

Safe Food Final Project

팀명: Do_or_Lock

강경도 김락현

CONTENTS

01

도입

- 기획배경 및 목표
- 아키텍처
- 사용 기술
- 작업 현황표

02

요구 사항 및 설계

- 요구사항
- usecase
- DB 설계
- business logic
- 화면 및 기능

03

구현

- class diagram
- 이벤트 플로우
- 기능

04

후기

- Lesson Learned
- Best Practice

01

- 안전 먹거리를 찾는 사용자 입장에서 생각하여 다양한 검색기능에 집중
- 안전하게 음식을 먹는 방법은 다양한 정보를 잘 알기 쉽고 편리하게 알아 볼 수 있어야 함
- 구글 비전 API를 이용한 이미지 검색
- 체크 박스를 이용한 검색 필터링 기능



[출처] 구글 이미지 검색
<https://www.google.co.kr/imghp?hl=ko>

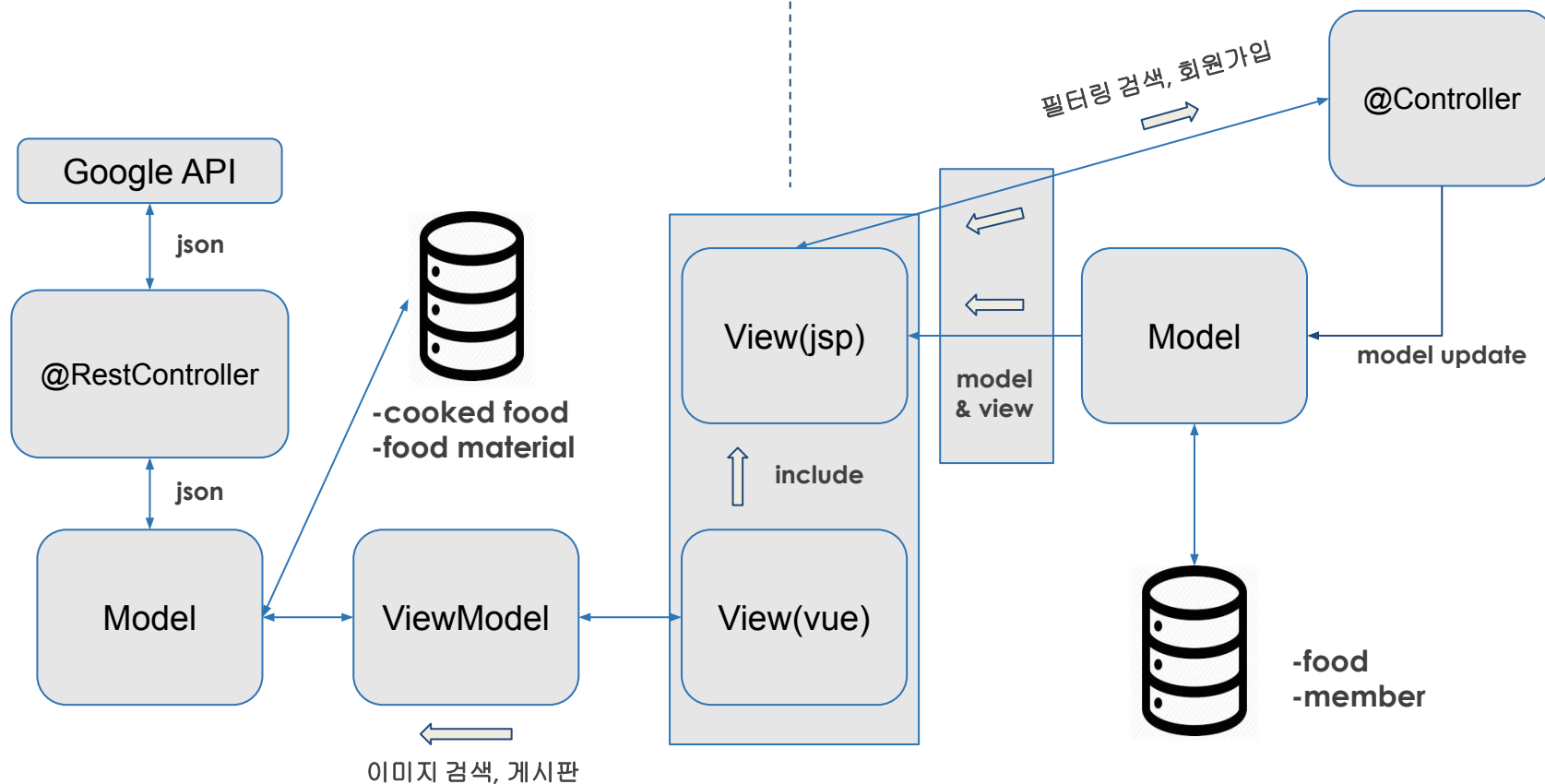


[출처] 다나와 검색
<http://search.danawa.com/dsearch.php?k1=rtx%202060>

01

spring boot

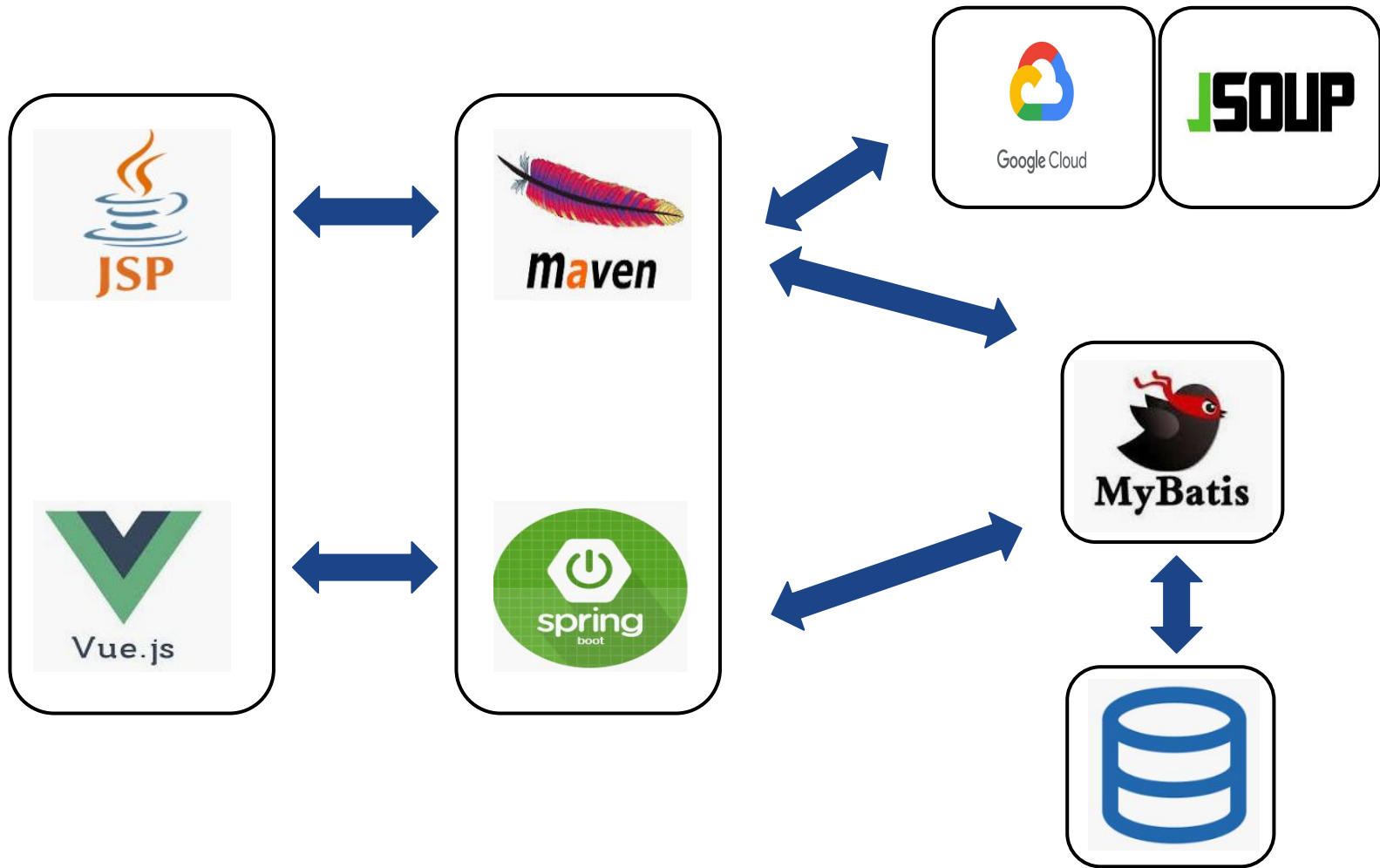
maven














MVVM

MVC

01



No.		작업 이름	기간	시작 날짜	완료 날짜	난이도	자원 이름	완료율
1		▲ 기본 기능	8 일	2019-11-06 (수)	2019-11-13 (수)			100%
2		DB 설계	3 일	2019-11-06 (수)	2019-11-08 (금)	상	강경도, 김락현	100%
3		Member 쿼리 작성	1 일	2019-11-09 (토)	2019-11-09 (토)	중	강경도	100%
4		Food 쿼리 작성	1 일	2019-11-09 (토)	2019-11-09 (토)	중	김락현	100%
5		Member DAO, Service, Controller 작성	1 일	2019-11-11 (월)	2019-11-11 (월)	하	강경도	100%
6		Food DAO, Service, Controller 작성	1 일	2019-11-11 (월)	2019-11-11 (월)	하	김락현	100%
7		사용중 불편사항 개선	2 일	2019-11-12 (화)	2019-11-13 (수)	중	강경도, 김락현	100%
8		▲ 추가 기능	13 일	2019-11-14 (목)	2019-11-26 (화)			100%
9		요구사항 분석	2 일	2019-11-14 (목)	2019-11-15 (금)	중	강경도, 김락현	100%
10		필터링 기능 구현 계획	1 일	2019-11-16 (토)	2019-11-16 (토)	중	강경도	100%
11		이미지 검색 기능 구현 계획	2 일	2019-11-16 (토)	2019-11-17 (일)	중	김락현	100%
12		필터링 기능 추가	2 일	2019-11-17 (일)	2019-11-18 (월)	상	강경도	100%
13		기존 기능에 필터링 기능 병합	1 일	2019-11-19 (화)	2019-11-19 (화)	중	강경도	100%
14		API 테스트 코드 작성	2 일	2019-11-18 (월)	2019-11-19 (화)	상	김락현	100%
15		데이터 수집 및 테이블 작성	2 일	2019-11-20 (수)	2019-11-21 (목)	중	김락현	100%
16		게시판 기능 뷰페이지 보수	1 일	2019-11-20 (수)	2019-11-20 (수)	하	강경도	100%
17		Vue페이지 RestController에 병합	3 일	2019-11-21 (목)	2019-11-23 (토)	상	강경도	100%
18		이미지로 크롤링 기능 구현	2 일	2019-11-22 (금)	2019-11-23 (토)	중	김락현	100%
19		Drag And Drop, 이미지 업로드 기능 구현	3 일	2019-11-24 (일)	2019-11-26 (화)	상	김락현	100%
20		발표 자료 제작	2 일	2019-11-26 (화)	2019-11-27 (수)		강경도, 김락현	80%
21		팀 자료 병합	2 일	2019-11-27 (수)	2019-11-28 (목)		강경도, 김락현	30%

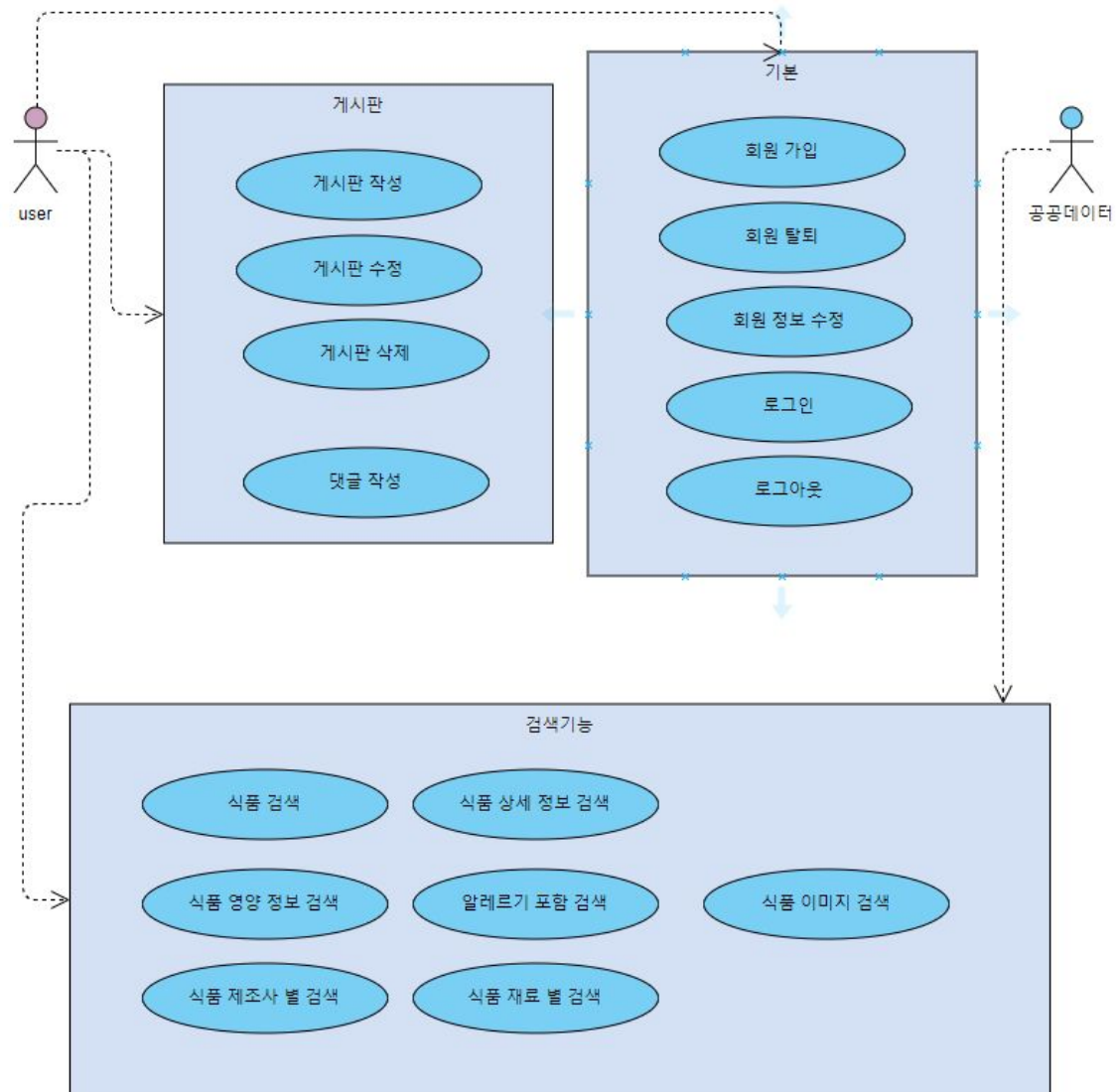
Food: F, Member: M, Board: B

ID	요구사항 명	요구사항 상세	우선순위	상태
F01	식품명 검색	식품명으로 식품을 검색하여 식품의 이미지 영양 정보, 첨가물 표기 상세검색시 영양소 함량을 그래프로 표시	필수	완료-(maven 기반)
F02	식품영양정보관리	식품에 대한 영양정보 검색	필수	완료-(maven 기반)
M01	회원관리	회원기본 정보, 알레르기 정보 CRUD	필수	완료-(maven 기반)
M02	로그인 관리	로그인/아웃	필수	완료-(maven 기반)

Food: F, Member: M, Board: B

ID	요구사항 명	요구사항 상세	우선순위	상태
F03	식품명 필터링 기능	상위 카테고리의 포함된 키워드를 체크박스 형태로 사용자가 체크한 정보만 도출 될 수 있도록 구현	추가	완료-(maven 기반)
F04	이미지 검색 기능	이미지를 드래그 앤 드롭 방식으로 검색창에 넣으면 음식 대한 조리법, 영양정보가 차트로 보여도록 구현	추가	미완-(spring boot 기반)
B01	게시판 기능	익명으로 사용자들 끼리 소통할 수 있는 게시판을 구현	추가	완료-(maven 기반)

02



02

food	
code	N/A N/A
name	N/A N/A
maker	N/A N/A
material	N/A N/A
img	N/A N/A
type	N/A N/A

member	
id	N/A N/A
password	N/A N/A
name	N/A N/A
address	N/A N/A
allergy	N/A N/A

nutrition	
food_name	N/A N/A
name	N/A N/A
supportpareat	N/A N/A
calory	N/A N/A
carbo	N/A N/A
protein	N/A N/A
fat	N/A N/A
sugar	N/A N/A
natrium	N/A N/A
chole	N/A N/A
fattyacid	N/A N/A
transfat	N/A N/A
bgn_year	N/A N/A

foodmaterial	
no	N/A N/A
name	N/A N/A
food_Material	N/A N/A
recipe	N/A N/A
img	N/A N/A

cookedfood	
code	N/A N/A
food_group	N/A N/A
name	N/A N/A
calory	N/A N/A
carbo	N/A N/A
protein	N/A N/A
fat	N/A N/A
sugar	N/A N/A
natrium	N/A N/A
chole	N/A N/A
fattyacid	N/A N/A
transfat	N/A N/A
year	N/A N/A

board	
no	N/A N/A
id	N/A N/A
contents	N/A N/A
date	N/A N/A

comment	
no	N/A N/A
board_no	N/A N/A
content	N/A N/A



```

v com.ssafy.safefood.vo
  > Food.java
  > Member.java
  v > com.ssafy.rest.api
    > FoodRelation.java
    > com.ssafy.rest.controller
    > com.ssafy.rest.repo
    > com.ssafy.rest.service
  v com.ssafy.rest.vo
    > Board.java
    > Comment.java
    > CookedFood.java
    > CrawlingFoodMaterial.java
  
```

* FoodRelation: foodmaterial, cookedfood 테이블과 관계 vo

* CrawlingFoodMaterial: vo는 foodmaterial테이블과 같음

02

FoodDAO

```
public interface FoodDAO {  
    public List<Food> findFoodByName(Food food) throws Exception;  
    public List<Food> findFoodByMaker(Food food) throws Exception;  
    public List<Food> findFoodByMaterial(Food food) throws Exception;  
    public List<Food> findFoodByNameAndMaterial(Food food) throws Exception;  
    public List<Food> findFoodByMaterialAndMaterial(Food food) throws Exception;  
    public List<Food> findFoodByMakerAndMaterial(Food food) throws Exception;  
    public Food findFoodByCodeAndMaterial(Food food) throws Exception;  
    public Food findFoodByCode(Food food) throws Exception;  
    public List<Food> findAllFood() throws Exception;  
}
```

MemberDAO

```
public public MemberDAO {  
    public void registerMember(Member member) throws Exception;  
    public void updateMember(Member member) throws Exception;  
    public void deleteMember(String id) throws Exception;  
    public Member findMemberById(Member member) throws Exception;  
    public Member findPassById(Member member) throws Exception;  
    public Object idcheck(String id) throws Exception;  
    public Member login(Member member) throws Exception;  
}
```

BoardDAO

```
public interface BoardDAO {  
    public List<Board> findAllBoard() throws Exception;  
    public List<Comment> findByNo(int no) throws Exception;  
    public Board findBoardByNo(int no) throws Exception;  
    public Board findById(String id) throws Exception;  
    public List<Board> findByContents(String contents) throws Exception;  
    public boolean addBoard(Board board) throws Exception;  
    public boolean updateBoard(Board board) throws Exception;  
    public boolean deleteBoard(int no) throws Exception;  
}
```

CommentDAO

```
public interface CommentDAO {  
    public boolean addComment(Comment comment) throws Exception;  
    public boolean updateComment(Comment comment) throws Exception;  
    public boolean deleteComment(int no) throws Exception;  
}
```

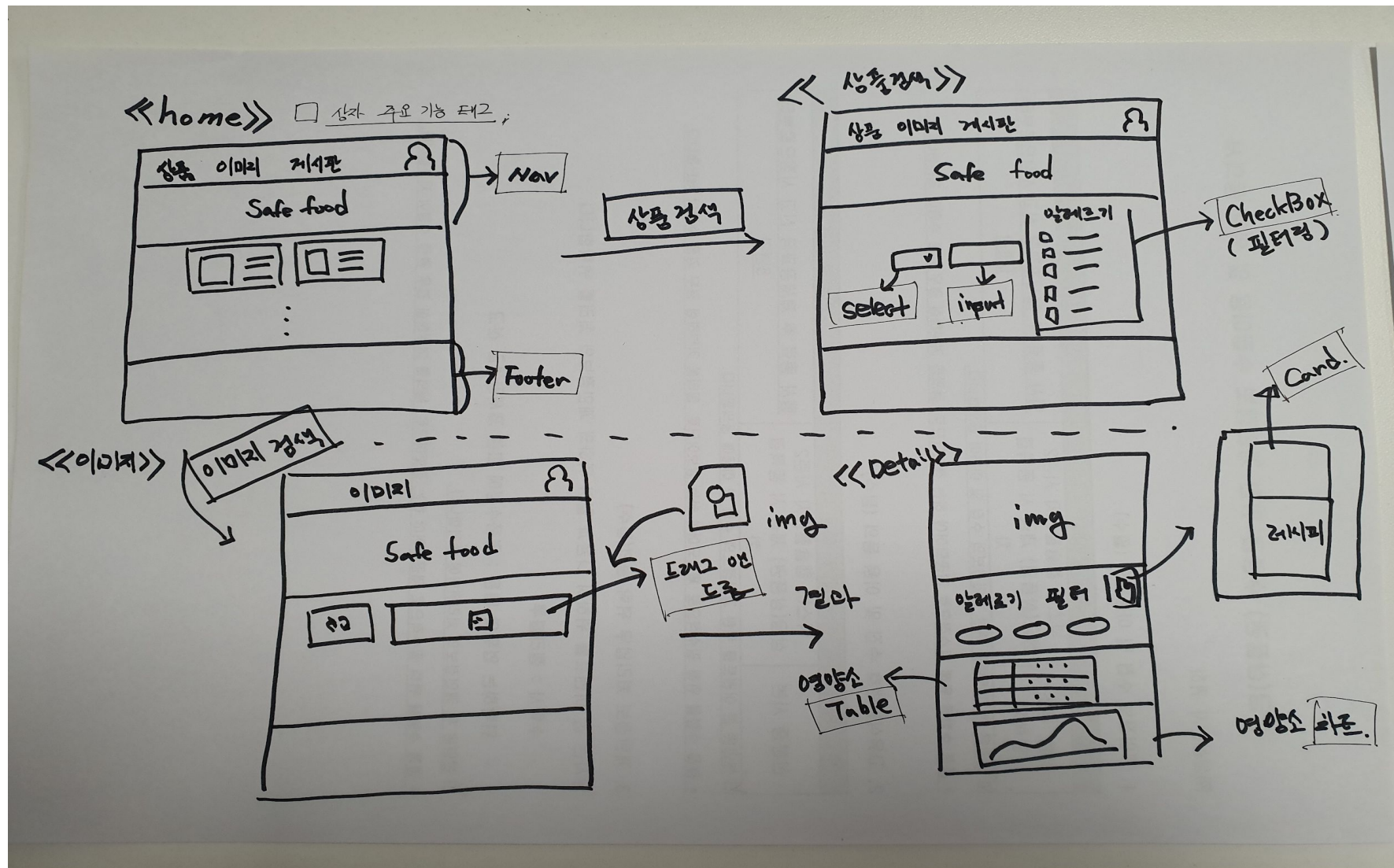
02

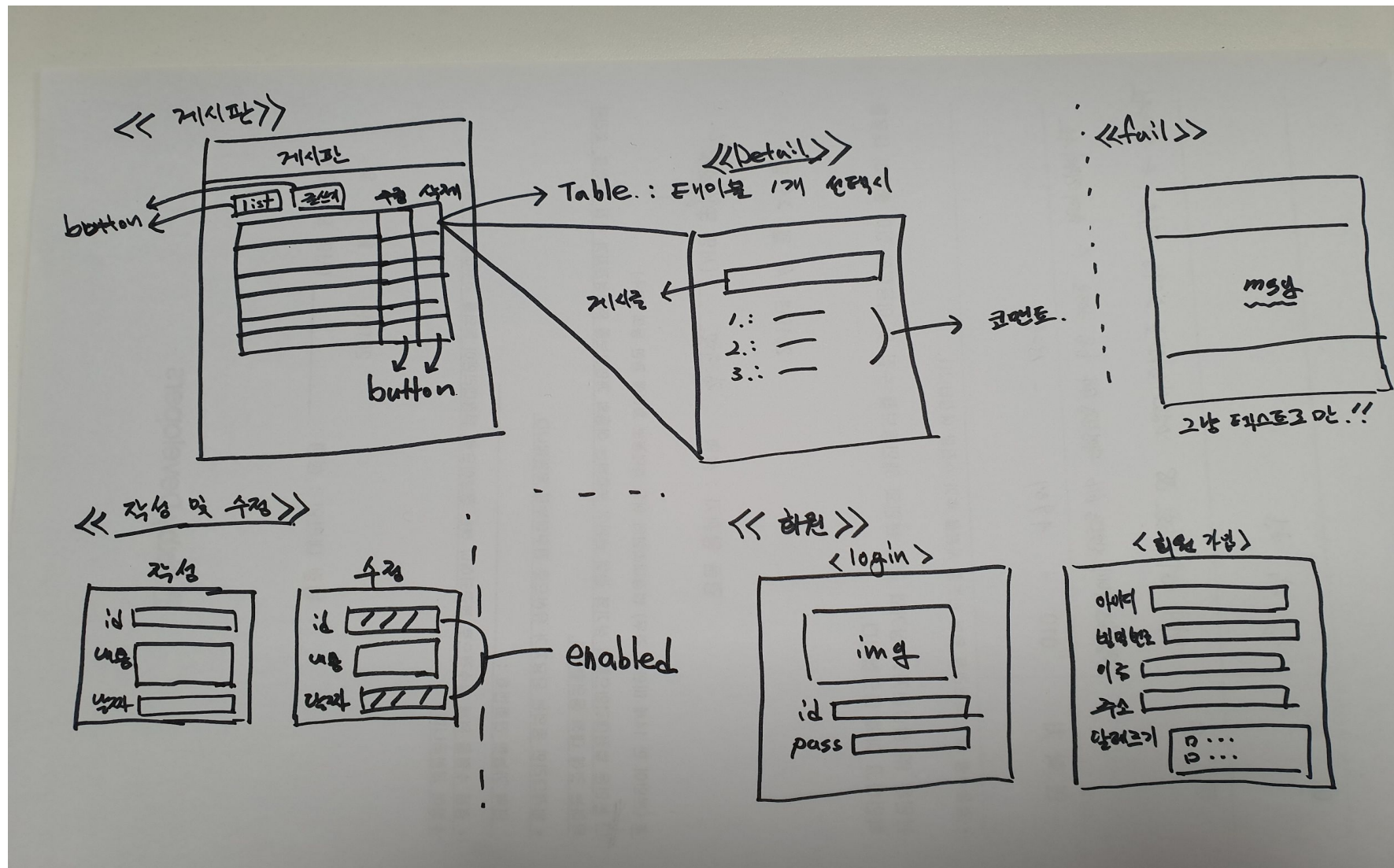
SearchFoodDAO

```
public interface SearchFoodDAO {  
    public CookedFood findNameCookedFood(String name) throws Exception;  
    public List<CrawlingFoodMaterial> findNameMaterial(String name) throws Exception;  
}
```

CrawlingFoodMaterialDAO

```
public interface CrawlingFoodMaterialDAO {  
    public boolean saveFoodMaterial(CrawlingFoodMaterial crawlingFoodMaterial)  
        throws Exception;  
}
```





MVC 기반 (회원관리, 필터링 기능)

@Controller

vo

persistence

Service

presentation



Restful API 기반 (게시판, 이미지 검색)

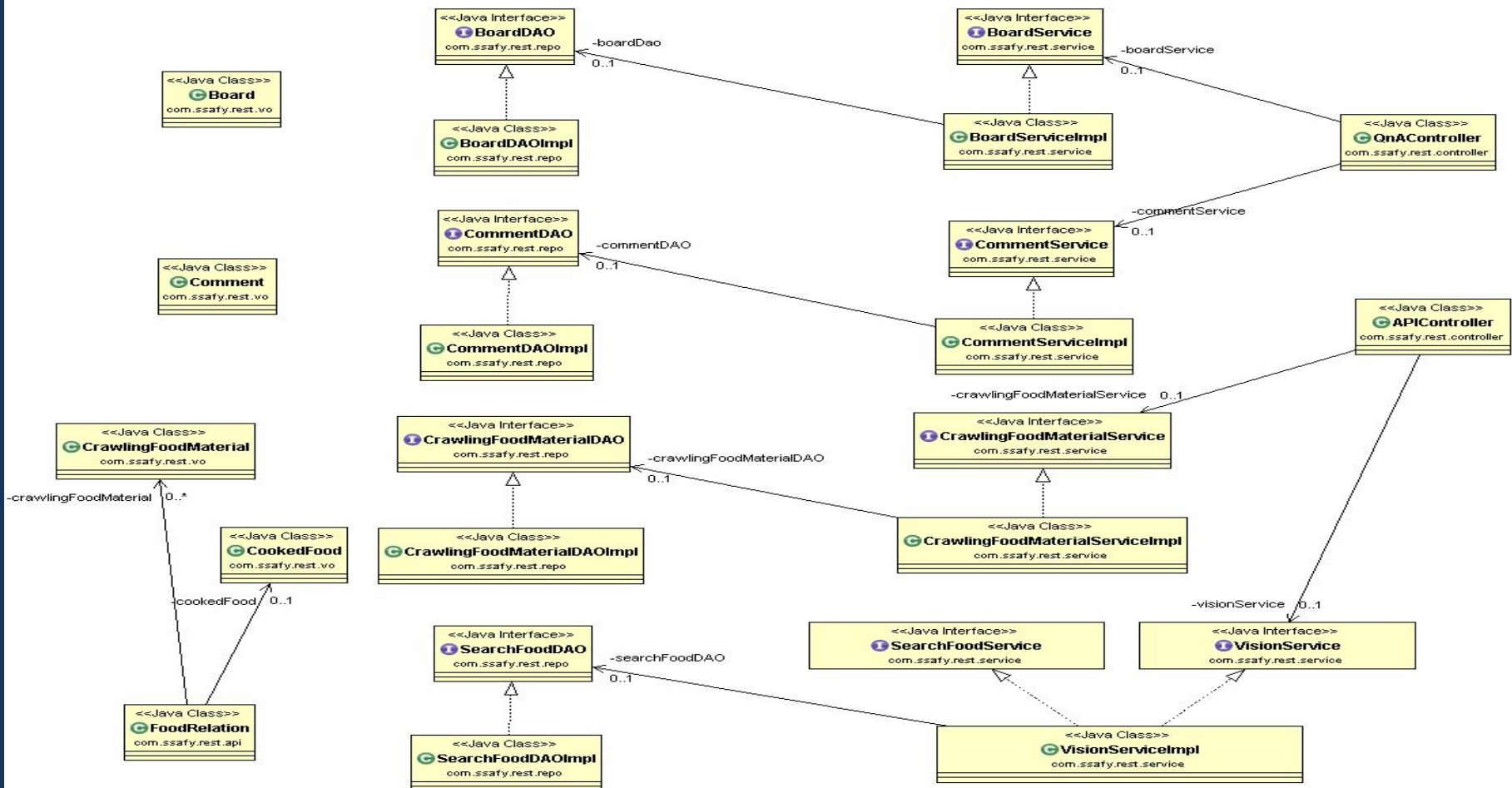
@RestController

vo

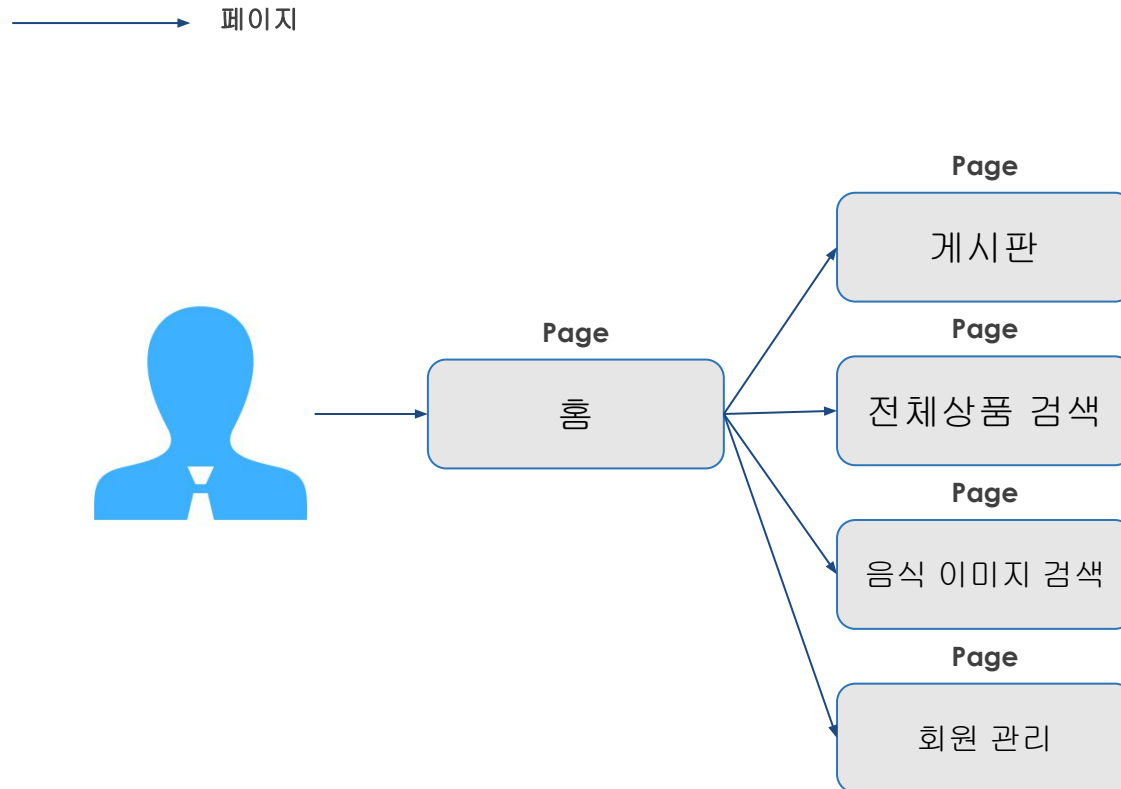
persistence

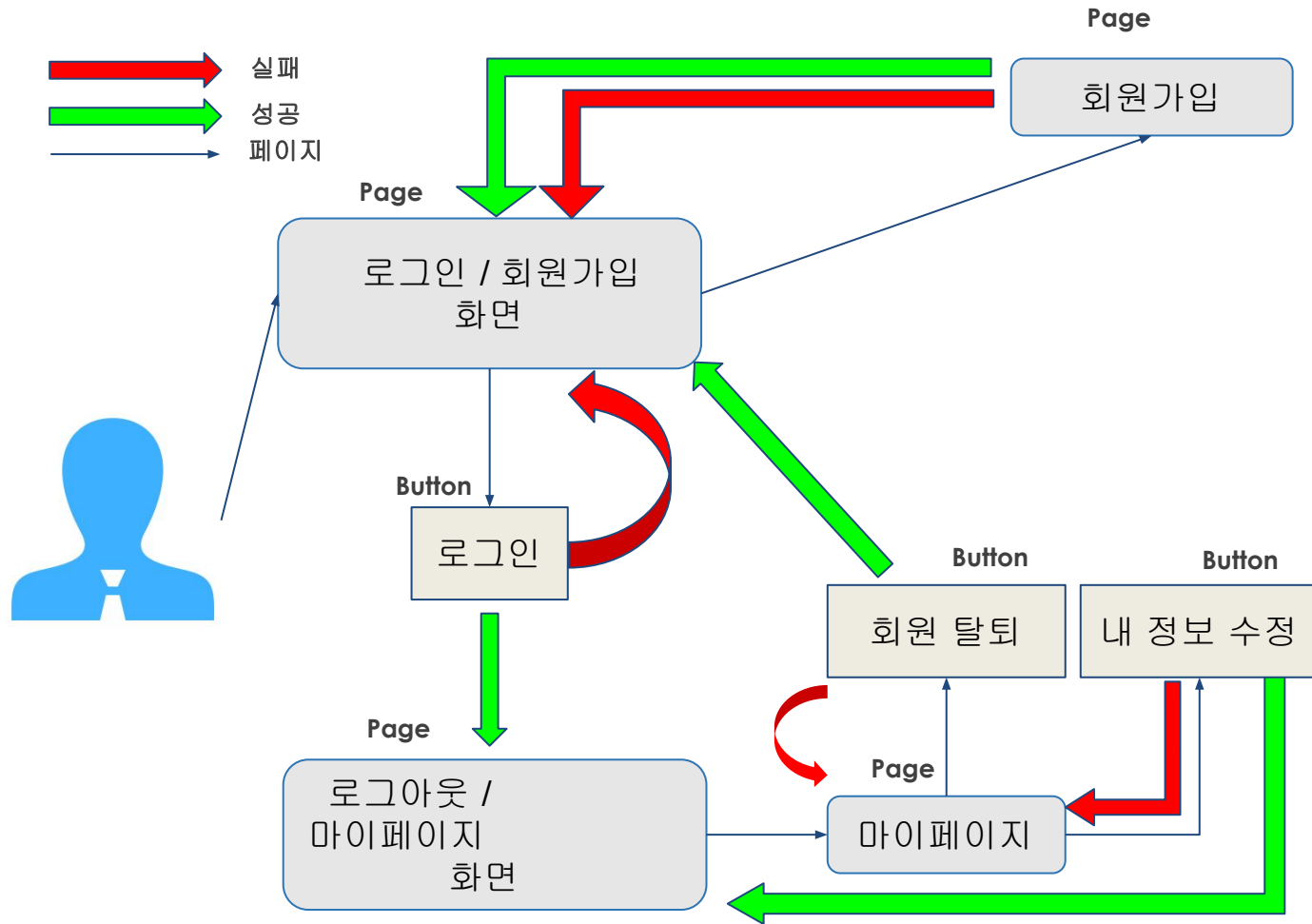
Service

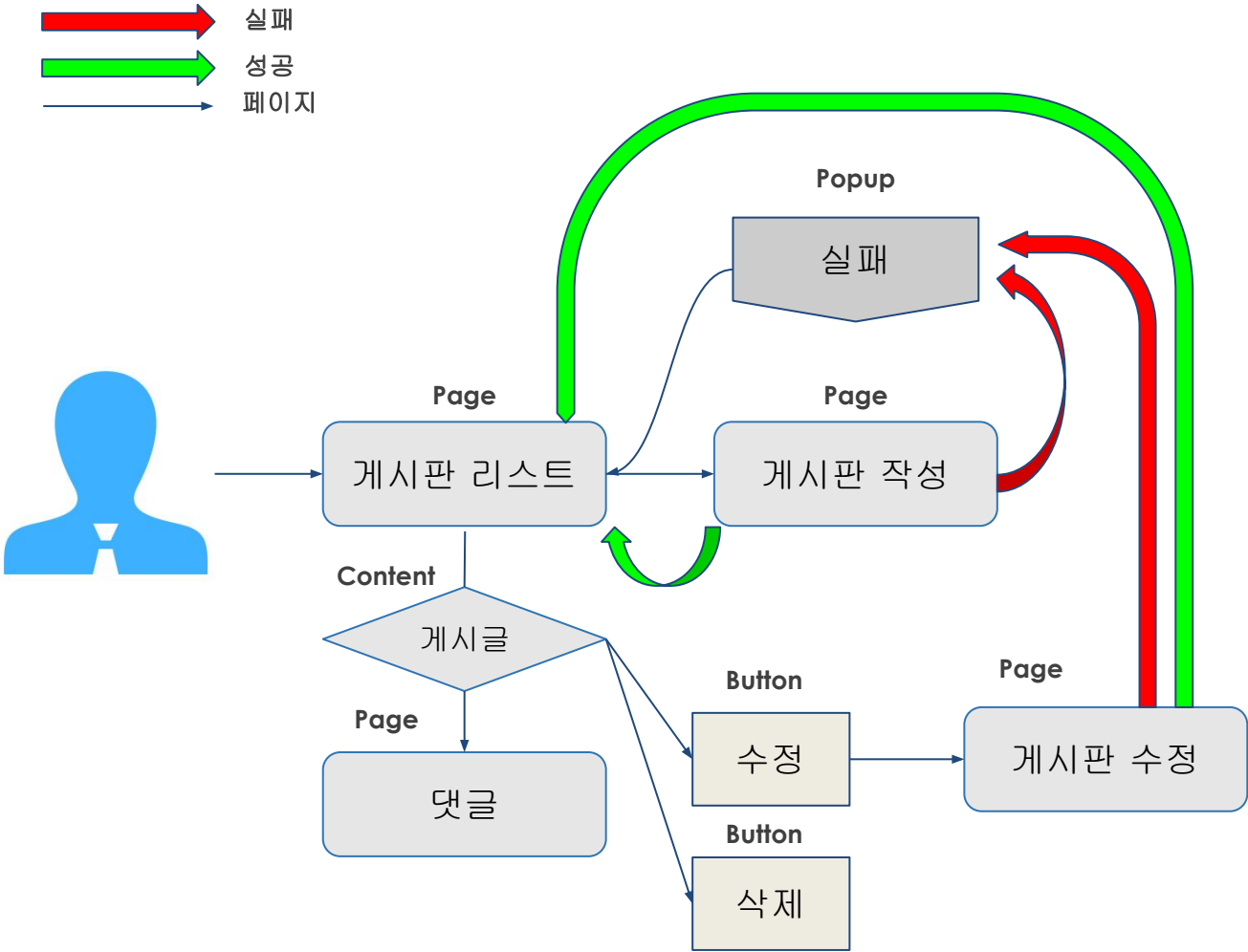
presentation

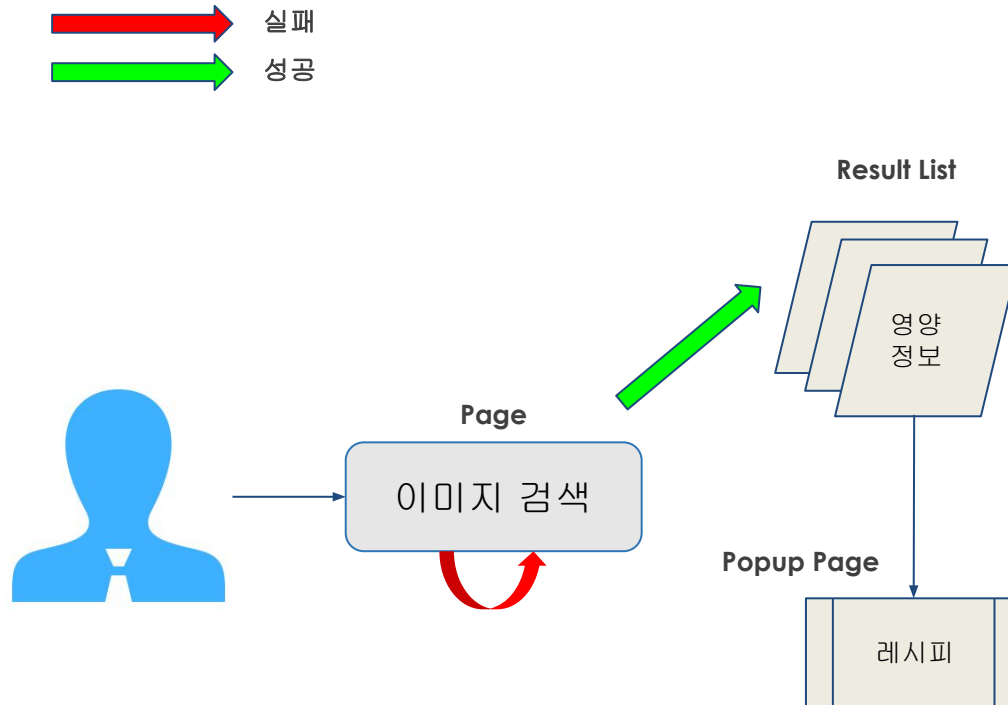


03









03

Safe Food



ID
<input type="text" value="Enter Username"/>
Password
<input type="password" value="Enter Password"/>
<input type="button" value="Login"/>
<input checked="" type="checkbox"/> Remember me

[비밀번호 찾기](#) | [회원가입](#)

전체상품정보 질문 게시판



회원가입

아이디:

아이디

중복확인

비밀번호:

비밀번호

이름:

이름

주소:

주소

알레르기

- | | | |
|------------------------------|---------------------------------|------------------------------|
| <input type="checkbox"/> 곡류 | <input type="checkbox"/> 메밀 | <input type="checkbox"/> 밀가루 |
| <input type="checkbox"/> 두류 | <input type="checkbox"/> 두유 | <input type="checkbox"/> 두부 |
| <input type="checkbox"/> 과일 | <input type="checkbox"/> 복숭아 | <input type="checkbox"/> 귤 |
| <input type="checkbox"/> 야채 | <input type="checkbox"/> 토마토 | <input type="checkbox"/> 오렌지 |
| <input type="checkbox"/> 육류 | <input type="checkbox"/> 돼지고기 | <input type="checkbox"/> 소고기 |
| <input type="checkbox"/> 난류 | <input type="checkbox"/> 계란 | <input type="checkbox"/> 닭고기 |
| <input type="checkbox"/> 우유류 | <input type="checkbox"/> 우유 | |
| <input type="checkbox"/> 어류 | <input type="checkbox"/> 등푸른 생선 | |
| <input type="checkbox"/> 갑각류 | <input type="checkbox"/> 게 | <input type="checkbox"/> 조개 |
| <input type="checkbox"/> 패류 | <input type="checkbox"/> 새우 | |
| <input type="checkbox"/> 견과류 | <input type="checkbox"/> 땅콩 | <input type="checkbox"/> 호두 |

가입하기

전체상품정보 질문 게시판



마이 페이지

아이디:

over210m

비밀번호:

123

이름:

김락현

주소:

경상북도 구미시 황상동 175-26

알레르기

- | | |
|-----|---|
| 곡류 | <input type="checkbox"/> 메밀 <input checked="" type="checkbox"/> 밀가루 |
| 두류 | <input type="checkbox"/> 두유 <input checked="" type="checkbox"/> 두부 |
| 과일 | <input type="checkbox"/> 복숭아 <input type="checkbox"/> 귤 <input type="checkbox"/> 오렌지 |
| 야채 | <input type="checkbox"/> 토마토 |
| 육류 | <input type="checkbox"/> 돼지고기 <input type="checkbox"/> 소고기 <input type="checkbox"/> 닭고기 |
| 난류 | <input type="checkbox"/> 계란 |
| 우유류 | <input type="checkbox"/> 우유 |
| 어류 | <input type="checkbox"/> 등푸른 생선 |
| 갑각류 | <input type="checkbox"/> 게 <input type="checkbox"/> 조개 |
| 패류 | <input type="checkbox"/> 새우 |
| 견과류 | <input type="checkbox"/> 땅콩 <input type="checkbox"/> 호두 |

내 정보 수정하기

회원 탈퇴하기

03

[리스트 보기](#) | [QnA 작성](#)

작성자	<input type="text"/>
내용	<input type="text"/>
작성날짜	<input type="text" value="연도-월-일"/>
<input type="button" value="글쓰기"/>	

BOARD LIST

[리스트 보기](#) | [QnA 작성](#)

번호	작성자	내용	작성날짜	수정	삭제	댓글보기
10	AAA	AAAAAA	2019-11-24	<input type="button" value="수정"/>	<input type="button" value="삭제"/>	<input type="button" value="댓글보기"/>
11	BBB	BBBBBBB	2019-11-28	<input type="button" value="수정"/>	<input type="button" value="삭제"/>	<input type="button" value="댓글보기"/>
12	CCC	CCCCCCC	2019-11-26	<input type="button" value="수정"/>	<input type="button" value="삭제"/>	<input type="button" value="댓글보기"/>
13	DDD	DDDDDDD	2019-11-26	<input type="button" value="수정"/>	<input type="button" value="삭제"/>	<input type="button" value="댓글보기"/>
14	EEE	EEEEEEE	2019-11-27	<input type="button" value="수정"/>	<input type="button" value="삭제"/>	<input type="button" value="댓글보기"/>

03

BOARD LIST

[리스트 보기](#)[QnA 작성](#)

번호	작성자	내용	작성일짜
10	AAA	AAAAAA	2019-11-24

10번 글의 댓글 리스트

1 :: aaaa

2 :: 11111

3 :: 222333

4 :: 44444

5 :: 55555

6 :: 123423

[댓글달기](#)

03

알레르기 위험식품 검색

제품 이름

라면

곡류

두류

과일

야채

육류

난류

우유류

어류

갑각류

패류

견과류

☐ 메밀 ☐ 밀가루☐ 두유 ☐ 두부☐ 복숭아 ☐ 귤 ☐ 오렌지☐ 토마토☒ 돼지고기 ☐ 소고기 ☐ 닭고기☒ 계란☒ 우유☐ 등푸른 생선☐ 게 ☐ 조개☐ 새우☒ 땅콩 ☐ 호두

상품



진라면컵매운맛

참맛매운육수분말,소맥분(미국산,호주산),변성전분,포도당,발효복합분,팜유(말레이시아산),진한감칠맛분,글루텐,숙성마늘맛분,정제소금,마늘양념분말,유화유지,난각분말,복음마늘분,치킨추출농축액,백설탕,마늘시즈닝,간장분말,야채추출물,식용유지가공품,면류첨가알칼리제(산도조절제),정제소금,비타민 B₂,호모조미분,칠리맛분말,녹차풍미유,전분,쇠고기간장분말,쇠고기향미분말,후추분말,고춧가루,구아검,조미아미노산분말,향미증진제,칠리추출물,고추농축소스,산도조절제,건파,조미쇠고기맛후레이크,건당근,건표고버섯,건조채심,건고추입자,[소맥분(밀),탈지대두(대두),유당(우유),난각분말(계란),땅콩단백(땅콩),돈굴(돼지고기)]



큰컵삼양라면

복음면조미분말,소맥분(미국산,호주산),부대찌개베이스,전분(태국산),팜유,설탕,정제염,정제염,맛베이스에스,미감에스유,식물성간장분말,면류첨가알칼리제(산도조절제),감자전분,양파엑기스,양념간장분,구아검,지미맛분말,비타민B₂,구연산,햄풍미분말,육수맛분말,녹차풍미유,김치조미분말,돈육풍미분말,복음양념분말,훈합양념분말,진사골추출물분말,전지분유,팜유,부대찌개분말,쇠고기양념분말,저염풍미분에스,후추분,향미증진제,매운조미고추맛분,김치찌개풍미분말,베이컨향분말,햄베이스시즈닝,파프리카추출색소,마늘맛오일,조미콩단백,실당근,건청파,청경채,고단백건조유부,[소맥분(밀),탈지대두(대두),돼지고기,전지분유(우유),계란,땅콩]

추가

찜하기

추가

찜하기

03

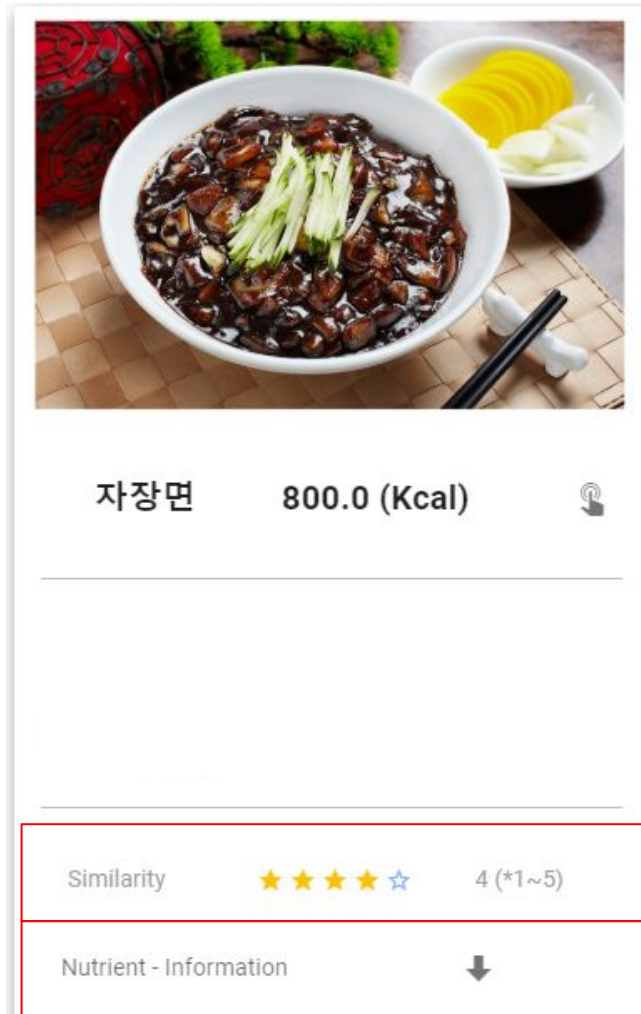
드래그 앤 드롭 방식



```
<v-card
  :elevation="this.degree2"
  id="drop-zone"
  v-bind:class="[isDragging?'drag-over':'']"
  v-on:dragover="isDragging=true"
  v-on:dragenter="isDragging=true"
  v-on:dragleave="isDragging=false"
  @mouseover="changeDegreeIn2()"
  @mouseout="changeDegreeOut2()"
>
  <div class="text-wrap">
    <div v-if="this.imageList.length == 0">
      <div>
        
      </div>
    </div>
    <div v-if="this.imageList.length > 0">
      <v-progress-circular :width="4" indeterminate color="green" v-if="this.loading"></v-progress-circular>
      <div v-else>
        
      </div>
    </div>
  </div>
  <div>
    <input id="uploadInput" name="uploadInput" type="file" @change="onChange" multiple />
  </div>
</v-card>
```

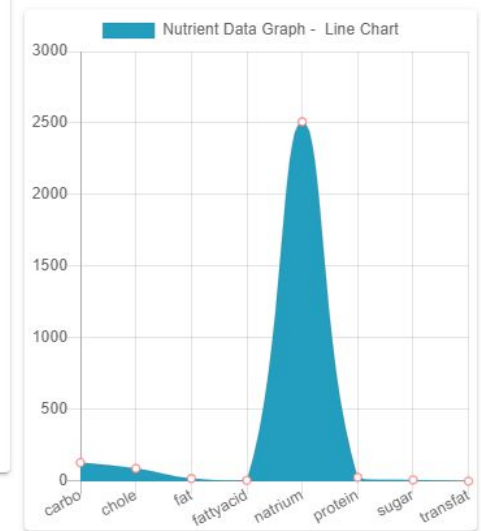
1. 드래그 앤 드롭 방식으로 이미지를 사진 영역에 가져다 두면 onChange이벤트로 이미지 파일을 서버로 보냄
2. axios로 통신

03



자장면 단어가 들어간 영양성분	
Supportpareat:	675.0 (g)
Carbo:	130.0 (g)
Chole:	89.0 (g)
Fat:	19.0 (g)
Fattyacid:	6.0 (g)
Natrium:	2510.0 (g)
Protein:	27.0 (g)
Sugar:	9.0 (g)
Transfat:	0.0 (g)

[그림1] 영양소 테이블



[그림2] 영양소 그래프

-> 사용자가 입력한 사진과 얼마나 유사한지 1~5 점으로 표시

-> 음식의 영양소 정보를 [그림1]과 [그림2] 각각 표현

03



자장면

800.0 (Kcal)



연관된 단어에 대한 레시피 정보

Similarity



4 (*1~5)

Nutrient - Information



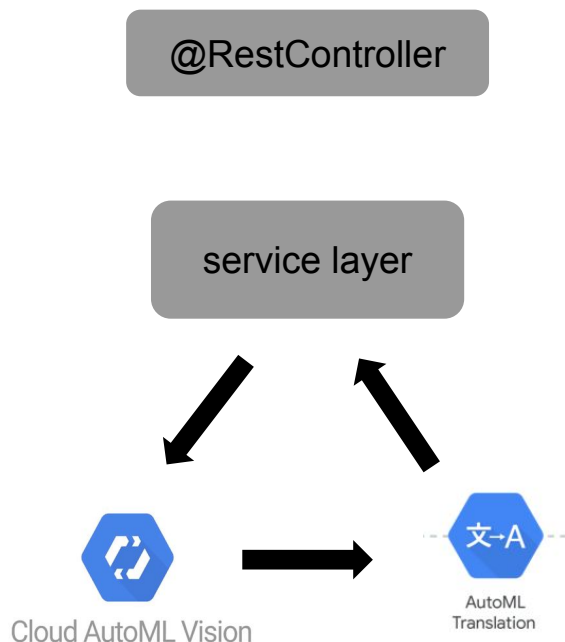
채소 자장면

* 재료

중화면 200g, 소스 감자 80g(1/2개), 양파 80g(1/3개), 애호박 30g(1/2개), 검은콩 2g(1/2작은술), 새송이버섯 30g(3개), 옥수수통조림 10g(2작은술), 설탕 20g(1½큰술), 저염간장 15g(1큰술), 자장분말 30g(2큰술), 녹말가루 30g(2큰술), 장식 오이 30g(1/5개), 완두콩 10g(2작은술)

* 만드는 법

1. 검은콩은 물에 담가 2시간 정도 불린다.a
2. 완두콩은 끓는 물에 데친다.
3. 감자, 양파, 애호박은 1.5cm로 깍둑썰다.b
4. 믹서에 불린 검은콩을 간다.
5. 면은 끓는 물에 삶아 체에 건져 물기를 뺀다.c
6. 새송이버섯은 슬라이스하고 오이는 채 썰다.
7. 팬에 썰어둔 3의 채소를 볶다가 저염간장, 자장분말, 설탕을 넣어 소스를 만든다
8. 소스에 검은콩, 옥수수를 넣어 끓인다.



1. 구글 비전 api에서 입력한 사진 분석(딥러닝) 음식의 텍스트 추출
2. 텍스트를 구글 번역 api 사용으로 한국어로 번역
3. 부분적, 완전, 유사한 사진들의 웹 주소 각각 10개를 가져옴

```

WebDetection annotation = res.getWebDetection();
out.println("Entity:Id:Score");
out.println("=====");
for (WebEntity entity : annotation.getWebEntitiesList()) {
    String desc = autoDetectTranslate(entity.getDescription(), "ko");
    if (desc == null || desc == "" || desc == " ") continue;
    description.add(desc);
    rating.add((float) Math.round(entity.getScore() + 2.5f));
    out.println(desc + "==" + entity.getDescription() + " : " + entity.getEntityId() + " : "
        + entity.getScore() + "==" + (float) Math.round(entity.getScore() + 2.5f));
}

for (WebLabel label : annotation.getBestGuessLabelsList()) {
    bestGuessLabel = label.getLabel();
    out.format("\nBest guess label: %s", bestGuessLabel);
}

out.println("\nPages with matching images: Score\n==");
for (WebPage page : annotation.getPagesWithMatchingImagesList()) {
    out.println(page.getUrl() + " : " + page.getScore());
}

out.println("\nPages with partially matching images: Score\n==");
for (WebImage image : annotation.getPartialMatchingImagesList()) {
    out.println(image.getUrl() + " : " + image.getScore());
}

out.println("\nPages with fully matching images: Score\n==");
for (WebImage image : annotation.getFullMatchingImagesList()) {
    out.println(image.getUrl() + " : " + image.getScore());
}

out.println("\nPages with visually similar images: Score\n==");
int count = 0;
for (WebImage image : annotation.getVisuallySimilarImagesList()) {
    if (count == 0) {
        getVisuallySimilarImages = image.getUrl();
    }
    out.println(image.getUrl() + " : " + image.getScore());
    count++;
}
  
```

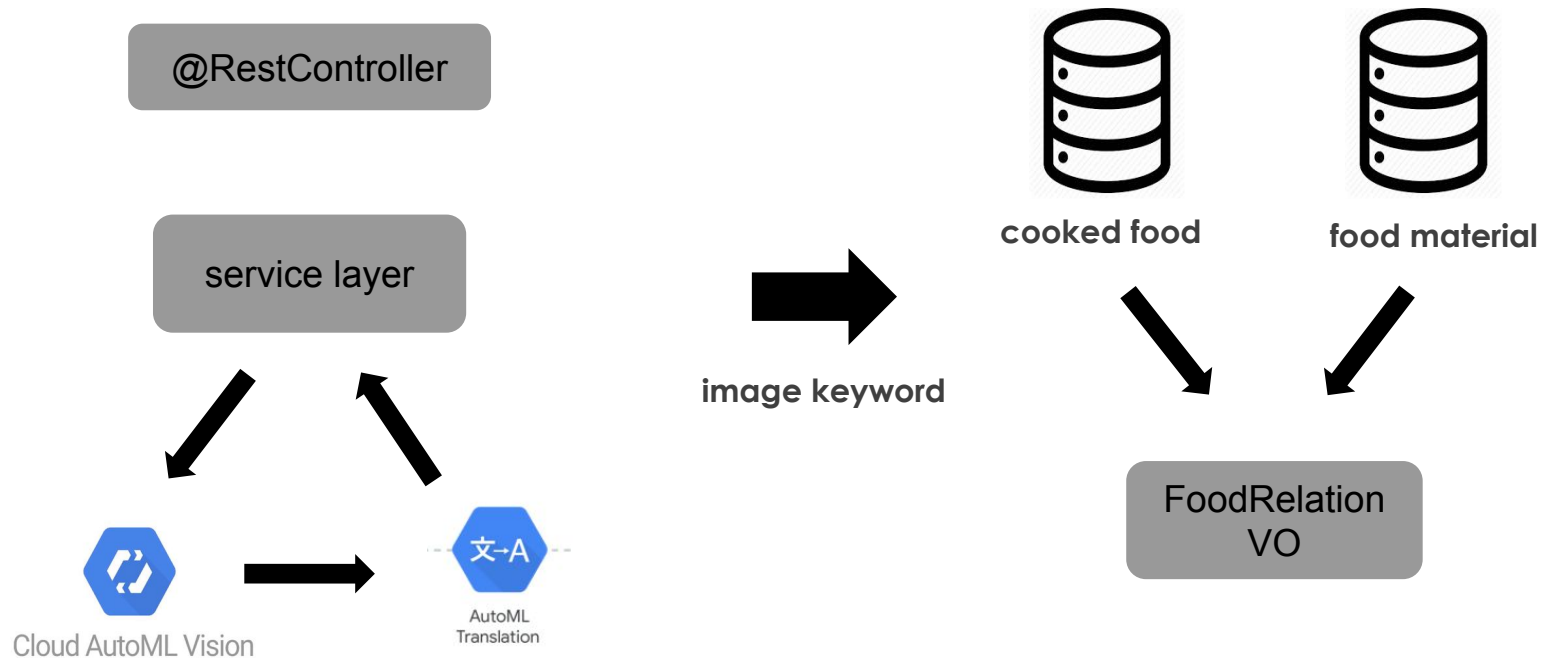
best 키워드

부분적으로

완전하게

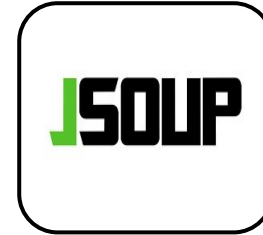
유사한

03



1. 분석된 이미지에 대한 키워드로 음식 영양소 table, 음식 재료 table에서 검색함
2. 각 table 관계는 없음 하지만 FoodRelationVO가 두 table 정보를 담음

03



이미지 없음

자장면 800.0 (Kcal)



Similarity



4 (*1~5)

Nutrient - Information



```
Document doc = Jsoup.connect("웹사이트 주소" + URLEncoder.encode(desc, "UTF-8")).get();  
Elements element = doc.select("div.recipe_list");  
Element img = element.get(0).select("img").get(0);  
String resultImgUrl = img.getElementsByAttribute("src").attr("src");
```

1. JSOUP 크롤링 라이브러리 사용
2. 해당사이트에 GET 방식으로 keyword 붙여서 검색
3. 결과로 해당 사이트 웹 Document 객체
4. keyword가 한글일 경우

URLEncoder.encode("keyword","UTF-8") 반드시 작성

04

1. 회원의 알레르기 정보를 검색 결과에 이용하지 못함

-> 회원 비회원 기능을 구분하여 회원 정보 이용을 요구사항에 포함 하여 설계로 가는 것 이 중요

2. 음식 사진분석을 API 활용만으로 부족함(특정 음식 이미지는 원하는 키워드를 추출하기 어려움)

-> 구글의 AutoML를 이요한 커스텀 머신러닝 이용하여 음식 이미지 데이터에 대해 추가적인 학습이 필요함

3. 전체적인 UI 도출을 하고 나서 세부적인 UI 설계 부족

-> 기능별 UI 설계에서 세부적인 UI 설계 포함

4. 검색어 필터링 기능이 알레르기에 한정적

-> 칼로리 수치 구간별, 영양 성분 수치 구간별 등 다양한 방면에서 검색 기능을 생각해야 함

1. Google Cloud Platform

-> 검색엔진API, 인공지능 관련 API, 컴퓨트 엔진등 통합 클라우드 서비스를 제공함

-> 그 중 인공지능 관련 vision API 사용으로 이미지 분석 기반을 알게됨

2. NoClassDefFoundError

-> .class파일을 동적으로 로드 하는 상황에서 사용하고자 하는 .class파일이 없을 때 발생

-> pom.xml에서 라이브러리 추가시 현재 버전이 있거나 버전에 맞지 않는 클래스를 사용할 때

3. jsoup 크롤링 라이브러리

-> 크롤링 하려는 사이트 url에 get 또는 post방식으로 요청하면 데이터를 Document 객체로 받아

편리하게 특정 태그, 속성, 값을 추출 할 수 있음

THANK
YOU