

재도전 멘토링 | AI와 함께하는 창업자의 로드맵

AI로 강화하는 스타트업 ROI와 Early-Stage에서의 활용 사례

효율성을 넘어 사업의 도구가 되어가는 AI

송건호

해그로시 대표 · 레스토랑라이즈 파운더 · 리보온 이사



기획과 고객 검증의 시간들



2012

창업 씬에서의 활동 시작

10만원으로 창업하기(이동 손세차 프로젝트), 중고 시계 폴리싱 서비스의 블로그 마케팅 비즈니스, IOS보안 어플리케이션 기획



2019

스타트업 그로스 총괄 (조직/브랜드 총괄)

0명, 4명, 10명대, 20명대, 50명대 이르기까지 다양한 조직이 우선순위와 지표 기반으로 빠르게 성장하는 법을 만드는 역할로.



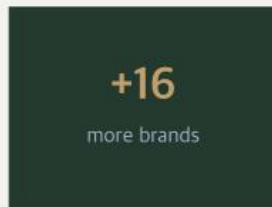
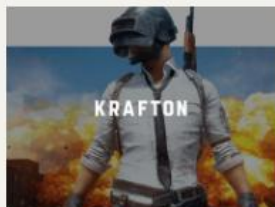
2025

얼리 스테이지 퍼널 교육 & 2개의 브랜드 창업

누적 핸들링 고객 수 500만 이상, 누적 집행 광고비 80억 이상, 제로베이스 신사업, 공동창업부터 대기업계열 시리즈C 리딩까지.

기획과 성취욕, 숫자로 증명되는 과정의 재미였습니다.

글로벌 대기업부터 창업까지



20

외부에서 대표의 고민을
함께 푼 스타트업

7

조직 내부, 대표 옆에서
지표를 키운 스타트업

3

지금 진행 중인
법인·초기 사업

매출 0원·팀원 0명·BM 부재 단계 ~ 매출 1천억·시리즈 C까지

NOW

다시, 2개의 브랜드 창업자로

미들라이프 헬스케어 살롱 및 제품 브랜드

리보온 (ReBoOn)

대한민국 중년들의 건강한 움직임을 위해

웰니스 소셜임팩트 캠페인 브랜드

레스트앤라이즈 (REST&RISE)

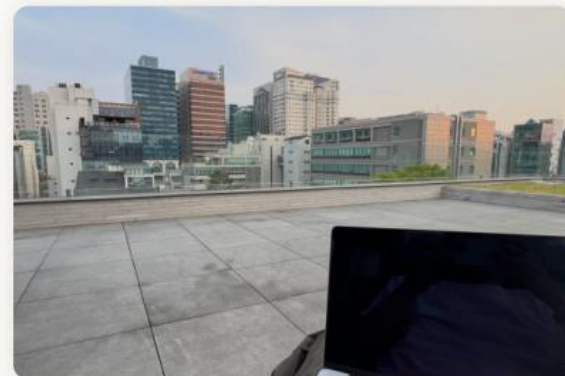
진짜 웰니스가 필요한 사람들을 위해

NOW

미들라이프 헬스케어 살롱 · 제품 브랜드

리보온 (ReBoOn)

대한민국 중년들의 건강한 움직임을 위해



NOW

웰니스 소셜임팩트 캠페인 브랜드

레스트앤라이즈 (REST&RISE)

진짜 웰니스가 필요한 사람들을 위해



봉사활동을 시작으로
사회에 도움 되는 웰니스
프로그램을 만들기로 했거든요.

자연, 사람, 동물...
그 누구를 대해서도 서툴지만
노력하고 있어요.

기존의 웰니스 브랜드 치곤
조금 독특하죠?



AGENDA

목차



PART 1

요즘 AI로 만들고 활용하는 것들

지금 돌리는 실제 사례 · CLI로의 전환 · 바뀌는 창업의 구조



PART 2

의사결정의 근간, 그로스해킹과 AI 연결

속도와 비용을 얻은 다음 · 가설 수립 · 통계적 검증 · 회고

PART 1

요즘 AI로 만들고 활용하는 것들

스킬 가이드가 아닌 사용하고 있는 툴에 대해

현황 | AI로 서포트하는 4개 영역

현재 법인부터 실제 수익 파이프라인의 팀원 역할을 하는 AI



①

리보온 헬스케어 피봇팅

공동창업자 2명의 효율성을 위한 진단지 AI



②

해그로시 프로젝트 관리

기존 법인의 인건비 축소를 대체하는 콘텐츠 AI



③

고객 여정 설계 전반

홈페이지부터 CRM 및 모든 여정을 AI 활용



④

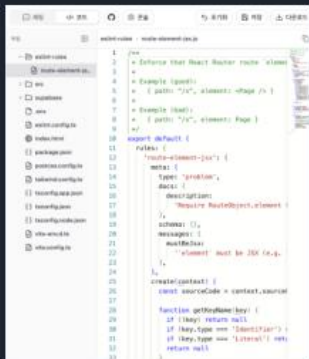
유튜브 채널 수익화

음악 제작부터 숏폼 디자인과 업로드까지 AI로

리보온 — 미드라이프 헬스케어 커뮤니티에서의 AI



실제 서비스 화면 — 리보온 미드라이프 헬스케어 진단 (10문항 → 맞춤 리포트)



AI가 작성한 코드베이스

노코드툴(개발자가 필요없는)이 기존에 많다고 하지만,
제대로 된 디자인-커스터마이징 작업물은
여전히 인적 자원이 필요했던 영역
> 만약 html코드를 직접 짜야 했다면...?

MVP 제작 과정에서의 비용 대신 — 대부분 AI로

기존 방식

직접 HTML 코딩 · 노코드 툴

— 홈페이지 설계

외주·퍼블리셔에 맡기거나 노코드로 한 화면씩 끄공

— 진단 검사지 제작 (노코드툴)

문항·로직·점수화를 따로 만들고 붙이는 데 손이 많이 감

— 서비스에 탑재

설계·검사지·발송 연결엔 결국 개발 리소스 필요

— 수정·확장

고칠 때마다 외주·대기 혹은 기능 한계

AI(CLI)로 했을 때

공동창업자 2명이 직접



홈페이지 설계

‘이런 페이지 만들어줘’ → 레이아웃·코드까지 그 자리에서



진단 검사지 제작 (나만의 어드민)

문항·점수 로직·결과 분기를 AI가 짜고 바로 엮음



서비스에 탑재

랜딩·검사지가 하나로 연결된 서비스가 굴러감 > API 연동 등 모든 걸 쉽고 빠르게



수정·확장

고칠 부분만 말하면 즉시 반영하며 점차 학습하며 디벨롭까지

그로스 코칭 채널 — 콘텐츠 퍼널 스튜디오 한 곳에서

The screenshot shows a web interface for '콘텐츠 퍼널 스튜디오' (Content Funnel Studio). On the left, there's a sidebar with a list of items. The main area is titled '기획 요청' (Planning Request) and contains a form. The form has a '대상' (Target) dropdown set to '단발 (기타 - 시리즈 없음)' (One-off (Other - No series)). Below it, '파일 유형' (File Type) is set to '카드뉴스' (Card News) and '장수' (Length) is set to '5'. There are buttons for '일괄 설명' (Batch Description) and '슬라이드별' (Slide-by-slide). A text box contains the instruction: '이번 콘텐츠 설명 - AI가 5장으로 말아서 나눠 구성해요.' (This content description - AI will divide it into 5 slides and structure it). At the bottom, there's a '기획 안 추가' (Don't add plan) button and a '기획 산출' (Generate plan) button. On the right, a '산출 결과' (Output Result) section shows a message: '원목에서 요청하면 여기 기획안이 나오요.' (If you request from the source, the plan will appear here).

실제 화면 — 콘텐츠 기획소 (조건 입력 → AI 기획안 산출)

스쳐 가는 콘텐츠감을, 놓치지 않고 한 흐름으로



1 메모

머릿속에 스치는 콘텐츠감을 그 자리에서 바로 기록



2 AI 디벨롭

메모를 카드뉴스 등 세부 콘텐츠 형태로 확장



3 다듬기

톤·구성·카피를 채널에 맞게 손볼



4 업로드 · 편성표

채널 업로드와 발행 편성을 한 곳에서 관리



5 레퍼런스 학습

레퍼런스를 올려 다음 기획을 AI가 더 똑똑하게



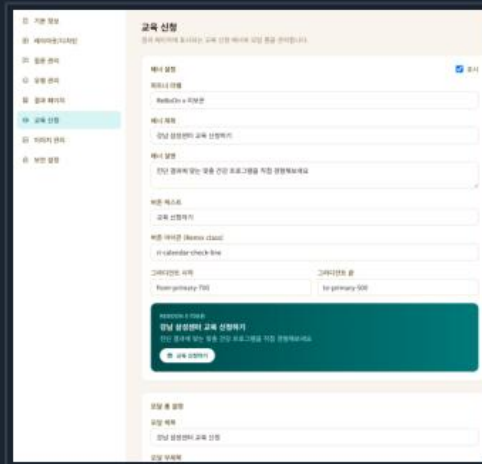
실제 산출물

기획안이 카드뉴스로 바로 디벨롭됩니다

고객 여정 설계 — 리드 수집부터 CRM·어드민 관리까지



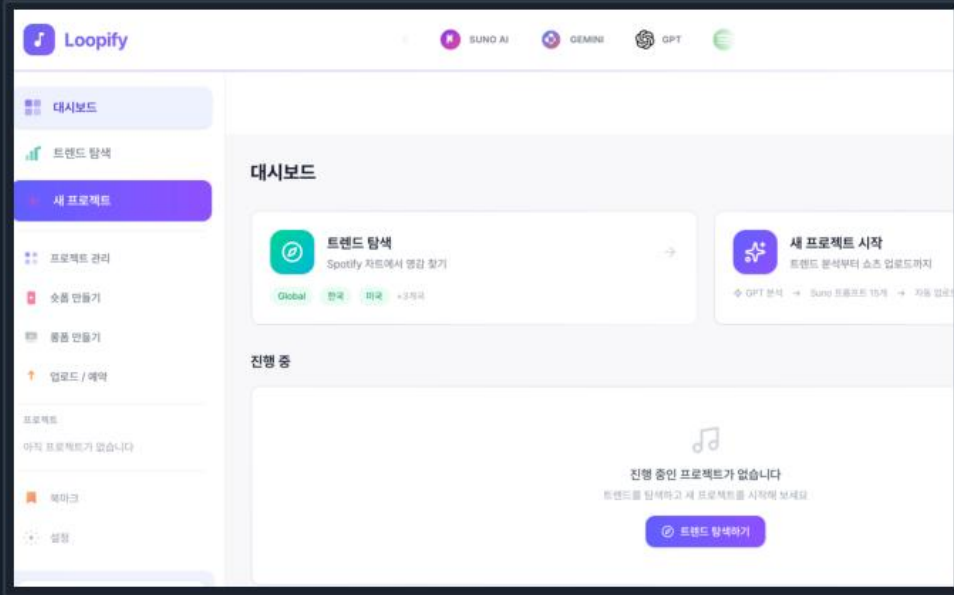
실제 화면 — 어드민에서 보는 리드·응답 관리



AI로 달라지는 지점

- ✓ **랜딩·퍼널을 직접 만든다** 기존 아임웹, 프레이머 등의 편한 홈페이지 툴도 많지만 요즘은 랜딩페이지를 만드는 것도 기획안만 있으면 5분, 10분 안에 링크 배포까지 준비 끝
- ✓ **연동·자동화를 AI가 거든다** CRM API 연결·데이터 정리를 코드로 자동화해서, 기존에는 데이터 대시보드도 작업하고 구글시트로 정리하던 영역들을 모두 정합성 맞는 리포트로 완성
- ✓ **관리가 한 화면으로 모인다** 흩어진 리드를 어드민에서 한눈에 운영. 결국 일을 하다 보면 내가 편한 방식, 우리가 원하는 UI/UX가 필요한데 기획자가 아니어도 충분히 만들 수 있음

유튜브 음악 채널 — 제작부터 업로드·분석까지 한 번에



실제 화면 — 음악 채널 스튜디오 (SUNO·GPT·Gemini 연동, 트렌드→숏폼→업로드)

내가 전하려는 음악을, 영상까지 한 흐름으로



1 음악 제작

전하고자 하는 메시지·분위기를 AI로 만들어 1곡이든 10곡이든 빠르게 추출 가능



2 숏폼 이미지·형태

음악에 어울리는 비주얼과 숏폼 포맷 생성을 디자이너 대신 1-2분 안에 템플릿으로



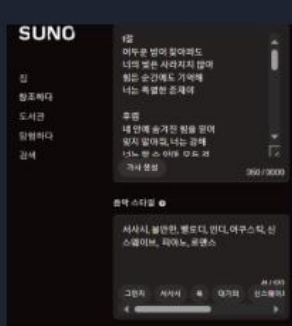
3 업로드

완성된 콘텐츠를 채널에 바로 발행



4 채널 분석

반응·지표를 받아 다음 제작에 반영 (일부 콘텐츠 지표까지 연동해서 인사이트 확보)



AI 음악 생성 · 선곡

전하고 싶은 분위기를 입력하면 음악이 나오고, 어울리는 선곡까지

실행 환경의 전환 | 클로드에게 바로 묻고, CLI와 바로 작업

Chat

Cowork

Code

새 작업

프로젝트

아티팩트

예정됨

발송 베타

사용자 지정

최근 항목

- PC integration setup
- Instagram tool token usage concerns
- Seeking guidance on next steps
- MacBook Pro keyboard shortcuts not
- Troubleshoot error message
- WeWork vs Fivespot membership com
- MacBook Pro screen mirroring not wo
- Fix unwanted photo library sync issue

Cowork 알아보기

1/3

- 역할에 맞게 Claude 맞춤 설정
- 반복 작업 예약
- 알림 켜기

Seeking guidance on next steps

사고 과정 >

1) 어떤 키를 줘야 하나

→ Secret 키를 줍니다.

Supabase가 최근에 UI를 바꿔서 이름이

새 이름 (지금 화면)	옛 이름	용도
Publishable	anon	공개 OK
Secret	service_role	절대비공개

다른 터미널이 요청한

SUPABASE_SERVICE_ROLE_KEY = S

다. eyJ... 로 시작하는 건 JWT거나,

sb_secret... 로 시작합니다. Reve

클릭해서 가져가세요.

2) .env.local 에 저장하면서 보안 유

그 다른 터미널의 Claude가 알아서 처리

본인이 직접 확인하고 싶으시면:

(a) .gitignore 에 들어

↓

확인

hneh

메시지를 입력하세요...

yoonse-brown

MVP 웹사이트 구축 및 배포

caffeinate · claude

128x54

Last login: Wed Jun 24 13:16:19 on ttys000

gunho@geonhooi-MacBookPro: ~ % yoonse-brown % claude

Claude Code v2.1.187

Opus 4.8 (1M context) · Claude Max

~/Documents/yoonse-brown

내가 만드는 비즈니스 모델을 검증하기 위한 MVP로 웹사이트 하나를 빠르게 열어보려고 해. 로그인이나 가입, 고객 DB를 받는 과정이 있다 보니 보안을 신경 써야 해. 클로드로 작업하고, 그 이후에는 깃허브에 올린 후 슈파베이스-버셀까지 작업해 배포해볼까?

● 좋은 계획이에요. Claude → GitHub → Supabase → Vercel 순서는 이 종류의 MVP에 사실상 표준 조합이고, 특히 로그인·가입·고객 DB가 있으면 Supabase가 보안 부담을 크게 덜어줍니다. 핵심만 정리할게요.

이 스택이 보안상 유리한 이유

Supabase Auth + RLS(Row Level Security)가 보안의 90%를 책임집니다.

- 비밀번호 해싱, 세션/JWT, 이메일 인증, 비밀번호 재설정 → 직접 구현 안 함 (직접 만들면 거의 항상 틀림)

- RLS가 진짜 핵심: "이 row는 이 유저만 본다"를 DB 레벨에서 강제. 앱 코드에 버그가 있어도 DB가 막아줌. 고객 DB 다룰 때 이거 하나로 사고의 대부분을 예방합니다.

순서는 살짝 바꾸는 걸 추천

1. Claude로 로컬 개발 (Next.js + Supabase 클라이언트)
2. Supabase 프로젝트 먼저 생성 → 스키마 + RLS 정책 짜기
3. GitHub 푸시
4. Vercel 연결 → 환경변수 (Supabase 키) 주입 → 배포

Supabase를 GitHub보다 먼저 세팅하는 이유: 인증 코드를 짜려면 Supabase URL·키가 있어야 하고, RLS를 처음부터 켜고 개발해야 나중에 "썻더니 다 안 보임" 사태를 피합니다.

처음부터 꼭 지킬 보안 기본기

- 키 분리: anon key(브라우저 노출 OK) vs service_role key(절대 클라이언트 금지, 서버 전용). .env.local은 .gitignore에, Vercel엔 환경변수로만.

- RLS는 모든 테이블에 기본 ON, 정책 없으면 접근 차단되게.

- 고객 개인정보(이메일·전화 등) 다루면 → Vercel WAF/rate limit, 그리고 한국이면 개인정보처리방침 페이지도 MVP에 같이.

한 가지 확인 - 이게 갈림길이라 짚고 갑니다

지금 이 레포(yoonse-brown)는 어드민/스튜디오가 들어있는 기존 프로젝트예요. 이번 MVP를 어디에 만들지로 작업 방식이 갈립니다.

또 하나, 이 프로젝트의 AGENTS.md에 "이 Next.js는 너가 아는 버전과 다르다, 코드 짜기 전 node_modules/next/dist/docs 먼저 읽어라"라고 못박혀 있어서, 실제 코딩 들어가면 제가 그 문서부터 확인하고 진행할게요.

질문 하나만:

이 레포 선택

이번 MVP 사이트는 어디에 만들까요?

지금, 직접 따라 만들기

막히면 `claude.ai` 채팅에 묻고 · 실행은 `claude CLI`로 — 아래 4개 도구만 준비하면 바로 시작합니다.

1 GitHub

코드가 저장되는 저장소

2 Supabase

DB · 로그인 백엔드

3 Vercel

웹에 바로 배포

4 Claude Code

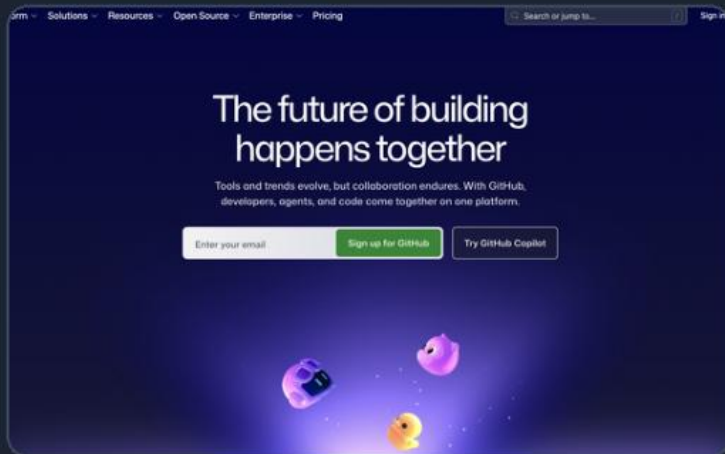
AI 코딩 파트너 (CLI)

폰으로 이 가이드 열기



`consultant-sgh.vercel.app`

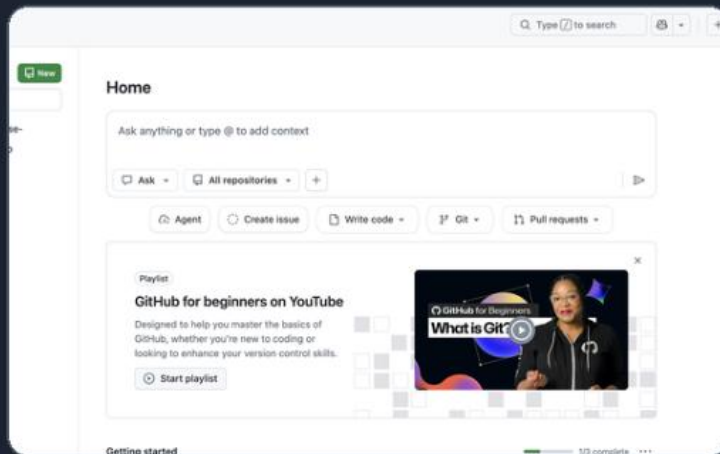
GitHub — 코드가 저장되는 곳



STEP ①

접속 · 가입

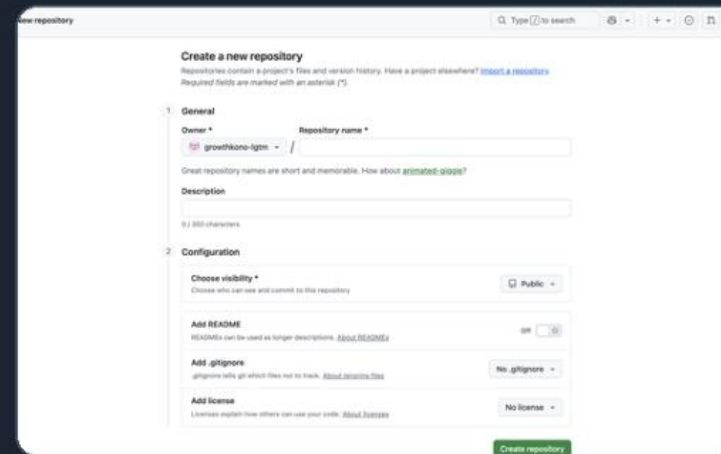
github.com에서 우측 상단 Sign up으로 계정 만들기



STEP ②

첫 화면

로그인하면 나오는 Home 대시보드

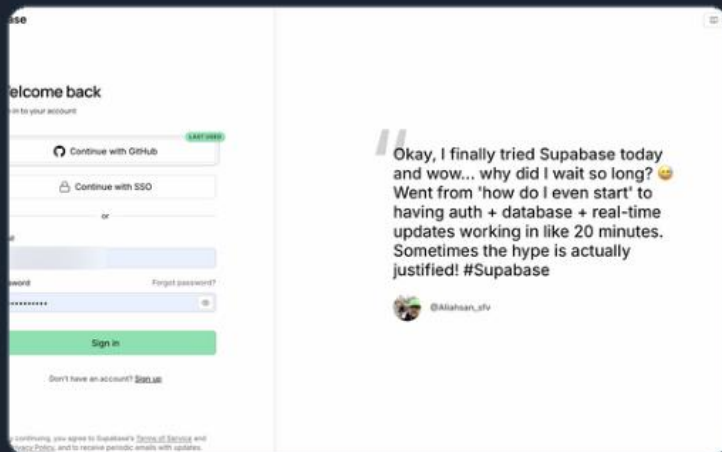


STEP ③

저장소 생성

New repository로 프로젝트가 담길 저장소 만들기

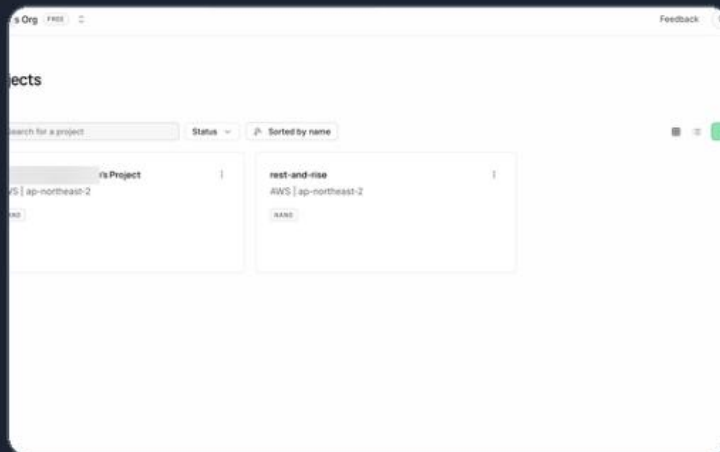
Supabase — DB · 로그인 백엔드



STEP ①

로그인

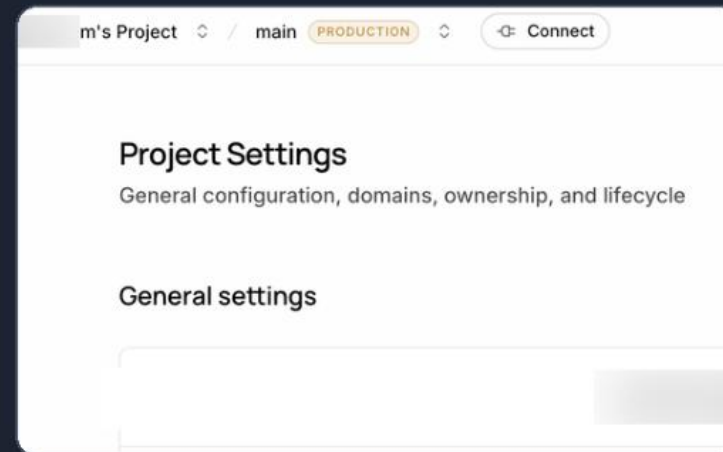
Continue with GitHub로 1초 로그인 (방금 만든 깃허브 계정)



STEP ②

프로젝트 생성

우측 New project로 무료 프로젝트 만들기



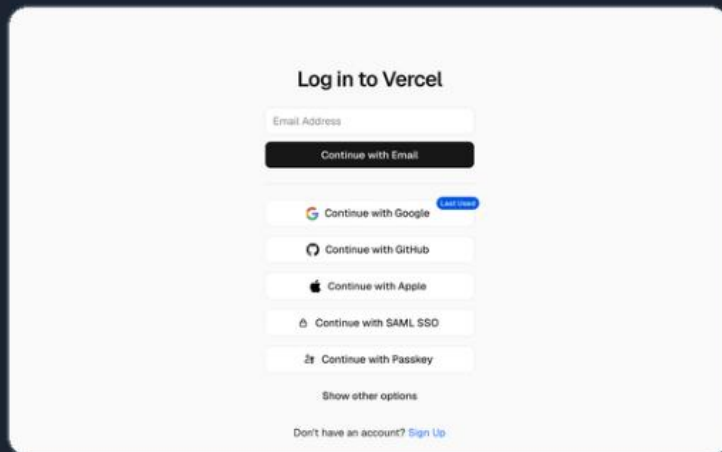
STEP ③

키 값 확인

Settings > API Keys에서 URL · 키를 복사해 둡니다

복사한 Project URL · API Key를 다음 단계 Vercel의 Environment Variables에 붙여넣으면 서비스와 DB가 연결됩니다.

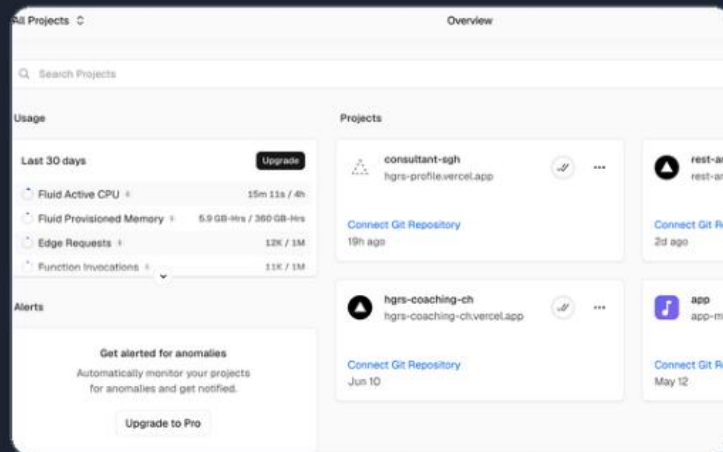
Vercel — 웹에 바로 배포



STEP ①

로그인

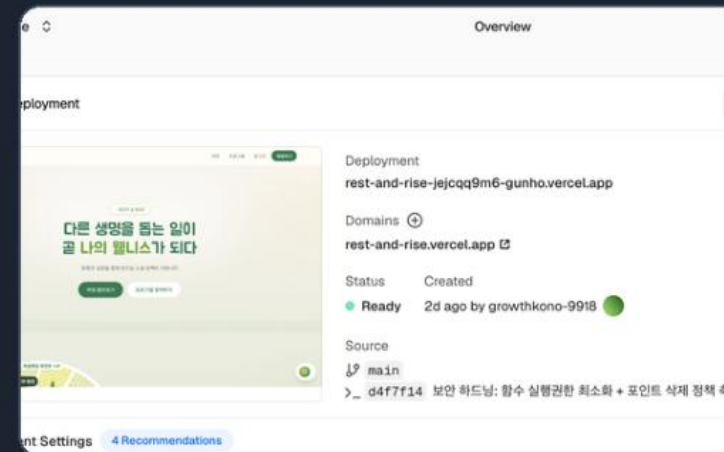
Continue with GitHub로 로그인



STEP ②

배포

Add New > Project로 깃허브 저장소를 골라 배포 · 좌측 Environment Variables에 키 값 입력



STEP ③

완료

...vercel.app 주소로 바로 접속되는 내 웹이 생깁니다

Claude Code — 터미널에서 바로 시키기



빠른 시작 문서

설치 → 실행

code.claude.com/docs/ko/quickstart · 설치 후 프로젝트 폴더에서 claude 실행

```
# macOS / Linux / WSL

curl -fsSL https://claude.ai/install.sh | bash

# Windows PowerShell

irm https://claude.ai/install.ps1 | iex

# then run in your project folder

claude
```

설치 후엔 — 막히면 claude.ai 채팅에 묻고, 실행은 claude CLI에 바로 시키면 됩니다.

결국 현장에 대한 이해와 세일즈의 영역은, 여전히 창업자의 몫



AI

빠르고 정확한 '손'



시키면 빠르고 정확하게 실행

요청한 작업을 즉시, 반복해서



내 손을 열 배 빠르게

만들고 고치는 속도를 폭발적으로



방대한 자료를 정리·생성

코드·카피·데이터를 순식간에



창업자

방향을 정하는 '머리'



현장에 대한 이해

고객이 진짜 뭘 불편해하는지 — AI는 모른다



세일즈의 영역

누구에게·왜 파는지, 사람을 설득하는 일



맥락과 판단

이 카피가 왜 안 먹히는지, 어디로 갈지

AI는 손이 빨라지게 해주지, 어디로 갈지를 대신 정해주지 않는다 — 그래서 다음 사례들도 결국 '사람이 문제를 알았기에' 성공했습니다.

초기 창업의 구조 변화 | 1인 풀스택의 등장



1인 창업이 진짜로 된다

예전엔 사람 셋이 하던 일을 한 명이 한다. 기획·개발·마케팅·운영을 혼자.



자문 외엔 스스로 해결

꼭 필요한 액셀러레이팅 자문만 받고, 나머지는 직접 부딪혀 해결한다.



ROI↑ · 러닝커브↓

비용을 낮추고, 떠먹여주는 게 아니라 스스로 빠르게 배운다.

비개발자가 AI로 만든 수익화 실사례

코딩을 모르던 사람들이, AI로 '진짜 파는 서비스'를 만들었습니다.



의사 출신 · 김솔

육아기록 앱 'Pieceful'

앱스토어 1위 · 수익화

전직 외과의. 코딩 경험 없이 클로드 코드로 육아기록 앱을 만들어 앱스토어 유료 1위.

사용 클로드 코드 · 요즘IT (2026.2)



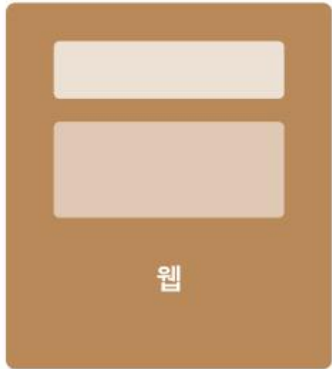
화학공학 전공 · 앱인토스

2달 만에 앱 21개

공장식 출시 · CTR 20%

코딩 0줄에서 시작. 제미나이로 아이디어, 러버블로 하루 만에 제작하는 '공장식'.

사용 Gemini · Lovable · 토스 공식 블로그 (2026.3)



작가·강사 · 신태순

하루 만에 수익형 웹

애드센스 웹사이트

개발 못 하는 베스트셀러 작가. ChatGPT로 기획, 러버블로 하루 만에 사이트.

사용 ChatGPT · Lovable · 브런치 (2025.8)



직장인 1인 개발 · 김윤후

퇴근 후 만든 앱

월 1천만 원 · DAU 2.5만

본업 있는 직장인. 노코드·AI로 앱을 만들고 데이터 분석으로 전환율을 끌어올림.

사용 노코드 · AI 도구 · 패스트캠퍼스

공통점은 스펙이 아닙니다. '내가 겪은 문제'를 정확히 알고, AI로 빠르게 만들어 시장에 던졌다는 것.

주의 — 만든 다음이 진짜

바이브코딩 | '만드는 것'은 시작일 뿐

수익화를 먼저 보든, 좋은 서사를 좇든 — 그 자체는 문제가 아닙니다. 문제는 '만들면 끝'이라고 착각하는 것.



원칙 1

MVP로 뽀족하게 검증

거대한 프로젝트를 한 번에 노리지 말 것. 초기 프로젝트(MVP) 개념으로, 뽀족한 기능 '하나씩' 명확히 검증한다.



원칙 2

PMF 기반 세일즈·비즈니스로

만든 것에서 멈추지 말 것. PMF를 확인하면 그 위에서 세일즈와 비즈니스 영역으로 밀고 나간다.

1

기본 설계 후 제작 (MVP)



2

린 스타트업 방식 기능 검증



3

PMF 달성까지 지속 반복 (크기변화)



4

세일즈,마케팅·비즈니스 확대

적용 범위 — 섹터를 가리지 않는다

적용 범위와 방법 | 섹터 무관, 출발점은 CLI



비용 절감

사람·외주 없이 직접



더 많은 가설 검증

실험 횟수를 늘린다



매우 빠른 속도 효율성

학습 속도를 올린다

그래서 뭘로 시작하나?

CLI 하나면 충분?

어떤 AI 툴이든 사용하기 나름. CLI는 선택지 중 하나

실제 컴퓨터 화면으로 따라하기 | 4단계

```
Terminal - reboon

$ cd ~/projects/reboon

# ① 작업할 폴더로 이동

$ claude

# ② 그 폴더에서 CLI를 연다

> 이 폴더에 SNS 업로드 스크립트 만들어줘

# ③ 작은 단위로 쪼개서 시킨다

✓ AI가 폴더의 파일을 읽고 코드를 작성합니다

> 슈퍼베이스에 연결해서 자동화해줘

✓ ④ 실행·수정·배포까지 한 자리에서
```

① 폴더로 이동

작업할 프로젝트 폴더로 cd. AI가 그 폴더 파일만 보고 일하니 범위가 명확하다.

② 그 폴더에서 CLI 실행

claude 한 줄이면 열린다. 설치는 한 번, 이후엔 폴더에서 바로.

③ 작은 단위로 시킨다

"결제 시스템 통째로"(X) → "API 연결 → 검증 → 웹훅"처럼 쪼개서(O).

④ 실행·수정·배포까지

AI가 코드를 쓰고 돌리고 고친다. 결과를 보고 다시 시키면 된다.

도구를 다루는 기술이 아니라, '무엇을 만들지' 설계하는 안목이 핵심입니다.

어렵게 말고, 딱 이것만 지키면 됩니다



1. 전용 폴더 하나만 정해서, 거기서 CLI를 연다

CLI가 접근할 수 있는 전용 작업 폴더를 하나 지정한다. 그 폴더 안에서만 CLI를 열어 쓰면, AI가 건드릴 수 있는 범위가 그 폴더로 제한된다. 내 컴퓨터 전체가 아니라 '이 방 안에서만' 일하게 만드는 셈. 실수로 엉뚱한 파일을 건드릴 위험이 확 줄어든다.



2. 토큰 소비·서버 호출이 잦은 설정을 조심한다

AI와 연결된 각종 어드민(제미나이·나노바나나·버셀 등)은 호출이 잦게 설정돼 있으면 토큰을 빠르게 소진한다. 자동 호출 주기나 반복 실행 옵션을 켜두면 모르는 사이 비용이 새어 나간다. 처음엔 수동·소량으로 두고, 필요한 만큼만 늘려가는 게 안전하다.



3. 개인정보는 올리지 않는다

고객 명단·연락처·실명 같은 개인정보는 AI 작업 폴더에 올리지 않는다. 꼭 다뤄야 하면 가명·샘플로 바꿔서 쓴다. 한 번 올라간 데이터가 어디까지 흘러가는지 통제하기 어렵기 때문에, 애초에 민감 정보를 작업 공간에 두지 않는 습관이 가장 확실하다.



4. 작업 히스토리를 md 파일로 남긴다

무엇을 시켰고 어떻게 고쳤는지 히스토리를 마크다운(md) 파일로 저장하게 한다. 나중에 같은 작업을 다시 하거나 문제가 생겼을 때, 이 기록이 그대로 재료가 된다. AI도 이 파일을 다시 읽어 맥락을 이어가니, 매번 처음부터 설명할 필요가 없어진다.

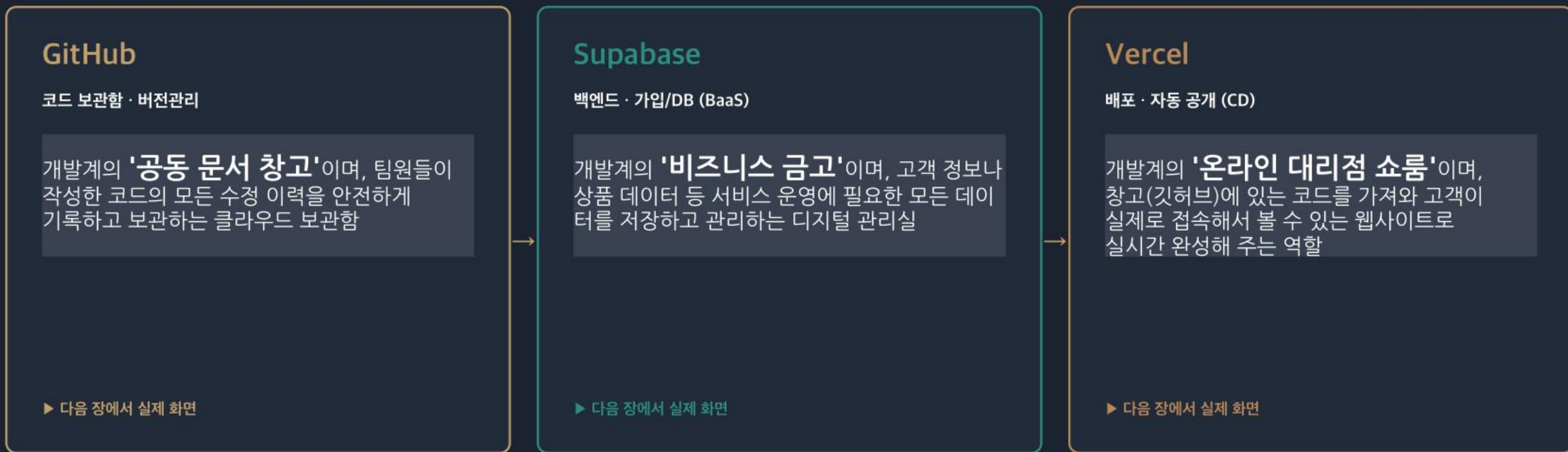


5. 폴더 안에서, 그러나 로컬 전체에 닿는 영역도 안다

기본은 전용 폴더 안에서 작업하지만, 설치·환경 설정처럼 컴퓨터 전체에 영향을 주는 작업도 있다는 걸 알아둔다. 그런 작업은 한 번 더 확인하고 실행한다. '대부분은 이 방 안, 일부는 집 전체'라는 감각만 있으면 사고를 크게 줄일 수 있다.

이 구조, 왜 GitHub·Supabase·Vercel로 나뉘나

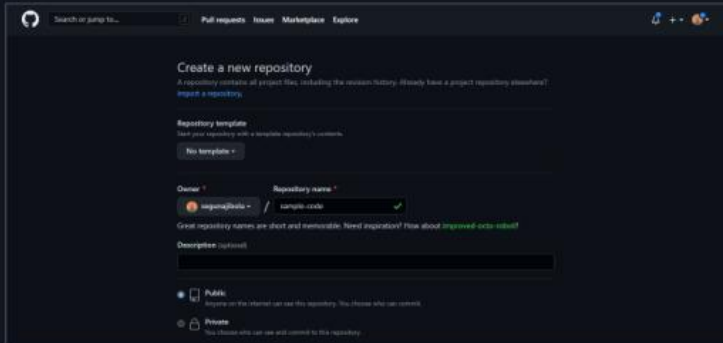
역할이 다릅니다. 코드는 GitHub에 쌓이고 → Vercel이 받아 자동 배포하고 → Supabase가 가입·데이터를 맡습니다.



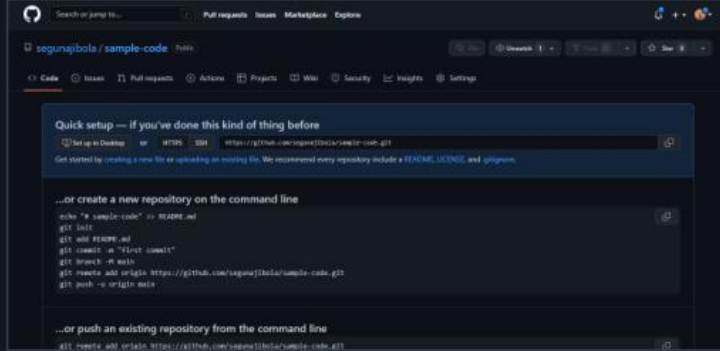
도구 이름·구조는 외울 필요 없습니다. CLI에게 '연결해줘'만 하면, 이 셋을 알아서 묶어줍니다.

다음 3장에서 GitHub → Supabase → Vercel 각각의 실제 화면을 보여드립니다.

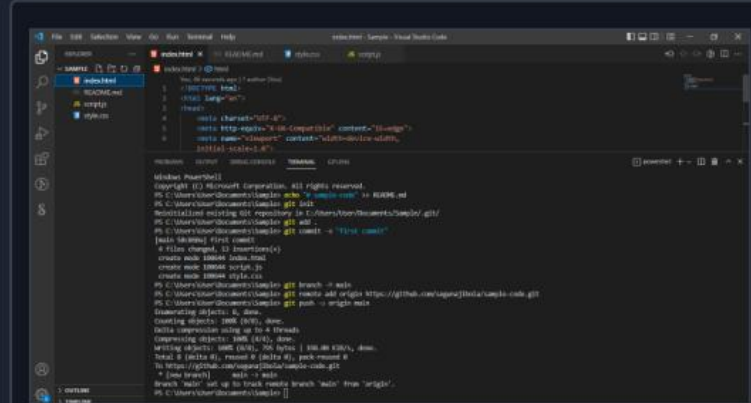
GitHub 코드 보관함 · 버전관리



① 새 저장소 만들기



② 저장소 생성 직후 안내



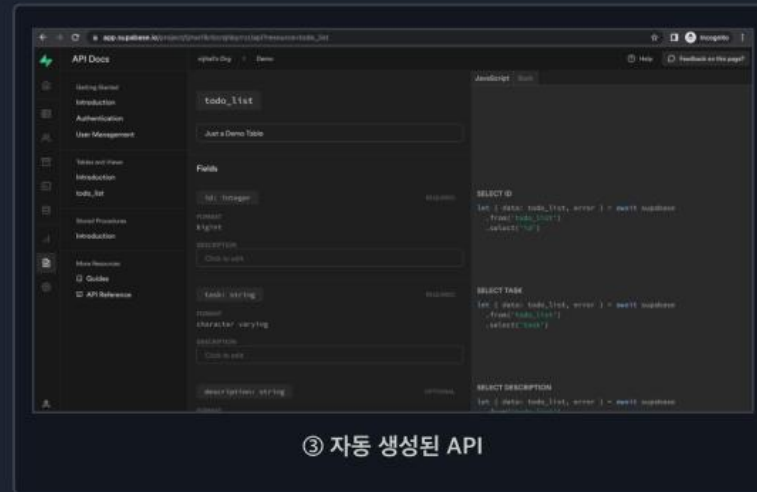
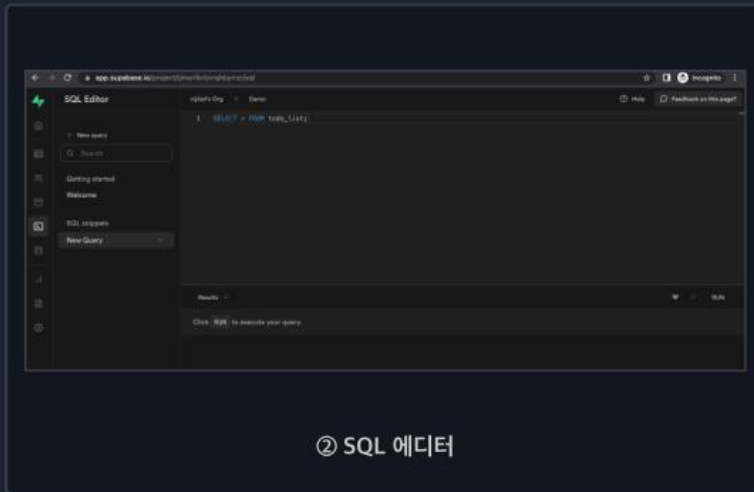
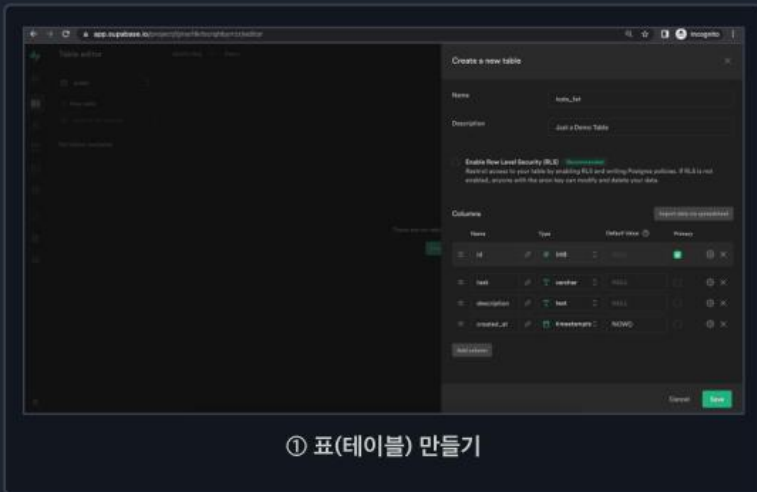
③ git push — 코드 업로드

이게 왜 필요한가

💡 내 소중한 기술 자산을 지키는 '디지털 백업 창고'

- 100% 철통 보안 (비공개 설정): 전 세계 모든 기업이 쓰는 공간입니다. 'Private(비공개)'으로 설정하면 나랑 우리 팀원 외에는 그 누구도 절대 볼 수 없는 구조
- 마이크로소프트(MS)의 초거대 서버: 내 컴퓨터나 개인 서버가 아닙니다. 세계 최고 IT 기업인 MS가 관리하는 초대형 클라우드 서버에 안전하게 보관
- 컴퓨터가 고장 나도 안심: 코드가 날아갈 걱정이 없으며, 오류가 나도 언제든 잘 작동하던 과거 버전으로 1초 만에 탄핵.
- 외주 개발사 관리의 핵심: 우리 회사 웹사이트의 '설계도'를 외주업체 PC가 아닌, 우리 회사 소유의 안전한 깃허브 계정에 묶어두는 필수 장치 역할

Supabase 백엔드 · 가입/DB (BaaS)

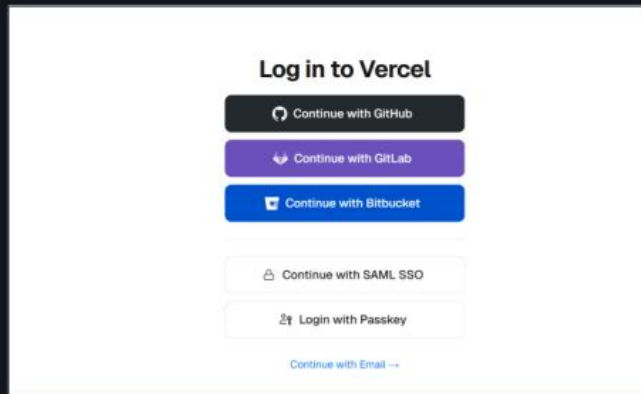


이게 왜 필요한가

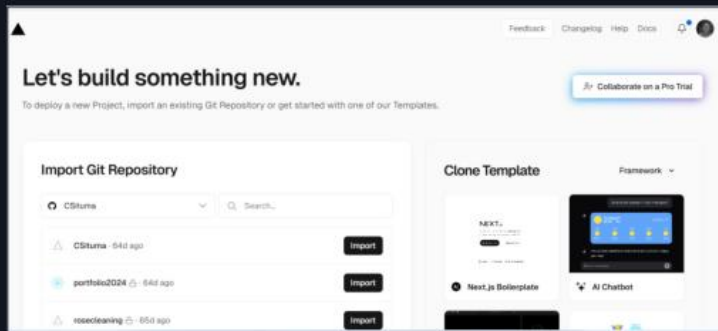
💡 보안 이슈를 원천 차단하는 '대기업급 철통 보안 금고'

- 비전문가의 보안 취약점 해결: 초보자가 코딩(바이브 코딩)으로 회원가입을 직접 만들면 해킹에 뚫리기 쉽습니다. 슈퍼베이스는 글로벌 최고 수준 암호화 보안 기술이 탑재
- 비밀번호 자동 암호화: 고객이 가입하는 즉시 비밀번호가 알아볼 수 없는 외계어 코드로 암호화되어 저장되므로, 원본 자체를 볼 수 없음
- 비즈니스 정식 출시 가능: 대충 만든 연습용 토이 프로젝트가 아닙니다. 실제 금융권과 글로벌 기업들이 검증한 표준 데이터 보안 규격을 따르므로 정식 서비스에 바로 사용
- 서버 개발 인건비 절감: 보안 전문가를 고용하려면 월 수백만 원이 들지만, 슈퍼베이스 덕분에 1인 창업자도 대기업 수준의 보안 시스템을 무료로 장착하고 시작

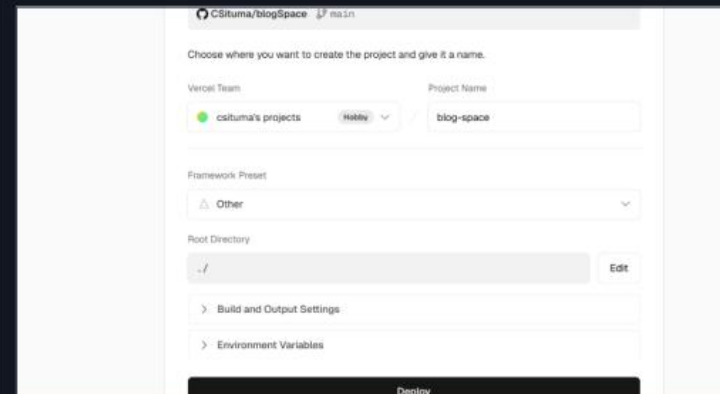
Vercel 배포 · 자동 공개 (CD)



① GitHub로 로그인



② 저장소 가져오기(Import)



③ Deploy — 배포 실행

이게 왜 필요한가

💡 클릭 한 번으로 내 가게를 오픈하는 '24시간 자동 출시 쇼룸'

- 컴퓨터 몰라도 1분 만에 웹 출시: 원래 홈페이지를 인터넷에 띄우려면 리눅스 서버를 사고, 도메인을 연결하는 등 복잡한 작업이 필요했으나 버셀은 클릭 몇 번으로 과정 해결
- 수정사항 실시간 반영: 웹사이트 오타를 고치거나 상품 이미지를 바꿀 때, 1단계 깃허브에 저장만 하면 버셀이 알아서 감시하다가 5초 만에 새 화면으로 자동 업데이트
- 전 세계에서 가장 빠른 속도: 대기업들이 쓰는 초고속 글로벌 네트워크(CDN)망을 빌려 쓰기 때문에, 고객이 전 세계 어디서 접속하든 막힘없이 화면을 빠르게 배포.
- 초기 비용 0원: 하루에 수만 명이 들어오는 대형 사이트가 되기 전까지는, 이 강력한 자동 출시 인프라를 평생 무료로 사용 가능

손이 빨라졌으니, 이제 머리를 제대로 쓸 차례

AI로 얻은 것 — 빠른 손

속도

만드는 시간이 1/10로

비용 효율

사람·외주 비용↓

더 많은 실험

테스트 횟수↑

창업자가 더해야 할 것 — 옳은 방향 (가설 → 검증 → 회고의 반복)



가설

무엇을 검증할지
'이 변화 → 이 지표' 형태로



검증

변수를 통제해 실험
숫자로 결과를 본다



회고

왜 맞고 틀렸는지
다음 가설의 재료로

↻ 반복



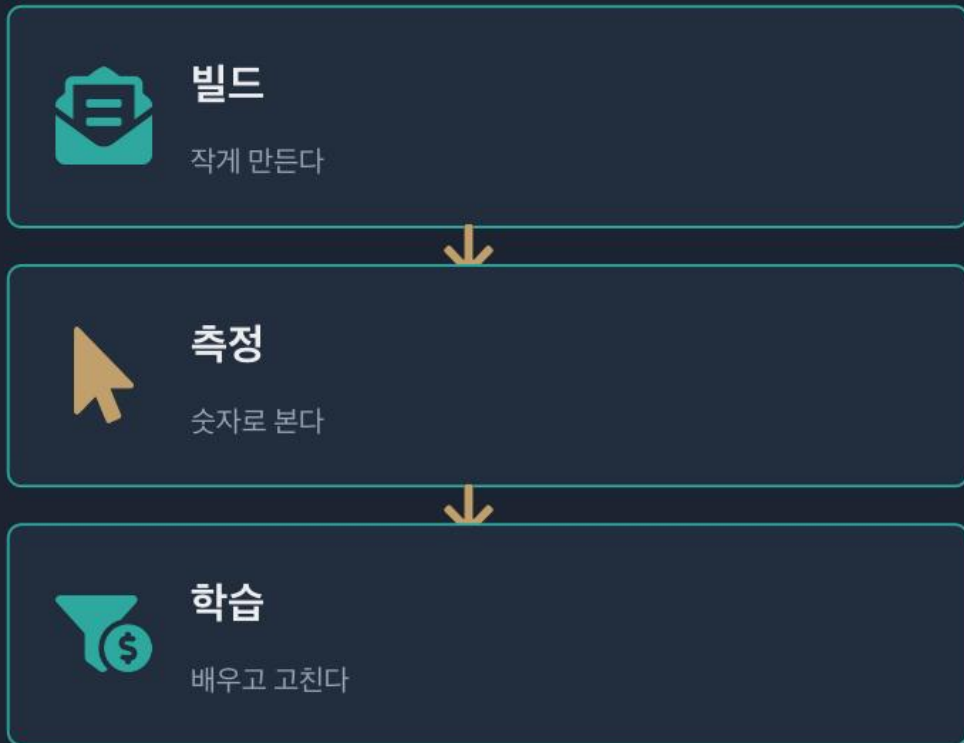
PART 2

창업자는 어떻게 일하나

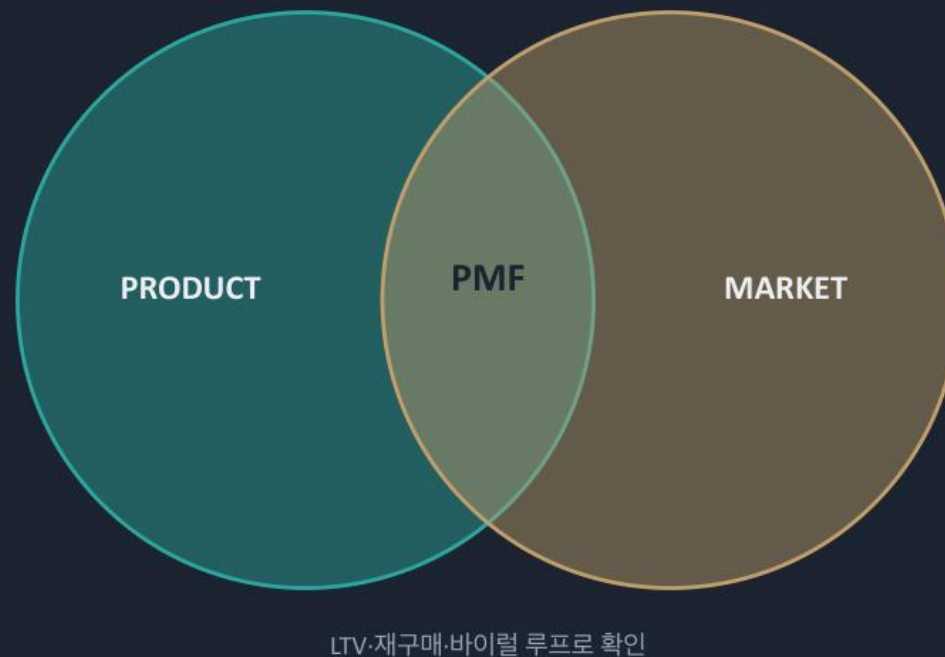
제품·서비스의 그로스해킹 — 가설을 세우고, 숫자로 검증한다

만드는 게 쉬워진 시대일수록, '무엇을 만들지'가 승부수

MVP — 빌드·측정·학습 루프



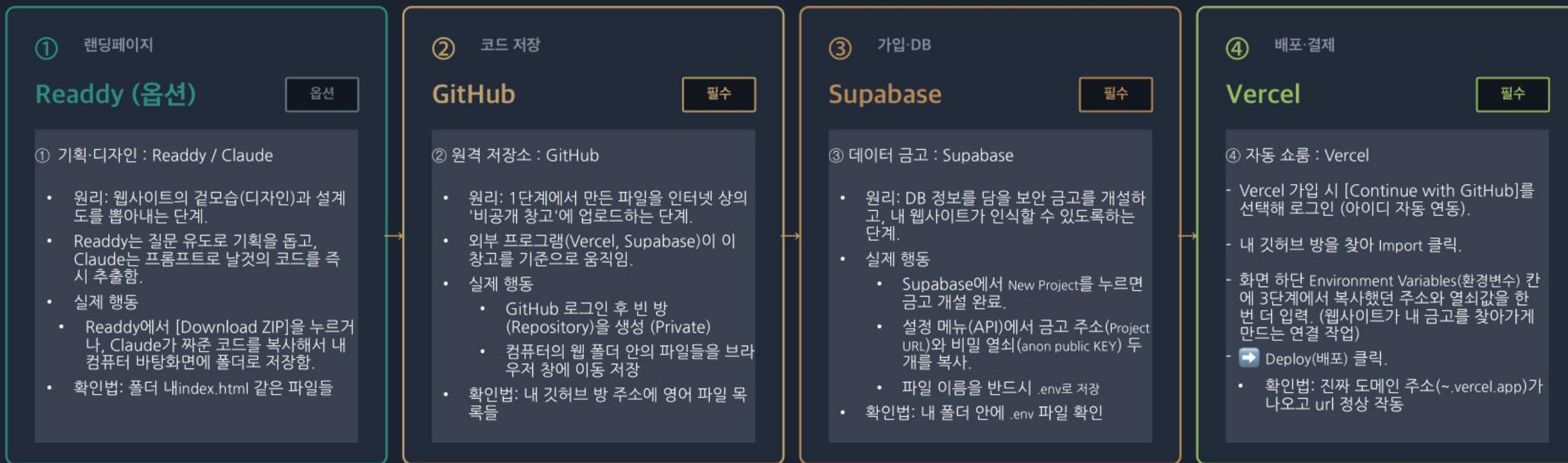
PMF — Product와 Market이 겹치는 지점



‘빠르게 만들어 → 숫자로 측정 → 배워서 고친다’를 반복하다, *Product*와 *Market*이 겹치는 *PMF*를 찾는 게 출발점입니다.

하루 안에 '가입·결제 되는' MVP

검증에 필요한 건 '완성품'이 아니라 '실험'. 네 단계가 각자 한 부분씩 말합니다. (실제 화면은 앞의 3장 참고)



핵심 흐름

Readdy(또는 CLI) → GitHub(원본) → Vercel(자동 배포) → Supabase(가입·DB) → 측정·검증도 AI로. 완성품이 아니라 '검증 도구'를 하루 만에.

결국 남은 사람의 영역 | 기획 · 세일즈



① 검증의 전제와 기준이 무너짐

기획력의 부재

A만 내놓고 빨리 피드백 받으면 될 것을, A·B·C·D를 다 넣어 출시하려다 일정이 늘어진다. 처음 검증하려던 가설 자체가 흐려진다.

예: 2주면 될 MVP가 공급 라인·시장 이슈에 끌려 3개월로



② 고객에게 USP를 전달 못함

마케팅·세일즈의 부재

제품은 있는데 '누구에게, 왜'를 못 콧는다. 공급자 관점에 갇혀 "제대로 배워야 해"라고만 외치다 수요를 놓친다.

예: 정작 고객이 원하는 건 '가볍게 따라할 인플루언서'

월가의 뱅커를 당신의 선생님으로 | AI였다면 어떻게 달랐을까?



그때의 패착

- ❌ MVP 검증에 1~2달을 더 소비
'팡 터지지 않았다'며 완성품처럼 다듬다 시간을 까먹음
- ❌ 잠재고객 딥다이브 없이 피봇
고객을 더 파보지 않고 빠르게 방향을 틀어 기준이 무너짐
- ❌ 시드를 쓰는 동안 멘탈이 흔들림
매달 비용은 나가는데 확신이 없어, 쌓이던 전환을 접어버림

AI였다면

- ✅ 검증 속도가 확 줄었다
랜딩·발송·트래킹을 하루면 세팅 — 1~2달이 며칠로
- ✅ 개발자·디자이너 인건비 절감
제작을 AI가 맡으니, 큰 팀 없이도 실험을 돌릴 수 있다
- ✅ 1~2명이 린하게 치고 나간다
기획·전략 총괄이 실무까지 빠르게 — 멘탈 소모 없이 검증 지속

MVP는 '최소 기능 제품'이지 '1년 뒤 개선할 완성품'이 아니다 — 속도가 빨라지면, 같은 실수를 안 하게 된다.

월스트리트 뉴스레터 MVP | 2주 만에 사업성 검증

'월스트리트 매거진형 뉴스레터'를 결제·개발 없이 띄워, 진짜 읽고 돈 낼 사람이 있는지부터 했습니다.



1. 콘텐츠 발송

결제·개발 없이 매거진 리포트를 주차별로 발송



2. 열람·클릭 추적

누가 열어보고 어디를 클릭하는지 전부 기록



3. 유료 전환 의사 확인

'유료라면 내시겠어요?'를 직접 묻고 의사를 수집



4. 전환율 측정

사전에 잡은 기준(x%)과 비교해 사업성 판정

THE WEEKLY

Vol.3 · 이번 주 마켓 인사이트

Feature Report

매거진형 리포트

결과: 생각 이상의 전환율



객단가 · CAC · LTV

2주 안에 전부 추정 완료



사전 지표 기준

'유료 전환 x% 이상 → 사업성 OK'

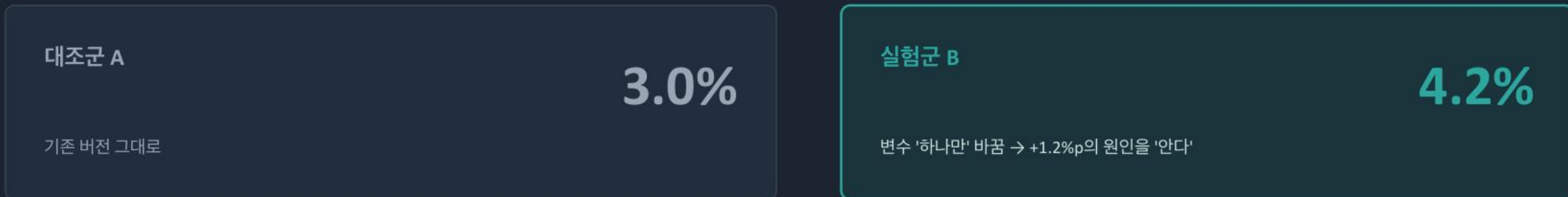


판단

숫자가 먼저 박혀 있어 '오케이/노'가 명확

그때는 노코드로 2주. 지금은 CLI로 랜딩·발송·잔존율 트래킹을 하려면 세팅이다 — '검증의 속도'가 또 빨라져요.

변수 하나만 바뀌야 원인을 안다 | A/B 검증



AI 없이 (기존)

AI로 할 때

실험 설계	A/B안을 사람이 직접 기획·분기	✓ 변형안 후보를 AI가 여러 개 생성
트래픽 분배	태깅·세팅을 수동으로	✓ 분배·태깅 코드를 AI가 자동 세팅
집계	스프레드시트로 직접 계산	✓ 로그를 AI가 수집·실시간 집계
판정	유의성 감으로 추정 — 틀리기 쉬움	✓ 유의성·표본수 계산으로 해석 보조

AI로 실험은 늘어난다 → 해석을 틀리면 더 빠르게 잘못된 방향으로. 변수 통제·유의성 판단은 여전히 창업자 몫.

가설 → 검증 → 회고 | AI가 없을 때 vs AI를 활용할 때



가설의 형태 '이 변화'를 주면 '이 지표'가 '이만큼' 움직인다, '이 근거' 때문에. (지표·근거 없으면 가설이 아니라 바람)



가설 수립

AI 없이 (기존)

경험·직관에 기대 혼자 — 후보가 적다

AI로 할 때



데이터 기반 후보를 AI가 여러 개 제시



근거 수집

과거 데이터를 일일이 찾아 붙인다



과거 실험·지표를 AI가 근거로 정리



회고

로그를 수동으로 들춰 원인 추정



로그를 AI가 분석해 원인 후보 제시



분류

무엇이 틀렸나 감으로



가설/실행/측정 중 어디서 깨졌나 정리

실패한 실험도 '왜'를 남기면 자산이 된다 — AI는 그 '왜'를 더 빠르고 촘촘하게 정리해준다.

가설·검증의 안목이 '조직의 퍼널'로 | 콘텐츠 퍼널 + ICE

지금까지가 '1인~소수'의 검증이었다면 — 같은 안목을 조직 단위 퍼널에 얹으면 이렇게 커집니다.



스쿼드 + ICE 스코어링으로 우선순위를 '숫자'로



결국 AI로 손이 빨라져도, 퍼널을 설계하고 우선순위를 숫자로 판단하는 '그로스해킹의 안목'이 조직의 성패를 가릅니다.

기본적인 툴 활용력 + 기획과 전략의 관점



AI가 준 것

속도 · 비용 · 실험량



창업자가 더한 것

가설 · 통제된 검증 · 회고



그래서

빠르고 깊은 고객집착

20곳의 스타트업, 서너 번의 창업 여정 속

좋은 사업은 결국,
AI로 빨라진 손과, 숫자를 읽는 눈과,
흔들리지 않는 사명감에서 온다고 믿습니다.

AI는 앞의 하나를 폭발적으로 키워줬지만, 뒤의 둘은 여전히 우리 몫입니다.

감사합니다

스타트업 20곳을 거쳐, AI와 함께 일하는 브랜드 창업자가 되다

E-Mail ceo@h-grs.com

