

- 식단 데이터.xml
- 식단 데이터.json

2) 개발언어 및 툴

- Java / STS

3) 필수 라이브러리 / 오픈소스

- Jackson

3. 작업 순서

- 1) 팀원과 같이 요구사항(기본/추가/심화)을 검토하고, 클래스 다이어그램을 작성한다.
- 2) 기본 기능의 요구사항을 이해하고 비워진 부분 또는 오류 발생 부분을 구현하여 완성한다.
- 3) 기본 기능 구현을 모두 완료하고, 추가 기능, 심화 기능을 구현한다.

4. 요구사항

식단 분석을 활용한 개인 맞춤형 건강 코칭 서비스(이하 "남남코치"라 한다.)를 구현하려고 한다. 남남코치에서는 식품의약품안전처에서 제공한 음식 별 영양성분 데이터를 활용하여 작성한 식단을 사용자가 원하는 목표(운동, 질병 등)에 따라 분석하여 정보를 제공한다. 또한 건강 목표를 위한 챌린지, 식단 리뷰 및 다양한 정보를 교류할 수 있는 커뮤니티 등 다양한 기능을 지원한다. 팀원과 상의 후 요구사항을 만족할 수 있도록 요구 사항 명세서를 작성 및 구현해보자.

이번 관통 프로젝트는 '남남코치' 프로젝트의 기능을 Java Project의 Swing을 이용하여 화면을 구성하고 기능이 동작하도록 구현한다.

아래의 참고용 요구사항 예시를 활용하여 보다 개선된 프로젝트의 요구사항을

정리하고 분석하여 요구사항 기능 명세서를 작성해보자. (아래 필수 기능 포함)

● 요구사항 예시(참고용)

- 시장 조사를 통하여 '남남코치' 프로젝트의 요구사항을 완성해보자
- 아래의 내용을 참고하여 추가적인 아이디어에 대해 요구사항을 추가 또는 수정하여 기능을 구현한다. 단, 필수 기능은 구현해야 한다.
(ex. 컬럼의 내용을 추가할 수도 있음, 요청 URL 등)

번호	분류	요구사항명	요구사항 상세	우선순위
기능적 요구사항				
F01	식단	식단 작성	섭취한 음식을 DB에서 선택하여 식단기록을 작성하는 기능	필수
F02	식단	식단 조회	식단 ID를 활용하여 식단기록 내용을 상세하게 조회하는 기능	필수
F03	식단	식단 수정	작성한 식단기록의 내용을 수정하는 기능	필수
F04	식단	식단 삭제	작성한 식단기록을 삭제하는 기능	필수
F05	식단	식단 분석	입력한 식단 기록을 통해 (공식 or AI) 영양 정보를 분석하여 표현하는 기능	필수
F06	회원	회원 작성	회원 가입을 통해 User의 정보를 등록하는 기능 (개인 프로필 - 키, 몸무게, 질환 등을 관리)	필수
F07	회원	회원 조회	회원 정보를 조회하는 기능	필수
F08	회원	회원 수정	회원 정보를 수정하는 기능	필수
F09	회원	회원 삭제	회원 정보를 삭제하는 기능 (삭제 대신 '비활성화'로 상태를 바꿀 수도 있음)	필수
F10	회원	로그인/로그아웃	로그인/로그아웃 기능	필수
F11	회원	팔로우/팔로잉	팔로우 추가, 취소, 목록 조회할 수 있는 기능	추가

F12	챌린지	챌린지 정보관리	챌린지 정보 CRUD (기간, 설명, 이미지 등 관리)	추가
F13	챌린지	챌린지 사용자참여	사용자의 챌린지 참여, 달성도 등 정보 관리	추가
F14	커뮤니티	게시판	식단 리뷰, 전문가, 자유 게시판 글 작성, 조회, 수정, 삭제 기능	추가
F15	커뮤니티	댓글	게시글에 댓글 작성, 조회, 수정, 삭제 기능	추가
F16	AI	AI 식단 분석	생성형 AI를 활용한 식단 분석 기능	심화
F17	AI	AI 운동 코칭	생성형 AI를 활용한 운동 코칭 기능	심화
...
비 기능적 요구사항				
NF01	UX	사용자 편의성	서비스에 대한 사전지식이 없어도 사용하기 편해야 함.	-
NF02	UI	가용성	언제나 (어떤 디바이스로든) 서비스 가능해야 함.	-
NF03	효율성	응답성	조회에 대한 결과를 빠르게 응답해야 함.	-
NF04	효율성	정확성	신뢰성 있는 데이터를 사용하여 분석 결과 등의 내용이 정확해야 함.	-
...

1) 기본(필수) 기능

‘남남코치’ 프로젝트의 식단 기록을 조회할 수 있는 기능을 구축하려고 한다. 먼저 주어진 다양한 타입의 데이터를 파싱하여 객체화 시켜보자.

● 식단기록 데이터 분석

- 식단 기록 정보가 식단 ID, 날짜, 식사구분(아침/점심/저녁/간식), 음식(식품코드, 식품명, 에너지, 탄수화물, 단백질, 지방, 식품중량)의 정보가 XML, JSON 파일로 제공된다.
- 식단 기록 정보 XML 문서를 파싱하여 객체화 하고,

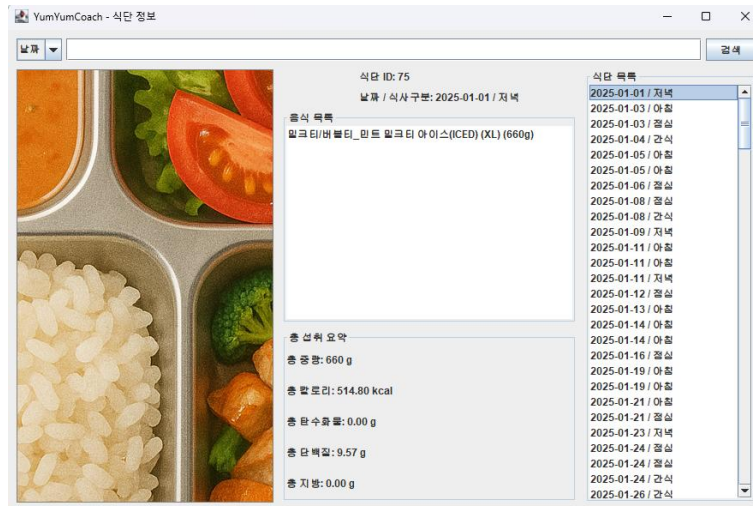
검색기능(날짜/음식명)을 제공한다.

● 요청 조건

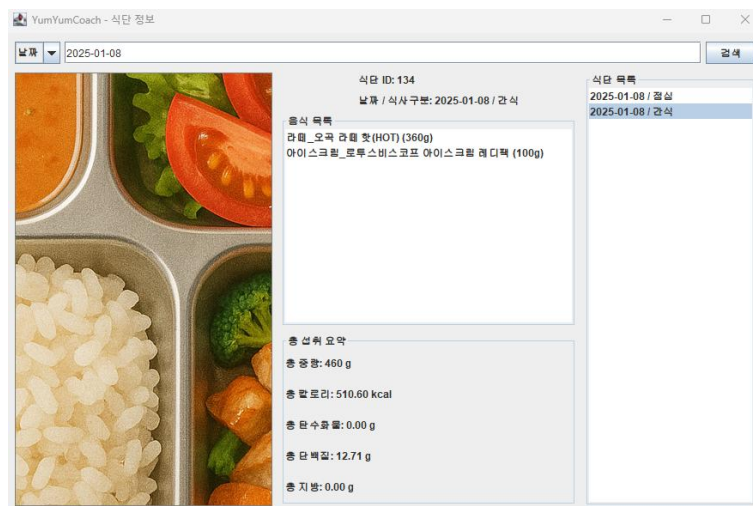
- 식단데이터.xml을 파싱하여 DietLogDto 객체에 저장한다.
- XML 파싱 한 식단 기록 목록을 화면에 표시한다.
- 식단 기록 클릭 시 상세 내용이 화면에 출력되어야 한다.
- 식단 기록을 검색(날짜/음식명)할 수 있게 한다.

● 구현해야 할 부분

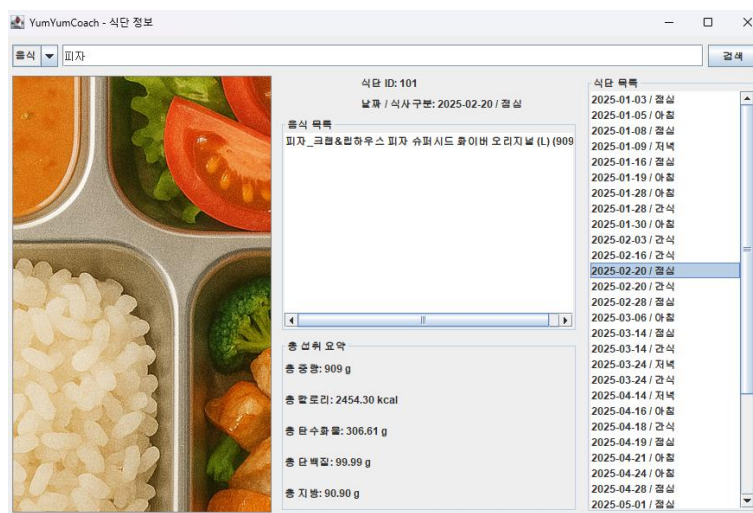
#번호	Package & Class	구현 내용	비고
01	com.ssafy.yummyumcoach.view. CoachInfoView.java	searchBtn 에 addActionListener 을 이용하여 검색 기능 처리 기능 완성하기	166 Line
02	com.ssafy. yummyumcoach.model.service. DietLogServiceImpl.java	search()를 Dao Layer 의 method 를 호출하도록 완성하세요.	56 Line
03	com.ssafy. yummyumcoach.model.dao. DietLogDaoImpl.java	검색기능을 완성하세요.	45 Line
04~05	com.ssafy. yummyumcoach.util. DietDataSAXHandler.java	코드를 완성하세요.	51 Line ~
06	com.ssafy. yummyumcoach.util. DietDataSAXParser.java	코드를 완성하세요.	43 Line



[첫 실행 화면 예시]



[날짜를 이용한 검색 화면]



[음식("피자")를 검색한 화면]

2) 추가 기능

위 필수 기능을 모두 구현했다면 추가 기능을 구현해보자.

- 음식 DB 활용 분석

- res/음식DB.csv 파일을 활용하여 실제 음식 정보를 활용할 수 있도록 사용한다.
- 음식을 검색하여 관리할 수 있도록 작성한다.
- 별도 UI를 가지는 Application을 통해 보여줄 수 있다.

- 식단 기록 심화

- 음식 DB를 활용하여 직접 식단 기록 CRUD를 할 수 있는 기능을 완성한다.
- 식단 정보를 활용하여 식단에 대한 점수를 부여하는 등 분석하여 활용할 수 있다.
- 별도 UI를 가지는 Application을 통해 보여줄 수 있다.

3) 심화 기능

- 그 외 프로젝트의 다양한 기능을 위해서 타 사이트를 벤치마킹 하거나, 각 팀별 새로운 아이디어를 추가하여 프로젝트를 확장해보자
- 생성형 AI 활용 방안: OpenAI 등 생성형 AI를 활용할 수 있는 방법에 대해서 팀원과 이야기를 해보고, 추가할 수 있는 기능을 정리하여 실제 프로젝트에 적용 및 구현해보자.
- 적용할 심화 기능 항목을 선정하면, 요구사항을 정리해보자

5. 참고자료

- Class Diagram 참고예시

	목록 선택 이벤트처리
Main	프로그램 시작 지점
YumYumCoachException	프로젝트 전역 커스텀 예외

XML 데이터 구조

```
<?xml version="1.0" encoding="UTF-8"?>
<식단데이터>
  <식단>
    <식단ID>1</식단ID>
    <날짜>2025-01-29</날짜>
    <식사구분>아침</식사구분>
    <음식>
      <식품>
        <식품코드>D101-004160000-0001</식품코드>
        <식품명>국밥_돼지머리</식품명>
        <에너지>1233.0</에너지>
        <탄수화물>143.46</탄수화물>
        <단백질>60.30</단백질>
        <지방>46.44</지방>
        <식품중량>900g</식품중량>
      </식품>
    </음식>
  </식단>
  <식단>
    <식단ID>2</식단ID>
    <날짜>2025-04-07</날짜>
    <식사구분>저녁</식사구분>
    <음식>
      <식품>
        <식품코드>D101-004200000-0001</식품코드>
        <식품명>국밥_순대</식품명>
        <에너지>884.8</에너지>
        <탄수화물>121.6</탄수화물>
        <단백질>36.32</단백질>
        <지방>35.20</지방>
        <식품중량>700g</식품중량>
      </식품>
    </음식>
  </식단>
  <!-- 이하 생략... -->
</식단데이터>
```

6. 결과

프로젝트 최종적으로 제출해야 할 항목은 아래와 같고, GIT의 README.md파일이나 워드, 파워포인트 문서를 활용하여 작성하고, GitLab에 업로드 한다.

● 산출물과 제출

- 클래스 다이어그램(+ 개선사항 정리 내용)

- README.md
- 구현 소스코드 및 실행 화면 캡처본
- 남남코치_Java_지역_반_성명1_성명2.zip으로 제출

7. 채점 기준

난이도	구현 기능	점수	비고
기본	화면 UI + Service + Dao 완성	30	이벤트 처리 - 10 Service 완성 - 10 Dao 완성 - 10
	XML Parsing	20	SAXParser, SAXHandler 완성
	전체 프로그램 동작	10	오류 없이 전체 동작
추가	음식DB.csv 관리	10	파일 내용 화면
	식단 기록 CRUD	10	
심화	생성형 AI 적용 기능/아이디어	10	
	팀별 아이디어	10	타 사이트 벤치마킹 또는 새로운 아이디어 구현