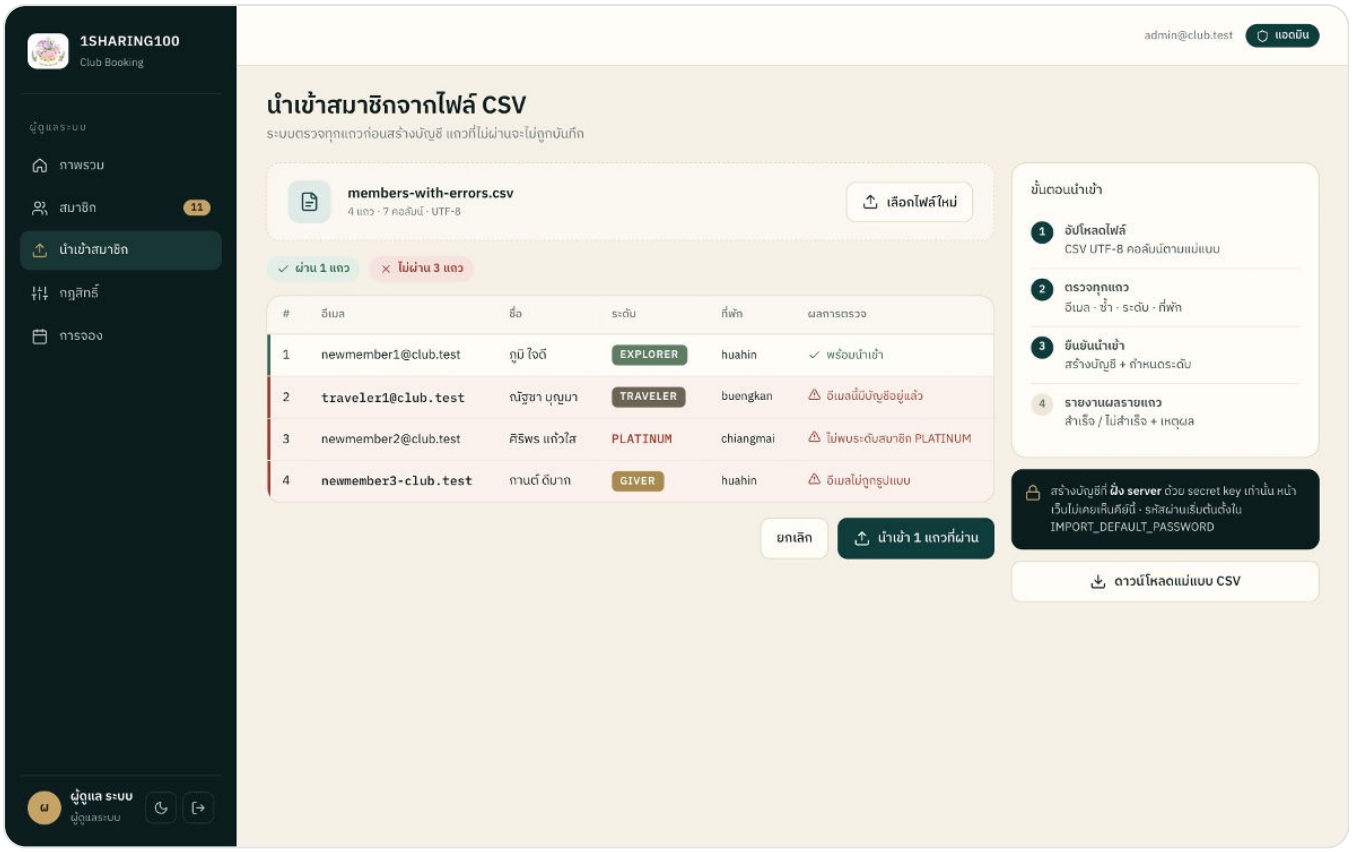
ผลที่ต้องเห็นเมื่อนำเข้าไฟล์ที่ 2

#	อีเมล	tier	ผล	เหตุผล
1	`newmember1@club.test`	EXPLORER	ผ่าน	—
2	`traveler1@club.test`	TRAVELER	ไม่ผ่าน	อีเมลนี้มีบัญชีแล้ว (นำเข้าจากไฟล์ที่ 1)
3	`newmember2@club.test`	PLATINUM	ไม่ผ่าน	ไม่มีระดับ PLATINUM ในระบบ
4	`newmember3-club.test`	GIVER	ไม่ผ่าน	อีเมลไม่ถูกรูปแบบ (ไม่มี @)

ถ้อยคำเหตุผลบนจอของแต่ละคนอาจต่างจากนี้เล็กน้อย ขอให้ผลผ่าน/ไม่ผ่านตรงกัน · สมาชิกที่นำเข้าทุกคนใช้รหัสผ่าน `Member@2569` (ค่า `IMPORT_DEFAULT_PASSWORD` · ใช้ในคลาสเท่านั้น)

ผลจริง · นำเข้าสมาชิก

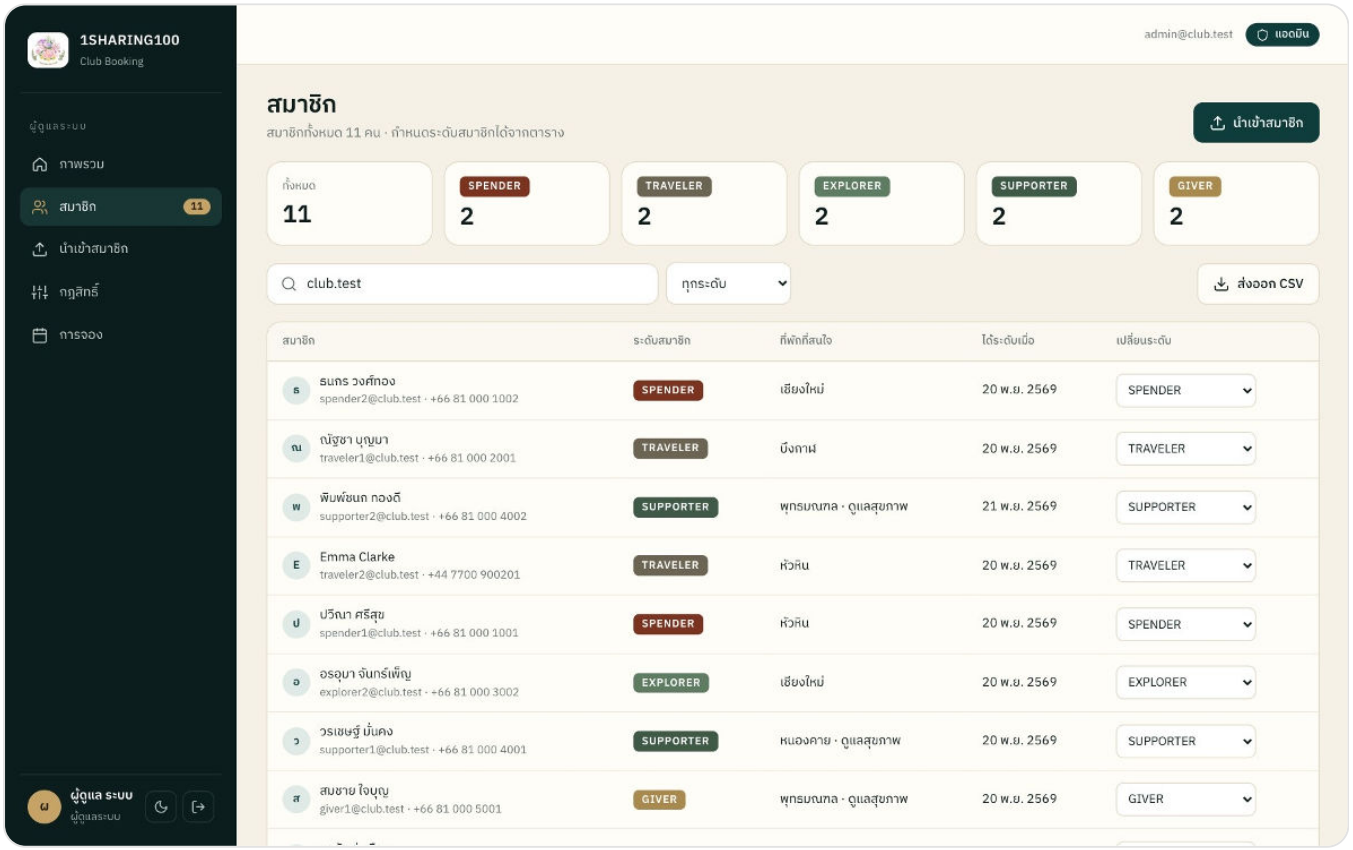
`/admin/import · Phase 2`



ภาพจริง ไฟล์ members-with-errors.csv · แถวผ่าน (เขียว) และไม่ผ่านพร้อมเหตุผล (แดง) · ปุ่ม "นำเข้า 1 แถวที่ผ่าน" · ยังไม่สร้างบัญชีจนกว่าจะกดยืนยัน

ผลจริง · สมาชิก

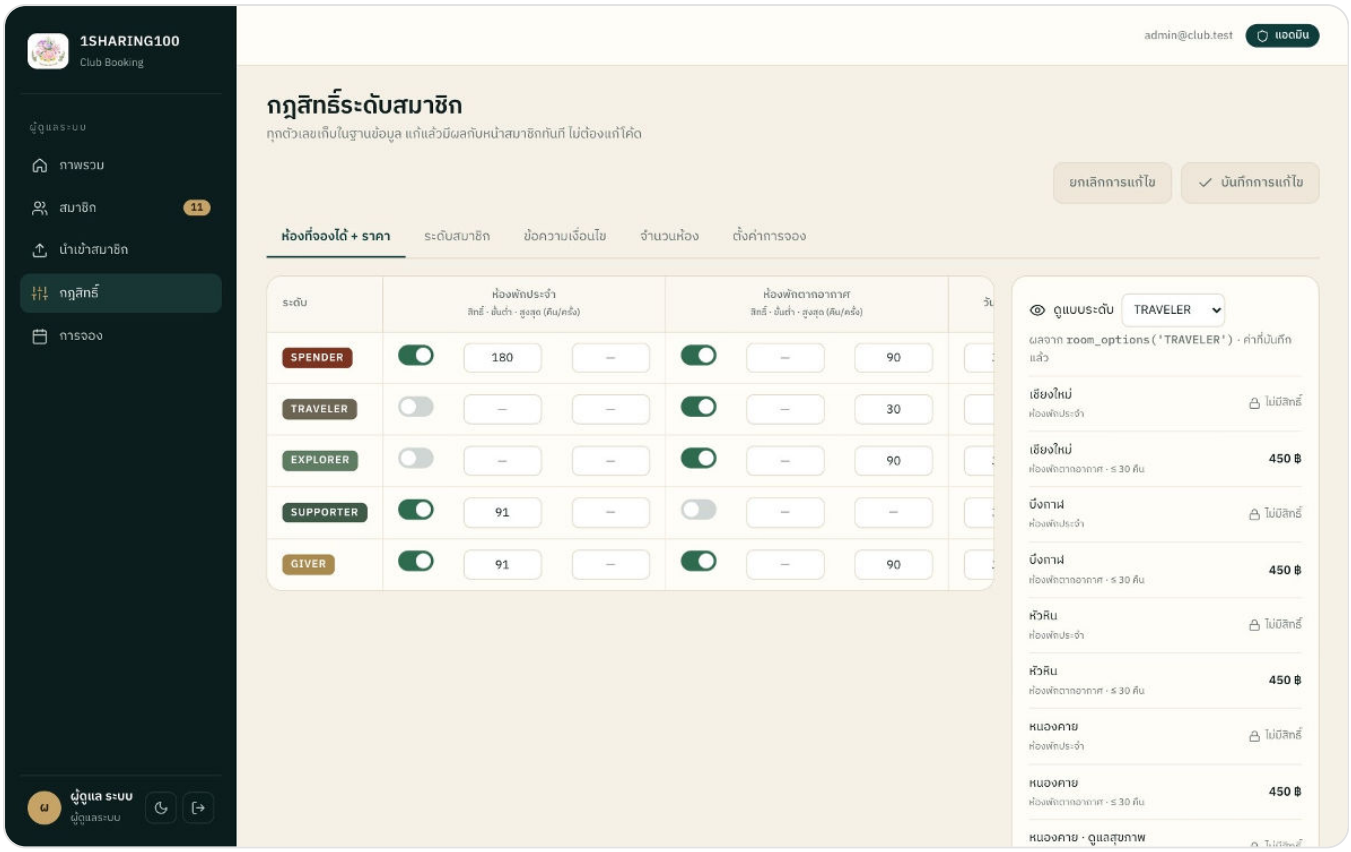
`/admin/members · Phase 2`



ภาพจริง ตัวเลขสรุปจำนวนสมาชิกแต่ละระดับ · ตารางชื่อ อีเมล ระดับ ที่พักที่สนใจ วันที่ได้ระดับ · เปลี่ยนระดับจาก dropdown · ข้อมูลจำลอง

ผลจริง · กฎสิทธิ์

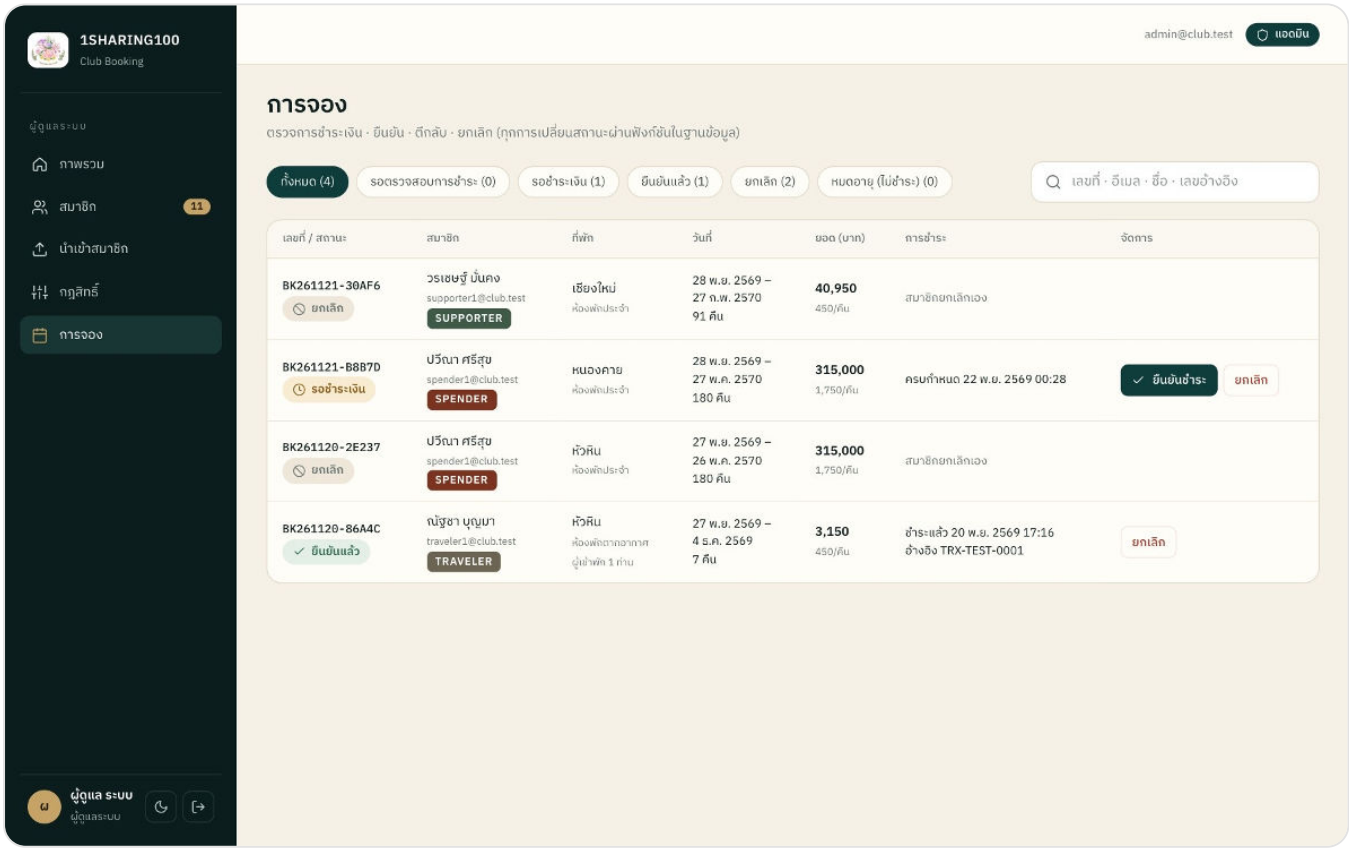
`/admin/tiers`



ภาพจริง แอดมินแก้กฎ ราคา และจำนวนห้องได้เอง · แผงด้านขวาแสดงผล "ดูแบบระดับ..." ทันที

ผลจริง · การจอง

`/admin/bookings`



ภาพจริง กรองตามสถานะ · ยืนยันชำระ · ยกเลิก · ภาพนี้มีการจองแล้ว (หลัง Phase 3) · ตอนจบ Phase 2 จะขึ้น "ยังไม่มีการจอง"

ด่าน 2

- นำเข้า `members-demo.csv` ได้ 10/10
- นำเข้า `members-with-errors.csv` ได้ 1 แถว อีก 3 แถวขึ้นเหตุผล
- เปลี่ยนระดับได้ · แก้ราคา/กฎได้ · หน้าการจองเปิดได้
- ล็อกอิน traveler1@club.test ด้วย `Member@2569` ได้ แต่เปิด `/admin` ไม่ได้

ทดสอบผ่านแล้ว พิมพ์บอก Claude Code ว่า **"ผ่านแล้ว"** → ตรวจไฟล์ลับ · commit · ติ๊กเฟสใน CLAUDE.md ให้เอง

BREAK

พักเบรก 15.00–15.15 น.

WORKSHOP 4 · 15.15–16.30

P3 พรอมต์ 6 · สิทธิ์ตามระดับ + จองห้อง + ชำระเงิน

เป้าหมาย: สมาชิก 5 ระดับเห็นห้องไม่เหมือนกัน · จองห้องได้จริง · แนบรูปสลิปแจ้งชำระ · แอดมินยืนยันแล้วขึ้นในประวัติทั้งสองฝั่ง · **ยังไม่ชำระ = จองใหม่ไม่ได้ · ห้ามจองซ้อนวัน** — ระบบครบในวันเดียว



จอง 1 ครั้ง ผ่าน 4 สถานะ

01	02	03	04
เลือกวัน → ระบบตรวจสิทธิ์ทุกข้อ	**ยืนยัน = "รอชำระ"**	**แนบรูปสลิป = "รอตรวจ"**	**แอดมินยืนยัน = "ยืนยันแล้ว"**

แอดมินตีกลับ = กลับไป "รอชำระ" พร้อมเหตุผล · ไม่ชำระภายในเวลาที่ตั้งไว้ (ค่าเริ่มต้น 24 ชั่วโมง) = หมดอายุ คืนห้อง · ยกเลิกเองได้ก่อนวันเช็คอิน

กฎการจอง 9 ข้อ — ฐานข้อมูลตรวจให้

#	จองไม่ได้ถ้า...
1	ยังไม่มีระดับ (รอแอดมินอนุมัติ)
2	ระดับนี้ไม่มีสิทธิ์ห้องประเภทนี้
3	วันที่ผิด · จองล่วงหน้าเกินที่ระดับกำหนด · สิทธิ์หมดอายุ
4	จำนวนคืนไม่ตรงเงื่อนไข · ผู้เข้าพักเกิน 5 ท่าน
5	**ยังมีการจองที่รอชำระหรือรอตรวจสลิป — ต้องจ่ายครั้งก่อนให้เสร็จก่อน**
6	**ช่วงวันซ้อนกับการจองเดิมของตัวเอง**
7	SPENDER ใช้ห้องประจำแล้ว จะจองตากอากาศในปีเดียวกัน (หรือกลับกัน)
8	วันพักรวมทั้งปีเกินโควตาของระดับ
9	ห้องเต็มในช่วงวันที่เลือก

PROMPT 6

พรอมต์ 6 · Phase 3 · สิทธิ์ตามระดับ + จองห้อง + ชำระเงิน

อ่าน CLAUDE.md แล้วทำ Phase 3: สิทธิ์ตามระดับ + จองห้อง + แจ้งชำระ + ประวัติ (ห้ามแก้ Phase 1–2 ที่ผ่านแล้ว)

1. สิทธิ์ของฉัน /app: การ์ดสิทธิ์ · ระดับ · วันพักต่อปีและที่ใช้ไปแล้ว (my_quota()) · จองล่วงหน้าได้กี่เดือน · อายุสิทธิ์ (ว่าง = ตลอดชีพ) · ประเภทห้องที่ได้ — อ่านจากตาราง tiers และ tier_room_rules ของระดับตัวเอง · การจองล่าสุด · ถ้ายังไม่มีระดับ แสดง "บัญชีนี้รอแอดมินอนุมัติ"
2. ห้องพัก /app/rooms: เรียก my_room_options() เท่านั้น แล้วแสดงการ์ดแยกตามที่พัก
   - จองได้: ราคา/คืน · เงื่อนไขจำนวนคืน · ปุ่ม "จอง" ไปหน้าจอง
   - ไม่มีสิทธิ์: ไอคอนกุญแจ + เหตุผลที่ได้จากฟังก์ชัน
   - กรองตามจังหวัด / ประเภทห้อง / เฉพาะที่จองได้
3. จองห้อง /app/book: เลือกวันเช็คอิน–เช็คเอาต์ · ชื่อผู้เข้าพัก · หมายเหตุ
   - ทุกครั้งที่เปลี่ยนค่า เรียก quote_booking() แล้วแสดง ราคา/คืน · จำนวนคืน · ห้องว่าง · โควตาที่เหลือ · ยอดชำระ 100% · ถ้าจองไม่ได้ให้แสดงเหตุผลจากฐานข้อมูล และปิดปุ่ม
   - ปุ่ม "ยืนยันการจอง" เรียก create_booking() แล้วพาไปหน้าการจองของฉัน
   - ห้ามตรวจกฎการจองเองในหน้าเว็บ — กฎทั้ง 9 ข้อในหัวข้อ 5 ของ CLAUDE.md ฐานข้อมูลตรวจให้แล้ว (ค้างชำระจองใหม่ไม่ได้ · ห้ามจองซ้อนวัน · โควตา · ห้องเต็ม)
4. การจองของฉัน /app/bookings: แท็บ กำลังดำเนินการ / ประวัติ · การ์ดแต่ละการจองแสดงสถานะ (รอชำระ · รอตรวจ · ยืนยันแล้ว · ยกเลิก · หมดอายุ) และกำหนดชำระ
   - รอชำระ: แนบรูปสลิป (ย่อรูปก่อนอัปโหลด · เก็บใน Storage payment-slips โฟลเดอร์ <id ผู้ใช้>/<id การจอง>/) + เลขอ้างอิง (ไม่บังคับ) แล้วเรียก submit_payment() · ถ้าแอดมินตีกลับ แสดงเหตุผลและให้แนบใหม่
   - ยกเลิกได้ก่อนวันเช็คอิน (cancel_booking) · การจองที่ยืนยันแล้วกด "ต่อการจอง" ได้
   - กดดูรูปสลิปเต็มจอได้
5. ตรวจว่าหน้า "การจอง" ของแอดมิน (จาก Phase 2) เห็นการจองใหม่และรูปสลิปทันที ถ้ายังไม่เห็นให้แก้
6. ห้ามเขียนตัวเลขหรือชื่อระดับในโค้ด · ดีไซน์ตามรูป mockups/mock-member-home.jpg · mockups/mock-member-rooms.jpg

เสร็จแล้วบอกวิธีทดสอบตามหัวข้อ 9 ของ CLAUDE.md ครบทุกข้อ (ดูห้อง 5 ระดับ + จองห้อง 6 ขั้น) แล้วหยุดรอฉันทดสอบ

บ่าย · พรอมต์ 6 · Phase 3 · รันเพื่อ

สมาชิกเห็นห้องตามระดับ จองได้จริง แจ้งชำระด้วยสลิป และแอดมินเห็นประวัติ — ครบในวันเดียว

Claude Code จะทำให้

1. **การ์ดสิทธิ์ + หน้าห้องพัก**
   จองได้ = ราคา + ปุ่มจอง · ไม่มีสิทธิ์ = ล็อก + เหตุผล
2. **หน้าจองห้อง**
   เลือกวัน → ระบบตรวจสิทธิ์ทุกข้อ → ยืนยันการจอง
3. **หน้าการจองของฉัน**
   แนบรูปสลิป · ยกเลิก · ดูประวัติ · ต่อการจอง
4. **ฐานข้อมูลตรวจกฎให้**
   ค้างชำระจองใหม่ไม่ได้ · ห้ามจองซ้อนวัน · โควตา · ห้องเต็ม

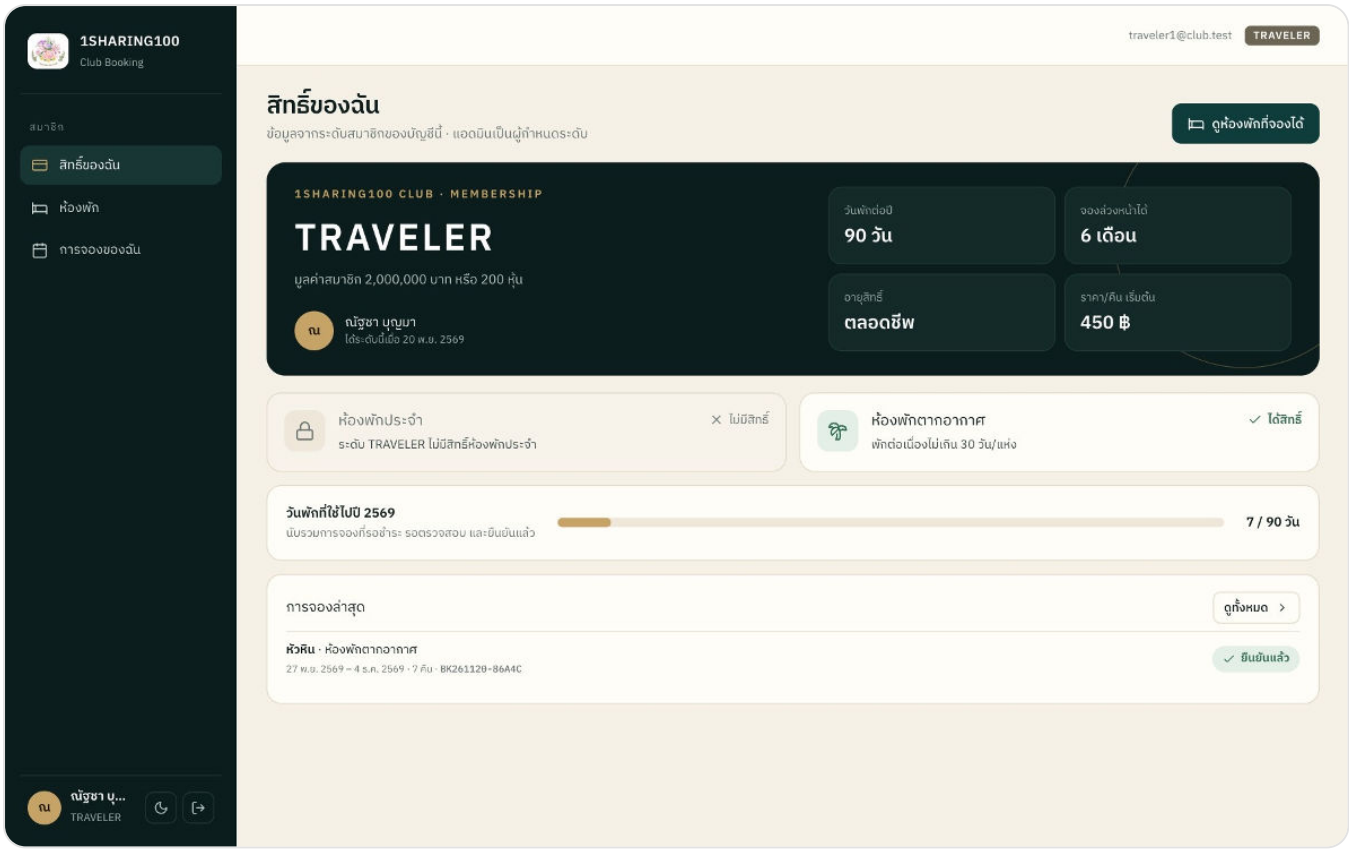
ได้อะไรกลับมา

- ✓ **5 ระดับเห็นห้องคนละแบบ และจองได้จริง**
- ✓ **แอดมินเห็นการจองและรูปสลิปทันที**

เช็คว่าผ่าน ผลดูห้องตรงตารางทดสอบครบ 5 ระดับ | จองแล้วสถานะ "รอชำระ" | ยังไม่ชำระ → จองใหม่ไม่ได้ | แนบสลิป → แอดมินยืนยัน → จองครั้งถัดไปได้ | จองซ้อนวันไม่ได้

ผลจริง · สิทธิ์ของฉัน (TRAVELER)

`/app · traveler1`

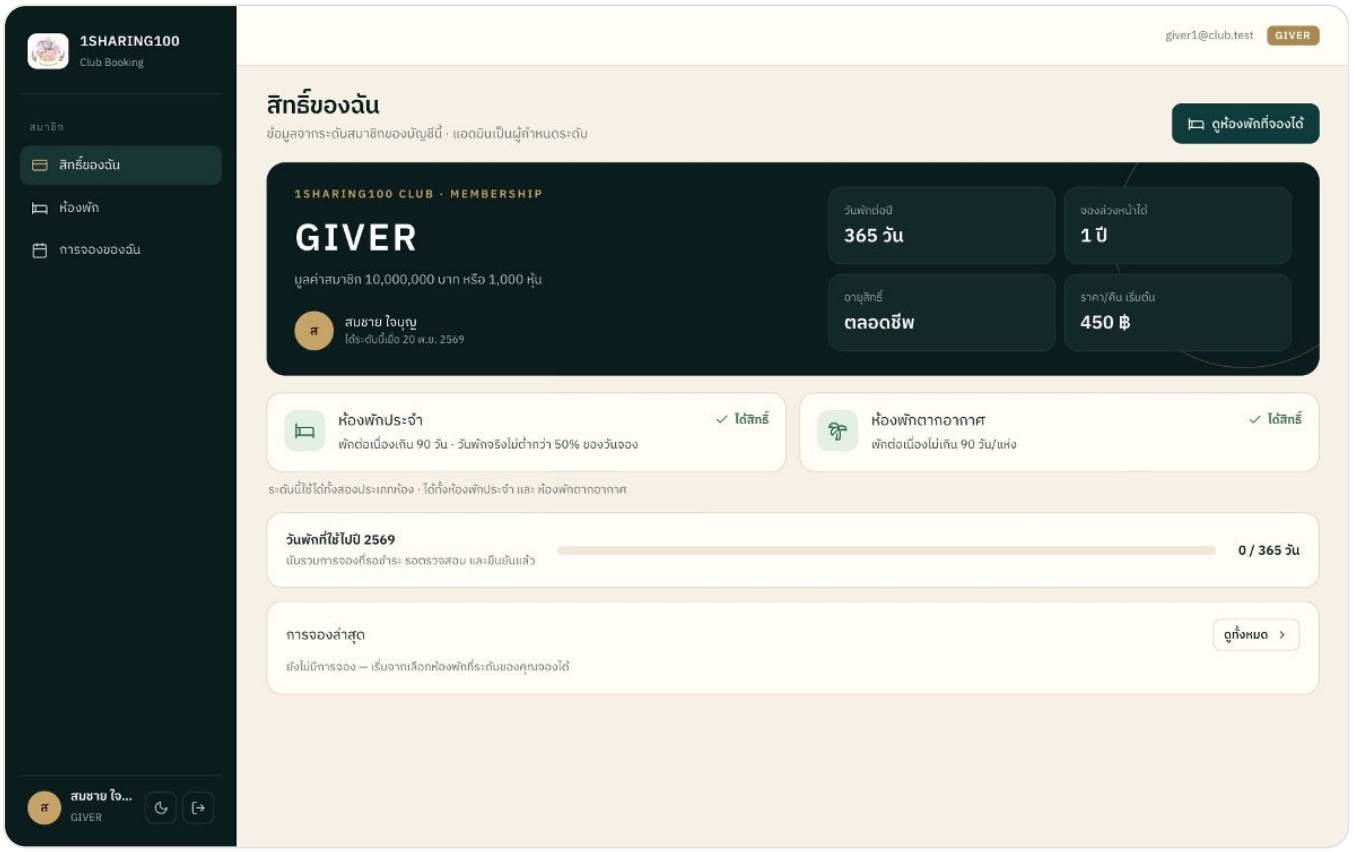


ภาพจริง การ์ด TRAVELER · 90 วัน/ปี · จองล่วงหน้า 6 เดือน · ตลอดชีพ · ห้องพักประจำ "ไม่มีสิทธิ์" · ห้องพักตากอากาศ "ได้สิทธิ์" · แถบวันพักที่ใช้ไปและการจองล่าสุด

> ภาพจริงจากระบบเดโมที่สร้างด้วยพรอมต์ชุดนี้ · ข้อมูลจำลอง · หน้าตาของแต่ละคนอาจต่างกันเล็กน้อย

ผลจริง · สิทธิ์ของฉัน (GIVER)

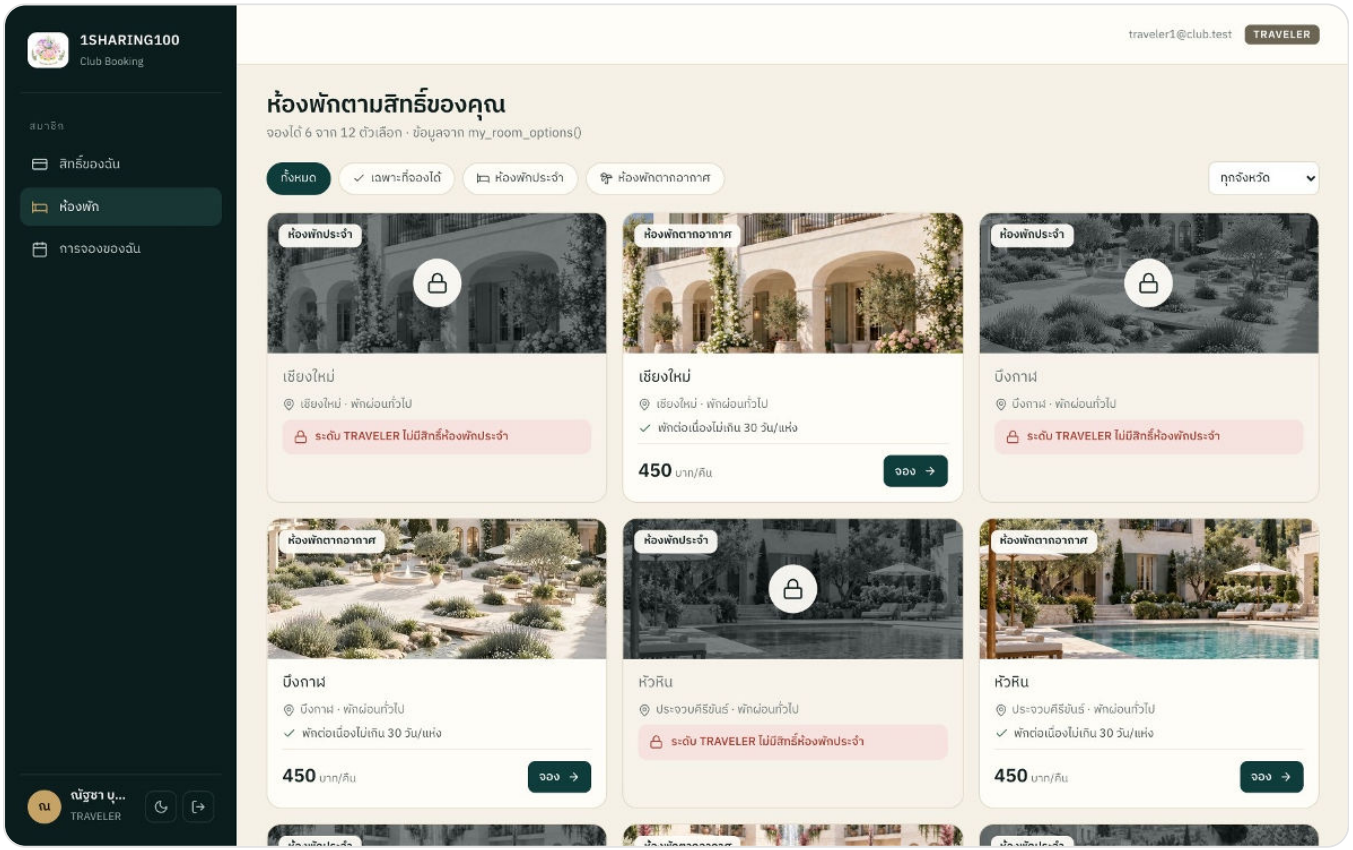
`/app · giver1`



ภาพจริง การ์ด GIVER · 365 วัน/ปี · จองล่วงหน้า 1 ปี · ได้ทั้งห้องพักประจำและห้องพักตากอากาศ

ผลจริง · ห้องพัก (TRAVELER)

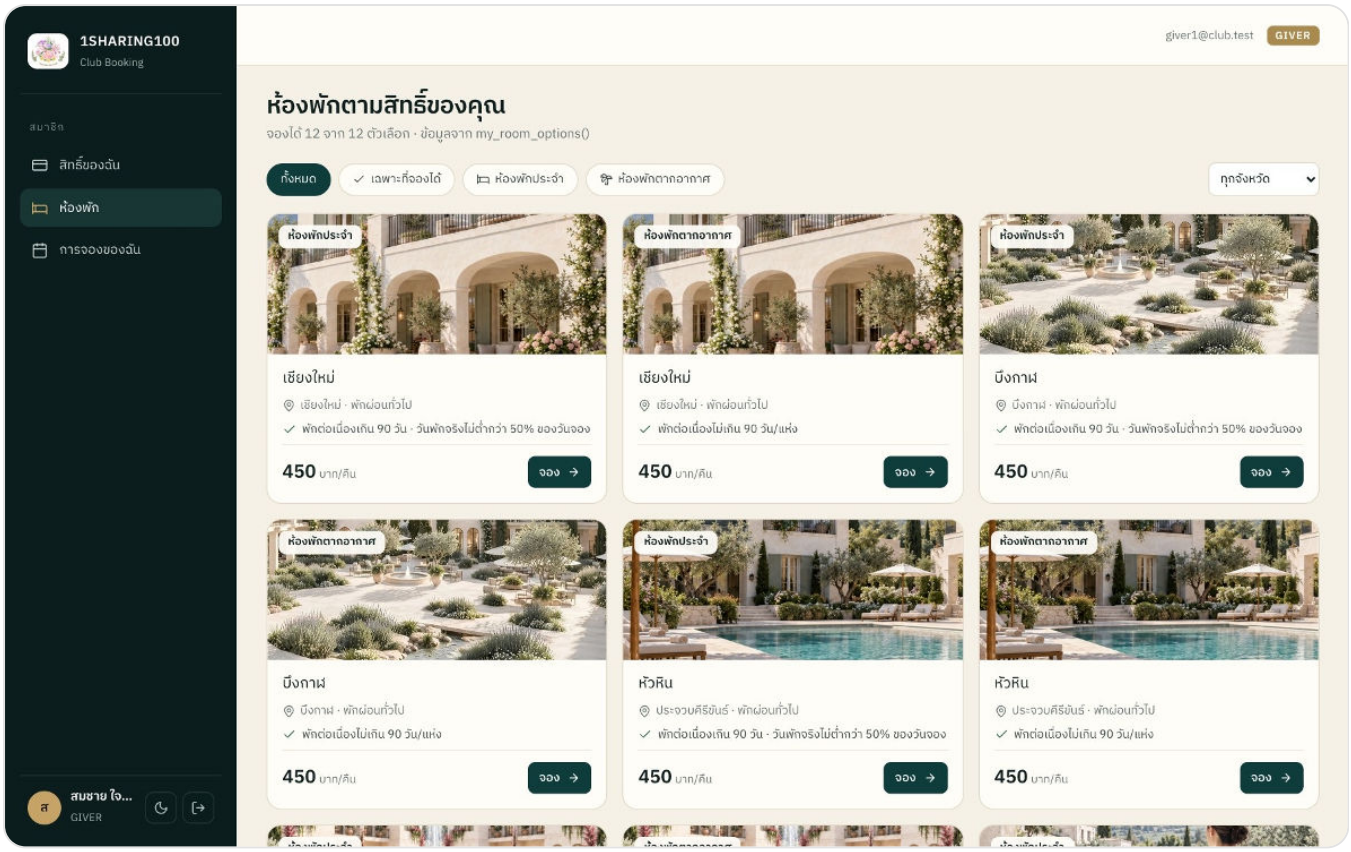
`/app/rooms · traveler1`



ภาพจริง ห้องพักตากอากาศ 450 บาท/คืน ≤ 30 วัน/แห่ง · ห้องพักประจำล็อก "ระดับ TRAVELER ไม่มีสิทธิ์ห้องพักประจำ" · ปุ่ม "จอง" พาไปหน้าจองห้อง

ผลจริง · ห้องพัก (GIVER)

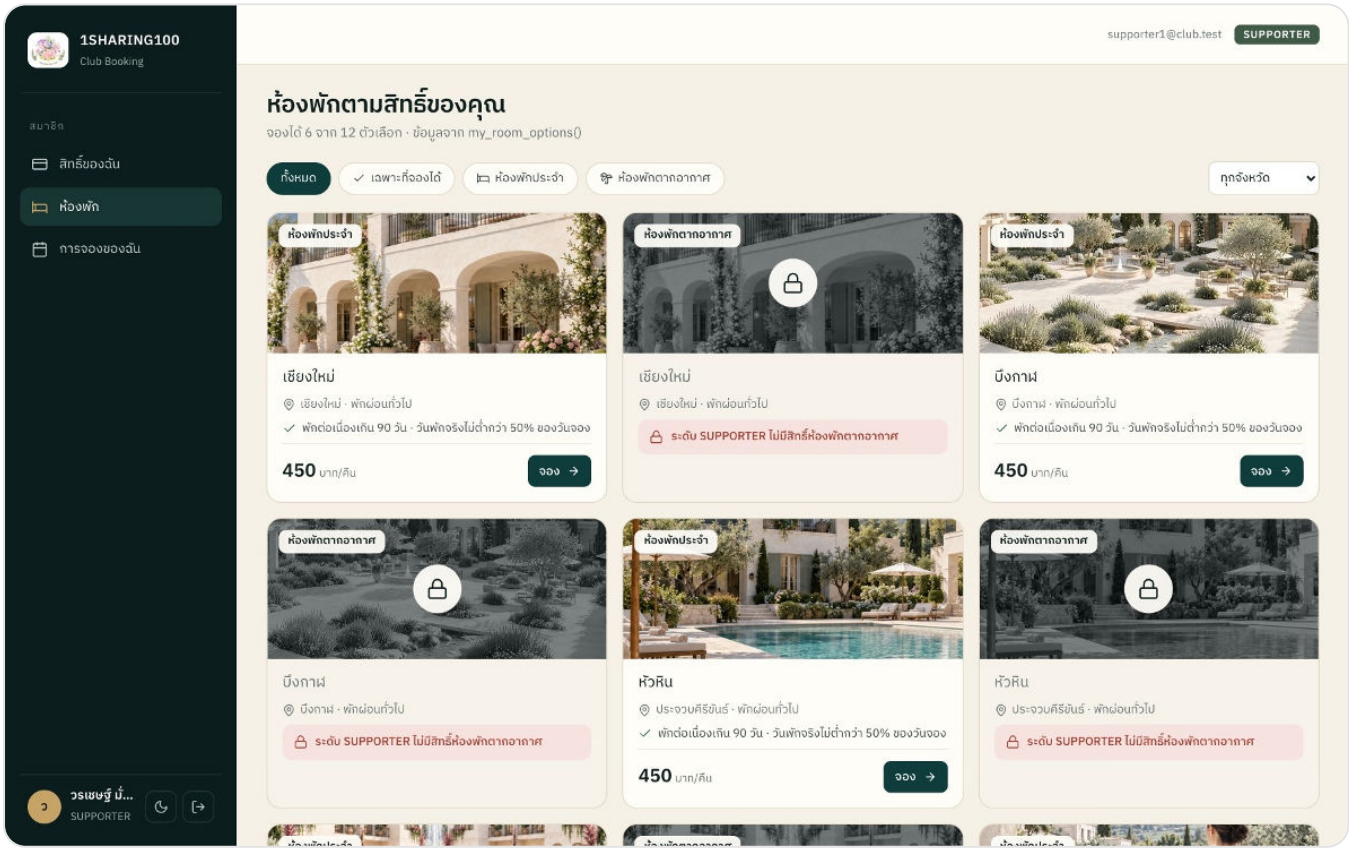
`/app/rooms · giver1`



ภาพจริง จองได้ทั้ง 12 ตัวเลือก (6 ที่พัก × 2 ประเภทห้อง) พร้อมราคาและเงื่อนไขจำนวนคืน

ผลจริง · ห้องพัก (SUPPORTER)

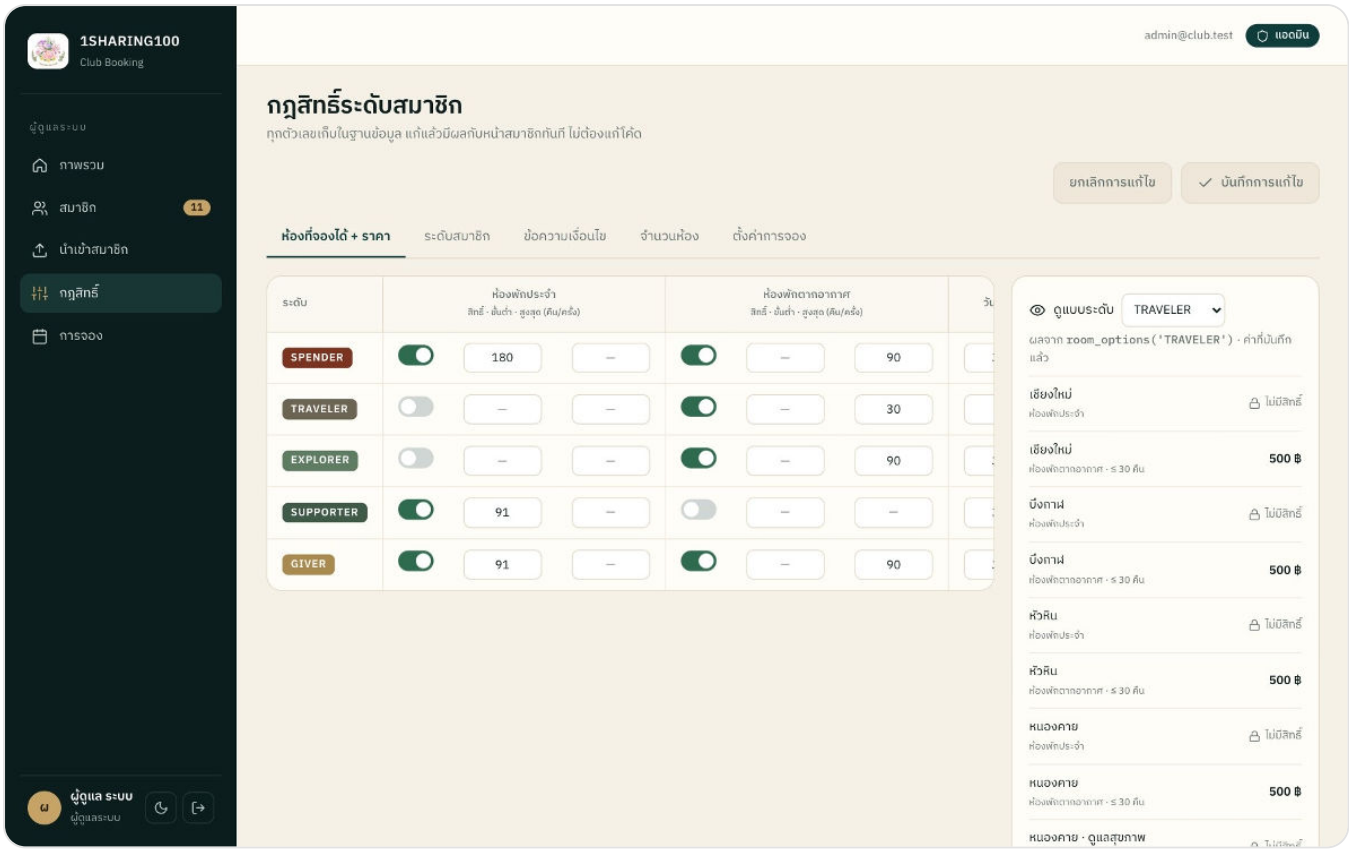
`/app/rooms · supporter1`



ภาพจริง กลับกับ TRAVELER: ห้องพักประจำจองได้ (เกิน 90 วัน · พักจริง ≥ 50%) · ห้องพักตากอากาศล็อก

ผลจริง · กฎสิทธิ์ (แอดมิน)

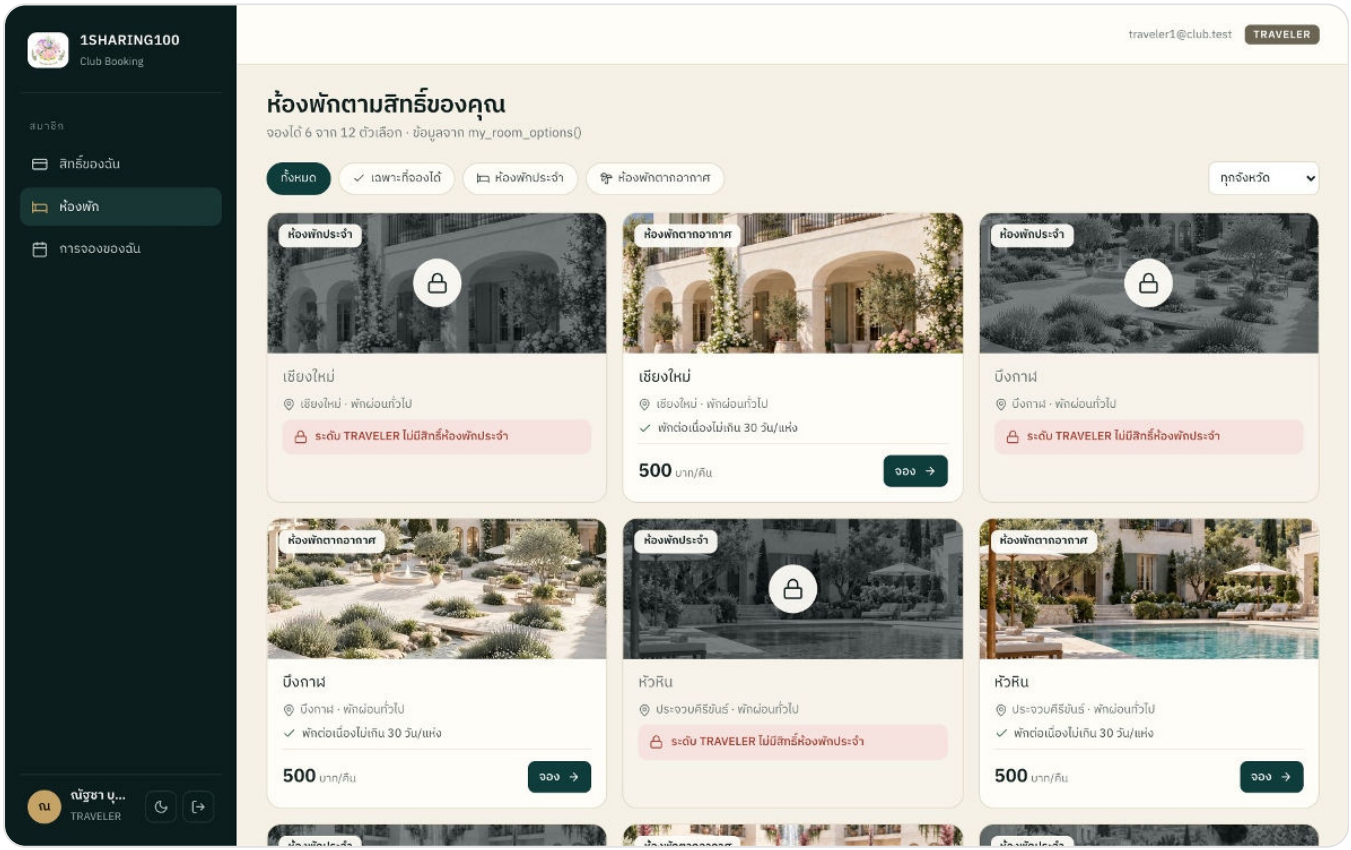
`/admin/tiers · admin`



ภาพจริง แท็บห้องที่จองได้ + ราคา · แก้ราคา TRAVELER ที่พักทั่วไปเป็น 500 แล้วแผง "ดูแบบระดับ TRAVELER" แสดงผลทันที

ผลจริง · traveler1 รีเฟรช

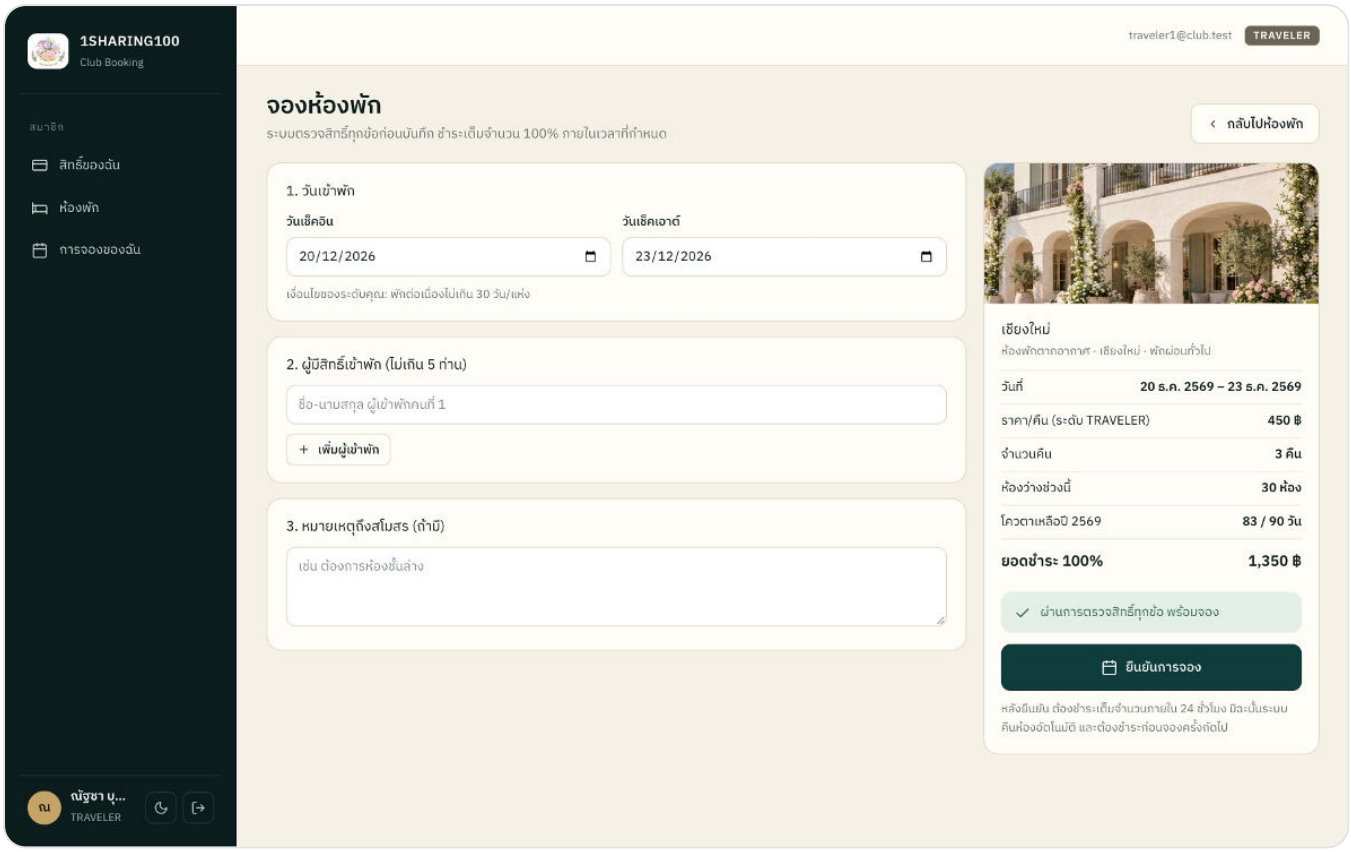
`/app/rooms · traveler1`



ภาพจริง traveler1 รีเฟรชแล้วเห็น 500 บาท/คืน โดยไม่ต้องแก้โค้ด · ทดลองแล้วตั้งราคากลับเป็น 450 ตามตารางทดสอบ

ผลจริง · จองห้อง

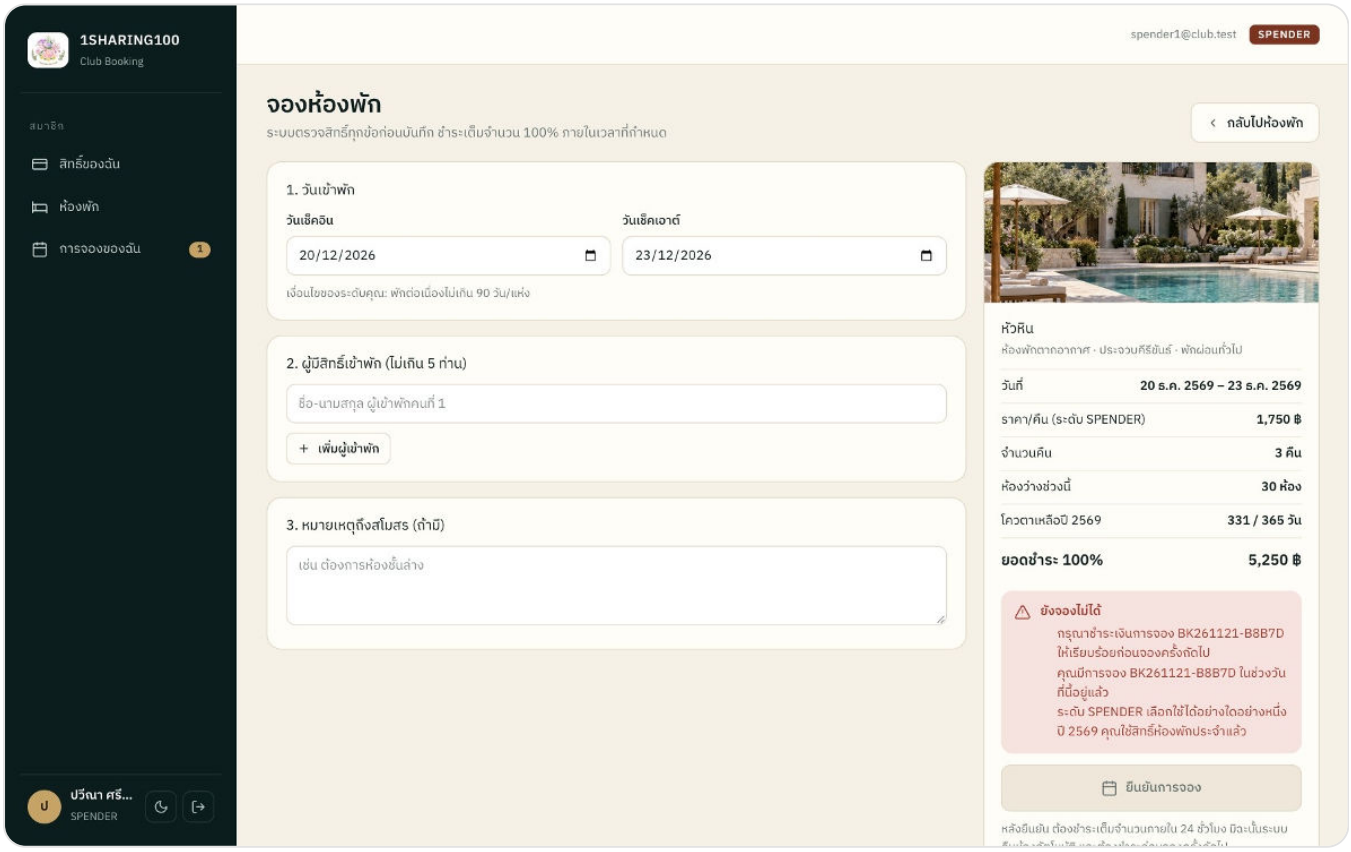
`/app/book · traveler1`



ภาพจริง เลือกวัน → ระบบบอกราคา/คืน จำนวนคืน ห้องว่าง โควตาที่เหลือ และยอดชำระ 100% · ผ่านทุกข้อ → ปุ่ม "ยืนยันการจอง" กดได้

ผลจริง · จองไม่ได้ พร้อมเหตุผล

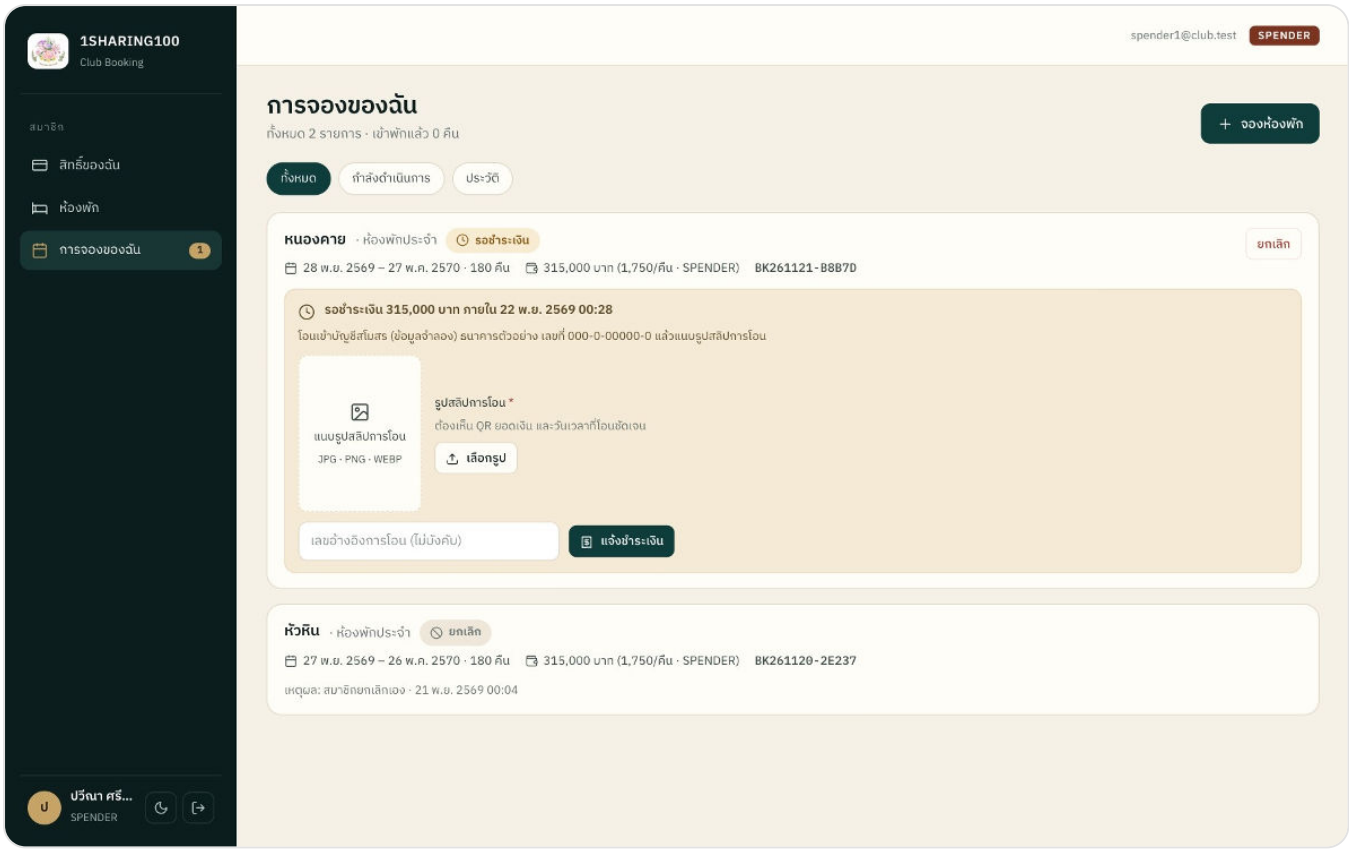
`/app/book · spender1`



ภาพจริง spender1 ยังไม่ชำระการจองเดิม → "กรุณาชำระเงินการจอง ... ให้เรียบร้อยก่อนจองครั้งถัดไป" · วันซ้อนกับการจองเดิม · SPENDER ใช้ห้องประจำไปแล้ว → ปุ่มปิด

ผลจริง · แนบรูปสลิป

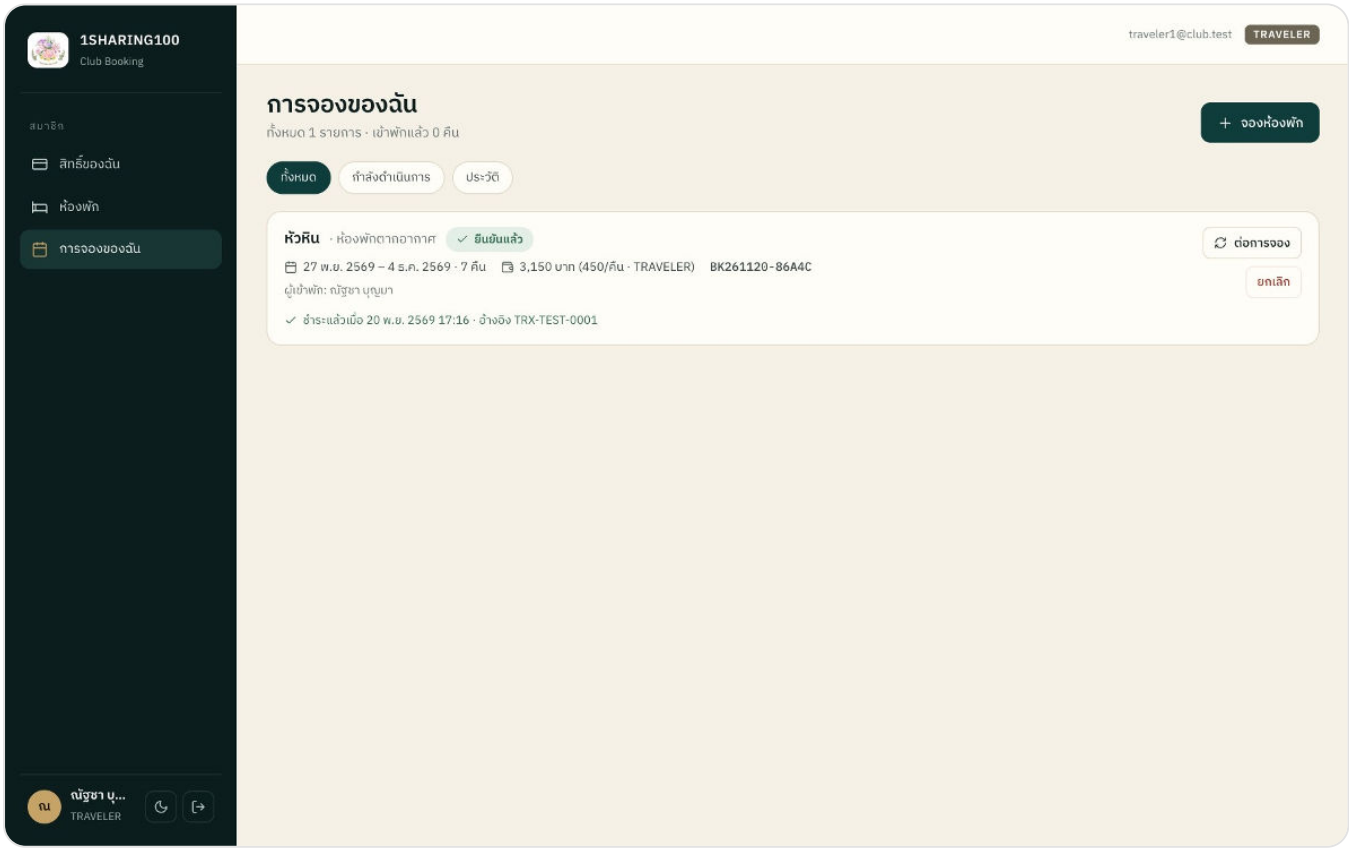
`/app/bookings · รอชำระ`



ภาพจริง สถานะ "รอชำระ" · แนบรูปสลิปการโอน + เลขอ้างอิง (ไม่บังคับ) → "แจ้งชำระเงิน" → สถานะเปลี่ยนเป็น "รอตรวจ" แอดมินเห็นทันที

ผลจริง · ยืนยันแล้ว + ประวัติ

`/app/bookings · traveler1`



ภาพจริง แอดมินยืนยันแล้ว → "ยืนยันแล้ว" ขึ้นในประวัติ · กด "ต่อการจอง" ได้ · จองครั้งถัดไปได้แล้ว

ตรวจด้วยตารางทดสอบ — ดูห้อง 5 ระดับ + จองห้อง 6 ขั้น

ล็อกอินทีละบัญชีด้วยรหัสผ่าน `Member@2569` แล้วเปิดหน้า "ห้องพัก" — ผลต้องตรงทุกแถว (ตารางเดียวกับหัวข้อ 9 ใน CLAUDE.md)

ล็อกอินเป็น	ห้องพักประจำ	ห้องพักตากอากาศ	ราคา/คืน
`spender1`	**จองได้ · ≥ 180 คืน**	**จองได้ · ≤ 90 คืน/แห่ง**	1,750 · 2,500
`traveler1`	ล็อก	**จองได้ · ≤ 30 คืน/แห่ง**	450 · 1,850
`explorer1`	ล็อก	**จองได้ · ≤ 90 คืน/แห่ง**	450 · 1,850
`supporter1`	**จองได้ · ≥ 91 คืน**	ล็อก	450 · 1,850
`giver1`	**จองได้**	**จองได้**	450 · 1,850
แอดมินแก้ราคา TRAVELER เป็น 500		**traveler1 เห็น 500 ทันที**	—

ราคา/คืน = ที่พักทั่วไป · ที่พักดูแลสุขภาพ (หนองคาย · ดูแลสุขภาพ และ พุทธมณฑล) · รหัสผ่านสมาชิกทุกบัญชี `Member@2569` · บัญชีลงท้าย `@club.test`

จองห้อง 6 ขั้น (ใช้ traveler1)

1. จองห้องพักตากอากาศ 3 คืน → สถานะ "รอชำระ"
2. จองอีกครั้งทั้งที่ยังไม่ชำระ → **จองไม่ได้** ขึ้นเหตุผลให้ชำระครั้งก่อนก่อน
3. แนบรูปสลิป → "รอตรวจ" · แอดมินเห็นการจองและรูปสลิป แล้วกดยืนยัน → "ยืนยันแล้ว"
4. จองครั้งใหม่ช่วงวันอื่น → ได้ · จองช่วงวันซ้อนกับการจองเดิม → **จองไม่ได้**
5. จองห้องพักประจำ → จองไม่ได้ (ระดับไม่มีสิทธิ์)
6. ประวัติขึ้นทั้งหน้า "การจองของฉัน" และหน้า "การจอง" ของแอดมิน

ด่าน 3

- ผลตรงตารางทดสอบทุกระดับ
- แอดมินแก้ราคา TRAVELER เป็น 500 แล้ว traveler1 เห็นทันที (ทดลองแล้วตั้งกลับเป็น 450)
- จองห้องครบ 6 ขั้น: ค้างชำระจองใหม่ไม่ได้ · ห้ามซ้อนวัน · แนบสลิปแล้วแอดมินยืนยันได้ · ประวัติขึ้นทั้งสองฝั่ง

ทดสอบผ่านแล้ว พิมพ์บอก Claude Code ว่า **"ผ่านแล้ว"** → ตรวจไฟล์ลับ · commit · ติ๊กเฟสใน CLAUDE.md ให้เอง

ผลไม่ตรงตาราง ใช้พรอมต์ตัวช่วยในหน้า "แก้ปัญหาที่พบบ่อย"

DEPLOY · 16.30–16.50

7 พรอมต์ 7 · ขึ้นเว็บจริง + ทดสอบบนมือถือ

เป้าหมาย: เว็บเวอร์ชันล่าสุดบน Cloudflare ล็อกอินได้จริงบนมือถือ ครบแอดมิน + 5 ระดับ ผลตรงตารางทดสอบ

1. คัดลอกพรอมต์ 7 · Claude Code ตั้งคีย์ฝั่ง Cloudflare ให้โดยไม่แสดงค่าบนจอ

2. ได้ลิงก์เว็บแล้ว เปิด Supabase → **Authentication → URL Configuration** ใส่ลิงก์ใน Site URL และ Redirect URLs

3. เปิดลิงก์บนมือถือ ล็อกอินทีละบัญชีตามตารางด้านล่าง แล้วลองจอง 1 ครั้ง + แนบรูปสลิปจากมือถือ ให้แอดมินยืนยัน

4. ผลตรงตารางแล้ว พิมพ์ "ผ่านแล้ว" ให้ Claude Code บันทึก Git

PROMPT 7

พรอมต์ 7 · ขึ้นเว็บจริง + ทดสอบบนมือถือ

```
อ่าน CLAUDE.md แล้วขึ้นเว็บเวอร์ชันล่าสุดบน Cloudflare:
1. ตั้งคีย์ฝั่ง Cloudflare ให้ครบ: NEXT_PUBLIC_SUPABASE_URL · NEXT_PUBLIC_SUPABASE_PUBLISHABLE_KEY (ต้องมีตอน build) · SUPABASE_SECRET_KEY · IMPORT_DEFAULT_PASSWORD เป็น secret ด้วย wrangler — ส่งค่าจาก .env.local ด้วยคำสั่งที่ไม่แสดงค่าบนจอ ห้ามใส่ในโค้ดหรือ Git
2. npm run deploy แล้วใช้ MCP Cloudflare ตรวจว่า Worker อัปเดตแล้ว จากนั้นส่งลิงก์ให้ฉัน
3. บอกให้ฉันเพิ่มลิงก์นี้ใน Supabase → Authentication → URL Configuration (Site URL + Redirect URLs)
4. commit ด้วย Git
5. ให้เช็คลิสต์ทดสอบบนมือถือ: ล็อกอินแอดมิน · สมาชิกครบ 5 ระดับ · จองห้อง 1 ครั้ง แนบรูปสลิปจากมือถือ แล้วแอดมินยืนยัน — ผลต้องตรงตารางทดสอบใน CLAUDE.md
```

ง่าย · พรอมต์ 7 · รันเพื่อ

เว็บจริงบนมือถือทำงานเหมือนในเครื่อง

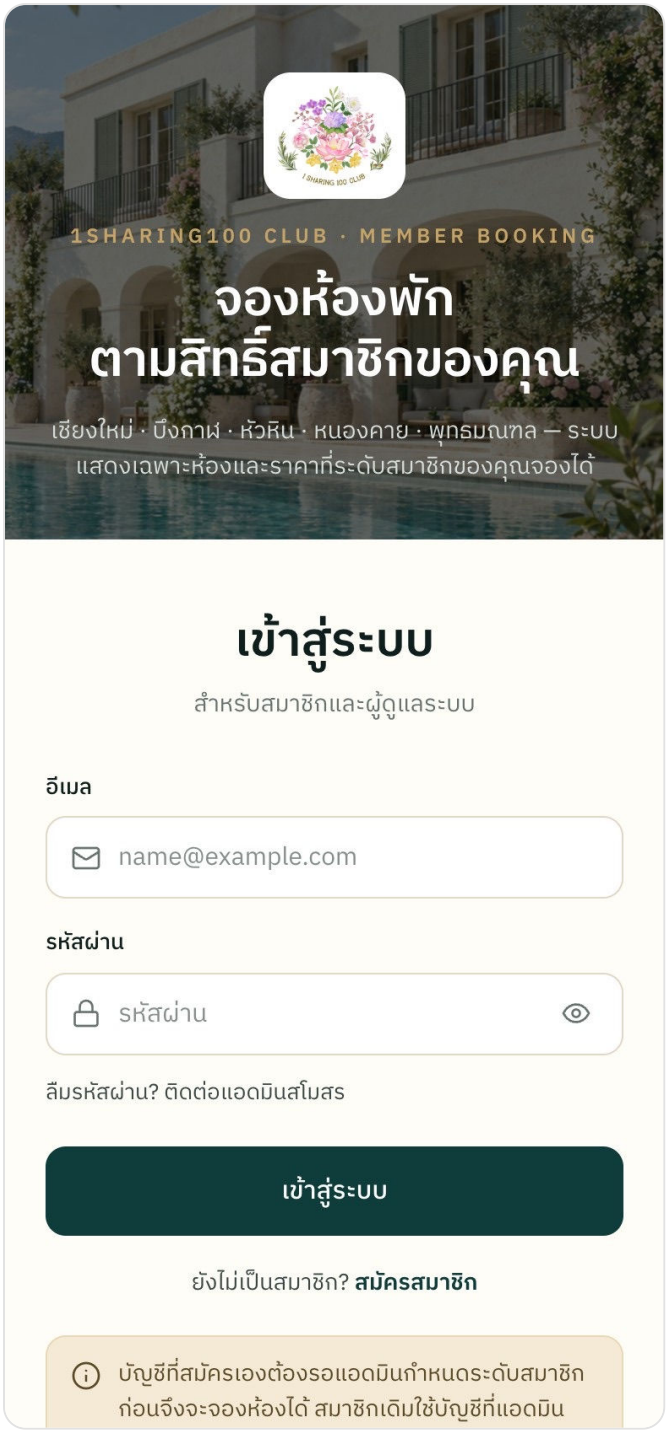
Claude Code จะทำให้

1. **ตั้งคีย์ฝั่ง Cloudflare**
   ไม่แสดงคีย์บนจอ ไม่ใส่ในโค้ด
2. **ขึ้นเว็บ แล้วตรวจผ่าน MCP**
   ส่งลิงก์มาให้
3. **บอกจุดที่ต้องตั้งใน Supabase**
   เพิ่มลิงก์เว็บจริง

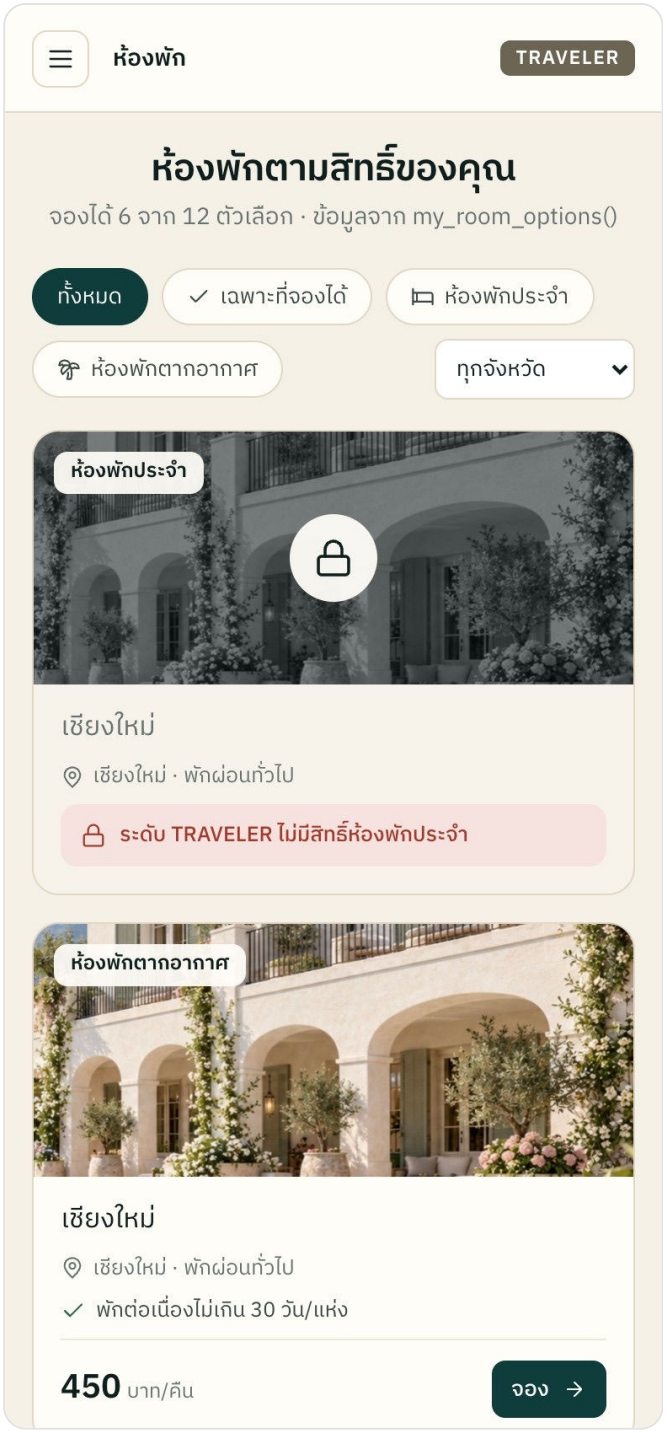
ได้อะไรกลับมา

- ✓ **ลิงก์เว็บจริงเวอร์ชันเต็ม**
- ✓ **เช็คลิสต์ทดสอบบนมือถือ**

เช็คว่าผ่าน ล็อกอินแอดมินบนมือถือได้ | ล็อกอินครบ 5 ระดับ ผลตรงตารางทดสอบ | จอง + แนบสลิปจากมือถือได้



ภาพจริง เว็บจริง (.workers.dev) บนมือถือ



ภาพจริง traveler1 บนจอมือถือ · จองได้ / ล็อก

บัญชีทดสอบบนมือถือ

บัญชี	รหัสผ่าน	ต้องเห็น
`admin@club.test`	`Admin@2569`	เมนูจัดการ 5 เมนู
`spender1@club.test`	`Member@2569`	ห้องประจำ + ตากอากาศ · 1,750 · 2,500
`traveler1@club.test`	`Member@2569`	ตากอากาศเท่านั้น · ห้องประจำล็อก
`explorer1@club.test`	`Member@2569`	ตากอากาศเท่านั้น · ≤ 90 คืน/แห่ง
`supporter1@club.test`	`Member@2569`	ห้องประจำเท่านั้น · ตากอากาศล็อก
`giver1@club.test`	`Member@2569`	ได้ทั้งสองประเภท

WRAP-UP · 16.50–17.00

สรุปวันนี้ · ระบบครบในวันเดียว



คนออกแบบ Agent ลงมือ

CLAUDE.md 1 ไฟล์ + 7 พรอมต์ = เว็บจองห้องพักจริง 1 เว็บ

บทบาท ≠ ระดับ + กฎเป็นข้อมูล

บทบาทเปิดเมนู ระดับเปิดสิทธิ์ ตัวเลขอยู่ในตาราง แอดมินแก้เอง

ด่านตรวจอยู่ที่ฐานข้อมูล

สิทธิ์และกฎการจอง 9 ข้อ ตรวจที่ฐานข้อมูล หน้าเว็บแค่บอกเหตุผล

ฝากทีมยืนยันก่อนใช้งานจริง

จุดที่คู่มือสมาชิกฉบับร่างขัดกัน 2 ข้อ (สิทธิ์ตากอากาศของ SUPPORTER · อายุสิทธิ์ตลอดชีพหรือ 1 ปี) และจำนวนห้องจริงของแต่ละที่พัก — แก้ได้เองในหน้ากฎสิทธิ์ของแอดมิน

TROUBLESHOOTING

แก้ปัญหาที่พบบ่อย

อาการ	สาเหตุ	ทางแก้
สมัครแล้วขึ้นให้ไปยืนยันอีเมล	หน้าสมัครใช้การสมัครแบบปกติ และโปรเจกต์ Supabase ใหม่เปิด Confirm email ไว้เป็นค่าเริ่มต้น	สั่ง "สมัครแล้วต้องเข้าได้ทันที ไม่ต้องยืนยันอีเมล ตามข้อ 2 ของพรอมต์ 4" · หรือปิดเองที่ Supabase → Authentication → Sign In / Providers → Email → ปิด Confirm email → Save
กดจองแล้วขึ้น"กรุณาชำระเงินการจอง..."	ยังมีการจองที่รอชำระหรือรอแอดมินตรวจสลิป	ปกติ ตามกฎข้อ 5 — แนบสลิปแล้วให้แอดมินยืนยันก่อน จึงจองครั้งใหม่ได้
Claude Code บอกว่ายังไม่เชื่อม MCP	ยังไม่ได้ Authenticate หรือเพิ่ม MCP แล้วยังไม่ได้เปิด Claude Code ใหม่	ปิดแล้วเปิด Claude Code ใหม่ → พิมพ์ `/mcp` → เลือกตัวที่ยังไม่เชื่อม → Authenticate
ล็อกอินแอดมินแล้วไม่เห็นเมนูจัดการ	สคริปต์สร้างแอดมินยังไม่ได้รัน หรือยังไม่ได้วางคีย์ลับ	วางคีย์ลับใน .env.local แล้วสั่ง "รัน scripts/create-admin.mjs อีกครั้ง" · ทางสำรอง: รัน `01-make-admin.sql` (ภาคผนวก ข)
หน้าเว็บขึ้นว่าเชื่อมฐานข้อมูลไม่ได้	ยังไม่วางคีย์ใน `.env.local` หรือยังไม่รีสตาร์ต `npm run dev`	วางคีย์ แล้วหยุด (Ctrl+C) และรันใหม่
นำเข้า CSV แล้วภาษาไทยเพี้ยน	บันทึกไฟล์ไม่ใช่ UTF-8	Excel → บันทึกเป็น "CSV UTF-8"
นำเข้าแล้วขึ้น "อีเมลนี้มีบัญชีแล้ว"	นำเข้าไฟล์เดิมซ้ำ	ปกติ ระบบกันซ้ำให้
สมาชิกเห็นห้องทุกประเภท	หน้าเว็บกรองเองแทนการเรียก `my_room_options()`	สั่ง "ใช้ผลจาก my_room_options() เท่านั้น ห้ามคำนวณสิทธิ์ในหน้าเว็บ" หรือใช้พรอมต์ตัวช่วยด้านล่าง
เว็บจริงล็อกอินไม่ได้ แต่ในเครื่องได้	ยังไม่ตั้งคีย์ฝั่ง Cloudflare หรือยังไม่เพิ่ม URL ใน Supabase	ตั้ง secret + URL Configuration
เผลอ commit ไฟล์ลับ	`.gitignore` ไม่ครบ	ให้ Agent ลบออกจาก Git และเปลี่ยนคีย์ใหม่ทันที

ผลที่เห็นไม่ตรงตารางทดสอบ

เปลี่ยนข้อความตัวแดงในวงเล็บเป็นสิ่งที่เห็นจริงก่อนส่ง เช่น `traveler1@club.test` · "ห้องพักประจำจองได้" · "ห้องพักประจำล็อก"

ตัวช่วย ตัวช่วย · ผลไม่ตรงตารางทดสอบ

```
ล็อกอินเป็น [อีเมลที่ทดสอบ] แล้วเห็น [สิ่งที่เห็น] แต่ตามตารางทดสอบใน CLAUDE.md ควรเห็น [สิ่งที่ควรเห็น]
ช่วยหาสาเหตุก่อน: หน้าเว็บใช้ผลจาก my_room_options() เท่านั้นไหม · มีตัวเลขฝังในโค้ดไหม · ค่าในตาราง tier_room_rules
และ tier_prices ของระดับนั้นถูกไหม
อธิบายสาเหตุให้ฉันฟังก่อนแก้ แล้วหยุดให้ฉันทดสอบใหม่
```

ตัวช่วย · ใช้เมื่อผลไม่ตรง · รันเพื่อ

ให้ Claude Code หาสาเหตุก่อน ไม่ใช่เดาแก้

Claude Code จะทำให้

1. **บอกว่าเห็นอะไร ควรเห็นอะไร**
   เติม 3 ช่องในวงเล็บ
2. **Claude Code ตรวจ 3 จุด**
   หน้าเว็บ · โค้ด · ค่าในฐานข้อมูล
3. **อธิบายสาเหตุก่อนแก้**
   แล้วหยุดให้เราทดสอบใหม่

ได้อะไรกลับมา

- ✓ **สาเหตุ + วิธีแก้**

เช็คว่าผ่าน ทดสอบใหม่แล้วตรงตาราง

QUICK REFERENCE

Prompt Cheat Sheet

Prompt ทั้งหมดในคู่มือ เรียงตามลำดับการใช้ — เปิดหน้าของขั้นนั้นแล้วกดคัดลอก

PROMPT	ขั้น	ใช้ทำอะไร
คำสั่ง	เช้า · 1.0 (Terminal)	ก่อนพรอมต์ 1 · เชื่อม MCP 2 ตัว (พิมพ์ใน Terminal)
พรอมต์ 1	เช้า · 1.1	พรอมต์ 1 · เตรียมเครื่องครบในรอบเดียว
พรอมต์ 2	เช้า · 1.3	พรอมต์ 2 · หน้าเว็บ "สวัสดี" ขึ้นเว็บจริง
พรอมต์ 3	เช้า · 1.4	พรอมต์ 3 · สร้าง CLAUDE.md (แผนทั้งระบบ เขียนไว้ให้แล้ว)
พรอมต์ 4	บ่าย · Phase 1	พรอมต์ 4 · Phase 1 · Login + สมัครสมาชิก + บทบาท
พรอมต์ 5	บ่าย · Phase 2	พรอมต์ 5 · Phase 2 · หลังบ้านแอดมินทั้งหมด
พรอมต์ 6	บ่าย · Phase 3	พรอมต์ 6 · Phase 3 · สิทธิ์ตามระดับ + จองห้อง + ชำระเงิน
พรอมต์ 7	บ่าย · ขึ้นเว็บจริง	พรอมต์ 7 · ขึ้นเว็บจริง + ทดสอบบนมือถือ
ตัวช่วย	ใช้เมื่อผลไม่ตรง	ตัวช่วย · ผลไม่ตรงตารางทดสอบ

APPENDIX

ภาคผนวก · ไฟล์ starter

ไฟล์เดียวกับในชุด `club-booking-starter.zip` — ปกติเปิดจากโปรเจกต์ได้เลย ใช้หน้านี้เมื่อต้องการคัดลอกไปวางใน Supabase SQL Editor โดยตรง

ก `supabase/00-setup.sql`

ตาราง 7 ตาราง · trigger สร้าง profiles อัตโนมัติ · `is_admin()` · Row Level Security · ฟังก์ชันด่านตรวจสิทธิ์ · ข้อมูลตั้งต้นจากคู่มือฉบับร่าง

FILE supabase/00-setup.sql

```
-- ======================================================================
-- 1SHARING100 CLUB · ระบบจองห้องพักตามสิทธิ์สมาชิก · Day 1 starter
-- วางทั้งไฟล์ใน Supabase → SQL Editor → Run (รันซ้ำได้ ข้อมูลที่แอดมินแก้แล้วจะไม่ถูกเขียนทับ)
-- ตัวเลขทั้งหมดมาจาก "คู่มือสมาชิกสโมสร ฉบับร่าง" — แก้ได้ภายหลังในหน้า Admin
-- ======================================================================

-- ---------- 1) ตารางข้อมูลระบบ (กฎสิทธิ์เก็บเป็น "ข้อมูล" ไม่เขียนลงในโค้ด) ----------

-- ระดับสมาชิก 5 ระดับ
create table if not exists public.tiers (
  code            text primary key,                  -- SPENDER, TRAVELER, ...
  sort            int  not null,
  value_text      text not null,                     -- มูลค่าสมาชิก (ข้อความแสดงผล)
  nights_per_year int  not null check (nights_per_year between 0 and 366),
  advance_months  int  not null check (advance_months >= 0),
  validity_days   int  check (validity_days is null or validity_days > 0), -- null = ตลอดชีพ
  room_choice     text not null default 'one_of' check (room_choice in ('one_of','all')),
  note            text
);

-- ประเภทห้องพัก 2 แบบ
create table if not exists public.room_types (
  code    text primary key,                          -- residence, resort
  name_th text not null,
  sort    int  not null
);

-- กฎ: ระดับไหนจองห้องประเภทไหนได้ + เงื่อนไขจำนวนคืน
create table if not exists public.tier_room_rules (
  tier_code    text not null references public.tiers(code) on update cascade on delete cascade,
  room_type    text not null references public.room_types(code) on update cascade on delete cascade,
  allowed      boolean not null default false,
  min_nights   int check (min_nights is null or min_nights > 0),  -- ขั้นต่ำต่อการจอง 1 ครั้ง
  max_nights   int check (max_nights is null or max_nights > 0),  -- สูงสุดต่อครั้งต่อแห่ง
  condition_th text,
  primary key (tier_code, room_type)
);

-- ที่พัก 6 แห่ง (หนองคายมี 2 แบบ)
create table if not exists public.branches (
  code        text primary key,
  name_th     text not null,
  province    text not null,
  kind        text not null check (kind in ('general','care')),  -- general = พักผ่อนทั่วไป, care = ดูแลสุขภาพ
  description text,
  sort        int  not null
);

-- จำนวนห้องของแต่ละที่พัก แยกตามประเภทห้อง (ใช้กันจองเกินจำนวนในวันที่ 2)
create table if not exists public.branch_rooms (
  branch_code text not null references public.branches(code) on update cascade on delete cascade,
  room_type   text not null references public.room_types(code) on update cascade on delete cascade,
  total_rooms int  not null default 0 check (total_rooms >= 0),
  primary key (branch_code, room_type)
```

```
);

-- ราคาต่อคืน ตามระดับสมาชิก × ประเภทที่พัก
create table if not exists public.tier_prices (
  tier_code       text not null references public.tiers(code) on update cascade on delete cascade,
  branch_kind     text not null check (branch_kind in ('general','care')),
  price_per_night int  not null check (price_per_night >= 0),
  primary key (tier_code, branch_kind)
);

-- ---------- 2) โปรไฟล์ผู้ใช้ (ผูกกับระบบล็อกอินของ Supabase) ----------
-- role = บทบาทในระบบ (เข้าเมนูไหนได้) · tier_code = ระดับสมาชิก (จองห้องอะไรได้) — คนละเรื่องกัน
create table if not exists public.profiles (
  id              uuid primary key references auth.users(id) on delete cascade,
  email           text unique,
  first_name      text,
  last_name       text,
  phone           text,
  nationality     text,
  role            text not null default 'member' check (role in ('admin','member')),
  tier_code       text references public.tiers(code) on update cascade on delete set null,
  tier_since      date,
  preferred_branch text references public.branches(code) on update cascade on delete set null,
  created_at      timestamptz not null default now()
);

-- สร้างแถว profiles ให้อัตโนมัติทุกครั้งที่มีบัญชีใหม่
create or replace function public.handle_new_user()
returns trigger language plpgsql security definer set search_path = public as $$
begin
  insert into public.profiles (id, email, first_name, last_name, phone)
  values (new.id, new.email,
          new.raw_user_meta_data->>'first_name',
          new.raw_user_meta_data->>'last_name',
          new.raw_user_meta_data->>'phone')
  on conflict (id) do nothing;
  return new;
end $$;

drop trigger if exists on_auth_user_created on auth.users;
create trigger on_auth_user_created
  after insert on auth.users
  for each row execute function public.handle_new_user();

-- ตัวช่วยเช็คว่าคนที่ล็อกอินอยู่เป็นแอดมินหรือไม่ (ใช้ในกฎ RLS)
create or replace function public.is_admin()
returns boolean language sql stable security definer set search_path = public as $$
  select exists (select 1 from public.profiles where id = auth.uid() and role = 'admin');
$$;

-- ---------- 3) Row Level Security — ฐานข้อมูลเป็นคนเฝ้าประตู ไม่ใช่แค่ซ่อนปุ่ม ----------
alter table public.profiles        enable row level security;
alter table public.tiers           enable row level security;
alter table public.room_types      enable row level security;
alter table public.tier_room_rules enable row level security;
alter table public.branches        enable row level security;
alter table public.branch_rooms    enable row level security;
alter table public.tier_prices     enable row level security;

-- profiles: สมาชิกเห็นเฉพาะแถวตัวเอง · แอดมินเห็นและแก้ได้ทุกแถว · สมาชิกแก้ tier/role ตัวเองไม่ได้
drop policy if exists "profiles read own or admin" on public.profiles;
create policy "profiles read own or admin" on public.profiles
  for select to authenticated using (id = auth.uid() or public.is_admin());
drop policy if exists "profiles admin update" on public.profiles;
create policy "profiles admin update" on public.profiles
  for update to authenticated using (public.is_admin()) with check (public.is_admin());

-- ตารางกฎ/ที่พัก/ราคา: ทุกคนที่ล็อกอินอ่านได้ · แก้ได้เฉพาะแอดมิน
do $$
declare t text;
begin
```

```
  foreach t in array array['tiers','room_types','tier_room_rules','branches','branch_rooms','tier_prices'] loop
    execute format('drop policy if exists "%s read" on public.%I', t, t);
    execute format('create policy "%s read" on public.%I for select to authenticated using (true)', t, t);
    execute format('drop policy if exists "%s admin write" on public.%I', t, t);
    execute format('create policy "%s admin write" on public.%I for all to authenticated using
(public.is_admin()) with check (public.is_admin())', t, t);
  end loop;
end $$;

-- ---------- 4) ตรรกะตรวจสิทธิ์ (หลังบ้าน = พ่อครัวที่ถือกฎ) ----------
-- คืนรายการ "ที่พัก × ประเภทห้อง" ทั้งหมด พร้อมบอกว่าระดับนี้จองได้ไหม ถ้าไม่ได้เพราะอะไร และราคาเท่าไร
create or replace function public.room_options(p_tier text)
returns table (
  branch_code text, branch_name text, province text, branch_kind text,
  room_type text, room_type_name text,
  allowed boolean, reason text,
  price_per_night int, min_nights int, max_nights int, condition_th text
)
language sql stable set search_path = public as $$
  select b.code, b.name_th, b.province, b.kind,
         rt.code, rt.name_th,
         (p_tier is not null and t.code is not null and coalesce(r.allowed, false)) as allowed,
         case
           when p_tier is null   then 'บัญชีนี้ยังไม่มีระดับสมาชิก กรุณาติดต่อแอดมิน'
           when t.code is null   then 'ไม่พบระดับสมาชิก ' || p_tier || ' ในระบบ'
           when not coalesce(r.allowed, false)
                                 then 'ระดับ ' || p_tier || ' ไม่มีสิทธิ์' || rt.name_th
         end as reason,
         tp.price_per_night, r.min_nights, r.max_nights, r.condition_th
  from public.branches b
  join public.branch_rooms br on br.branch_code = b.code and br.total_rooms > 0
  join public.room_types rt   on rt.code = br.room_type
  left join public.tiers t           on t.code = p_tier
  left join public.tier_room_rules r on r.tier_code = p_tier and r.room_type = rt.code
  left join public.tier_prices tp    on tp.tier_code = p_tier and tp.branch_kind = b.kind
  order by b.sort, rt.sort;
$$;

-- สำหรับสมาชิก: ตรวจจากระดับของคนที่ล็อกอินอยู่เอง (ส่งระดับปลอมมาไม่ได้)
drop function if exists public.my_room_options();
create function public.my_room_options()
returns table (
  branch_code text, branch_name text, province text, branch_kind text,
  room_type text, room_type_name text,
  allowed boolean, reason text,
  price_per_night int, min_nights int, max_nights int, condition_th text
)
language sql stable set search_path = public as $$
  select * from public.room_options((select tier_code from public.profiles where id = auth.uid()));
$$;

-- ---------- 5) ข้อมูลตั้งต้น (จากคู่มือฉบับร่าง · รันซ้ำจะไม่เขียนทับค่าที่แอดมินแก้แล้ว) ----------
insert into public.tiers (code, sort, value_text, nights_per_year, advance_months, validity_days, room_choice,
note) values
  ('SPENDER',   1, '1,000,000 บาท/ปี',                 365,  3, 365,  'one_of', 'เลือกห้องพักประจำ หรือ ห้องพักตากอากาศ
อย่างใดอย่างหนึ่ง'),
  ('TRAVELER',  2, '2,000,000 บาท หรือ 200 หุ้น',       90,   6, null, 'one_of', 'ห้องพักตากอากาศเท่านั้น'),
  ('EXPLORER',  3, '3,880,000 บาท หรือ 388 หุ้น',       365,  6, null, 'one_of', 'ห้องพักตากอากาศเท่านั้น'),
  ('SUPPORTER', 4, '3,880,000 บาท หรือ 388 หุ้น',       365,  8, null, 'one_of', 'ห้องพักประจำเท่านั้น · ต่อสิทธิ์ห้องประจำต่อ
เนื่องได้ จำกัด 20 สิทธิ์/แห่ง'),
  ('GIVER',     5, '10,000,000 บาท หรือ 1,000 หุ้น',    365, 12, null, 'all',    'ได้ทั้งห้องพักประจำ และ ห้องพักตากอากาศ')
on conflict (code) do nothing;

insert into public.room_types (code, name_th, sort) values
  ('residence', 'ห้องพักประจำ',    1),
  ('resort',    'ห้องพักตากอากาศ', 2)
on conflict (code) do nothing;

insert into public.tier_room_rules (tier_code, room_type, allowed, min_nights, max_nights, condition_th) values
  ('SPENDER',   'residence', true,  180,  null, 'วันเข้าพักจริงขั้นต่ำ 6 เดือน'),
  ('SPENDER',   'resort',    true,  null, 90,   'พักต่อเนื่องไม่เกิน 90 วัน/แห่ง'),
```

```
  ('TRAVELER',  'residence', false, null, null, null),
  ('TRAVELER',  'resort',    true,  null, 30,   'พักต่อเนื่องไม่เกิน 30 วัน/แห่ง'),
  ('EXPLORER',  'residence', false, null, null, null),
  ('EXPLORER',  'resort',    true,  null, 90,   'พักต่อเนื่องไม่เกิน 90 วัน/แห่ง'),
  ('SUPPORTER', 'residence', true,  91,   null, 'พักต่อเนื่องเกิน 90 วัน · วันพักจริงไม่ต่ำกว่า 50% ของวันจอง'),
  ('SUPPORTER', 'resort',    false, null, null, null),
  ('GIVER',     'residence', true,  91,   null, 'พักต่อเนื่องเกิน 90 วัน · วันพักจริงไม่ต่ำกว่า 50% ของวันจอง'),
  ('GIVER',     'resort',    true,  null, 90,   'พักต่อเนื่องไม่เกิน 90 วัน/แห่ง')
on conflict (tier_code, room_type) do nothing;

insert into public.branches (code, name_th, province, kind, description, sort) values
  ('chiangmai',      'เชียงใหม่',                'เชียงใหม่', 'general', 'พักผ่อนทั่วไป', 1),
  ('buengkan',       'บึงกาฬ',                   'บึงกาฬ',    'general', 'พักผ่อนทั่วไป', 2),
  ('huahin',         'หัวหิน',                   'ประจวบคีรีขันธ์', 'general', 'พักผ่อนทั่วไป', 3),
  ('nongkhai',       'หนองคาย',                  'หนองคาย',   'general', 'พักผ่อนทั่วไป', 4),
  ('nongkhai-care',  'หนองคาย · ดูแลสุขภาพ',     'หนองคาย',   'care',    'สนับสนุนทุนการศึกษา และรับบริการรักษาจากแพทย์และ
พยาบาล', 5),
  ('phutthamonthon', 'พุทธมณฑล · ดูแลสุขภาพ',    'นครปฐม',    'care',    'รับบริการรักษาจากแพทย์และพยาบาล', 6)
on conflict (code) do nothing;

-- จำนวนห้องเป็นตัวเลขสมมติสำหรับฝึก (ยังไม่มีในคู่มือ) — แก้ได้ในหน้า Admin
insert into public.branch_rooms (branch_code, room_type, total_rooms)
select b.code, rt.code, case rt.code when 'residence' then 20 else 30 end
from public.branches b cross join public.room_types rt
on conflict (branch_code, room_type) do nothing;

insert into public.tier_prices (tier_code, branch_kind, price_per_night) values
  ('SPENDER',   'general', 1750), ('SPENDER',   'care', 2500),
  ('TRAVELER',  'general',  450), ('TRAVELER',  'care', 1850),
  ('EXPLORER',  'general',  450), ('EXPLORER',  'care', 1850),
  ('SUPPORTER', 'general',  450), ('SUPPORTER', 'care', 1850),
  ('GIVER',     'general',  450), ('GIVER',     'care', 1850)
on conflict (tier_code, branch_kind) do nothing;

-- เสร็จแล้ว: ควรเห็นระดับสมาชิก 5 แถว
select code, nights_per_year, advance_months, room_choice from public.tiers order by sort;
```

ข supabase/01-make-admin.sql

FILE

supabase/01-make-admin.sql

```
-- ตั้งบัญชีแรกให้เป็นแอดมิน (รันหลังสร้างบัญชี admin@club.test ในเมนู Authentication → Users → Add user)
-- ถ้าใช้อีเมลอื่น ให้เปลี่ยนอีเมลในบรรทัดล่างให้ตรงกัน
update public.profiles
set role = 'admin', first_name = 'ผู้ดูแล', last_name = 'ระบบ'
where email = 'admin@club.test';

-- ควรเห็น admin@club.test บทบาท admin 1 แถว
select email, role from public.profiles where role = 'admin';
```

ค ตัวอย่างไฟล์ CSV

`data/members-demo.csv` สมาชิกสมมติ 10 คน · ข้อมูลจำลองทั้งหมด

FILE data/members-demo.csv

```
email,first_name,last_name,phone,nationality,tier,preferred_branch
spender1@club.test,ปวีณา,ศรีสุข,+66 81 000 1001,ไทย,SPENDER,huahin
spender2@club.test,ธนกร,วงศ์ทอง,+66 81 000 1002,ไทย,SPENDER,chiangmai
traveler1@club.test,ณัฐชา,บุญมา,+66 81 000 2001,ไทย,TRAVELER,buengkan
traveler2@club.test,Emma,Clarke,+44 7700 900201,British,TRAVELER,huahin
explorer1@club.test,กิตติพัฒน์,แสงทอง,+66 81 000 3001,ไทย,EXPLORER,nongkhai
explorer2@club.test,อรอุมา,จันทร์เพ็ญ,+66 81 000 3002,ไทย,EXPLORER,chiangmai
supporter1@club.test,วรเชษฐ์,มั่นคง,+66 81 000 4001,ไทย,SUPPORTER,nongkhai-care
supporter2@club.test,พิมพ์ชนก,ทองดี,+66 81 000 4002,ไทย,SUPPORTER,phutthamonthon
giver1@club.test,สมชาย,ใจบุญ,+66 81 000 5001,ไทย,GIVER,phutthamonthon
giver2@club.test,มาลัย,รุ่งเรือง,+66 81 000 5002,ไทย,GIVER,huahin
```

`data/members-with-errors.csv` ไฟล์ทดสอบการตรวจ 4 แถว

FILE data/members-with-errors.csv

```
email,first_name,last_name,phone,nationality,tier,preferred_branch
newmember1@club.test,ภูมิ,ใจดี,+66 81 000 6001,ไทย,EXPLORER,huahin
traveler1@club.test,ณัฐชา,บุญมา,+66 81 000 2001,ไทย,TRAVELER,buengkan
newmember2@club.test,ศิริพร,แก้วใส,+66 81 000 6002,ไทย,PLATINUM,chiangmai
newmember3-club.test,กานต์,ดีมาก,+66 81 000 6003,ไทย,GIVER,huahin
```

ติดต่อสถาบัน ZenityX

โทร
062-262-9644

อีเมล
admin@zenityxai.com

ออกแบบหลักสูตรโดยสถาบัน ZenityX · Full-Stack Vibe Coding · ระบบจองห้องพักตามสิทธิ์สมาชิก · Day 1: Login · บทบาท · สิทธิ์ตามระดับสมาชิก · ข้อมูลสมาชิก อีเมล และเบอร์โทรทั้งหมดในคู่มือเป็นข้อมูลจำลองเพื่อการเรียนการสอน · ตัวเลขสิทธิ์มาจากคู่มือสมาชิกสโมสรฉบับร่าง