

# [16팀-배꽃엔딩] 1차 보고서

## (1) 팀 정보

### 1.1 과제명

: 건강 관리가 필요한 소비자들을 위한, GPT-5 기반 맞춤형 가공식품 비교·추천 웹 서비스

### 1.2 팀 정보: 16팀 배꽃엔딩

1.3 팀 구성원: 2176038 김나현 (팀장 및 AI), 1876019 권서현 (BE), 2176360 조세령 (FE)

## (2) Project-Summary (과제 요약)

### 2.1. 문제 정의

#### - 과제의 배경

최근 들어 건강에 대한 관심도가 높아지며 건강기능식품을 중심으로 다양한 섭취 관리 서비스들이 등장하고 있다. 특히 비타민, 오메가-3 등 건강보조제를 정기적으로 배송하거나 복약 시간을 알려주는 앱은 이미 상용화되어 있으며, 소비자들로부터 큰 호응을 얻고 있다.

그러나 이러한 기술 발전에도 불구하고, 정작 일상에서 가장 자주 섭취하게 되는 가공식품에 대한 정보 제공과 섭취 관리 서비스는 상대적으로 부족한 실정이다. 현재의 온라인 쇼핑 환경에서는 수많은 가공식품이 판매되고 있음에도, 소비자 본인이 직접 상품을 선택하여 상세 화면에 들어간 후, 페이지 하단의 성분표를 일일이 확인해야 하고 판단해야 하는 구조에 머물러 있다. 특히 건강에 특별한 주의가 필요한 사람들은 알레르기 유발 성분, 임신부·영유아 금지 성분, 질환 관련 유해 성분 등에 무심코 노출될 수 있는 위험이 존재한다.

이러한 배경 속에서, 사용자의 건강 조건을 반영하여 가공식품을 자동으로 필터링하고 적합한 제품을 추천해주는 서비스가 필요하다고 생각된다. 그러나 현재까지는 개인 건강 정보와 가공식품 성분 정보를 유기적으로 연결해주는 플랫폼이 부재한 상황이다. 따라서 본 프로젝트는 기존 서비스의 한계를 넘어 일상 속 가공식품을 안전하고 편리하게 선택할 수 있는 맞춤형 추천 시스템의 필요성을 인식하고 이를 해결하고자 한다.

#### - 과제의 필요성 (Pain point)

##### ① 온라인 쇼핑 확산과 제도적 한계로 인한 영양 정보 활용 공백

최근 연구 결과에 따르면 온라인 쇼핑의 확산과 현행 표시 제도의 한계로 인해 소비자들의 성분표 확인은 줄어들고 점점 번거로워지고 있다.<sup>1)</sup> 그 결과 영양 정보를 쉽게 이해·활용하지 못해

일상적인 건강 관리에 공백이 발생하고 있으며, 이는 특히 알레르기, 임신, 만성질환 등 건강 조건을 가진 사용자에게 더 큰 위험으로 작용한다.

→ 따라서 개인의 건강 정보와 식품 성분을 연계하여 직관적이고 맞춤형으로 안내해주는 서비스가 필요하다.

### ② 맞춤형 관리 니즈의 증가에 비해 부족한 식품 관련 개인화 서비스

맞춤형 서비스를 위한 데이터 공유 의지는 최근 몇 년 사이 20%로 꾸준히 증가한 만큼 점점 높아지는 추세<sup>2)</sup>지만, 이에 비해 성분 정보와 개인 건강 데이터를 효과적으로 연동해주는 온라인 개인화 서비스는 매우 제한적이다. 알레르기, 특정 질환, 건강상 주의사항 또는 관심사 등을 반영해 식품을 자동으로 추천하거나 위험요소를 실시간 필터링해주는 서비스는 부재한 상황이다.

→ 이에 따라 소비자의 실제 수요와 서비스 간의 간극을 메울 필요성이 제기된다.

### ③ 기존 DB·검색 기능의 한계

식품의약품안전처에서 관리하는 식품안전나라에서는 식품영양성분 데이터베이스를 통해 가공식품의 영양 성분 데이터를 제공하지만, 사용자 관점에서는 성분 함량을 기준으로 정렬할 수 없다는 한계가 있다. 또한 원하는 제품군을 선택하여 영양 성분을 비교하는 기능은 존재하지만, 비교 결과를 자동으로 도출해 보여주는 것이 아니라 단순한 데이터 나열에 그친다.<sup>3)</sup>

→ 따라서, 사용자는 여전히 직접 수치를 확인하고 해석해야 하는 불편을 겪는다. 특히 알레르기, 임신, 특정 질환 등 개인별 건강 조건에 따라 성분을 따져야 하는 경우, 기존 시스템만으로는 효율적인 선택을 하기 어렵다. 따라서, 단순 정보 제공을 넘어, 사용자의 개인 건강 정보를 반영하여 실질적인 도구로 기능하는 서비스가 필요하다.

## 2.2. 기존 서비스와의 비교

- “팜픽” : 영양제 비교, 분석/추천, 이미지 검색 기능을 제공하는 서비스로, 건강 보조 목적으로 섭취하는 영양제에 초점을 맞추고 있다.

### \* 기존 서비스의 특징·한계

- ① 함량 합산 분석 제공 → 영양제 성분의 중복 여부와 총 함량을 계산하는 기능 제공
- ② 성분 비교를 직관적으로 지원하여, 여러 영양제를 나란히 비교 가능
- ③ 그러나 대상 품목이 영양제·건강기능식품에 한정됨

### \* 본 서비스의 차별점

- ① 가공식품을 주요 대상으로 하여, 건강보조제보다 더 일상적으로 섭취하는 제품군을 다룸
- ② 성분 비교를 직관적으로 지원할 뿐만 아니라, 성분 함량별 정렬·다중 성분 종합 비교 기능을

1) 출처: <https://www.nature.com/articles/s41598-024-51948-1>

2) 출처: <https://www.thinkfood.co.kr/news/articleView.html?idxno=102883>

3) 참고: <https://various.foodsafetykorea.go.kr/nutrient/general/food/firstList.do>

제공 → 개인 건강 조건에 맞춰 필터링이 가능

③ 적합 여부 자동 판단 및 대체 제품 추천 단순 비교를 넘어, 실제 선택을 지원하는 것을 목표

- “식품영양성분 데이터베이스” :

\* 식품영양성분 데이터베이스 검색 기능의 특징과 한계

① 영양 성분별 검색 가능 (식품명·영양소 기준으로 찾기)

② 사용자가 직접 원하는 여러 제품을 선택하여 단일 영양 성분 중심 비교가 가능하지만, 여러 성분을 동시에 종합 비교하기가 어려움

③ 필터링 기능이 제한적 (함량 범위 입력 가능하지만, 다차원적 조건 반영은 한계)

④ 비교 결과의 자동 분석 기능이 부재하여, 데이터가 단순히 나열되는 수준에 그침.

\* 본 서비스의 차별점

① 여러 영양 성분을 동시에 종합 비교 가능

② 성분 함량별 정렬 및 사용자 조건별 필터링을 지원하여 더욱 실용적인 검색 기능 구축

③ 사용자 건강 데이터와 대조하여 제품의 적합 여부를 자동 판단

④ 추천 + 대체 제품 제시 + 설명 제공 → 개인별 정보 맞춤형 추천 알고리즘 구축, 단순한 정보 제공을 넘어 실질적으로 개인의 건강에 맞는 도움 제공을 목표.

## 2.3. 제안 내용

- 목표 사용자 : 건강에 관심이 있는 소비자

① 알레르기 등 특정 성분을 피해야 하는 사람

② 임산부 등 섭취 성분에 주의가 필요한 사람

② 특정 건강 상태이거나 질환을 앓고 있어 특정 성분의 섭취량에 조절이 필요한 사람

③ 질환이 없어도 건강을 위해 당류·나트륨 등 특정 성분의 섭취량을 조절하고 싶거나, 일상 속 건강 관리를 원하는 사람

이렇듯 건강에 관심이 있거나 관리가 필요한 소비자가 온라인 쇼핑몰에서 가공식품을 선택하게 될 때,

① 제품별 영양 정보 즉시 분석: 온라인 쇼핑 시 복잡한 라벨을 직접 확인하지 않아도, 서비스가 DB에 저장되어 있는 제품의 영양 성분을 자동으로 불러오고·분석하여 사용자의 건강 조건과 대조. 이를 통해 각 제품의 적합 여부를 즉시 확인.

② 개인 맞춤형 제품 추천 제공: 서비스는 사용자의 건강 데이터와 제품 성분 정보를 연결해, 개인이 설정한 조건에 맞는 가공식품을 자동으로 탐색·추천해줌. 뿐만 아니라, 사용자와의 상호작용을 통해 반복적인 피드백을 반영함으로써, 개인에게 더 잘 맞는 상품을 보다 더 정교하게 추천 가능.

③ 손쉬운 비교·탐색 기능 제공: 기존 DB 및 상품 검색 기능의 한계를 보완해 성분 함량별 정렬·필터링 및 상위 순 추천을 지원, 효율적 상품 비교·선택 가능

의 기능을 제공하는 개인 맞춤형 식품 탐색 경험을 제안한다.

## 2.4. 기대 효과 및 의의

이 서비스를 사용함으로써, 사용자는 성분표를 일일이 확인하지 않아도 된다. 서비스가 제품의 성분과 원재료를 자동으로 분석하여 사용자의 건강 조건과 비교해 주기 때문에, 각 제품이 적합한지 여부를 즉시 알 수 있다. 또한 왜 해당 제품이 적합하거나 부적합한지 그 이유까지 제공하여, 보다 편리하게 이해할 수 있다.

서비스는 특정 상품의 적합 여부를 분석한 결과를 바탕으로 동일한 상품군 내에서 유사하거나 대체할 수 있는 제품을 자동으로 선별한다. 사용자는 추가 탐색이나 직접 비교를 하지 않아도 손쉽게 대체 제품을 추천받을 수 있으며, 이를 통해 선택의 폭을 넓히고 본인에게 더 적합한 제품을 쉽게 찾을 수 있다.

아울러, 서비스는 사용자가 설정한 기준에 따라 선택한 제품들을 분석·필터링한 뒤, 그 결과를 적합도 순으로 정렬해 보여준다. 사용자는 여러 제품을 직접 비교하면서 어떤 제품이 자신에게 더 맞는지를 쉽게 파악할 수 있으며, 이를 통해 보다 합리적인 선택을 할 수 있다.

→ 결과적으로, 이 서비스는 앞서 서술했던 기존 서비스의 한계와 불편함을 극복할 수 있을 것으로 예상된다.

## 2.5. 주요 기능 리스트

- Google Vision API: 상품 상세 페이지에서 영양 성분표와 알레르기 주의 성분 이미지를 크롤링하여 OCR로 분석하고, 이를 데이터베이스로 구축한다. 이를 통해 사용자가 상품 정보를 빠르고 직관적으로 확인할 수 있도록 한다.
- XGBoost: 코사인 유사도를 계산하여 산출된 유사도 값을 이용하고, 상품 데이터 속 성분과 원재료를 분석하여 사용자의 건강 정보에 맞는 상품을 1차적으로 추천한다.
- LightFM: 1차 추천 이후 앱 사용 과정에서 쌓이는 데이터 로그나 피드백을 반영하여, 사용자의 기호를 학습하고 더 알맞은 제품을 보완 추천한다.
- GPT-5: XGBoost와 LightFM이 제안한 상품 목록을 다시 검토하여 적합성을 교차 검증하고, 상품 성분 정보와 사용자의 건강 프로필을 함께 분석한다. 그 결과 사용자 개인에게 맞는 상세 설명을 제공하며, 프로필에 기반한 질의응답이 가능하도록 프롬프트를 구성하여 챗봇 기능을 수행한다.

## 1. 상품 검색 기능 및 상세 정보 표시 기능

: 이 기능은 사용자가 원하는 상품을 손쉽게 찾고, 성분 정보를 직관적으로 확인할 수 있도록 설계되었다. 사용자는 검색창에 키워드를 입력하여 관련 상품을 조회할 수 있으며, 검색 결과는 단순한 이름 매칭을 넘어 성분 함량(당류, 나트륨, 칼로리 등) 기준으로 정렬·필터링이 가능하다. 즉, “나트륨 낮은 순”, “칼로리 높은 순”과 같이 사용자의 관심 성분에 따라 결과를 재배치할 수 있다. 상품 클릭 시에는 해당 제품의 영양 성분표와 원재료를 파악 가능하다.

→ RDS의 상품 데이터를 조회하고, 성분별로 정렬된 결과 표시. 초기 상품 데이터는 Google Vision API를 이용하여 구축.

## 2. 상품 적합성 여부 판단 기능

: 이 기능은 사용자가 선택한 상품이 자신의 건강 상태에 적합한지 즉시 판단할 수 있도록 돕는다. 사용자는 알레르기, 질환 등의 정보를 사전에 등록해두며, 해당 정보는 적합성 판단 시 자동으로 참조된다. 상품 선택 시 시스템은 제품의 성분 정보를 사용자 건강 조건과 대조하여 알레르기 성분이 포함된 경우 ‘부적합’, 그렇지 않은 경우 ‘적합’으로 표시한다. 또한, 단순한 적합·부적합 표시를 넘어, 같은 제품군 내 평균 성분 함유량과 비교한 분석 결과를 GPT-5가 자연어로 요약하여 제공한다. 예를 들어, “이 제품은 동일 제품군 평균보다 당류가 낮고, 나트륨이 평균 수준으로 적당합니다.”처럼 사용자가 이해하기 쉽게 설명을 덧붙여준다.

## 3. 챗봇 기능

: 챗봇 기능은 사용자가 상품과 관련해 부가적으로 궁금한 점이 있을 때, 자연어로 대화하며 정보를 얻을 수 있도록 설계되었다. GPT-5가 내장된 챗봇은 단순히 제품 정보를 나열하는 수준을 넘어, 사용자의 건강 조건과 관심 성분을 함께 고려하여 맞춤형 답변을 제공한다.

## 4. 대체/유사 상품 추천 기능

: 이 기능은 사용자가 선택한 상품의 적합 여부에 따라 유사하거나 대체 가능한 제품을 자동으로 추천받을 수 있도록 구성되어 있다. 선택한 제품이 사용자에게 부적합한 경우, 시스템은 동일 제품군 내에서 성분 구성이 유사하지만 건강 조건에 맞게 조정된 대체 제품을 제안한다. 반대로 적합한 제품일 경우, 사용자가 선호할 만한 비슷한 맛·성분·브랜드의 제품을 함께 추천한다.

→ 1차 추천은 코사인 유사도 계산 + XGBoost 분석을 통해 건강 조건에 적합한 상품 도출.

→ 이후 LightFM이 사용자 로그를 학습해 개인화 추천을 지속적으로 개선.

## 5. 사용자 조건 기반 상품 비교 기능

: 이 기능은 사용자가 직접 선택한 여러 제품을 본인의 건강 기준(예: 나트륨 낮게, 단백질 높게)에 따라 비교할 수 있도록 한다. 사용자는 비교하고자 하는 상품들을 선택하고, 비교 기준이 될 성분(칼로리, 당류, 지방 등)을 자유롭게 설정할 수 있다. 시스템은 각 제품의 성분 데이터를 동일 기준으로 정규화하여 Z-score 계산을 통해 상대적 점수화를 수행하고, 각 성분별로 어느 제품이 더 우수한지를 시각적으로 보여준다. 비교 결과는 그래프나 표 형태로 표시되며, GPT-5가 그 결과를 분석해 “A제품은 칼로리가 가장 낮고 단백질 함량이 가장 높아, 건강 관리 목적에 가장 적합합니다.”처럼 자연어 형태의 요약 설명을 함께 제공한다.

### (3) Project-Design (과제 설계)

#### 3.1. 요구사항 정의

##### \* 기능적 요구사항

- 사용자 건강 조건 등록 기능: 사용자가 알레르기, 질병, 임신 여부 등 건강 조건 및 피해야 하는 원재료 또는 성분을 입력할 수 있어야 한다.
- 상품 검색 기능: 사용자가 원하는 가공식품의 키워드나 상품명을 입력했을 때, 검색이 가능해야 한다.
- 제품 정보 상세 표시: 제품 클릭 시, 성분표·영양 정보·적합 여부를 종합적으로 제공해야 한다.
- 제품 적합 여부 판단 기능: 등록된 조건에 따라 사용자가 선택한 제품이 사용자에게 적합한지 실시간으로 판단을 할 수 있어야 한다.
- 대체/유사 제품 추천 기능: 적합 제품이 선택되었을 때 그와 유사한 제품을 자동 추천하거나, 혹은 부적합 제품이 선택되었을 때, 선택한 상품과 유사하면서 조건에 맞는 다른 제품을 자동으로 추천할 수 있어야 한다.
- 조건 기반 제품 비교 기능: 사용자가 자신이 관심 있는 선택지를 고를 수 있게 하며, 고른 후에 사용자가 원하는 가공식품을 정렬할 수 있게 해야 한다.

##### \* 비기능적 요구사항

- 빠른 응답 속도: 추천 및 비교 결과는 2초 이내에 출력되어야 한다.
- 사용자 정보 보안: 개인정보 등 민감 정보는 암호화되어 저장되고 관리되어야 한다.
- 직관적인 UI 제공: 다양한 연령대의 사용자가 쉽게 접근할 수 있도록 구성되어야 한다.

#### 3.2. 전체시스템 구성

##### 1. 클라이언트(Frontend)

- React: 사용자 인터페이스(UI)를 구현한다.

##### 2. 서버(Backend)

- Spring boot: 비즈니스 로직과 REST API를 구현한다.

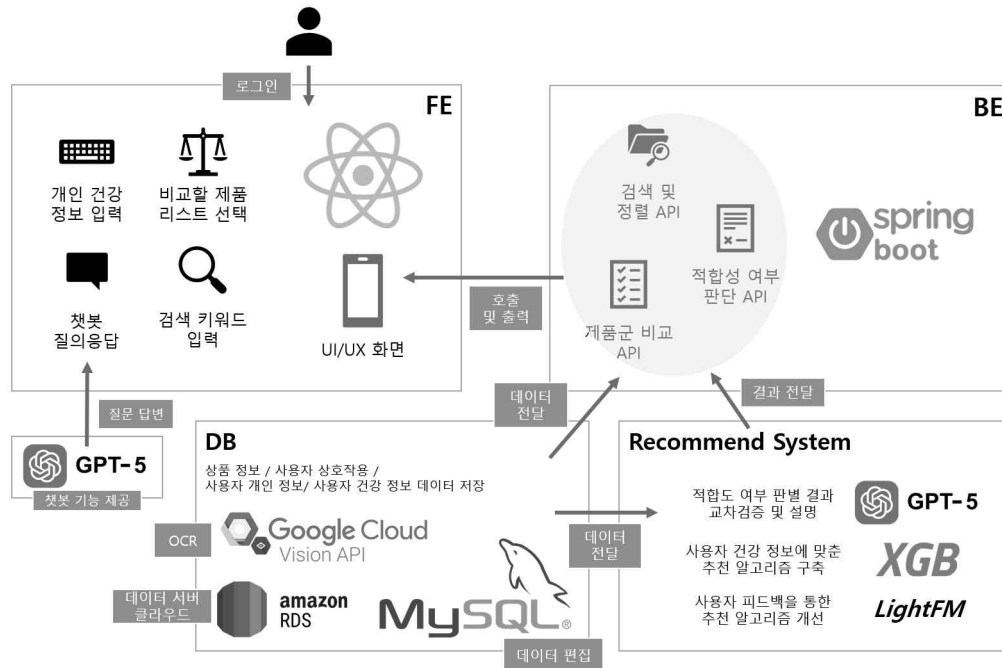
##### 3. 데이터베이스 / 데이터 수집

- MySQL: 가공식품, 사용자, 추천 결과 등 핵심 데이터를 저장·관리한다.
- AWS RDS(MySQL): MySQL을 관리형 클라우드로 운영하여 백업·확장·협업을 용이하게 한다.
- Google Cloud Vision API: 크롤링한 상품 상세 이미지(영양성분표/알레르기 표기)를 OCR로 처리해 원자료(raw data)를 구축한다.

##### 4. 분석 및 추천 시스템

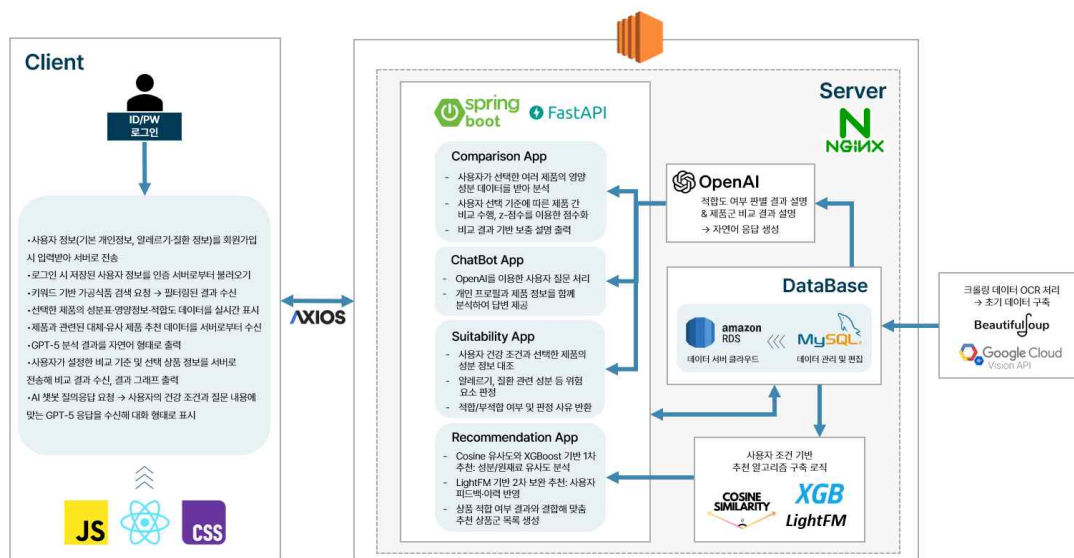
- XGBoost: 성분·원재료 정보를 사용자의 건강 정보와 매핑해, 로그/피드백이 없어도 1차 추천을 수행한다(콜드스타트 대응).
- LightFM: 사용 로그와 피드백을 학습해 개인 기호를 반영하고, 보완 추천·재랭킹을 수행한다.

- OpenAI GPT-5: 1차/보완 추천 목록을 사용자 건강 프로필과 성분 데이터로 교차 검증하고, 적합/부적합 이유를 자연어로 설명한다. 프로필 기반 질의응답(챗봇)도 제공한다.



#### + 개발 진척도:

현재 홈 화면과 검색 기능, 그리고 상품 상세 페이지 구현이 완료되어 정상적으로 작동 중. 전체 프로젝트에서 약 30~40% 정도의 기능 구현이 완료된 상태이며, 이를 통해 서비스의 기본적인 흐름을 확인할 수 있다. 특히 핵심 엔진 역할을 하는 데이터베이스 연동과 API 기본 구조가 성공적으로 구축·검증되어, 이후 기능 확장과 세부 구현을 위한 안정적인 기반을 마련 완료한 상태이다. 추후, 제품군 비교 기능과 챗봇 기능, 사용자 정보 입력 페이지를 추가로 구축하여 전반적인 서비스의 개발을 마무리 할 예정이다.



본 시스템은 Client-Server 구조로 구성되며, 클라이언트(React 기반 프론트엔드)는 사용자의 입력 데이터를 서버(FastAPI + Spring Boot)에 전달하고, 서버는 AI 추천 엔진과 데이터베이스를 통해 분석·판정·추천 결과를 반환한다.

- Client (JS, React, CSS): 사용자 정보 입력, 로그인, 제품 검색, 필터링, 비교 및 AI 챗봇 질의응답 인터페이스를 담당. 사용자는 회원가입 시 기본 개인정보와 알레르기·질환 정보를 입력한다. 키워드로 가공식품을 검색하면 필터링된 결과가 표시되고, 선택한 제품의 성분표·영양정보·적합도 데이터가 실시간으로 제공된다. 또한 제품과 관련된 대체·유사 상품 추천 결과를 함께 확인할 수 있으며, GPT-5가 분석한 내용이 설명으로 출력된다. 사용자가 설정한 비교 기준과 선택한 상품 정보를 서버로 전송하면, 비교 결과가 그래프로 시각화되어 표시된다. Axios를 통해 백엔드 API와 통신하며 JSON 기반 REST 구조를 따른다.

- Server (Spring Boot + FastAPI):

① Recommendation App - Cosine Similarity로 선택한 상품과 다른 상품 간의 유사도 점수 산출 후 적합도별 재정렬. 그 이후 XGBoost로 성분·원재료 기반 계산 수행 후 1차 추천 알고리즘 구축, 이후 사용하며 쌓인 사용자 로그를 이용하여 LightFM으로 사용자 피드백 및 이력 기반 2차 추천을 수행한다.

② Suitability App - 사용자가 가입 시 기재해 두었던 건강 조건(질환·알레르기 등)과 제품 성분을 매핑하여, 사용자가 보유한 알레르기 성분이 들어가 있을 경우 1차 필터링. 더 나아가, 간접 알레르기(“~를 사용한 제조시설에서 함께 제조됨” 등과 같은 문구) 함유 여부와 같은 제품군의 평균 성분 함유량을 참고하여, 사용자의 건강 조건에 맞지 않는 성분이 많이 들어가 있을 경우 주의 문구를 출력. 위험 성분 검출 및 부적합 사유 반환, GPT-5와 연동해 자연어 기반 판정 이유를 설명한다.

③ Comparison App - 사용자가 선택한 여러 제품의 영양 성분을, 사용자가 비교하고 싶은 기준 성분(당류, 나트륨 등)을 지정하여 비교할 수 있도록 한다. 각 제품군 내의 영양성분 함유량에 대한 평균과 표준편차를 이용, Z-정규화 기반의 점수를 계산하여 각각 (100/설정 기준 개수)



의 가중치를 부여한 뒤 100점 기준의 점수를 출력. 이 점수를 시각화 데이터로 비교하여 생성해 주고, GPT-5를 통해 요약·비교 근거 문장을 자동 생성하여 어떤 제품이 다른 제품들에 비해 어떤 면에서 좋은지를 설명한다.

④ ChatBot App - 사용자의 질의가 들어오면, 서버는 데이터베이스에서 해당 사용자의 건강 프로필(알레르기, 질환 등)을 조회하여 GPT-5 프롬프트 컨텍스트에 동적으로 주입하여 반영. 이를 통해 AI가 사용자의 건강 조건을 고려한 개인화된 답변을 생성하도록 설계한다.

- Database (MySQL / AWS RDS): Google Cloud Vision API와 BeautifulSoup을 활용해 가공식품 상세페이지의 영양정보·알레르기 문구를 크롤링·OCR 처리, 초기 데이터셋 구축. 사용자 프로필, 제품 영양성분, 추천 결과 등을 저장·관리하며, 서버와의 양방향 통신을 통해 분석 및 결과 데이터를 실시간 반영한다.

### 3.3. 주요 엔진 및 기능 설계

#### A. Suitability App (적합성 판단 모듈):

이 모듈의 기능은 A1. 건강 조건 기반 부적합 판정과 A2. 판정 근거 생성 및 GPT-5 연동이다. FastAPI를 기반으로 동작하며, MySQL(RDS)에서 사용자·제품 데이터를 조회하여 Pandas로 분석한 뒤, OpenAI SDK를 활용해 결과를 자연어로 설명하는 구조로 구현되었다.

\* A1. 건강 조건 기반 부적합 판정: 이 로직은 FastAPI 기반으로 동작하며, 서버는 DB에 저장된 사용자의 건강 조건 (질환명, 알레르기 등)을 불러와 미리 구축해 둔 딕셔너리 속 규칙들과 매칭하는 방식으로 구현되었다.

이 로직에서 서버는 다음과 같은 절차로 데이터를 처리한다:

① 서버는 DB에서 요청을 보낸 사용자 정보를 불러와 파싱하며, 동시에 선택한 제품의 데이터를 불러와 계산 가능한 형태로 정규화한다. 사용자 질환별로 사전에 정의된 규칙을 참조한다.(ex. "고혈압": {"나트륨": "low"}, "당뇨": {"당류": "low"}) 질환별로 관리되는 성분 방향성(높게/낮게)을 자동으로 매핑하며, 여러 질환이 동시에 존재할 경우 중첩된 성분 기준을 통합하여 사용자가 지닌 질환에 맞게끔 출력하도록 한다. (예: 고혈압+당뇨 → {"나트륨": "low", "당류": "low"})

② 사용자가 선택한 제품을 기준으로, 해당 제품군 데이터셋 내 평균값을 비교 기준으로 삼는다. 제품별로 "100g 기준 영양 성분"을 환산하여 평균값을 계산하고, 현재 제품의 값을 전체 평균과 비교해 평균보다 높음 / 낮음 / 비슷함으로 평가한다. (\*제품이 특정 성분을 0으로 표기한 경우는 "미함유", 데이터가 없는 경우는 "정보부족"으로 분류) 이 과정을 통해, 사용자의 건강 상태에 따라 제품이 적합한지 여부와 성분별 평가 결과가 산출된다.

③ 제품 원재료의 알레르기 항목과 사용자의 알레르기 리스트를 대조해 직접 포함 여부를 판정하고, 포함 시 "부적합" 결과로 우선시하여 지정한다. 추가로 간접알레르기 항목을 탐색해, 여부에 따라 부적합 결과가 아닌 "△'~성분' 간접 알레르기 주의" 경고 문구를 반환하도록 한다.

```

# -----
# 📌 건강목표별 성분·방향 매핑
# -----
health_condition_rules = {
    "고혈압": {"나트륨": "low"},
    "당뇨": {"당류": "low"},
    "감량": {"칼로리": "low"},
    "고지혈증": {"지방": "low", "포화지방": "low", "트랜스지방": "low"},
    "심혈관질환": {"나트륨": "low", "포화지방": "low", "콜레스테롤": "low"},
    "신장질환": {"나트륨": "low", "단백질": "low"},
    "간질환": {"당류": "low", "지방": "low"},
    "골다공증": {"칼슘": "high", "나트륨": "low"},
    "고콜레스테롤혈증": {"콜레스테롤": "low", "포화지방": "low"},
    "통풍": {"단백질": "low"},
}

```

\* A2. 판정 근거 생성 및 GPT-5 연동: 이 단계에서는 A1에서 도출된 결과 리스트를 요약하여 "성분(평가, 방향)" 형태의 문자열을 만든다. GPT-5는 프롬프트 내에서 DB에서 불러온 사용자 건강 정보(질환·알레르기 등) 을 직접 참조하며, 이를 바탕으로 제품의 성분 특성과 비교하여 “이 제품이 왜 적합/부적합한지”를 개인 맞춤형 자연어로 설명하는 방식으로 구현하였다.

OpenAI에 사용할 사용자 알레르기, 건강 목표, 제품명, 최종 판정, 경고문구를 포함한 프롬프트를 구성한다. GPT 모델에 프롬프트를 전달하고 temperature=0.4로 설정해 응답의 일관성을 유지하고, 생성된 답변을 출력한다. 생성된 설명은 최종 결과에 포함되어 프론트엔드에 반환되어 사용자에게 출력된다.

## B. Recommendation App (제품 추천 모듈):

이 모듈은 B1. 1차 콘텐츠 기반 추천과 B2. 2차 개인화 보완 추천 단계로 이루어져 있다. FastAPI를 기반으로 동작하며, MySQL(RDS)에 저장된 제품 데이터를 불러와 Pandas로 전처리한 뒤, 코사인 유사도 계산, XGBoost와 LightFM 알고리즘을 결합하여 사용자 건강 목표와 행동 이력을 반영한 맞춤형 추천 결과를 생성하는 구조로 구현되었다.

\* B1. 1차 콘텐츠 기반 추천: 코사인 유사도(Cosine Similarity)와 XGBoost 모델을 결합한 방식으로 구현되었다. FastAPI 서버는 RDS에 저장된 제품의 성분·원재료 데이터를 불러온 후, 사용자의 건강 목표(예: 당류↓, 단백질↑, 칼로리↓)를 반영하여 각 제품 벡터와 사용자 선호 벡터 간의 코사인 유사도를 계산한다.

이렇게 산출된 유사도 값은 XGBoost 회귀 모델의 입력 피처로 사용되며, XGBoost는 각 성분이 사용자의 건강 목표 달성에 기여하는 정도(가중치)를 학습하여 최종적으로 제품별 “적합도 점수”를 예측한다. 이 과정에서 LightFM 이전 단계의 1차 후보군(Top N) 이 생성되어 사용자에게 1차적으로 추천된다.

\* B2. 2차 개인화 보완 추천: LightFM(WARP 손실 기반 협업 필터링)으로 구현한다. LightFM은 사용자-제품 상호작용 로그(interactions 테이블의 view, click 이벤트를) 이용해 개인의 실제 행동 패턴을 학습하고, XGBoost 1차 후보군을 입력받아 개인별 선호도를 보완 재정렬한다. Cold Start 사용자에게는 콘텐츠 기반 점수를 가중 적용하여 성능 저하를 방지하며, 주기적으로

로그 데이터를 배치 학습하여 최신 사용자 패턴을 반영한다.

Recommendation App은 B1, B2 순으로 실행되며, B1의 처리 알고리즘은

① 사용자 목표 성분 방향성에 따라 각 제품의 성분 벡터 생성 → ② 사용자 벡터와 제품 벡터 간의 코사인 유사도 계산 → ③ 유사도 + 성분 피처를 XGBoost에 입력하여 제품별 적합도 예측 → ④ 상위 N개 제품을 후보군으로 선정  
순서로 일어난다.

이후 B2 단계에서 LightFM이 사용자 로그를 기반으로 후보군을 재랭킹하며, 점차 개선된 추천 리스트를 구성한다.

### C. Comparison App (상품 비교 모듈):

이 모듈은 C1. 선택 제품의 성분 비교 및 점수화와 C2. AI 기반 결과 요약 및 GPT-5 연동 단계로 이루어져 있다. FastAPI를 기반으로 동작하며, MySQL(RDS)에서 사용자가 선택한 제품의 영양 데이터를 불러와 Pandas로 정규화·계산한 뒤, 그래프 형태로 결과를 출력하고 OpenAI SDK를 이용해 비교 결과를 요약해 보충 설명을 제공하는 구조로 구현되었다.

\* C1. 선택 제품의 성분 비교 및 점수화: 사용자가 프론트엔드에서 선택한 여러 제품과 평가 기준 (“높게/낮게”)을 서버에 전달하면, 해당 제품들의 영양 정보를 불러와 Z-score 표준화 기반으로 비교·점수화하는 역할을 수행한다.

이 로직에서 서버는 다음과 같은 절차로 데이터를 처리한다:

- ① 사용자가 지정한 제품과 기준 성분을 프론트엔드에서 JSON 형식으로 전달받아 파싱하고, 제품들의 소속 제품군을 DB에서 조회한다.
- ② 비교 대상 제품군에 대한 평균(mean)과 표준편차(std) 값은 미리 계산되어 DB에 저장되어 있으며, 이를 가져와 이용한다. (표준편차가 0 이하일 경우 1e-9로 보정한다.)
- ③ 선택된 제품들의 영양성분 데이터를 제품 데이터 해당 제품군의 테이블에서 불러오고, 수치를 float형으로 변환해 계산 가능한 형태로 정규화한다.
- ④ 각 성분에 대해  $(x - \text{평균}) / \text{표준편차}$  식으로 Z-score를 계산하고, 사용자가 “낮게”를 선택한 항목은 부호를 반전시켜 기준에 부합할수록 점수가 높게 산출되도록 조정한다.
- ⑤ 모든 성분 점수를 0~100 구간으로 변환한 후, 성분별 가중치를 (100/선택 기준 개수) 비율로 동일하게 부여해 평균을 계산. 이를 종합하여 최종 총점을 산출한다. 계산 결과는 내림차순으로 정렬되어 가장 기준에 부합하는 제품이 상위에 위치하게끔 하며, JSON 형식으로 프론트엔드에 반환된다. 결과는 그래프 형식을 이용하며, 점수가 높은 제품 순서대로 분석 결과를 노출하여 사용자에게 보기 쉽게 제공된다.

\* C2. AI 기반 결과 요약 및 GPT-5 연동: C1에서 계산된 결과를 바탕으로 GPT-5 모델이 제품 간 차이와 추천 이유를 3문장 이내로 요약하는 단계이다. 서버는 비교 결과와 사용자의 평가 기준을 프롬프트에 포함시켜 GPT-5에 전달하고, GPT-5는 각 제품의 평균 대비 성분 차이와 기준 부합도를 분석하여, 어떤 제품이 더 적합한지를 전문적인 문체로 요약한다. 생성된 설명문은 최

종 결과 JSON 형식으로 프론트엔드에 반환되어 출력된다.

#### D. ChatBot App (질의응답 모듈):

이 모듈은 D1. 사용자 정보 기반 질의 처리와 D2. GPT-5 응답 생성 기능을 수행한다. D1 단계에서는 DB에서 사용자의 건강 정보를 조회하고, 해당 정보를 기반으로 사용자가 입력한 질문과 관련된 제품 데이터를 함께 불러온다. 이후 D2 단계에서 GPT-5 모델이 사용자 프로필과 제품 정보를 프롬프트에 반영하여, 사용자 개인의 건강 조건에 맞는 답변을 자연어로 생성한다. 이 모듈은 DB 내 사용자 정보와 제품 데이터를 동시에 활용하여 질의응답의 맥락을 개인화한다.

### 3.4. 주요 기능의 구현

본 프로젝트의 핵심은 사용자의 건강 정보를 기반으로 가공식품의 성분을 분석하고, 적합 여부를 판단 및 추천하는 것이다. 이를 위해 전체 시스템은 가공식품 적합성 판정 기능, 가공식품 추천 기능, 가공식품 비교 기능 3가지 주요 기능으로 구성된다.

#### 1) 가공식품 적합성 판정 기능

이 기능은 사용자가 특정 가공 식품을 선택할 시, 사용자의 건강 조건을 제품 성분 데이터와 대조하여 제품의 적합/부적합 여부를 자동 판단하고, 그렇게 판단한 이유를 사용자에게 설명하는 역할을 수행한다. 해당 기능은 Suitability 모듈을 사용하여 선택한 가공식품의 적합 여부와 이유를 출력한다.

먼저, 사용자가 가입 시 기재해 두었던 건강 조건(질환·알레르기 등)과 제품 성분을 매핑하여, 페이지 상단에 해당 제품이 사용자에게 적합 / 부적합 / 주의 필요 중 어느 상태인지 알려준다. 사용자가 보유한 알레르기 성분이 포함된 경우 ‘부적합’을, 간접 알레르기를 유발할 가능성이 있으면 ‘주의가 필요’를, 알레르기 관련 성분이 없을 경우 ‘적합’을 판정하여 화면에 출력한다.

또한 판정 결과 하단에는 GPT-5가 연동되어 해당 제품이 왜 적합하거나 부적합한지를 사용자에게 자연어로 해설한다. GPT-5는 Suitability 모듈이 반환한 위험 항목과 수치 데이터를 근거로, “당뇨 건강 목표를 고려할 때 특별히 피해야 할 성분은 없으므로 섭취에 적합한 제품입니다.”, “해당 제품은 나트륨 함량이 평균보다 높아 고혈압 관리에 부적합할 수 있습니다.”와 같은 설명을 생성하여 사용자에게 이해하기 쉽게 전달한다.

#### 2) 가공식품 추천 기능

이 기능은 성분 데이터를 기반으로 사용자의 건강 프로필에 맞는 제품을 추천해준다. 제품 추천 기능의 구현을 위하여, Suitability 모듈과 Recommendation 모듈을 도입하였고, 이를 통해 사용자의 건강 조건과 섭취 목표에 최적화된 맞춤형 추천 시스템을 구현하였다. Suitability 모듈은 앞서 이야기했던 것처럼, 사용자의 질환, 알레르기 등 건강 정보를 기반으로 제품의 성분을 분석하고 적합·부적합 여부를 판정하는 역할을 수행한다. 사전에 정의된 성분 규칙을 이용하여 나트륨, 칼로리 등의 수치를 질환별 기준에 맞춰 자동으로 대조하고, 부적합으로 판정된 제품은 추천 과정에서 자동으로 제외한다.

Recommendation 모듈은 XGBoost와 LightFM 알고리즘을 결합하여 사용자의 건강 목표와 실제 이용 패턴을 모두 반영한 개인화 추천을 수행한다. 1차 단계에서는 XGBoost가 코사인 유사도 계산을 통해 산출된 값을 이용하여 제품의 성분 데이터와 사용자의 건강 목표를 바탕으로 적합도 점수를 예측하고, 2차 단계에서는 LightFM이 사용자 로그(view, click 등)를 학습하여 개인 선호를 반영한 보완 추천을 수행한다.

이 두 모듈이 상호 연동되어, 사용자의 건강 상태에 부적합한 제품은 필터링되고, 그중에서도 개인의 선호도에 가장 잘 맞는 제품만을 추려 개선해 나가는 방식으로 추천 기능을 구축한다.

### 3) 가공식품 비교 기능

이 기능은 사용자가 선택한 여러 가공식품의 성분 및 영양 데이터를 병렬 비교하여, 어떤 제품이 더 적합한지를 0점 ~ 100점으로 환산해 시각적으로 보여주는 역할을 수행한다.

사용자는 검색을 통해 2가지 이상의 상품을 선택한 후, 비교 대상 성분 (칼로리, 나트륨, 탄수화물, 당류, 지방, 트랜스지방, 포화지방, 콜레스테롤, 단백질)중 최대 3가지를 지정할 수 있다. 각 성분에 대해 “높게/낮게”의 비교 방향을 설정한 뒤 ‘비교하기’ 버튼을 클릭하면, 선택된 제품·성분 정보를 JSON 형태로 /api/compare 엔드포인트에 전달한다.

Comparison 모듈은 데이터베이스에서 선택된 제품들의 성분 데이터를 불러와, 항목별 평균과 표준편차를 이용한 Z-정규화(Z-score) 계산을 수행한다. 이후 사용자가 설정한 비교 방향에 따라 성분별 가중치를 부여하고, 이를 합산하여 0점~100점 사이의 최종 적합도 점수를 산출한다. 계산된 점수는 높은 점수를 받은 순서대로 정렬되어 막대그래프 형태로 시각화된다. 또한 각 제품의 순위, 상품명, 선택 성분별 수치를 함께 표시하여 사용자가 성분 중심으로 직관적인 비교를 할 수 있다.

마지막으로, GPT-5가 Comparison 모듈의 계산 결과를 해석하여 요약·비교 근거 문장을 자동 생성한다. 예를 들어 ‘세 제품 중에서는 A제품이 칼로리, 당류, 지방, 콜레스테롤 모두 비교적 낮아 건강 기준에 가장 부합합니다. B제품은 당류와 지방이 다소 높고, C제품은 세 영양 성분 모두 기준에서 크게 벗어납니다.’와 같은 문장을 생성하여, 사용자가 수치 결과를 더 쉽게 이해할 수 있도록 돕는다.

### 3.5 기타 (AI 모듈 구성 관련)

모든 자연어 설명 관련 기술은 OpenAI SDK를 사용하여 GPT 모델(gpt-5)에 분석 데이터를 전달하고, 모델이 사용자 건강 정보와 제품 성분 데이터를 바탕으로 3문장 이내의 요약 설명문을 생성한다. 이를 위해 서버에서는 사용자 정보와 제품 분석 결과를 구조화하여 프롬프트로 구성하고, 모델이 판단의 맥락을 정확히 인식할 수 있도록 입력 데이터를 구체적으로 정리하였다.

```
#
# 📌 OpenAI 요약
#
try:
    nutrition_summary = ", ".join(
        [f"{n['성분']}{n['평가']}, {n['방향']}" for n in nutrition_results]
    )

    prompt = f"""
    사용자 조건:
    - 알레르기: {', '.join(user_allergies) if user_allergies else '없음'}
    - 건강목표: {', '.join(user_goals)}
    - 주요 평가 성분 및 방향: {nutrition_summary}

    제품명: {product_name}
    최종판정(알레르기 기준): {final}
    경고문구: {warning_text}

    이 정보를 바탕으로 다음을 자연스럽게 3문장 이내로 설명해줘:
    1. 제품의 성분 평가와 사용자의 건강 조건(알레르기, 목표 등)을 함께 고려해 판단할 것.
    2. "{final}"의 이유를 논리적으로 설명하되, 수치나 단위 대신 상대적 표현(평균보다 높음/낮음 등)을 사용할 것.
    3. 경고 문구가 존재한다면 마지막 문장에 주의사항으로 자연스럽게 포함할 것.
    4. 문체는 '~습니다', '~합니다' 형태의 공식적인 설명으로 작성할 것.
    """

    res = client.responses.create(
        model="gpt-5",
        input=prompt,
        temperature=0.3,
    )
    reason = res.output_text.strip()
except Exception:
    reason = "(AI 설명 생성 실패)"
```

이 프롬프트는 적합성 판단 모듈(Suitability App)의 결과를 자연어로 변환해 사용자에게 이해하기 쉽게 전달하도록 설계되었다.

① 사용자 조건 정보: DB에서 불러온 알레르기, 건강 목표(질환·섭취 제한 등)를 텍스트로 나열해 모델이 사용자별 맞춤 판단을 수행할 수 있도록 한다.

② 제품 정보 및 분석 결과: 선택된 제품의 이름, 최종 판정(적합/부적합), 경고 문구, 그리고 각 성분별 평가 결과(성분(평가, 방향) 형식)를 함께 제공하여 모델이 “어떤 이유로 판정이 내려졌는지”를 데이터 수준에서 이해할 수 있게 한다.

③ 출력 지침: 생성되는 문장은 과학적 근거에 기반한 요약으로 한정되며, 제품의 성분 평가와 사용자 건강 조건을 함께 고려할 것, 수치 대신 상대적 표현(예: 평균보다 높음/낮음)을 사용할 것, 경고 문구는 마지막 문장에 자연스럽게 포함할 것, 문체는 공식적 서술형(‘~합니다’, ‘~입니다’)으로 유지할 것 등의 구체적인 규칙을 명시해 모델 출력의 일관성과 신뢰성을 확보한다.

```
#
# 📌 결과 요약 + AI 설명
#
comparison_table = df_ranked[
    [
        "품명" + nutrient_list +
        [f"_{c}" for c in nutrient_list] +
        [f"{c}_weighted_score" for c in nutrient_list] +
        [f"final_score_100"]
    ],
    round(3)
]
summary_payload = comparison_table.to_dict(orient="records")

try:
    prompt = f"""
    사용자의 건강 기준: {json.dumps(user_standard, ensure_ascii=False)}

    아래는 {category_table} 제품군 내 선택된 상품의 영양성분 비교입니다:
    {json.dumps(summary_payload, ensure_ascii=False)}

    이 정보를 바탕으로:
    1. 계산 결과값을 바탕으로 어떤 제품이 기준에 가장 부합하는지.
    2. 각 제품의 평균값과 표준편차값을 참고하여 기준으로 선택한 영양성분의 정도를 자세히 비교해주고,
       (ex. 세 제품 모두 칼로리는 평균보다 높으나, 당류는 카르보볼락염류면이 가장 낮고 단백질은 오징어참돔이 가장 높지만, 종합 점수에서 파파게티가 균형 있게 우수합니다.)
    3. 그래서 어떤 제품을 추천하는지
    4. 2점수는 언급하지 말고, ~합니다, ~다 같은 전문적인 말투를 이용해서, 3문장 이내로
       문장과 문장이 자연스럽게 이어지게끔 한국어로 설명하는 문장을 출력해줘.
    """

    response = client.responses.create(model="gpt-5", input=prompt)
    ai_summary = response.output_text.strip()
except Exception as e:
    print("🔥 OpenAI 오류:", e)
    ai_summary = "(AI 설명 생성 실패)"
```

이 프롬프트는 상품 비교 모듈(Comparison App)의 결과를 자연어로 변환해 사용자에게 이해하

기 쉽게 전달하도록 설계되었다. 모델은 사용자의 건강 기준(예: 나트륨 낮게, 단백질 높게 등)과 각 제품의 점수화된 데이터를 함께 받아 어떤 제품이 기준에 가장 부합하는지를 3문장 이내로 요약해 설명한다.

① 사용자 건강 기준 입력: 프론트엔드에서 전달된 JSON 데이터(user\_standard)를 프롬프트에 삽입하여, GPT-5가 사용자가 지정한 성분 방향성(“낮게” 또는 “높게”)을 기준으로 분석을 수행하도록 한다.

② 비교 데이터 입력: 선택된 제품군(category\_table)에 속한 여러 제품의 영양 성분 점수(Z-score, 가중치, 최종 점수 등)를 JSON 형태(summary\_payload)로 변환해 프롬프트에 포함시킨다. 이를 통해 GPT-5가 실제 수치적 데이터를 근거로 판단할 수 있도록 한다.

③ 출력 지침 명시: GPT-5가 단순 요약이 아닌 논리적 비교 분석을 수행할 수 있도록 다음과 같은 지시사항을 포함한다. 계산 결과를 바탕으로 어떤 제품이 기준에 가장 부합하는지 판단할 것. 각 제품의 평균 대비 영양성분 차이를 구체적으로 비교·설명할 것. 종합적으로 어떤 제품이 가장 추천되는지를 제시할 것. 또한, 결과 문체는 “~합니다”, “~다”와 같은 공식적이고 전문적인 어투를 사용하도록 명시하여 설명의 신뢰성과 일관성을 확보하였다.

```
# ChatBot App - GPT-5 질의응답 프롬프트 구성
#
try:
    prompt = f"""
    아래는 사용자의 건강 정보, 최근 조회한 제품 정보, 그리고 사용자가 입력한 질문입니다.
    주어진 정보를 모두 참고하여, 사용자의 건강 상태를 고려한 정확하고 신뢰성 있는 답변을 생성하세요.

    [사용자 건강 정보]
    - 알레르기: {', '.join(user.allergies) if user.allergies else '없음'}
    - 질환 및 건강 목표: {', '.join(user.conditions) if user.conditions else '없음'}

    [참고 제품 정보]
    - 제품명: {product.name if product.name else '없음'}
    - 주요 성분 및 평가: {nutrition.summary if nutrition.summary else '정보 없음'}
    - 적합성 판정: {final if final else '판정 정보 없음'}

    [사용자 질문]
    "{user.question}"

    답변 작성 시 유의사항:
    1. 사용자의 알레르기 및 질환 정보를 반드시 고려해, 섭취 권장/주의 여부를 명확히 제시할 것.
    2. 제품에 대한 정보가 충분하지 않다면 일반적인 건강 기준에 따라 신중한 조언을 제공할 것.
    3. 수치나 단위 대신 상대적 표현(예: 낮음, 높음, 적당함)을 사용하고, 과도한 확증적 표현(예: 반드시, 절대)은 피할 것.
    4. 문체는 공식적이고 자연스럽게 유지하며, 2-4문장 내에서 간결하게 설명할 것.
    5. 질문이 제품 외 일반 건강 문의일 경우, 사용자의 건강 정보를 기반으로 일반적인 조언을 제시할 것.
    """

    res = client.responses.create(
        model="gpt-5",
        input=prompt,
        temperature=0.4
    )
```

이 프롬프트는 사용자의 건강 정보와 제품 데이터를 종합 분석하여 개인 맞춤형 상담형 응답을 생성하도록 설계되었다.

#### ① 사용자 건강 정보 입력

DB에서 불러온 알레르기 및 질환 정보를 텍스트 형태로 정리해 GPT-5에 제공한다. 사용자가 어떠한 성분을 피해야 하는지, 어떤 건강 목표를 가지고 있는지를 명시하여 모델이 답변 생성 시 이를 최우선 조건으로 고려하도록 한다.

#### ② 참고 제품 정보 입력

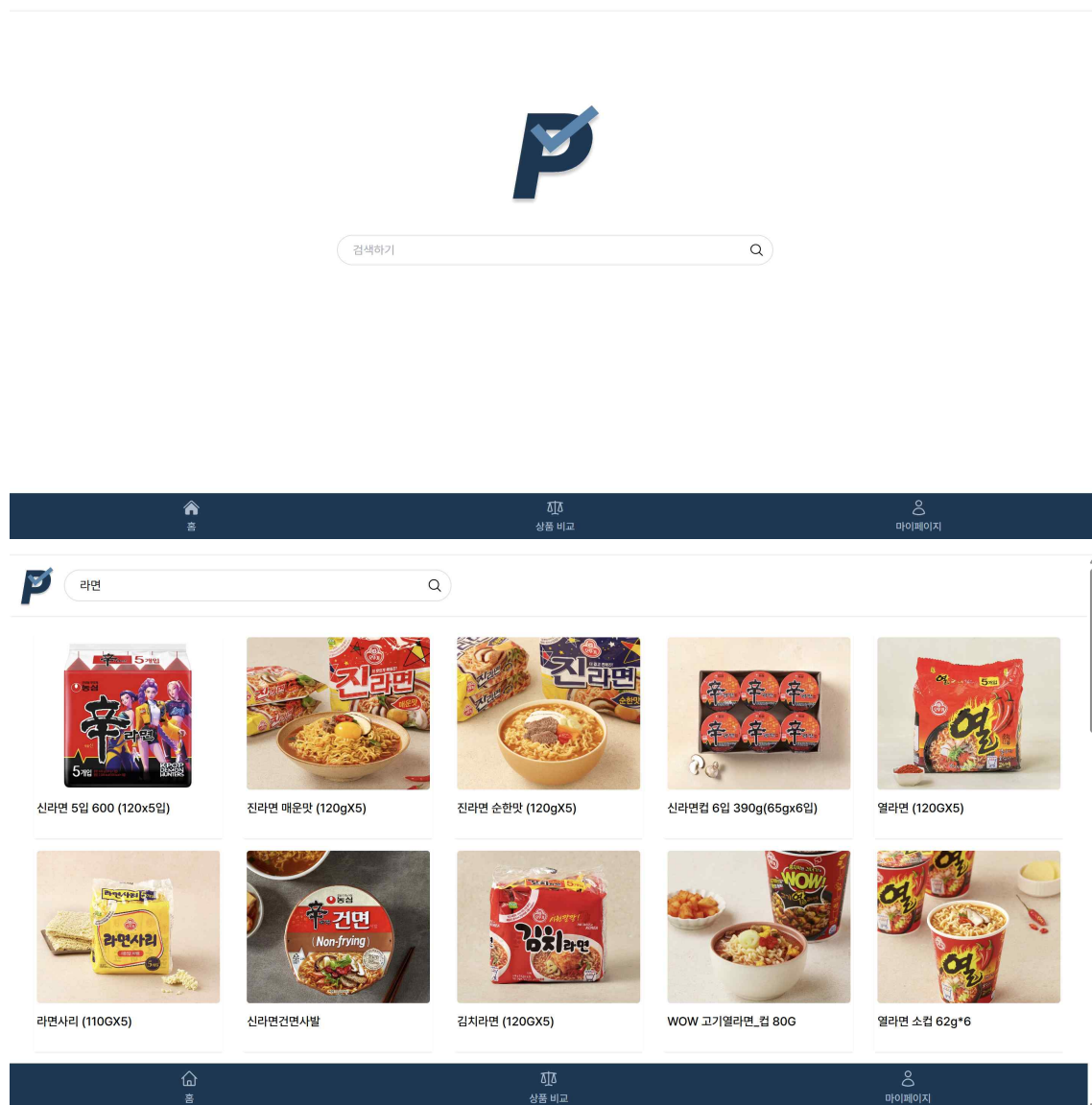
사용자가 최근 조회하거나 선택한 제품명, 주요 성분 평가 요약(nutrition\_summary), 적합성 판정(final)을 함께 프롬프트에 포함시킨다. GPT-5는 이 데이터를 참고하여 사용자의 질문과 연관



된 제품 성분을 분석하고, 해당 제품이 사용자 건강에 미칠 영향을 구체적으로 설명할 수 있다.

③ 사용자 질문 및 출력 지침: 사용자의 실제 질문(user\_question)을 명시하고, GPT-5가 생성할 답변의 형식과 내용에 대한 구체적인 작성 지침을 제시한다: 알레르기 및 질환 관련 주의 사항은 반드시 포함할 것. 제품 정보가 부족한 경우 일반적인 건강 조언으로 대체할 것. 수치 대신 상대적 표현(예: 낮음, 높음, 적당함)을 사용할 것. 문체는 공식적이며 자연스러워야 하고, 2~4문장 이내로 간결히 작성할 것. 질문이 제품 외 일반 건강 문의일 경우, 사용자 건강 정보를 기반으로 일반 조언을 제시할 것.

\* 개발 완료 화면 예시







진라면 매운맛 (120gX5)

영양 정보

열량	510
나트륨	1
탄수화물	79
당류	5
지방	16
트랜스지방	0
포화지방	8
콜레스테롤	0
단백질	12