

사진 한 장으로 시작하는 나만의 공간 꾸미기

방꾸 | AI기반 가구 배치 시뮬레이터





목 차

- 01. 방꾸 소개
- 02. 문제 정의
- 03. 방꾸 차별점
- 04. 모델 선정
- 05. 시연 영상
- 06. 트러블 슈팅 & 기술 아키텍처
- 07. 협업 방식 및 팀원 역할

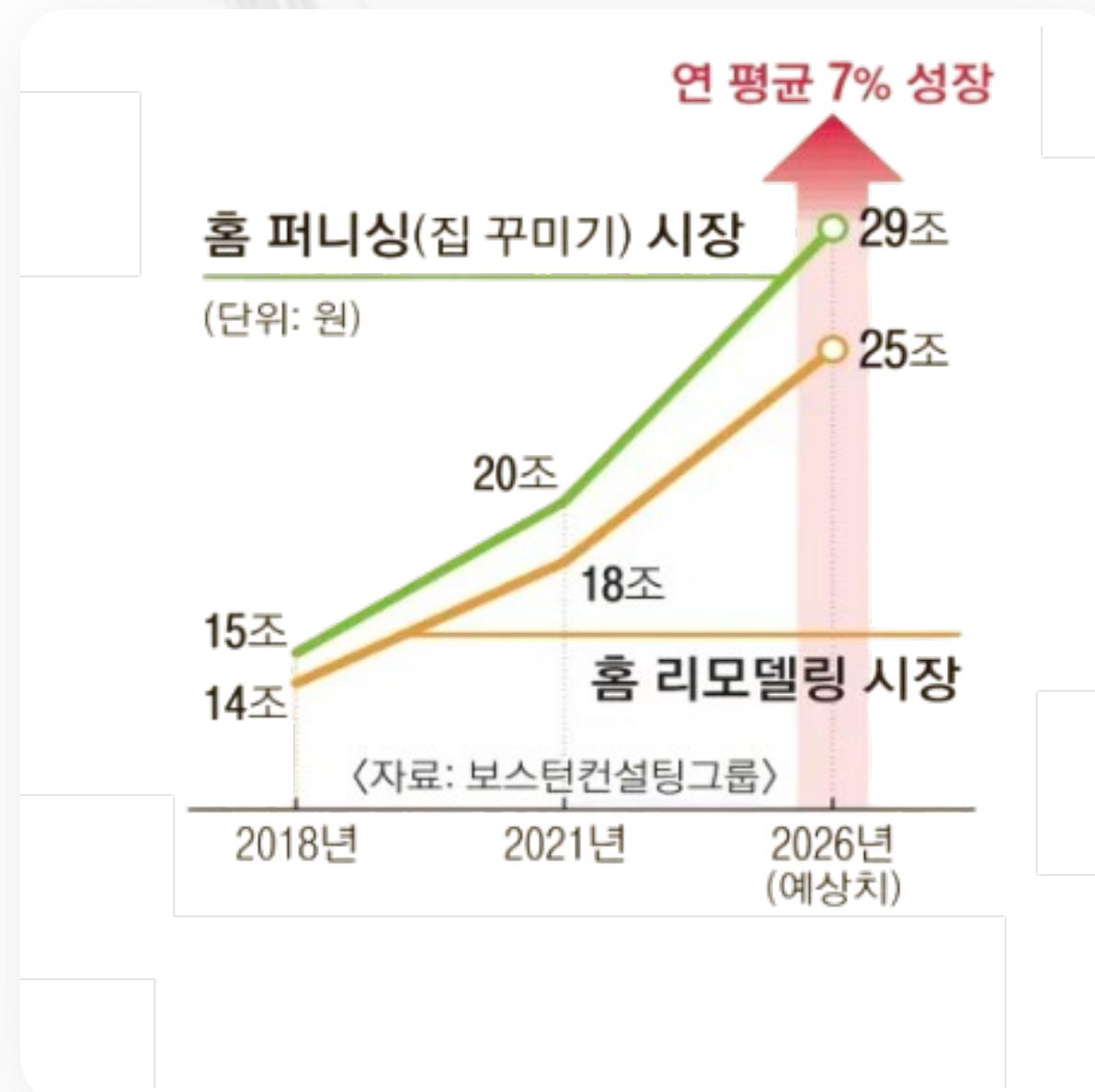
소개

방꾸, AI 가구 배치 시뮬레이터

사진 한 장으로 기존 가구 인식은 물론
새 가구 추천, 배치 결과까지 확인 가능한 서비스



시장 배경 & 문제 정의



사용자

- 기존 가구와 어울릴지 시각적으로 예측 어려움
- 마음에 안 들 경우 반품 부담
- 반품 과정 번거로움 + 비용 발생



판매자

- 반품 1건당 손실 발생
- 물류비, CS 인력 낭비
- 수익성 저하

기존 유사 서비스 분석

IKEA Kreativ

- 브랜드 제한
- 기존 제품 반영 불가

오늘의집

- 3D 도면 기반 UX 복잡
- 진입장벽 높음
- 기존 제품 반영 불가

방꾸

- 기존 제품 반영 가능
- AI 검색 지원
- 실제 공간 배치 시뮬레이션 제공
- 쉽고 직관적인 UI/UX 설계

주요 기능

방꾸

관리자 페이지

TALK 로그아웃

튜토리얼

초기 이미지

이미지 생성

저장

AI 추천 조건

검색

나의 가구 추천 가구 내 인테리어

이미지 삭제

가구 추가, 이동, 제거가 완료되면 배치 결과 보기버튼을 눌러주세요

방꾸는 실내 디자인 앱으로, 사용자가 원하는 가구와 소품을 배치해 볼 수 있는 기능을 제공합니다. 이 화면은 실제 거실을 모델로 한 배치 예시입니다. 사용자는 가구를 추가하거나 移動하거나 제거할 때마다 배치 결과를 확인할 수 있습니다.

가구 컨트롤

모델 선정 이유



Sam2 GroundingDINO

&

Gemini 2.0 Flash Image Generation



Box 영역 인식



정확한 객체 탐지



객체 세분화 방지



Inpainting



Outpainting

모델 선정 - inpainting을 위한 객체 탐지 및 이미지 분할

1) Box 영역 인식



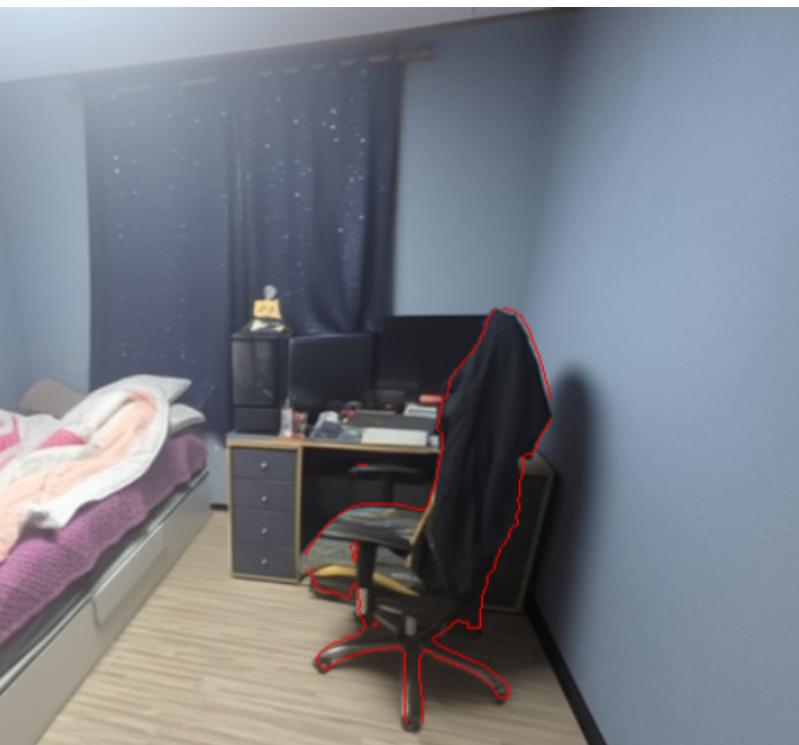
2) 정확한 객체 탐지



카테고리	Yolov8	Detectron2	GroundingDINO
속도	200-300 FPS	30-80 FPS	3-10 FPS
정확도	30%	70%	90%
GPU 메모리	1~3 GB	4~8 GB	10~16 GB
SAM2 조합	실패	실패	성공
결론	사용 불가	사용 불가	사용 가능

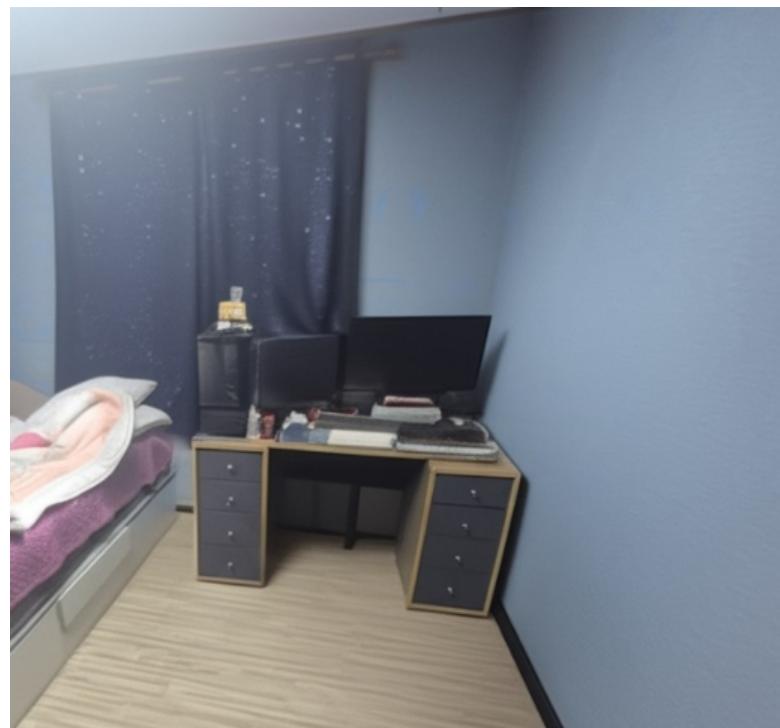
모델 선정 - Inpainting 및 Image Generation

Before



Inpainting 객체 선택

After



객체 삭제 완료

카테고리	LaMa	Deep Fill v2	Gemini 2.0 Flash Image Generation
속도	100ms	300~500ms	1000~2000ms
정확도	30%	70%	90%
텍스트 조건 지원	불가능	불가능	가능
출력품질	이질감이 있음	이질감이 있음	자연스러움
결론	사용 불가	사용 불가	사용 가능

AI 검색 & 추천

기능 고도화



CLIP & E5 & Gemini 2.0



하이브리드 서치 고도화



AI Agent기반 제품 추천



FAISS를 이용한 확장성

AI 검색 및 추천

1초

성능

- 키워드 매칭 & BM25
- 사용자 피드백 기반 점수 적용

속도

- 서치 인덱스
- 파이프라인 수정
- 의미기반의 고속 추천

모델 선정 - 텍스트 임베딩

1) 텍스트와 유사도 측정

☰ 갈색 테이블

갈색 테이블 (17)

추천 이유



쿼리 '갈색 테이블' 와 결합 유사도
0.739

시그니처 라운드 테이블 1100

라운드 테이블

₩1,090,000

STRANDTORP 스트란드토르프 / MÅRENÄS 모레네스

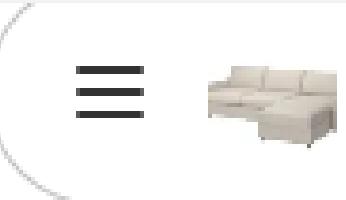
테이블+암체어6, 150/205/260 cm

₩1,843,000

카테고리	Sentence-BERT	LaBSE	intfloat/ e5-base-v2
속도	매우 빠름	느림	빠름
유사도 성능	높음 (긴 문장에 취약)	높음 (특히 다국어)	매우 높음 (SOTA 급)
모델 크기	33M	470M	110M
장점	초경량, 실시간 서비스 최적	다국어 지원 매우 우수	긴 문장도 잘 다룸, 검색 최적화
결론	사용 불가	사용 불가	사용 가능

모델 선정 - 이미지 임베딩

1) 이미지와 유사도 측정


소파 (49)

VIMLE 빙레



추천 이유
쿼리 '소파' 와 결합 유사도 0.766

PÄRUP 페루프



카테고리	OpenAI Clip	LAION-CLIP	jina-clip-v2
추론 속도	느림 (ViT-L/32)	빠름	빠름 (ViT-B/32)
성능 (텍스트 - 이미지 매칭)	최고급 (ViT-L)	준수 (ViT-B/32)	매우 우수 (도메인 적응 좋음)
메모리 사용량	10GB 이상	4~6GB	4~6GB
로컬 배포	불가	가능	가능
결론	사용 불가	사용 불가	사용 가능

시연 영상

내 방을, 내 손으로, AI와 함께

<https://youtu.be/WQUKahTEBVA>

트러블 슈팅

문제 : [객체탐지] SAM2가 객체를 과도하게 분할 → 탐지 성능 저하

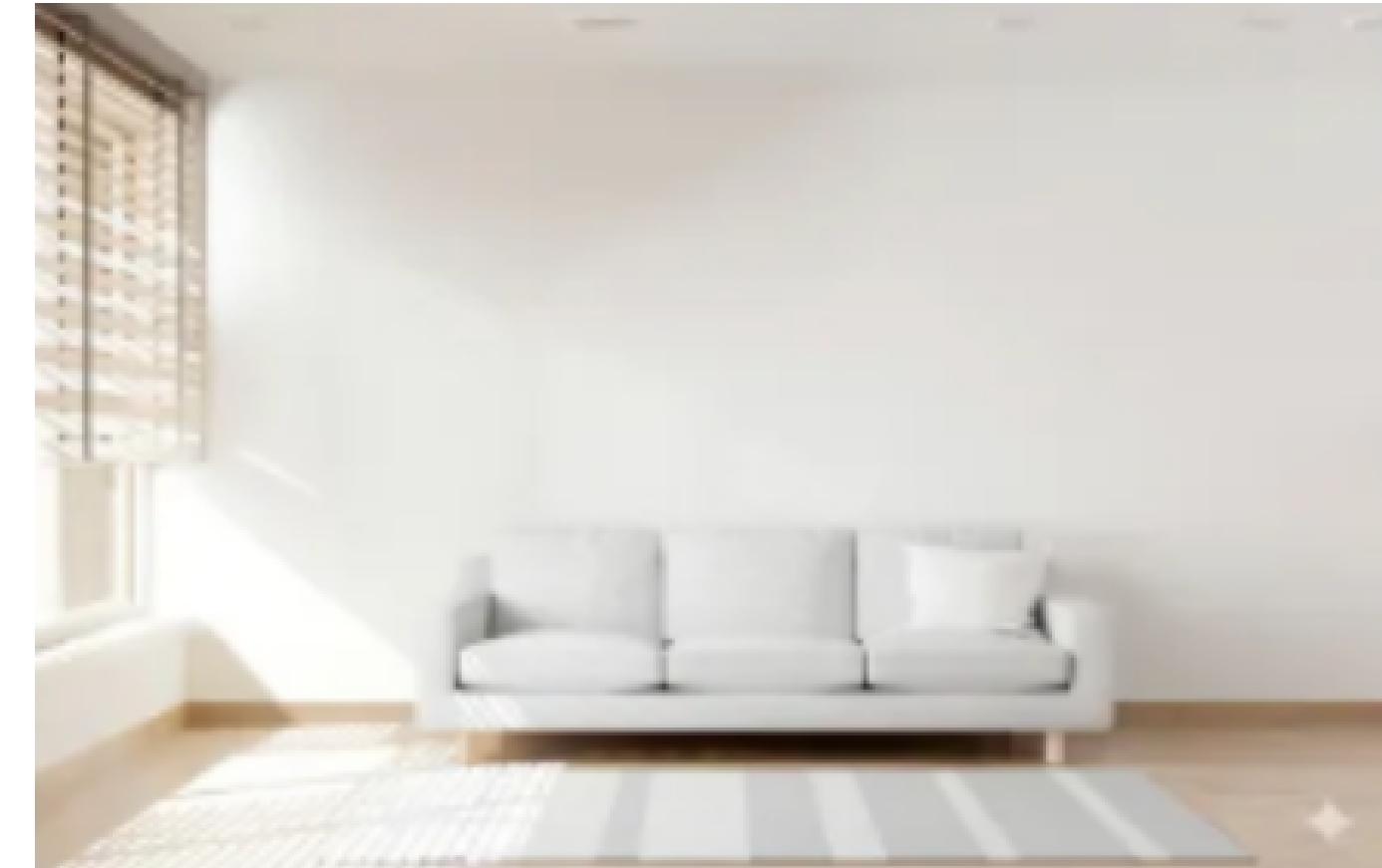
해결 : Grounding DINO 결합 → 단일 객체 단위 탐지 보완



트러블 슈팅

문제 : [Gemini] 객체 삭제 시 오작동 발생

해결 : 삭제할 객체를 마스크(빨간색)로 시각적으로 지정, 객체 이름을 구체화



트러블 슈팅

문제 : [Gemini] 마스킹이 이미지 생성 모델에 정확히 인식되지 않음

해결 : 윤곽선(edge mask) 방식으로 대상 경계를 모델에게 시각적으로 명확히 전달

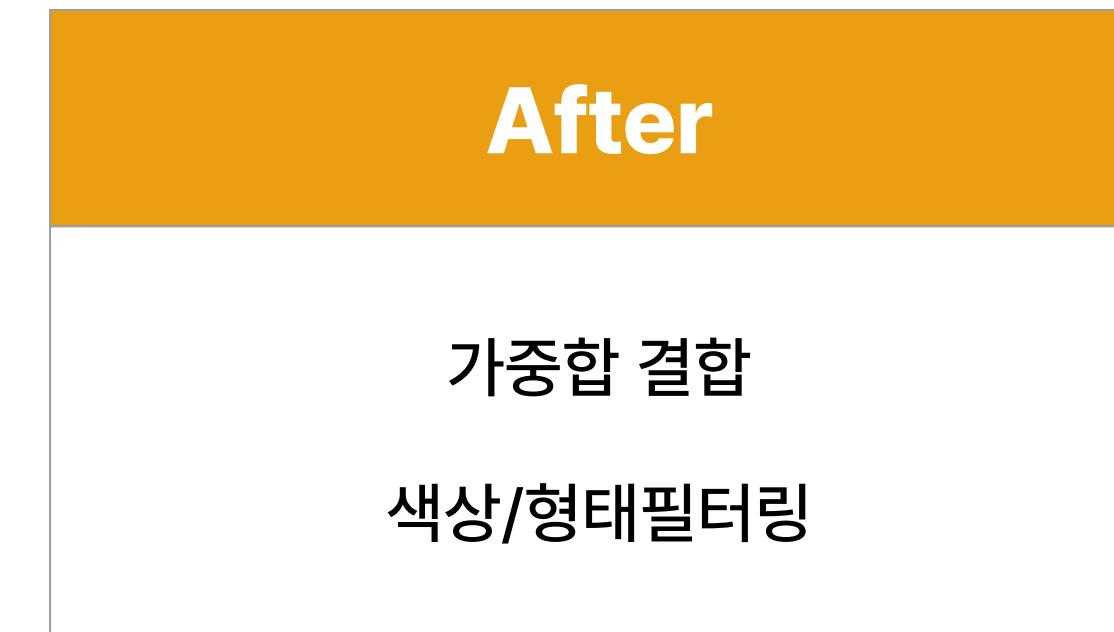
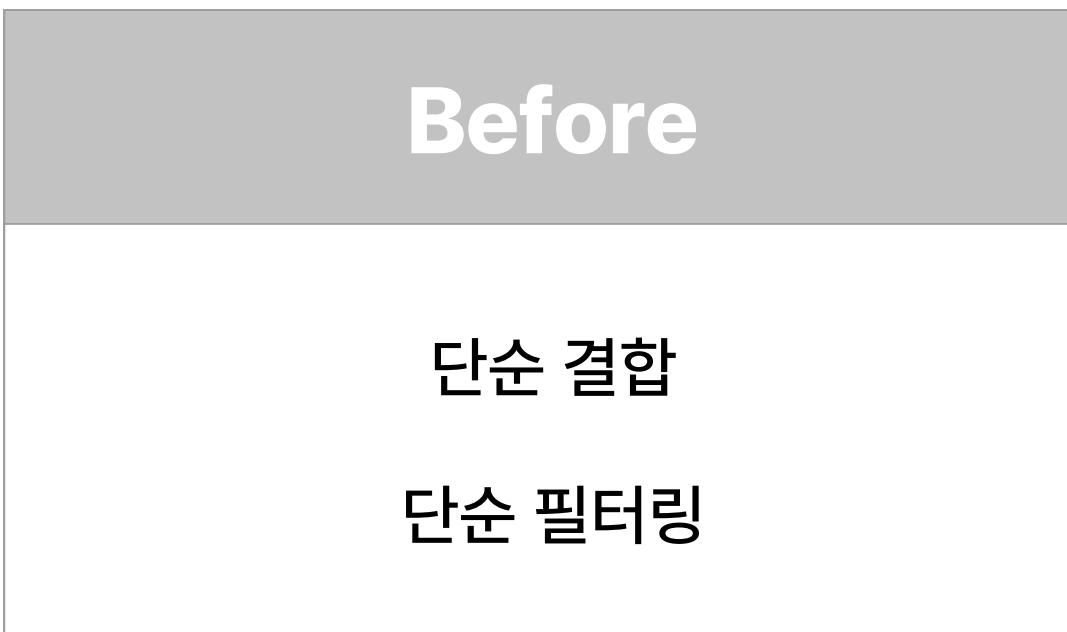


트러블 슈팅

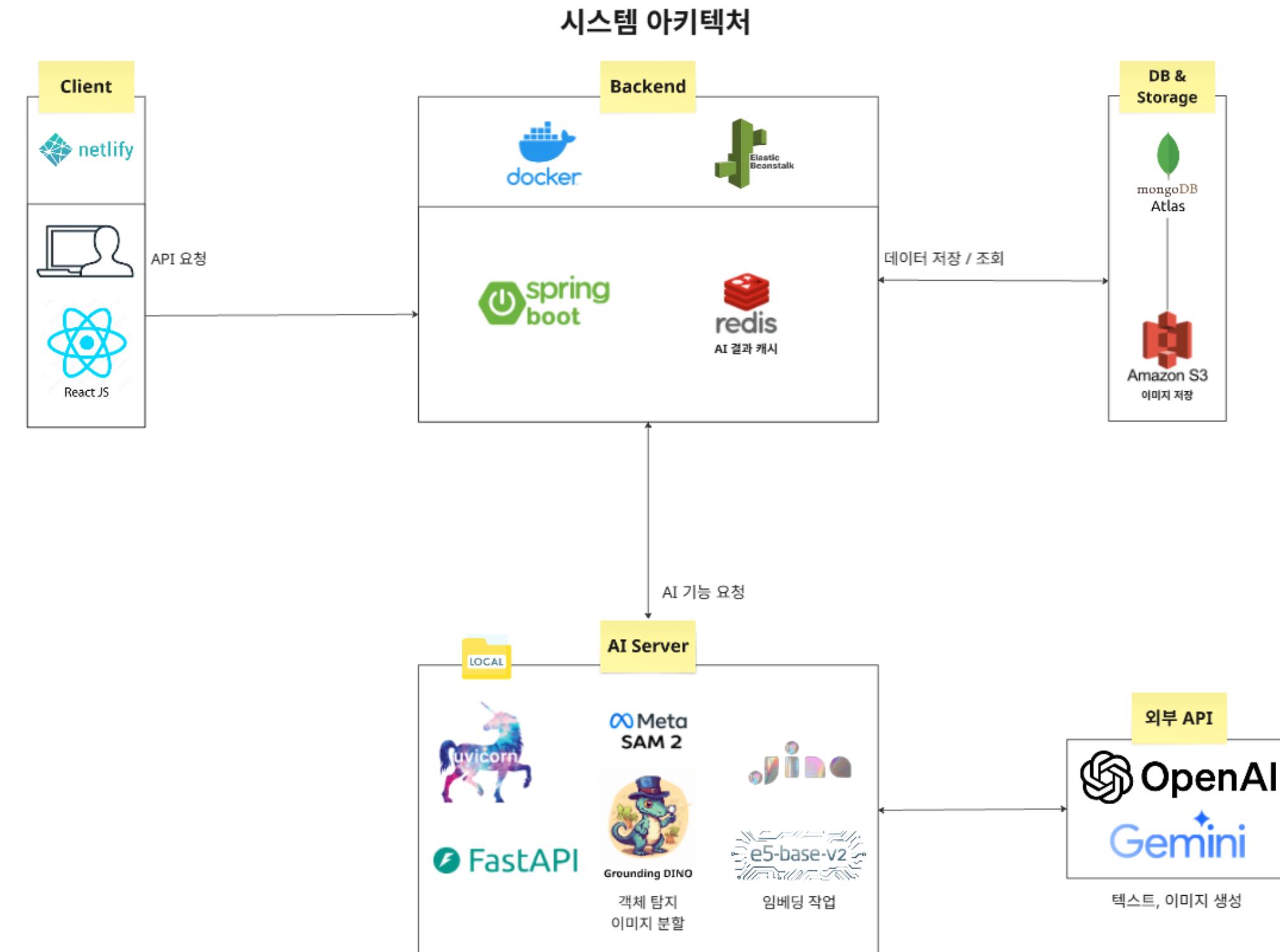
문제 : [검색] 이미지 기반 검색 정확도 한계

개선 : 가중합 적용, 재랭킹, 색상/형태 필터링 강화

결과 : 불필요한 제품을 제외하려 검색 품질 향상



기술 아키텍처 & 스택

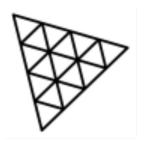


기술 스택 요약

프론트엔드



React
19.1.0



Three.js

백엔드



Java 17



Spring Boot
3.4.4

AI 서버



Python
3.10



FastAPI
0.115.12

객체 인식 / 분할

Grounding DINO
+
SAM2

이미지 생성



Gemini 2.0 Flash (exp)

텍스트 이미지 임베딩



jina-clip-v2 e5-base-v2

텍스트 임베딩



OpenAI GPT-4o

상태관리



redis

Redis

이미지 저장



AWS
S3



MongoDB atlas

협업 방식

01

GIT 브랜치 전략 / API 명세 문서화

Git Flow 기반 브랜치 운영 (feature → dev → main)
Slack, Notion 기반 API 명세 작성 및 버전 관리
API 변경사항 발생 시 즉시 업데이트

02

NOTION, FIGMA 기반 실시간 기획·설계

Notion을 통한 기획안 작성 및 공유
Figma를 활용한 화면 설계 및 디자인 수정 실시간 반영
기획-디자인-개발 간 간극 최소화

03

기능별 분담 + 기술 연계 설계

기능별 담당자 지정 후 독립 개발
공통 컴포넌트/모듈 명세화 및 공유
종합 테스트를 통해 통합성 검증 및 최종 연계

팀원 역할



김범석/팀장

- 프로젝트 전반 관리
- 팀 협업 흐름 조율
- 최종 발표 콘텐츠 총괄
- (Dino)객체 탐지 기능 구현
- (SAM2)Segment 기능 구현
- 정확도 향상을 위한 프롬프트
- 각 모델에 대한 엔지니어링 적용
- 2D 캔버스 업로딩 기능 구현
- 가구 썸네일 생성 기능 구현
- 3D 캔버스 Three.js 세팅 및 구현
- 캔버스 관련 사용자 예외 처리 구현



김동규

- 텍스트·이미지 유사도 기반
- 가구 검색 기능 개발
- MongoDB 연동
- AI 검색 및 추천 알고리즘 설계
- 유사도 스코어링 및 후보 필터링 로직 구현
- 대용량 데이터 효율 처리를 위한 Bulk 연산 및 성능 최적화 실험 주도
- 배포 전반 리딩: Docker 및 EC2 활용한 백엔드 서비스 배포 및 연동 환경 구축



김병훈

- OAuth2 기반 소셜 로그인 인증 로직 구현, 사용자 정보 처리 및 세션 연동
- 관리자 페이지 UI 및 백오피스 기능 개발, 접근 권한 설계 및 데이터 관리 구조 설계
- 역할 기반 접근 제어 설계 및 시나리오 별 권한 분기 처리
- AI 기반 가구 추천 기능 연동, 백오피스에서 후보군 제어 및 추천 결과 확인 구조 설계
- FAISS를 이용한 추천 시간 단축



김태원

- 서비스 목적 및 기능 방향 정리,
- 흐름 중심 구조 설계
- 팀 일정 · 역할 조율,
- 발표 흐름 · 배포 전략 등 구상
- AI 편집 기능의
- 처리 흐름(탐지-마스킹-생성) 및
- 기술 연계 구조를 설계
- 객체 탐지 및 이미지 생성 모델 실험, 프롬프트 구조 최적화
- Redis · AWS S3 연동



박지은

- Figma를 활용한 화면 설계 및 디자인
- 사용자 흐름 기반 UI/UX 구조 설계
- 프론트 전반 리딩
- 컴포넌트 기반 인터페이스 구현
- 텍스트, 이미지, 이미지+텍스트 조합, 음성 검색 기능 백엔드 API 연동
- 사용자 행동 기반 튜토리얼 시스템을 구축, 자연스러운 온보딩
- 최종 발표 PPT 작업

발표를 마치며...

Thank you.