

# 요구사항 분석서

AI 기반 영양성분 제공 서비스

2025.04.06

<식전식 KU>

박민규 202011292

한승현 202011388

지도교수: 박소영 교수님

# 목차

## 1. 개요

- 1.1 프로젝트 기획 배경
- 1.2 기술 동향 및 핵심 기술
- 1.3 프로젝트 주요 기능 및 특징
- 1.4 조원 구성 및 역할 분담
- 1.5 일정

## 2. 기능적 요구사항

- 2.1 Use Case Diagram
- 2.2 Use Case Document

## 3. 비기능적 요구사항

- 3.1 사용편리성
- 3.2 신뢰성
- 3.3 성능
- 3.4 이식성
- 3.5 유지관리
- 3.6 구현상 제약사항
- 3.7 인터페이스
- 3.8 법적 제약사항

# 1. 개요

## 1.1 프로젝트 기획 배경

건강관리와 식습관에 대한 관심은 여전히 높고, 사람들은 건강한 음식을 위해 기꺼이 계획하며 소비합니다. 그러나 영양성분 정보를 얻는 일은 때때로 부정확하며 번거롭습니다. 본 서비스는 AI 기반 영양성분 정보 제공 서비스이며, 음식 사진만으로 음식의 양과 종류를 분석한 후 영양성분 정보를 제공합니다. 음식의 칼로리, 탄수화물, 단백질, 지방 등의 영양성분을 알려줌으로써 사용자의 건강관리를 효율적으로 돕습니다.

이 기능은 크게 두 가지로 나뉩니다. 사진이나 텍스트 입력으로 상용 제품의 영양성분을 제공하는 것, 사진에 담긴 음식을 분석해 영양성분 정보로 제공하는 것. 이는 영양성분 정보를 확인해야 하거나, 식단 관리가 중요한 사람들에게 유용합니다. 사용자가 직접 타이핑해 식품명을 검색하는 기존의 방식을 넘어 사진촬영을 이용한 방식은 보다 편리하고 신속합니다.

Open API 를 이용해 영양성분 정보를 취득하고, 이미지 분석을 위해 음식 데이터셋을 학습한 머신러닝 모델을 이용합니다. 높은 정확도를 위해 다양하고 많은 음식 이미지 데이터를 학습시킵니다. 이를 통해 생성한 정보는 가공해 사용자에게 제공합니다.

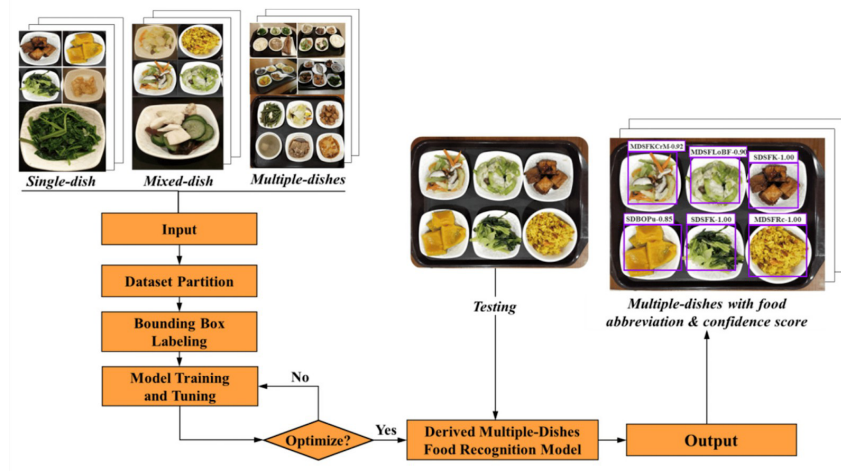
앞서 언급한 것처럼, 가공식품 등의 상품명을 직접 입력 받는 기능과 더불어 촬영한 음식 사진으로 영양성분을 제공하는 기능을 탑재할 것입니다. 뿐만 아니라 각 유저의 식단을 분석하여, 개인에게 적합한 식단을 제안해주는 등의 기능을 갖춘 애플리케이션을 개발하고자 합니다.

AI 기반 식품 분석 시스템은 음식의 영양성분을 빠르고 효율적으로 제공함으로써 사용자의 건강관리에 기여합니다. 또한 섭취한 음식을 기록하거나, 데이터 기반의 맞춤형 식단을 추천할 수도 있습니다. 이는 바쁜 현대인들의 건강을 실용적으로 돕기 위한 방안이 될 것입니다.

## 1.2 기술 동향 및 핵심 기술

### 1.2.1 딥러닝 기반 음식 인식 및 분석 시스템

이미지에서 음식을 인식하는 과정은 입력으로 들어온 이미지에서 음식을 검출하는 객체 인식 단계와, 검출된 음식을 분석하는 단계로 구분됩니다. 이때 문자를 함께 인식함으로써 라벨링 된 가공식품도 함께 검출합니다.



Proposed multiple-dish food recognition model workflow

(Deep-Learning-Assisted Multi-Dish Food Recognition Application for Dietary Intake Reporting)

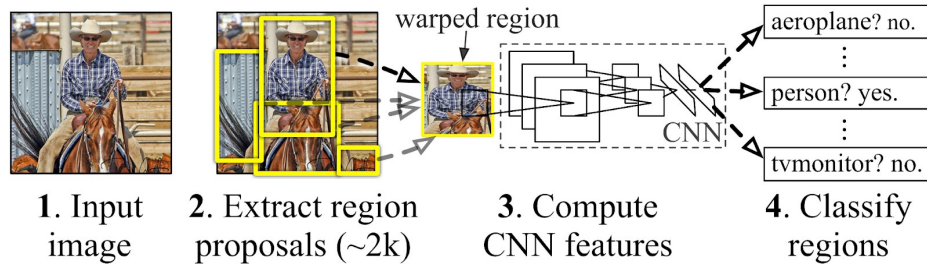
#### 1.2.1.1 Object Detection

이미지를 분석하여 특정 객체를 인식하고 그것의 위치를 지정하는 기술입니다. 이 과정에서 이미지 속에 있는 여러 개의 객체를 탐지하고, 각 객체의 종류와 위치정보(bounding box)를 제공합니다.

객체를 탐지하고 분류하기 위해선 딥러닝 모델을 활용합니다. 예를 들어, 음식 사진에서 탐지된 객체는 “피자”, “샐러드”, “햄버거” 등의 카테고리로 분류할 수 있습니다. 이후 객체의 위치를 좌표로 반환해 Bounding Box 를 지정하고 사각형으로 나타냅니다.

객체 탐지에는 대표적으로 R-CNN(Region-based Convolutional Neural Networks), Fast R-CNN, Faster R-CNN, YOLO, SSD 가 사용됩니다. 객체를 탐지하는 성능과 비용, 속도 등을 고려해야 합니다.

## R-CNN: Regions with CNN features

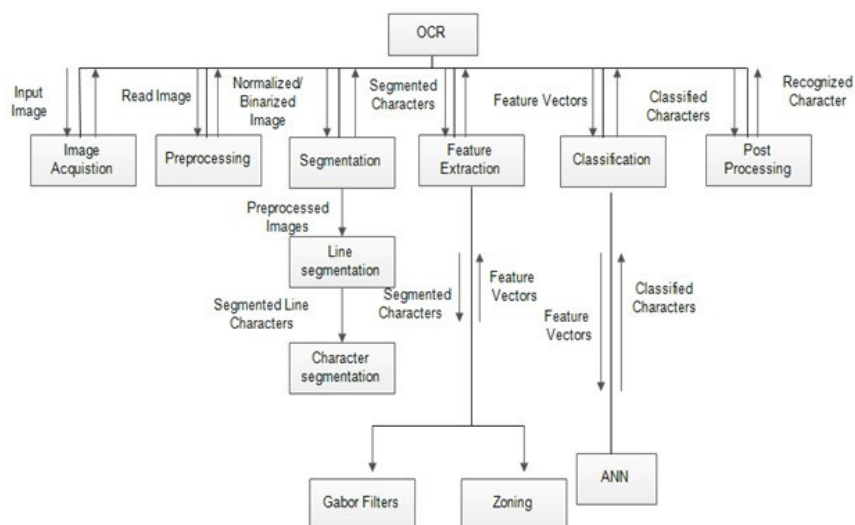


R-CNN: Regions with CNN Features(Rich feature hierarchies for accurate object detection and semantic segmentation)

모델을 학습시키기 위해 다양한 객체가 포함된 대규모 이미지 데이터셋이 필요합니다. 또한 각 이미지에 대한 객체의 클래스 정보와 Bounding Box 좌표를 포함한 라벨링 작업이 요구됩니다. CNN 을 이용해 이미지의 Feature Map(특징 맵)을 추출하고, 이를 바탕으로 객체를 분류하는 과정을 학습합니다. 이후 Bounding Box Regression으로 객체의 위치를 예측하는 방법을 학습합니다.

### 1.2.1.2 OCR

이미지를 분석해 그 안에 포함된 문자를 인식하고 디지털 데이터로 변환하는 기술입니다. 가공식품의 패키징 글자를 인식해 제품명을 파악하고, 사용자에게 해당 제품의 영양성분 정보를 제공합니다. 이는 사용자가 제품의 정보를 직접 입력할 필요 없이 자동으로 영양성분 정보를 얻을 수 있게 합니다.



Structure Chart of OCR System(Recognition of Historical Records Using Gabor and Zonal Features)

OCR은 이미지 처리와 문자 인식의 결합입니다. 이미지 전처리(Image Preprocessing)로 이미지 속 문자를 보다 명확하게 나타낸 후, 각 문자의 패턴을 인식하고 이를 텍스트 데이터로 변환합니다.

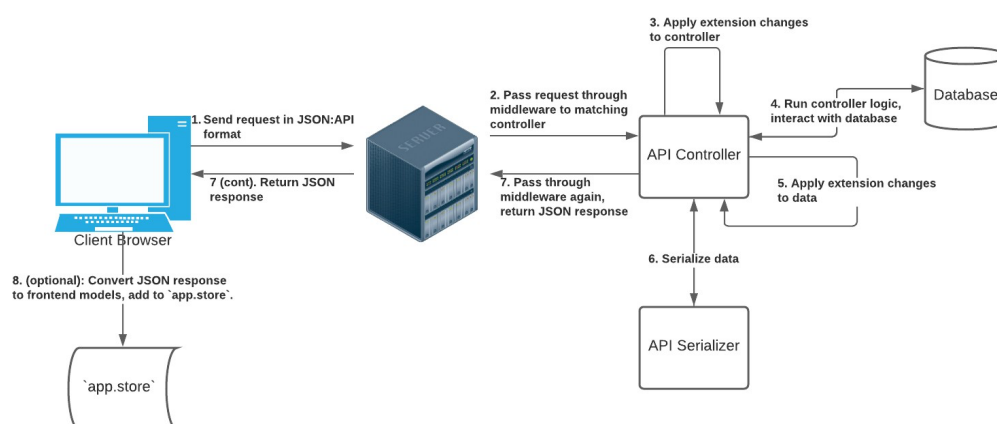
주요 이미지 전처리 기술로는 이진화(Binarization), 노이즈 제거(Noise Removal), 경계 추출(Edge Detection) 등이 있습니다. 이미지 속 노이즈를 제거하거나 문자와 배경을 구분 짓기 위한 경계를 추출합니다.

글자를 개별적으로 분할하는 Segmentation을 거친 후 실제 텍스트로 변환합니다. 이 과정에서 딥러닝 기반 모델을 사용하면 이미 학습된 패턴과 비교함으로써 다양한 글꼴 등이 사용된 복잡한 텍스트도 보다 잘 인식합니다. 또한 인식된 텍스트에 자연어 처리(NLP) 기법을 적용해 문법적 오류를 교정해 글자의 정확도를 높일 수 있습니다.

CNN(Convolutional Neural Networks)과 RNN(Recurrent Neural Networks)을 결합한 방식이 많이 사용되며, 이는 문자의 패턴 뿐 아니라 문맥적인 연관성을 파악하는 데도 도움을 줍니다. CRNN (Convolutional Recurrent Neural Network), CTC (Connectionist Temporal Classification) 등으로 문자열의 연관성을 분석하고 더 나아가 예측하게 합니다.

## 1.2.2 Open API 를 이용한 데이터 처리 시스템

공개된 데이터베이스에서 제공하는 API 를 통해 다양한 데이터를 가져와 분석하고 활용합니다.



API Request Lifecycle(<https://docs.flarum.org/extend/api/#api-request-lifecycle>)

기관에서 Open API 형식으로 공개한 공공 데이터를 기반으로 머신러닝이나 딥러닝과 결합해 예측 모델을 구축합니다. 대용량 데이터를 효율적으로 가공하고 처리하는 데 중점을 두고 있으며, 이를 통해 사용자에게 영양성분 정보를 제공합니다.

Open API 는 외부 개발자가 접근하고 사용할 수 있는 공개된 프로그래밍 인터페이스로, 특정 시스템의 기능이나 데이터를 외부에 제공하는 방식입니다. 이를 통해 개발자는 다른 서비스의 데이터를 가져와 가공할 수 있습니다. 자체적으로 데이터를 구축하지 않아도 외부에서 제공되는 공공 데이터나 기업의 데이터를 사용할 수 있어 데이터 수집의 로드 비용이 절감됩니다. Open API 를 사용하면 대용량의 데이터를 손쉽게 받아오고, 이를 기반으로 특정 기능을 구현할 수 있습니다. RESTful API 형식이 가장 일반적이며, JSON 이나 XML 과 같은 형식으로 데이터를 주고받습니다.

데이터를 요청할 때 HTTP 프로토콜을 사용하여 RESTful 방식으로 API 에 요청을 보냅니다. 이를 위해 HTTP 클라이언트(Axios, Fetch API)를 사용해 요청을 처리합니다. 응답 받은 데이터를 처리하기 위해 JSON 파싱이나 XML 파싱이 필요하며, 이 데이터를 가공하여 활용할 수 있는 포맷으로 변환해야 합니다.

## 1.3 프로젝트 주요 기능 및 특징

### 1.3.1 가공식품 영양성분 분석 기능

가공식품 영양성분 분석 기능이란, 사용자에게 가공식품의 영양성분을 정확하게 제공하는 데 중점을 두고 있습니다. 예를 들어, 초코파이와 같은 가공식품은 영양성분이 일관되고 명확하게 정의되어 있습니다.

Open API 를 활용하여 초코파이 1 봉지에 포함된 영양성분(예: 탄수화물 17g, 단백질 2g, 지방 8g 등)을 신속하고 정확하게 제시할 수 있습니다. 사용자는 제품의 이름을 검색함으로써 손쉽게 영양정보를 확인할 수 있으며, 이는 사용자에게 올바른 식단 결정을 내리는 데 도움을 줍니다.

제품 이름을 직접 입력해 검색하는 것이 번거롭다면 제품에 명시된 식품성분표인 영양성분표를 촬영해 제품을 검색할 수 있습니다. 영양성분표에서 제품명과 품목보고번호를 탐지해 제품의 영양성분 정보를 제공합니다.

### 1.3.2 일반음식 영양성분 분석 기능

일반음식 영양성분 분석 기능이란, 사용자가 직접 요리한 음식의 영양성분 분석을 위해, 사진 촬영 기능을 도입합니다. 사용자가 요리한 파스타를 예로 들면, 이 기능은 딥러닝 기반의 이미지 인식 기술인 CNN(Convolutional Neural Network)을 사용하여 사용자가 촬영한 사진에서 음식의 종류를 식별합니다. 모델은 다양한 파스타 종류를 학습하여, 면의 칼로리(예: 100g 당 150kcal)와 추가 재료(예: 바지락 50g, 20kcal)의 영양 정보를 자동으로 추출합니다. 사용자가 촬영한 이미지에서 특정 음식이 인식되면, 그 음식의 전반적인 영양성분을 사용자에게 제시하고, 제공된 정보는 지속적으로 업데이트되어 더욱 정확한 결과를 보장합니다. 이로 인해 사용자는 자신이 만든 요리의 영양성을 대략적으로 이해하고, 건강한 선택을 할 수 있게 됩니다.

### 1.3.3 사용자 식단 분석 및 관리 기능

사용자 식단 분석 및 관리 기능은 사용자가 입력한 음식의 영양 정보를 분석하여, 개인의 목표에 따라 맞춤형 식단 관리를 지원합니다. 예를 들어, 사용자가 다이어트를 목표로 설정하고 일일 칼로리 섭취량을 1500kcal로 정했다고 가정합니다. 사용자가 섭취한 음식의 총 칼로리가 1600kcal에 달할 경우, 시스템은 사용자가 섭취한 칼로리를 바탕으로 “목표한 칼로리를 100kcal 초과하였습니다”라는 알림을 보냅니다. 또한 특정 영양소가 부족하다면, “단백질 섭취가 부족합니다. 20g 더 섭취하세요”와 같이 구체적인 안내를 보내 사용자가 보다 건강한 선택을 할 수 있도록 유도합니다. 이를 통해 사용자는 자신의 영양 목표를 보다 효과적으로 관리할 수 있습니다.



## 1.4 조원 구성 및 역할 분담

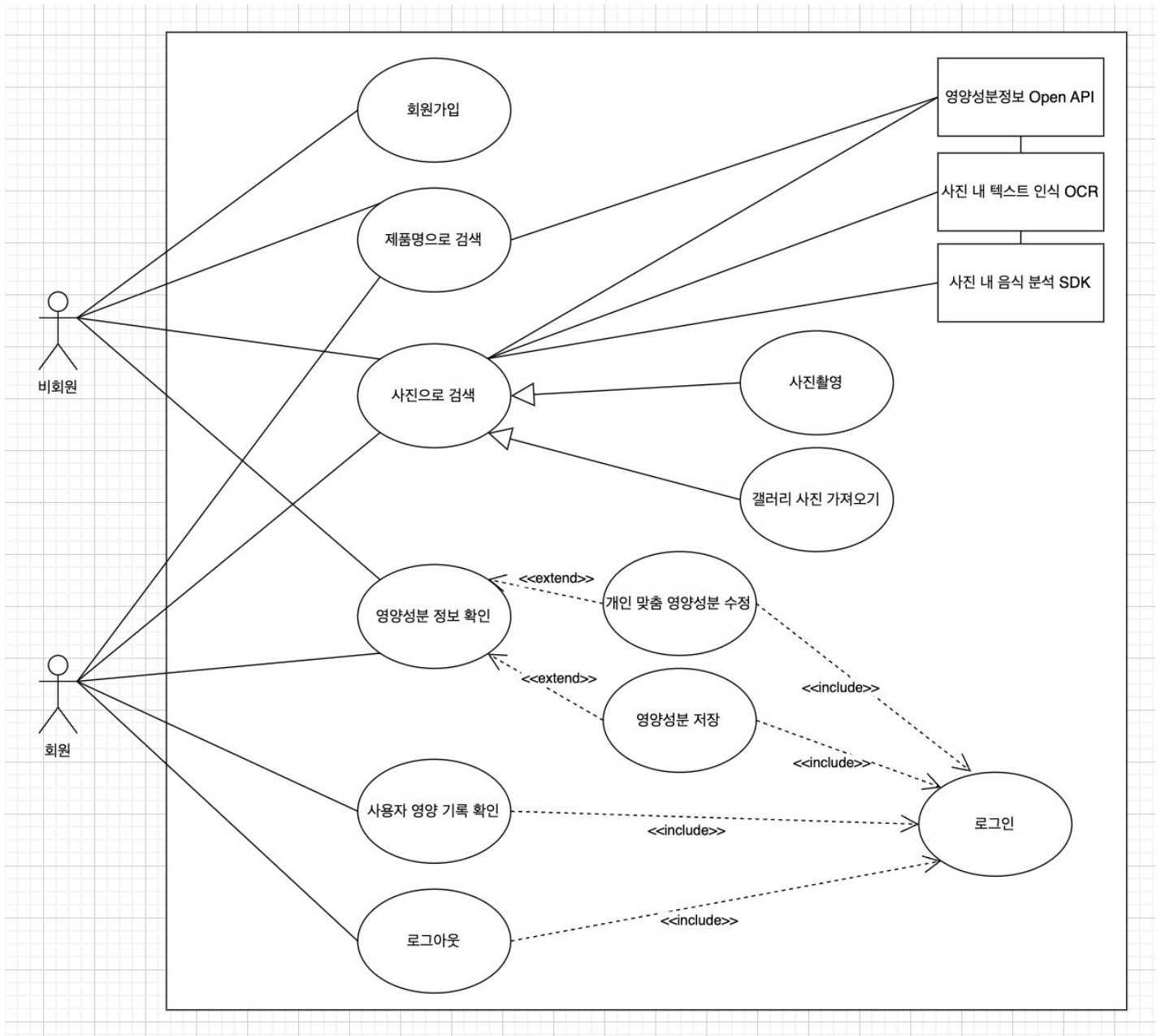
한승현	데이터베이스 구조 설계 회원가입, 로그인 사진촬영 후 영양성분 분석 기술 OCR 기반 제품 분석 제품명을 기반으로 한 영양성분 제공
박민규	스케치를 통한 애플리케이션 UI/UX 설계 피그마를 통한 애플리케이션 UI/UX 디자인 애플리케이션 내의 스크린(UI) 구현 전반적인 애플리케이션 기능 구현 사용자의 음식 섭취 기록 기능

## 1.5 일정

10 월	기능 구체화 시스템 설계
11 월	시스템 설계 구체화 & 확정 기본 구현 테스트
12 월	프로토타입 설계 및 구현 기말 보고서 작성 및 시연 준비
1 월	Open API 를 통한 영양성분 제공 이미지 분석을 통한 음식 정보 탐색 기능 구현
2 월	이미지 분석을 통한 음식 정보 탐색 기능 구현 섭취 기록 저장/열람 기능 구현 OCR 을 이용한 제품 분석
3 월	OCR 을 이용한 제품 분석 이미지를 통한 음식 탐색 기능의 정확도 측정
4 월	이미지 분석 결과에 대한 수정 기능 작동 확인 및 보완, UI/UX 보완
5 월	전반적인 디자인 완성 전체 프로그램 테스트 및 디버깅
6 월	전체 프로그램 테스트 및 디버깅

## 2. 기능적 요구사항

### 2.1 Use Case Diagram



## 2.2 Use Case Document

<b>Use Case</b>	1. 제품명으로 검색
<b>Actors</b>	비회원, 회원
<b>Overview</b>	사용자가 원하는 음식/제품명을 입력해 검색한다.
<b>Pre-requisites</b>	사용자는 음식/제품명을 알고 있어야 한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (A): 원하는 음식/제품명을 입력한다. 2. (S): 영양성분 정보 API 에서 제품명에 해당하는 정보를 가져온다.
<b>Alternative Courses of Events</b>	사진으로 검색
<b>Exceptional Courses of Events</b>	E1. 사용자가 입력하지 않으면 검색을 진행하지 않는다. E2. 입력값에 해당하는 데이터가 없다면 영양성분 정보를 제공하지 않는다.

<b>Use Case</b>	2. 사진으로 검색
<b>Actors</b>	비회원, 회원
<b>Overview</b>	사용자가 사진을 통해 검색한다.
<b>Pre-requisites</b>	사용자는 시스템에 사진을 제공한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (A): 사진을 통해 원하는 음식/제품명을 검색하길 희망한다. 2. (S): 사용자에게 이미지 제공 방식에 대한 선택지를 제공한다. 3. (A): 원하는 방식으로 음식/제품의 이미지 정보를 제공한다. 4. (S): 음식 분석 SDK 와 텍스트 인식 OCR 로 이미지를 분석한다. 5. (S): 음식 분석 SDK 와 텍스트 인식 OCR 로부터 정보를 얻는다. 5-1. (S): 인식된 텍스트가 있다면 영양성분 정보 API 에서 정보를 가져온다.
<b>Alternative Courses of Events</b>	제품명으로 검색
<b>Exceptional Courses of Events</b>	E1. 사용자가 사진을 제공하지 않으면 검색을 진행하지 않는다. E2. 정보에 응하는 데이터가 없다면 영양성분 정보를 제공하지 않는다.

<b>Use Case</b>	2-1. 사진촬영
<b>Actors</b>	비회원, 회원
<b>Overview</b>	사용자는 카메라를 이용해 식품을 촬영한다.
<b>Pre-requisites</b>	사용자는 사진으로 검색을 선택한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (S): 사용자가 카메라에 접근할 수 있도록 한다. 2. (A): 원하는 음식/제품을 촬영해 이미지 정보를 제공한다.
<b>Alternative Courses of Events</b>	갤러리 사진 가져오기
<b>Exceptional Courses of Events</b>	N/A

<b>Use Case</b>	2-2. 갤러리 사진 가져오기
<b>Actors</b>	비회원, 회원
<b>Overview</b>	사용자는 갤러리에서 사진을 선택한다.
<b>Pre-requisites</b>	사용자는 사진으로 검색을 선택한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (S): 사용자가 갤러리에 접근할 수 있도록 한다. 2. (A): 원하는 음식/제품을 갤러리에서 선택해 이미지 정보를 제공한다.
<b>Alternative Courses of Events</b>	사진촬영
<b>Exceptional Courses of Events</b>	N/A

<b>Use Case</b>	3. 영양성분 정보 확인 불가
<b>Actors</b>	비회원, 회원
<b>Overview</b>	사용자에게 영양성분 정보를 제공할 수 없음을 알린다.
<b>Pre-requisites</b>	사용자가 부정확한 음식/제품명을 입력했거나, 사용자가 제공한 사진에서 적합한 음식/제품을 탐색하지 못한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (A): 텍스트 입력이나 사진 제공을 통해 음식/제품명 정보를 제공한다. 2. (S): 영양성분 정보 제공 관련 SDK 와 API 에서 정보를 탐색하지 못한다. 3. (S): 사용자에게 정보 탐색에 실패했음을 알린다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

<b>Use Case</b>	4. 영양성분 정보 확인
<b>Actors</b>	비회원, 회원
<b>Overview</b>	사용자가 영양성분 정보를 확인한다.
<b>Pre-requisites</b>	사용자는 음식/제품을 검색한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (S): 사용자가 제공한 정보를 토대로 탐색에 성공한다. 2. (S): 사용자에게 영양성분 정보를 제공한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

<b>Use Case</b>	4-1. 개인 맞춤 영양성분 정보 수정
<b>Actors</b>	회원
<b>Overview</b>	사용자가 영양성분 정보를 수정한다.
<b>Pre-requisites</b>	사용자는 영양성분 정보를 확인한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (S): 사용자에게 영양성분 정보를 제공한다. 2. (S): 영양성분 정보를 수정할 수 있는 선택지를 함께 제공한다. 3. (A): 개개인의 식재료, 섭취량에 맞게 영양성분 정보를 수정한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

<b>Use Case</b>	4-2. 영양성분 정보 저장
<b>Actors</b>	회원
<b>Overview</b>	사용자가 영양성분 정보를 저장한다.
<b>Pre-requisites</b>	사용자는 영양성분 정보를 확인한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (S): 사용자에게 영양성분 정보를 제공한다. 1-1. (A): 개개인의 식재료, 섭취량에 맞게 영양성분 정보를 수정한다. 2. (A): 원하는 영양성분 정보를 저장한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A

<b>Use Case</b>	5. 사용자 영양 기록 확인
<b>Actors</b>	회원
<b>Overview</b>	사용자는 자신이 저장해둔 영양성분 정보 기록을 확인한다.
<b>Pre-requisites</b>	사용자는 영양성분 정보를 저장한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (S): 사용자가 저장해둔 영양성분 정보를 제공한다. 2. (A): 자신의 영양성분 정보 기록을 확인한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	E1. 저장된 영양성분 정보 기록이 없다면 해당 사실을 사용자에게 알린다.

<b>Use Case</b>	6. 회원가입
<b>Actors</b>	비회원
<b>Overview</b>	사용자는 더 많은 기능을 사용하기 위해 회원가입을 진행한다.
<b>Pre-requisites</b>	사용자는 회원등록이 되지 않은 상태여야 한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (A): 회원가입을 위한 사용자 정보를 입력한다. 2. (S): 회원정보를 등록하고 가입 성공 여부를 알린다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	E1. 이미 등록된 회원은 회원가입을 진행할 수 없다. E2. 모든 입력 조건을 만족시키지 못한 사용자는 회원가입 할 수 없다.

<b>Use Case</b>	7. 로그인
<b>Actors</b>	회원
<b>Overview</b>	사용자는 더 많은 기능을 사용하기 위해 로그인을 진행한다.
<b>Pre-requisites</b>	사용자는 회원가입을 마친 상태여야 한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (A): 로그인을 위한 사용자 정보를 입력한다. 2. (S): 등록된 회원정보와 일치하는지 확인한다. 3. (A): 로그인에 성공한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	E1. 회원정보 불일치시 로그인 할 수 없다.

<b>Use Case</b>	8. 로그아웃
<b>Actors</b>	회원
<b>Overview</b>	시스템에서 로그아웃 한다.
<b>Pre-requisites</b>	사용자는 로그인을 마친 상태여야 한다.
<b>Typical Courses of Events</b>	(A): Actor, (S): System 1. (A): 시스템 접근 해제를 위해 로그아웃한다. 2. (S): 시스템 접근을 해제한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	N/A



## 3. 비기능적 요구사항

### 3.1 사용 편리성

본 서비스의 주요 기능은 AI를 이용한 사진 분석 및 영양성분 정보 제공입니다. 사용자가 카메라로 사진을 찍으면 즉시 음식 종류를 인식하고 영양성분 정보를 제공하는 직관적인 인터페이스를 가집니다. 애플리케이션을 구성하는 기능과 아이콘을 최대한 간결하고 직관적으로 구현합니다.

가공식품의 라벨을 스캔해 해당 제품의 영양양성분 정보를 제공합니다. 사용자는 직접 제품명을 검색하기보다 사진을 이용해 더욱 빠르고 간편하게 정보를 확인합니다. 한 장의 사진으로 라벨이 없는 음식과 가공식품에 대한 영양성분 정보를 동시에 얻을 수 있습니다.

음식 분석 결과와 영양성분 정보를 일관된 포맷으로 제공합니다. 탄수화물, 단백질, 지방, 당류, 나트륨 등 사용자 호응이 높은 영양성분을 표로 시각화합니다.

사용자는 AI가 분석하고 제공한 정보를 구체화할 수 있습니다. 음식명을 구체화하거나 제시된 영양성분 정보를 수정할 수 있습니다. 식재료에 따른 개인별 오차를 사용자가 직접 조정합니다. 이는 개인이 섭취한 음식을 더 정확하게 기록하는 데에 기여합니다. 음식 인식에 실패하거나 영양 정보를 찾지 못한 경우에도 에러 메시지를 띄우고 해당 정보를 직접 입력하도록 유도합니다.

### 3.2 신뢰성

어플리케이션에서 사용하는 음식 및 영양 성분 데이터는 공신력 있는 출처에서 얻어야 하며, 정기적으로 업데이트 합니다.

AI 모델의 학습 데이터 품질과 학습 방식이 신뢰성에 큰 영향을 미칩니다. 따라서 충분한 데이터를 바탕으로 AI 모델을 학습시키고, 다양한 텍스트를 통해 모델의 예측 정확도를 높입니다.

분석 과정에서 문제가 발생했을 때 사용자에게 오류 메시지를 제공하거나 대체 경로(예: 수동 입력 방식)를 안내하여, 사용자의 불편을 최소화합니다.

어플리케이션이 사진 분석 및 영양 성분 추출 과정을 예측 가능한 시간 안에 처리할 수 있도록 성능을 최적화합니다. 이를 통해 사용자가 신뢰할 수 있는 일관된 경험을 제공할 수 있습니다.

시스템 성능을 지속적으로 모니터링하고, 필요할 경우 성능을 조정하여 처리 속도나 정확도 저하를 방지합니다.

사용자 식단 관리 및 개인 정보가 저장될 경우, 이를 안전하게 보호하는 암호화 기술과 보안 프로토콜을 사용합니다. 신뢰할 수 있는 보안 체계를 통해 사용자의 신뢰를 얻습니다.

## 3.3 성능

음식 사진을 분석하거나 OCR 로 가공식품 라벨을 읽는 기능은 최대한 빠른 시간 내에 이뤄져야 합니다. 이 기능을 정상적으로 사용할 때, 영양성분 정보 생성까지 걸리는 시간은 7 초 이내입니다.

OCR 을 이용한 제품 탐색의 정확도의 목표치는 약 95%입니다. 이는 초기 정확도인 약 70%를 기초하여 정해졌으며, 제품명 내 띄어쓰기와 이미지 편집 등을 통해 최대한 정확도를 올릴 예정입니다.

이미지 분석 정확도의 목표치는 약 90%입니다. 이는 초기 정확도인 약 75%를 기초하여 정해졌으며, 이미지 내 음식의 구도와 품질 등을 향상시킴으로써 개선될 것으로 예측됩니다.

사용자가 섭취한 음식과 영양성분을 저장하기 위해 Firebase 의 Storage 를 이용합니다. 무료 요금제의 경우 5GB 의 저장공간을 이용할 수 있으며, 정보는 텍스트 형태로 저장됩니다.

결과 및 성능의 정확도를 측정하고 증명하기 위한 사용자 만족도 조사가 이뤄집니다. 애플리케이션 제작 후 시범 서비스를 제공하고, 사용자 만족치를 측정합니다. 사용자가 제공한 사진에 대한 서비스 결과를 제공하고, 이에 대한 만족도를 조사합니다. 다양한 항목의 조사 평균치가 10 점 만점 중 8 점이 되도록 목표합니다. 기준치가 8 점인 이유는 본 서비스의 핵심 기술인 사진 내 음식 분석 기술의 정확도가 약 95%이기 때문입니다.

## 3.4 이식성

Android 운영체제 및 기반환경에서 동작하도록 합니다.

Android 가 아닌 IOS 기반 운영체제 등에서는 사용할 수 없습니다.

모바일 외 PC, 태블릿 등 장치에선 동작하지 않습니다.

## 3.5 유지관리

추후 기능 및 성능 업데이트와 협업을 위해 GitHub를 이용해 버전 관리를 합니다.

새로운 음식 데이터나 영양 성분이 추가되는 것에 대비하고, AI 모델이 더 많은 데이터를 학습하여 성능을 개선할 필요가 있을 때 소프트웨어 업데이트를 합니다. 음식 및 영양 성분 데이터의 일관성을 유지하고, 최신 상태로 유지하기 위해 데이터베이스를 정기적으로 점검 및 업데이트 합니다.

사용자가 문제를 겪을 때 이를 보고할 수 있는 시스템(문의사항)을 마련하고 빠르게 버그를 파악하고 수정합니다. 사용자의 피드백을 지속적으로 수집하여, 이를 바탕으로 성능을 개선하고 업데이트합니다.

## 3.6 구현상 제약사항

모바일 환경에서 원활히 구동되기 위해선 Android 13 이상을 갖춘 환경이 요구되기에, Compile SDK Version을 33 이상으로 설정합니다.

실시간으로 정보를 처리해야 하기 때문에 네트워크 연결이 원활한 환경이 요구됩니다. 오프라인에서 사용할 수 없습니다.

영양성분 데이터를 제공하는 외부 API의 조건에 따라 데이터의 갱신주기가 제한적일 수 있습니다.

사진 분석 및 영양성분 정보 제공은 doinglab에서 제공하는 기술을 이용합니다.

기본적인 영양성분 정보는 식품의약품안전처에서 제공하는 식품영양성분 표준데이터를 이용합니다.

가공식품의 라벨링에서 제품명을 찾기 위한 OCR 기술은 오픈소스를 이용합니다.

## 3.7 인터페이스

Firebase - Cloud Storage

식품의약품안전처 - 식품영양성분 데이터베이스 API

doinglab - FoodLensSDK SDK

애플리케이션 내에서 카메라를 실행

## 3.8 법적 제약사항

사용되는 기술이 가진 라이선스 조건, 외부 API에서 제공받는 데이터의 사용권한 등을 명확히 확인하고 준수합니다.

본 서비스는 수익을 창출하지 않으므로 수익성이 없음을 밝힙니다.