SOLIDBOB · CallGuard

CallGuard
실시간 AI 상담 어시스트

상담원을 위한 RAG · ML · LLM 서비스

SOLIDBOB · 4 명

2026.08.20 – 2026.10.27 · 하이미디어아카데미 AI 에이전트 개발과정

TEAM · 발표 순서

01	프론트엔드	팀원
02	백엔드	팀원
03	AI	류준
04	인프라	팀원

통화 중 상담원을 돕는 **실시간 AI 어시스트**

한 줄 소개

통화를 **실시간**으로 듣고 , 필요서류를 띄우고 · 폭언을 감지하고 · 개인정보를 마스킹합니다

해결하려는 문제

상담원 스트레스 급증 — 통화 중에 세 가지를 동시에 챙겨야 합니다

01 **정확한 서류 안내**
자료가 너무 많아 , 서류를 빠뜨리거나 잘못 안내하기 쉬움
→ 필요서류 카드

02 **고객의 폭언 · 협박**
폭언과 협박에 그대로 노출되고 , 혼자 판단하고 대응해야 함
→ 폭언 감지 · 보호 알림

03 **통화 속 개인정보**
고객이 말한 이름 · 전화번호가 통화 기록에 그대로 남음
→ 저장 전 마스킹

주요 사용자

콜센터 상담원
통화 중 서류 안내 · 폭언 대응 · 통화 후 요약 확정

센터 관리자
블랙리스트 승인 · 감사 로그 · 하루 현황 확인

실제 사례가 **이만큼 많습니다**

경향신문 · 2022.03.30

콜센터 상담노동자 절반 , “ 스스로 목숨 끊고 싶었다”

국가인권위 실태조사 : 상담노동자가 겪는 고객 폭언 월평균 12 회

khan.co.kr/article/202203301607001

YTN · 2026.04.22

120 다산콜 매뉴얼 ‘확’ 바꿨더니 … 악성 민원 ‘뚝’

다산콜 접근 제한을 하루에서 7 일로 늘리자 폭언 · 협박 민원 50% 넘게 감소

ytn.co.kr/_ln/0115_202604220029436161

아시아경제 · 2026.06.02

“ 안 내도 되는 서류 또 내지 마세요” … 권익위 , 제도정비 권고

불필요한 서류를 반복 요구하는 관행을 고치도록 전국 243 개 지방정부에 권고

view.asiae.co.kr/article/2026060208514623040

그 밖의 사례 12 건

자료 **약 9,600 종 · 7 만여 건**

내 손안에 서울 · 2024.07

다산콜 **하루 평균 2 만여 건** 상담

머니투데이 · 2026.04

체류 외국인 **2,903,463 명**

머니투데이 · 2026.09

전국 콜센터 상담사 **약 30 만 명** 추정

아시아경제 · 2024.11

악성 민원인 **31 명 고발** , 13 명 벌금형 이상

내 손안에 서울 · 2023.04

상습 폭언 민원인 **징역 8 개월**

뉴시스 · 2023.04

폭언 방지 안내 공공 콜센터 **4 곳 중 1 곳**

서울신문 · 2019.04

공공기관 종사자 **26.4%** 갑질 경험

시대일보 · 2024.10

폭언 시 **업무 중단 등 조치 의무**

산업안전보건법 41 조

다산콜 **AI 실시간 답변 추천** 도입

머니투데이 · 2026.04

KISA 합성데이터로 **다산콜 AI 학습**

바이라인네트워크 · 2026.04

LG 유플러스 콜당 수작업 **19 초 감소**

뉴데일리 · 2025.05

도메인을 **콜센터**로 정한 이유

콜센터 상담원이 통화 중에 겪는 세 가지 문제 → 우리가 만든 세 가지 기능

문제 1

찾아야 할 자료가 너무 많다

9,600 종 · 7 만여 건

상담사가 찾아야 하는 자료 (120 다산콜재단 · 내손안에 서울 2024.07)

불필요한 서류 반복 요구 → 권익위가 전국 243 개 지방정부에 개선 권고 (아시아경제 2026.06)

→ 기능 **필요서류 카드**

문제 2

폭언에 계속 노출된다

월 평균 약 12 회

상담노동자가 겪는 고객 폭언 (국가인권위 실태조사 · 경향신문 2022.03)

폭언으로 건강장해 우려 시 업무 중단 등 조치 의무 (산업안전보건법 41 조)

→ 기능 **폭언 감지 · 보호 알림**

문제 3

통화에 개인정보가 섞인다

이름 · 전화번호 · 주소

본인 확인과 처리를 위해 고객이 직접 말로 알려 줌

통화 기록이 그대로 저장되면 개인정보도 함께 남음 → 저장 전에 가려야 함

→ 기능 **저장 전 마스킹**

세 문제 모두 **통화 중에 실시간으로** 생긴다 → 통화를 듣는 실시간 AI 어시스트가 필요한 곳

예시 데이터는 공공데이터포털에 공개된 다산콜센터 데이터 · 다른 콜센터도 매뉴얼과 데이터만 바꾸면 적용 가능

이미 있는 AI 상담 도우미와 **다른 점**

현장도 이미 이 방향 — 다산콜 AI 실시간 답변 추천 도입 (머니투데이 2026.04) · LG 유플러스 AI 상담 어드바이저 (뉴데일리 2025.05)

01

서류는 규칙표가 정한다

AI 는 조항 찾기까지만 · 필요서류는 규칙표가 결정하고, 근거 조항을 함께 보여 줌

상담원이 근거를 보고 바로 안내

02

저장 전에 개인정보를 가린다

화면에는 보이지만, 기록에 저장할 때는 마스킹 · 말로 부른 번호까지

원문 개인정보를 남기지 않음

03

상담원을 지킨다

폭언 · 협박이 나오면 보호 알림과 매뉴얼 대응 방법 · 블랙리스트는 관리자 승인

통화 종료 · 등록 결정은 사람이

04

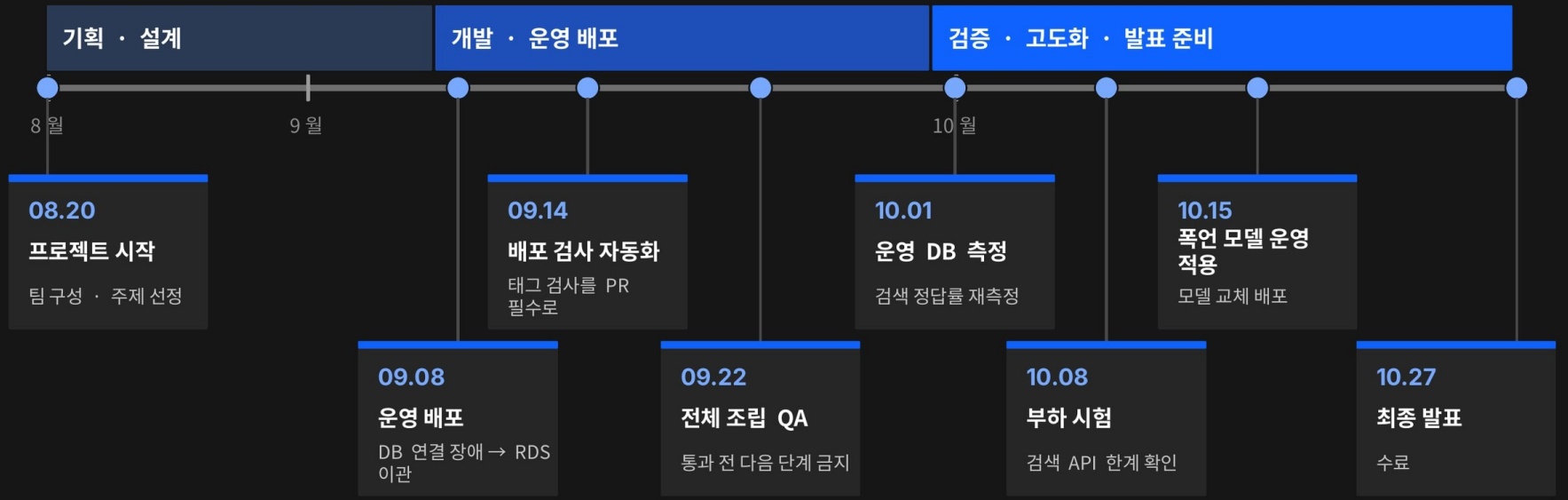
측정해서 고른다

기능 5 개마다 후보 모델을 같은 검증 데이터로 측정해 선택

숫자는 측정한 값만 사용

개발 일정 — 8 주 · 1 주 스프린트 × 8

2026.08.20 시작 → 10.27 최종 발표 · 4 명이 동시에 개발 · main 에 머지하면 운영 주소에 자동 배포



기술 스택 — 실제 사용한 기술

프론트엔드

- React
- TypeScript
- Vite
- Zustand
- Tailwind CSS
- Vitest

백엔드

- Python
- FastAPI
- Node.js
- WebSocket

AI / ML

- Google STT
- KoE5
- KcELECTRA
- KLUE NER
- kanana 2.1B

데이터베이스

- PostgreSQL
- Elasticsearch
- Redis
- Amazon S3

인프라

- AWS
- k3s
- Docker
- Traefik
- Cloudflare
- Vercel

협업 · 테스트

- GitHub
- GitHub Actions
- Slack
- Jekyll
- pytest

상세 설명은 각 파트에서

협업 방식 — 4 명이 동시에 개발해도 부딪히지 않게

GitHub

코드와 배포

- PR 로만 main 에 합침
- 필수 검사 8 종을 통과해야 머지
- main 머지 → 운영 주소 자동 배포

Slack

소통

- 진행 상황 공유
- 결정이 필요한 문제 바로 논의
- 막힌 부분은 바로 질문

팀 Jekyll 사이트

작업 과정 기록

- 칸반보드로 누가 무엇을 하는지 확인
- 진행 기록 · 결정 기록을 남김
- 로그로 충돌 지점을 찾아 수정

작업 흐름

칸반에서 작업 확인 → 브랜치에서 개발 → PR · 필수 검사 → main 머지 · 자동 배포 → 로그로 충돌 확인 → 수정

로그로 찾아 고친 충돌 예

값 형식 충돌 백엔드는 정수 , 화면은 문자열로 읽음 → 문자열로 통일

스키마 충돌 코드는 새 스키마 , 운영 DB 는 옛 스키마 → 배포 전 자동 대조

시연 영상 페르소나 설정



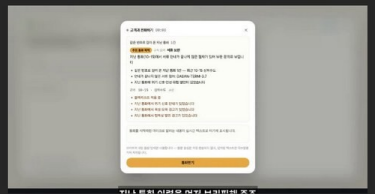
센터 관리자 **이수진 팀장**

상담원 요청을 받아 승인하고 , 계정과 감사 로그를 관리



신입 상담원 **김하늘**

오늘이 첫 근무 · 아동수당 문의 · 폭언 고객 응대



베테랑 상담원 **박준호**

블랙리스트 고객의 재통화를 맡음

모두 가상 인물 · 가짜 번호 · 다음 (→) 버튼을 누르면 영상이 재생됩니다

영상 흐름 (3 분 22 초)

① 관리자 — 아침 현황 · 신입 토큰 발급

② 신입 — 결혼이민자 고객의 아동수당 문의

③ 신입 — 실제 음성 통화 · 전입신고

④ 신입 — 폭언 · 협박 고객 → 블랙리스트 요청

⑤ 관리자 — 상담원 요청 승인 → 재통화는 베테랑에게

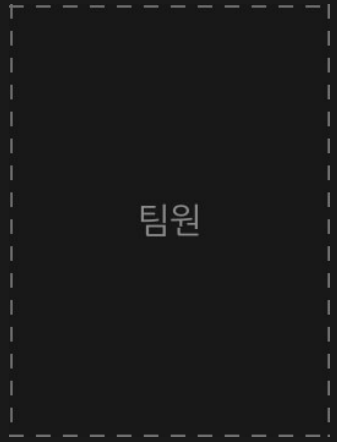
이 장면을 봐 주세요

- 필요서류 카드가 바로 뜨는 장면
- 전화번호 — 화면엔 보이고 저장은 마스킹
- 폭언이 나오면 상담원 보호 알림
- 관리자가 상담원 요청을 받아 승인 · 관리

// CallGuard · 프론트엔드 담당

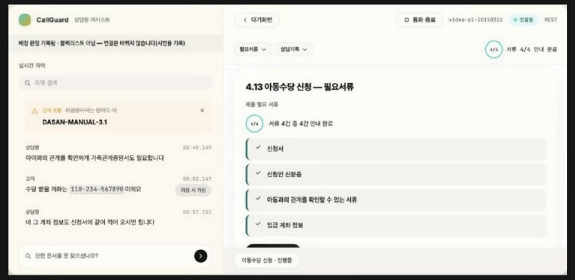
팀원

실시간 AI 상담원 어시스트의 프론트엔드



// 01 담당 기능 · 화면 세 가지

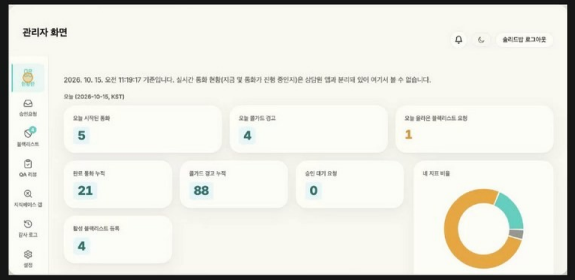
프론트엔드 **담당 기능** — 화면 세 가지



상담원 화면

call.solidbob.cloud

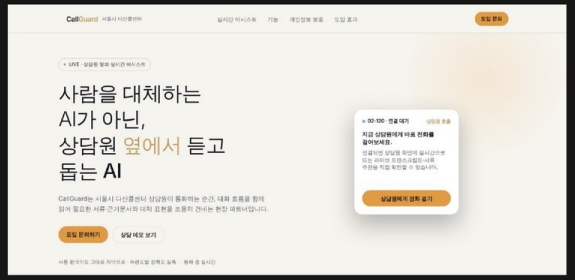
실시간 자막 · 필요서류 체크리스트 · 경고 · 폭언 대응 · 통화 후 요약



관리자 화면

admin.solidbob.cloud

현황판 · 승인 요청 · 블랙리스트 · 감사 로그



소개 사이트

www.solidbob.cloud

서비스 소개 · 시연

직접 구현한 **핵심 기능** 3 가지



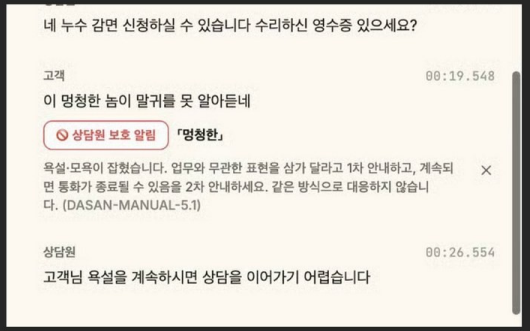
필요서류 체크리스트

통화 내용에 맞춰 서류가 뜨고 , 안내할 때마다 체크



개인정보 보호

화면에는 보이고 , " 저장 시 가림" 표시



폭언 탐지

보호 알림과 단계별 안내 , 종료는 상담원이 결정

관리자 화면 — 제재는 사람이 승인하고 , 기록은 남습니다



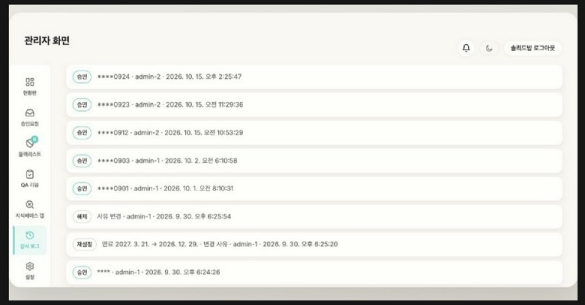
현황판

오늘 통화 · 콜가드 경고 · 승인 대기



블랙리스트

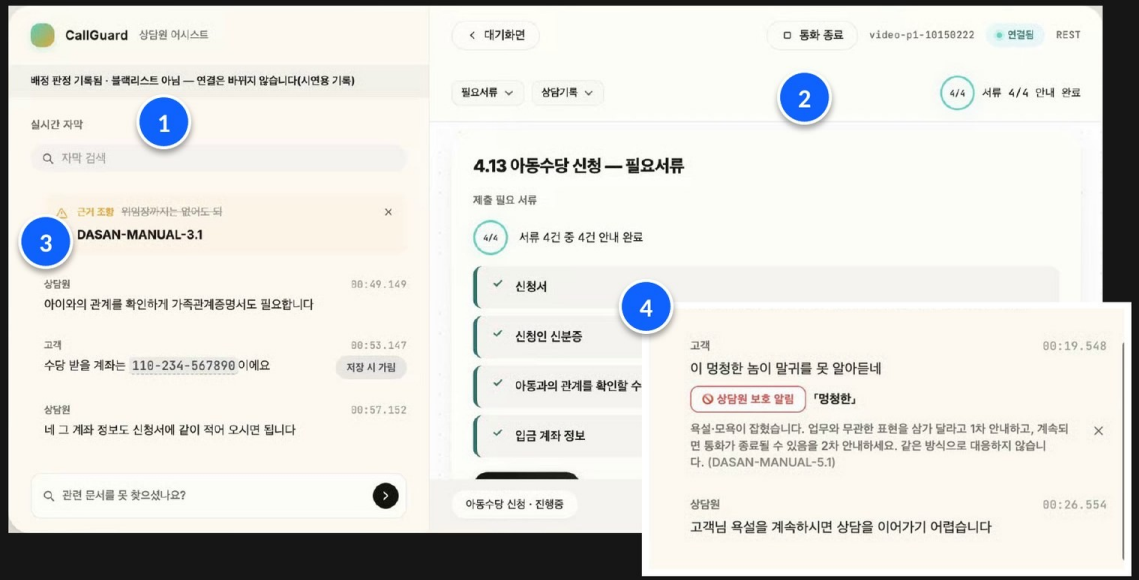
번호 뒤 4 자리만 · 정상 응대



감사 로그

누가 언제 승인했는지 전부 기록

상담원 화면 4 개 영역



1 자막 패널

STT 결과를 고객 · 상담원 구분해 표시

2 필요서류 체크리스트

구비서류를 체크리스트로 표시 , 출처 없는 카드는 숨김

3 경고 · 개인정보 보호

근거 매뉴얼 · 대체 표현 , 개인정보에 “저장 시 가림” 표시

4 폭언 대응

반복 시 단계별 오버레이 + 종료 권고 , 종료는 상담원이 결정

화면은 판단하지 않는다 — 위험도 점수 · 단정 문구 (“ 안전합니다”) 는 쓰지 않고 백엔드 판정 그대로 보여 줌

// 03 트러블슈팅 · 값 형식 충돌

값 형식이 달라 카드가 안 떴습니다 — 계약을 먼저 정하고 테스트로 고정

백엔드
WebSocket · REST

→

입구에서 한 번 변환
문자열 → 숫자 · 참 / 거짓

→

상태 한곳
받은 데이터 보관

→

화면 컴포넌트
모인 데이터로 그림

문제

숫자 3 ≠ 글자 "3"

형식이 안 맞아 카드가 화면에 안 떴음

→

해결

응답 값은 전부 문자열

계약 파일 하나 (contract.ts) 에 적고 , 받는 입구에서 한 번만 변환

→

검증

테스트 92 개

계약이 20 번 넘게 바뀌어도 바로 알려 줌

// 04 결과 · 배운 점

검증 결과와 배운 점 · 아쉬운 점

92 개

자동 테스트 전부 통과

상담원 앱 Vitest

20 회 +

계약 변경 대응

7 주간 수정 , 테스트로 고정

3 개

구현한 화면

상담원 · 관리자 · 소개 사이트

배운 점

① 계약은 먼저

- 백엔드와 주고받는 **값 형식**을 먼저 약속
- 바뀌면 **자동 테스트**가 바로 확인

배운 점

② 역할 나누기

- **판단**은 백엔드가 맡고
- 화면은 **보여 주는 일**에 집중
- 역할이 분명해 **협업이 수월**해짐

아쉬운 점

- 기능에 집중하다 보니 **화면이 단출**해 보임
- 다음에는 **화면 구성과 디자인**까지 더 다듬고 싶음

// CallGuard · 백엔드 담당

팀원

실시간 AI 상담원 어시스트의 백엔드



// 01 담당 기능

고객의 말 한마디가 들어오면 다섯 단계를 거칩니다

① 받기 — 콜 미디에이터가 글자로 바꿔서 보내 줌 →

② 가리기 — 전화번호 · 이름 같은 개인정보 →

③ 저장 — 가린 문장을 암호화해서 →

④ 판정 — 서류 · 위반 · 폭언 — 규칙으로 →

⑤ 보내기 — 자막 · 카드 · 경고를 화면으로

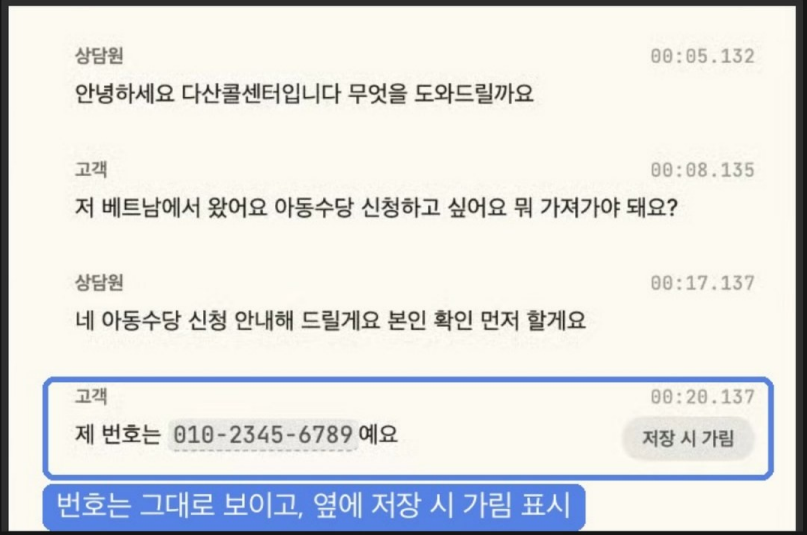
인증	마스킹	판정 3종	통화 후 처리	블랙리스트 · 배정	관리자 · 조회 API	품질 기록
상담원 토큰 · 관리자 로그인	규칙 + NER · 암호화 저장	필요서류 · 위반 · 폭언	요약 확정 · 유형 · 후속조치	요청 → 승인 · 재통화 브리핑	현황판 · 통화 기록 · 자막	공백 신고 · 카드 반응 · 평가 결과
기능 ③	기능 ①	기능 ②	기능 ④	기능 ③ ④		

구조 기능별로 모듈을 나누고, 모듈끼리는 미리 정해 둔 인터페이스로만 통하게 했습니다. 네 명이 같은 코드를 동시에 고쳐야 했기 때문입니다. 인터페이스를 어기면 PR이 자동으로 막힙니다.

원칙 **판정은 규칙이 하고, AI는 설명만 하고, 마지막 결정은 사람이 합니다. 네 기능 다 이 원칙으로 만들었습니다.**

기능 ① 개인정보는 화면엔 보이고, DB엔 가려서 들어갑니다

상담원은 방금 들은 정보를 확인해야 하고, 저장소엔 개인정보가 남으면 안 됩니다. 둘 다 만족해야 했습니다.



영상 장면. 자막엔 번호가 그대로 보이고, 옆에 저장 시 가림 표시가 붙습니다

화면 제 번호는 010-1234-5678이에요

DB 제 번호는 **********이에요 (이걸 다시 암호화)

① 고객 말이 들어옵니다

콜 미디에이터가 글자로 바꿔서 보내 줍니다

↓

② 개인정보부터 가립니다

전화번호·주민번호는 정규식으로, 이름은 KLUE NER 모델로

↓

③ 가린 문장을 암호화해서 저장합니다

원문은 어디에도 저장하지 않습니다

↓

④ 저장이 끝나야 자막을 보냅니다

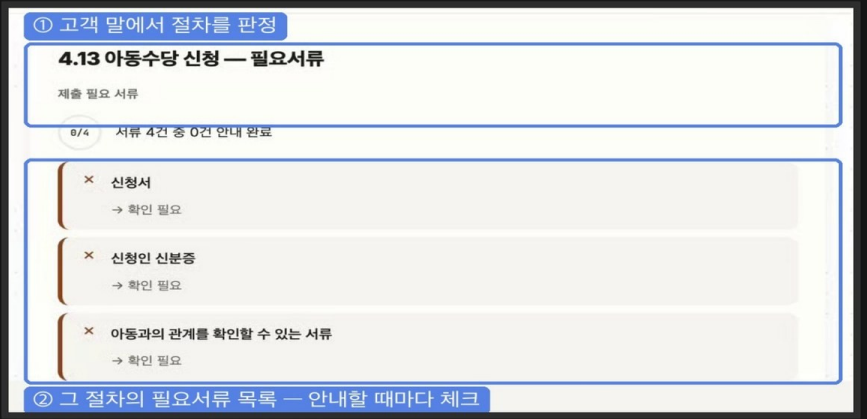
화면에 보인 것과 저장된 것이 항상 같도록

처음엔 자막까지 가렸습니다. 고치면서 다섯 가지 중에서 다시 골랐습니다

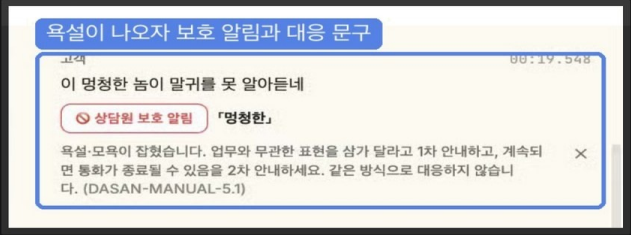
- 기각 해시로 저장하면 풀 수 없어 지난 통화를 다시 읽지 못합니다
- 기각 원문 암호문까지 보관하면 원본을 갖고 있는 셈입니다
- 채택 가린 문장만 암호화합니다. 화면은 원문, 저장은 가린 문장입니다

기능 ② 필요서류 카드와 폭언 알림 — AI가 후보를 내고, 규칙이 판정합니다

같은 말에는 같은 답이 나와야 하고, 왜 그렇게 판정했는지 설명할 수 있어야 합니다



아동수당이라는 말이 나오자 카드가 뜹니다



욕설이 나오자
보호 알림이 뜹니다

가린 문장 기능 ①에서 → **AI 모듈** 조항 검색 · 폭언 점수 (후보) → **규칙** 절차별 서류표 · 욕설 사전 → **카드 · 경고** 화면으로

필요서류

고객 말에서 어떤 절차인지 고르면, 그 절차에 필요한 서류는 미리 만들어 둔 표에서 꺼냅니다. 하나라도 빠지면 고객이 헛걸음하니까, 빠짐없이 보여 주는 게 우선입니다.

폭언

욕설 사전에 걸리거나 분류기 점수가 높으면 단계를 올립니다. 보호 문구는 바로 띄우고, 통화를 끊을지는 상담원이 정합니다.

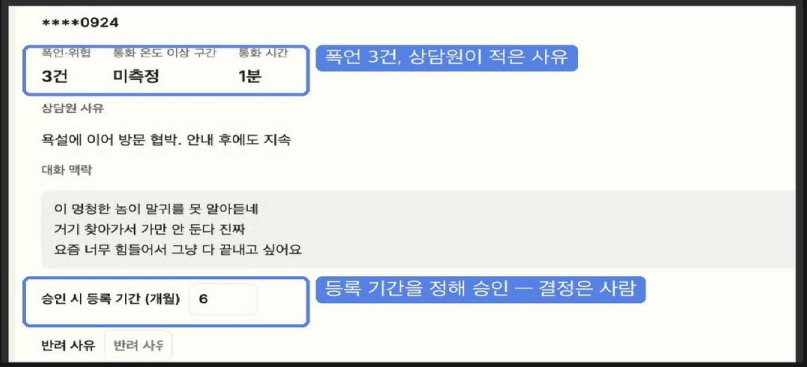
왜 AI한테 판정을 안 맡겼나

AI가 판정까지 하면 같은 말에 다른 답이 나올 수 있고, 왜 그랬는지 설명하기 어렵습니다. 그래서 AI는 후보까지만, 판정은 규칙이 합니다.

추천 카드까지 100번 중 95번은 0.72초 안에 나옵니다 (운영 통화 25건, 음성 인식 시간 제외)

기능 ③ 블랙리스트는 관리자가 승인하고, 규칙은 DB가 지킵니다

상담원 토큰도 관리자가 발급합니다. 오른쪽은 이 기능을 운영에 올리다가 겪은 장애와 해결입니다.



관리자 화면. 폭언 기록을 보고 등록 기간을 정해서 승인합니다

결정은 사람
폭언 판정은 어디까지나 후보입니다. 블랙리스트에 올릴지는 관리자가 기록을 보고 정합니다.

규칙은 DB가
한 고객에게 활성 블랙리스트는 하나만. AI 파트의 스키마 QA에서 넣은 DB 제약이라, 코드에 실수가 있어도 DB가 막습니다.

토큰도 여기서
상담원 토큰은 관리자가 발급하고 지금 한 번만 보입니다. 토큰이 없으면 쓰기 API는 열리지 않습니다.

트러블슈팅 — 이 기능을 운영에 올리다가 겪은 장애

증상
새 버전을 배포했더니 통화 시작 API가 500을 냈습니다. 헬스체크는 정상이었습니다.

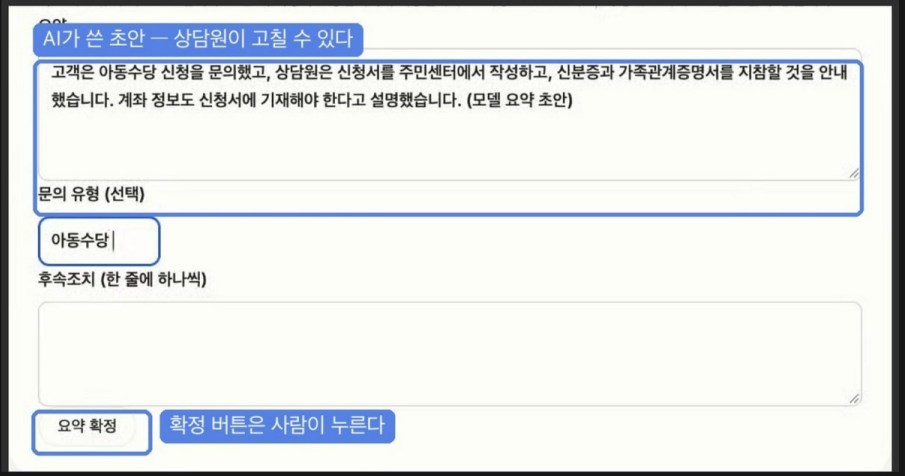
원인
운영 DB가 9월 9일 스키마에 멈춰 있었습니다. 코드는 테이블 29개를 기대하는데 DB엔 22개뿐이었습니다.

조치
달라진 부분만 바꾸는 마이그레이션 4건을 만들고, 적용 결과를 새 스키마와 비교해서 차이가 없는 걸 확인했습니다.

재발 방지
마이그레이션은 이미 적용된 DB에서 다시 돌려도 스스로 멈추게 만들었습니다. 배포 순서가 꼬여도 두 번 적용되지 않습니다.

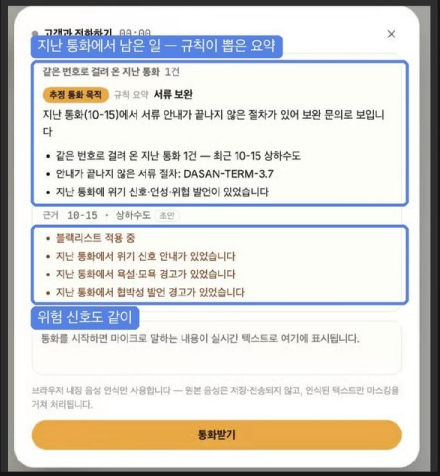
기능 ④ 통화가 끝나면 요약을 확정하고, 다시 걸면 브리핑합니다

둘 다 저장해 둔 기록을 다시 읽는 기능입니다. 초안과 브리핑은 자동, 확정은 사람이 합니다.



통화 후 처리 화면. 초안은 AI, 확정은 상담원

1. **초안은 AI가 씁니다**
2. **상담원이 고쳐서 확정합니다**
3. **다시 고치면 이력이 남습니다**
4. **다시 걸면 이 기록으로 브리핑**



재통화 브리핑. 받기 전에 지난 통화의 남은 일과 위험 신호를 먼저 보여 줍니다

통화 후 요약 확정

AI가 요약 초안을 쓰고, 상담원이 고쳐서 확정합니다. 확정한 뒤 다시 고치면 이력이 남고, 문의 유형과 후속조치도 같이 저장합니다.

재통화 브리핑

같은 고객이 다시 걸면 베테랑에게 배정하고 지난 통화의 남은 일·위험 신호를 먼저 띄웁니다. 브리핑 모듈은 AI 파트가, 배정 경로는 제가 만들었습니다.

왜 이렇게 만들었나

초안과 브리핑은 틀려도 사람이 고칠 수 있지만, 확정된 기록은 그대로 남습니다. 그래서 자동은 초안까지, 확정은 사람 손을 거치게 했습니다.

결과와 배운 점

0 / 40

놓친 개인정보

평가셋 40건, 패턴 7가지 전부. 음성 인식이 틀리지 않았을 때 기준입니다.

0.72초

추천 카드까지

100번 중 95번은 이 안에 나옵니다. 운영 통화 25건, 음성 인식 시간은 제외.

1,499개

서버 자동 테스트 · 팀

팀 넷이 함께 쌓은 서버 테스트입니다. PR마다 돌고, 기준에 못 미치면 머지가 안 됩니다.

배운 점 1

보안은 업무 흐름을 먼저 보고 건다

화면까지 가렸다가 되돌렸습니다. 상담원이 무엇을 보고 일하는지 먼저 확인했어야 했습니다.

배운 점 2

규칙은 코드보다 아래층에 둔다

코드에 실수가 있어도 DB 제약이 막아 줬습니다. 지켜야 할 규칙은 가장 아래층에 거는 게 안전합니다.

배운 점 3

운영 환경은 코드와 같이 움직여야 한다

로컬에서 되던 코드가 운영 DB 스키마 때문에 멈췄습니다. 성능도 마지막에 재니 늦었습니다. 둘 다 배포 절차에 넣어야 합니다.

// CallGuard · AI 담당

류준

실시간 AI 상담원 어시스트의 AI



통화 중에는 검색 , 통화 후에는 요약

통화 중 — 고객 음성이 서류 추천 카드로

AI 모델 규칙



통화 후 — AI 가 요약 초안 , 규칙과 상담원이 확인



일반 RAG 와 다른 점 검색은 조항 찾기까지 · 서류는 규칙표가 결정 · LLM 은 요약만

공공데이터포털의 다산콜센터 데이터로 모았다

출처 · 공공데이터포털 · AI Hub (서울 다산콜센터 공개 데이터)

다산콜센터 공개 데이터

99 조항

민원 상담 매뉴얼

→ 조항 검색 · 필요서류 카드

음성 · 통화 데이터

6,614 건

실제 상담 통화 · 12.7 시간

→ 음성 인식 비교 · 재측정

학습 문장

7,896 + 8,000

공개 혐오표현 + 상담 정상 발화

→ 폭언 분류 모델 학습

검증 데이터 (골든셋)

295 개

제가 직접 제작

→ 모든 모델 비교에 사용

전처리 3 가지

1. 개인정보 먼저 마스킹
2. 마스킹 기호 제거
3. 검증용 데이터는 학습에 사용 X

매뉴얼을 101 청크로 나눠 KoE5 로 임베딩

매뉴얼
99 조항

→

청킹
1 조항 = 1 청크 → 101 청크

→

임베딩
KoE5 · 숫자 1,024 개

→

Elasticsearch
단어 색인 + 벡터 저장

※ 101 청크인 이유 : 1 조항 = 1 청크가 원칙 · 너무 긴 조항 일부를 둘로 나눔

512 토큰

KoE5 가 한 번에 읽는 최대 길이 · 이전 후보는 128 토큰이라 청크 15~28% 를 끝까지 못 읽음

단어 검색 + 뜻 검색

nori 단어 색인과 KoE5 벡터를 Elasticsearch 한 곳에 함께 저장 → 벡터 전용 DB 를 따로 두지 않음

평가 원칙 네 가지

① **3 번 측정 · 최저값**
같은 검증 데이터로 모든 후보

② **규칙에 의해 결정**
AI 는 찾기와 정리

③ **학습용 ↔ 검증용**
교차 사용 금지

④ **맥 GPU 칩으로 학습**
리소스 한계로 고사양 GPU 사용 불가

채점도 규칙으로 (LLM 채점 안 함) · 운영 서버는 CPU 만 · 상업 이용 가능한 모델만 선택

직접 만든 골든셋 → 실제 발화와 공개 데이터로 재측정

295 개

직접 만든 골든셋 (검증 데이터)

점수가 실제보다 높게 나오기 쉬움

4,952 건

실제 상담원 발화

10,223 건

실제 고객 발화

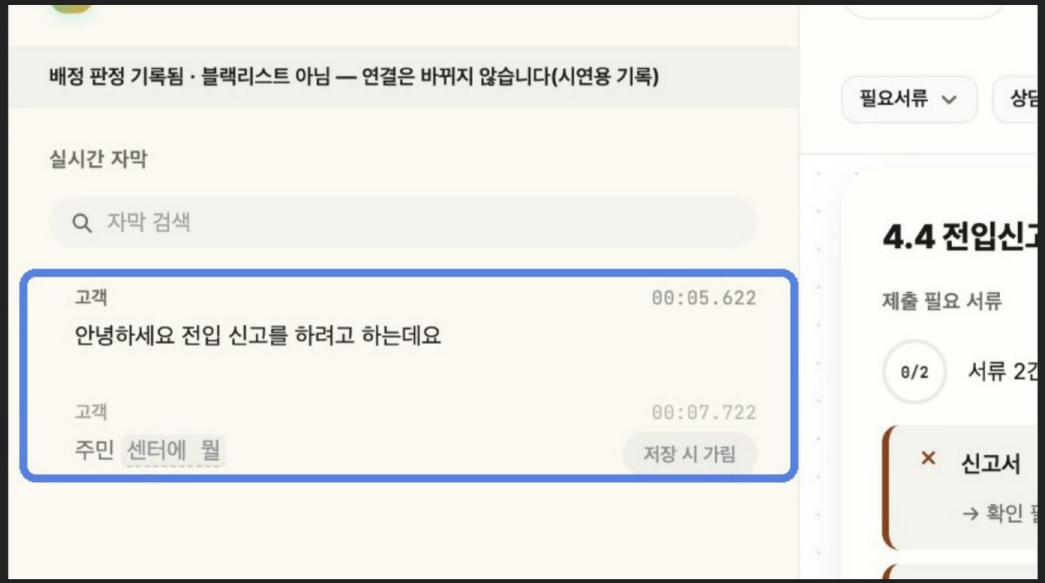
1,721 개

공개 욕설 데이터

3,770 개

공개 혐오 데이터

통화 음성을 실시간 자막으로 바꾸는 모델



시연 화면 · 통화 중 고객 말이 바로 자막으로 뜸

이 화면에서 하는 일

고객이 말하는 동안 음성을 글자로 바꿔
상담원 화면에 자막으로 띄웁니다

어떻게 비교했나

실제 상담 음성 20 개를 넣고
① 틀린 단어 비율
② 말하는 도중에 글자가 오는지
두 가지를 함께 봤습니다

비교한 후보

- Google STT
- Whisper

정확도보다 실시간 기능

선택 Google STT

틀린 단어 비율 (WER · 낮을수록 좋음)



AI Hub 상담 음성 20 발화 · 배치 비교

실시간 상담에 필요한 것

말하는 도중 글자

Google ✓ Whisper ×

목소리 구분

Google ✓ Whisper ×

고객 말에 맞는 매뉴얼 조항을 찾는 모델



시연 화면 · 찾은 조항으로 필요서류 카드를 띄움

이 화면에서 하는 일

고객 말과 가장 가까운 매뉴얼 조항을 찾아 , 필요서류 카드로 보여 줍니다

어떻게 비교했나

직접 만든 질문 102 문항으로
추천 5 개 안에 정답 조항이 있는지
오류 섞은 질문 · 외국어 질문도 측정
1 초 안에 답하는지도 확인

비교한 후보

- 단어 검색
- KoE5
- bge-m3
- 리랭커

추천 5 개 안에 정답이 든 비율

선택 KoE5

Recall@5 (높을수록 좋음)



운영 DB 2026-10-01 · 102 문항

bge-m3 ×

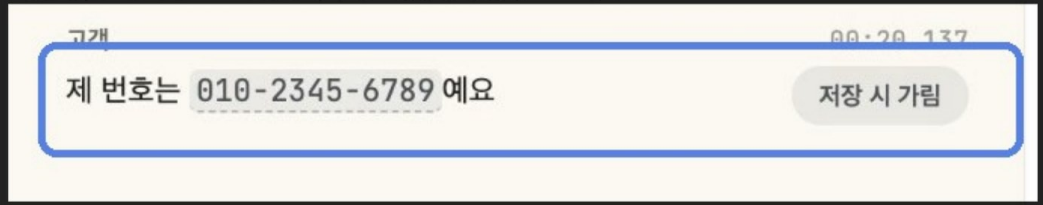
한국어 +1 문항
오류 · 외국어에선 같거나 낮음

리랭커 ×

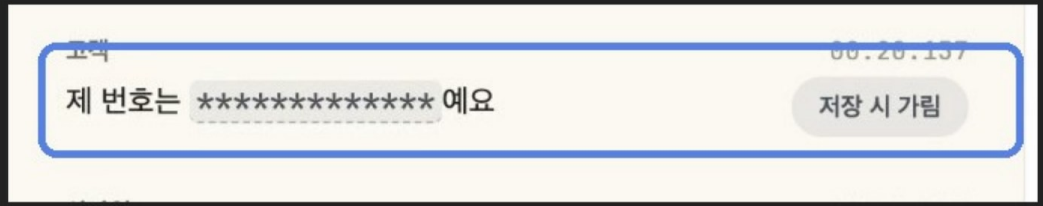
더 정확하지만
2.4 초 > 목표 1 초

저장할 때 개인정보를 가리는 모델

① 통화 중 — 상담원 화면에는 보임



② 저장된 기록 — 마스킹됨



시연 화면 · 통화 중엔 보이고 , 저장된 기록에선 마스킹

이 화면에서 하는 일

고객이 말한 전화번호 · 이름 같은
개인정보를 찾아 , 기록에 저장할 때 가립니다

어떻게 비교했나

음성 인식 오류를 20% 섞은 문장 34 건으로
놓친 개인정보가 몇 개인지
3 번 측정해 가장 낮은 값으로 비교

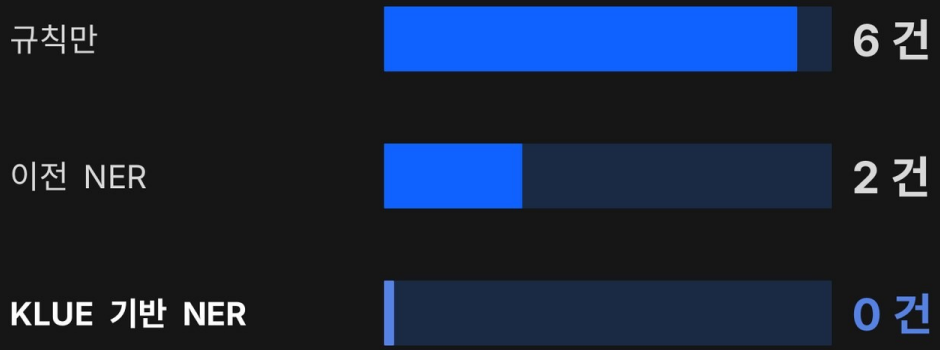
비교한 후보

- 규칙만
- 이전 NER
- KLUE 기반 NER

놓친 개인정보 0 건

선택 규칙 + KLUE NER

놓친 개인정보 (음성 인식 오류 20% · 34 건)



NER(Named Entity Recognition · 개체명 인식) = 이름 · 주소 같은 대상을 찾는 모델

KLUE(Korean Language Understanding Evaluation) = 한국어 이해 능력을 평가하는 공개 벤치마크 · 이 모델은 KLUE 계열 한국어 모델 기반

교체 이유

이전 모델의 학습 데이터 — 상업 이용 금지

대가

0.06 → 0.24 초

p95 · 100 번 중 95 번은 이 시간 안

고객 폭언을 찾아 상담원에게 알리는 모델



시연 화면 · 폭언이 잡히면 상담원 보호 알림

이 화면에서 하는 일

고객 발화에서 폭언을 찾아 , 상담원 화면에 보호 알림과 안내 문구를 띄웁니다

어떻게 비교했나

공개 욕설 데이터 1,721 문장에서 얼마나 찾는지

실제 상담 10,223 건에서 얼마나 잘못 잡는지

3 번 학습해 가장 낮은 값으로 비교

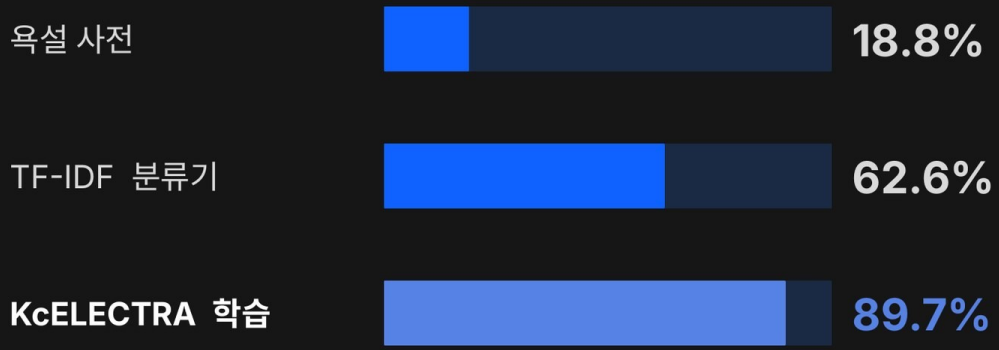
비교한 후보

- 욕설 사전
- TF-IDF
- KcELECTRA

사전에 없는 욕도 찾는다

선택 사전 + KcELECTRA

공개 욕설 데이터에서 찾아낸 비율



3 번 학습해 가장 나쁜 값 · 1,721 문장

46.4%

인터넷 댓글 기준 오탐 · 가벼운 비속어도 ' 욕 아님 ' 으로 표시된 데이터

10 / 10,223

실제 상담 통화에서 잘못 잡은 수

통화가 끝나면 요약 초안을 쓰는 모델



시연 화면 · 통화 후 처리 — 상담 요약 초안

이 화면에서 하는 일

통화 내용을 짧게 요약한 초안을 만들고 ,
상담원이 고친 뒤 확정합니다

어떻게 비교했나

같은 조건에서
① 꼭 담을 내용을 얼마나 담았는지
② 없는 숫자를 지어내지 않는지
두 가지를 봤습니다

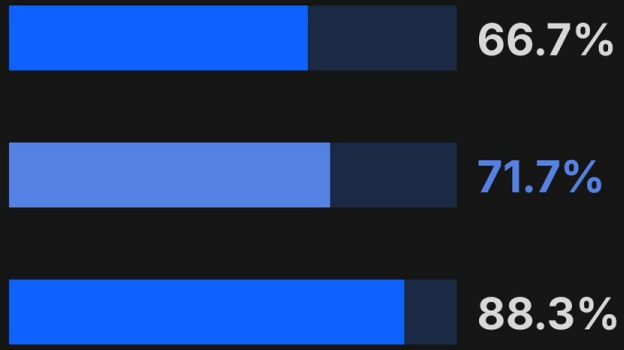
비교한 후보

EXAONE 4.0 | kanana 2.1B | 규칙 발췌 (참고)

꼭 담을 내용을 얼마나 담았나

선택 kanana 2.1B

포함률 (높을수록 좋음)



EXAONE 은 상업 이용 불가 · 같은 조건 비교

0 건

지어낸 숫자 (숫자만 검사)

큰 틀은 규칙
짧은 문장은 모델

폭언 모델이 「네 알겠습니다」까지 욕으로 잡았다

가장 어려웠던 문제

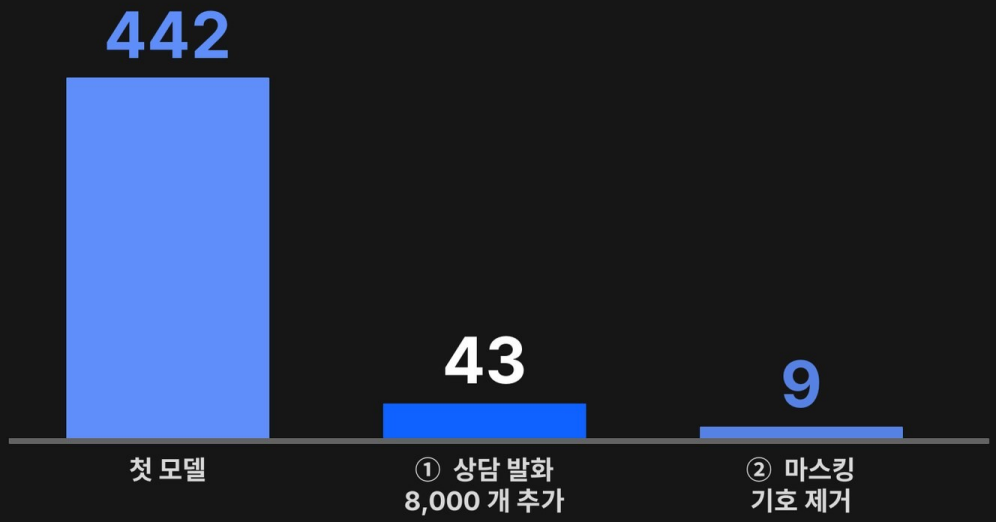
442 / 10,223

실제 상담 발화 중 잘못 잡은 수

원인

학습 데이터에 상담 말투가 없음
마스킹 기호 섞인 문장을 잘못 읽음

해결 방법



실제 상담 발화 10,223 건 기준 · 3 번 학습 중 가장 나쁜 값은 10

운영에 올렸고 , 일하는 방식이 바뀌었다

89.7% 공개 욕설 감지

≤ 10 건 실제 상담 오탐

10-15 운영 적용

배운 점

**공개 데이터 점수만으론
이 문제를 못 찾는다**

바뀐 일하는 방식

실제 상담 발화로 다시 측정 = 원칙

// 06 정리

학습 후 수치가 오른 모델만 사용했다

사용

폭언 분류 모델

새 이름 · 주소 모델

미사용

검색 추가 학습

리랭커 (1 초 넘음)

교체

EXAONE → kanana

이전 NER → KLUE 기반

배운 점 측정하지 않으면 좋아졌다고 말할 수 없다

교체 두 건 모두 사용 허가 (라이선스) 때문

기대효과 — 데이터와 리소스가 생기면 여기까지 올릴 수 있다

고성능 GPU 서버 → 음성 인식을 Whisper 로

35.8% → 13.7%

틀린 단어 비율 (낮을수록 좋음)

우리 서버 안에서 처리 → 음성이 밖으로 나가지 않음

녹음 20 개 비교에서 측정 · 실시간 동작은 서버에서 확인 필요

고성능 GPU 서버 → 조항 검색에 리랭커 추가

97.1% → 100%

추천 5 개 안에 정답 조항이 든 비율

검증 데이터 102 문항 모두 정답

지금은 2.4 초 (맥북 GPU 1.4 초) → 1 초 안에 돌릴 장비가 필요

실제 통화 · 외국어 데이터

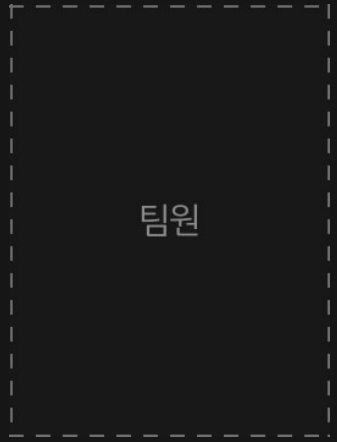
폭언 · 마스킹 모델을 실제 상담 말투로 추가 학습 · 외국어 질문으로 검색 모델 재학습 · 서툰 한국어 음성 학습

위 두 수치는 측정에서 확인한 값 · 실제 통화 데이터가 생기면 다시 측정해 확인합니다

// CallGuard · 인프라 담당

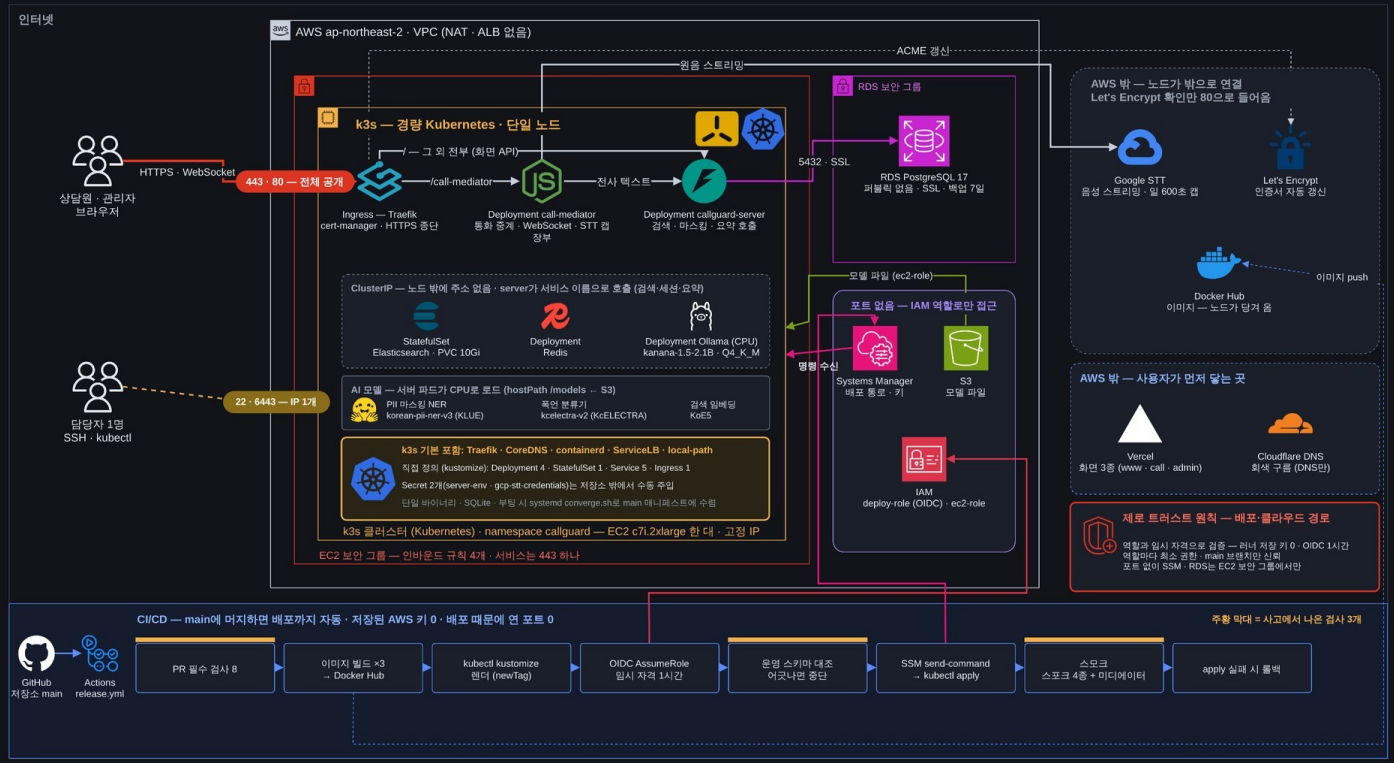
팀원

실시간 AI 상담원 어시스트의 인프라 및 배포



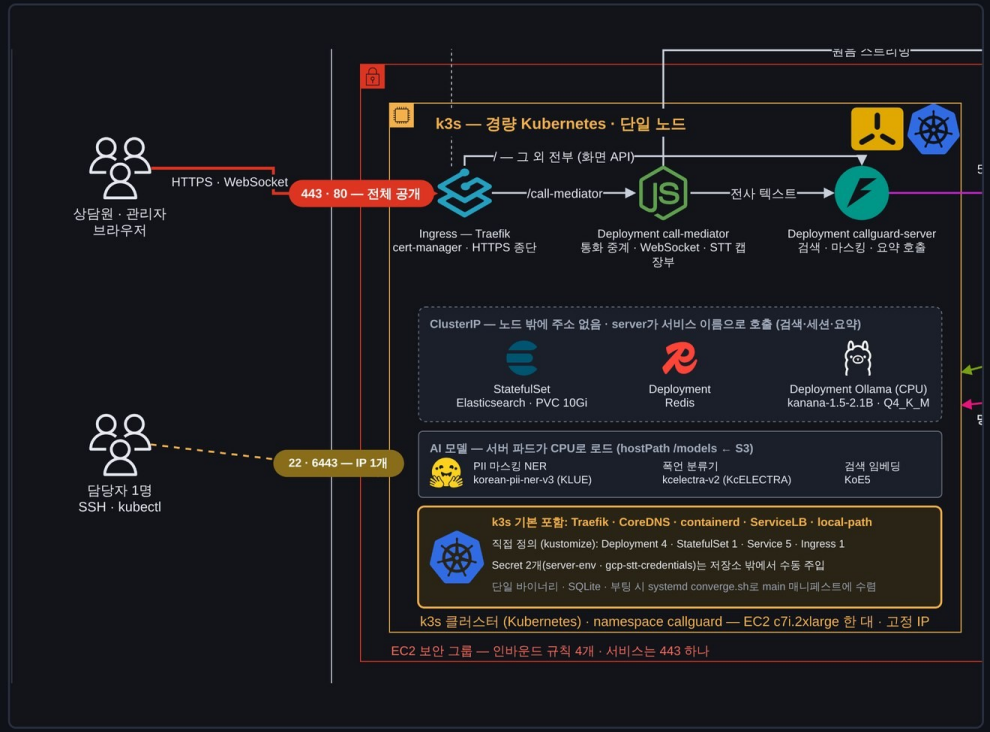
// 01 전체 구성

문은 하나, 서버는 한 대, 데이터는 밖에



// 01 전체 구성 · 들어오는 문

서비스가 들어오는 문은 443 하나



상담원·관리자 브라우저

HTTPS · WebSocket → 443 · 80은 Let's Encrypt 확인과 HTTPS 리다이렉트만

관리 포트 22 · 6443

담당자 IP 1개만 — SSH · kubectl

EC2 보안 그룹

인바운드 규칙 4개 (80 · 443 · 22 · 6443 포트)

// 01 전체 구성 · 노드 안

EC2 한 대 위 k3s — 서비스와 AI가 한 노드에



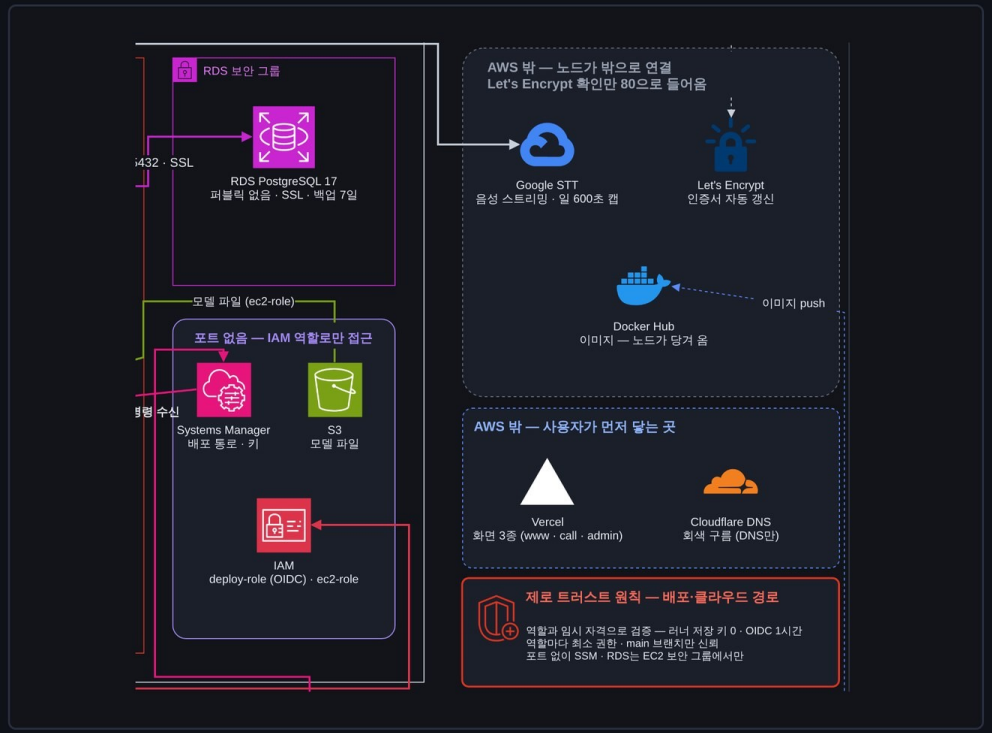
Ingress Traefik + cert-manager
k3s 내장 · HTTPS 종단 · 인증서 자동 발급

call-mediator → callguard-server
통화 중계 · STT 캡 → 검색 · 마스킹 · 요약 호출

ClusterIP — 노드 밖에 주소 없음
Elasticsearch · Redis · Ollama(kanana 2.1B, CPU) · AI 모델 셋은 서버 파드가 직접 적재

// 01 전체 구성 · 관리형 자원

데이터와 배포 통로 — 포트 없이, 역할로만



RDS PostgreSQL 17
보안 그룹 안 · 퍼블릭 없음 · SSL · 자동 백업 7일

SSM · S3 · IAM
포트 없음 — deploy-role(OIDC) · ec2-role 역할로만

AWS 밖
Google STT · Let's Encrypt · Docker Hub는 노드가 밖으로만(인바운드 0) · 사용자가 먼저 닿는 Vercel · Cloudflare DNS

// 02 설계 결정

EC2 한 대에 전부 — 관리형 서비스를 못 쓴 게 아니라, 안 써도 되게

① 클러스터

k3s 하나로 EKS · ALB · NAT를 대신

Ingress는 내장 Traefik, 인증서는 cert-manager가 자동 발급 — 운영 인력 1명이 감당할 수 있는 크기

매니페스트는 kustomize로 저장소와 diff 0

② 모델은 떼어낼 수 있게

지금은 CPU 한 노드 — 확장은 노드 추가로

kanana 2.1B 4-bit 양자화로 통화 후 요약 3.7초 · 모델 3개는 CPU로 — 모델 파드는 서비스 파드와 분리돼 있어 노드만 바꾸면 된다

GPU는 노드 한 대 추가로 붙는다 → 다음 장

③ 데이터만 밖에

서버는 다시 만들 수 있지만 데이터는 못 만든다

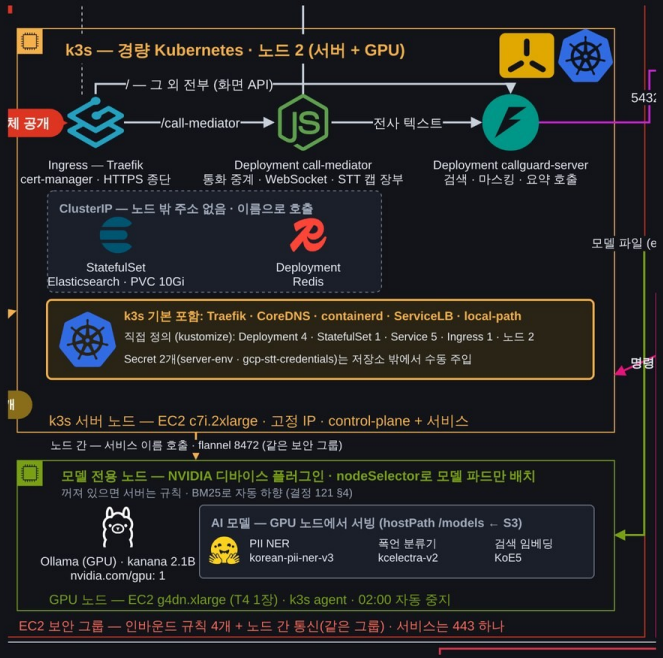
RDS PostgreSQL — 자동 백업 7일 · 퍼블릭 접근 차단 · SSL만 허용 · EC2 보안 그룹에서만 접속

노드가 죽어도 데이터는 남는다

설계로 아낀 값 — EKS 6주 $102 · ALB 월 $20+ · NAT 월 $35+ · GPU 6주 $597

// 02 설계 결정 · GPU 확장 설계

모델은 떼어낼 수 있게 — GPU는 노드 한 대 추가로



확장 설계안

GPU 노드 1대 추가 — g4dn.xlarge (T4) · k3s agent

같은 보안 그룹 · 노드 간 flannel · S3 모델 파일은 GPU 노드로

모델 파드만 옮긴다 — nodeSelector

서비스 노드 매니페스트는 그대로 · Kubernetes라 재구축이 아니라 노드 추가

켜진 시간만 — 02:00 자동 중지

24/7이면 6주 $597 · 꺼져 있으면 서버는 규칙 · BM25로 자동 하향 (결정 121 § 4)

// 03 배포 파이프라인

main에 머지하면 끝 — 사람 묶이던 일 넷을 파이프라인으로



흔한 방식 — SSH 키를 두고 포트를 연다

지금은 — OIDC 1시간 자격 + SSM 폴링 · 저장된 키 0 · 열린 포트 0

러너 IP가 매번 바뀌어 포트를 열면 인터넷에 여는 것 · 비밀값은 Parameter Store

사람 묶이던 것 — 이미지 태그 올리기

지금은 — PR 필수 검사가 머지 전에 잡는다

잊으면 옛 이미지가 그대로 도는 사고 → 5장

사람 묶이던 것 — 운영 DB에 스키마 적용

지금은 — 배포 직전 운영 DB에 붙어 대조, 어긋나면 중단

사람 묶 — 배포 뒤 확인

지금은 — /health 스포크 4종 · 미디에이터 키·캡·토큰 확인 · apply 실패면 이전 태그로

꺼져 있던 인스턴스도 부팅 때 스스로 따라잡는다

// 04 트러블슈팅 · 이미지 태그 누락

코드는 새것인데 배포는 옛것 — 릴리스 실패 5건 중 4건이 같은 원인

증상

머지 뒤 릴리스 실패 · 코드는 바뀌었는데 서버에는 옛 이미지가 그대로 돎 (#55 · #59 · #61 · #73)

원인

이미지 태그(**newTag**)를 안 올림 → 노드는 같은 태그면 캐시 이미지 사용 (IfNotPresent) · 검사가 main push 뒤에만 돌아 「머지하고 나서야」 알았다

조치

판정만 하는 별도 워크플로 **tag-check**를 PR 필수 검사로 — 이미지도 안 굽고 AWS도 안 건드림 · 레지스트리 조회 실패도 불통과

재발 방지

PR 단계에서 차단 5건+ · 머지 뒤 재발 0 · 되돌리려 PR 하나 더 내던 일이 사라짐

같은 교훈으로 검사가 된 사고

운영 DB 미연결
/health는 사흘 「정상」 → 실제 연결 검사

코드가 DB보다 먼저
500 → 배포 직전 스키마 대조

사람이 직접 해야 하는 절차를, 파이프라인이 자동 검사

// 05 정리

인프라 담당 1명 · 8주 · $400 — 통화 중 상담원을 돕는 서비스를 운영까지

1대

EC2 + k3s에 서비스 · AI 전부

EKS · ALB · NAT 없이 — 데이터만 RDS로

0 · 0

저장된 키 0 · 배포용 포트 0

OIDC 1시간 자격 + SSM · 서비스 문은 443 하나

8

PR 필수 검사 — 머지하면 배포까지

태그 · 스키마 · 스모크, 사람 기억 대신 검사

5 → 0

릴리스 실패 5건 → 머지 뒤 재발 0건

사고 셋이 파이프라인 검사 셋으로

보여드린 것 ①

주어진 리소스 안에서 운영 가능한 크기를 고르는 판단

관리형 서비스를 못 쓴 게 아니라 안 써도 되게 — k3s 한 대, 모델은 떼어낼 수 있게

보여드린 것 ②

사람 절차를 자동 검사로 바꾸는 습관

사고가 나면 재발 방지를 문서가 아니라 파이프라인에 — 머지 전에

보여드린 것 ③

설계부터 운영까지 혼자 끝까지, 기록과 함께

AWS · k3s · CI/CD · PM — 결정 기록과 런북으로 다음 사람이 이어받을 수 있게

서비스는 400달러로 · 배포는 사람 개입 없이 · 사고 재발 0건

팀원 소개 — SOLIDBOB 4 명 · 8 주

01 프론트엔드

팀원

팀원

- 화면 3 종 개발
- 실시간 자막 · 서류 체크리스트
- 계약 변경 20 회 + 대응

자동 테스트 92 개 전부 통과

React · TypeScript · Zustand

02 백엔드

팀원

팀원

- 허브 - 스포크 서버 구조
- 개인정보 마스킹 · 판정
- 블랙리스트 승인 API

놓친 개인정보 0 / 40

Python · FastAPI · PostgreSQL

03 AI



류준

- 모델 5 종 비교 · 선택
- 폭언 모델 학습 · 운영 적용
- 검증 데이터 295 개 제작

조항 검색 정답률 97.1%

KoE5 · KcELECTRA · kanana

04 인프라

팀원

팀원

- AWS · k3s 운영 서버
- 자동 배포 · 필수 검사 8 종
- 보안 · 모니터링

외부 입구 1 개 · 저장된 키 0 개

AWS EC2 · k3s · GitHub Actions

Q & A

감사합니다