

# NO GUTS FRIDGEAT

## 모두의 셰프

## 1. 프로젝트 개요

### 주제 선정 배경

장기간 냉장고에 보관된 후 폐기되는 식재료의 활용도 개선과 요리의 장벽을 낮춰 누구나 원할 때 쉽게 요리를 할 수 있도록 하기위해 기획됨

### 진행 목적

요리에 필요한 개인의 노력을 대신할 수 있는 **AI App**을 만들어, 식재료 활용에 도움을 주고, 요리의 장벽을 낮춰, 사람들이 보다 쉽게 요리를 시작할 수 있도록 하여 새로운 **뉴노멀**을 만들고자 한다.

### 개발 환경

- Flutter
- Firebase
- Tensorflow
- Git
- github

### 기대 효과

- 식재료 효율화 증가 → 음식물 폐기량 감소 → 쓰레기 처리 공간문제 감소 → 쓰레기 처리비용 감소 → 탄소 발자국 감소
- 개인 요리 빈도 증가 → 간편음식 & 배달음식 이용 감소 → 식비 감소 & 일회용품 사용량 감소

## 2. 프로젝트 수행 절차 및 방법

Task	계획 기간	활동
데이터 선정	10월 23일 ~ 24일	식재료 데이터 선정
데이터 수집	10월 24일 ~ 25일	Object Detection을 위한 식재료 데이터 수집
Firebase 서버 구축 및 Flutter 연동	10월 23일	YOLO, GPT4에서 얻은 결과를 저장하기 위한 서버 구축 및 Flutter 연동
App UI 구성	10월 23일 ~ 30일	Flutter UI구성
GPT API 관련 코딩	10월 23일 ~ 30일	GPT API 연동 및 프롬프트 엔지니어링
데이터 전처리	10월 25일 ~ 27일	데이터 전처리
모델 구현	10월 27일 ~ 30일	YOLO 모델구현
모델 학습	10월 30일 ~ 11월 24일	식재료 식별을 위한 YOLO 모델 학습
모델 테스트	11월 2일 ~ 11월 24일	YOLO 버전 별 정확도 테스트
I차 MVP	11월 10일	Flutter I차 패키징 및 테스트
최적화	11월 10일 ~	코드 개선 & 기능개선
총 개발 기간	5주	

## 3. 프로젝트 구조



## 4. 프로젝트 결과

### YOLO v8n

- 평균 mAP는 0.7, 가장 낮은 레이블 mAP는 0.26으로 평균 20%정도의 성능 하락.
- 실제 로컬에 탑재시 기존 모델보다 퍼포먼스 향상됨

아이펠톤 기간 내로 학습된 52개의 식재료 클래스에 한해, 인식률은 실제 사용자가 사용이 가능할 정도로 충분할 것으로 판단된다. 또, 사용자가 선택한 정보에 기반한 레시피 제공 단계까지의 주요 기능 구현이 완료됐지만, 상용화를 위해서는 백엔드 서비스 플랫폼 Firebase를 활용한 앱의 고도화, 최적화, 수익화가 필요하다.

