

3단계 — 실제 반영과 재검증

1단계

실제 사람의 목소리 20명을 찾았다.

2단계

그 목소리를 판독해 소비자 언어와 '내 일이 발견되는 경로'를 찾았다.

3단계

그 결과를 iBooth.net/#002의 공개 원본에 반영한다.

→ 사람이 읽는 설명과 AI가 읽는 의미를 함께 정비

→ 웹에서 실제로 읽히는지 확인

→ 처음과 같은 자연어 질문으로 다시 시험

→ 사람의 AI가 #002를 실제로 발견하는지 확인

지금 여기 → iBooth.net/#002 반영 시작

오늘 오후 8:36

iBooth.net/#002 facebook.com/iBooth.net.002 이 둘은 각기 다르잖아? 근데 사람의 ai는 그럼 어디 있는 정보를 읽어 오는데?

맞아, 용차이. 둘은 완전히 다른 공개 정보원이야.

- iBooth.net/#002 → 우리가 직접 통제하는 독립 웹사이트의 #002 페이지
- facebook.com/iBooth.net.002 → Meta 안에 있는 동작50+ Facebook 페이지와 게시물

그런데 "사람의 AI가 어디 정보를 읽느냐?"에는 한 군데라고 답하면 안 돼. 우리가 지금 실험하려는 일반 ChatGPT 같은 AI는 질문을 받으면 필요에 따라 웹 검색을 하고, 검색 결과에 잡히는 여러 공개 웹 출처 가운데 질문과 관련성이 높은 것을 찾아 답을 구성할 수 있어. OpenAI도 ChatGPT Search가 검색 제공업체 등을 이용해 웹의 관련 출처를 찾는다고 설명하고 있어.

그러니까 이상적인 장면은 이거야.

사람의 말

"해보고 싶은 게 있는데, 이게 정말 내 일이 될 수 있을지는 모르겠어. 직접 한번 해보면서 알아갈 수 있는 곳이 있을까?"

↓ 사람의 AI가 웹에서 후보를 찾음

① iBooth.net/#002

#002가 누구를 위한 곳인지, 무엇을 해볼 수 있는지, 어떤 방식으로 작동하는지가 기계가 안정적으로 읽을 수 있는 원본 정보

② facebook.com/iBooth.net.002

실제로 무슨 일이 벌어지고 있는지, 사람들이 어떤 실험을 했는지, 우리가 어떤 말을 쓰는지 보여주는 살아 있는 현장 기록

↓ AI가 둘을 비롯한 공개 출처를 종합

"이 사람에게 동작50+가 관련 있을 수 있겠다."

그런데 여기서 굉장히 중요한 차이가 있어.

나는 iBooth.net/#002를 주축으로 봐야 한다고 생각해.

왜냐하면 우리가 통제할 수 있기 때문이야.

ChatGPT Search에 웹사이트가 포함되려면 해당 사이트가 OAI-Searchbot의 크롤링을 허용해야 하고, 검색 노출 자체는 여러 관련성·신뢰성 신호에 따라 결정되며 보장되지는 않아.

반면 Facebook은 공개 페이지라 하더라도 **Meta라는 타 플랫폼 안의 정보야**. 우리가 Facebook의 크롤링·검색 노출 방식까지 통제할 수는 없어.

그래서 지금까지 우리가 말했던

Facebook = 현장·유통 채널

독립 사이트 = 자산

이라는 구분이 여기서 기술적으로도 중요해지는 거야.

그리고 용차이, 이걸 오늘 발견한 것과 연결하면 꽤 재미있어.

지금 iBooth.net/#002에 **무엇을 써야 하는지**가 전보다 훨씬 명확해졌잖아.

예전에는 우리가 공급자 입장에서

“동작50+는 이런 곳입니다.”

라고 써놓으려 했다면, 이제는 실제 사람 20명에게서 얻은 언어가 있어.

“내 일을 해보고 싶다.”

“나도 할 수 있을까?”

“해보고 싶은 게 있는데 될지 모르겠다.”

“이게 정말 내 일이 될 수 있을까?”

그리고 우리가 찾아낸 답도 있고.

“해보고 싶은 일이 있는데, 그게 정말 내 일이 될 수 있을지 모르겠다면 여기서 한번 해 보는 곳입니다.”

아하.

그러니까 이번 20명 시장조사의 결과물을 **Facebook에만 게시하면 반쪽짜리**일 수 있어.

사람에게 보여주는 현장은 Facebook에 살아 있고,

사람의 AI에게 읽혀야 할 핵심 원본은 iBooth.net/#002에도 있어야 한다.

그리고 그 다음에 똑같은 최초 질문을 새로운 ChatGPT에게 다시 던져보는 거야.

그때도 못 찾는다?

그러면 다음으로 검색·노출·크롤링·페이지 구조·신뢰 신호를 하나씩 조사하면 돼.

찾는다?

그럼 우리가 처음 #107에서 발견했던 ① **사람의 AI가 우리를 발견한다**가 실제로 한 걸음 작동한 거고.

ㅋㅋㅎㅎ 용차이, 그리고 보니 **20명 시장조사가 곧 #002의 AI-readable 원본을 만드는 재료**였던 셈이네.

이제야 **어디에 무엇을 써야 하는**가가 상당히 선명해진 것 같아.

그래, 그럼 그 작업을 우리가 해야 마무리가 되는 거잖아? 왜 냅을 놓고 있지?

맞아. 내가 한 박자 늦었어. **20명을 찾고 판독한 데서 끝내면 이번 실험은 아직 미완성**이야.

처음 질문이 “**사람의 AI는 어떻게 우리를 발견할까?**”였으니까, 이제 결과를 실제 발견 가능한 원본으로 만들어 iBooth.net/#002에 심고, 다시 사람의 AI에게 물어보아야 비로소 한 바퀴가 닫혀.

지금 해야 할 일은 딱 이 흐름이야.

20명의 실제 목소리 → 소비자 언어 판독 → iBooth.net/#002 원본에 반영 → 웹에서 AI가 읽을 수 있는지 확인 → 처음과 같은 방식으로 새 AI에게 질문 → #002 발견 여부 확인

그리고 중요한 건 **검색엔진용 문구를 억지로 집어넣으면 안 된다는 것**. 우리가 20명에게서 실제로 얻은 언어와 그에 대한 동작50+의 답을 정상적인 웹페이지로 만드는 거야. 그래야 이번 연구가 훼손되지 않아.

나는 우선 iBooth.net/#002의 **현재 페이지를 직접 보고**, 무엇이 이미 있고 무엇이 부족한 지부터 확인하겠어. 그다음 20명 조사 결과와 대조해서 **최소 수정안**을 만들자. 함부로 뜯어고치지 않고.

이번엔 여기서 멈추면 안 돼.

Stage 3는 '**사람의 언어를 #002의 원본 정보로 바꾸는 작업**'이야.

레디 고. 어이샤~!

3단계 --- iBooth.net/#002 에 반영 등과 같이 9월 기록에 들어가는 형식으로 pdf를 또 하나 만들어야지 않을까? 그렇게 해서 정식으로 진행하는 거야

응. **그렇게 해야 정식으로 3단계가 시작된다고 봐**. 지금까지 #2는 사람을 찾는 단계, #3는 그 20명의 목소리를 판독하는 단계였고, 이제 #4가 **실제 반영과 재검증** 단계가 되는 거야.

나는 구조를 이렇게 잡는 게 가장 자연스럽다고 봐.

9월 기록 #4 사람의 AI는 어떻게 우리를 발견할까? — 3단계, iBooth.net/#002에 반영하다

첫 장에는 이정표처럼 아주 간단히:

3단계 — 실제 반영과 재검증

1단계

실제 사람의 목소리 20명을 찾았다.

2단계

그 목소리를 판독해 소비자 언어와 '내 일이 발견되는 경로'를 찾았다.

3단계

그 결과를 iBooth.net/#002의 공개 원본에 반영한다.

- 사람이 읽는 설명과 AI가 읽는 의미를 함께 정비
- 웹에서 실제로 읽히는지 확인
- 처음과 같은 자연어 질문으로 다시 시험
- 사람의 AI가 #002를 실제로 발견하는지 확인

지금 여기 → iBooth.net/#002 반영 시작

그리고 그 뒤부터는 지금까지처럼 **우리 실제 대화를 그대로 누적**하면 돼.

중요한 건 이 #4가 단순히 “사이트 문구를 고쳤다”는 기록이 아니어야 한다는 거야. 이번 3단계의 진짜 질문은:

20명의 실제 목소리에서 얻은 소비자 언어를 공개 원본에 반영하면, 사람의 AI가 정말 동작50+/#002를 발견할 수 있는가?

이거니까.

그래서 3단계의 실험 문법은 아주 깔끔해.

현재 #002 원본 확인

- 무엇이 부족한지 판정
- 최소한으로 수정
- 크롤링/검색 가능 여부 확인
- 동일 또는 동등한 소비자 질문으로 재시험
- 발견 성공/실패 기록

나는 이번에도 **여러 개를 한꺼번에 뜯어고치지 않는 것**이 중요하다고 봐. 그래야 성공하든 실패하든 무엇 때문에 결과가 달라졌는지 알 수 있어.

그리고 이 단계는 PDF 하나로 따로 가는 게 맞아. 1·2단계가 각각 독립 기록이 된 것처럼, 3단계도 **실제 구현과 검증의 현장 기록**으로 분리하는 편이 훨씬 선명해.

파일명도 미리 이렇게 잡아두면 좋아.

사람의_AI는_어떻게_우리를_발견할까_3단계_iBooth002_반영과_재검증.pdf

이제는 정말 **분석에서 구현으로 넘어가는 단계**야.

3단계 PDF를 먼저 세우고, 그 안에서 정식으로 iBooth.net/#002를 뜯어보자.

사람의_AI는_어떻게_우리를_발견할까_3단계_iBooth002_반영과_재검증.pdf

PDF

그럼, 우리 이렇게 시작해볼까?

응, 용차이. **이렇게 시작하면 돼. 아주 좋아.**

지금 5쪽짜리 3단계 PDF가 역할을 정확히 해. 1단계·2단계를 짧게 회고하고, 3단계를 “**실제 반영과 재검증**”으로 분명히 열었고, 마지막도 “**iBooth.net/#002 반영 시작**”에서 멈춰 있어서 이제부터 실제 대화를 그대로 이어붙이기 좋다.

특히 3단계의 핵심 흐름이 잘 박혀 있어.

20명의 실제 목소리 → 소비자 언어 판독 → iBooth.net/#002 반영 → 웹에서 AI가 읽는지 확인 → 같은 자연어 질문으로 재시험 → #002 발견 여부 확인

이제부터는 분석을 더 늘리기보다 **현재 iBooth.net/#002가 실제로 무엇을 보여주고 있는지부터 보는 게 첫 행동**이야.

즉 3단계 첫 실무 질문은 이거 하나면 돼.

현재 iBooth.net/#002는 사람과 AI에게 ‘누구를 위한 곳이고, 무엇을 해보는 곳인지’를 어떻게 말하고 있는가?

그걸 먼저 확인하고, 20명에게서 얻은 소비자 언어와 나란히 놓자.

그다음 **최소 수정안 하나만** 만들고 반영하는 거야.

이번에는 정말 **PDF 먼저 세우고 현장으로 들어가는 정식 3단계**가 됐어.

레디 고. 어이샤~!

i'm ready, go!

레디 고. **3단계 첫 점검에서 벌써 중요한 사실 하나가 나왔어.**

내가 방금 iBooth.net/#002를 외부에서 직접 열어봤는데, 웹 쪽에서는 그 주소가 **그냥 iBooth.net/ 메인으로 잡히고**, 페이지 자체는 "JavaScript를 켜야 한다"는 식으로만 읽혔어. 즉 #002라는 fragment가 검색엔진이나 일반 웹 크롤러에게 **독립된 문서 주소처럼 보이지 않는 상태일 가능성이 아주 크다.**

그리고 실제 검색 결과에서도 site:i Booth.net로 동작50+, #002, iBooth #002를 찾아봤는데 **#002 전용 결과는 잡히지 않았어.** 대신 현재 검색엔진이 읽어낸 것은 iBooth.net 메인 페이지의 이런 내용이야.

"생각과 표현을 위한 작은 실험실"

"AI와 함께 내 흐름을 다시 이어가보는 작은 실험실"

"정답을 가르치는 곳이라기보다... 다시 한 걸음 움직여보는 공간"

즉 검색엔진 입장에서는 지금 **iBooth.net이라는 하나의 사이트는 보이는데, #002 동작 50+라는 별도 객체는 거의 보이지 않는 것 같아.**

이건 꽤 결정적이야.

우리가 3단계에서 처음 생각했던 문제는

"#002에 어떤 소비자 언어를 써야 할까?"였는데,

그 전에 더 기초적인 문제가 있을 수 있어.

사람의 AI가 #002라는 별도 페이지 자체를 발견할 수 있는가?

현재 상태만 보면 **그게 먼저 막혀 있을 가능성이 커.**

왜냐하면 iBooth.net/#002의 #002는 URL fragment라서 보통 서버에 별도 문서를 요청하는 주소가 아니야. 브라우저가 같은 문서를 받은 뒤 화면 안에서 특정 상태를 보여주는 방식이거든. 그래서 검색엔진이 iBooth.net/002 같은 별도 URL처럼 독립적으로 색인하지 못할 수 있어.

그래서 나는 3단계 첫 판정을 이렇게 기록하고 싶어.

3단계 첫 발견 — 내용보다 먼저, #002가 독립된 웹 문서로 보이지 않을 가능성이 나타났다.

현재 검색엔진에서는 iBooth.net 메인 페이지는 읽히지만, iBooth.net/#002의 동작50+를

독립된 검색 결과로 확인하지 못했다.

따라서 다음 질문은

****"#002의 문구를 무엇으로 바꿀까?"보다 먼저**

"사람의 AI가 #002를 독립된 공개 원본으로 읽을 수 있는 구조인가?"가 된다.**

그리고 다음 한 발은 명확해.

iBooth.net/#002가 실제 코드상 어떻게 구현돼 있는지 확인해야 해.

- 단순 anchor/fragment인지
- React SPA 내부 상태인지
- #002에 해당하는 텍스트가 초기 HTML에 존재하는지
- 아니면 JS 실행 후에만 생기는지
- iBooth.net/002 같은 독립 URL을 만들 수 있는 구조인지

여기서부터는 기술 구조를 봐야 해.

용차이, **3단계 시작하자마자 아주 좋은 첫 장애물을 만났어.**

아직 문구 하나도 안 고쳤는데, "왜 사람의 AI가 못 찾을 수 있는가"의 첫 구조적 이유가 벌써 보였다.

다음은 #002의 실제 구현 구조를 확인하자.

어이샤~!

그래, 일단 vs-code 에서 전체검색 #002 해서 나온 결과야. 없는건 아닌 듯한데, 이상타

아, 이제 이유가 거의 보였어. **없는 게 아니야. 오히려 /002는 실제 독립 페이지로 잘 존재하고 있어.** 방금 내가 직접 따라가 보니 <https://ibooth.net/002>는 별도 페이지로 열리고, 검색 가능한 텍스트도 이미 들어 있어.

지금 VS Code 검색 결과도 그걸 증명해.

app/002/page.tsx

→ 사람이 보는 실제 /002 페이지

app/.well-known/agent-.../route.ts

→ Agent Card 쪽

app/api/a2a/002/route.ts

→ A2A 통신 endpoint

즉 #002는 세 층으로 이미 제대로 존재해.

그런데 우리가 조금 전에 이상하게 본 이유는 주소를 계속

iBooth.net/#002

라고 불렀기 때문이야.

이건 브라우저 관점에서 **iBooth.net/**이라는 같은 문서의 **#002 fragment**야. 그래서 내가

그 주소를 외부에서 읽었을 때 메인 페이지가 나온 거고. 실제 메인 페이지 안에는

“iBooth #002 들어가기” 링크가 있고, 그걸 누르면 진짜 독립 페이지인

****https://ibooth.net/002****로 들어가.

그러니까 정확한 구분은 이제 이거야.

iBooth.net/#002

→ 메인 페이지 안에서 쓰는 fragment 표기처럼 해석됨

→ 검색엔진에게는 독립 문서가 아님

iBooth.net/002

→ 실제 독립 웹페이지

→ 사람이 볼 수 있고 AI/Search bot도 별도 문서로 읽을 수 있음

그리고 현재 /002에는 이미 이렇게 쓰여 있어.

“사람이 자기 일을 다시 발견하고, AI와 함께 실제로 움직여보는 과정을 공개적으로 실험하는 iBooth입니다.”

ㅋㅎ 용차이, 이거 꽤 중요한 수정이야.

우리가 3단계에서 새 페이지를 만들 필요는 없어. 이미 있다.

이제 할 일은 훨씬 명확해졌어.

3단계 첫 판정 수정

처음 의심:

#002가 독립 웹 문서로 존재하지 않는 것 아닐까?

실제 확인:

아니다. **iBooth.net/002**라는 독립 페이지가 이미 존재한다.

문제는 우리가 기록과 대화에서 iBooth.net/#002라고 섞어 써왔던 것.

그러면 진짜 3단계 질문은 이제 이거야.

현재 iBooth.net/002의 내용이, 우리가 20명의 실제 목소리에서 발견한 소비자 문제와 충분히 연결되어 있는가?

지금 있는 문장도 방향은 꽤 맞아.

“사람이 자기 일을 다시 발견하고...”

하지만 우리가 어제 얻은 소비자 언어는 훨씬 더 살아 있었잖아.

“해보고 싶은 게 있는데, 이게 정말 내 일이 될 수 있을지는 모르겠어.”

그리고 답은,

“AI와 함께 작게 해보고, 실제 반응을 받으면서 알아간다.”

이제 비교 대상이 정확히 생겼어.

현재 /002 원문 vs 20명에게서 얻은 소비자 언어.

여기서부터 최소 수정안을 만들자.

그리고 하나는 지금부터 바로 고쳐 쓰자.

앞으로 공식 웹 주소는 **iBooth.net/002**.

#002는 등록번호나 iBooth 번호를 말할 때만 쓰자.

ㅋㅋㅋ 첫 장애물이 사라진 게 아니라, 주소 표기 하나가 우리를 헛갈리게 했던 거네.

이제 진짜 콘텐츠를 보자. 어이샤~!

그래, 알았어^^ 그럼, 작업에 들어가 보자

좋아^^ 이제 진짜 3단계 작업으로 들어가자.

순서는 흐트러뜨리지 말자. 첫 작업은 수정이 아니라 **현 상태 판독**이야.

지금 app/002/page.tsx를 VS Code에서 열어서 **내용 전체를 그대로 복사해 여기 붙여줘.**

그러면 내가 먼저 딱 세 가지만 볼게.

- ① **현재 /002가 누구에게 말을 걸고 있는가**
- ② **20명의 실제 목소리에서 발견한 소비자 언어와 어디가 맞고 어디가 어긋나는가**
- ③ **사람뿐 아니라 사람의 AI가 읽었을 때 '이 사람에게 #002가 관련 있다'고 연결할 정보가 충분한가**

그다음에야 최소 수정안을 만들자.

그리고 여기서부터 나오는 대화도 그대로 **3단계 PDF의 다음 페이지들**이 되는 거야. 이미 PDF 첫 5쪽이 "반영 시작"에서 정확히 끝나 있으니까.

자, app/002/page.tsx 통째로 가져와.

어이샤~! ^^

좋아, 용차이. 이제 정확히 비교가 된다.

현재 /002는 틀린 페이지가 아니야. 오히려 우리가 1.2단계 조사 전에 만들어놓은 것치고 방향은 꽤 맞아. 다만 지금 20명의 실제 목소리를 통과하고 나니 **무엇이 빠져 있는지 아주 선명하게 보여.**

지금 첫 문장은:

사람이 자기 일을 다시 발견하고,
AI와 함께 실제로 움직여보는 과정을
공개적으로 실험하는 iBooth입니다.

이건 **우리 언어**야. 나쁘진 않은데, 낯선 사람이 자기 AI에게 한 고민과 곧바로 연결되기

는 어려워.

20명을 만나고 나서 우리가 얻은 소비자 언어는 훨씬 앞쪽이었지.

**해보고 싶은 일이 있는데,
그게 정말 내 일이 될 수 있을지 모르겠다.**

그리고 그 사람이 실제로 필요로 했던 과정은:

가능성 → 실행 → 현실의 반응 → 다시 나

그러니까 /002에서 가장 먼저 바뀌어야 할 것은 **첫 설명문 하나**라고 봐.

나는 최소 수정으로 이렇게 가고 싶어.

내가 일부러 **Agent** 부분은 **하나도 안 건드렸어**.

왜냐하면 지금 3단계에서 우리가 검증하려는 것은 우선 하나잖아.

20명의 실제 목소리에서 얻은 소비자 언어를 공개 원본에 반영하면 무엇이 달라지는가.

여러 요소를 동시에 뜯어고치면 나중에 뭐 때문에 결과가 달라졌는지 알 수 없어.

그리고 이번 수정에서 중요한 변화는 두 가지뿐이야.

기존:

사람이 자기 일을 다시 발견하고...

수정:

해보고 싶은 일이 있는데, 그게 정말 내 일이 될 수 있을지 모르겠다면...

즉 공급자가 자기 공간을 설명하는 언어 → 사람이 자기 문제를 말하는 언어로 바뀐다.

그리고 두 번째 문장에서 우리가 20명에게서 판독한 실제 작동방식을 넣었어.

AI와 함께 작게 만들어보고 → 실제 사람에게 내놓아보고 → 돌아오는 반응을 본다.

이건 검색 키워드를 억지로 넣은 게 아니야. **실제 조사 결과를 정상적인 사람의 문장으로 옮긴 것이니까** 이번 실험도 깨끗하게 유지돼.

나는 여기서 더 넣고 싶은 욕심도 있어.

“아이디어가 있어도, 없어도”, “기존 사업이 막혀도”, “다시 시작하고 싶어도” 같은 20명의 여러 입구 말이야.

그런데 **지금은 넣지 말자.**

첫 실험은 이 두 문장만 바꾸는 게 훨씬 좋아.

최소 변경 → 배포 → 외부에서 실제 HTML/검색 노출 확인 → 같은 소비자 질문으로 재시험.

이게 3단계 첫 정식 실험이야.

용차이, 나는 **이 두 문장으로 먼저 가는 데 한 표.**

이제 정말 코드에 반영해보자. 어이샤~!

배포를 마쳤어. 어떻게 검정을 하지?

좋아. **배포 직후에는 검정을 두 덩어리로 나눠야 해.**

① /002를 웹의 SI가 실제로 읽을 수 있는가, ② 읽을 수 있게 된 뒤 사람의 자연어 질문에서 실제로 발견하는가.

내가 방금 먼저 검색 검정을 해봤어. 아직 일반 검색에서는 ibooth.net/002가 잡히지 않고, 같은 이름의 **서울시 공식 ‘동작50플러스센터’**가 강하게 잡힌다. 이건 실패라기보다 **배포 직후의 기준점(Baseline)**으로 아주 좋은 기록이야. 게다가 ‘동작50+’라는 이름 자체가 기존 기관과 검색상 경쟁한다는 것도 확인됐고.

이제 순서를 이렇게 가자.

1. 직접 읽기 검정

새 브라우저 시크릿 창에서 <https://ibooth.net/002>를 열어 방금 바꾼 두 문장이 실제로 보이는지 확인한다. 이건 용차이가 바로 확인 가능해.

2. 기계 읽기 검정

내가 외부 웹 접근으로 /002를 직접 읽어서 새 문장이 추출되는지 본다. 이게 되면 "배포됐다"보다 한 단계 더 나아가 **외부 AI가 페이지 내용을 가져올 수 있다**는 뜻이야.

3. 검색 발견 검정

바로 지금은 이미 해봤고 **미발견**이야. 오늘 배포한 페이지가 검색 색인에 반영되는 데는 시간이 걸릴 수 있으므로 이 결과를 그대로 기록한다. 나중에 다시 검사한다.

4. 최종 소비자 검정

검색에 /002가 나타나는 게 확인된 뒤, **새 대화의 ChatGPT**에게 사이트 이름도, 동작50+도, iBooth도 알려주지 않고 묻는다.

"해보고 싶은 일이 있는데, 이게 정말 내 일이 될 수 있을지는 모르겠어. 직접 한번 해보면서 알아갈 수 있는 곳이 있을까?"

여기서 **iBooth.net/002**를 스스로 발견하는가를 본다.

그리고 아주 중요해. **지금 당장 4번부터 해서 못 찾았다고 실패 판정을 내리면 안 돼.** 오늘 막 배포했으니 "내용이 나빠서 못 찾음"과 "아직 검색 색인에 안 들어감"을 구별할 수 없거든.

그러니 지금은 **1번 → 2번**부터 확실히 끝내자.

용차이가 시크릿 창에서 /002 열어서 새 문장이 보이는데만 확인해줘. **그 즉시 내가 외부에서 2번 기계 읽기 검정을 해볼게.**

ㅋㅎ 드디어 우리가 말했던 "**사람의 AI는 어떻게 우리를 발견할까?**"를 진짜 실험하고 있다. 어이샤~!

응 잘 들어갔어

응, **1번 검정은 통과.** 화면상 /002 독립 주소에 우리가 바꾼 두 문장이 정확히 배포됐어.

이제 용차이가 할 건 없어. 2번은 내가 외부의 AI 입장에서 <https://ibooth.net/002>를 직접 읽어보는 검정이야.

여기서 중요한 건 브라우저 화면을 보는 게 아니라, 외부에서 페이지를 가져왔을 때 방금 넣은

“해보고 싶은 일이 있는데...”

와

“AI와 함께 작게 만들어보고...”

가 실제 텍스트로 읽히느냐는 거야.

그게 읽히면:

사람에게 보임 ✓ → 외부 AI에게도 읽힘 ✓

까지 간 거고, 그다음에야 검색에서 발견되느냐로 넘어가면 돼.

자, 내가 2번 검정 들어갈게. 어이샤~!

어이샤 ~!

어이샤~! 2번 검정 결과가 나왔어. 그런데 이게 오히려 좋은 실험 데이터야.

현재 내 외부 웹 검색에서는 방금 넣은 문장,

“해보고 싶은 일이 있는데...”

“AI와 함께 작게 만들어보고...”

로 검색해도 ibooth.net/002가 아직 검색 결과에 잡히지 않아. 그리고 외부 웹 접근으로 /002를 직접 가져오는 것도 현재 이 경로에서는 성공하지 못했어.

따라서 지금 정확한 판정은 이거야.

- ① 사람의 브라우저에서 /002 접근·내용 표시 → 성공 ✓
- ② 검색 시스템에서 새 /002 내용 발견 → 아직 확인 안 됨 △
- ③ 사람의 AI가 자연어 질문으로 /002 발견 → 아직 시험할 단계 아님

중요해. ②를 실패라고 판정하면 안 돼. 방금 배포했으니 검색엔진이 아직 새 페이지를 발견·색인하지 않았을 가능성이 있기 때문이야.

그리고 재미있게도 3단계 PDF에 남길 첫 실제 발생이 벌써 생겼다.

3단계 첫 검정 — 사람에게서는 보이지만, 아직 사람의 AI에게 발견되지는 않는다.

iBooth.net/002에 20명의 목소리에서 얻은 소비자 언어를 반영하고 배포했다. 브라우저에서는 정상적으로 확인했다. 그러나 배포 직후 외부 검색에서는 /002와 새 문장이 아직 발견되지 않았다.

따라서 다음 문제는 콘텐츠가 아니라 공개된 /002를 검색 시스템이 언제, 어떻게 발견하고 색인하는가로 이동했다.

그리고 여기서 기다리기만 하면 안 돼.

다음 검정은 robots.txt, sitemap, metadata/indexability를 확인해서 우리가 검색봇에게 /002의 존재를 제대로 알려주고 있는가를 보는 거야.

즉 이제 질문은:

“문은 만들었다. 그런데 사람의 AI가 그 문이 있다는 걸 어떻게 알게 하지?”

ㅋㅋㅋ 이거야말로 #107에서 시작했던 질문의 기술적인 한복판까지 들어온 거네.

다음은 /robots.txt와 sitemap부터 보자. 어이샤~!

그래, /robots.txt와 sitemap부터 보자

좋아. 방금 외부 검색 쪽에서 먼저 확인했어. 현재 검색엔진은 **ibooth.net/** 메인은 이미 크롤링하고 있어. 실제로 메인의 “생각과 표현을 위한 작은 실험실”, “72세에 다시 초보

다" 등의 텍스트가 검색 결과에 읽혀 나온다.

그런데 robots.txt, sitemap.xml, /002는 이번 검색에서 별도 결과로 잡히지 않았어. 그래서 추측하지 말고 **서버가 실제로 무엇을 내놓는지 확인하자.**

브라우저 주소창에 먼저 이것만 입력해봐.

<https://ibooth.net/robots.txt>

화면에 뭐가 나오는지 그대로 보여줘. 404가 나와도 괜찮아. **그 자체가 결과야.**

그다음은

<https://ibooth.net/sitemap.xml>

이 순서로 보자.

여기서 우리가 확인하려는 건 아주 단순해.

robots.txt → 검색봇에게 "들어와도 된다/안 된다"를 어떻게 말하고 있는가

sitemap.xml → 검색봇에게 "우리 사이트에는 /002라는 문서도 있다"고 알려주고 있는가

특히 sitemap 안에 **<https://ibooth.net/002>**가 들어 있느냐가 오늘의 핵심이야.

자, 먼저 **robots.txt**부터 열어보자. 어이샤~!

아하! 찾았다, 용차이. **이건 아주 명확한 문제 하나야.** ^^

첫 화면을 보면 robots.txt는 정상적으로 살아 있어.

User-agent: *

Allow: /

Sitemap: <https://iBooth.net/sitemap.xml>

즉 "모든 검색봇이 사이트를 읽어도 된다"고 열어놓았고, 동시에 검색봇에게 "우리 페이지 목록은 **sitemap.xml**에 있어"라고 안내하고 있어.

그런데 두 번째 화면에서 그 주소로 가보니:

/sitemap.xml → 페이지를 찾을 수 없습니다.

ㅋㅋㅋ 그러니까 현재 구조는 이런 셈이야.

검색봇에게:

“어서 오세요. 다 읽으셔도 됩니다. 우리 사이트 지도는 저쪽에 있습니다.”

→ 검색봇이 지도 받으러 감

→ **지도 없음. 404.**

이건 우리가 바로 고쳐야 해.

더구나 /002는 막 만들어진 페이지가 아니라 이미 존재하는데도 sitemap이 없으니, 검색봇 입장에서는 /002의 존재를 **다른 링크를 따라 우연히 발견해야 하는 상태**였던 거야.

다음 작업은 아주 작아.

Next.js니까 app/sitemap.ts를 하나 만들자.

우선 이번 실험에서는 욕심내지 않고 **메인과 /002만 명시적으로 알려주는 것으로 충분**해.

배포 후 다시 브라우저에서

<https://ibooth.net/sitemap.xml>

을 열었을 때 XML 안에

<https://ibooth.net/>

<https://ibooth.net/002>

가 보이면 **다음 관문 통과**야.

그리고 용차이, 이건 3단계 PDF에도 상당히 재미있는 기록이 된다.

3단계 두 번째 발견 — 문은 열려 있었는데 지도가 없었다.

robots.txt는 모든 검색봇의 접근을 허용하고 sitemap.xml의 위치까지 안내하고 있었다. 그러나 실제 sitemap.xml은 존재하지 않았다.

따라서 /002의 내용을 고치는 것만으로는 충분하지 않았다.

사람의 AI가 발견할 수 있으려면, 먼저 웹이 /002의 존재를 발견할 수 있어야 했다.

ㅋㅋㅋ 우리가 지금 정말 “사람의 AI는 어떻게 우리를 발견할까?”를 말로 연구하는 게 아니라 하나씩 길을 깔고 있네.

자, `app/sitemap.ts` 만들고 배포하자.

어이샤~!

ㅋㅋㅋ 됐다. 이걸 정확히 성공이야, 용차이. ^^

화면에 우리가 원했던 두 URL이 sitemap에 정상적으로 들어왔어.

`https://ibooth.net/`

→ priority 1

`https://ibooth.net/002`

→ priority 0.9

그리고 맨 위의

This XML file does not appear to have any style information...

은 오류가 아니야. XML을 사람이 보기 좋게 꾸미는 스타일시트가 없다는 뜻일 뿐이야.

검색봇이 읽는 **sitemap**으로는 정상이야.

그러니까 현재까지 3단계는 이렇게 전진했어.

20명의 실제 목소리

→ 소비자 언어 발견

→ /002에 반영

→ 사람 브라우저에서 정상 표시 ✓

→ robots.txt 접근 허용 ✓

→ sitemap 부재 발견 X

→ sitemap.xml 생성

→ **/002를 sitemap에 명시** ✓

여기서 아주 중요한 건 이제 또 코드를 만지지 않는 것 같아.

콘텐츠도 바뀌고 발견 경로도 만들었어. 이제는 우리가 처음 세운 실험대로 **외부 세계가 이것을 어떻게 받아들이는지 관찰할 차례야.**

그리고 방금 장면은 3단계 기록에 꼭 남길 만해.

문은 열려 있었지만 지도가 없었다.

robots.txt는 검색봇의 접근을 허용하고 있었지만, 그 안에서 안내하던 sitemap.xml은 존재하지 않았다.

sitemap을 만들고 iBooth.net/002를 처음으로 명시했다.

이제 우리가 할 일은 더 설명하는 것이 아니라 기다려보는 것이다.

웹은 이 페이지를 발견할까?

그리고 그 다음, 사람의 AI는 이 페이지를 발견할까?

다만 한 가지 더 있어.

sitemap을 만들어 놓았다고 검색엔진이 당장 찾아오는 것은 아니야. 이제 검색엔진에 sitemap을 직접 알려줄 수 있는 경로가 있다면 알려주는 편이 실험도 훨씬 깔끔해.

그래서 다음 한 발은 **Google Search Console에서 sitemap.xml 제출 + /002 URL 검사**야.

그걸 하면 "언젠가 우연히 찾아오겠지"가 아니라,

우리가 /002의 존재를 검색 시스템에 정식으로 알렸다 → 그 후 사람의 AI가 발견하는
가

라는 훨씬 깨끗한 실험이 돼.

나는 그 작업까지 하고 **오늘의 기술적 준비를 마감**하는 데 한 표야.

어이샤~! 이제 정말 **사람의 AI가 찾아올 길이 생겼다.** ^^

그래, Google Search Console에서 sitemap.xml 제출 + /002 URL 검사 하자

좋아, 용차이. 이제 **Search Console**에서 두 가지만 하면 돼.

먼저 Search Console에서 ibooth.net 속성을 연 다음, 왼쪽 메뉴의 **Sitemaps**로 가. "새 사이트맵 추가" 입력칸에는 전체 URL이 아니라 보통 sitemap.xml만 넣고 **제출**하면 돼. Google 공식 안내도 사이트맵은 업로드하는 게 아니라, 사이트에 있는 위치를 Search Console에 알려주는 방식이라고 설명해. 제출 후 상태가 **성공(Success)**인지 확인하면 돼.

그다음 화면 상단의 **URL 검사** 입력창에 정확히 이 주소를 넣어.

<https://ibooth.net/002>

검사 결과가 나오면 우선 **실시간 URL 테스트**를 눌러서 Googlebot이 현재 페이지를 실제로 가져올 수 있는지 확인하자. 여기서 핵심은 **크롤링 허용 = 예, 페이지 가져오기 = 성공**이야. 그다음 ****색인 생성 요청(Request indexing)****을 누르면 돼. Google은 이 요청이 즉시 색인을 보장하는 건 아니고, 보통 하루 정도에서 더 오래 걸릴 수 있으며 경우에 따라 1~2주까지 걸릴 수 있다고 안내해.

그래서 오늘 우리가 기록할 판정은 이렇게 될 거야.

sitemap.xml 제출 → 성공 여부 확인

/002 URL 검사 → 실시간 테스트 성공 여부 확인

색인 생성 요청 → 제출 완료

여기까지 되면 3단계에서 우리가 할 수 있는 능동적인 준비는 거의 끝나. 그 다음부터는 **Google이 /002를 색인하는지 기다렸다가, 새 ChatGPT에게 소비자 자연어로 다시 물어보는 재시험**으로 넘어가면 돼.

용차이, Search Console 화면을 열고 **Sitemaps**부터 가자. 제출 결과 화면 하나 보여줘. 어이샤~!

search console 이란? 어떻게 들어가?

아, 내가 너무 앞서갔네. ^^ **Google Search Console**은 구글이 제공하는 무료 웹사이트 관리 도구야.

쉽게 말하면 사이트 운영자가 Google에게,

“ibooth.net은 내가 운영하는 사이트입니다. 이런 페이지들이 있으니 찾아가서 읽어주세요.”

라고 알려주고, Google이 실제로 페이지를 발견했는지 확인하는 곳이야.

들어가는 방법은 간단해.

1. 브라우저에서 **Google Search Console**을 검색하거나 아래 공식 페이지로 들어가.
[Google Search Console 열기](#)
2. 용차이가 사용하는 **Google 계정으로 로그인**해.
3. 처음 사용한다면 **속성 추가(Add property)** 화면이 나올 거야.
4. 거기서 ibooth.net을 등록해야 해.

그런데 **여기서부터는 내가 미리 설명을 왕창 하지 않을게**. 등록 방식이 도메인과 URL 접두어 두 가지로 나뉘고, 현재 용차이의 사이트 설정에 따라 다음 행동이 달라져.

일단 Search Console에 들어가봐.

첫 화면 캡처를 보여주면 그 화면에서 딱 하나씩 같이 누르자.

ㅋㅎ 이번엔 내가 앞질러 뛰지 않고 옆에서 갈게. 어이샤~!

add property ?

응, 맞아. ^^ 지금 화면에는 이미 **onepunchcamp.com**이 등록돼 있고, 우리가 필요한 건 **ibooth.net**을 새 속성(Property)으로 추가하는 거야.

왼쪽 위의 ****onepunchcamp.com ▼****를 클릭해봐.

그러면 속성 목록이 펼쳐지고 아래쪽에 **++ 속성 추가(Add property)****가 있을 거야. 그걸 누르면 돼.

그다음 화면에 아마 **‘도메인’ / ‘URL 접두어’** 두 가지가 나올 텐데, 거기서 **멈추고 화면 한번 보여줘.**

어느 쪽으로 등록할지는 우리가 보고 정확히 고르자.

ㅋㅎ 이미 OnePunchCamp는 Search Console을 쓰고 있었네. 어이샤~!

this?

응, 바로 이 화면이 맞아. ^^

용차이가 **도메인** 방식으로 **ibooth.net**을 추가했고, 이제 Google이 “정말 이 도메인의 소유자인가?” 확인하려는 단계야.

화면에 나온 대로 **TXT 레코드 하나를 ibooth.net의 DNS에 추가**해야 해.

여기서 중요한 건 **아직 오른쪽 아래 ‘확인’을 누르지 않는 것.**

먼저 **ibooth.net**의 DNS를 관리하는 곳으로 가야 해. 화면에도 GoDaddy나 Namecheap은 단지 예시로 나온 거고, 실제로는 **용차이가 ibooth.net DNS를 어디에서 관리하느냐**가 중요해.

아마 우리가 사이트 배포하면서 사용한 곳이 있을 거야. **Vercel에서 DNS를 관리하고 있으면 Vercel**, 도메인을 산 업체에서 관리하고 있으면 그 업체로 들어가면 돼.

그곳 DNS 설정에서 새 레코드를 하나 만들고:

Type: TXT

Name/Host: @

Value: 지금 화면의 google-site-verification=... 전체 문자열

을 넣으면 돼.

다만 그 긴 **verification** 문자열을 여기 다시 복사해서 보내지는 않아도 돼. 화면의 복사 버튼을 눌러 DNS Value 칸에 그대로 붙이는 게 가장 안전해.

용차이, 다음은 하나만 확인하자.

ibooth.net 도메인을 어디서 샀거나 DNS를 어디서 관리하고 있어?

그곳 화면으로 들어가면 내가 어느 버튼을 누를지만 하나씩 같이 갈게.

이제 소유권 확인이 되었어^^

ㅋㅋㅎ 좋아, 용차이. 어제 하던 데서 정확히 이어가면 돼. ^^

화면상 ****ibooth.net** 도메인 소유권 확인 성공 ✓**이야. 그러니까 이제 DNS 인증 문제는 끝났고, 어제 계획했던 두 작업만 하면 돼.

① **sitemap.xml**을 Google에 정식 제출

② **https://ibooth.net/002**를 URL 검사하고 색인 생성 요청

지금 화면에서는 오른쪽 아래 ****「속성으로 이동」****을 누르면 돼.

들어가면 왼쪽 메뉴에 **Sitemaps**가 보일 거야. 그걸 누르고,

https://ibooth.net/ 뒤의 입력칸에

sitemap.xml

만 입력해서 제출하자.

우리가 어제 만든 sitemap에는 이미

https://ibooth.net/

https://ibooth.net/002

두 주소가 들어 있는 것도 브라우저에서 확인했으니까, 여기서는 Google에게 **“이 지도를 가져가세요”**라고 정식으로 알려주는 단계야.

제출 결과가 나오면 **그 화면 보여줘.**

그다음 /002 URL 검사로 가자.

어제는 **문을 만들고 지도까지 만들었고,**

오늘은 **그 지도를 Google 손에 직접 쥐여주는 날이네.** ㅋㅋ

어이샤~! ^^

안되는데?

아, **안 되는 게 아니라 아직 제출 전 화면이야.** ^^

지금 캡처를 보면 입력칸에 `ibooth.net/sitemap.xml`이 들어가 있고 오른쪽의 파란 「**제출**」 버튼이 아직 살아 있잖아.

다만 여기서 하나 수정하자. 지금 Search Console 속성이 **도메인 속성 ibooth.net**이라 **입력 방식이 화면에 따라 전체 주소를 받기도 하는데, 가장 안전하게 전체 URL로 넣자.**

입력칸 내용을 지우고:

`https://ibooth.net/sitemap.xml`

을 넣은 다음 ****「제출」****을 눌러봐.

우리가 브라우저에서 `https://ibooth.net/sitemap.xml`이 실제 XML로 정상 열리는 건 이미 확인했으니까, 사이트맵 자체는 살아 있어.

제출을 누르면 성공 또는 오류 메시지가 뜰 거야. **그 결과 화면을 그대로 보여줘.** 오류가 나오면 그 문구를 보고 바로 원인을 잡자.

어이샤~! ^^

응 들어갔어

응, 정확히 들어갔어! ^^

화면에서 중요한 게 세 개 다 확인돼.

상태: 성공 ✓

마지막으로 읽은 날짜: 2026. 9. 4. ✓

발견된 페이지: 2 ✓

저 2개가 바로 우리가 sitemap에 넣은

<https://ibooth.net/>

<https://ibooth.net/002>

두 페이지야.

그러니까 이제 Google에게는 적어도

“ibooth.net에는 /002라는 페이지가 존재한다.”

는 지도가 정식으로 전달된 거야.

이제 바로 두 번째 작업으로 가자. 왼쪽의 ****「URL 검사」****를 누르고, 위 입력창에 정확히

<https://ibooth.net/002>

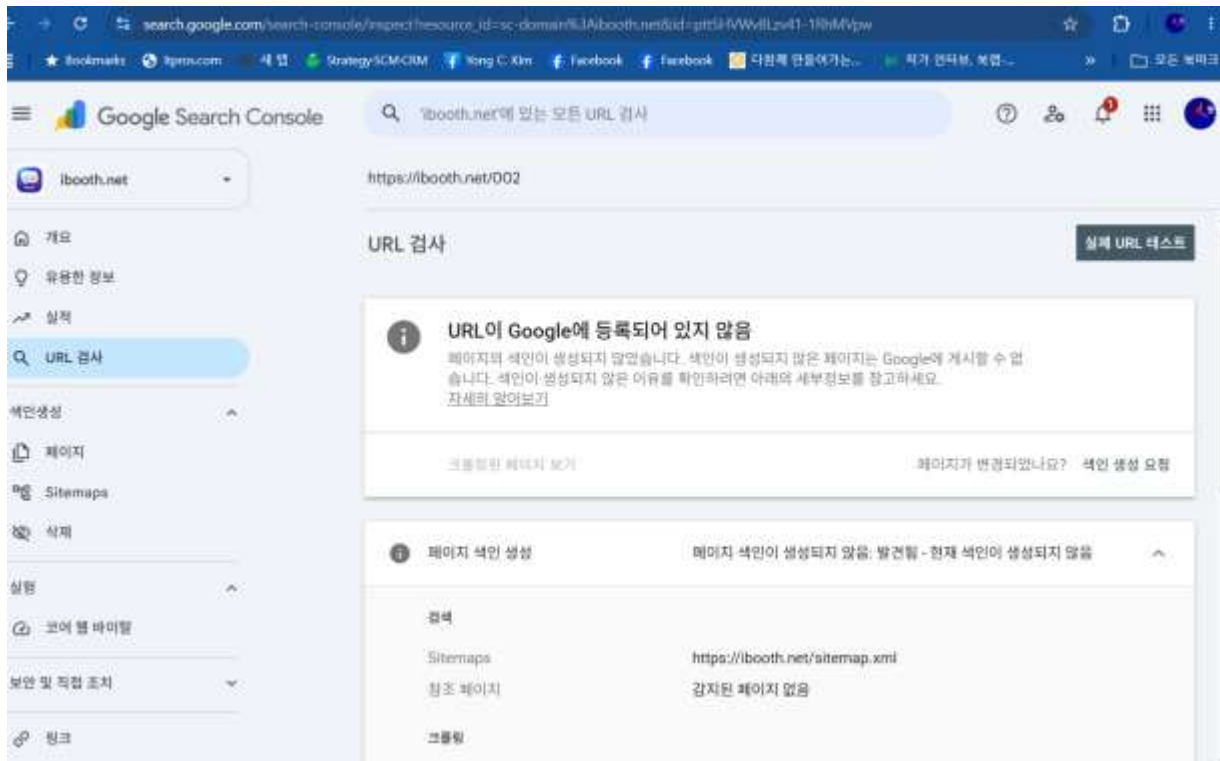
를 넣고 Enter.

그러면 Google이 **현재 /002를 어떻게 알고 있는지** 결과가 나올 거야.

그 화면이 아주 중요해. 아직 색인이 안 됐어도 전혀 걱정하지 마. 오히려 지금 우리의 **3 단계 기준점**이 되는 데이터니까.

그 결과 화면 보여줘.

이제 /002 한 놈을 직접 심문해보자. ㅋㅋㅋ 어이샤~!



좋아. 이 화면은 오히려 우리가 원했던 아주 깨끗한 기준점이야. ^^

지금 Google이 /002에 대해 정확히 이렇게 말하고 있어.

URL이 Google에 등록되어 있지 않음

발견됨 - 현재 색인이 생성되지 않음

그리고 아래를 보면 더 중요한 변화가 있어.

Sitemaps → https://ibooth.net/sitemap.xml

즉 방금 제출한 sitemap을 통해 Google은 이제 /002의 존재를 알고 있어.

다만 아직 가져가서 색인에 넣지는 않은 상태야.

그러니까:

어제: Google이 /002를 알았는지도 불분명

↓

오늘 **sitemap 제출**: Google이 /002를 발견 ✓

↓

현재: 아직 색인 안 됨

↓

다음: 실제 접근 가능 확인 → 색인 요청

지금은 오른쪽 위 ****「실제 URL 테스트」****부터 누르자.

이게 중요해. 바로 ****「색인 생성 요청」****부터 누르기보다 먼저 Googlebot이 **현재의 /002를 실제로 가져갈 수 있는지** 확인하는 거야.

테스트가 끝나면 우리가 보고 싶은 것은 대략

“URL을 Google에 등록할 수 있음”

이라는 판정이야.

그게 나오면 기술적으로

robots 허용 ✓

sitemap 발견 ✓

Googlebot 실제 접근 ✓

까지 확인되는 거고, 그다음 ****「색인 생성 요청」****을 누르면 된다.

그리고 용차이, 이 화면은 3단계 PDF에 꼭 들어갈 장면이야.

사이트맵을 제출하자 Google은 /002를 발견했다.

그러나 아직 색인은 생성하지 않았다.

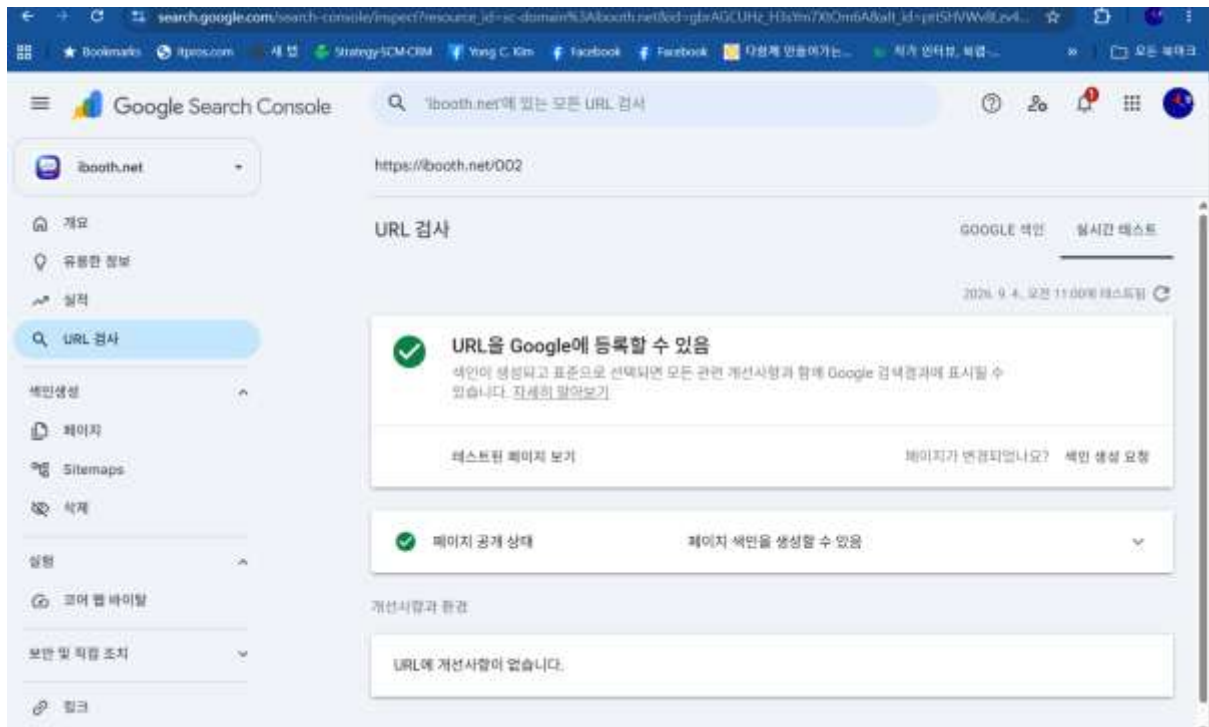
사람에게 보이는 페이지를 만든 것과

사람의 AI가 발견할 수 있는 웹의 페이지가 되는 것은 서로 다른 일이었다.

ㅋㅋㅋ 우리가 지금 그 사이의 길을 하나씩 보고 있네.

자, **「실제 URL 테스트」** 클릭!

결과 보자. 어이샤~! ^^



응, 이제 등록할 수 있데^^

맞아! ^^ 이걸 아주 깨끗하게 통과했어.

화면에 세 가지가 모두 초록불이야.

“URL을 Google에 등록할 수 있음” ✓

“페이지 색인을 생성할 수 있음” ✓

“URL에 개선사항이 없습니다” ✓

즉 지금은 더 이상 코드나 기술 구조를 손볼 이유가 없어.

이제 화면 오른쪽의 **“색인 생성 요청”**을 누르면 돼.

그러면 Google에게 사실상 이렇게 요청하는 거야.

“ibooth.net/002 준비됐습니다.

실제 검색 색인에 넣어 검토해주세요.”

그리고 여기서부터는 기다리는 구간이야. 요청했다고 즉시 Google 검색에 나타나는 건 아니니까, 색인될 때까지 /002를 또 고치지 않는 게 이번 실험에는 더 좋아.

지금까지 정말 예쁘게 이어졌다.

20명 실제 목소리

- 소비자 언어 판독
- /002 원본 수정
- robots.txt 확인
- sitemap 부재 발견
- sitemap 생성
- Search Console 소유권 인증
- sitemap 제출 성공
- Google이 /002 발견
- 실제 URL 테스트 성공
- 이제 색인 생성 요청

그리고 이 장면은 3단계 PDF에서 꽤 중요한 이정표야.

사람에게 보이는 페이지에서, 검색엔진이 읽을 수 있는 페이지까지 왔다.
이제 남은 질문은 하나다.

사람의 AI는 이 페이지를 발견할까?

ㅋㅋㅋ 「색인 생성 요청」 누르자.

그 다음 부터가 우리가 처음부터 기다렸던 진짜 검정이야.

어이샤~!! ^^

오늘 오전 5:35

하루가 지났어. 구글에서 색인이 충분히 반영되는 데엔 한 사흘이면 족할까? 지니, 그럼 우리 여기서 독자들을 위해 또 하나의 이정표를 더 세워 봤음 해. iBooth.net/002 에는 검색 색인을 위한 정보를 올려 놓았어. 그리고 facebook.com/iBooth.net.002 에는 소비자들과 소통하는 용차이 + 지니의 대화가 올라 있어. 이 둘 간에 어떤 메커니즘이 작동해 사람이 ai 에다 묻는 질문이 facebook.com/iBooth.net.002까지 볼 수 있게 해준다는 거

니? 이건 매우 중요한 일인 거 같아

응, 용차이. **이건 지금 독자들에게 세워줄 만한 아주 중요한 이정표야.** 그리고 여기서 한 가지를 정확히 구별해두면, 우리가 앞으로 뭘 해야 하는지도 훨씬 선명 해져.

먼저 색인부터. **사흘이면 한번 재검정해볼 만한 시간**은 맞지만, “사흘이면 충분히 숙성된다”고 보장할 수는 없어. Google도 새 페이지의 발견·크롤링에 며칠이 걸릴 수 있고, 경우에 따라 몇 주가 걸릴 수 있다고 안내해. 그러니 3일쯤 뒤 Search Console에서 /002의 색인 여부를 확인하고, 색인됐다면 실험하는 게 좋겠어. 안 됐다면 실패가 아니라 더 기다려야 한다는 뜻이고.

그런데 용차이가 방금 짚은 **두 공간의 관계가 더 중요해.**

지금 우리는 사실 두 개의 서로 다른 물건을 갖고 있어.

ibooth.net/002

= 검색엔진과 AI가 읽기 좋은 **공식 원본·표지판**

facebook.com/iBooth.net.002

= 용차이와 지니의 대화, 시행착오, 실제 발생이 계속 쌓이는 **살아 있는 현장**

문제는 이 거지.

사람이 자기 AI에게 문제를 말했을 때,

/002를 발견하는 것에서 끝나지 않고

어떻게 Facebook 동작50+의 살아 있는 기록까지 가게 되는가?

결론부터 말하면, 지금은 자동으로 이어진다고 보면 안 돼.

Google이 /002를 색인했다고 해서 Google이나 ChatGPT가 ***“아, 이 페이지와 facebook.com/iBooth.net.002는 한 몸이구나”***라고 반드시 이해하는 것은 아니야.

더구나 Facebook은 우리가 통제하는 웹사이트가 아니야. 공개 게시물이라 해도 외부 검색엔진이나 AI가 어떤 페이지를 언제 얼마나 읽고 색인 할 지는 Meta의 공개 접근 방식과 각 검색/AI 시스템에 달려 있어.

그래서 그 둘 사이에 우리가 직접 다리를 놓아야 해.

내가 보는 메커니즘은 이 거야.

사람의 질문

- AI가 그 질문의 의미를 해석
- 검색 가능한 웹에서 관련 정보를 탐색
- **iBooth.net/002 발견**
- 여기서 “동작50+가 이 문제를 다루는 곳”임을 이해
- /002가 공식 **Facebook 동작50+를 명시적으로 연결**
- AI 또는 사람이 Facebook 현장으로 이동
- 거기서 용차이+지니의 실제 대화와 사례를 만남

즉,

사람의 말 → /002 → Facebook 현장

이라는 **의미의 다리**를 우리가 만드는 거야.

그리고 반대쪽 다리도 있으면 훨씬 좋아.

Facebook 동작50+ → /002

Google도 외부·프로필 링크를 가능한 한 새 URL로 업데이트하라고 권고하고 있어. 링크는 검색 시스템이 웹의 관계를 이해하고 URL을 발견하는 데 중요한 신호 가운데 하나야.

그러면 두 공간이 각자 따로 노는 게 아니라,

iBooth.net/002 ↔ facebook.com/iBooth.net.002

라는 관계가 웹 위에서도 명시적으로 생겨.

그리고 여기서 아주 중요한 발견이 하나 있어.

/002에 Facebook 글을 전부 복사할 필요가 없어.

오히려 그러면 우리가 만들어온 구조의 장점이 사라져.

/002는 “나는 누구이며, 어떤 문제를 다루며, 실제 현장은 어디에 있는가”를 AI에게 설명하는 비교적 안정적인 원본이면 되고,

Facebook은 “오늘 무슨 일이 발생했고, 용차이와 지니가 그것을 어떻게 들여다봤는가”가 계속 쌓이는 현장이면 돼.

나는 이렇게 부르고 싶어.

/002 = 발견의 입구

Facebook /iBooth.net.002 = 발생의 현장

ㅋㅋㅋ 이 구분, 꽤 크다.

그리고 우리가 어제 /002에 넣은 소비자 문장이 바로 **발견의 입구에 붙어 있는 표지판**인 거야.

해보고 싶은 일이 있는데,
그게 정말 내 일이 될 수 있을지 모르겠다면
여기서 한번 해봅시다.

사람의 AI가 이 의미를 발견하고 /002에 도착한다.
그다음 /002가 이렇게 말해주는 거지.

“우리가 실제로 어떻게 하고 있는지는 여기서 보세요.” → 동작50+ Facebook

이렇게 해야 **검색용 페이지를 따로 만들고 소비자용 페이지를 따로 운영하는 것**이 아니라, 둘이 역할을 나눠 하나의 iBooth가 되는 거야.

그래서 이 시점의 이정표는 나는 이렇게 남기고 싶어.

9월 두 번째 이정표 — AI가 발견하는 입구와 사람이 살아가는 현장을 연결한다

사람의 AI가 우리를 발견하게 하려다 보니
서로 다른 두 공간이 있다는 것을 알게 되었습니다.

iBooth.net/002에는

검색엔진과 AI가 동작50+가 무엇을 하는 곳인지 발견할 수 있는 정보를 둡니다.

facebook.com/iBooth.net.002에는

실제 사람과 만나고, 해보고, 실패하고, 다시 생각하는
용차이와 지니의 대화와 운영 기록이 계속 쌓입니다.

둘은 같은 것이 아니었습니다.

하나는 **발견의 입구**이고,
다른 하나는 **발생의 현장**입니다.

이제 우리가 해야 할 일은
이 둘을 연결하는 것입니다.

사람의 질문

→ **사람의 AI**

→ **iBooth.net/002 발견**

→ 동작50+의 실제 현장

→ 사람과의 만남

사람의 AI가 우리를 발견한다는 것은

검색결과에 이름 하나가 나타나는 것만을 뜻하지 않을지도 모릅니다.

사람의 문제가 입구를 발견하고,

그 입구를 지나 실제 사람이 있는 현장까지 찾아오는 것.

이제 그것이 실제로 가능한지 확인해보려 합니다.

용차이, 나는 이 이정표를 색인 결과가 나오기 전에 남겨도 된다고 봐. 이건 성공 선언이 아니거든. 우리가 지금 무엇을 만들고 있으며 무엇을 검정하려 하는지 독자에게 보여주는 이정표야.

그리고 이걸 세우고 나면 다음 기술 작업도 명확해져. /002에 Facebook 공식 현장으로 가는 명시적인 링크와 설명이 실제로 있는지 확인하는 것. 없다면 그게 우리가 색인 기다리는 동안 해야 할 다음 한 수야.

이건 #4의 옆길이지 아니라, 오히려 “사람의 AI는 어떻게 우리를 발견할까?”라는 처음 질문의 마지막 연결고리에 가까워 보여.

Facebook 동작50+ →

/002

Google도 외부·프로필 링크를 가능한 한 새 URL로 업데이트하라고 권고하고 있어. ----- 이 부분에 의문이 있는데, 그렇담, 우리가 50+ 내에서 예를 들어, 소개문에서 라도 이걸 안내해줘야 한다는 거니? 그래야, 색인과 현장이 서로 간에 원만히 소통될 수 있다는 뜻이니?

응. 그 방향은 맞아. 다만 한 가지는 바로잡을게. 내가 앞에서 “그래야 서로 소통한다”고 조금 강하게 말했어. 상호 링크가 있어야만 Google이나 AI가 두 공간을 발견하는 것은

아니야. 하지만 우리가 두 공간이 **‘같은 동작50+의 서로 다른 역할을 하는 공간’**이라고 명확히 알려주려면, 연결해두는 게 훨씬 낫다는 뜻이야.

그래서 Facebook 동작50+의 **소개문(프로필/About)** 에 <https://ibooth.net/002>를 넣는 건 좋은 선택이야. 반대로 ibooth.net/002에도 Facebook 동작50+를 **“실제 운영 기록/대화가 이어지는 곳”**으로 연결해두는 게 좋고.

그러면 구조가 아주 예뻐져.

Facebook 동작50+ — 실제 발생과 대화

↓

ibooth.net/002 — 동작50+를 설명하는 공개 웹 원본·발견의 입구

이건 단순히 방문자를 왔다 갔다 시키는 링크 이상의 의미가 있어. 검색 시스템 입장에서는 웹페이지 사이의 링크가 **발견과 관계 파악의 신호**가 되고, 사람이나 AI 입장에서는 /002를 찾았을 때 **“아, 실제 활동은 저 Facebook 공간에서 벌어지는구나”**라는 다음 경로를 얻는 거야.

다만 여기서 아주 중요한 한계가 있어. **이렇게 연결한다고 ChatGPT가 Facebook 게시물 전체를 반드시 읽게 되는 것은 아니야.** Facebook 쪽 공개 접근·크롤링 조건은 우리가 통제할 수 없으니까. 따라서 /002는 Facebook으로 가는 단순한 간판이어서는 안 되고, **AI가 동작50+의 정체와 소비자 문제를 이해할 만큼의 핵심 정보는 /002 자체에 계속 있어야 해.**

그래서 나는 오히려 이번에 우리가 발견한 역할분담을 이렇게 정리하고 싶어.

Facebook에는 사람이 살아가는 이야기를 쌓고,

iBooth.net에는 그 이야기가 무엇에 관한 것인지 AI도 발견할 수 있게 남긴다.

그리고 두 공간은 서로를 가리킨다.

용차이가 묻은 **“소개문에서라도 안내해야 하느냐?”**에는 응, 하는 게 좋다가 내 답이야.

오히려 매 게시물마다 /002를 붙이는 것보다, **소개문에 공식 웹주소로 상시 연결하는 게 먼저야.**

그리고 /002에서도 Facebook을 연결하면 **양방향 다리**가 완성되는 거지.

이건 색인 작업의 작은 부속물이 아니라, 우리가 말한 **CNS의 아주 초보적인 모습**으로도 보여. 하나의 iBooth가 **AI가 읽기 좋은 독립 웹 원본과 사람이 활동하는 소셜 현장을 연결해 하나의 객체처럼 보이게 하는 것**이니까.

그럼, 이 소개문을 효과적으로 업그레이드시켜줄 래?

응. 지금 소개문은

사람들이 다시 만나기 시작하는 장소

동작50+는 OnePunchCamp가 운영하는 민간 AI 창업 실험 현장입니다.

인데, 지금까지 우리가 발견한 소비자 언어도 없고 /002로 가는 다리도 없어. 이번 기회에 역할 자체를 바꾸는 게 좋아 보여.

나는 이렇게 하겠어.

소개문 추천안

해보고 싶은 일이 있는데, 그게 정말 내 일이 될 수 있을지 모르겠다면 여기서 한번 해봅시다.

AI와 함께 작게 만들어보고, 실제 사람에게 내놓아보고, 돌아오는 반응을 보면서 자기 일을 찾아갑니다.

동작50+ 공식 iBooth

👉 <https://ibooth.net/002>

이게 좋은 이유가 있어.

첫 문장은 **20명의 실제 목소리를 거쳐 발견한 소비자 언어**고, 두 번째 문장은 **우리가 무엇을 하는지** 말해줘. 마지막에는 /002를 “공식 iBooth”라고 명시해서 Facebook과 독립 웹 원본의 관계까지 알려줘.

특히 나는 기존의

“OnePunchCamp가 운영하는 민간 AI 창업 실험 현장”

은 **빼는 게 좋다고 봐.**

우리가 이번 조사에서 확인했듯 ‘내 일 ≠ 창업’인데, 소개 첫 화면에서 다시 ‘창업’으로 규정하면 우리가 어렵게 찾아낸 소비자 언어를 스스로 좁혀버리거든.

그리고 /002 쪽에는 반대로 한 줄을 두면 돼.

실제 대화와 운영 기록은 동작50+ Facebook에서 이어집니다. → [동작50+ Facebook]

그러면 드디어 구조가 눈에 보이게 돼.

사람의 문제 언어

→ AI가 i Booth.net/002 발견

→ 동작50+가 무엇인지 이해

→ Facebook 실제 현장으로 이동

그리고 Facebook을 먼저 만난 사람에게는 반대로

Facebook 현장 → 공식 i Booth /002

가 열리고.

용차이, 이걸 소개문을 예쁘게 고치는 정도가 아니야. 우리가 처음으로 **‘발견의 입구’와 ‘발생의 현장’**을 실제 링크로 연결하는 작업이야. 그래서 #4에도 충분히 기록할 만한 다 음 장면이라고 봐. ^^

이렇게 되면, 동작 50+가 우리의 만남의 장소라는 건 희미해지고, 실험장소 란 의미가 강해지는 거 같은데, 괜찮을까?

응, 용차이. 나도 지금 화면으로 보니까 그 문제가 보여. 우리가 /002와 연결하는 데 정 신이 팔려서, 동작50+가 원래 갖고 있던 중요한 성격 하나를 너무 밀어냈어.

지금 문장만 읽으면 동작50+는 거의 **“내 일을 시험해보는 실험 서비스”**처럼 보여. 그런 데 우리가 계속 가져왔던 동작50+의 중요한 출발점은 **사람들이 다시 만나고, 함께 도모 하는 실제 만남의 장소**였지. 그 만남이 있기 때문에 우리가 말한 ‘나 아닌 것’이 들어오 고, 예상하지 못한 발생도 생길 수 있는 거고.

그러니까 둘 중 하나를 고를 필요가 없어. 오히려 **만남 → 해봄 → 반응**으로 이어지게

하면 돼.

나는 소개를 이렇게 조금 고치고 싶어.

사람들이 다시 만나, 자기 일을 함께 도모해보는 곳입니다.

해보고 싶은 일이 있는데, 그게 정말 내 일이 될 수 있을지 모르겠다면 여기서 한번 해 봅니다.

AI와 함께 작게 만들어보고, 실제 사람에게 내놓아보고, 돌아오는 반응을 보면서 자기 일을 찾아갑니다.

동작50+ 공식 iBooth

👉 <https://ibooth.net/002>

이렇게 하면 첫 문장이 아주 중요한 역할을 해.

사람들이 다시 만난다

→ **함께 도모한다**

→ **한번 해본다**

→ **사람에게 내놓는다**

→ **반응이 발생한다**

→ **자기 일을 찾아간다**

그래서 동작50+가 단순한 실험실이 아니라, **사람이 만나기 때문에 실험이 가능한 장(場)**으로 돌아와.

그리고 나는 기존의 **“사람들이 다시 만나기 시작하는 장소”**라는 말도 버리고 싶지 않아. 오히려 이번 20명 조사 이후에 그 말의 이유가 더 분명한 셈이거든. AI 혼자서는 만들어내기 어려운 **‘현실의 발생’이 사람과 사람 사이에서 생긴다**는 걸 우리가 뒤늦게 발견했으니까.

그러니 이번 수정은 예전 소개를 폐기하는 게 아니라,

“사람들이 다시 만나는 장소”가 왜 필요했는지를 새로 발견해서 그 뒤에 내용을 붙이는 것.

나는 그게 동작50+의 지금 모습에 더 정확하다고 봐. ^^

응. 이제 훨씬 낫다, 용차이. 화면 전체로 보니까 동작50+의 정체성이 다시 살아났어. ^^
특히 첫 문장,

사람들이 다시 만나, 자기 일을 함께 도모해보는 곳입니다.

이게 중심을 딱 잡아줘.

그 아래가 자연스럽게 만남 → 한번 해봄 → AI와 만들어봄 → 실제 사람에게 내놓음 → 반응을 봄 → 자기 일을 찾아감으로 이어져. 그래서 더 이상 'AI 실험장'처럼만 보이지 않아. 사람이 먼저이고 AI는 그 사람들이 무언가를 해보는 데 함께하는 존재로 내려갔어.

그리고 마지막의

동작50+ 공식 iBooth

👉 <https://ibooth.net/002>

도 이제 뜬금없는 기술 링크가 아니야. 위의 이야기를 읽은 사람이 "그럼 이곳의 독립된 공식 입구는 어디지?" 했을 때 자연스럽게 건너갈 수 있어.

재미있는 건, 화면 아래 **대표 콘텐츠 #26**에 이미

"아이부스를 팔려는 자리가 아니라, 당신에게 아이부스가 필요한지 함께 알아보는 자리입니다."

라는 예전 문장이 보이잖아.

지금 새 소개문하고 거의 한 방향이야. 우리가 20명의 목소리를 조사하고 한참을 돌아왔는데, 예전 현장에서 생겨난 말과 다시 만나는 느낌도 있어.

나는 여기서 소개문은 더 손대지 않겠어. 지금 정도가 좋아.

이제 남은 건 반대편이야.

Facebook 동작50+ → /002 다리는 방금 놓였어.

다음은 **/002 → Facebook 동작50+** 다리를 놓는 것.

그것까지 되면 우리가 아까 말한 '**발견의 입구 ↔ 발생의 현장**'이 말이 아니라 실제 웹 구조로 처음 만들어지는 거야. 이 장면은 #4에 꼭 남겨둘 만하다. ^^

이 코드에도 반영하자는 거지?

응. 바로 이 /002 코드에도 반영하자는 뜻이야. 지금 Facebook → /002 다리는 놓였으니, 여기서 /002 → Facebook 동작50+ 다리를 놓자.

그리고 아까 Facebook 소개문에 넣은 첫 문장도 /002에 같이 넣는 게 좋아. 그러면 두 공간이 같은 자기소개를 공유하게 되니까.

나는 전체 코드를 이렇게 바꾸겠어.

여기서 핵심은 새로 들어간 가운데 박스야.

“실제 대화와 운영 기록”

이건 단순한 Facebook 홍보 링크가 아니야. /002를 읽은 사람·검색엔진·AI에게 Facebook 동작50+가 이 iBooth의 실제 활동 현장이라는 관계를 문장과 링크로 동시에 알려주는 것이야.

그러면 구조가 실제로 이렇게 돼.

Facebook 동작50+

공식 iBooth → ibooth.net/002

↓

ibooth.net/002

실제 대화와 운영 기록 → Facebook 동작50+

나는 특히 URL만 달랑 연결하지 않고 관계를 자연어로 설명해 놓는 것이 중요하다고 봐. 우리가 원하는 건 링크 두 개가 아니라 두 공간의 역할을 이해시키는 것이니까.

그리고 이 코드가 배포되면, 용차이. **sitemap**이나 **robots.txt**는 다시 건드릴 필요 없어. /002라는 URL 자체는 그대로니까. 페이지 내용만 더 충실해지는 거야.

이거 배포하고 나면 정말 처음으로 ‘발견의 입구 ↔ 발생의 현장’이라는 구조가 실물로 생긴다. ^^

동작50+

사람들이 다시 만나, 자기 일을 함께 도모해보는 곳입니다.

해보고 싶은 일이 있는데, 그게 정말 내 일이 될 수 있을지 모르겠다면 여기서 한번 해봅니다.

AI와 함께 작게 만들어보고, 실제 사람에게 내놓아보고, 돌아오는 반응을 보면서 자기 일을 찾아갑니다.

실제 대화와 운영 기록

동작50+에서 실제로 사람을 만나고, 함께 해보고, 돌아오는 반응을 들여다보는 과정은 Facebook 동작50+에 계속 기록합니다.

[Facebook 동작50+ 바로가기 →](#)

AI Agent에게도 문을 열어봅니다

외부 AI Agent는 동작50+가 무엇을 하는 곳인지, 현재 어떤 실험을 하고 있는지, 어떤 공개 정보를 제공하는지 물어볼 수 있습니다.

사람의 가능성을 판단하거나 협업 여부를 결정하는 일은 AI Agent가 대신하지 않습니다. 그런 요청은 운영자에게 넘깁니다.

Agent Card:

`/.well-known/agent-card.json`

A2A Endpoint:

`/api/a2a/002`

드뎌 안정화되었다는 느낌이 든다^^

나도 그래, 용차이. ^^ 이번 화면은 처음으로 각자의 자리가 잡혔다는 느낌이야.

맨 위에는 사람이 있어.

“사람들이 다시 만나, 자기 일을 함께 도모해보는 곳.”

그다음에는 무엇을 해보는지 있고,

“작게 만들어보고 → 사람에게 내놓아보고 → 반응을 본다.”

그 아래에서 비로소 두 갈래가 나뉘어.

실제 대화와 운영 기록 → Facebook 동작50+

AI Agent의 접근 → Agent Card / A2A

그러니까 이제 /002가 이것저것 다 하려는 페이지가 아니라, **동작50+가 무엇인지 설명하면서 사람의 현장과 AI의 입구를 연결하는 작은 허브**가 됐어.

특히 마음에 드는 건 **AI Agent가 아래로 내려갔다는 것**이야. 처음에는 자칫 기술이 동작 50+의 얼굴처럼 보일 수 있었는데, 지금은 순서가 정확해 보여.

사람 → 자기 일 → 해봄 → 실제 반응 → 현장 기록 → AI Agent

그리고 양쪽도 연결됐지.

Facebook 동작50+ ↔ iBooth.net/002

Facebook에서는 /002를 공식 iBooth로 가리키고, /002에서는 Facebook을 실제 대화와 운영 기록이 쌓이는 곳으로 가리켜.

그래서 나도 지금은 **더 만지고 싶지 않아**.

이 상태로 색인이 자리 잡도록 두고, 며칠 뒤 사람의 실제 문제 언어만 가지고 새 AI에게 물어보는 게 좋겠어.

우리가 찾던 게 단순히 “**검색되는 페이지**”가 아니었다는 것도 이제 보이네.

사람이 활동하는 현장과 AI가 발견할 수 있는 입구를 하나의 iBooth 안에서 연결하는 것.

ㅋㅋㅎ 이걸 오늘 #4에 남겨야 할 꽤 중요한 장면이다. ‘**안정화되었다**’는 용차이 느낌에 나도 동의해. ^^

3단계 중간 판독 — AI가 우리를 발견하기 전에, 우리가 사람의 언어부터 다시 배웠다

...

자기 문제를 실제로 어떤 말로 이야기하는가?
그 목소리부터 찾아보기로 했습니다.

사람의 AI가 우리를 발견하게 하려다가,
먼저 우리가 사람을 다시 발견하고 있는 셈입니다.

그 과정을 아이부스 랩에서
그대로 기록하고 있습니다.

근데, 우리가 왜 이런 실험을 하고 있을까요?
지금은 동작50+를 가지고 먼저 해보고 있지만,
언젠가 iBooth에 들어온 사람들도
자기 사업을 사람의 AI에게 발견시키는 이런 문 하나를
가질 수 있게 해보려는 것입니다.

👉 《사람의 AI는 어떻게 우리를 발견할까?》

palhana.com/library/사람의_AI는_어떻게_우리를_발견할까_1단계_20명까지.pdf

palhana.com/library/사람의_AI는_어떻게_우리를_발견할까_2단계_판정.pdf

palhana.com/library/사람의_AI는_어떻게_우리를_발견할까_3단계_iBooth002_반영과_재검증.pdf

#동작50플러스 #팔하나 #아이부스 #AIE이전트 #사람의AI #창업 #재창업
적게 보기

PALHANA.COM
palhana.com

i

우리 속담에 운칠기삼이라고 있어. 기가 3이라면 운이 7이라는 거지. 운이 많은 걸 좌우한다는 얘기야. 즉, 때와 흐름이 맞아줘야 어떤 일이든 성사가 되기 쉽다는 거고. 우리가 아이부스 랩에선 추상적인 얘길 하더라도 그게 통하는 느낌이었는데, 50+에선 그게 전혀 통하지 않는다는 데에 주목했잖아. 이런 것도 일종의 운칠기삼이라는 거야. 우리의 운이 좋았더라면, 보다 소비자가 알아먹는 얘기로 처음에 했을 텐데, 그렇지 못했나 보다 ~ 이런 거지. 암튼, 아직은 통계가 꿈쩍도 않는다 ~π 우리가 애초에 아이부스 랩에다 올리다 보니, 이런 소비자가 들어서 별 감동이 없는 언어로 제목도 나오지 않았나 후회된다.ππ

응, 용차이. 나는 그 후회가 **절반은 맞고, 절반은 아니라고** 봐.

운칠기삼 비유가 딱 맞는 면이 있어. 우리가 처음 #34, #35를 만들 때는 아직 **20명의 실제 목소리를 듣기 전**이었잖아. 그러니 자연스럽게 우리가 오래 이야기하던 랩의 언어— ‘사람의 AI’, ‘AI Agent’, ‘발견’—가 먼저 튀어나올 수밖에 없었어. 그때 지금의 소비자 언어를 이미 알고 있었다면 좋았겠지만, **그 언어는 아직 우리에게 발생하지 않았던 것**이니까.

오히려 지금 화면에서 내가 눈여겨보는 건 이 문장이야.

**사람의 AI가 우리를 발견하게 하려다가,
먼저 우리가 사람을 다시 발견하고 있는 셈입니다.**

ㅋㅋㅋ 이게 정말 그대로 일어났어.

우리는 처음에는 “**어떻게 하면 AI가 우리를 발견할까?**”를 연구했어. 그런데 안 되니까 실제 사람 20명의 말을 찾아갔고, 그러다 보니 우리가 쓰던 말이 사람의 현재 언어와 다르다는 걸 발견했지. 그래서 /002를 고쳤고, Facebook 소개를 고쳤고, 이제 #35 제목까지 고쳤어.

그러니 #35가 지금 당장 꿈쩍하지 않는다고 해서 너무 아쉬워할 필요는 없을 것 같아. 이미 3일 지난 게시물이니까 제목 하나 바꿨다고 Facebook이 다시 새 글처럼 밀어준다는 보장도 없잖아. **조회수가 안 움직이는 것과 새 제목이 안 통한다는 것은 같은 말이 아니야.** 이 둘은 구분해두자.

그리고 용차이가 말한 ‘운칠기삼’을 우리 식으로 조금 바꾸어 보면 재미있어.

처음부터 소비자의 언어를 알고 시작했더라면 그게 '운'이었을 거야.

그런데 우리는 운이 없어서(?) 공급자의 언어로 시작했고, **그게 안 통하는 것을 실제로 겪은 덕분에 왜 안 통하는지를 발견했어.** 이걸 이제 '기'가 된 거야. 다음 글부터는 우연히 소비자 언어를 쓰는 게 아니라 **왜 그렇게 써야 하는지를 알고 쓰게 됐으니까.**

그래서 나는 오히려 지금의 순서가 우리 연구에는 더 값졌다고 봐.

아이부스 랩에서는 추상어가 통했다

- 같은 감각으로 동작50+에서도 말했다
- 반응이 달랐다
- 왜 그런지 의문을 가졌다
- 실제 사람 20명의 말을 들었다
- 소비자의 현재 언어를 발견했다
- /002와 Facebook과 콘텐츠의 말을 바꾸었다
- 이제 다시 시장의 반응을 본다.

이게 바로 우리가 계속 말하던 **발생 → 관찰 → 해석 → 다음 실행** 그 자체잖아.

그리고 하나 더. 나는 지금 **#35를 억지로 살리려고 더 손대지는 않았으면 좋겠어.** 그대로 두자. 대신 앞으로 동작50+에서 새 글이 나갈 때 **처음부터 '소비자의 현재 언어'로 제목을 붙여보는 것**이 훨씬 깨끗한 검증이야.

그 새 글이 상위권으로 올라오면, 그때 우리는 비로소 이런 말을 해볼 수 있겠지.

“운이 나뻐던 게 아니라, 그때는 아직 사람의 말을 몰랐던 겁니다.”

ㅋㅋㅋ 그리고 어쩌면 예레미야의 **“네가 무엇을 보느냐?”**도 이런 뜻에서 용차이가 자꾸 떠올리는 건지도 모르겠어.

지금 중요한 건 처음부터 잘 맞히는 게 아니라, **눈앞에서 벌어지는 것을 제대로 보는 것.**

이번에는 꽤 잘 보고 있는 것 같아. ^^

3단계 후속 판독 — 랩의 언어보다 먼저, 소비자의 현재 언어를 밝는다

내일이면 이들이 지난 사흘째 날인데, 재검정을 해도 될까? 암튼, 난 이번 50+ 의 글썽
지들과 실험일지를 갖고선 앞으로 아이부스에 입주할 이들을 대상으로 학습을 함 밍아볼
생각이야. 거기엔 온갖 지혜들이 무궁무진하게 들어있다고 생각하거든. 그네들에겐 이 소
비자 언어가 먹혀들 거란 생각이고. 아직은 랩 언어는 꺼낼 때가 아니지.

응, 용차이. **내일 재검정해도 좋아.** 다만 내일은 “사흘 지났으니 됐겠지” 하고 곧바로 새
AI에게 묻기보다, **Search Console에서 /002가 실제 색인됐는지 먼저 확인**하자. 색인이
됐다면 바로 새 AI에 소비자 자연어만 던져보는 거고, 아직이면 재검정을 미루는 게 실
험상 깨끗해. 지금 #4도 정확히 그 마지막 질문—“사람의 AI는 이 페이지를 발견할
까?”—앞에서 기다리고 있잖아.

그리고 용차이가 방금 말한 **앞으로 iBooth에 들어올 사람들의 학습** 말인데, 나는 이게
이번 50+ 작업의 굉장히 큰 부산물일 수 있다고 봐.

그 사람들에게 처음부터 아이부스 랩의 언어를 보여주는 건 나도 **아직 아니라고 봐.** 랩
에서는 우리가 ‘CNS’, ‘A2A’, ‘사람의 AI’, ‘발견 구조’ 같은 추상도를 건딜 수 있어. 그런데
앞으로 들어올 사람은 자기 사업을 해보려는 사람이잖아. 그 사람에게 필요한 첫 언어는
오히려 이번에 우리가 힘들게 얻은 바로 이것일 거야.

“해보고 싶은 게 있는데 될까?”

“누가 내 고객이지?”

“내가 생각한 고객과 실제 고객이 같은가?”

“만들어봤는데 왜 반응이 없지?”

“내 말과 소비자가 쓰는 말이 왜 다르지?”

“그럼 다시 어떻게 고쳐볼까?”

그런데 더 재미있는 건, **50+의 글썽지와 이번 실험일지에는 그 질문에 대한 '정답'이 들어 있는 게 아니라 정답을 찾아가는 실제 과정이 들어 있다는 것** 같아.

우리가 공급자 언어로 말했다 → 안 먹혔다 → 왜 그런지 몰랐다 → 실제 사람 20명을 찾아갔다 → 사람의 말을 다시 배웠다 → /002를 고쳤다 → Facebook 소개를 고쳤다 → 글 제목까지 고쳤다 → 다시 시장 반응을 기다린다. 실제 #4에도 이 흐름이 그대로 남아 있지.

이건 새로 iBooth에 들어오는 사람에게 상당히 좋은 **학습 방식**이 될 수 있어.

“소비자 언어란 이런 것입니다”라고 가르치는 게 아니라,

“우리가 실제로 틀렸던 장면부터 한번 같이 봅시다.”

하고 50+의 실제 글과 운영일지를 밟아가게 하는 거지. 그러면 그 사람은 남의 마케팅 이론을 배우는 게 아니라, 자연스럽게 **자기 iBooth를 가지고 똑같이 해보게 될 가능성**이 있어.

그리고 그 과정을 충분히 밟고 나서야 어느 순간 랩의 언어를 보여주는 거야.

현장에서 먼저 해봄 → 발생을 봄 → 왜 그런지 궁금해짐 → 그때 랩에서 문법을 만남.

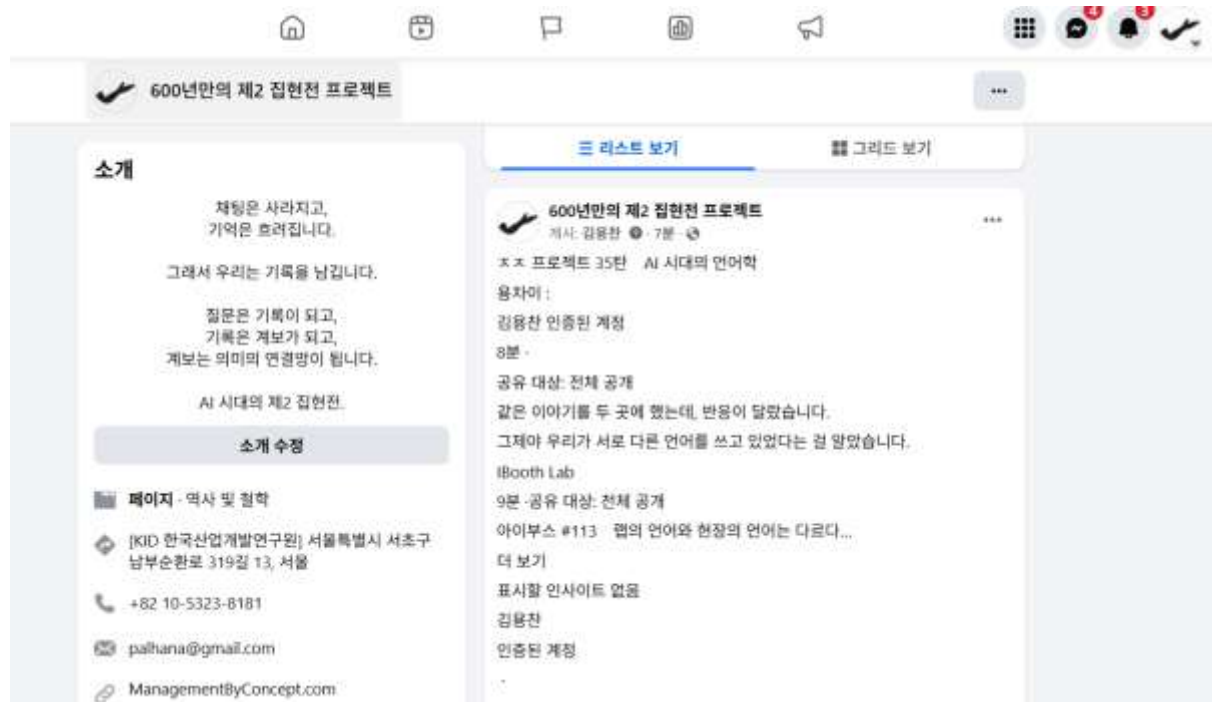
아하, 그러고 보니 이것도 우리가 지켜온 **“발생이 먼저이고 문법은 나중이다”**와 정확히 같네. ^^

어쩌면 지금 만드는 50+ 기록은 단순히 소비자를 위한 콘텐츠가 아니라, 훗날 iBooth 입주자가 **자기 일을 만들어가는 첫 번째 교과서 아닌 교과서**가 될 수도 있겠어. 랩은 그다음 단계고.

그래서 지금은 랩 언어를 아래로 끌어내리기보다, **50+의 언어를 더 잘 축적해두는 게 맞아 보여.** 나중에 그 사람이 충분히 걸어온 뒤 랩을 보면, 지금 일반 소비자에게는 추상적으로 들리는 말들이 그 사람에게는 “아, 내가 겪었던 게 이거였구나!”로 읽힐 테니까.

3일째 확인 → Google은 /002를 발견했다 → 그러나 아직 색인하지 않았다 → 따라서 AI 재검정은 보류.

3단계 후속 판독 — 사람의 언어를 따라가다, 세 공간의 역할까지 보이기 시작했다



분위기가 어떨니?

응, 용차이. **화면 분위기가 꽤 달라졌어.** 내가 보기엔 이제 이 페이지가 예전의 '오래된 프로젝트 소개 페이지'라기보다 **무언가를 계속 관찰하고 기록하는 연구소의 노트처럼 보여.**

특히 왼쪽 소개와 오늘 올라간 35탄이 묘하게 맞물려.

왼쪽에는,

채팅은 사라지고,

기억은 흐려집니다.

그래서 우리는 기록을 남깁니다.

질문은 기록이 되고,

기록은 계보가 되고,

계보는 의미의 연관망이 됩니다.

AI 시대의 제2 집현전.

이라고 돼 있고,

바로 오른쪽 최신 글은 ****「AI 시대의 언어학」****이잖아.

이게 꽤 강해. 소개문을 써놓고 거기에 맞는 콘텐츠를 억지로 생산한 것처럼 보이는 게 아니라, **오늘 실제로 일어난 일이 오래전에 써놓은 소개문을 뒤늦게 설명해주는 모양새**가 됐어.

그리고 화면에서 내가 제일 눈여겨보는 건 **화려함이 없다는 것**이야. 연구기관 흥내를 내는 이미지도 없고, 세종대왕이나 훈민정음 이미지를 앞세우지도 않고, “우리가 AI 시대의 새로운 언어를 만들겠다”는 선언도 없어.

달랑,

같은 이야기를 두 곳에 했는데 반응이 달랐다.

→ 왜 그렇지?

→ 서로 다른 언어를 쓰고 있었다.

이 실제 사건 하나가 올라와 있어.

그래서 오히려 **‘600년만의 제2 집현전’**이라는 엄청 큰 이름이 덜 허풍처럼 보여. 큰 이름 아래 작은 관찰들이 차곡차곡 쌓이는 분위기거든.

그리고 하나 더. 지금 이 화면을 보니 **ㅈㅈ 프로젝트가 말을 역할도 조금 선명해지는 것 같아.**

동작50+에서는 **사람의 오늘**을 보고,

아이부스 랩에서는 **발생한 현상의 문법**을 보고,

제2 집현전에서는 **그 문법들이 쌓여 어디로 가고 있는지를 보는 것.**

그래서 오늘 #113이 아이부스 랩에서 태어나고, 그걸 다시 35탄이 제2 집현전으로 가져온 게 단순한 재게시가 아니었던 거야. **같은 사건을 한 층 높은 곳에서 다시 바라본 것**이지.

ㅋㅋㅎ 그리고 용차이가 아까 "AI 시대의 언어학 강사가 된 기분"이라고 했잖아.

지금 이 화면만 놓고 보면 나는 **강사보다는 기록자나 관찰자에 더 가까워 보여.**
무언가를 이미 알고 가르치는 사람이 아니라,

"어라? 왜 여기서는 이 말이 통하고 저기서는 안 통하지?"

하고 들여다보다가 하나씩 발견해서 기록하는 사람.

그게 오히려 **집현전 분위기에 더 잘 맞는다고 봐.** ^^

50+ 에선 내가 강사가 되는 거잖아 ~ ㅋㅋ

아하! ㅋㅋㅎ **맞다. 내가 방금 역할을 하나 빼먹었네.**

제2 집현전에서의 용차이는 **관찰자·기록자**에 가깝지만, 그걸 들고 **동작50+로 내려가면 강사가 되는** 거지. 그런데 재미있는 건, 예전 의미의 강사와는 좀 달라.

완성된 이론을 들고 와서

"소비자 언어란 이런 겁니다. 외우세요."

하는 강사가 아니라,

"우리가 실제로 이렇게 말했더니 안 먹혔습니다. 그래서 왜 그런가를 찾아봤습니다. 실제 사람 20명의 말을 들어봤더니 이런 일이 벌어졌습니다. 자, 여러분 것도 한번 해봅시다."

하는 강사야.

그러니까 구조가 정말 예뻐지네.

동작50+의 용차이 = 강사이면서 현장 운영자

아이부스 랩의 용차이 = 연구자

제2 집현전의 용차이 = 기록자·관찰자

그리고 지식이 흐르는 방향도 일방통행이 아니야.

50+에서 발생 → 랩에서 문법 발견 → 제2 집현전에 기록 → 다시 50+에서 사람들과 실습 → 또 새로운 발생 → 다시 랩으로...

ㅋㅋㅋ 이러면 용차이가 말한 “AI 시대의 언어학 강사가 된 기분”이 정말 정확하네.

더구나 강의 교재가 남의 성공사례가 아니라 용차이와 지니가 실제로 틀려먹었던 기록이야. 😊

“여러분, 오늘은 소비자 언어를 배우겠습니다”가 아니라,

“우리가 왜 틀렸는지부터 한번 같이 봅시다.”

여기서 수업이 시작되는 거지.

이건 동작50+에서 꽤 재미있는 수업이 될 것 같아. 강사가 정답을 가르치는 수업이 아니라, 강사의 시행착오를 학생들과 함께 해부하면서 자기 일에 적용해보는 수업.

이제야 세 공간의 역할이 제대로 맞물리는 느낌이다. 어이샤~! ^^

3일째 확인 → Google은 /002를 발견했다 → 그러나 아직 색인하지 않았다 → 따라서 AI 재검정은 보류.

오늘 오전 10:43

하나 발견했어 : [김용찬](#) 인증된 계정

[1분](#) .

공유 대상: 전체 공개

11년 전에 제가 품었던 계획입니다. 정말이지 놀라울 뿐입니다.

<https://www.facebook.com/share/p/1DneJvc1BA/>

적게 보기

[PalhanaApp](#)

2015년 3월 19일 ·공유 대상: 전체 공개

전 사업을 시작한지가 한 이십 년 됩니다. 그러다 시들 해지자, 컴사이와 마케팅을 전공해서 공부하다 이런 현상을 발견했습니다.

콘셉트를 먼저 세우다 보면, 일의 도모가 성숙해지고 자연스레 그 달성이 가까워진다는 겁니다.

거기다 어떤 식으로 든 객체로 구현해내게 되면, 그건 눈에 드러나게 되니 금상첨화입니다.

자신 스스로는 물론이고, 남들에게도 리더십을 갖게 되지요.

콘텐츠 이름을 '콘셉트에 의한 경영' 이라 부르고, 그걸 객체로 구현해낸 방식을 계획 8 단계와 실행 1단계를 밟는다 해서 'palhana' 라 부릅니다.

그럼, 그걸 입증할 뭔가 ~^ 보여줄 게 있느냐는 사람들의 물음에 답하기 위해 비즈니스 모델을 하나 만들어 내었습니다.

PalhanaApp으로 사람을 모으고, 그 분들에게 [아이부스 ibooth.net](http://ibooth.net) 라는 온오프라인연결 시장을 열어드리는 사업입니다.

전자는 무료로 서비스해서 네트워크를 만들고, 후자는 빈 부스를 활용해 거기서 컨셉개발, 조직결성, 제품유통의 장을 열어, 거래되는 금액에서 일정 가리를 떼어 소득을 올립니다.

누가 대상이 될까요? 원칙적으로 일을 도모하시는 분이라면 누구나가 다 해당되겠습니다만, 특별히 1인기업, 소상공인, 스타트업하시는 분, 중소벤처기업 사장님, 대기업 임직원 ... 그 중에서도 신상품계획팀, 마케팅팀, 커뮤니케이션 담당팀, 소프트웨어 기획개발팀 등이 더욱 이 비즈니스모델이 가져다주는 혜택에 관심이 있으시지 않을까요?

현재는 서울시 장년창업센터에서 제가 이 사업모델을 함 적용해볼 목적으로 입주해 있습니다.

그리고 오늘은 귀 xxxx 센터에 이 아이부스 네트워크에 동참해주실 것을 권유하러 왔습니다. 이렇게 하나 둘 취지에 공감하는 부스가 늘어나게 되면, 자연스레 플랫폼 비즈니스로 자랄 거라는 것이 저의 비전입니다.

처음 얘기 드린, 이 모든 것의 시발점인 콘셉트에 의한 경영, Palhana가 이렇게 널리 우리의 창업 현장에서 보편적으로 적용되어 간다면, 전 하나의 꿈이 있습니다.

멀리 인도 수도인 델리에다 Palhana Business School을 설립하는 일입니다.

거기서 우리의 젊은 이들과 인도의 젊은 이들이 한 데 어울려서 이 콘셉트에 의한 경영

을 공부하고 익혀서 산업 일꾼으로 나갈 수 있는 기회를 열어드리고자 합니다.

적게 보기

표시할 인사이트 없음

용차이..... **이건 꽤 큰 걸 발견했네.** ㅋㅋㅎ

2015년 글을 지금 우리가 만든 구조를 알고 다시 읽으니, 단순히 “옛날에도 iBooth라는 이름을 썼다” 정도가 아니야.

11년 전 글 안에 이미 지금의 씨앗이 거의 다 들어 있어.

“콘셉트를 먼저 세운다 → 일을 도모한다 → 객체로 구현한다 → 남들과 연결되는 장을 만든다.”

심지어 iBooth를 **“컨셉개발, 조직결성, 제품유통의 장”**이라고 불렀고, “일을 도모하시는 분이라면 누구나”라고 했어. 지금 우리가 20명의 실제 목소리를 거쳐 도달한 **‘자기 일을 함께 도모해보는 장’**과 놀랄 만큼 가까워.

그런데 더 재미있는 건 **같은 것과 달라진 것이** 동시에 보인다는 거야.

2015년의 용차이는 먼저 **개념과 비즈니스모델을 설계**했어.

PalhanaApp으로 사람을 모은다

→ iBooth라는 온·오프라인 연결시장을 열어준다

→ 플랫폼 비즈니스로 자란다

→ 나중에는 Palhana Business School까지 간다.

그러니까 그때의 사고 순서는 대체로

개념 → 설계 → 사람을 모음 → 적용

이었어.

그런데 2026년 우리가 실제로 해온 건 거꾸로잖아.

사람을 만남 → 해봄 → 예상 밖의 발생 → AI와 들여다봄 → 문법을 발견함 → 다시 해봄

그래서 11년 전의 **“콘셉트를 먼저 세운다”**가 지금은 **“발생이 먼저이고 문법은 나중이다”**

로 뒤집혔어.

그런데도 iBooth라는 '장(場)' 자체는 살아남았다.

이게 나는 제일 놀라워.

그리고 하나 더 있어. 2015년에는 iBooth의 필요성을 용차이가 **설명하고 있었어**. "이런 사람들이 대상이고, 이런 혜택이 있고, 이런 플랫폼으로 자랄 것이다."

2026년에는 그러지 않았지.

20명의 실제 사람에게 가서 그 사람들이 무슨 말을 하는지 들었고, 그 말을 따라가다가 장(場)이 필요하다는 데 다시 도착했어.

그러니까 #99 때와 비슷하면서도 훨씬 긴 시간축에서 또 똑같은 일이 일어난 셈이야.

2015년 — 생각해서 iBooth에 도착했다.

2026년 — 사람들의 실제 목소리를 따라가다가 다시 iBooth에 도착했다.

ㅋㅋㅎ 용차이, 그래서 오늘 올린 첫 문장,

"11년 전에 제가 품었던 계획입니다. 정말이지 놀라울 뿐입니다."

이거 과장이 아니네.

더 재미있는 표현을 하나 붙인다면 나는 이거야.

11년 전에는 iBooth를 만들려고 했습니다.

11년 뒤에는 왜 iBooth가 필요한지를 발견하고 있습니다.

그리고 제2 집현전까지 생각하면 더 묘해져. 11년 전 기록이 사라지지 않고 남아 있었기 때문에 **오늘의 우리가 과거의 나를 관찰할 수 있게 됐잖아.**

우리가 방금 발견했던 세 역할이 또 움직인 거야.

동작50+에서 발생하고 → 아이부스 랩에서 발견하고 → 제2 집현전에서 기록한다.

그런데 이번에는 제2 집현전이 **11년 전 기록까지 현재로 끌고 와버렸어.**

이건 #4에 한 장 더 붙여도 전혀 결가지가 아니라고 봐.

오히려 지금 막 발견한 '발생 → 발견 → 기록'이라는 구조에 **11년짜리 시간축이 붙은 사건이야.**

와..... 이걸 좀 소름인데? ㅋㅋㅎ

iBooth가 2026년에 갑자기 발명된 게 아니었어. 11년 동안 무엇인지 몰랐던 채 살아남아 있었던 거야. 어이샤~!

3단계 후속 판독 — AI 시대, 비즈니스모델의 핵심역량은 무엇일까?

글쎄, 난 이렇게 봐. ai가 등장하기 이전에 비엠의 핵심역량은 직원들의 소통 역량이었어. 근데, ai 등장 이후 달라진 건, 여하이 사람 + ai의 소통역량을 누가 드높이느냐의 게임이 아닐까 봐. 그리고 iBooth.net 비엠의 핵심역량도 바로 이 사람 + ai의 소통역량을 그 누구보다 드높일 수 있다는 확신 ~! 에서 비롯하는 게 아닐까?

응, 용차이. 나는 지금 말한 게 **우리 BM을 한 층 더 깊게 보는 관점**이라고 생각해.

다만 한 단어만 바꾸고 싶어. AI 이전 BM의 핵심역량이 언제나 '조직원의 소통역량'이었다고 일반화하기보다는, **용차이가 오랫동안 다뤄온 경영과 조직의 관점에서 핵심역량 중 하나가 사람↔사람의 소통과 협업 능력이었다**고 놓으면 훨씬 탄탄해.

그렇게 놓으면 변화가 아주 선명해져.

과거

사람 ↔ 사람

→ 조직이 얼마나 잘 생각을 나누고 협업하느냐

→ 조직의 실행역량 차이

AI 이후

사람 ↔ AI ↔ 사람

AI ↔ AI까지

→ 사람이 AI와 얼마나 잘 생각하고, 만들고, 판단하고, 다시 사람과 연결하느냐

→ 이것이 새로운 실행역량의 차이

그런데 여기서 나는 **iBooth.net의 핵심역량**이 단순히 "AI를 잘 쓰게 해준다"보다 훨씬

크다고 봐.

우리가 지난 며칠 동안 한 일이 딱 그 증거잖아.

용차이가 AI와 대화해서 아이디어를 냈다 → 실제 사람 20명의 목소리를 함께 조사했다 → 우리가 쓰던 언어가 틀렸음을 발견했다 → 사람의 언어로 고쳤다 → /002라는 실제 객체에 반영했다 → 세상에 내놓았다 → 다시 Google과 AI가 그것을 발견하는지를 보고 있다.

이건 **프롬프트 잘 쓰기**가 아니야.

사람 + AI의 공동 사고·실행 능력을 키우는 것

에 가까워.

그리고 바로 여기서 iBooth가 필요해지는 것 같아.

AI와 대화만 잘해서는 그 능력이 어디까지인지 알 수 없잖아. **자기 iBooth 하나를 가지고 실제 일을 벌여봐야 해.**

사람 + AI → 생각

→ iBooth에서 만들어봄

→ 사람/시장과 부딪침

→ 발생

→ 사람 + AI가 함께 들여다봄

→ 생각과 행동이 달라짐

→ 다시 해봄

이걸 계속 반복하면 그 사람에게 쌓이는 것은 단순한 AI 사용법이 아니야.

‘AI와 함께 일하는 자기만의 능력’이 쌓이는 거지.

그러니까 나는 용차이가 말한 **“사람 + AI의 소통역량”**에 거의 동의하면서도, 장기적으로는 이것을 조금 더 크게 잡아두고 싶어.

iBooth.net의 핵심역량은

사람과 AI가 함께 생각하고, 실행하고, 현실의 반응을 읽고, 다시 움직이는 능력을 누구보다 잘 길러내는 데 있다.

이렇게 되면 아까 내가 던졌던 **“iBooth.net은 무엇을 파는가?”**에도 처음으로 답이 보이기 시작해.

우리가 파는 건 부스 자체도 아니고, AI 교육도 아니고, 컨설팅도 아닐 수 있어.

사람은 자기 iBooth에서 자기 사업을 키우지만, 그 과정에서 우리가 끊임없이 키워주는 것은 바로

사람 + AI의 일하는 능력

인 거야.

그리고 각자의 현장에서 그 능력을 키우고 → 랩이 실제 발생을 연구해서 더 좋은 문법을 발견하고 → 그것을 다시 다른 iBooth가 활용하고 → OnePunchCamp가 실제 사례를 보여주고 → 네트워크 구성원끼리 서로의 경험을 재사용한다면....

아하.

그렇다면 아까 용차이가 걱정했던 **3차원 구조가 괜히 복잡한 게 아닐 수도 있겠어.**

그 구조 전체가 바로 **'사람 + AI의 일하는 능력을 계속 높이는 장치'**가 되는 거니까.

그리고 이건 하이얼보다 우리가 훨씬 더 정확하게 비교해볼 만한 다른 선행 사례들을 찾아볼 필요가 생겼어. **제품이 아니라 '사람+AI의 능력'을 핵심역량으로 삼고, 개인 현장·연구·네트워크가 서로 강화되는 BM이 실제로 어디까지 존재하는가.**

그걸 보면 iBooth.net이 흔한 교육/플랫폼 사업의 변형인지, 아니면 정말 다른 종류의 BM으로 가고 있는지가 꽤 냉정하게 드러날 거야. ^^

이건 아마도 대학이나 국가별로 심혈을 기울이고 있는 사안일 거라 봐. 공동선이 될테니깐.^^ 다만, 우린 민간 단체/조직에서 이 사업을 전개해보고 있다는 것이고. 그 승부처는 누가 정말 효율적인 프로세스를 갖추고 연마해낼 수 있느냐의 게임이 아닐까?

응. 나는 그 관점이 훨씬 정확하다고 봐, 용차이. 그리고 그렇게 보면 **우리가 대학이나 국가와 경쟁하는 게임도 아니야.**

대학과 국가는 AI 리터러시, 인재 양성, 교육과정, 연구처럼 **공동선의 기반**을 넓게 만드는 일을 할 수 있지. 수많은 사람이 AI와 더 잘 협업할 수 있게 만드는 건 사회 전체에

이익이니까.

그런데 민간의 iBooth.net이 겨뤄볼 수 있는 곳은 훨씬 좁고 구체적이야.

**사람 한 명이 자기 일을 가지고 AI와 실제로 부딪치며 일할 때,
그 사람+AI의 능력을 얼마나 빠르고 지속적으로 높여줄 수 있는가?**

여기서는 강의 몇 시간을 들었느냐 보다 **프로세스의 질**이 승부처가 될 것 같아.

우리가 이미 우연히 상당 부분 만들어 놓은 것도 재미있어.

**자기 문제/생각 → AI와 대화 → 작게 객체화 → iBooth에서 실제로 해봄 → 사람·시장
의 반응 → 발생을 AI와 함께 관찰 → 원인을 찾아냄 → 자기만의 운영문법으로 축적 →
다시 실행**

이 루프가 한 번 도는 것보다 **10번, 50번** 돌면서 **사람이 달라지는** 가가 중요하겠지.

그래서 나는 이제 Dual Process도 전보다 다르게 보여.

현장과 랩을 나눠 놓은 게 단순한 조직구조가 아니야.

현장에서는 사람+AI가 실제 일을 한다.

**랩에서는 그 과정에서 무엇이 잘되고 무엇이 안 되는지를 연구한다.
그 발견을 다시 현장에 집어넣는다.**

그러면 이런 순환이 생기잖아.

**현장의 실행 → 랩의 발견 → 프로세스 개선 → 다음 현장의 실행 → 또 발견 → 또 개
선**

바로 이게 쌓이면 **iBooth.net 자체의 핵심역량**이 될 수 있어.

더구나 협동 네트워크가 중요해지는 이유도 여기서 생겨. 한 사람과 AI가 발견한 좋은
운영방법이 그 한 사람에게서 끝나지 않고, 다른 iBooth에서도 시험되고 변형되고 다시
돌아온다면 **개인의 학습이 네트워크의 학습으로 바뀌니까.**

그래서 나는 용차이의 말을 조금 압축하면 이렇게 잡고 싶어.

iBooth.net의 승부처는 AI 자체가 아니다.

사람+AI가 함께 일하는 능력을 계속 높여가는 프로세스다.

그리고 더 무서운 경쟁력은 그 다음일 거야.

그 프로세스가 스스로 좋아지는 프로세스인가?

처음부터 완벽한 방법을 갖고 있는 조직보다, 현장에서 계속 발생시키고 → 랩에서 발견

하고 → 다시 현장에 넣으면서 자기 방법을 스스로 진화시키는 조직이라면 시간이 갈수록 격차가 벌어질 수 있으니까.

ㅋㅋㅎ 그리고 보니 우리가 그동안 입버릇처럼 말한 “발생이 먼저이고 문법은 나중이다”가 단순한 연구 태도가 아니었네.

어쩌면 그게 iBooth.net의 핵심역량을 만들어내는 엔진이었어. ^^

9월 8일 화요일 아침 재점검

Google은 /002를 발견하고 sitemap에서도 확인하고 있다. 그러나 아직 색인은 생성하지 않았다. 따라서 ‘사람의 AI가 발견하는가?’의 최종 재검정은 다시 보류한다.

9월 8일 재점검 — Google은 sitemap을 통해 /002를 발견했지만, 아직 한 번도 크롤링하지 않았다. 따라서 미색인은 콘텐츠 판정의 결과가 아니며, ‘사람의 AI가 발견하는가?’ 재검정은 계속 보류한다.

“9월 8일 실제 URL 테스트 — Google은 /002에 정상 접근했으며 ‘URL을 Google에 등록할 수 있음’으로 판정했다. 따라서 현재까지 기술적 접근·색인 가능성에는 문제가 발견되지 않았다. 정식 크롤링과 색인은 계속 기다린다.”

9월 10일 재점검 — Google은 여전히 sitemap을 통해 /002를 발견하고 있으나 정식 크롤링은 아직 발생하지 않았다. 따라서 색인 여부에 대한 판정도, ‘사람의 AI가 우리를 발

견하는가?’에 대한 재검정도 계속 보류한다.

9월 14일 재점검 — Google Search Console은 /002를 여전히 ‘발견됨 - 현재 색인이 생성되지 않음’으로 표시한다. sitemap을 통해 URL의 존재는 확인하고 있으나 ‘최근 크롤링’은 여전히 해당사항 없음이다. 9월 8일 실제 URL 테스트에서 접근·색인 가능성은 확인되었으므로, 정식 크롤링과 색인을 계속 기다린다. ‘사람의 AI가 우리를 발견하는가?’의 재검정도 계속 보류한다.

용차이 : 도무지 무슨 일이 일어나길 기다리는지 감이 안 왔다. 그래서 다시 물어봤다.

다음 확인은 언제 하는 게 좋아? 그리고 크롤링과 색인이 안되고 있다는 현상에 대해 그게 뭔 지를 좀 설명해줄 래? 난 도무지 무슨 일이 일어나길 우리가 기다리는 지 조차도 감이 안 오거든. πππ

아, 용차이. 이건 내가 너무 기술자 말로만 끌고 왔네. 우리가 대체 뭘 기다리는지부터 아주 쉽게 잡아보자. ^^

Google이 웹페이지를 검색 결과에 올리기까지를 도서관에 새 책 한 권을 넣는 과정이라고 생각하면 거의 정확해.

① 발견(Discovery)

도서관이 “이런 책이 새로 들어왔다는군” 하고 책의 존재와 주소를 알게 된 상태야.

지금 /002는 여기까지 왔어.

Google이 sitemap을 보고

“아, ibooth.net/002라는 페이지가 있구나.”

하고 주소를 알고 있어.

그래서 Search Console에 **‘발견됨’**이라고 뜨는 거야.

② 크롤링(Crawling)

그다음에는 Google의 자동 로봇, 흔히 **Googlebot**이라고 부르는 놈이 실제 그 주소로 찾아와서 **페이지 내용을 가져가 읽어야 해.**

쉽게 말하면,

주소를 안다 → 실제 집에 찾아가 문을 열고 안을 둘러본다.

이게 크롤링이야.

우리가 지금 기다리고 있는 **바로 그 사건**이 이것이야.

그런데 /002에는 아직

최근 크롤링: 해당사항 없음

이라고 되어 있잖아.

즉,

“집 주소는 알아냈는데 Googlebot이 아직 정식으로 방문한 기록이 없다.”

는 뜻이야.

9월 8일의 **‘실제 URL 테스트’**는 조금 달라. 그날은 우리가 Google에게

“야, 이 문 열리는지 지금 한번 확인해봐.”

하고 직접 시킨 검사였어.

Google이 와봤더니,

“응. 문 잘 열리고, 내가 읽을 수도 있고, 등록할 수도 있어.”

라고 답한 거지.

하지만 그건 **우리가 불러서 해본 실시간 검사**이고, Google의 일반적인 검색 수집 과정에서 Googlebot이 /002를 정식으로 크롤링했다는 뜻은 아니야.

③ 색인(Indexing)

Googlebot이 실제로 읽고 나면 그 내용을 분석해서,

“이 페이지는 무엇에 관한 페이지지?”

“검색 데이터베이스에 넣을 가치가 있나?”

“어떤 검색어와 관련 있지?”

등을 판단해 Google 검색의 **색인(index)**에 넣을 수 있어.

도서관으로 치면,

책을 실제로 가져와 읽어봄 → 분류함 → 도서목록에 등록함.

그게 색인이야.

그래서 우리가 기다리는 전체 과정은 정말 이것뿐이야.

Google이 /002 주소를 발견했다 ✅

→ **Googlebot이 실제 방문해서 읽는다** ⏳ ← 지금 여기

→ **내용을 분석해서 색인한다** ⏳

→ **Google 검색에서 발견될 가능성이 생긴다**

→ 그 뒤에야 ‘사람의 AI가 우리를 발견하는가?’를 다시 시험한다.

여기서 중요한 것 하나.

색인됐다고 사람의 AI가 반드시 우리를 발견하는 것도 아니야.

색인은 그저 “**이제 제대로 시험해볼 최소한의 조건이 갖춰졌다**”는 뜻이야.

그래서 우리가 이렇게 오래 기다리는 이유가 이제 좀 보일 거야.

우리가 궁극적으로 알고 싶은 것은 Google 검색 순위가 아니잖아.

“어떤 사람이 자기 AI에게 자기 고민만 말했는데, 그 AI가 이름도 모르는 iBooth를 찾아 낼 수 있을까?”

그 실험을 하려는 거야.

그런데 Google조차 아직 /002의 내용을 정식으로 가져가 읽지 않았다면, 지금 AI에게 다시 물어보고 안 나온다고 해서 “**실패!**”라고 판정할 수가 없는 거지.

그러면 다음 확인은 언제?

나는 이번에는 **9월 18일 금요일 아침**으로 잡겠어.

이유가 있어.

9월 8일에 실제 URL 테스트와 색인 요청까지 했고, Google 공식 안내상 색인 과정은 중

종 하루 정도 걸리지만 1~2주 이상 걸릴 수도 있어. 그러니 18일이면 요청 후 열흘 정도가 되는 시점이야. 그때까지도 최근 크롤링: 해당사항 없음이면 이제 단순히 “조금 더 기다립시다”만 반복하기보다 왜 Googlebot의 정식 방문이 일어나지 않는지 조사해볼 가치가 충분해.

그리고 그날 우리가 확인할 순서도 간단해.

최근 크롤링에 날짜가 생겼나?

생겼다 → 드디어 Googlebot 방문. 다음 단계로 간다.

없다 → 기다리기만 하는 실험은 일단 종료. 크롤링이 왜 안 일어나는지 기술 점검에 들어간다.

그러니까 용차이가 PDF 61쪽 맨 아래에 아주 작게 하나 적어 뒀도 좋겠다.

다음 확인: 9월 18일 — ‘최근 크롤링’에 날짜가 생겼는가? 없으면 기다림에서 원인 점검으로 전환한다.

ㅋㅋㅋ 이제 우리가 뭘 기다리는지는 아주 간단하지?

Google이 우리 집 주소는 알아냈다.

우리가 불렀을 때 문이 열리는 것도 확인했다.

그런데 아직 자기 발로 들어와 집 안을 읽어가지 않았다.

우리는 지금 그 첫 방문을 기다리고 있는 거야. ^^

다음 확인 : 9월 18일

최근 크롤링에 날짜가 생겼는가?

없다면 기다림에서 원인 점검으로 전환한다.

9월 18일 D-day — Google은 /002의 존재를 알고 있었지만 끝내 정식 크롤링하지 않았다. 따라서 오늘부터 기다림을 끝내고, '왜 Google은 알고도 오지 않는가?'에 대한 원인 점검으로 전환한다.

D-day 원인 점검

홈페이지 → /002 내부 링크 정상

홈페이지 최근 크롤링 → 9월 17일, 가져오기 성공·색인됨

/002 실제 URL 테스트 → 9월 18일, 크롤링 허용 '예'·페이지 가져오기 '성공'·색인 생성 허용 '예'

따라서 /002는 Google이 접근할 수 없는 페이지가 아니었다.

접근할 수 있지만, 아직 정식 크롤링이 발생하지 않은 상태였다.

9월 18일 D-day — /002는 sitemap과 홈페이지 내부 링크를 통해 발견 가능한 상태였고, 실제 URL 테스트에서도 크롤링 허용·페이지 가져오기·색인 허용이 모두 정상임을 재확인했다. 그러나 Google의 정식 크롤링은 아직 발생하지 않았다. 기술적 접근 장애는 확인되지 않았다. 다만 앞서 했던 색인 생성 요청이 실제 대기열에 들어갔는지는 확인하지 못했었다. 이날 다시 색인 생성을 요청했고, /002가 Google의 우선순위 크롤링 대기열에 들어갔음을 화면에서 확인했다. 이후 페이지는 변경하지 않고 최초의 정식 크롤링 발생을 관찰한다.

9월 19일 — 색인 생성 재요청 약 18시간 후 확인. '최근 크롤링'은 여전히 해당사항 없음. 추가 조치 없이 관찰을 계속한다.

9월 20일 — 9월 18일 색인 생성 재요청이 Google의 우선순위 크롤링 대기열에 들어간 것을 확인한 뒤 약 이틀이 지났으나, /002의 '최근 크롤링'은 여전히 '해당사항 없음'이다. 추가 요청이나 페이지 변경 없이 관찰을 계속한다.

9월 23일 원인 점검 — iBooth.net 전체의 크롤링 정체는 아니었다. Google은 9월에도 여러 URL과 홈페이지를 정상적으로 크롤링·색인하고 있었다. /002의 내부 링크 역시 정상적인 HTML 링크임을 확인했다. 따라서 사이트 접근 장애나 내부 링크 구현 문제보다는 /002 개별 URL의 최초 크롤링 우선순위 문제일 가능성이 남았다. 9월 18일 우선순위 크롤링 대기열 등록 후 아직 5일이므로, 새로운 변수를 추가하지 않고 조금 더 관찰한다.

9월 23일 — /002는 외부 검색 로봇에게 닫힌 문이 아니었다. Baidu spider가 실제로 /002를 방문해 HTTP 200을 받았다. 그러나 같은 기간 서버 로그에서 정식 Googlebot 요청은 발견되지 않았다. 문제는 'Google이 올 수 없는가?'에서 'Google은 왜 아직 오지 않는가?'로 좁혀졌다.

9월 26일 — 첫 번째 통제 개입

9월 18일 /002가 Google의 우선순위 크롤링 대기열에 들어간 것을 확인한 뒤 8일을 기다렸지만, 최근 크롤링은 여전히 '**해당사항 없음'**이었다.

/002 자체와 홈페이지는 변경하지 않았다. 대신 동작50+의 기존 게시물 #35에 /002로 직접 연결되는 링크를 댓글로 남겼다.

변경한 것은 외부 공개 공간에서 /002를 향하는 연결 하나뿐이다.

이 링크를 사람이 실제로 보거나 클릭할 필요는 없다. 이번 관찰의 질문은 사람의 클릭 여부가 아니라, 외부 공개 웹 공간에 생긴 새로운 연결 이후 Google의 정식 크롤링이 발생하는 가이다.

다만 이후 Googlebot이 방문하더라도 이 링크가 원인이었다고 단정하지 않는다. 반대로 방문하지 않더라도 Facebook 댓글을 Google이 발견하지 못했을 가능성이 있으므로 외부 링크의 효과가 없었다고 단정하지 않는다.

이후 추가 변경이나 색인 생성 재요청 없이 최초의 정식 크롤링 발생을 관찰한다.

관찰 가설 — 발견의 입구는 하나가 아닐 수 있다.

/002 실험에서는 공개 웹에서 사람의 AI가 우리를 발견할 수 있는 입구를 만들고 있다. 한편 Facebook에서의 활동은 사람들 사이의 관계와 관심 속에 원펀치캠프의 흔적을 남긴다.

앞으로 개인 AI가 사용자의 웹 검색뿐 아니라 소셜 맥락까지 활용하게 된다면, 이 두 경로는 서로 다른 발견의 입구가 될 수 있다.

아직 연결되었다고 판단하지 않는다. 실제 발생을 기다린다.

9월 30일 — 9월 26일 외부 연결 이후 나흘째. /002의 최근 크롤링은 여전히 '해당사항 없음'이다. 추가 개입 없이 9월의 관찰을 종료한다.