

바이브코딩으로 만들었는데, 왜 아무도 안 쓸까

1인 개발자를 위한 첫 사용자 확보 실전 매뉴얼

지은이 호현수 (CoreLabs)

대상 만드는 건 끝냈는데 사용자가 0명인 1인 개발자

분량 본문 10장 + 복불 템플릿 12종 + 실전 케이스 스터디

버전 2026년 8월 1판

이 책을 사기 전에 읽어주세요

이 책은 코딩을 가르치지 않습니다.

Claude Code, 커서, 코덱스로 서비스 하나 똑딱 만드는 법을 알려주는 자료는 이미 유튜브에 널려 있습니다. 무료요. 그런데 그 영상들이 전부 끝나는 지점이 어디인지 아십니까?

"자, 배포까지 완료됐습니다. 여러분도 도전해보세요!"

거기서 끝납니다. 그리고 진짜 지옥은 **그 다음 날 아침**에 시작됩니다.

어제 배포한 사이트에 접속자 수를 확인합니다. 3명입니다. 그중 2명은 본인이고, 1명은 봇입니다. 일주일 뒤에도 똑같습니다. 한 달 뒤에는 확인하는 것 자체를 그만두게 됩니다.

이 책은 **정확히 그 다음 날 아침부터** 시작합니다.

이 책이 맞는 사람

- 서비스를 배포했는데 가입자가 한 자릿수인 사람
- 마케팅 예산이 사실상 0원인 사람
- "홍보 좀 해야지"라고 3개월째 생각만 하고 있는 사람
- 광고비 태우는 법 말고, 몸으로 때우는 법을 알고 싶은 사람

이 책이 안 맞는 사람

- 아직 만들 게 안 정해진 사람 (이 책은 만들어진 다음의 이야기입니다)
- 광고비 월 300만 원 이상 태울 수 있는 사람 (그러면 그냥 광고 돌리세요, 그게 빠릅니다)
- "대박 나는 법"을 찾는 사람 (여기엔 대박이 없습니다. 노가다만 있습니다)

이 책의 약속

이 책에 나오는 모든 방법은 **제가 직접 실행하고, 결과를 세어본 것만** 담았습니다.

"~하면 좋다더라"는 한 줄도 없습니다. 실패한 방법은 실패했다고 쓰고, 응답률이 10%면 10%라고 씁니다.

읽고 나서 "그래서 오늘 뭘 하면 되지?"에 답할 수 없다면, 그건 제 잘못입니다.

1장. 진단: 당신의 서비스가 조용한 진짜 이유

1-1. 병목은 이미 이동했다

3년 전까지 1인 개발의 병목은 명확했습니다. **만들 줄 아느냐**였습니다.

아이디어가 있어도 못 만들었습니다. 프론트엔드 배우고, 백엔드 배우고, 배포 배우다가 1년이 갔습니다. 그래서 그 시절엔 "완성해서 배포까지 했다"는 사실 자체가 희소성이었습니다. 만들기만 하면 최소한 개발자 커뮤니티에서는 관심을 받았습니다.

지금은 어떻습니까.

주말 이틀이면 결제 붙은 서비스가 나옵니다. AI에게 시키면 됩니다. 진입장벽이 무너졌다는 건 좋은 소식처럼 들리지만, 정확히 같은 크기의 나쁜 소식이 따라옵니다.

만드는 비용이 0에 수렴하면, 만들어진 것의 가치도 0에 수렴합니다.

당신이 주말에 만들었다면, 다른 3,000명도 그 주말에 만들었습니다. "완성했다"는 더 이상 아무런 신호가 아닙니다. 병목은 **제작에서 유통으로 완전히 이동**했습니다.

그런데 대부분의 1인 개발자는 여전히 제작 시대의 습관을 갖고 있습니다. 시간의 95%를 만드는 데 쓰고 5%를 알리는 데 씁니다. 병목이 옮겨갔는데 자원 배분은 그대로인 겁니다.

이 책의 첫 번째 요구는 이겁니다. **비율을 뒤집으세요.** 이미 만들었다면, 지금부터는 시간의 80%를 유통에 쓰셔야 합니다.

1-2. "좋은 제품은 알아서 알려진다"는 거짓말

이 말을 들어보셨을 겁니다. 성공한 창업자들이 인터뷰에서 자주 합니다.

"저희는 마케팅을 따로 안 했어요.
제품이 좋으니까 사람들이 알아서 찾아오더라고요."

이 문장은 **생존자 편향의 교과서적 사례**입니다.

이 말을 하는 사람은 이미 성공한 사람입니다. 성공했기 때문에 인터뷰를 하고 있고, 성공했기 때문에 그 말이 기억됩니다. 같은 시기에 똑같이 좋은 제품을 만들었지만 아무도 몰라서 사라진 수천 개의 서비스는 인터뷰를 하지 않습니다. 그들이 침묵하기 때문에, 우리는 "좋으면 알려진다"는 명제의 반례를 볼 기회 자체가 없습니다.

더 중요한 건 이겁니다. **그들도 사실은 초기에 몸으로 때웠습니다.** 다만 기억하지 못하거나, 인터뷰에서 말할 만큼 맛있는 이야기가 아니라서 생략할 뿐입니다.

- 에어비앤비 창업자들은 초기에 뉴욕 집주인들 집에 직접 찾아가서 사진을 찍어줬습니다.
- 초기 스타트업 대부분은 창업자가 직접 한 명씩 연락해서 첫 100명을 만들었습니다.

멋없죠. 그래서 인터뷰에 안 나옵니다. 하지만 그게 실제로 일어난 일입니다.

결론: 알아서 알려지는 일은 일어나지 않습니다. 당신이 알려야 합니다.

1-3. 콜드스타트 문제 — 특히 당신 서비스가 양면 시장이라면

여기서 상황을 정확히 진단해야 합니다. 당신의 서비스는 둘 중 어느 쪽입니까?

단면(1-sided) 서비스

사용자 한 명이 혼자 써도 가치가 있습니다. 메모 앱, 이미지 변환기, 가계부 같은 것들입니다.

→ 콜드스타트 난이도: **중**

양면(2-sided) 서비스

공급자와 수요자가 둘 다 있어야 굴러갑니다. 중고거래, 매칭 플랫폼, 마켓플레이스입니다.

→ 콜드스타트 난이도: **극악**

양면 시장의 지옥은 이렇습니다. 판매자에게 가면 "구매자 있어요?"라고 묻고, 구매자에게 가면 "물건 있어요?"라고 묻습니다. 양쪽 다 상대방이 먼저 있어야 들어오겠다고 합니다. 이걸 **닭과 달걀 문제**라고 하는데, 1인 개발자에게는 사실상 사망 선고처럼 느껴집니다.

해법은 하나뿐입니다.

한쪽을 먼저, 수동으로, 무리해서 채웁니다.

양쪽을 동시에 키우려는 시도는 100% 실패합니다. 자원이 반으로 갈리고 양쪽 다 임계점을 못 넘기기 때문입니다.

어느 쪽을 먼저 채워야 하는가?

"더 아쉬운 쪽"이 아니라 "**더 설득하기 쉬운 쪽**"을 먼저 채웁니다. 보통은 공급자 쪽입니다. 공급자는 잃을 게 없는 경우가 많고(어차피 놓고 있는 자산), 수요자는 볼 게 없으면 바로 이탈하기 때문입니다.

1-4. 자가 진단 체크리스트

다음 항목에 정직하게 답해보십시오. 3개 이상 해당되면 이 책의 나머지가 당신에게 필요합니다.

#	항목	Y/N
1	서비스를 배포한 지 2주 이상 지났다	
2	누적 가입자가 30명 미만이다	
3	지난 일주일간 "알리기 위한 행동"을 한 번도 안 했다	
4	홍보라고 한 것이 커뮤니티 게시물 1~2개가 전부다	
5	"누구에게 이걸 팔아야 하는지" 구체적인 사람 이름을 5명 못 낸다	
6	제품을 더 다듬으면 사람이 올 거라고 생각하고 있다	

특히 **6번에 Y**라고 답하셨다면 위험합니다. 그건 유통이 무서워서 익숙한 제작으로 도망치는 중이라는 신호입니다. 개발자에게 가장 흔한 회피 패턴입니다.

2장. 발상의 전환: 사용자는 모으는 게 아니라 찾아가는 것

2-1. 수동적 채널이 안 먹히는 구조적 이유

첫 사용자를 구하려는 1인 개발자가 가장 먼저 하는 행동은 정해져 있습니다.

커뮤니티에 글을 씁니다.

개발자 커뮤니티, 창업 카페, 오픈채팅방, 관련 서브레딧에 "제가 이런 걸 만들었는데 한번 봐주세요"라고 올립니다. 그리고 조회수 40, 댓글 1개("고생하셨습니다")를 받고 좌절합니다.

왜 안 될까요? 열심히 안 써서가 아닙니다. **구조적으로 안 되게 되어 있습니다.**

제가 실제로 이 함정에 빠졌다가 발견한 사실입니다. 제가 만든 서비스는 **방치된 사이드프로젝트를 사고파는 마켓플레이스**였습니다. 그러니 제 타겟은 명확합니다. "프로젝트를 만들었다가 조용히 접은 사람"입니다.

그래서 저는 바이브코딩 커뮤니티들을 뒤졌습니다. 거기에 개발자들이 모여 있으니까요. 그런데 결과가 처참했습니다. 한참을 헤매다가 이유를 깨달았습니다.

커뮤니티에 글을 쓸 만큼 활동적인 사람은, 정의상 "조용히 포기한 사람"이 아닙니다.

이건 제 서비스만의 문제가 아닙니다. 훨씬 일반적인 원리입니다.

커뮤니티는 "그 주제에 대해 아직 열정이 남아 있는 사람"만 필터링해서 모아둔 곳입니다.

- 다이어트 앱을 만들었다면? 다이어트 커뮤니티에는 이미 다이어트를 잘하고 있는 사람이 모여 있습니다. 정작 당신 앱이 필요한 사람은 포기하고 커뮤니티를 나갔습니다.
- 공부 관리 앱을 만들었다면? 공부 카페에는 이미 공부 잘하는 사람이 있습니다.
- 재테크 도구를 만들었다면? 재테크 카페 사람들은 이미 엑셀로 다 관리하고 있습니다.

커뮤니티는 **문제를 이미 어느 정도 해결한 사람들의 집합**입니다. 당신의 제품이 필요한 사람은 대체로 **문제 앞에서 조용해진 사람**이고, 조용해진 사람은 커뮤니티에 없습니다.

커뮤니티 게시글은 비유하자면, 이미 밥을 먹은 사람들이 모인 식당 앞에서 전단지를 돌리는 일입니다.

2-2. 그렇다면 커뮤니티 글은 쓰지 말아야 하나?

아닙니다. 다만 기대값을 정확히 알고 쓰셔야 합니다.

커뮤니티 게시글의 진짜 가치는 즉시 유입이 아니라 검색 자산입니다.

- 오늘 그 글을 읽는 사람: 30~50명, 그중 가입 0~1명
- 6개월 뒤 구글에서 "○○ 서비스 후기" 검색해서 그 글에 도착하는 사람: 누적으로 수백 명 가능

즉 커뮤니티 글은 **오늘의 유입 채널이 아니라 6개월 뒤의 SEO 자산**입니다. 그렇게 이해하면 쓸 가치가 충분합니다. 다만 이것 "홍보를 했다"고 착각하고 그 자리에 멈추면 안 됩니다.

정리

| 채널 유형 | 즉시 전환율 | 장기 가치 | 역할 |

| ---|---|---|---|

| 커뮤니티 게시글 | 매우 낮음 | 높음(SEO) | 자산 축적 |

| 1:1 직접 접촉 | 높음 | 낮음 | 초기 사용자 확보 |

첫 100명은 반드시 1:1로 만듭니다. 이건 선택이 아닙니다.

2-3. 핵심 원리: "흔적 역추적"

그럼 1:1로 누구에게 가야 할까요? 여기가 이 책에서 가장 중요한 부분입니다.

대부분의 사람은 타겟을 이렇게 정의합니다.

"20~30대 개발자, 사이드프로젝트에 관심 있는 사람"

이건 타겟이 아니라 **인구통계**입니다. 이걸로는 아무에게도 연락할 수 없습니다. 이름도 연락처도 없으니까요.

실행 가능한 타겟 정의는 이런 형태여야 합니다.

"GitHub에 README를 한국어로 쓰고, 그 안에 '운영을 종료했습니다'라는 문장을 적어둔, 컨트리뷰터가 1명뿐인 저장소의 주인"

차이가 보이십니까? 두 번째 정의는 **오늘 당장 검색해서 명단을 뽑을 수 있습니다.**

이 전환을 만드는 질문이 하나 있습니다. 이 질문을 이 책에서 딱 하나만 가져가신다면 이걸 가져가십시오.

 "내 타겟은, 자신이 내 타겟이라는 사실을 어디에 공개적으로 남겨두는가?"

사람들은 자기 상태를 의외로 많이 공개합니다. 그것도 **검색 가능한 텍스트로** 남깁니다.

당신의 타겟	그들이 남기는 공개 흔적	검색 방법
프로젝트를 접은 개발자	README의 "서비스 종료" 문구	GitHub 코드 검색
이직을 고민 중인 사람	링크드인 프로필 "구직 중" 배지	링크드인 필터
특정 도구에 불만인 사람	"○○ 대안 없나요" 게시글	커뮤니티/X 검색
혼자 일하는 프리랜서	크몽/숨고 프로필	플랫폼 내 검색
자동화가 필요한 사람	"○○ 수작업 너무 힘들다" 트윗	X 키워드 검색
특정 앱을 쓰다 지친 사람	앱스토어 별점 1~2점 리뷰	스토어 리뷰 정렬

당신의 타겟이 남기는 흔적을 찾으면, 그때부터 마케팅은 **창의력의 영역이 아니라 검색과 반복의 영역**이 됩니다. 그리고 후자가 훨씬 쉽습니다.

2-4. 흔적을 찾는 3단계 워크시트

이 작업을 지금 15분 안에 해보십시오.

STEP 1. 타겟이 겪는 "상태"를 한 문장으로 쓴다

- 나쁜 예: "생산성을 높이고 싶은 직장인"
- 좋은 예: "엑셀로 재고를 관리하다가 실수로 데이터를 날려본 소상공인"

STEP 2. 그 상태에 있는 사람이 무심코 남길 문장을 5개 상상해서 적는다

- "엑셀 재고관리 파일 날렸어요"
- "재고 프로그램 추천 좀"
- "수기로 관리하다 한계가 왔습니다"

STEP 3. 그 문장이 실제로 존재하는 플랫폼에서 검색해본다

- 네이버 카페, X, GitHub, 앱스토어 리뷰, 유튜브 댓글, 블로그

STEP 3에서 검색 결과가 나오면, 그게 당신의 첫 사용자 명단입니다. 이제 연락만 하면 됩니다.

3장. 실전 채널 ①: GitHub README 고고학

적합한 서비스: 개발자 대상 서비스, 프로젝트 관련 서비스

난이도: 중 / 비용: 0원 / 1건당 소요시간: 약 10분

실측 데이터: 후보 40여 건 탐색 → 유효 5건 → 발송 5건

3-1. 왜 GitHub인가

개발자를 타겟으로 한다면 GitHub는 세상에서 가장 정직한 데이터베이스입니다.

이유는 단순합니다. 사람들은 커뮤니티에서는 자기를 포장하지만, README에는 사실을 적습니다. 프로젝트를 접었으면 접었다고 씁니다. 그것도 날짜까지 박아서요.

"2026년 6월 29일부로 운영을 종료했습니다."
"베타테스트 결과 사용성이 너무 낮아서 중지했습니다."
"더 이상 유지보수하지 않습니다."

이건 객관적 증거입니다. 추측이 아닙니다. 커뮤니티 활동량으로 "이 사람 프로젝트 접은 것 같은데?"라고 추측하는 것과는 차원이 다릅니다.

3-2. 실전 검색 명령어

GitHub CLI(`gh`)를 쓰면 가장 빠릅니다. 설치가 부담되면 웹 검색창에서도 됩니다.

CLI 방식

```
gh search repos "운영을 종료했습니다" --match readme --limit 50
```

```
gh search repos "서비스가 종료되었습니다" --match readme --limit 50
```

```
gh search repos "더 이상 유지보수하지" --match readme --limit 50
```

웹 방식

GitHub 검색창에 입력합니다.

```
"운영을 종료했습니다" language:Korean
```

검색어 품질 등급 (제가 실제로 돌려본 결과입니다)

검색어	결과 수	유효율	등급
"중지했습니다"	적음	높음	☆☆☆
"운영을 종료했습니다"	적음	높음	☆☆☆
"서비스가 종료되었습니다"	중간	중간	☆☆
"1인 개발"	많음	매우 낮음	☆
"사이드프로젝트"	매우 많음	매우 낮음	☆

교훈: 넓은 키워드는 부트캠프 과제와 튜토리얼 저장소만 잔뜩 나옵니다. 구체적인 "고백 문장"을 노리십시오.

3-3. ⚠ 반드시 피해야 할 오탐 4종

여기서 제가 시간을 크게 낭비했습니다. 검색 결과 스니펫만 보고 판단하면 **절반이 오탐**입니다. 반드시 README 전문을 읽으십시오.

오탐 유형 1: 이사 (가장 흔함)

"이 저장소는 더 이상 사용하지 않습니다 → 새 모노레포로 이전했습니다"

→ 프로젝트는 **멀쩡히 살아 있습니다**. 주소만 옮긴 겁니다. 여기에 연락하면 무례한 사람이 됩니다.

오탐 유형 2: 부분 폐기

"이 폴더의 레거시 코드는 폐기되었습니다"

→ 프로젝트 전체가 아니라 **구성요소 하나만** 죽었습니다.

오탐 유형 3: 구버전 스냅샷

v1 저장소는 종료, v2 저장소에서 활발히 개발 중.

오답 유형 4: 무관한 문구

"API 키가 만료되었습니다" 같은 문장이 검색에 걸린 경우.

검증 규칙

"이 프로젝트 전체가, 다른 어디에서도 이어지지 않고, 실제로 멈췄는가?"

이 질문에 확신을 갖고 Yes라고 답할 수 있을 때만 연락하십시오.

3-4. 🚨 아카이브 함정 (반드시 먼저 확인)

이건 제가 실제로 부딪힌 벽입니다. README 내용이 완벽하게 맞아떨어지는 후보를 찾아서 이슈를 작성했는데, 이런 에러가 났습니다.

```
Repository was archived so is read-only.
```

GitHub에서 저장소를 **아카이브** 처리하면 읽기 전용이 되어 **이슈를 달 수 없습니다**. 아이러니하게도 "확실하게 접은 사람"일수록 아카이브를 해둡니다. 즉 조건이 완벽한 후보일수록 연락이 불가능한 경우가 생깁니다.

연락 시도 전 1초 만에 확인하는 법

```
gh api repos/OWNER/REPO --jq .archived
```

`true`가 나오면 포기하고 다음 후보로 넘어가십시오. (또는 프로필의 이메일/블로그로 우회)

3-5. 후보 선별 기준표

기준	확인 방법	왜 중요한가
✅ 솔로 프로젝트인가	컨트리뷰터 1명	여러 명이면 동의 문제 발생
✅ 아카이브 안 됐는가	<code>.archived == false</code>	아카이브면 접촉 불가
✅ 라이브 데모가 있(었)는가	<code>homepage</code> 필드	실제 서비스였다는 증거
✅ README에 명시적 중단 문구	전문 읽기	추측 금지
⚠️ 중단 시점이 최근인가	커밋 로그	5년 전이면 응답률 급락
⚠️ 다른 저장소는 활발한가	프로필	활동 중이면 응답 확률 ↑

마지막 항목이 의외로 중요합니다. 완전히 GitHub를 떠난 사람은 알림도 안 보니다. "다른 건 계속 만들고 있는데 이 프로젝트만 접은 사람"이 최고의 후보입니다. 살아 있고, 만드는 걸 좋아하고, 이 프로젝트에 미련이 남아 있으니까요.

3-6. 접촉 방법: GitHub Issue

이메일 주소를 공개 안 한 사람이 많습니다. 하지만 **Issue**는 누구나 열 수 있고, 알림이 이메일로 갑니다.

```
gh issue create --repo OWNER/REPO --title "제목" --body "본문"
```

⚠️ 매너 경고

Issue는 공개됩니다. 그 사람의 저장소에 영구히 남습니다. 그러니 **스팸처럼 쓰면 안 됩니다**. 아래 원칙을 지키십시오.

1. 반드시 **그 프로젝트를 실제로 읽었다는 증거**를 첫 문단에 넣습니다
2. 제목에 광고 티를 내지 않습니다
3. 링크는 본문에 딱 1개만 넣습니다
4. **거절이 쉽게** 마무리합니다 ("관심 없으시면 닫아주세요")

템플릿은 부록 T-03에 있습니다.

4장. 실전 채널 ②: 인플루언서 콜드메일

적합한 서비스: 전 분야

난이도: 하 / 비용: 0원

실측 데이터: 10명 발송 → 열람 다수 → 답장 1명 (응답률 10%)

4-1. 응답률 10%를 어떻게 볼 것인가

먼저 기대치를 정확히 세팅하겠습니다.

저는 관련 분야 유튜버 10명에게 콜드메일을 보냈고, **1명에게 답장을 받았습니다.**

이 숫자를 보고 "10%밖에 안 되네"라고 생각하셨다면, 관점을 바꾸셔야 합니다. 콜드메일 업계에서 **응답률 10%는 상당히 좋은 편**입니다. B2B 콜드메일 평균 응답률은 1~5% 수준입니다.

그리고 더 중요한 건 이겁니다.

10명 보내서 1명 답장 = 100명 보내면 10명 답장.

이건 창의력이 아니라 **산수**입니다.

콜드메일의 최대 장점이 여기 있습니다. **재현 가능하고 선형적으로 확장됩니다.** 오늘 잘 안 됐어도 내일 10명 더 보내면 됩니다. 커뮤니티 글은 이게 안 됩니다.

4-2. 답장을 받았다고 일이 되는 건 아니다 (중요)

여기서 정신적으로 준비하셔야 할 부분이 있습니다.

제가 받은 답장은 **거절도 수락도 아닌, 애매한 답장**이었습니다. 요지는 이랬습니다.

"좋은 프로젝트는 알아서 사람이 모입니다."

이런 답장을 받으면 하루 종일 기분이 가라앉습니다. "내 서비스가 홍보할 가치도 없다는 뜻인가?"라는 해석으로 넘어가기 쉽습니다.

그렇게 해석하지 마십시오. 두 가지 이유가 있습니다.

첫째, 그 답장은 당신 제품에 대한 평가가 아니라 그 사람의 기본 정책입니다.

그런 위치에 있는 사람은 매주 수십 개의 비슷한 제안을 받습니다. 하나하나 진지하게 검토할 시간이 없기 때문에 **모두에게 쓰는 정중한 기본 답변**을 갖고 있습니다. 당신 제품을 뜯어보고 판단한 게 아닙니다.

둘째, 답장을 받았다는 것 자체가 성과입니다.

9명은 답장조차 안 했습니다. 답장이 왔다는 건 **최소한 읽혔고, 무시할 수준은 아니었다**는 뜻입니다.

콜드 아웃리치의 심리 규칙

거절과 무응답을 데이터로 처리하십시오. 감정으로 처리하면 3일 안에 그만두게 됩니다.

스프레드시트에 한 줄 적고 다음으로 넘어가는 겁니다. 그게 전부입니다.

4-3. 답장에서 진짜 건져야 할 것

애매한 답장이라도 **거의 항상 힌트가 하나씩 들어 있습니다.**

제가 받은 답장에도 스킷치 언급된 정보가 있었습니다. "이런 종류의 서비스를 소개하는 플랫폼이 있다"는 언급이었습니다. 저는 그 언급을 붙잡고 해당 플랫폼을 찾아가서 제 서비스를 등록했습니다.

즉, 그 콜드메일의 실제 성과는 "**홍보 성사**"가 아니라 "**새로운 채널 발견**"이었습니다.

이게 콜드 아웃리치의 숨은 가치입니다. 목표는 A였는데 B를 얻는 경우가 많습니다.

답장에서 반드시 뽑아야 할 3가지

1. 언급된 **플랫폼/커뮤니티 이름** → 즉시 방문해서 등록
2. 언급된 **다른 사람 이름** → 다음 접촉 대상
3. 지적된 **제품의 약점** → 무료 컨설팅으로 취급

4-4. 콜드메일 5원칙

원칙 1. 제목에 부탁을 넣지 않는다

- ❌ "제 서비스 홍보 부탁드립니다"
- ⓪ "○○ 영상 보고 만든 서비스가 있어서 연락드립니다"

원칙 2. 첫 문장에서 "당신을 알고 있다"를 증명한다

그 사람의 특정 콘텐츠 하나를 구체적으로 언급합니다. "항상 잘 보고 있습니다"는 아무 증명도 안 됩니다. "지난달 ○○편에서 말씀하신 △△ 부분이..."라고 써야 합니다.

원칙 3. 요구사항을 하나만, 아주 작게 만든다

- ✕ "영상으로 소개해주실 수 있나요?" (큰 부탁)
- ○ "한번 써보시고 별로면 별로라고 한 줄만 주셔도 감사하겠습니다" (작은 부탁)

원칙 4. 거절 경로를 열어둔다

"관심 없으시면 답장 안 주셔도 전혀 괜찮습니다." → 역설적으로 답장률이 올라갑니다. 부담이 사라지기 때문입니다.

원칙 5. 길이는 스마트폰 한 화면

스크롤이 생기는 순간 읽히지 않습니다. 7문장 이내를 목표로 하십시오.

템플릿은 부록 T-01에 있습니다.

4-5. 접촉 대상 명단 만들기

한 번에 10명씩, 리스트로 관리합니다.

이름	채널/소속	접촉 경로	발송일	열람	답장	비고

열람 여부 확인법: Gmail 사용 시 메일 추적 확장 프로그램을 쓰면 열람 여부가 보입니다. **열람은 했는데 답장이 없다** = 제목은 통했고 본문이 약했다는 뜻입니다. 본문을 고쳐서 다음 10명에게 보내십시오.

5장. 실전 채널 ③④⑤: 나머지 세 갈래

5-1. 채널 ③ — 제품 전시 플랫폼

각 나라마다 "내가 만든 것 자랑하는 사이트"가 있습니다. 한국에도 여러 개 생겼고, 해외에는 규모 있는 곳들이 있습니다.

장점

- 등록이 무료이고 5분이면 끝납니다
- 방문자가 애초에 "새로운 서비스를 보러 온 사람"입니다 (커뮤니티와 결정적 차이)
- 검색 노출 자산이 됩니다

단점

- 등록 후 아무것도 안 하면 하루 만에 묻힙니다

⚠️ **실전 주의사항 3가지** (제가 실제로 겪은 것들)

1) 게시판 잘못 고르지 마십시오.

"자료실"과 "커뮤니티"처럼 성격이 다른 게시판이 있는 경우, 홍보물을 자료실에 올리면 아무도 안 봅니다. **사람이 실제로 머무는 곳에** 올려야 합니다.

2) 썸네일 없이 올리지 마십시오.

텍스트만 있는 게시물은 클릭률이 처참합니다. 서비스 화면 캡처 3~4장이면 충분합니다. 포토샵 필요 없습니다.

3) 에디터 마크업이 새는지 확인하십시오.

저는 등록 후 실제 페이지를 스크롤하다가 `{{color:blue}}` 같은 마크업 코드가 그대로 노출된 걸 발견했습니다. 에디터에서는 정상으로 보였는데 게시된 페이지에서는 깨진 겁니다.

철칙: 등록 후 반드시 "로그아웃 상태로" 자기 페이지를 처음부터 끝까지 스크롤해서 읽어보십시오.

작성자 화면과 방문자 화면은 다릅니다.

5-2. 채널 ④ — 리뷰/불만 역추적

앱스토어와 플레이스토어의 **별점 1~2점 리뷰**는 금광입니다.

경쟁 앱의 낮은 별점 리뷰를 읽으면 **불만을 가진 실명 사용자 명단**이 나옵니다. 그들은 이미:

- 그 문제를 인지하고 있고
- 돈이나 시간을 써서 해결하려 시도했고
- 실패해서 화가 나 있습니다

이보다 좋은 잠재 고객은 없습니다.

단, 여기엔 명확한 선이 있습니다.

- ○ 리뷰를 읽고 **내 제품의 소개 문구를 그들의 언어로 다시 쓰는 것**
- ○ 그 불만이 자주 나오는 커뮤니티를 역추적해서 그쪽에 글을 쓰는 것

- ❌ 경쟁사 리뷰란에 답글로 내 서비스를 홍보하는 것 (거의 모든 플랫폼 규정 위반이며 계정이 정지됩니다)

5-3. 채널 ⑤ — 영상 플랫폼 댓글 (⚠️ 위험 채널)

관련 영상 댓글에 자연스럽게 서비스를 언급하는 방법입니다. 효과는 있지만 매우 위험합니다.

제가 직접 확인한 사실입니다.

여러 영상에 짧은 간격으로 같은 브랜드명을 언급하는 댓글을 달면, 플랫폼이 자동으로 삭제합니다. 알림도 없이, 조용히 사라집니다.

더 고약한 건 **본인 계정으로 보면 댓글이 그대로 보인다**는 점입니다. 로그아웃하거나 다른 계정으로 보면 없습니다. 그래서 며칠 동안 열심히 댓글을 달면서 "왜 반응이 없지?"라고 생각하게 됩니다. 애초에 아무에게도 안 보이고 있었던 겁니다.

생존 규칙

1. 하루 1~2개를 넘기지 마십시오
2. 같은 문장을 복사하지 마십시오 — 매번 그 영상 내용에 맞춰 다시 쓰십시오
3. 브랜드명 없이 **가치만** 언급하는 댓글도 섞으십시오
4. 주기적으로 시크릿 창에서 자기 댓글이 보이는지 확인하십시오

솔직히 말씀드리면, 이 채널은 **비용 대비 효율이 나쁘고 리스크가 큼**니다. 다른 채널을 먼저 소진하신 후에 고려하십시오.

6장. 30일 실행 계획

지금까지의 내용을 실제 행동으로 바꾸는 부분입니다. **읽지 말고 따라 하십시오.**

6-1. 원칙

- **매일 접속 3건**을 하한선으로 잡습니다. 0건인 날을 만들지 않는 것이 핵심입니다.
- 제품 수정은 **주 1회, 하루만** 허용합니다. (제작으로 도망치는 걸 구조적으로 막습니다)

- 모든 접촉은 기록합니다. 기록하지 않으면 개선되지 않습니다.

6-2. Week 1 — 명단 만들기

일차	할 일	산출물
1	2장 워크시트로 "흔적" 정의	검색어 5개
2	검색 실행, 후보 30건 수집	후보 리스트
3	후보 전수 검증 (오탐 제거)	유효 후보 10~15건
4	콜드메일 템플릿 작성	템플릿 1종
5	첫 3건 발송	발송 3
6	3건 발송	누적 6
7	회고 + 템플릿 수정	v2 템플릿

6-3. Week 2 — 채널 확장

일차	할 일
8~10	매일 3건 발송 (누적 15건)
11	제품 전시 플랫폼 2곳 등록 (썸네일 포함)
12	커뮤니티 1곳 게시 (SEO 자산 목적)
13	답장 온 건 전부 회신 + 힌트 추출
14	회고: 열람률/답장률 계산

6-4. Week 3 — 반복과 최적화

일차	할 일
15~19	매일 3건 (누적 30건)
20	A/B 테스트 : 제목 2종으로 각 5건 발송
21	승자 제목 확정

6-5. Week 4 — 전환

일차	할 일
22~26	매일 3건, 승자 템플릿 사용
27	가입자 전원에게 개인 감사 메일 발송
28	사용자 1명과 통화 또는 화상 인터뷰
29	인터뷰 내용으로 소개 문구 재작성
30	30일 리포트 작성

6-6. 30일 후 목표 수치

30일을 성실히 하면 나오는 현실적인 숫자입니다. 과장하지 않았습니다.

지표	기대값
누적 접촉	60~90건
답장	6~9건
실제 가입/사용	3~10명
새로 발견한 채널	1~3개 ← 진짜 성과

"고작 5명?"이라고 생각하셨습니까?

지금 0명입니다. 그리고 어떻게 5명을 만들었는지 정확히 아는 상태의 5명입니다.

이 상태에서는 같은 방법을 10배로 돌리면 50명이 됩니다. 재현 가능한 5명은 우연히 얻은 500명보다 가치 있습니다.

6-7. 추적 시트 양식

[접촉 로그]

날짜 | 대상 | 채널 | 사용 템플릿 | 열람 | 답장 | 결과 | 배운 점

[주간 지표]

주차 | 접촉 | 열람률 | 답장률 | 가입 | 비교

7장. 데려온 사람을 놓치지 않는 법

지금까지는 사람을 데려오는 법이었습니다. 이 장은 데려온 사람을 안 흘리는 법입니다.

많은 1인 개발자가 여기서 조용히 실패합니다. 어렵게 30명을 데려왔는데 29명이 3초 만에 나갑니다. 그리고 "역시 홍보가 부족했나"라고 결론 내리고 다시 홍보를 합니다. 틀린 진단입니다. 새는 양동이에 물을 더 붓는 셈입니다.

7-1. 3초 룰: 방문자가 판단하는 시간

방문자가 당신의 랜딩 페이지에 머무는 평균 시간은 3초 미만입니다. 이 3초 안에 답해야 하는 질문은 딱 세 개입니다.

#	방문자의 질문	답이 없으면
1	이게 뭐야?	즉시 이탈
2	나한테 왜 필요해?	즉시 이탈
3	지금 뭘 누르면 돼?	둘러보다 이탈

세 질문 모두 첫 화면(스크롤 없이 보이는 영역)에서 답이 나와야 합니다.

7-2. 개발자가 흔히 저지르는 랜딩 페이지 실수 7가지

실수 1. 기능을 설명한다

- ❌ "실시간 동기화, 다크모드, 태그 시스템을 지원합니다"
- ❌ "이제 폰과 노트북에서 같은 메모를 봅니다"

사용자는 기능을 사지 않습니다. 결과를 삽니다.

실수 2. 슬로건이 추상적이다

- ❌ "당신의 일상을 더 스마트하게"
- ❌ "매달 새는 구독료 3만 원을 찾아드립니다"

"스마트하게", "혁신적인", "새로운 경험" 같은 단어는 아무 정보도 전달하지 않습니다. 지우십시오.

실수 3. 스크린샷이 없다

텍스트만 있는 랜딩 페이지의 전환율은 처참합니다. 방문자는 읽지 않고 봅니다. 화면 캡처 3장이 설명

문 30줄보다 강력합니다.

실수 4. CTA 버튼이 여러 개다

"둘러보기", "가입하기", "문서 보기", "가격 보기"가 다 같은 크기로 있으면, 방문자는 **아무것도 안 누릅니다**. 선택지가 많을수록 결정은 미뤄집니다.

→ **가장 중요한 버튼 1개만 크고 진하게**. 나머지는 작고 흐리게.

실수 5. 가입을 먼저 요구한다

아직 가치를 보여주지도 않았는데 회원가입 폼부터 나오면 대부분 이탈합니다.

→ **가능하면 로그인 없이 한 번 써보게 하십시오**. 가치를 느낀 다음에 가입을 요구해야 합니다.

실수 6. 만든 사람이 안 보인다

1인 개발 서비스의 최대 약점은 **신뢰**입니다. 회사가 아니니까요.

그런데 역설적으로, 이것 숨기지 않고 드러내면 강점이 됩니다.

→ "혼자 만들고 있습니다. 문의는 이 메일로 주시면 제가 직접 답합니다." 이 한 줄이 대기업보다 강한 신뢰를 만듭니다.

실수 7. 모바일에서 안 본다

개발은 데스크톱에서 하니까 데스크톱만 확인합니다. 그런데 **콜드메일이나 커뮤니티에서 들어오는 유입의 대다수는 모바일**입니다.

→ 배포 후 반드시 **본인 휴대폰으로** 접속해서 처음부터 끝까지 확인하십시오.

7-3. 랜딩 페이지 구조 (그대로 따라 쓰는 순서)

- | | |
|-----------------|---------------------------|
| [1] 한 줄 슬로건 | - 무엇을 해주는지, 결과 중심, 20자 이내 |
| [2] 보조 설명 1~2문장 | - 누구를 위한 것인지 |
| [3] CTA 버튼 (1개) | - 동사로. "지금 무료로 시작" |
| [4] 스크린샷 또는 GIF | - 실제 작동 화면 |
| [5] "이런 분께" 3줄 | - 방문자가 자기를 알게 |
| [6] 작동 방식 3단계 | - 1-2-3으로 단순하게 |
| [7] 만든 사람 소개 | - 신뢰 확보 |
| [8] CTA 버튼 (반복) | - 스크롤 끝에 한 번 더 |

순서를 바꾸지 마십시오. 특히 [1]~[3]이 첫 화면 안에 들어가야 합니다.

7-4. 문구를 사용자의 언어로 바꾸는 법

이건 부록 T-10과 연결되는 작업입니다.

당신이 쓰는 단어와 사용자가 쓰는 단어는 다릅니다.

당신이 쓰는 말	사용자가 쓰는 말
온보딩 최적화	처음에 뭘 눌러야 할지 모르겠어요
데이터 마이그레이션	쓰던 거 그대로 옮길 수 있나요
워크플로우 자동화	매번 똑같은 거 복붙하는 게 지겨워요
협업 기능	같이 보면서 얘기하고 싶은데

작업 방법

1. 경쟁 서비스의 리뷰, 커뮤니티 질문글을 20개 읽습니다
2. 그들이 반복해서 쓰는 **명사와 동사**를 그대로 적어둡니다
3. 랜딩 페이지의 단어를 그 목록으로 교체합니다

이 작업만으로 전환율이 눈에 띄게 달라집니다. 카피라이팅 재능이 필요 없습니다. **베끼면 됩니다.**

7-5. 첫 100명까지는 "수동 온보딩"

자동화하지 마십시오. 아직은 아닙니다.

- 가입자가 생기면 → **직접 메일을 보냅니다** (부록 T-06)
- 이탈한 사람이 보이면 → **왜 나갔는지 물어봅니다**
- 오류가 나면 → **직접 사과 메일을 보냅니다**

비효율적으로 보이지만, 이 단계에서 얻는 정보의 밀도는 어떤 분석 도구보다 높습니다. **100명이 넘어가면 그때 자동화하십시오.**

8장. 콘텐츠를 장기 자산으로 만들기

3~6장의 방법은 **직접 발로 뛰는 방식**입니다. 강력하지만 한계가 있습니다. **내가 멈추면 유입도 멈춥니다.**

그래서 병행해야 하는 것이 **자산형 채널**입니다. 한 번 만들어두면 자는 동안에도 사람을 데려옵니다.

8-1. 두 채널의 역할 분담

	아웃리치 (3~5장)	콘텐츠 자산 (8장)
효과 시작	당일	2~6개월
지속성	멈추면 정지	계속 누적
초기 사용자 확보	○ 최적	✗ 너무 느림
장기 유입	✗ 확장 한계	○ 최적

결론: 아웃리치로 오늘을 버티고, 콘텐츠로 6개월 뒤를 준비합니다. 둘 다 해야 합니다.

8-2. 왜 영상인가

1인 개발자에게 가장 효율적인 자산형 채널은 **영상**입니다. 이유가 있습니다.

1. **제작 과정 자체가 콘텐츠입니다.** 어차피 만들고 있으니 추가 비용이 거의 없습니다.
2. **얼굴 없는 제품에 사람을 붙입니다.** 신뢰의 문제를 해결합니다.
3. **검색 자산이 됩니다.** 영상은 몇 년간 검색에 노출됩니다.
4. **다른 채널로 재활용됩니다.** 영상 하나로 블로그 글, 쇼츠, 커뮤니티 글이 나옵니다.

8-3. 초보 채널이 반드시 해야 할 설정 18가지

제가 실제로 채널을 개설하고 **하나씩 전부 적용한 목록**입니다. 이건 알고리즘 노출과 직결되는데, 대부분의 초보자가 절반도 안 해두고 "왜 조회수가 안 나오지"라고 합니다.

채널 단위 설정 (①~⑨)

① 채널 설명문 첫 줄에 키워드를 넣는다

검색 엔진은 앞부분에 가중치를 둡니다. "안녕하세요 저는~"으로 시작하면 낭비입니다. **다루는 주제 키워드를 첫 문장에 배치**하십시오.

② 비즈니스 이메일을 공개한다

협업 제안이 오는 유일한 통로입니다. 없으면 기회 자체가 안 옵니다.

③ 채널 핸들(@아이디)을 브랜드와 통일한다

다른 플랫폼과 같은 문자열을 쓰십시오. 검색 시 한 번에 묶입니다.

④ 채널 키워드를 10개 등록한다

설정 > 채널 > 기본정보에 있습니다. 대부분 비워둡니다. 알고리즘에게 "이 채널이 무슨 채널인지" 알려주는 직접적인 신호입니다.

⑤ 동영상 언어와 제목/설명 언어를 지정한다

지정하지 않으면 노출 대상 국가 판별이 부정확해집니다.

⑥ 업로드 기본 설정을 구성한다

매번 손으로 입력하면 반드시 빠뜨립니다. 설명 템플릿(링크.이메일 포함), 기본 카테고리, 기본 공개범위를 미리 저장하십시오.

⑦ 기본 공개범위를 "비공개"로 둔다

업로드 직후 자동 공개되면, 썸네일과 설명이 없는 상태로 노출되어 초기 성과 데이터가 망가집니다. 완성 후 수동 공개하십시오.

⑧ 자동 챍터를 끄고 수동 챍터를 쓴다

자동 챍터는 부정확합니다. 수동 챍터는 검색 노출에 유리하고, 시청자가 원하는 지점으로 바로 갑니다.

⑨ 전화번호 인증을 완료한다

안 하면 15분 이상 영상 업로드, 맞춤 썸네일 등 핵심 기능이 잠깁니다.

영상 단위 설정 (⑩~⑮)

⑩ 아동용 여부를 정확히 설정한다

"아동용"으로 잘못 설정하면 댓글이 막히고 알림이 안 갑니다. 성장이 사실상 불가능해집니다. 가장 치명적인 실수입니다.

⑪ 맞춤 썸네일을 반드시 넣는다

자동 추출 썸네일은 클릭률이 절반 이하입니다.

⑫ 제목 A/B 테스트를 돌린다

제목 후보를 3개까지 등록하면 플랫폼이 자동으로 성과를 비교해줍니다. 무료인데 안 쓰는 사람이 대부분입니다.

⑬ 설명란에 서비스 링크를 넣는다

영상은 결국 유입 통로입니다. 링크가 없으면 아무 의미가 없습니다.

⑭ 재생목록에 반드시 넣는다

재생목록은 연속 시청을 만들고, 연속 시청은 알고리즘이 가장 중시하는 지표입니다.

⑮ 최종 화면을 설정한다

"구독 버튼 + 추천 영상 1개" 조합이 기본입니다. 영상 끝의 이탈을 회수합니다.

⑩ 수동 챕터를 넣는다

5분 이상 영상은 필수입니다.

⑪ 첫 댓글을 직접 달고 고정한다

가장 저평가된 기능입니다. 고정 댓글은 모든 시청자에게 노출되는 무료 배너입니다. 서비스 링크나 핵심 요약을 넣으십시오.

⑫ 태그를 채운다

효과는 예전보다 줄었지만 여전히 0은 아닙니다. 5분이면 끝납니다.

체크리스트 사용법

새 영상을 올릴 때마다 ⑩~⑫를 확인하십시오. 9개 중 하나라도 빠지면 그 영상은 손해를 보고 시작합니다.

8-4. 조회수가 안 나올 때의 정확한 진단

초보 채널의 조회수가 안 나오면 대부분 "알고리즘이 안 밀어준다"고 생각합니다. 반응 맞고 반응 틀립니다.

알아야 할 사실: 플랫폼은 모든 영상을 일단 소수에게 노출시켜 봅니다.

이 초기 노출 집단의 크기는 채널의 신뢰도(구독자 수, 과거 성과)에 비례합니다. 신생 채널은 이 집단이 매우 작습니다. 수십~수백 명 수준입니다.

그래서 초기에는 이런 일이 벌어집니다.

- 영상이 아무리 좋아도 모수가 작아서 조회수 자체가 작게 나옵니다
- 이걸 처벌이 아니라 표본 크기의 문제입니다

따라서 초기에 봐야 할 지표는 조회수가 아닙니다.

지표	의미	이걸 봐야 하는 이유
✗ 조회수	노출 모수에 좌우됨	초기엔 무의미
○ 클릭률(CTR)	썸네일·제목의 힘	모수와 무관
○ 평균 시청 지속시간	콘텐츠의 힘	모수와 무관
○ 초반 30초 이탈률	혹의 힘	가장 중요

조회수 낮아도 시청 지속시간이 길면 그 채널은 정상 궤도입니다. 반대로 조회수가 조금 나왔는데 30초 이탈률이 70%라면, 다음 영상은 더 안 나옵니다.

8-5. 콘텐츠 → 서비스 유입 연결 설계

콘텐츠를 만들면서 가장 흔한 실수는 콘텐츠와 서비스가 따로 노는 것입니다.

연결 3중 장치

- 1. 영상 설명 첫 줄에 링크 (더보기 누르기 전에 보이는 영역)
- 2. 고정 댓글에 링크
- 3. 영상 후반부에 구두 언급 ("이 과정에서 쓴 도구는 제가 만든 건데...")

세 개 다 하십시오. 하나만 하면 놓칩니다.

9장. AI로 아웃리치를 10배로 돌리기

이 책의 방법은 노가다입니다. 하지만 노가다의 상당 부분은 AI에게 넘길 수 있습니다.

9-1. 넘길 수 있는 것과 없는 것

작업	AI 위임	이유
후보 검색	○ 완전 위임	반복 검색은 기계가 빠름
후보 1차 필터링	○ 완전 위임	조건 판별은 규칙 기반
후보 최종 검증	⚠ 부분 위임	오탐 판단은 사람이 확인
메일 초안 작성	○ 위임	빠대는 AI가
첫 문단 개인화	✗ 직접	여기가 성패를 가름
발송	✗ 직접	실수하면 되돌릴 수 없음
로그 정리	○ 완전 위임	단순 정리

절대 규칙: 첫 문단만큼은 직접 쓰십시오.

그 한 문단이 "스팸"과 "제안"을 가릅니다. 나머지는 전부 AI에게 시켜도 됩니다.

9-2. 후보 수집 자동화 프롬프트

AI 코딩 도구(클로드 코드, 커서 등)에 그대로 넣으십시오.

GitHub에서 다음 조건을 모두 만족하는 저장소를 찾아줘.

1. README에 "운영을 종료했습니다" 또는 "중지했습니다" 또는 "서비스가 종료되었습니다" 문구가 있음
2. 컨트리뷰터가 1명
3. `archived` 필드가 `false`
4. `homepage` 필드가 비어있지 않음

각 후보에 대해 아래를 표로 정리해줘:

- 저장소 이름 / 소유자 / 마지막 커밋일
- README에서 종단을 선언한 원문 문장 (그대로 인용)
- 소유자의 다른 저장소 최근 활동 여부

⚠️ 중요: README 전문을 읽고, 아래는 제외해줘.

- 다른 저장소로 "이전"한 경우 (프로젝트는 살아있음)
- 일부 폴더/컴포넌트만 폐기된 경우
- 구버전만 종료되고 신버전이 있는 경우

마지막 제외 조건이 핵심입니다. 이걸 안 넣으면 결과의 절반이 오탐입니다.

9-3. 개인화 초안 프롬프트

아래 저장소의 README 내용을 읽고,
이 프로젝트를 실제로 읽어본 사람만 쓸 수 있는
개인화 문장 2줄을 만들어줘.

조건:

- 칭찬은 구체적으로 (기술 선택, 기능, 문제의식 중 하나를 짚을 것)
- "대단하네요", "인상적입니다" 같은 일반적 표현 금지
- 2문장 이내

[README 내용 붙여넣기]

이렇게 뽑은 2줄을 **읽어보고 손으로 다듬어서** T-03 템플릿의 앞부분에 넣습니다.

9-4. 권한 설정 함정 (실전 팁)

AI 코딩 도구로 외부 저장소에 이슈를 자동 생성하려 하면 **차단**되는 경우가 많습니다. 남의 공개 저장소에 글을 쓰는 행위라 보안상 기본 차단되어 있기 때문입니다.

해결하려면 도구의 설정 파일에 해당 명령을 허용 목록에 추가해야 합니다. 다만 **자동 승인 목록은 AI가 스스로 수정하지 못하도록 막혀 있는 경우가 많으므로**, 그 부분은 직접 편집하셔야 합니다.

⚠ **설정 파일 수정 후 반드시 JSON 문법 검사를 하십시오.**

쉼표나 괄호 하나만 틀려도 **설정 파일 전체가 조용히 무시됩니다**. 에러 메시지도 안 나옵니다.

```
bash python -c "import json; json.load(open('설정파일경로'))"
```

9-5. 자동화해도 절대 늘리면 안 되는 것

AI를 붙이면 하루 100건도 보낼 수 있습니다. **하지만 보내지 마십시오.**

이유는 두 가지입니다.

1. **플랫폼이 차단합니다.** 동일 브랜드명을 짧은 시간에 반복 노출하면 자동 필터에 걸립니다. 조용히 삭제되며 알림도 없습니다.
2. **품질이 무너집니다.** 개인화 없는 100건은 개인화된 3건보다 성과가 낮습니다. 그리고 브랜드 이미지에 영구적인 손상을 남깁니다.

AI로 늘려야 하는 것은 "발송량"이 아니라 "후보 탐색량"입니다.

후보 100건을 AI가 찾아주고, 그중 좋은 3건에 당신이 정성껏 보내는 것. 이게 올바른 사용법입니다.

10장. 케이스 스터디: 실제 30일 기록

이 장은 제가 실제로 실행한 기록입니다. 성공담이 아니라 **있는 그대로의 기록**입니다.

10-1. 배경

- 서비스: 방치된 사이드프로젝트를 거래하는 **양면 마켓플레이스**
- 난이도: 양면 시장 → 콜드스타트 극악
- 예산: 0원
- 인력: 1명

전략적 결정: 양쪽 중 **공급자(프로젝트를 접은 사람)**를 먼저 채우기로 했습니다. 구매자는 볼 물건이 없으면 즉시 이탈하지만, 판매자는 어차피 놓고 있는 자산이라 설득 비용이 낮기 때문입니다.

10-2. 시도한 순서와 결과

시도 1 — 인플루언서 콜드메일 (10건)

- 결과: 답장 1건 (10%)
- 답장 내용: 완곡한 거절에 가까움
- **실제 성과: 답장 속에서 언급된 플랫폼 1개 발견** ← 이게 진짜 수확

시도 2 — 발견한 플랫폼에 서비스 등록

- 결과: 등록 완료
- 배운 점 3가지:
 - 게시판 종류를 잘못 고르면 아무도 안 봄 (자료실 vs 커뮤니티)
 - 썸네일 없으면 클릭률 처참 → 서비스 화면 캡처로 해결
 - **에디터 마크업이 실제 페이지에서 깨져서 노출됨** → 로그아웃 후 검수 필수

시도 3 — 개발자 커뮤니티 게시 (복수)

- 결과: **구조적 실패**
- 원인 분석: 커뮤니티에 글 쓸 만큼 활동적인 사람은, 정의상 "조용히 프로젝트를 접은 사람"이 아님
- **이 실패에서 이 책의 2장이 나왔습니다**

시도 4 — GitHub README 고고학

- 검색 시도: 40여 회
- 유효 후보: 5건
- 발송: 5건
- **오탐 4건 발견** (이사 / 부분폐기 / 구버전 / 무관문구)
- **차단 1건** (아카이브 저장소는 이슈 작성 불가)

10-3. 숫자로 본 결론

채널	투입 시간	접촉	유효 반응	시간당 효율
콜드메일	2시간	10	1	☆☆☆
플랫폼 등록	3시간	1	-	☆☆
커뮤니티	4시간	3	0	☆
GitHub	5시간	5	진행중	☆☆☆

10-4. 가장 큰 교훈 3가지

교훈 1. 실패한 채널이 다음 채널을 알려준다

커뮤니티 실패가 없었다면 GitHub 방법을 생각하지 못했을 겁니다. 실패를 "왜 실패했는가"까지 분석하면, 그 답이 다음 방법이 됩니다. 그냥 "안 되네"하고 넘기면 아무것도 안 남습니다.

교훈 2. 목표한 성과가 아니라 부산물이 더 크다

콜드메일의 목표는 "홍보 성사"였지만, 실제 수확은 **새 플랫폼 발견**이었습니다. 아웃리치는 정보 수집 활동이기도 합니다.

교훈 3. 검증 없이 접촉하면 절반이 오발이다

검색 결과 스니펫만 보고 판단했다면 오탐 4건에 전부 연락했을 겁니다. 살아있는 프로젝트 주인에게 "접으셨네요"라고 연락하는 건 **최악의 첫인상**입니다.

10-5. 그때로 돌아간다면 다르게 할 것

1. 커뮤니티부터 시작하지 않았을 것 — 4시간을 GitHub에 썼다면 후보 15건은 더 찾았습니다
 2. 타겟의 "공개 흔적"을 먼저 정의했을 것 — 2장의 질문을 처음부터 던졌어야 했습니다
 3. 1일 3건 규칙을 처음부터 정했을 것 — 몰아서 하다 지치는 패턴을 반복했습니다
-

이 케이스에서 당신이 가져갈 것

저는 30일 만에 대박을 내지 못했습니다. 이 책은 그런 약속을 하지 않습니다.

대신 저는 어떤 채널이 왜 작동하고 왜 작동하지 않는지 정확히 아는 상태가 되었습니다.

그 상태에서는 다음 30일이 훨씬 빠릅니다. 그리고 그게 이 책이 드릴 수 있는 전부이자, 실은 가장 중요한 것입니다.

마치며 — 노가다에 대하여

이 책을 끝까지 읽으신 분은 아마 이런 생각을 하고 계실 겁니다.

"결국 다 노가다잖아."

맞습니다. 정확합니다. 이 책에 마법은 없습니다.

하지만 한 가지만 짚고 마치겠습니다.

바이브코딩이 등장하면서 **만드는 일은 누구나 할 수 있게 되었습니다**. 진입장벽이 사라졌다는 건, 뒤집으면 **만드는 능력으로는 아무도 차별화되지 않는다는** 뜻입니다.

그런데 이 노가다는 다릅니다. **AI가 대신 해주지 않고, 대부분의 사람이 하지 않습니다**.

만드는 사람 3,000명 중에 하루 3건씩 30일 동안 연락을 보내는 사람은 3명도 안 됩니다. 지루하고, 거절당하고, 티가 안 나기 때문입니다.

그래서 이게 지금 유일하게 남은 해자입니다.

당신의 경쟁자는 오늘도 기능을 하나 더 붙이고 있습니다. 그동안 당신은 3명에게 연락하십시오. 30일 뒤에 그 차이는 90건입니다.

오늘 3건부터 시작하십시오.

이 책의 모든 방법은 저자가 직접 실행한 것입니다.

질문이나 결과 공유는 언제나 환영합니다.

CoreLabs — corelabs.studio@gmail.com

부록. 바로 쓰는 복붙 템플릿 12종

사용법: [대괄호] 부분만 바꾸면 즉시 발송 가능합니다.

경고: 템플릿을 그대로 쓰면 티가 납니다. 최소한 첫 문단은 반드시 직접 쓰십시오.
이 템플릿들의 진짜 역할은 "빈 화면 공포"를 없애는 것입니다.

T-01. 인플루언서 콜드메일 (기본형)

제목: [채널명] [특정 영상 제목] 보고 만든 서비스가 있어요

안녕하세요, [이름]님.

[특정 영상 제목] 편에서 [구체적인 내용]에 대해 말씀하신 부분이 계속 기억에 남아서 메일 드립니다.

저는 [지역/상황]에서 혼자 [서비스명]이라는 걸 만들고 있는 [본인 이름]입니다.
[한 문장으로 서비스 설명 - 무엇을 하는 것인지만]

[영상에서 언급된 문제]를 겪는 분들에게 도움이 될 것 같아서,
혹시 한 번만 눈으로 훑어봐 주실 수 있을까 해서 연락드렸습니다.

[링크]

홍보를 부탁드리는 건 아니고, 별로면 별로라고 한 줄만 주셔도
저한테는 큰 도움이 됩니다. 물론 바쁘시면 답장 안 주셔도 전혀 괜찮습니다.

좋은 영상 감사합니다.

[이름] 드림
[이메일]

체크포인트

- [] 첫 문단에 구체적인 영상 제목이 들어갔는가
- [] 링크는 1개뿐인가
- [] 스마트폰에서 스크롤 없이 읽히는가
- [] 거절 경로를 열어뒀는가

T-02. 인플루언서 콜드메일 (숫자 강조형)

이미 약간의 성과가 있을 때 씁니다. 신뢰도가 올라갑니다.

제목: [분야] 관련해서 [숫자]명이 쓰고 있는 서비스를 만들었습니다

안녕하세요, [이름]님.

[채널명]에서 [구체적 주제]를 다루시는 걸 보고 연락드립니다.

[서비스명]을 [기간] 동안 혼자 만들어서 운영 중이고,
현재 [숫자]명이 사용하고 있습니다.

[핵심 가치 한 문장]

[이름]님 시청자분들 중에도 [구체적 상황]인 분들이 계실 것 같아
공유드립니다.

[링크]

부담 갖지 마시고, 필요 없으시면 그냥 넘겨주세요.

감사합니다.

[이름]

T-03. GitHub Issue (방치 프로젝트 접촉)

제목: [프로젝트명]에 대해 여쭙보고 싶은 것이 있습니다

안녕하세요, [사용자명]님.

[프로젝트명] README를 읽다가 "[README에서 실제로 인용한 문구]"라는
부분을 보고 조심스럽게 이슈를 남깁니다.
(홍보성 이슈라 불편하시면 바로 닫아주셔도 괜찮습니다.)

[프로젝트의 구체적인 특징 1~2가지를 언급 - 실제로 읽었다는 증거]
혼자서 여기까지 만드셨다는 게 인상 깊었습니다.

저는 [서비스명]이라는 걸 운영하고 있는데,
[이런 식으로 멈춘 프로젝트를 다른 사람에게 넘기거나 이어받게 하는 곳]입니다.

[프로젝트명]처럼 [구체적 강점]이 있는 프로젝트는
그대로 두기엔 아까워서요.

[링크]

관심 없으시면 이 이슈는 닫아주시면 됩니다.
좋은 프로젝트 만들어주셔서 감사합니다.

⚠ 발송 전 필수 확인

```
gh api repos/OWNER/REPO --jq .archived # false여야 함
```

T-04. 커뮤니티 게시글 (SEO 자산형)

제목: [문제 상황]을 해결하려고 [기간] 동안 만든 것 - 회고

왜 만들었나

[개인적인 계기 – 솔직하게. 이 부분이 글의 생명입니다]

어떤 문제였나

[문제를 구체적으로. 숫자가 있으면 넣기]

어떻게 만들었나

- 기간: [기간]
- 스택: [기술]
- 혼자: [예/아니오]

[개발 과정에서 막혔던 지점 1~2개 – 정보성 가치를 넣는 부분]

만들고 나서 배운 것

[솔직한 실패담 1개는 반드시 포함. 이게 신뢰를 만듭니다]

링크

[링크]

써보시고 이상한 점 있으면 편하게 말씀해주세요.

핵심: 홍보글이 아니라 **회고글**로 씁니다. 정보 가치가 있어야 6개월 뒤 검색으로 유입됩니다.

T-05. 제품 전시 플랫폼 등록문

[서비스명]

[한 줄 슬로건 - 15자 이내]

****이런 분께****

- [상황 1]

- [상황 2]

- [상황 3]

****무엇을 하나요****

[3문장 이내]

****왜 만들었나요****

[개인적 계기 1~2문장]

****가격****

[명확하게. 무료면 무료라고]

 [링크]

필수: 서비스 화면 캡처 **3~4장** 첨부. 텍스트만 올리면 문힙니다.

T-06. 첫 가입자 환영 메일 (수동 발송)

가입자가 30명 미만일 때는 **전원에게 직접** 보내십시오. 전환율이 극적으로 다릅니다.

제목: [서비스명] 가입 감사합니다 (만든 사람입니다)

안녕하세요, [서비스명]을 만든 [이름]입니다.

자동 발송 메일이 아니라 직접 쓰고 있습니다.
아직 사용자가 많지 않아서 한 분 한 분께 인사드리고 있어요.

혹시 [서비스명]을 어떻게 알고 오셨는지,
그리고 어떤 걸 기대하고 들어오셨는지 한 줄만 알려주실 수 있을까요?

지금 단계에서는 그 한 줄이 저에게 가장 큰 도움이 됩니다.

불편한 점이나 안 되는 기능이 있으면 이 메일에 그대로 답장 주세요.
제가 직접 고칩니다.

감사합니다.
[이름]

목적은 감사가 아니라 "유입 경로 파악"입니다. 어느 채널이 실제로 작동했는지 알아내는 유일한 방법입니다.

T-07. 사용자 인터뷰 요청

제목: **혹시 15분만 시간 내주실 수 있을까요?**

[이름]님, 안녕하세요. [서비스명] 만든 [본인]입니다.

[서비스명]을 실제로 써주신 몇 안 되는 분이셔서
염치없이 부탁 하나 드립니다.

15분 정도 통화하면서 [서비스명]의 어떤 점이 별로였는지
가감없이 말씀해주실 수 있을까요?

칭찬은 안 하셔도 됩니다. 오히려 불편했던 점이 필요합니다.

[가능한 시간대 2~3개 제시]

어려우시면 그냥 무시하셔도 괜찮습니다.
감사합니다.

"칭찬은 안 하셔도 됩니다"가 핵심 문장입니다. 이 한 줄이 진짜 피드백을 끌어냅니다.

T-08. 무응답 후속 메일 (1회만)

원본 발송 5~7일 후, 같은 스레드에 회신으로 보냅니다.

[이름]님, 지난 메일에 이어 한 번만 더 연락드립니다.

바쁘실 텐데 번거롭게 해드려 죄송합니다.

이번이 마지막 연락입니다.

혹시 [서비스명]이 [이름]님 [채널/작업]과 안 맞는다고 판단되시면
그냥 넘겨주셔도 괜찮습니다.

[한 줄로 핵심 가치 재언급]

[링크]

감사합니다.

규칙: 후속은 1회까지입니다. 2회 이상은 스팸이고, 브랜드에 손해입니다.

T-09. 제휴/협업 제안

안녕하세요, [상대 서비스명] 팀 여러분.

저는 [서비스명]을 운영하는 [이름]입니다.

[상대 서비스]를 쓰는 분들이 [구체적 상황]에서
저희 서비스가 필요해지는 지점이 있는 것 같아 연락드립니다.

제안:

- [상대방이 얻는 것 - 먼저 씁니다]
- [내가 얻는 것 - 나중에 씁니다]

금전적 거래 없이 서로 소개해주는 형태로 시작해보면 어떨까 합니다.

[링크]

관심 있으시면 회신 부탁드립니다.

철칙: 상대방 이득을 먼저 쓰십시오. 내 이득부터 쓴 제안서는 읽히지 않습니다.

T-10. 앱스토어 저평점 리뷰 분석 시트

경쟁 앱 리뷰를 이 표로 정리하면 내 서비스 소개 문구가 저절로 나옵니다.

리뷰 원문(요약)	불만의 진짜 원인	그들이 쓴 표현	내 서비스가 해결하는가

"그들이 쓴 표현" 열이 핵심입니다.

당신이 "온보딩 UX 개선"이라고 부르는 것을 사용자는 "처음에 뭘 눌러야 할지 모르겠어요"라고 말합니다. 마케팅 문구는 사용자의 언어로 써야 합니다.

T-11. 30일 접촉 로그 (복사용)

날짜	대상	채널	템플릿	열람	답장	결과	배운 점
-----	-----	-----	-----	-----	-----	-----	-----

주간 집계

주차	접촉	열람	열람률	답장	답장률	가입
-----	-----	-----	-----	-----	-----	-----
W1						
W2						
W3						
W4						

진단 기준

- 열람률 30% 미만 → **제목**을 고치십시오
- 열람률 높는데 답장률 5% 미만 → **본문**을 고치십시오
- 답장은 오는데 가입이 없음 → **랜딩 페이지**를 고치십시오

병목이 어디인지 모르면 엉뚱한 곳을 고치게 됩니다. 이 표가 병목을 알려줍니다.

T-12. GitHub 후보 검증 체크리스트

발송 전 **1분 안에** 전부 확인하십시오.

- 컨트리뷰터가 1명인가?
- `archived == false` 인가? `gh api repos/O/R --jq .archived`
- homepage 필드가 있는가? `gh api repos/O/R --jq .homepage`
- README 전문을 읽었는가?
- "이사"가 아닌 "종료"가 확실한가?
- 프로젝트 "전체"가 멈췄는가? (부분 폐기 아님)
- 소유자가 다른 저장소에서는 활동 중인가?
- 중단 시점이 3년 이내인가?

8개 중 6개 이상 체크되면 발송, 아니면 다음 후보로.

자주 묻는 질문

Q. 이거 스팸 아닌가요?

경계는 명확합니다. "이 사람에게만 해당되는 문장이 한 줄이라도 있는가"입니다.

그 한 줄이 있으면 개인화된 제안이고, 없으면 스팸입니다.

100명에게 똑같은 문장을 보내면 스팸이고, 100명에게 각각 다른 첫 문단을 쓰면 영업입니다.

Q. 거절당하는 게 너무 힘듭니다.

정상입니다. 두 가지를 권합니다.

- ① **발송과 확인을 분리하십시오.** 보낸 직후에 답장을 기다리지 말고, 정해진 시간(예: 매일 저녁 9시)에만 확인하십시오.
- ② **목표를 "답장 수"가 아니라 "발송 수"로 잡으십시오.** 발송은 통제 가능하고 답장은 통제 불가능합니다. 통제 불가능한 것을 목표로 삼으면 반드시 지칩니다.

Q. 며칠 하다가 성과가 없어서 그만두게 됩니다.

30일 중 성과가 눈에 보이는 시점은 보통 3주차입니다. 1~2주차는 거의 무반응입니다.

그래서 이 책이 30일 캘린더 형태인 겁니다. 판단을 30일 뒤로 미루십시오.

Q. 제품이 아직 부족한데 알려도 될까요?

이 질문 자체가 5장에서 말한 회피 패턴입니다.

부족한 제품을 알려서 얻는 피드백이, 혼자 완성도를 올리는 것보다 항상 빠릅니다.

지금 알리십시오.

저자 호현수 (CoreLabs)

문의 corelabs.studio@gmail.com

무단 복제 및 재배포를 금합니다.