



2차 스프린트

| 염윤호 고민표 박희수 이재원 이창훈

목차

01 프로젝트 개요

- 1-1. Selfit 소개
- 1-2. 팀원 소개
- 1-3. 일정 소개
- 1-4. 벤치마킹

02 1차 스프린트 회고

03 요구사항 분석

- 3-1. 페르소나
- 3-2. 컨셉모델
- 3-3. 워크플로우
- 3-4. UI 프로토타입
- 3-5. 데이터 모델링

04 프로젝트 설계

- 4-1. 소프트웨어 아키텍처
- 4-2. 협업 환경

05 구현 및 테스트

- 5-1. 구현 산출물
- 5-2. 테스트
- 5-3. 시연

06 회고

01

프로젝트 개요

1-1. Selfit 소개

1-2. 팀원 소개

1-3. 일정 소개

1-4. 벤치마킹

1-1. Selfit 소개



Selfit은 스스로 운동을 계획하고 기록하며, 건강한 습관을 만들어 가는 사람들을 위한 동기부여 커뮤니티 플랫폼입니다.

1-2. 팀원 소개

염윤호

팀장

대시보드, Jira 관리

고민표

팀원

로그인, 회원가입, 마이페이지

박희수

팀원

대시보드, 업무상세설계서

이재원

팀원

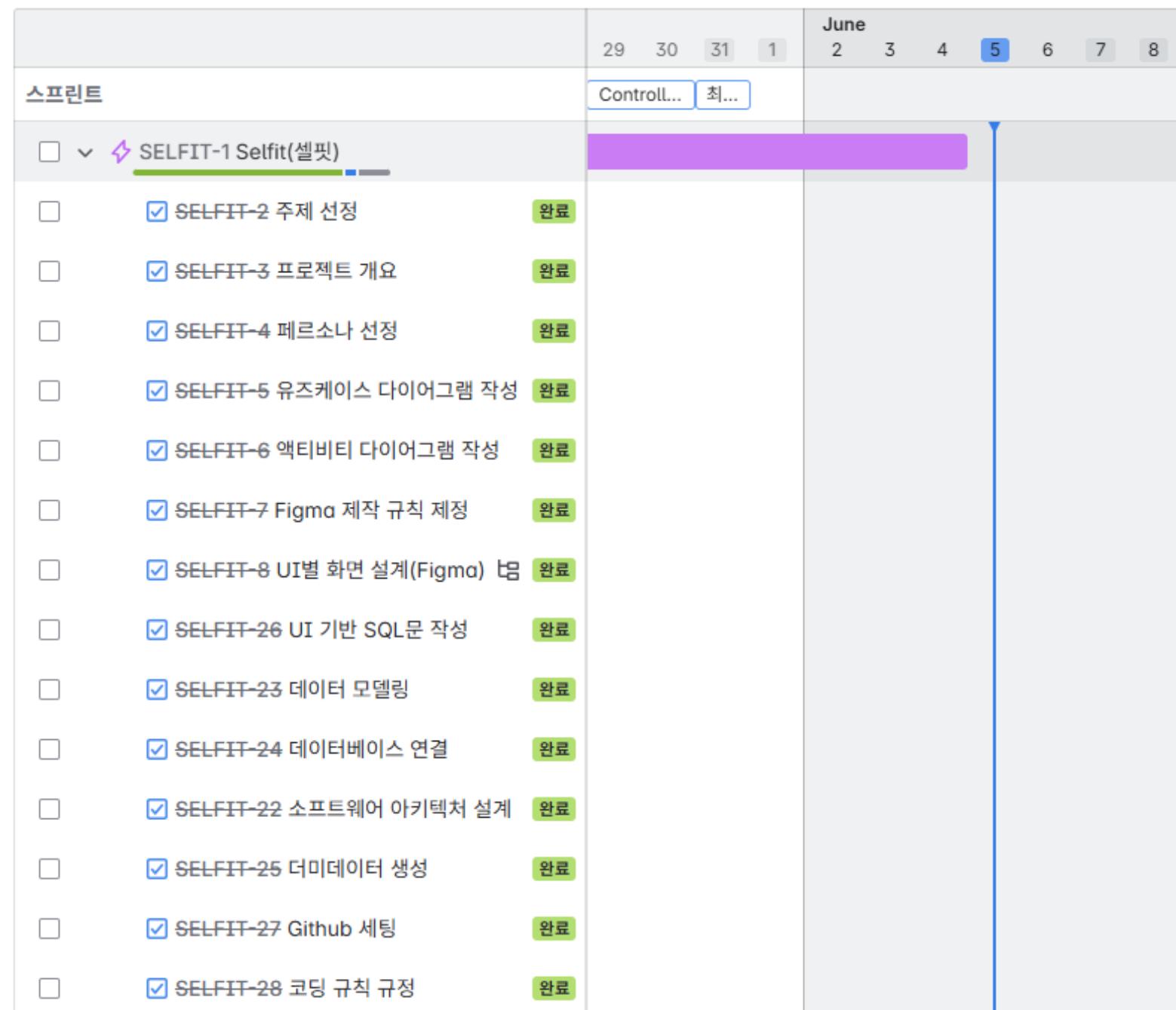
커뮤니티, Firebase

이창훈

팀원

대시보드, 통합테스트

1-3. 일정 소개



1-4. 벤치마킹

분류	사이트	주요 기능	특징
건강관리	Cronometer	섭취·운동 기록, 일일 요약, 물 섭취량 체크	개인 트래킹 집중
커뮤니티	몬스터짐	카테고리별 글 작성·검색, 쇼핑몰 연동	활발한 커뮤니티 기능
다이어트	다이어트신	칼로리 계산, 커뮤니티	기본 기능 중심
제약·영양제	Examine	성분별 검색, 질환별 추천	정보 제공 집중
제약·영양제	drugs.com	약 비교, 부작용, 성분표	전문 정보 제공
뷰티·화장품	쿠스	원료 검색, 포럼	커뮤니티 연계

→ 기록을 할 수 있지만 커뮤니티 기능이 없음

→ 커뮤니티 기능은 있지만 단순 계산만 제공

02

1차 스프린트 회고

2. 1차 스프린트 회고

Keep(유지)

정기적인 회의

Jira를 통한 일정 및 업무 관리

아키텍처 설계의 역할 분리

Git 커밋 메시지 컨벤션

<업무>

Problem(문제점)

코드의 일관성 부족

수정 사항 및 공유사항에 대한 소통 부족

명확하지 않은 업무 분담

<기술>

Problem(문제점)

테스트 미비

FrontController + ActionFactory 확장성 한계
GitHub 브랜치 병합 시 충돌

보일러 플레이트 코드 반복

신규 업무에 MyBatis 적용 어려움

Try(시도)

일관성 노트 제작

Confluence 회의록 및 공유 템플릿 활용

Jira 보드 내 하위 업무 분담 활성화

Try(시도)

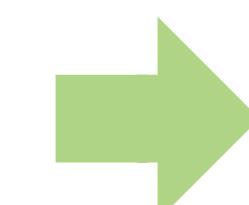
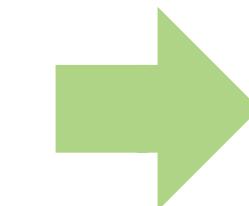
Spring Boot Framework 도입

Junit5 기반 테스트 프레임워크 적극 활용
Rest API 설계 도입

정기적인 코드 리뷰

일일 merge 시간 확보

JPA 활용 검토



03

요구사항 분석

- 3-1. 페르소나
- 3-2. 컨셉모델
- 3-3. 워크플로우
- 3-4. UI 프로토타입
- 3-5. 데이터 모델링

3-1. 페르소나



이름	이민준
성별	남자
나이	21
직업	대학생
사는 곳	서울시 동작구

성향

계획 세우기를 잘 못하고 강제성이 없으면 금방 포기함
소속감을 중요시함

니즈

운동 및 식단 관리 시 다른 사람과 정보·경쟁·비교하여 동기부여 얻기
계획을 대신 짜줬으면 함

문제점

다른 사람에게 식습관·체형·생활습관 기반 식단 추천을 받기 어려움
"식단 관리"라 하면 닭가슴살밖에 떠오르지 않음

주요기능

게시판에서 정보 얻기, 목표 체크리스트, 차트 대시보드(나 vs 비교군)

3-1. 페르소나



성향	다른 사람에게 조언하는 것을 좋아하고, 호기심이 많음
니즈	커뮤니티를 통해 운동·식단 관리에 어려움을 겪는 사람들을 돋고 싶음
문제점	질문을 받아도 개인에 맞춘 피드백을 줄 정보가 부족해 맞춤형 조언 제공이 어려움

이름 김종국

성별 남자

나이 39

직업 헬스트레이너

사는 곳 서울시 성북구

주요기능 운동·식단 게시판, 질문 템플릿

3-1. 페르소나



이름	송민지
성별	여자
나이	28
직업	프리랜서 UI 디자이너
사는 곳	성남시 분당구

성향

자기 관리에 관심도가 높고, 일정이 불규칙해 기록을 좋아함
시각적 피드백(그래프, 차트)이 동기부여 수단

니즈

섭취, 활동 칼로리 계산 및 기록

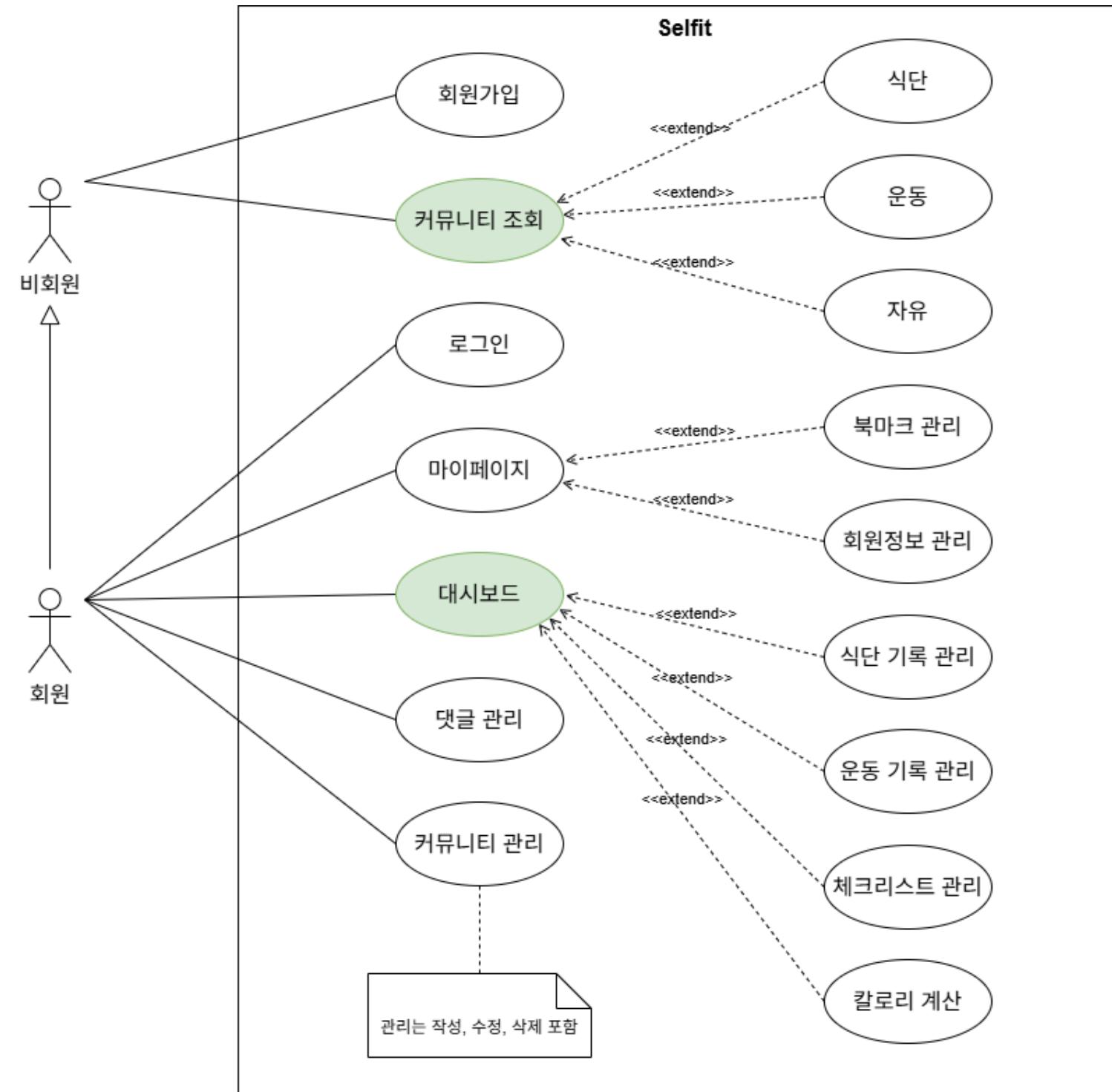
문제점

단순 칼로리 계산만 가능한 사이트는 많지만, 매일 기록을 남기고 주, 월 평균을 확인할 수 있는 서비스가 없음

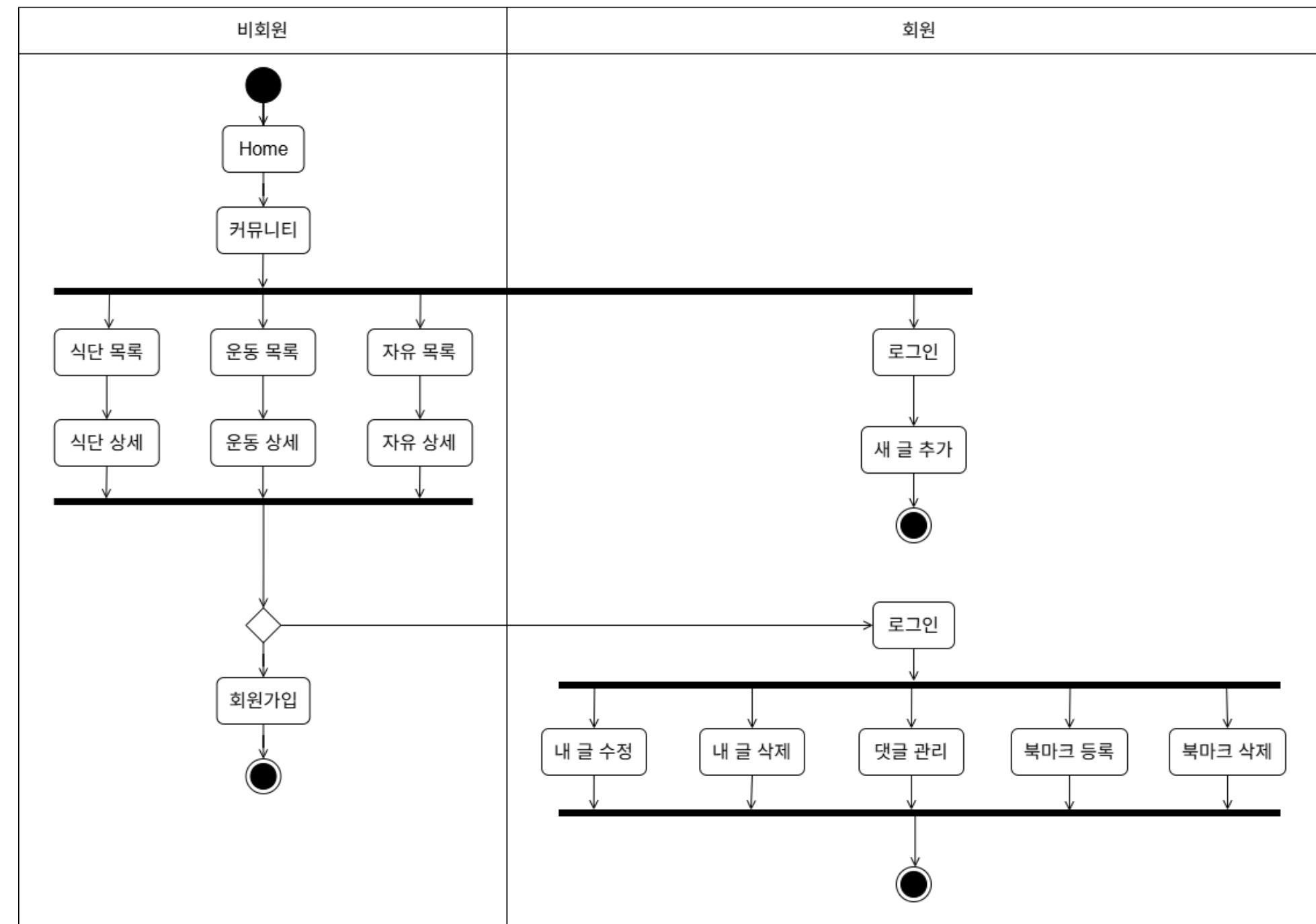
주요기능

칼로리 계산, 차트 대시보드(나 vs 비교군), 목표 체크리스트

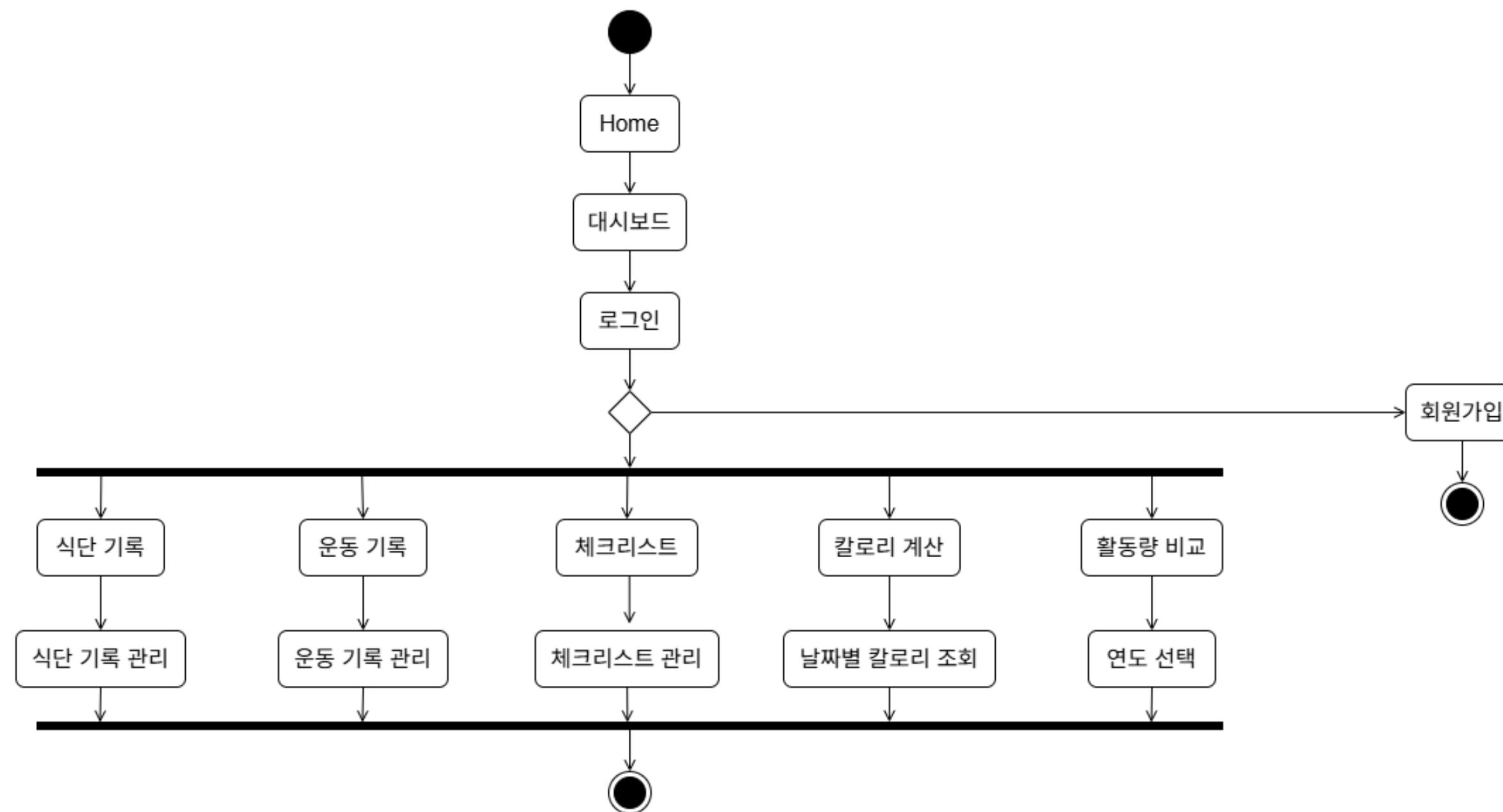
3-2. 컨셉 모델



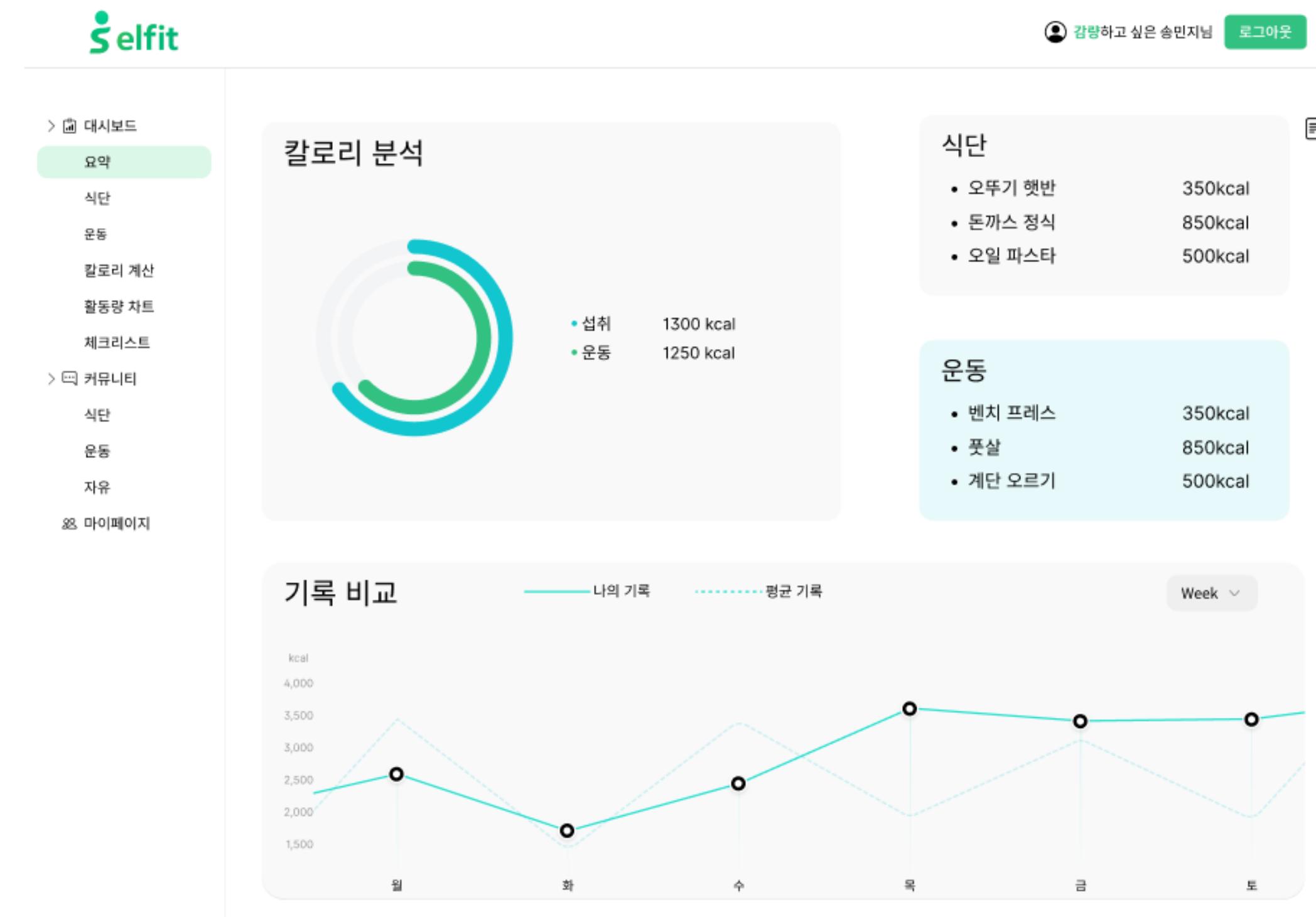
3-3. 워크 플로우 (커뮤니티 기능)



3-3. 워크 플로우 (대시보드 기능)



3-4. UI 프로토타입



3-4. UI 프로토타입

The screenshot shows a user interface for a meal planning application. At the top, there is a navigation bar with the logo 'Selfit' and a search bar containing the text '김량하고 싶은 송민지님'. To the right of the search bar are two buttons: '로그아웃' (Logout) and '최신순' (Latest) which is currently selected.

The main content area is titled '식단' (Meal Plan). On the left, there is a sidebar with a tree-like menu:

- 대시보드 (Dashboard)
- 커뮤니티 (Community)
- 식단** (Meal Plan) - This item is highlighted with a green background.
- 운동 (Exercise)
- 자유 (Free)
- マイ페이지 (My Page)

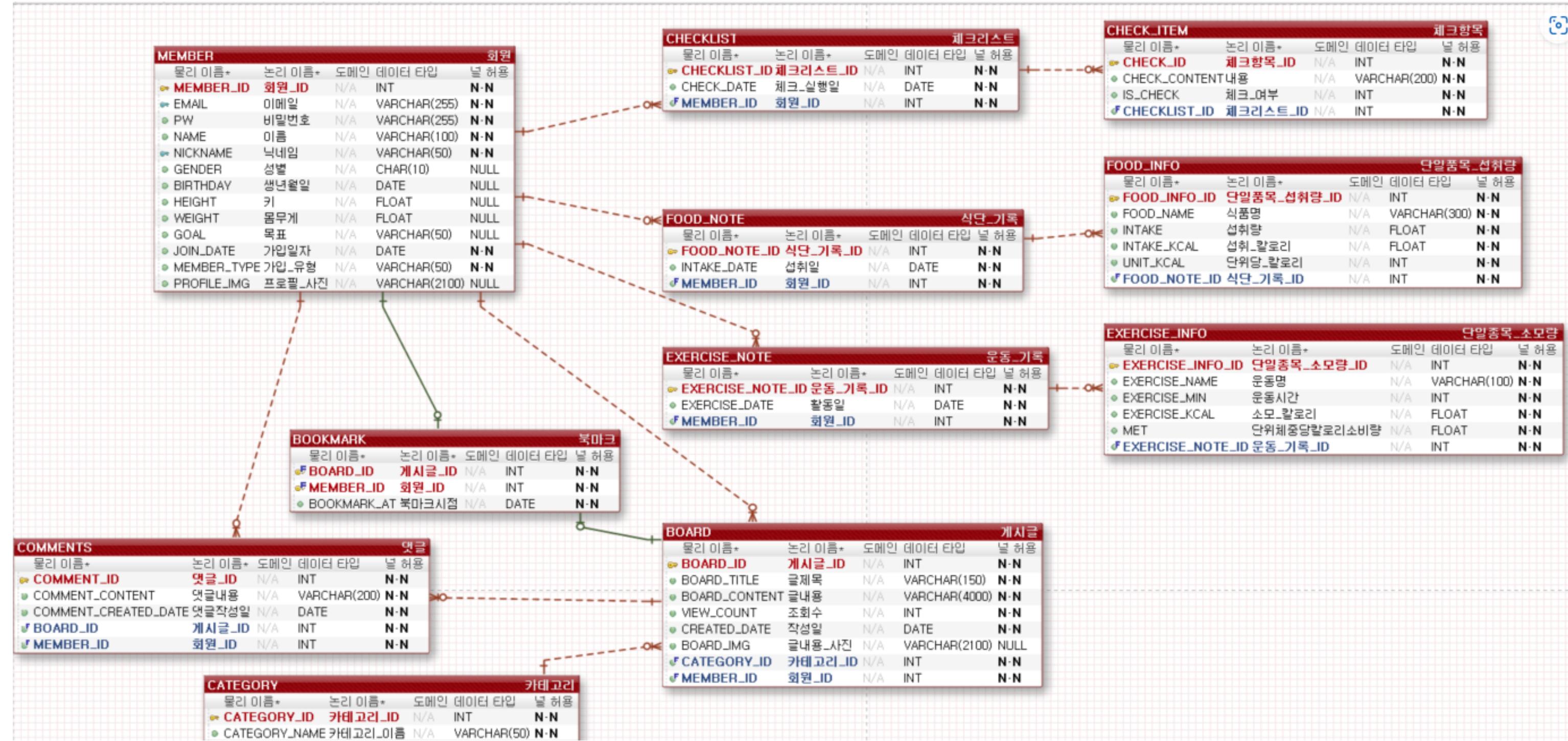
Below the sidebar, there is a search bar with the placeholder '글 쓰기' (Write a post). To the right of the search bar are two radio buttons: '조회순' (Viewed first) and '최신순' (Latest), with '최신순' being selected.

The main content area displays a table of meal plans:

글 번호	제목	작성자	조회수	작성일
10	첫 일주일 식단 도전기	귀요미마미	85	25.05.17
9	아침: 오트밀&바나나 플레이트	슬림	112	25.05.16
8	점심: 닭가슴살 샐러드 + 밸사믹 드레싱	하루한끼	97	25.05.15
7	저녁: 두부 스테이크와 구운 야채	유지어터	54	25.05.14
6	간식: 그릭 요거트 & 블루베리	집콕운동	123	25.05.14
5	식단 힘드네요	극단적	76	25.05.14
4	간식: 애플칩 & 피넛버터 디핑	스무디조아	89	25.05.14
3	회사원 식단 좀 짜주세요!!	샐러디	45	25.05.13
2	저녁: 현미 파스타 + 토마토 바질 소스	현실탈출	68	25.05.13
1	다이어트 할때 빵 먹어도 되나요?	베이글	102	25.05.13

At the bottom of the page, there is a navigation bar with icons for back, forward, and search, followed by a page number indicator showing pages 1 through 5.

3-5. 데이터 모델링



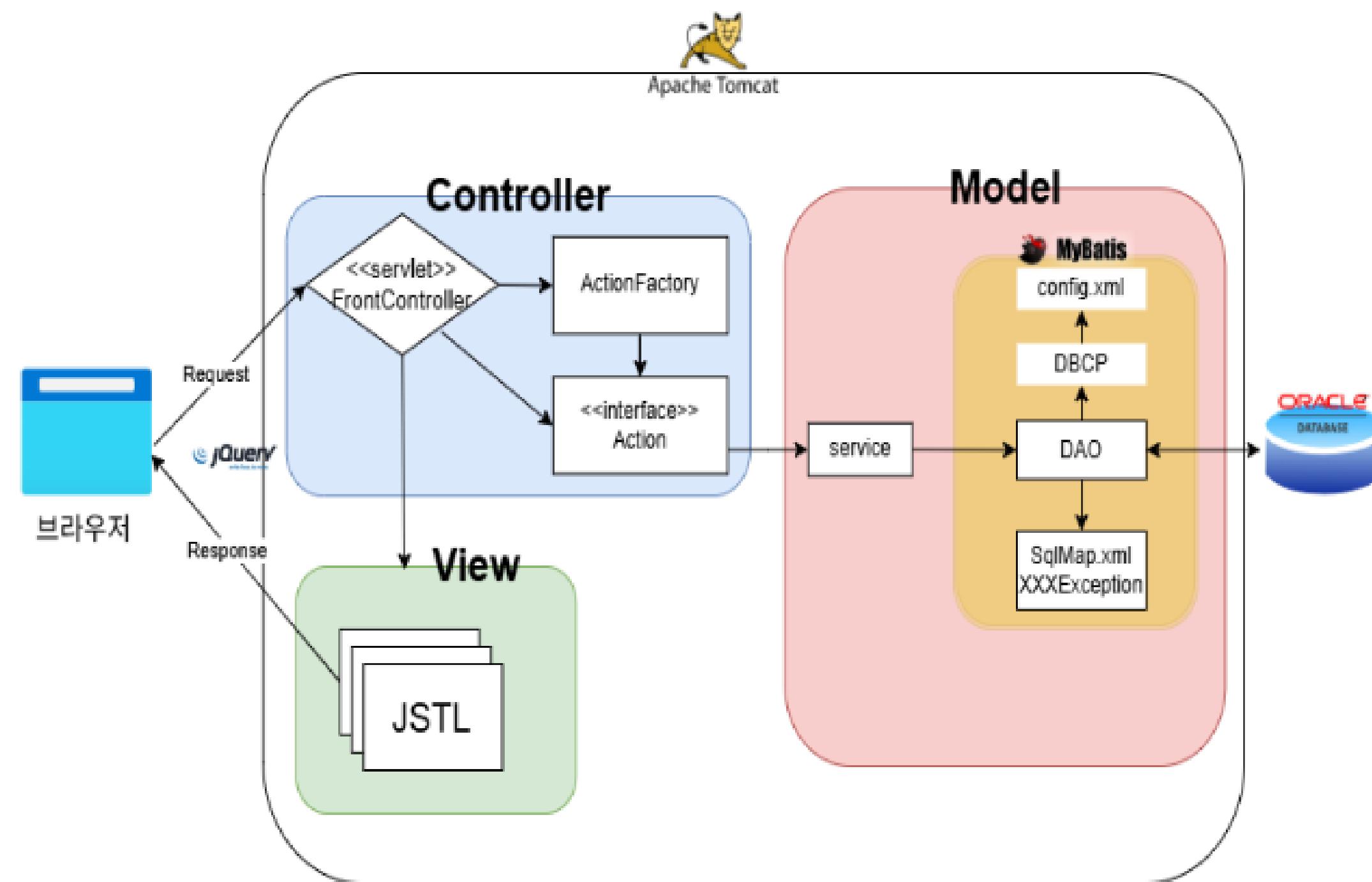
04

프로젝트 설계

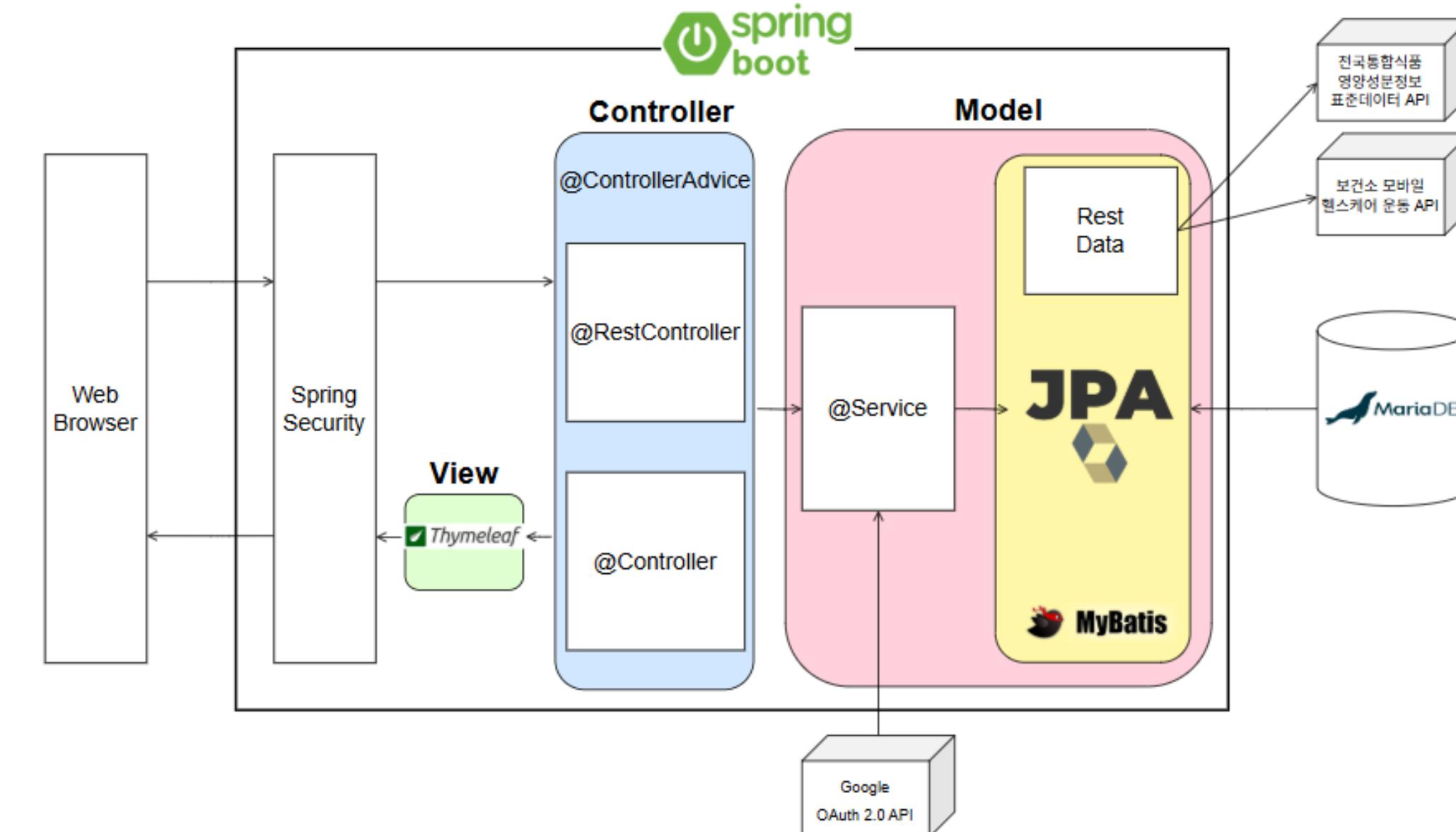
4-1. 소프트웨어 아키텍처

4-2. 협업 환경

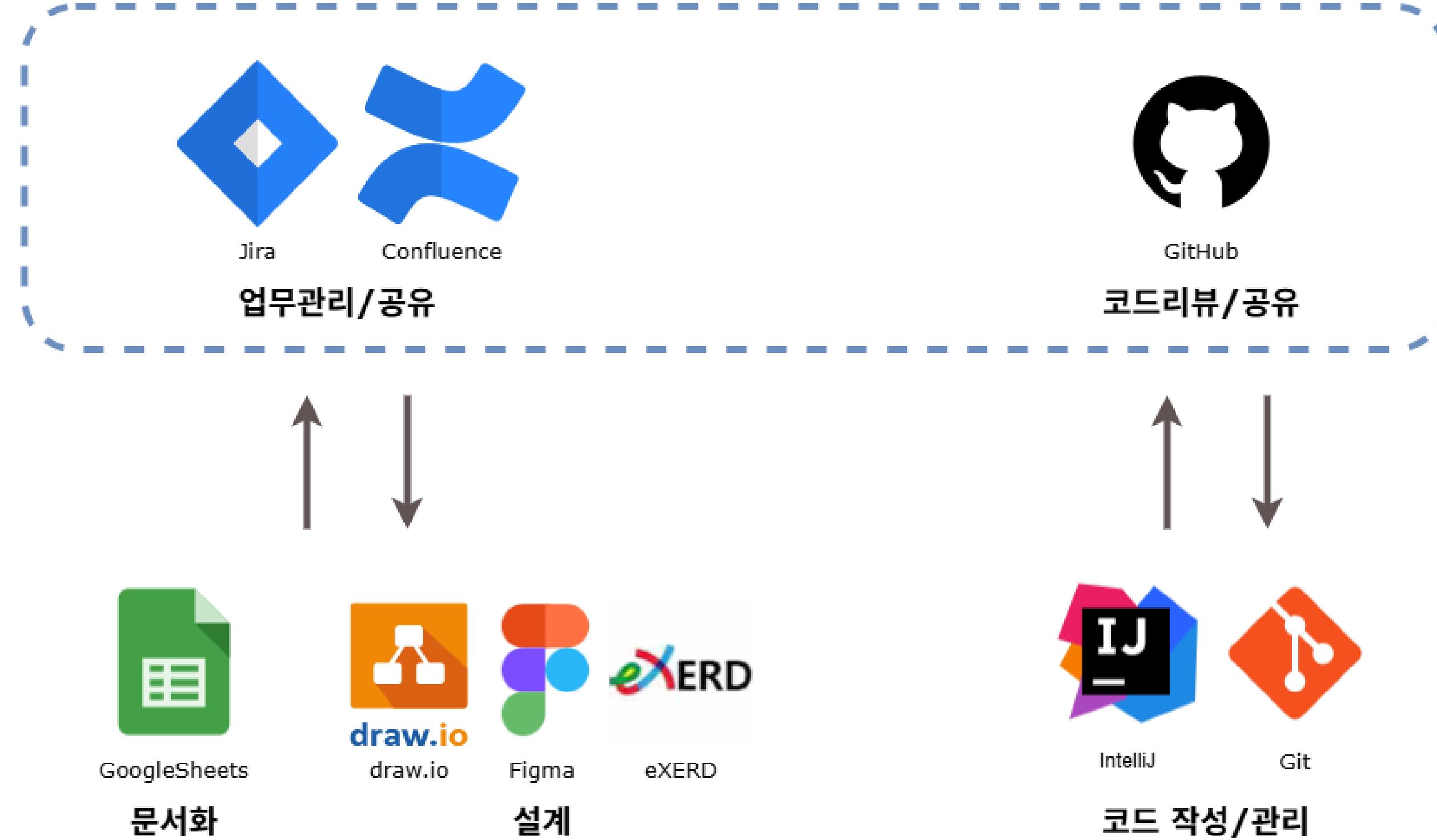
4-1. 소프트웨어 아키텍처 (1차 스프린트)



4-1. 소프트웨어 아키텍처 (2차 스프린트)



4-2. 협업 환경



05

구현 및 테스트

5-1. 구현 산출물

5-2. 테스트

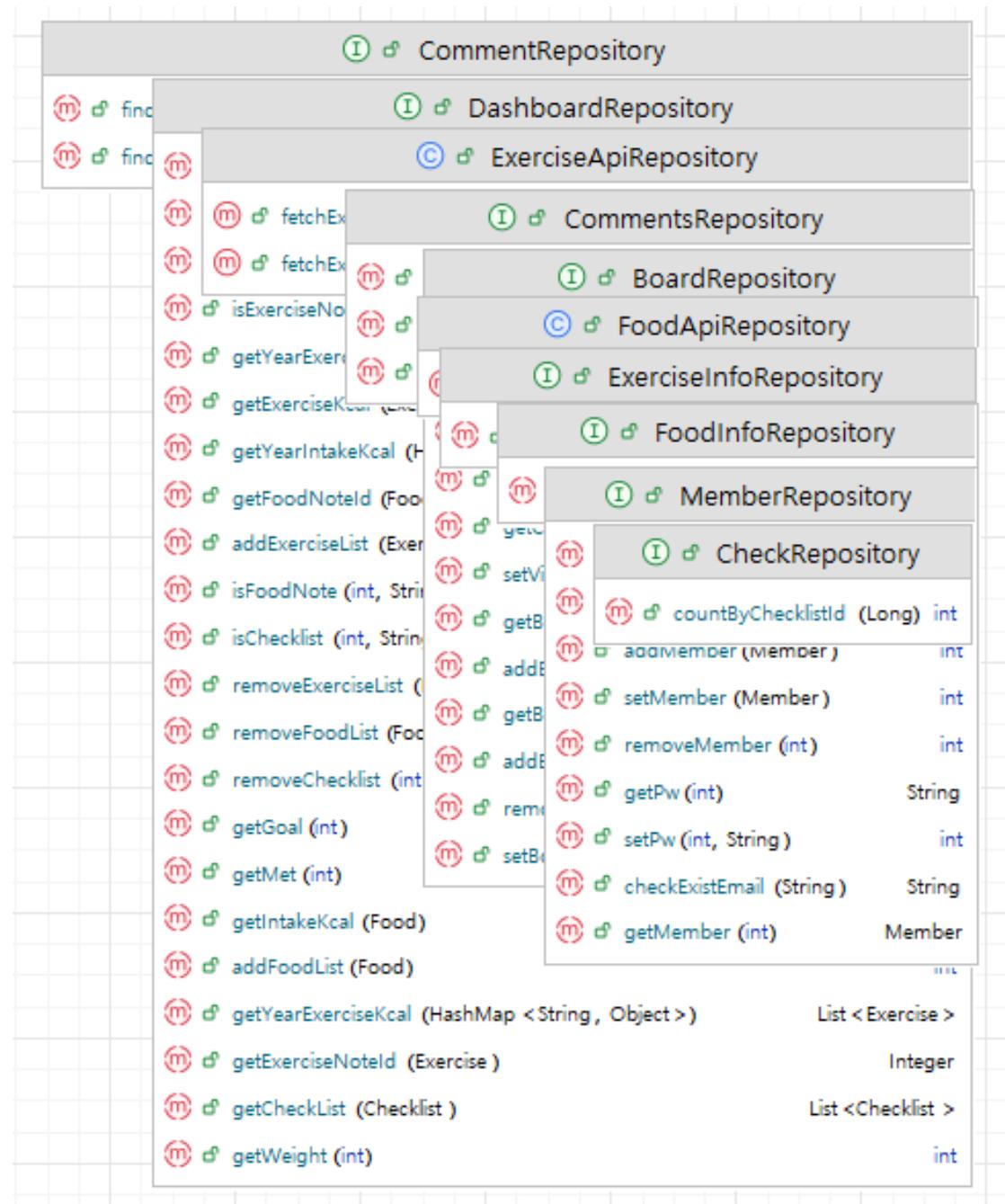
5-3. 시연

5-1. 구현 산출물 (업무상세설계서)

업무상세설계서

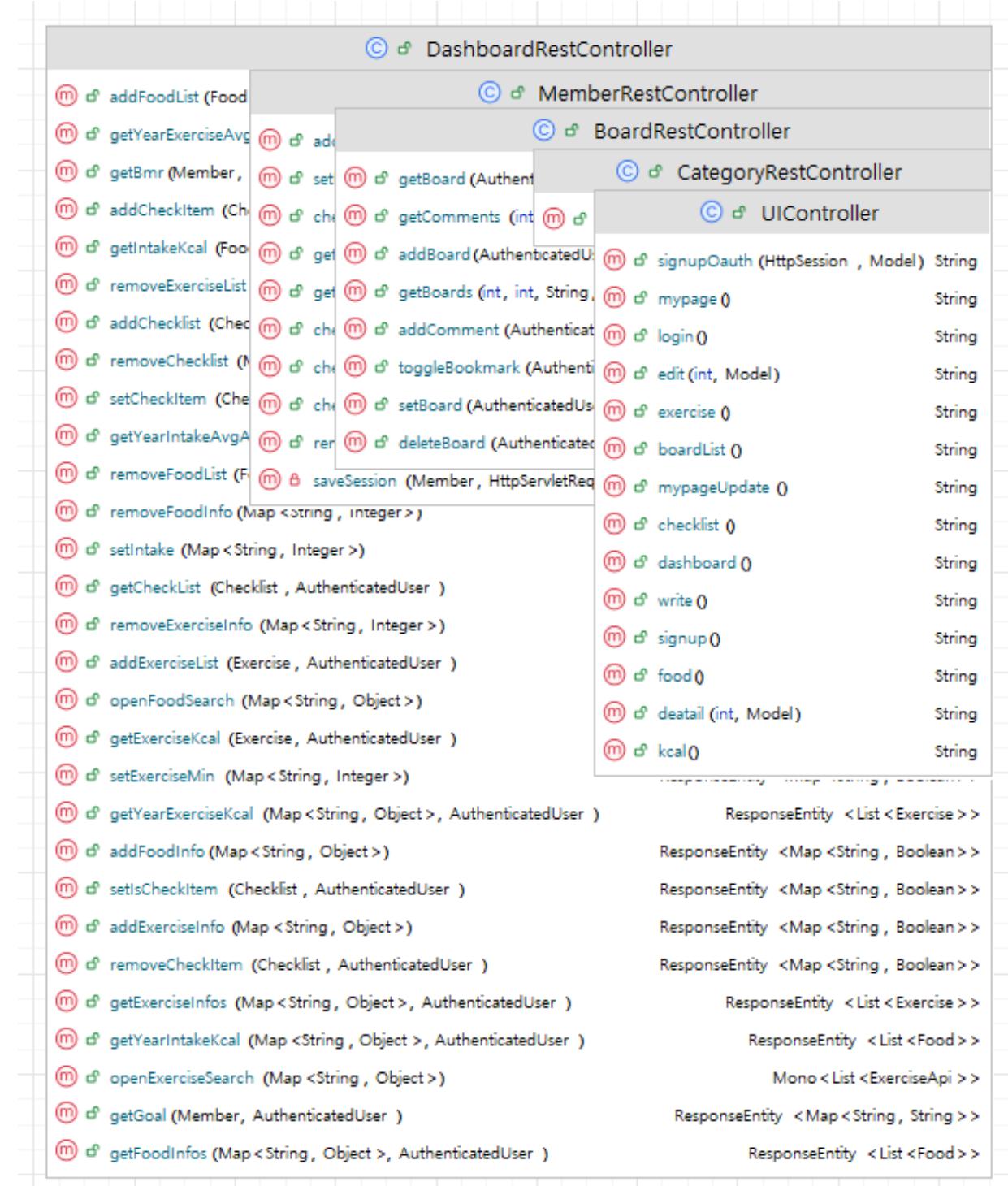
업무	구분	흐름	URL	Security		Controller				Exception	MemberService			Repository			권한
				req	res	클래스명	종류	req	res		res	method	input	output	클래스명	input	output
로그인	화면 ▼	홈페이지 첫화면에서 대쉬보드 or 커뮤니티 or 헤더에서 로그인 버튼 클릭	/account/login	-	-	UIController	GET	-	String	-	-	-	-	-	-	-	ROLE_ANONYMOUS
회원가입	화면 ▼	로그인 화면에서 회원가입 클릭	/account/signup	-	-		GET	-	String	-	-	-	-	-	-	-	ROLE_ANONYMOUS
마이페이지	화면 ▼	사이드바에서 마이페이지 클릭 or 헤더에서 아이콘클릭	/account/mypage	-	-		GET	-	String	-	-	-	-	-	-	-	ROLE_USER
마이페이지 수정	화면 ▼	마이페이지에서 수정버튼 클릭	/account/mypage-update	-	-		GET	-	String	SQLSyntaxErrorException	-	-	-	-	-	-	ROLE_USER
개인정보 조회	처리 ▼	마이페이지 들어갈 때 요청	/api/account/member	-	-	MemberRestController	GET	@AuthenticationPrincipal AuthenticatedUser loginUser	ResponseEntity<Member>	-	getMember	memberId	Member	Member Repository	memberId	Member	ROLE_ANONYMOUS
로그인 처리	처리 ▼	로그인 버튼 누를 때 요청	/account/login-process	email, pw	redirect		POST	-	-	loadUserByUsername	email	UserDetails	-		email	LoginInfo	ROLE_ANONYMOUS
로그아웃 처리	처리 ▼	헤더에 로그아웃 버튼 누를 때 요청	/account/logout-process	-	redirect		POST	-	-	-	-	-	-		-	-	ROLE_USER
이메일 중복체크	처리 ▼	회원가입 페이지에서 이메일 중복체크 누를 때 요청	/api/account/check-email	-	-		POST	@RequestBody String jsonData	ResponseEntity<Map<String, Boolean>>	-	isEmailExists	email	boolean		email	String	ROLE_ANONYMOUS
닉네임 중복체크	처리 ▼	회원가입 페이지에서 닉네임 중복체크 누를 때 요청	/api/account/check-nickname	-	-		POST	@RequestBody String jsonData	ResponseEntity<Map<String, Boolean>>	-	isNicknameExists	nickname	boolean		nickname	String	ROLE_ANONYMOUS
회원가입 처리	처리 ▼	회원가입 페이지에서 계정등록 누를 때 요청	/api/account/member	-	-		POST	@RequestBody Member member, HttpServletRequest request	ResponseEntity<Map<String, Boolean>>	IllegalArgumentExcepton	addMember	member	boolean		member	int	ROLE_ANONYMOUS
유저 북마크 조회(페이징)	처리 ▼	마이페이지 들어갈 때 요청	/api/account/member/bookmarks/{offset}	-	-		GET	loginUser, offset	List<BookMark>	-	getBookmarks	memberId, limit, offset	List<Bookmark>	Board Repository	memberId, limit, offset	List<Bookmark>	ROLE_USER

5-1. 구현 산출물 (Repository)



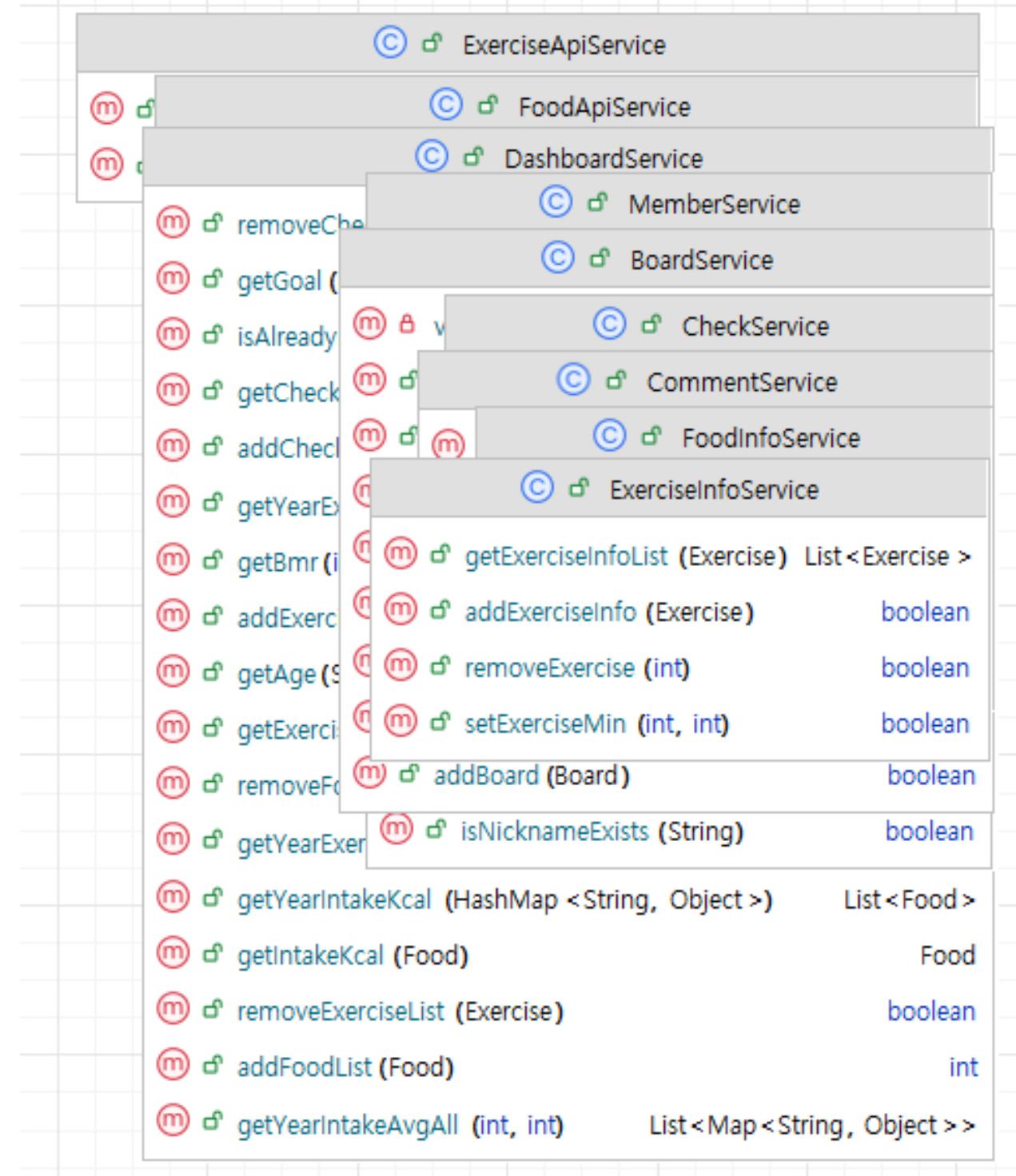
Repository 총 10개

5-1. 구현 산출물 (Controller)



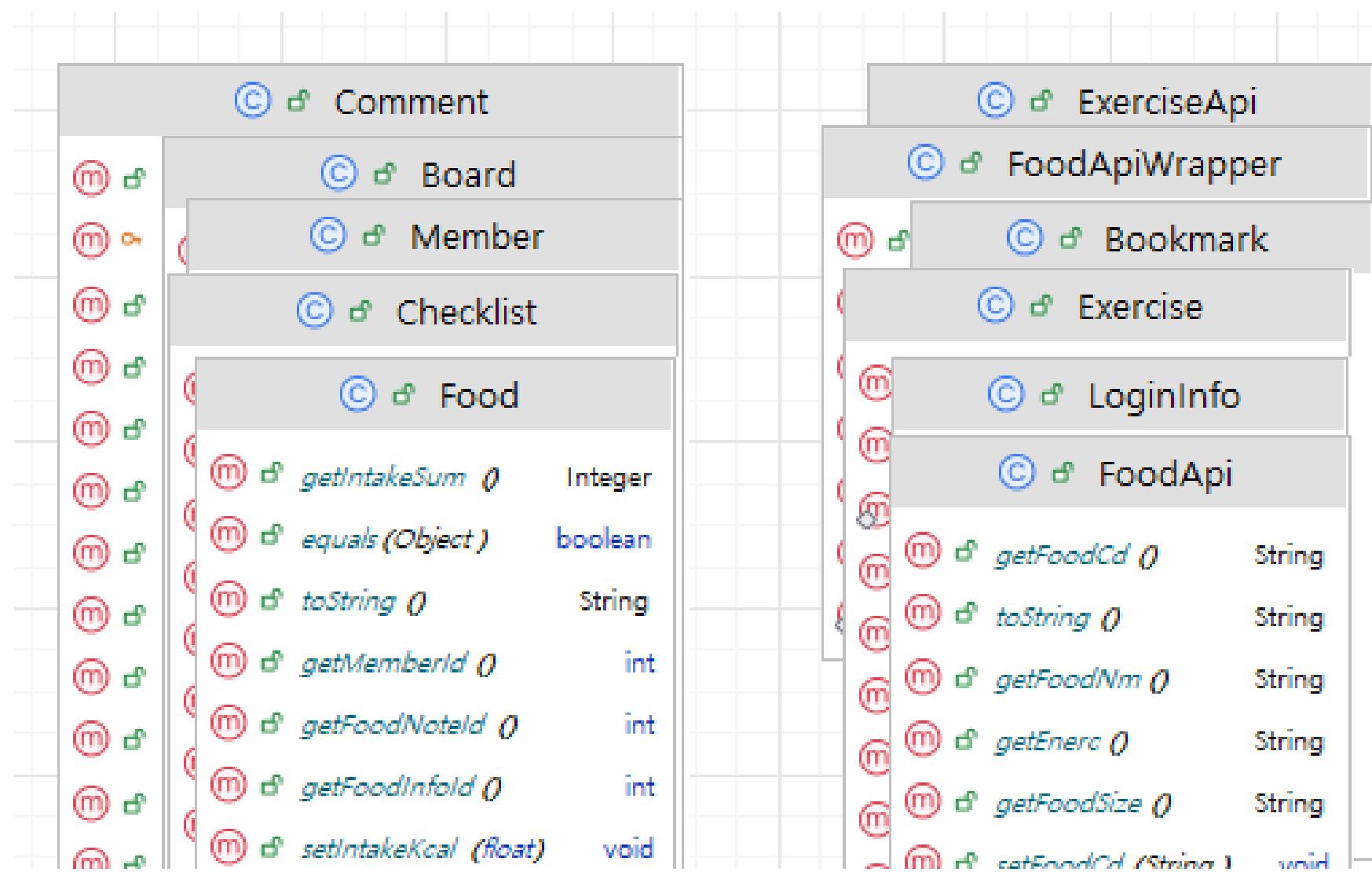
Controller 총 5개

5-1. 구현 산출물 (Service)



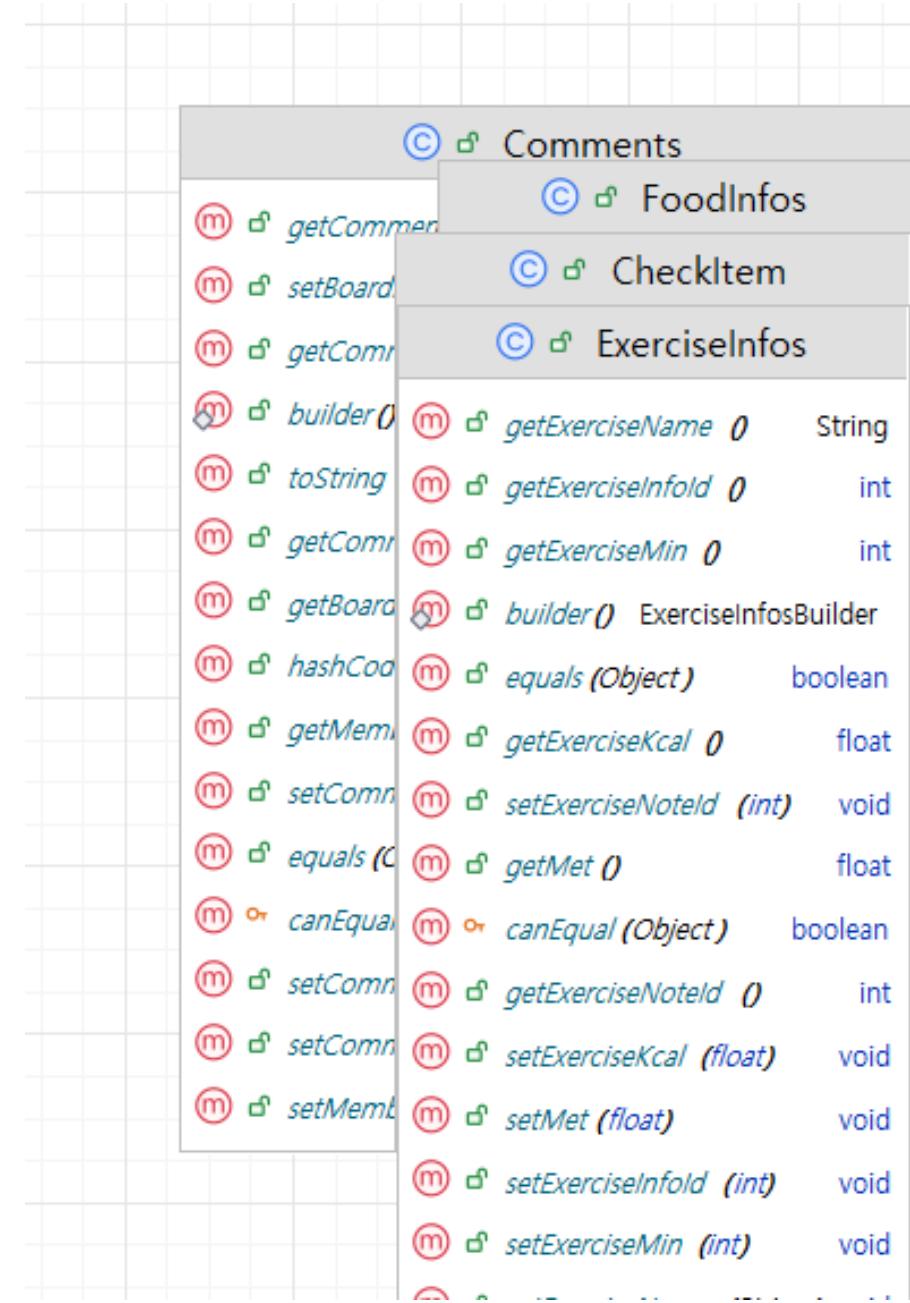
Service 총 9개

5-1. 구현 산출물 (DTO)



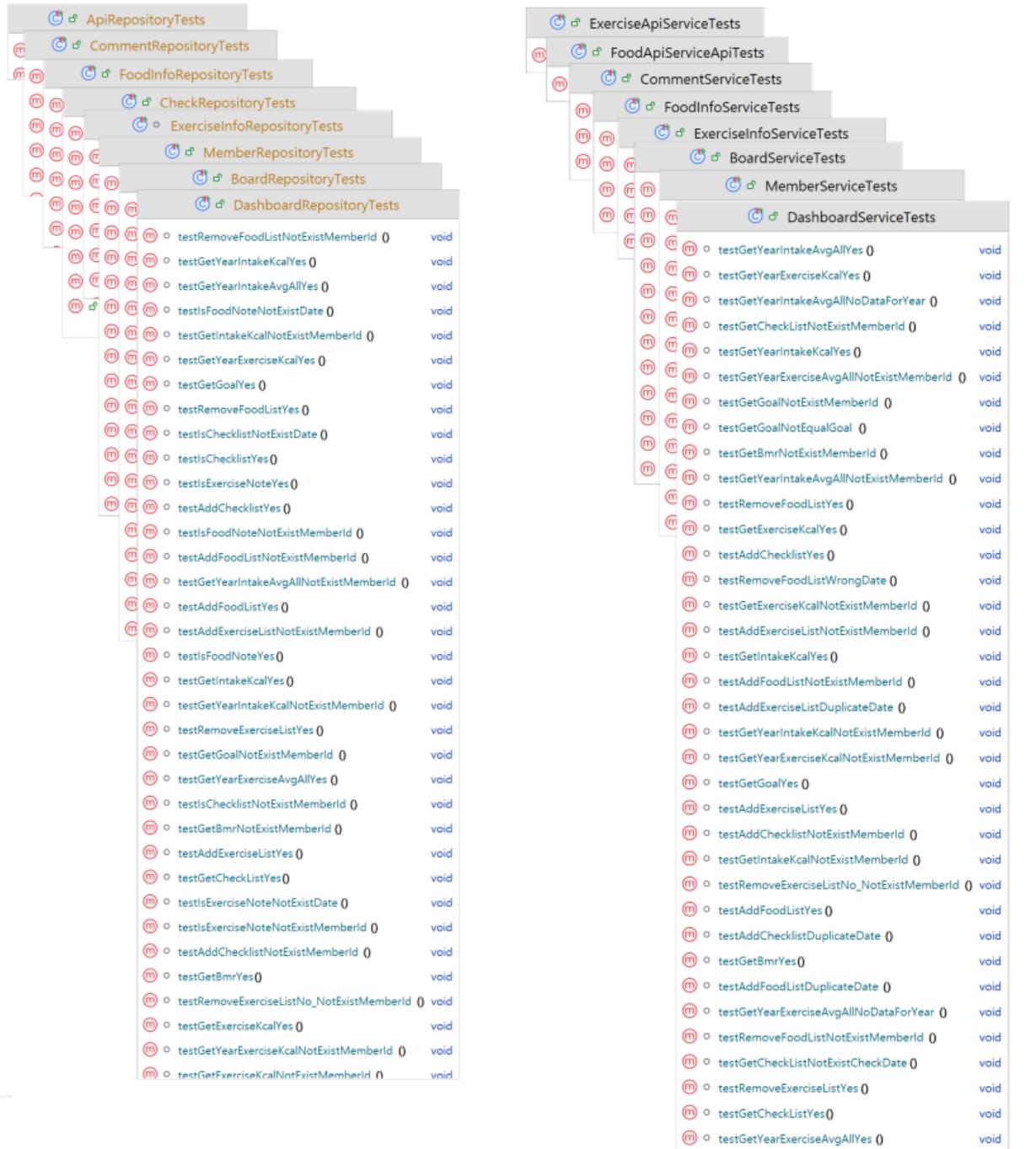
DTO 총 12개

5-1. 구현 산출물 (Domain)



Domain 총 4개

5-2. 테스트 (Junit5)



Repository 총 98개

Service 총 74개

5-2. 테스트 (Swagger)

The screenshot displays the Swagger UI interface, which provides a visual representation of the API's endpoints and their methods. The interface is divided into several sections:

- OpenAPI definition**: Shows the version (OAS 3.0) and the generated server URL (<http://192.168.0.8:8881>).
- dashboard-rest-controller**: A large section listing various API endpoints:
 - PUT /api/dashboard/food
 - POST /api/dashboard/food
 - DELETE /api/dashboard/food
 - PUT /api/dashboard/exercise
 - POST /api/dashboard/exercise
 - DELETE /api/dashboard/exercise
 - PUT /api/dashboard/checklist/item
 - POST /api/dashboard/checklist/item
 - DELETE /api/dashboard/checklist/item
 - PUT /api/dashboard/checklist/item/check
 - POST /api/dashboard/goal
 - POST /api/dashboard/foods
 - POST /api/dashboard/food/openSearch
 - POST /api/dashboard/food/list
 - DELETE /api/dashboard/food/list
 - POST /api/dashboard/food/kcal
 - POST /api/dashboard/food/kcal/year
 - POST /api/dashboard/exercises
 - POST /api/dashboard/exercise/openSearch
 - POST /api/dashboard/exercise/list
 - DELETE /api/dashboard/exercise/list
 - POST /api/dashboard/exercise/kcal
 - POST /api/dashboard/exercise/kcal/year
 - POST /api/dashboard/exercise/kcal/svg/year
 - POST /api/dashboard/checklist
 - DELETE /api/dashboard/checklist/items
 - POST /api/dashboard/checklist/items
 - POST /api/dashboard/beer
- member-rest-controller**: A detailed view of the GET endpoint for account/member:
 - Parameters**: No parameters.
 - Responses**:
 - Curl**: curl -X 'GET' \ "<http://192.168.0.8:8881/api/account/member>" \ -H 'accept: */*'
 - Request URL**: <http://192.168.0.8:8881/api/account/member>
 - Server response**:
 - Code**: 200
 - Response body**:

```
<!DOCTYPE html>

  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>gradient Background Layout</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">
    <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.1.0/font/bootstrap-icons.css">
    <script src="https://accounts.google.com/gsi/client" async defer></script>
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>
    <script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>
    <script src="/js/account/login.js"></script>
  </head>
  <body>
    <div>Hello, Swagger!</div>
    <div id="container">
      <!-- 배경 그레이디언트 색상 -->
      <div class="background-layer"></div>
      <!-- 오른쪽 위 모서리 그레이디언트 -->
      <div class="top-right-circle"></div>
      <!-- 왼쪽 아래 모서리 그레이디언트 -->
      <div class="bottom-left-circle"></div>
    </div>
  </body>
</html>
```
- board-rest-controller**
- member-rest-controller**
- category-rest-controller**
- Schemas**

5-3. 시연

Selfit

> 대시보드
> 커뮤니티
マイ페이지

대시보드

체크리스트

칼로리 분석

평균 14%

● 섭취 288 kcal
● 운동 252 kcal

식단

- 친환경 F&B 베이커리 소시지 100g (288 kcal)

운동

- 테니스 단식경기 30분 (252 kcal)

기록 비교

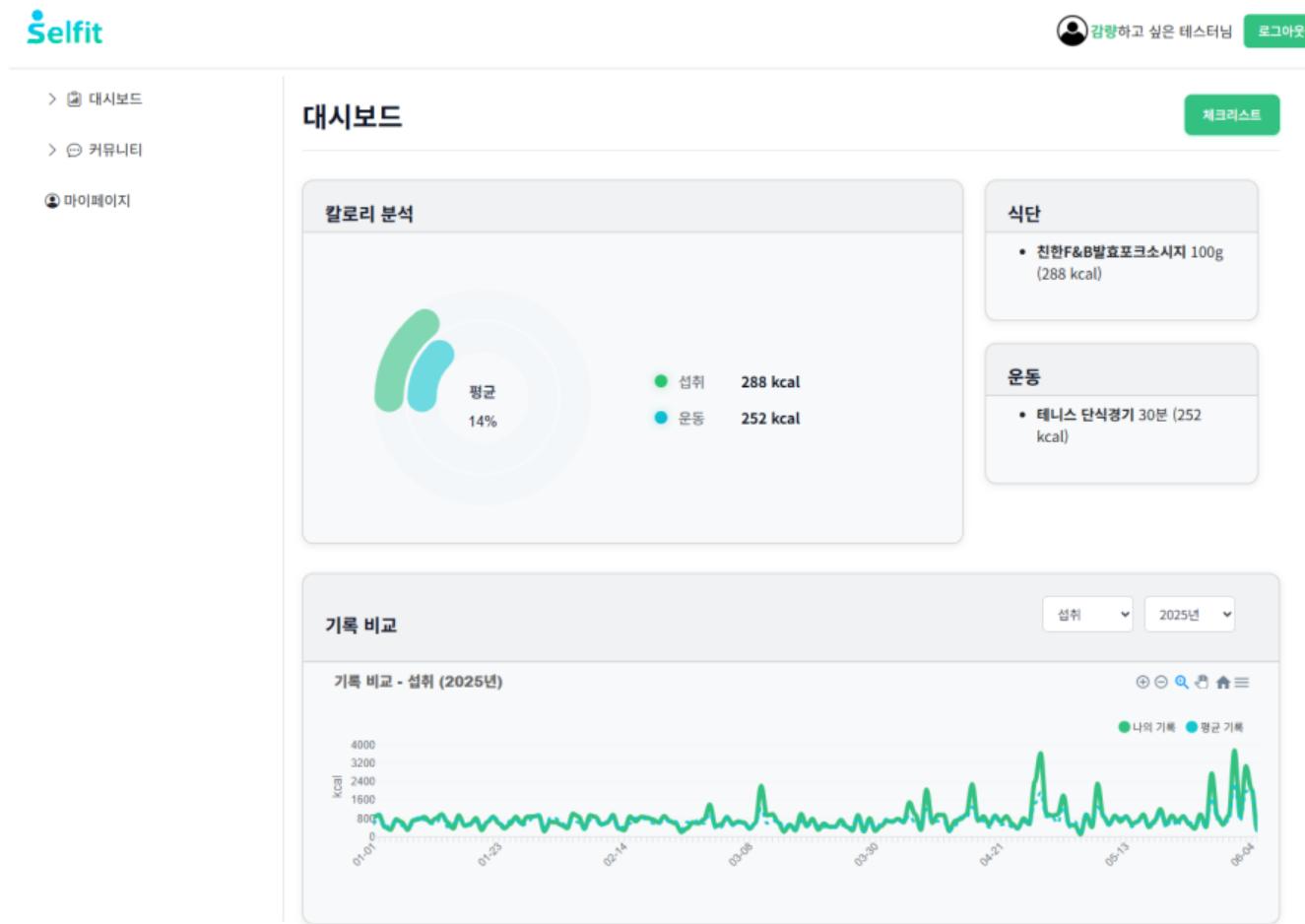
기록 비교 - 섭취 (2025년)

선택 기록 2025년

Kcal

01-01 ~ 06-01

나의 기록 ● 평균 기록 ●



Selfit

건강한 변화의 시작

매일의 기록이 만드는 변화

대시보드

기록을 통해 변화를 확인해보세요 :)

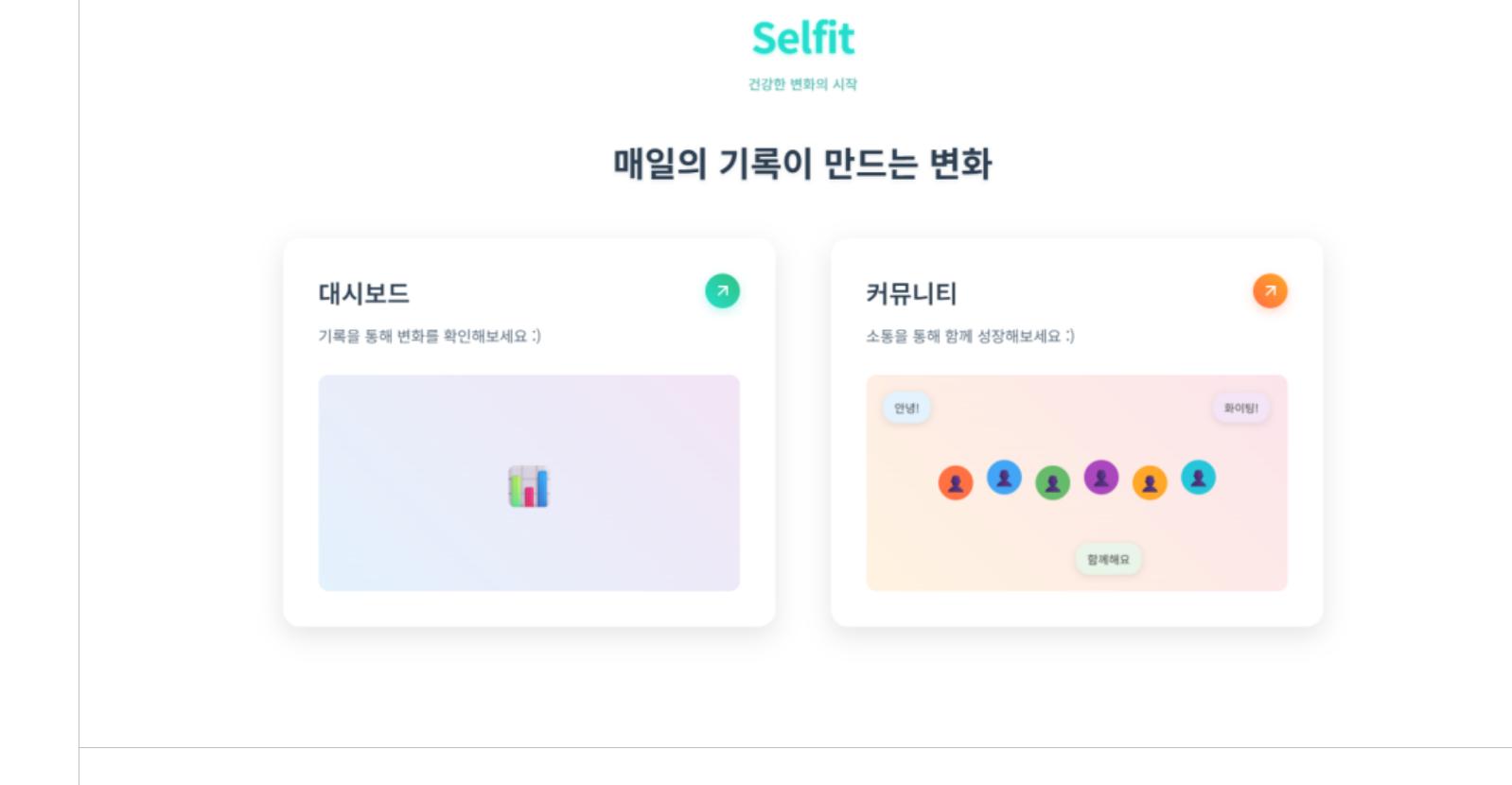
커뮤니티

소통을 통해 함께 성장해보세요 :)

안녕!

화이팅!

함께해요



5-3. 시연

Selfit

회원 정보

회원탈퇴 수정

이메일 test@test.com 성별 남자

이름 테스터 생년월일 1996-12-03

닉네임 테스터 키(cm) 173cm

운동 목적 감량

북마크

저녁 식단으로 연어 스테이크 도전!

Selfit

글 작성

제목을 입력하세요

카테고리 선택

내용을 입력하세요

클릭하여 파일을 업로드하세요

게시하기

5-3. 시연

Selfit

로그인하고 싶은 테스터님 로그아웃

대시보드

커뮤니티

식단

운동

자유

マイ페이지

식단

글쓰기

조회순 최신순

제목	작성자	조회수	작성일
점심 도시락으로 통곡물빵 샌드위치	sujin	25	25.06.04
단백질 보충 위해 삶은 달걀 추가	도윤55	22	25.06.03
집에 남은 야채로 오이냉국 만들었어요	준우34	30	25.06.02
지방 적게 든 생선구이 시도	시우41	25	25.05.31
점심 대신 시리얼과 견과류	seoyeon	38	25.05.26
간단하게 요거트에 과일 얹어서 먹었어요	minsu	18	25.05.19
저녁 다이어트 식사: 브로콜리 스텀	테스터	29	25.05.12
샐러드에 두부 대신 렌틸콩 넣어서 단백질 보충	현우23	11	25.05.07
점심으로 고구마 두 개 삶아 먹었어요	donghyun	34	25.05.01
간편하게 닭가슴살 구워 오븐에 구웠어요	도윤55	41	25.04.25

검색어를 입력하세요.

1 2 3 > >>

Selfit

로그인하고 싶은 테스터님 로그아웃

대시보드

식단 기록

운동 기록

칼로리 계산

체크리스트

커뮤니티

マイ페이지

체크리스트

2025.06.05

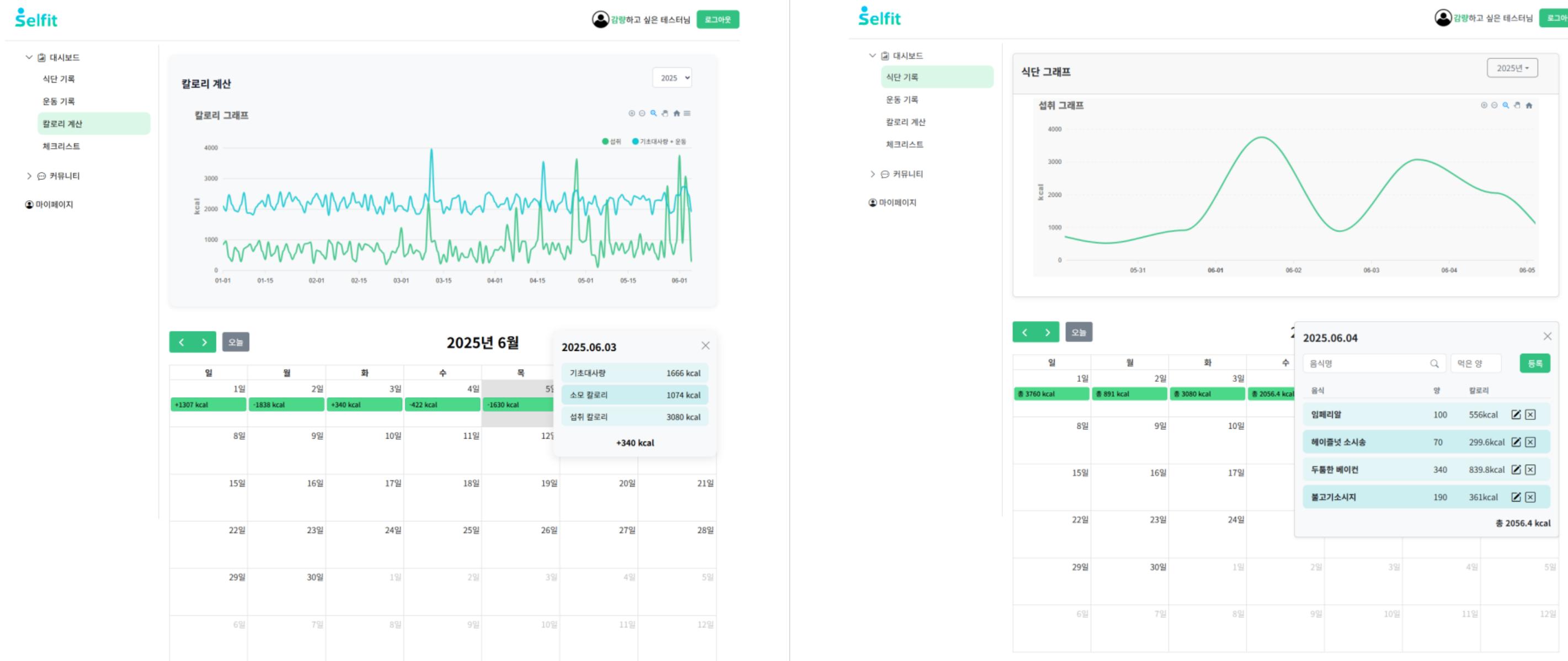
등록 7일

등 운동하기

인바디 체크

일	월	화	수	목	금	토	일
1일	2일	3일					
8일	9일	10일	11일	12일	13일	14일	
15일	16일	17일	18일	19일	20일	21일	
22일	23일	24일	25일	26일	27일	28일	
29일	30일	1일	2일	3일	4일	5일	
6일	7일	8일	9일	10일	11일	12일	

5-3. 시연



06

회고

6. 회고

-
- 염윤호** Repository, Service, Controller 구조 각각의 역할을 지키는 게 유지보수에 효과적이라는 것을 느꼈다.
팀 내에서 Mybatis와 JPA를 병행해서 사용하였다.
우리는 Repository에서 데이터 값을 받아오는 역할, Service에서 값을 가공하는 등 비즈니스 로직의 역할, Request를 받아 Response 보내는 역할에 충실했도록 약속을 하고 시작했다. 그 결과 Mybatis와 JPA 어떤 기술을 적용하든 구조 각각의 역할을 지키면 크게 달라지지 않는 것을 느꼈다.
만약 차후 프로젝트에서 JPA로 Refactoring을 하더라도 어렵지 않게 잘 바꿀 수 있을 것이라고 생각했다.
- 고민표** JUnit을 활용한 테스트 코드를 미리 잘 구성해두면 코드 수정 시 수동 테스트보다 빠르게 정상 작동 여부를 확인할 수 있어 개발 효율성이 높아졌습니다.
또한 상호 코드리뷰와 일관성 문서 작성으로 코드의 통일성을 유지하며, 협업에서 명확한 의사소통의 중요성을 느꼈습니다.
- 박희수** 이번 프로젝트에서는 MyBatis 대신 JPA를 사용하면서 개발 효율을 높일 수 있었고, Spring Boot를 활용해 설정이나 환경 구성을 훨씬 간편하게 할 수 있었다. 또한 Lombok 덕분에 반복적인 코드를 줄일 수 있어서 코드 가독성도 좋아졌다.
전체적으로 진행하면서 작은 기능 하나를 바꾸는 일조차 전체 시스템에 영향을 줄 수 있다는 것을 경험하면서 기능을 좀 더 분리하고 독립적으로 운영할 수 있는 구조의 필요성도 느꼈다.

6. 회고

이재원

지난 1차 스프린트에서는 업무·기술 합의가 부족해 데이터 모델링 논의 시 서로 다른 관점을 공유하지 못했고, 종종 “무슨 얘기인지 모르겠다”는 혼란이 발생했습니다.

그래서 2차 스프린트 시작 전에는, 목표를 간단히 설명하고 용어를 미리 정의하며 회의 전에 자료를 지라에 공유하는 규칙을 정했습니다. 그 결과, 서로의 관점을 빠르게 파악할 수 있어 의사소통이 매끄러워졌고, 데이터 모델링도 이전보다 훨씬 수월하게 진행되었습니다.

이창훈

프로젝트 아키텍처를 Controller, Service, Repository로 분리하고, 각 레이어에서 해야 할 역할을 팀원들끼리 내부적으로 정의했습니다. 덕분에 외부 API나 OAuth2 같은 여러 라이브러리를 사용하더라도 “어느 레이어에서 동작하는지”를 업무로 직접 맡지 않아도 파악할 수 있었습니다.

GitHub 협업 과정에서 브랜치를 나누고 Pull Request를 진행할 때, 저는 팀원들의 코드를 가장 많이 검토하고 Merge해 주었습니다. 1차 프로젝트와 달리 팀원들의 코드가 훨씬 읽기 쉬워진 점이 가장 좋았습니다.

제가 API를 담당했는데, 아키텍처 구조상 Repository 레이어에서 통신을 처리하고 Service 레이어에서 비즈니스 로직을 구현해야 했습니다. 처음에는 그 흐름이 다소 헷갈렸지만, 큰 틀 안에서 고민하며 작업을 이어가다 보니 자연스럽게 문제를 해결할 수 있었습니다.