



# 개인 맞춤형 식품 알레르기 관리 시스템

YOLO 기반 음식 인식 및 LLM 기반 성분 분석 통합 서비스

인공지능학과 21012047 강민성

# 프로젝트 기획 배경 및 필요성

식품 알레르기는 경미한 증상부터 생명을 위협하는 아나필락시스 쇼크까지 유발할 수 있는 심각한 건강 문제입니다. 현재 대부분의 소비자는 식품 성분표를 직접 확인해야 하며, 이는 번거로울 뿐만 아니라 부주의 시 심각한 사고로 이어질 수 있습니다.

기존 식품 검색 서비스는 다음과 같은 문제점을 가지고 있습니다.

- 이미지 기반 인식 기능 부재
- 개인 알레르기 정보를 반영한 맞춤형 판단 미제공
- 성분표 자동 분석 시스템 부족

본 프로젝트는 "이미지 기반 음식 인식 + LLM 성분 분석 + 개인 맞춤형 알레르기 위험 판단"을 하나의 서비스로 통합하여 이러한 문제들을 해결하고자 합니다.

# 프로젝트 목표 및 범위

본 프로젝트는 사용자가 음식 이미지 업로드만으로 알레르기 위험 여부를 즉시 확인할 수 있는 통합 솔루션을 제공하는 것을 목표로 합니다.

1

## 음식 이미지 인식

이미지 업로드 시 YOLO 모델을 활용하여 자동으로 음식을 인식합니다.

2

## AI 기반 성분 분석

성분표를 자동 OCR로 분석하고 LLM을 통해 정확한 성분 정보를 추출합니다.

3

## 개인 맞춤형 위험 판단

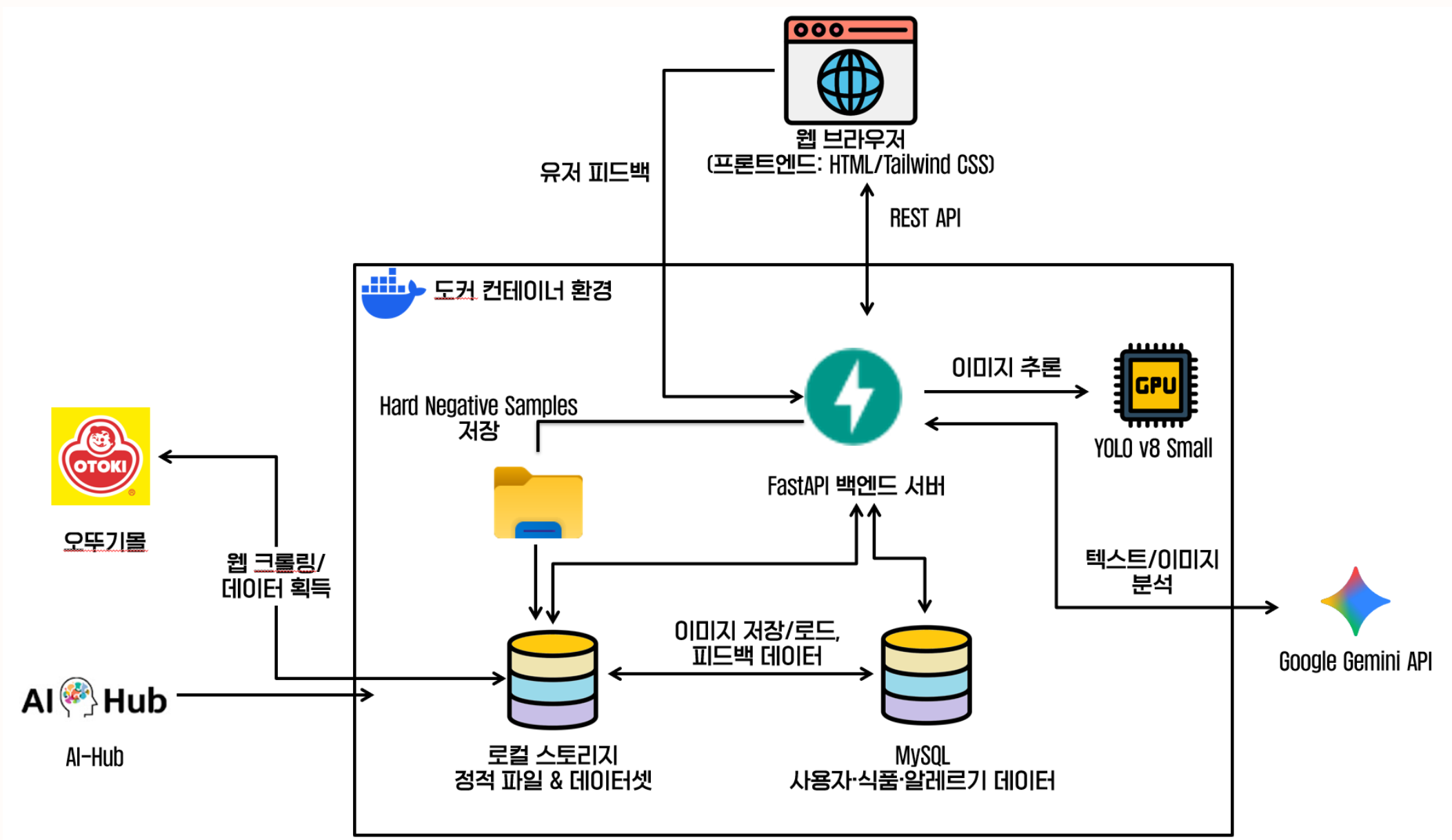
사용자 알레르기 정보와 분석된 성분을 비교하여 위험 여부를 즉시 판단합니다.

4

## 웹 서비스 구축

검색, 관리, 피드백 기능을 포함한 웹 서비스를 개발하고 Docker 기반 배포 환경을 구축합니다.

# 전체 시스템 아키텍처



본 시스템은 사용자 친화적인 웹 인터페이스와 강력한 백엔드 AI 모델을 결합하여 seamless한 알레르기 관리 경험을 제공합니다.

# 데이터 수집 및 전처리

1

## 오뚜기몰 기반 웹 크롤링

오뚜기몰에서 다양한 상품 이미지, 상세 성분표, 알레르기 정보를 체계적으로 수집했습니다.

2

## 성분표 텍스트 추출 및 정규화

수집된 성분표 이미지에서 OCR 기술을 활용하여 텍스트 데이터를 추출하고, AI가 이해하기 쉬운 형태로 정규화했습니다.

3

## 이미지 최적화

- 모든 이미지를 서버 로컬 저장 방식으로 변환하여 처리 속도 향상
- WebP 포맷 적용 및 80% 압축을 통해 웹 트래픽을 효율적으로 최적화

# 데이터베이스 설계

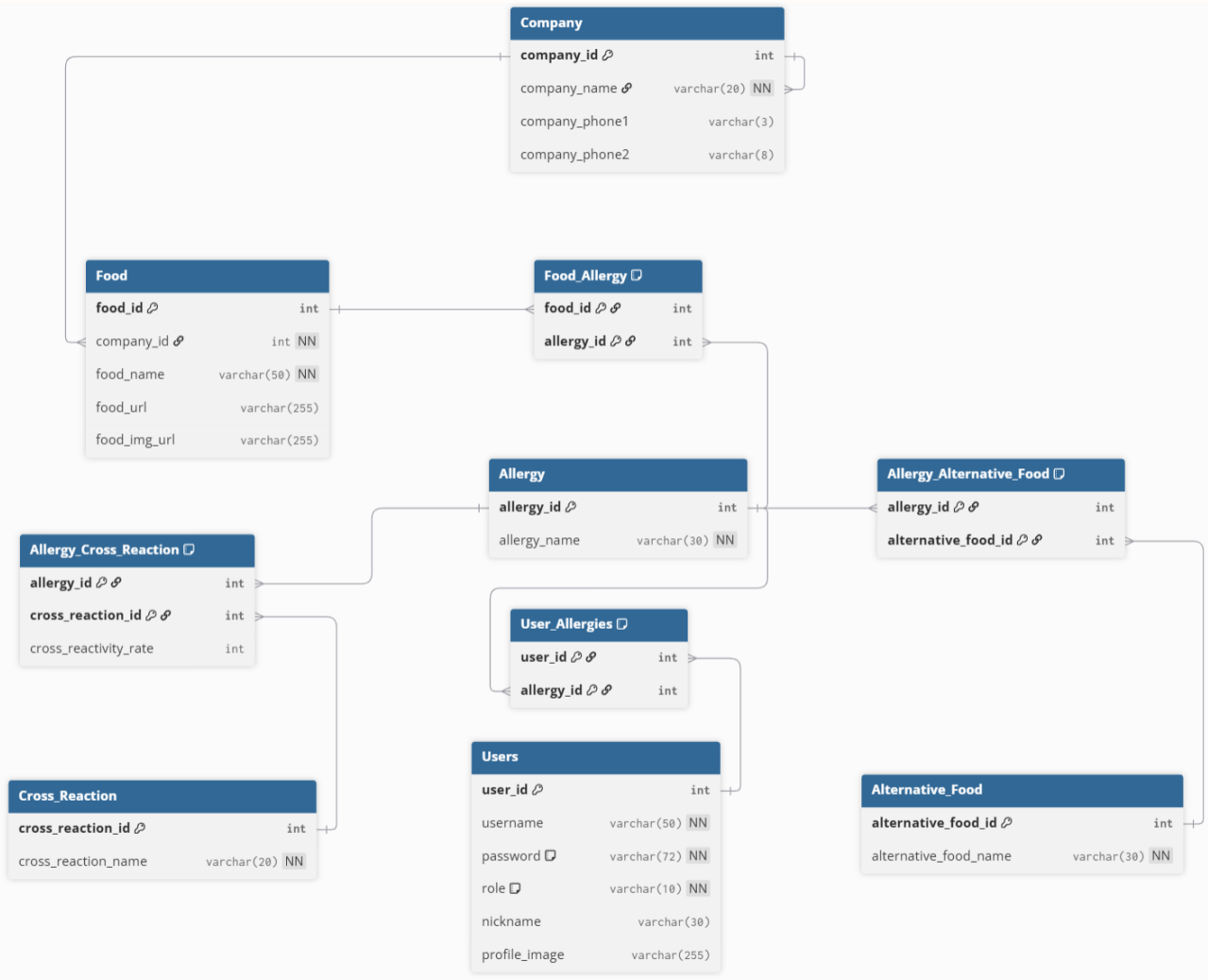
MySQL 기반의 관계형 데이터베이스는 제3정규화를 적용하여 데이터 중복을 최소화하고 일관성을 유지합니다. 외래키 및 무결성 제약조건을 통해 데이터의 신뢰성을 확보했습니다.

## 핵심 테이블

- Users: 사용자 정보
- Food: 음식 정보
- Allergy: 알레르기 유발 성분
- Company: 제조사 정보
- Cross Reaction: 교차 반응 정보







## 관계 테이블

- User\_Allergy: 사용자-알레르기 매핑
- Food\_Allergy: 음식-알레르기 매핑
- Allergy\_Cross\_Reaction: 알레르기-교차 반응 매핑



# AI 모델 구성: YOLO v8 Small

음식 이미지 인식 시스템은 웹 서비스 환경에서 실시간 추론이 가능한 객체 탐지 모델을 목표로 설계되었습니다.  
이에 따라 정확도와 추론 속도의 균형이 우수한 YOLO v8 Small 모델을 최종 선정하였습니다.

<div>모델</div> <div>YOLO v8 Small (Object Detection)</div>	
<div>적용 문제</div> <div>571개 음식 클래스 다중 객체 탐지</div>	
<div>입력 해상도</div> <div>640 x 640</div>	
<div>추론 환경</div> <div>GPU 기반 실시간 분석 구조</div>	
<div>학습 데이터</div> <div>AI-Hub 건강관리를 위한 음식 이미지</div>	
<div>GPU</div> <div>NVIDIA RTX 4070</div>	

# 데이터셋: 건강관리를 위한 음식 이미지(AI-Hub)

YOLO 음식 인식 모델 학습을 위해 AI-Hub에서 제공하는 건강관리를 위한 음식 이미지 데이터셋을 사용하였습니다.

해당 데이터셋은 총 571개 음식 클래스에 대해 클래스당 약 60,000장 이상의 이미지가 존재하는 대규모 공공 데이터셋입니다.

그러나 개인 개발 환경의 GPU 메모리 및 저장 공간의 한계로 인해 각 클래스별로 이미지 300장을 무작위 랜덤 샘플링 방식으로 추출하여 학습 데이터로 활용하였습니다.

- 데이터 출처: AI-Hub 건강관리를 위한 음식 이미지 데이터셋
- 원본 데이터 규모: 571개 클래스 x 클래스당 약 60,000장
- 학습 데이터 구성: 각 클래스당 300장 랜덤 샘플링
- 총 학습 이미지 수: 약 17만 장
- 데이터 유형: 음식 객체 검출(Object Detection)
- 라벨 형식: YOLO Bounding Box Annotation





# 모델 학습 전략

효율적인 학습과 성능 향상을 위해

COCO Pretrained Weight를 기반으로 전이학습을 수행했습니다.

다양한 데이터 증강 기법을 적용하여 모델의 일반화 성능을 극대화했습니다.



## 전이학습

COCO Pretrained Weight 기반



## Epoch

30



## Batch Size

16



## 데이터 증강

Mosaic, HSV 변화, 회전 및 반전

```
model = YOLO('yolov8s.pt')

results = model.train(
    data=r"D:\데이터셋\YOLO\datasets\data.yaml",
    epochs=30,
    patience=5,
    batch=16,
    imgsz=640,
    device=device,
    workers=4,
    cache=False,

    project='Food_Detection_Project',
    name='train_result_300sample',
    exist_ok=True,

    hsv_h=0.015,
    hsv_s=0.7,
    hsv_v=0.4,
    degrees=10.0,
    fliplr=0.5,
    flipud=0.0,
    mosaic=1.0,

    verbose=True
)
```

# 모델 성능 평가

대규모 571개 클래스 환경에서도 실사용 가능한 수준의 높은 성능을 확보했습니다. 주요 지표들은 모델의 정확성과 효율성을 입증합니다.

0.648

Precision

정밀도

0.70

Recall

재현율

0.721

mAP@50

평균 정밀도

0.624

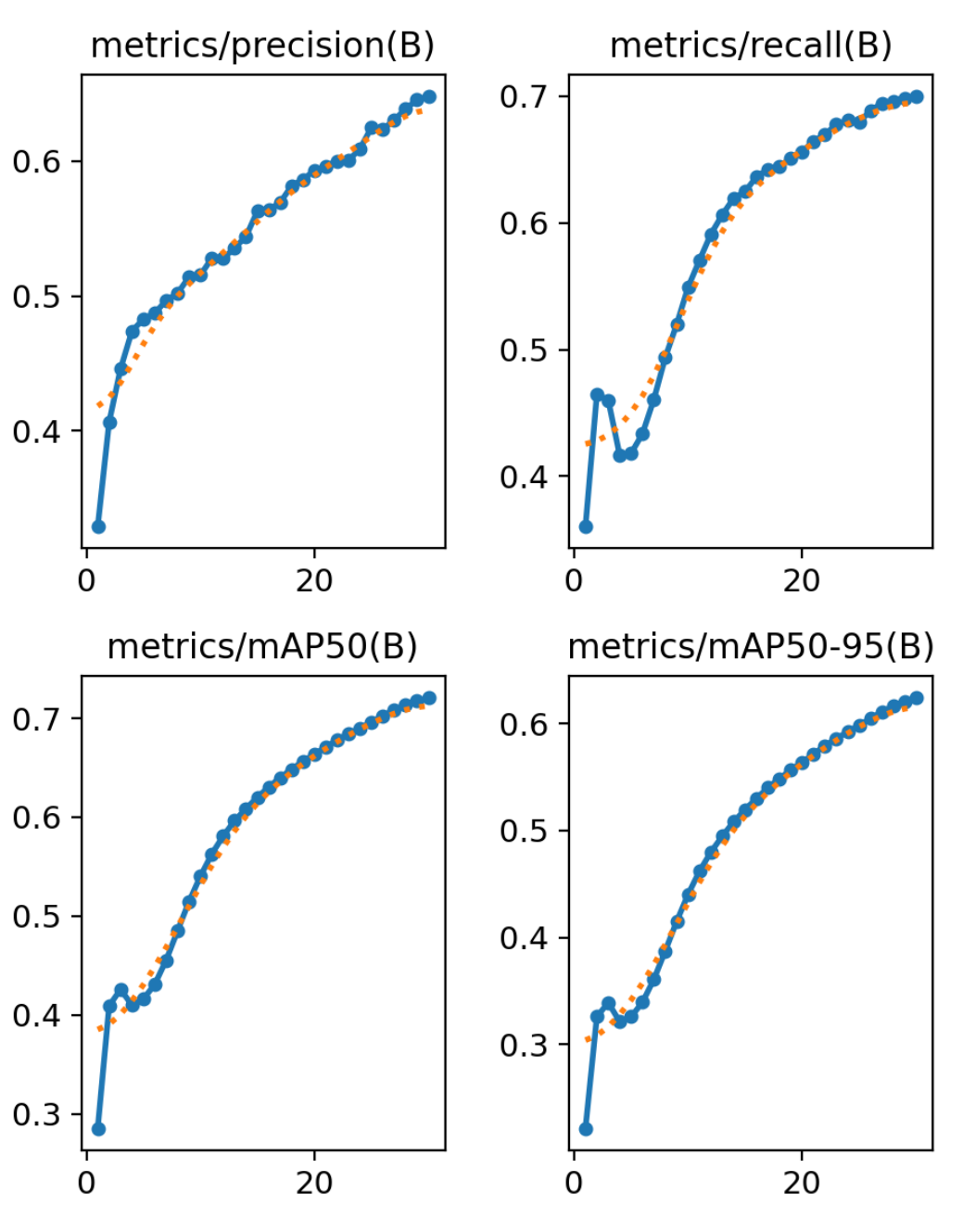
mAP@50-95

다중 IoU 임계값

0.66

F1-score

최적 균형 지표



# Gemini 하이브리드 보정 시스템

YOLO 모델의 초기 인식 결과를 Gemini 2.5 Flash LLM을 활용하여 보정함으로써, 시스템의 최종 판단 정확도를 극대화했습니다.



## YOLO 신뢰도 기반 호출

YOLO 인식 신뢰도가 낮을 경우 Gemini API를 자동으로 호출하여 보정 과정을 시작합니다.



## 시각적 유사 음식 보정

시각적으로 유사하여 YOLO가 오인식할 수 있는 음식들을 Gemini가 문맥 정보를 활용하여 정확하게 식별합니다.



## 성분표 문맥 재확인

성분표의 텍스트 정보를 LLM이 분석하여, 인식된 음식과 성분 간의 일관성을 검증하고 성분을 재확인합니다.



## 최종 정확도 향상

하이브리드 시스템을 통해 사용자에게 제공되는 알레르기 위험 판단의 최종 정확도를 크게 높였습니다.

# 주요 기능 ① 사용자 맞춤형 필터링

## 사용자 알레르기 정보 등록

개인의 알레르기 유발 항원 정보를 상세히 등록하여 맞춤형 서비스 기반을 마련합니다.

## 검색 시 DB 자동 비교

등록된 알레르기 정보와 음식 데이터베이스를 실시간으로 비교하여 위험 성분을 즉시 파악합니다.

## 위험 성분 시각적 경고

위험 성분 포함 여부를 명확하고 직관적인 시각적 경고로 표시하여 사용자에게 경각심을 줍니다.

## 교차 반응군 기반 잠재 위험도

알려진 교차 반응군 정보를 활용하여 잠재적인 알레르기 위험도를 예측하고 표시합니다.

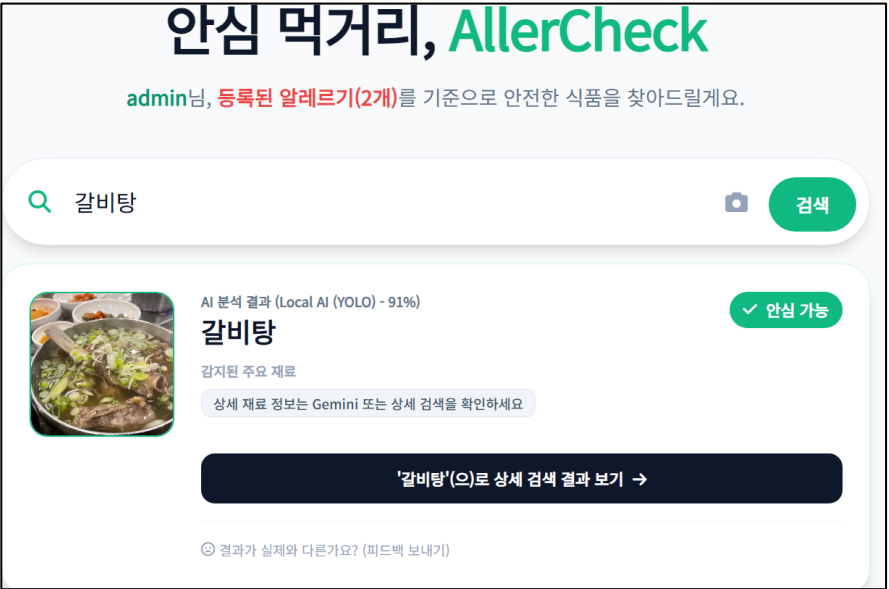
### 나의 알레르기 관리

가지고 계신 알레르기 항목을 모두 선택해주세요.  
선택된 정보는 검색 결과에서 **위험 경고**를 띄우는 데 사용됩니다.

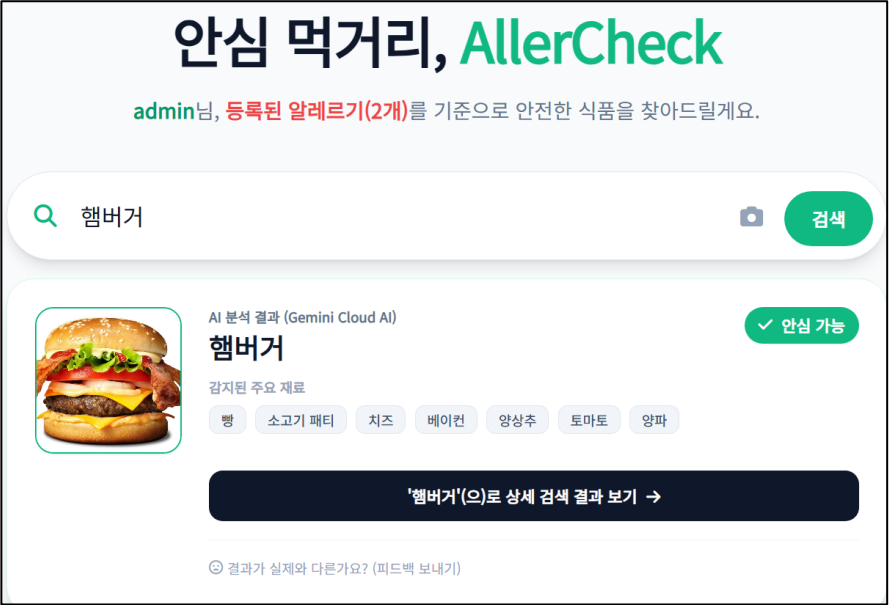
2개 선택됨

난류	가금류	계란	소고기
쇠고기	돼지고기	닭고기	새우
게	오징어	고등어	조개류
굴	전복	홍합	✓ 우유
✓ 땅콩	호두	잣	대두
복숭아	토마토	밀	메밀
이황산류			

# 주요 기능 ② 이미지 기반 검색



YOLO 기반 이미지 식별



Gemini 이미지 식별

텍스트 입력 없이도 검색 가능한 직관적인 서비스 구현을 통해 사용자 편의성을 극대화합니다.



## 음식 사진 업로드

사용자가 직접 촬영한 음식 사진을 업로드하여 검색을 시작합니다.



## YOLO 자동 인식

YOLO(You Only Look Once) 모델이 이미지 내의 음식을 자동으로 감지하고 식별합니다.



## 음식명 자동 검색

인식된 음식 정보를 기반으로 데이터베이스에서 관련 음식명을 자동으로 검색합니다.



## Gemini 보정 수행

음식 인식이 불확실할 경우, Gemini AI가 추가적인 분석 및 보정을 수행하여 정확도를 높입니다.

# 주요 기능 ③ 성분표 자동 입력 시스템 (관리자 기능)

1

## 성분표 이미지 업로드

관리자가 제품의 성분표 이미지를 시스템에 업로드합니다.

2

## Gemini 자동 분석

Gemini AI를 활용하여 성분표의 텍스트를 자동으로 추출하고 분석합니다.

3

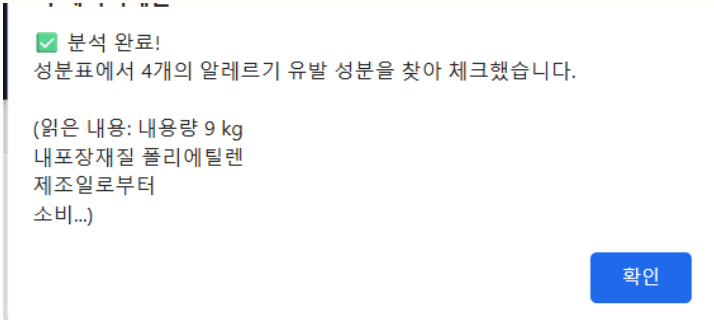
## 알레르기 성분 자동 체크

분석된 성분 중 알레르기 유발 성분을 자동으로 식별하고 체크합니다.

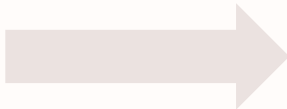
4

## DB 자동 저장

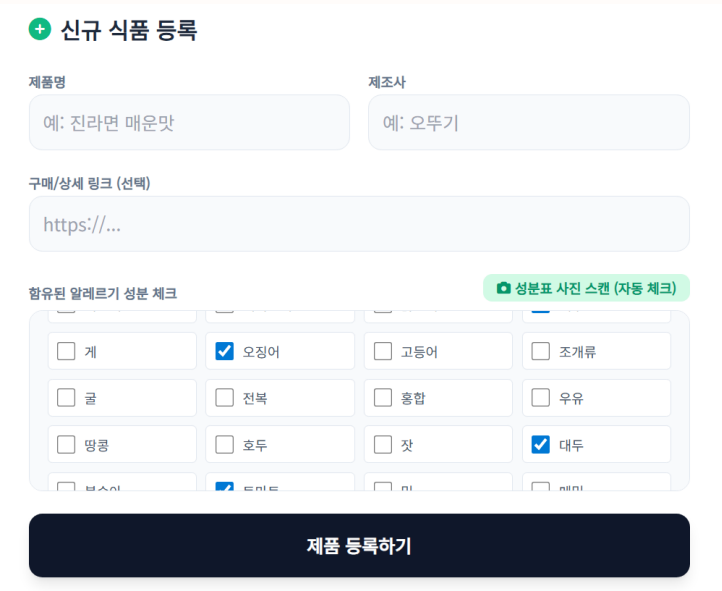
추출 및 분석된 성분 정보는 데이터베이스에 저장되어 서비스에 활용됩니다.



성분표 분석 결과



알레르기 성분  
자동 체크



# 주요 기능 ④ 관리자 및 통계 기능

## 상품 등록 및 삭제

관리자는  
새로운 음식 상품을 등록하거나  
기존 상품을 삭제할 수 있습니다.

## 알레르기 통계 TOP 5 제공

가장 많이 등록된 사용자 알레르기  
TOP 5 통계를 제공하여  
인사이트를 얻습니다.

## 전체 사용자 목록 조회

시스템에 등록된 전체 사용자 목록을 조회하고 관리할 수 있습니다.

A

AllerCheck Admin

관리자 모드로그아웃

총 회원 수

6명

현재 가입된 전체 사용자

가장 많이 등록된 알레르기 TOP 5

1위. 우유3명

2위. 땅콩2명

3위. 난류1명

4위. 새우1명

5위. 대두1명

+ 신규 식품 등록

제품명예: 진라면 매운맛

제조사예: 오뚜기

구매/상세 링크 (선택)  
https://...

함유된 알레르기 성분 체크

☐ 난류

☐ 가금류

☐ 계란

☐ 소고기

☐ 쇠고기

☐ 돼지고기

☐ 닭고기

☒ 새우

성분표 사진 스캔 (자동 체크)

가입 회원 목록

ID	이름	권한
1	admin	관리자
2	kim	일반
3	lee	일반
4	park	일반
5	kang	일반
6	test	일반

# 사용자 피드백 기반 데이터 수집



AI 분석 결과 (Gemini Cloud AI)

잡채

감지된 주요 재료

당면

각종 채소

버섯

✓ 안심 가능

'잡채'(으)로 상세 검색 결과 보기 →

☹ 결과가 실제와 다른가요? (피드백 보내기)

정확한 음식 이름을 알려주세요. AI 학습에 큰 도움이 됩니다!

잡채

제출

사용자 피드백을 통해 모델의 지속적인 성능 향상을 위한 학습 데이터를 구축합니다.

01

## 인식 오류 시 사용자 정답 입력

AI 인식에 오류가 발생할 경우, 사용자가 올바른 정보를 직접 입력합니다.

02

## 원본 이미지 + 정답 라벨 저장

사용자가 입력한 정답과 원본 이미지를 함께 저장하여 고품질 학습 데이터를 확보합니다.

03

## Hard Sample 데이터 구축

특히 어려운 사례(Hard Sample) 데이터를 집중적으로 수집하여 모델의 약점을 보완합니다.

04

## 향후 재학습 가능한 구조 확보

수집된 데이터를 활용하여 모델을 주기적으로 재학습할 수 있는 유연한 구조를 마련합니다.



# 주요 기술적 문제 해결 사례

## GPU OOM (Out Of Memory)

GPU 메모리 부족 문제를 해결하기 위해  
Batch Size 감소를 적용하여 학습 효율을 높였습니다.

## 학습 시간 과다

과도한 학습 시간을 단축하기 위해  
Epoch 조절을 통해 최적의 학습 반복 횟수를 찾았습니다.

## 오인식 문제

AI의 오인식 문제를 해결하고자  
Gemini 하이브리드 보정을 도입하여 정확도를 향상시켰습니다.

## RAM 부족

시스템 RAM 부족 현상에는  
DataLoader Worker 수 조절로 메모리 사용량을 최적화했습니다.

```
model = YOLO('yolov8s.pt')

results = model.train(
    data=r"D:\데이터셋\YOLO\datasets\data.yaml",
    epochs=30,
    patience=5,
    batch=16,
    imgsz=640,
    device=device,
    workers=4,
    cache=False,
```

# Docker 기반 배포 환경

일관된 개발 및 운영 환경을 통해 효율적이고 안정적인 서비스 제공을 목표로 합니다.

1

## CUDA 기반 GPU Docker 이미지 구축

딥러닝 모델 학습 및 추론을 위한 GPU 가속을 지원하는 Docker 이미지를 개발했습니다.

2

## FastAPI + YOLO + Gemini + MySQL 통합 컨테이너화

각 구성 요소를 독립적인 컨테이너로 통합하여 모듈성 및 관리 용이성을 확보했습니다.

3

## 개발 및 운영 환경 일관성 확보

Docker를 통해 개발 환경과 실제 운영 환경 간의 불일치를 최소화하여 안정성을 높였습니다.

```
FROM nvidia/cuda:12.1.0-cudnn8-runtime-ubuntu22.04 (last pushed 2 years ago)

WORKDIR /app

# 시스템 패키지 설치
RUN apt-get update && apt-get install -y \
    python3 \
    python3-pip \
    build-essential \
    libgl1 \
    libgl1-mesa-glx \
    && rm -rf /var/lib/apt/lists/*

# pip 최신화
RUN pip3 install --upgrade pip

# 파이썬 라이브러리 설치
COPY requirements.txt .
RUN pip3 install --no-cache-dir -r requirements.txt

# 프로젝트 전체 복사
COPY . .

# FastAPI 포트 오픈
EXPOSE 8000

CMD ["python3", "-m", "uvicorn", "api_main:app", "--host", "0.0.0.0", "--port", "8000"]
```

# Docker 배포 검증 결과

성공적인 Docker 배포를 통해 서비스의 안정성과 기능성을 확보했습니다.

- **컨테이너 재시작 후에도 프로필 이미지 유지**

컨테이너가 재시작되어도 사용자 프로필 이미지가 영구적으로 보존됨을 확인했습니다.

- **AI 결과 이미지 영구 저장**

AI 분석 결과로 생성된 이미지들이 안정적으로 저장되고 유지됩니다.

- **외부 포트 8000 정상 접근**

서비스 접속을 위한 외부 포트 8000번이 정상적으로 작동함을 검증했습니다.

- **모든 API 정상 동작 확인**

구현된 모든 API 엔드포인트가 예상대로 동작하며 데이터 통신에 문제가 없음을 확인했습니다.

# 최종 구현 성과 요약

1

## 이미지 기반 알레르기 위험 판단 서비스 구현

음식 이미지를 통해 알레르기 위험을 직관적으로 판단하는 서비스를 성공적으로 구현했습니다.

2

## YOLO + LLM 하이브리드 AI 구조 완성

YOLO와 대규모 언어 모델(LLM)을 결합한 하이브리드 AI 아키텍처를 구축했습니다.

3

## DB-웹-AI 완전 통합

데이터베이스, 웹 애플리케이션, AI 모델 간의 완벽한 통합을 이루어냈습니다.

4

## Docker 기반 실사용 가능한 배포 환경 구축 성공

Docker를 활용하여 실제 환경에서 즉시 사용 가능한 안정적인 배포 시스템을 완성했습니다.

# 프로젝트 완성도 및 차별성

## 단순 모델 실험을 넘어선 실제 웹 서비스 구현

단순한 모델 테스트가 아닌, 사용자에게 가치를 제공하는 실질적인 웹 서비스를 개발했습니다.

## 사용자-관리자-AI-DB 연계 구조 완성

각기 다른 주체와 시스템 간의 유기적인 연동을 통해 완전한 생태계를 구축했습니다.

## 학습, 예측, 피드백, 재학습까지 고려한 실무형 프로젝트

모델의 생애주기 전체를 고려하여 지속적인 개선이 가능한 실용적인 시스템을 설계했습니다.

# 프로젝트의 한계점과 도전 과제

## 학습 데이터 수의 한계

모델 학습에 사용된 데이터의 양이 제한적이었으며, 이는 모델의 일반화 성능에 영향을 미쳤습니다.  
더 다양한 시나리오에 대한 학습이 필요합니다.

## 단일 GPU 기반 학습 환경

단일 GPU 환경에서는 대규모 데이터셋과 복잡한 모델을 학습하는 데 시간적 제약이 발생했습니다.  
분산 학습 환경의 필요성을 느꼈습니다.

## OCR 일부 정확도 문제

특정 이미지나 텍스트 형식에서 OCR 정확도가 떨어지는 문제가 발견되었습니다.  
이는 후처리 과정을 통해 보완해야 할 부분입니다.

## 관리자 기능 확장 필요

현재 관리자 기능은 제한적이며, 사용자 관리, 데이터 통계, 모델 업데이트 등 더 다양한 기능이 요구됩니다.

# 향후 개선 계획 및 발전 방향



## Hard Sample 기반 추가 재학습

모델이 어려워하는 특정 데이터(Hard Sample)를 선별하여 추가 재학습을 진행, 모델의 성능과 견고성을 향상시킬 것입니다.



## 전용 OCR 모델 도입

범용 OCR 대신 특정 도메인에 최적화된 전용 OCR 모델을 도입하여, 텍스트 인식 정확도를 획기적으로 높일 계획입니다.



## 클라우드 GPU 기반 확장 학습

AWS, GCP와 같은 클라우드 기반 GPU 자원을 활용하여 대규모 데이터와 복잡한 모델을 효율적으로 학습시킬 것입니다.



## 외부 사용자 접속 가능한 서버 배포

개발된 시스템을 실제 서비스 환경에 배포하여 외부 사용자들이 직접 접근하고 활용할 수 있도록 할 예정입니다.

# 결론: End-to-End AI 플랫폼의 가치

본 프로젝트는 딥러닝 모델, LLM, 웹 서비스, 데이터베이스, Docker 배포까지 통합된 실전형 AI 시스템입니다.

모델 구현을 넘어 실제 서비스로 동작하는 End-to-End AI 플랫폼을 완성했다는 점에서 기술적·실무적 가치가 매우 큼니다.

이러한 통합적인 접근 방식은 단순한 연구를 넘어 실제 문제 해결에 기여하는 AI 솔루션 개발의 가능성을 제시합니다.



# 오픈소스 공개 및 재현성 확보



## 전체 소스코드 GitHub 공개

모든 개발된 소스코드는 GitHub에 공개되어 투명성을 확보하고 커뮤니티의 기여를 유도합니다.



## 실행 매뉴얼 포함

누구나 쉽게 프로젝트를 설정하고 실행할 수 있도록 상세한 실행 매뉴얼을 제공합니다.



## Docker 기반 동일 환경 재현 가능

Docker를 활용하여 개발 환경과 동일한 실행 환경을 구축, 손쉬운 재현을 보장합니다.



## 프로젝트 지속 관리 예정

공개된 프로젝트는 지속적으로 관리하고 업데이트하여 안정성과 확장성을 유지할 것입니다.



## 질의응답

# Q & A

발표를 경청해주셔서 감사합니다.

질문 있으시면 편하게 말씀해주시면 감사하겠습니다.