

취약 계층을 위한 음식 나눔 냉장고 서비스

발표일 : 2025/05/01

발표자 : 차민건

프로젝트 개요

Project Overview

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable



프로젝트명

취약계층을 위한
음식 나눔 서비스



프로젝트 목표

유통기한 임박 음식물을
취약계층과 연결해
음식물 낭비를 줄이고
식생활을 지원하는 서비스



프로젝트 배경

가정 내 음식물 쓰레기 증가와
취약 계층의 식품 접근 어려움을
해결하기 위해

프로젝트 내용

JAVA와 Oracle SQL을
사용하여 MVC 구조로
서비스 제작

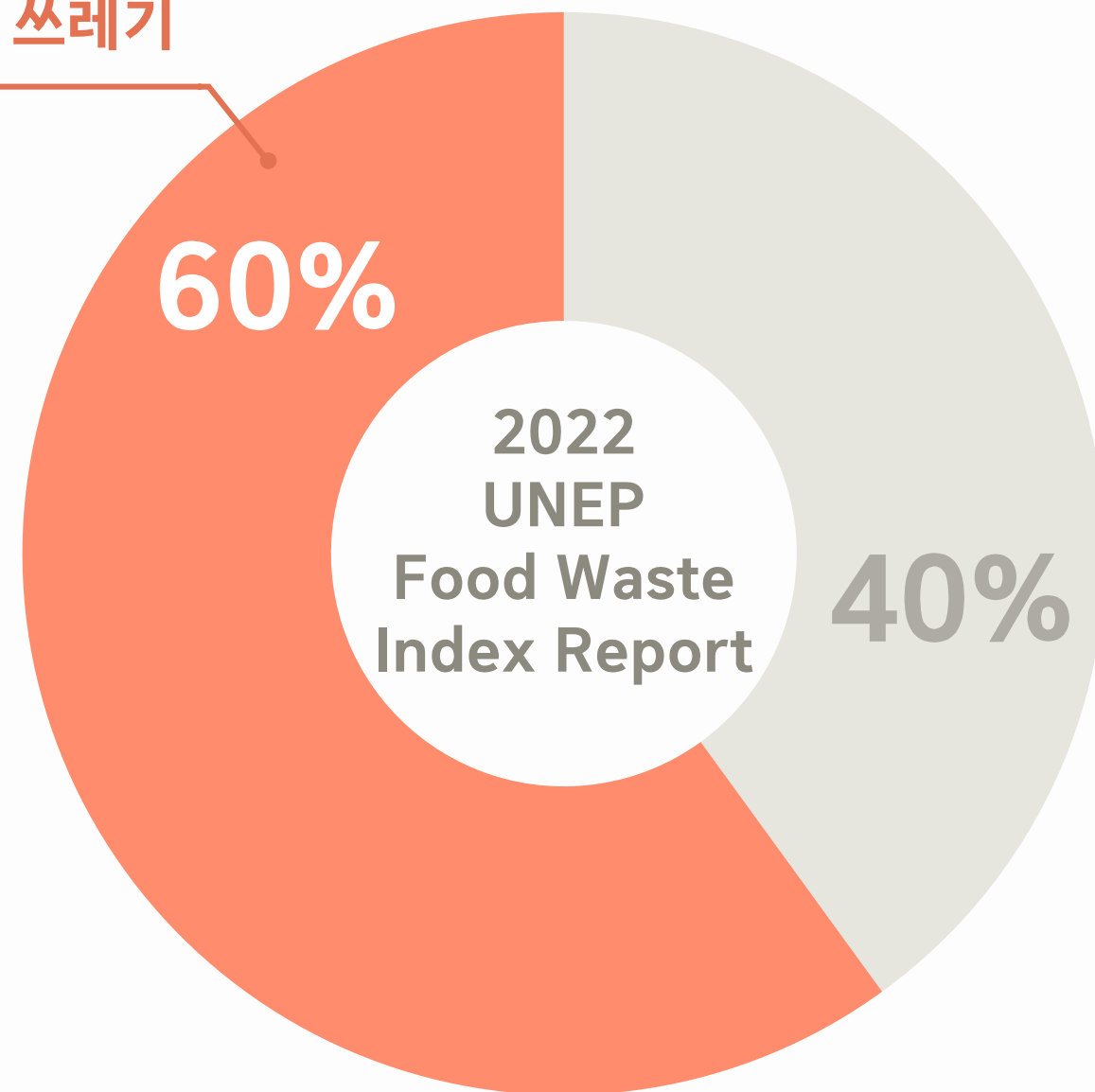
- ER 다이어그램 제작
- Oracle SQL 작성
- JAVA로 MVC 구조 서비스 제작

현황 분석

Current Status Analysis

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable

가정 음식물 쓰레기



- 버려진 식품 10억 5000만 톤 중 60%를 차지하는 약 6억 3100만 톤이 가정 음식물 쓰레기
- 가정에서 평균 55.0kg 음식물 보관, 이 중 6.0%가 버려짐
2012년 국립환경과학원
- 현재 정부에서 취약 계층에게 농식품 바우처 제공

개발 환경 및 시스템 흐름도

Development Environment and System Flowchart

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable

개발 환경

사용 언어

JAVA 17



Oracle SQL



사용 프로그램

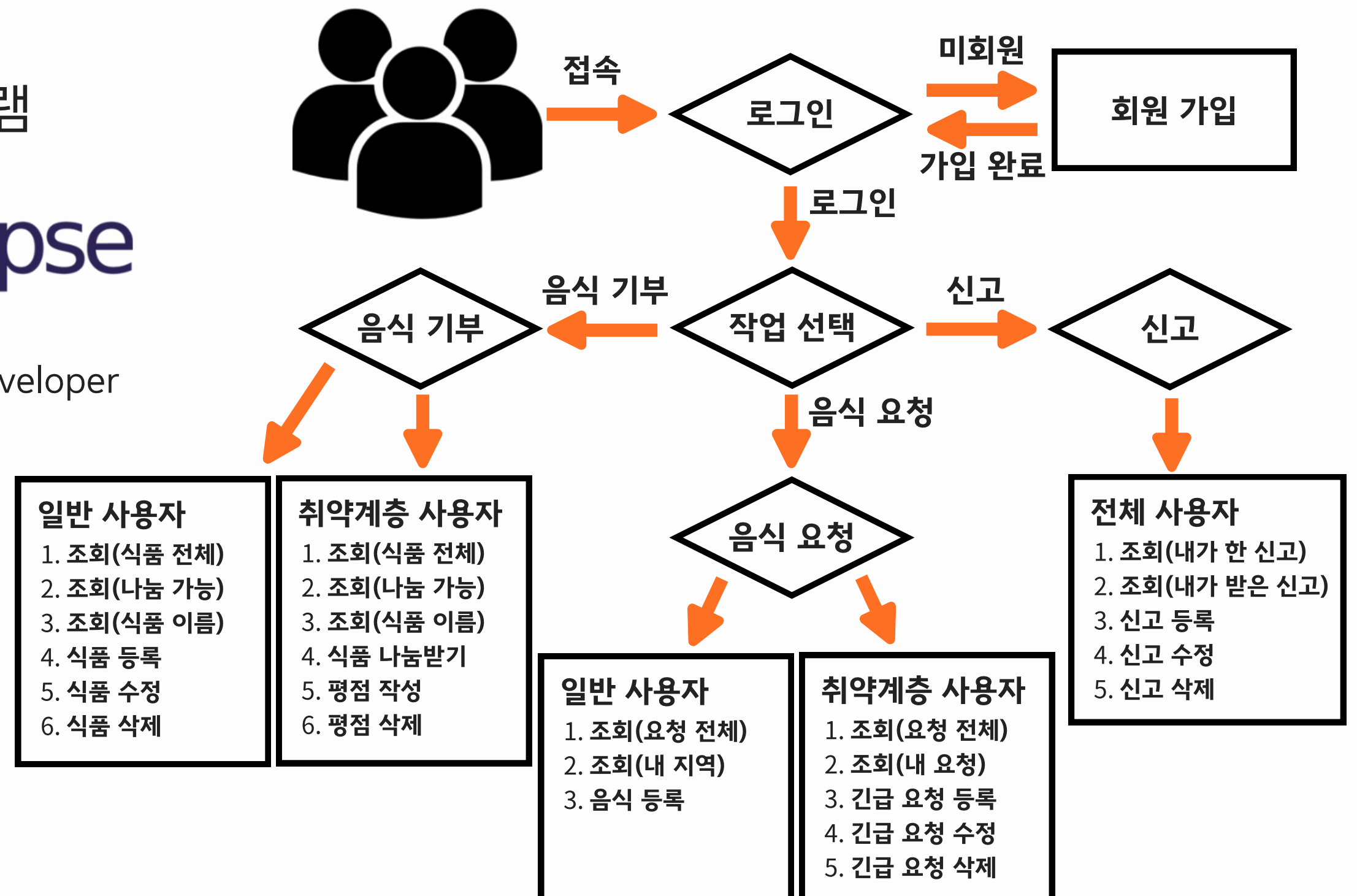
eclipse



Oracle SQL Developer



시스템 흐름도



Controller-Service-DAO-DTO

Controller-Service-DAO-DTO

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable

01 Controller

1. FrontController
2. EmergencyRequestController
3. FoodController
4. LoginController
5. ReportController

02 Service

1. EmergencyRequestService
2. FoodService
3. RatingService
4. ReportService
5. UsersService

03 DAO

1. EmergencyRequestDAO
2. FoodDAO
3. RatingDAO
4. ReportDAO
5. SharingDAO
6. UsersDAO

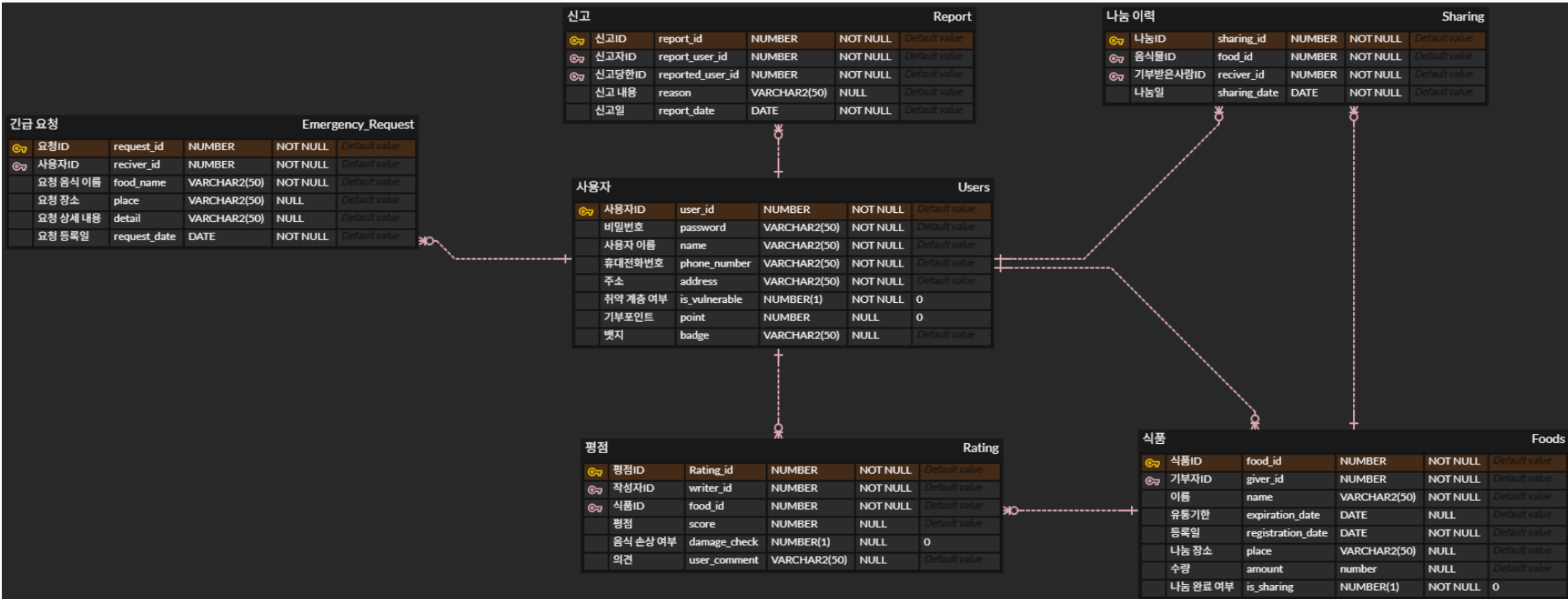
04 DTO

1. DTOParent
2. EmergencyRequestDTO
3. FoodDTO
4. RatingDTO
5. ReportDTO
6. SharingDTO
7. UsersDTO

ER-다이어그램

ER-Diagram

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable



자동 ID/입력 날짜 생성 트리거

Automatic ID/Enter Date Generation Trigger

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable

```
drop sequence user_id_sequence;
create sequence user_id_sequence;

CREATE OR REPLACE TRIGGER user_id_trigger
before insert on users
for each row
begin
    if :NEW.user_id IS NULL THEN
        :new.user_id := user_id_sequence.NEXTVAL;
    end if;
end;
/
```

```
drop sequence food_id_sequence;
create sequence food_id_sequence;

CREATE OR REPLACE TRIGGER food_id_trigger
before insert on foods
for each row
begin
    if :NEW.food_id IS NULL THEN
        :new.food_id := food_id_sequence.NEXTVAL;
        :new.REGISTRATION_DATE := sysdate;
        :new.IS_SHARING := 0;
    end if;
end;
/
```

```
drop sequence sharing_id_sequence;
create sequence sharing_id_sequence;

CREATE OR REPLACE TRIGGER sharing_id_trigger
BEFORE INSERT ON Sharing
FOR EACH ROW
BEGIN
    IF :NEW.sharing_id IS NULL THEN
        :NEW.sharing_id := sharing_id_sequence.NEXTVAL;
        :NEW.sharing_date := SYSDATE;
    END IF;
END;
/
```

```
drop sequence report_id_sequence;
create sequence report_id_sequence;

CREATE OR REPLACE TRIGGER report_id_trigger
before insert on report
for each row
begin
    if :NEW.report_id IS NULL THEN
        :new.report_id := report_id_sequence.NEXTVAL;
        :new.report_DATE := sysdate;
    end if;
end;
/
```

```
drop sequence request_id_sequence;
create sequence request_id_sequence;

CREATE OR REPLACE TRIGGER request_id_trigger
before insert on emergency_request
for each row
begin
    if :NEW.request_id IS NULL THEN
        :new.request_id := request_id_sequence.NEXTVAL;
        :new.request_DATE := sysdate;
    end if;
end;
/
```

```
drop sequence rating_id_sequence;
create sequence rating_id_sequence;

CREATE OR REPLACE TRIGGER rating_id_trigger
before insert on rating
for each row
begin
    if :NEW.rating_id IS NULL THEN
        :new.rating_id := rating_id_sequence.NEXTVAL;
    end if;
end;
/
```

업무 로직 트리거

Business Logic Trigger

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable

```
CREATE OR REPLACE TRIGGER update_giver_point_trigger
AFTER UPDATE ON foods
FOR EACH ROW
BEGIN
    IF :OLD.amount > :NEW.amount THEN
        UPDATE users
        SET point = point + ((:OLD.amount - :NEW.amount) * 10)
        WHERE user_id = :NEW.giver_id;
    END IF;
END;
/
```

기부된 식품 수량 1개 당 10점씩 증가

```
CREATE OR REPLACE TRIGGER is_sharing_trigger
BEFORE UPDATE ON foods
FOR EACH ROW
BEGIN
    IF :NEW.amount = 0 THEN
        :NEW.is_sharing := 1;
    END IF;
END;
/
```

식품 수량이 0이 되면 식품 수령 불가

```
CREATE OR REPLACE TRIGGER user_badge_trigger
BEFORE UPDATE ON users
FOR EACH ROW
BEGIN
    IF :NEW.point > 500 THEN
        :NEW.badge := '숲 기부자';
    ELSIF :NEW.point > 400 THEN
        :NEW.badge := '나무 기부자';
    ELSIF :NEW.point > 300 THEN
        :NEW.badge := '가지 기부자';
    ELSIF :NEW.point > 200 THEN
        :NEW.badge := '앞새 기부자';
    ELSIF :NEW.point > 100 THEN
        :NEW.badge := '씨앗 기부자';
    ELSIF :NEW.point > 0 THEN
        :NEW.badge := '새싹 기부자';
    ELSE
        :NEW.badge := '없음';
    END IF;
END;
/
```

포인트에 따라 뱃지 획득

프로그램 화면

Program View

초기 화면

```
-----
1. 로그인 2. 회원 가입 3. 종료
-----
번호 입력 >
```



주요 기능

- 목록에 없는 기능 입력시 예외 처리

```
-----
1. 로그인 2. 회원 가입 3. 종료
-----
번호 입력 > 로그인
숫자만 입력해주세요.
번호 입력 > 5
알림 > 서비스 준비중입니다.
-----
1. 로그인 2. 회원 가입 3. 종료
-----
번호 입력 >
```

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable

회원 가입

```
-----
1. 로그인 2. 회원 가입 3. 종료
-----
번호 입력 > 2
id 입력 > hihi
password 입력 > hihi
이름 입력 > 홍길동
전화번호 입력(010-####-####) > 010-1234-5678
주소 입력 > 홍대
취약 계층 여부 입력 (취약 계층일 경우 1) > 1
알림 > 회원 가입 완료
```



주요 기능

- 각 형식에 맞지 않을 경우 예외 처리

```
-----
1. 로그인 2. 회원 가입 3. 종료
-----
번호 입력 > 2
id 입력 > byebye
password 입력 > byebye
이름 입력 > 홍길순
전화번호 입력(010-####-####) > 01056781234
전화번호 형식이 잘못되었습니다. 010-####-#### 형식으로 입력해주세요.
전화번호 입력(010-####-####) > 010-5678-1234
주소 입력 > 건대
취약 계층 여부 입력 (취약 계층일 경우 1) > 0
알림 > 회원 가입 완료
```

프로그램 화면

Program View

로그인

1. 로그인 2. 회원 가입 3. 종료

```
번호 입력 > 1
id 입력 > hihi
password 입력 > hihi
알림 > 로그인 완료
알림 > hihi님 접속을 환영합니다.
```

1. 음식 기부 21. 로그인 2. 회원 가입 3. 종료

```
번호 입력 > 1
id 입력 > byebye
password 입력 > byebye
알림 > 로그인 완료
알림 > 없음 등급 byebye님 접속을 환영합니다.
```



주요 기능

- 잘못된 ID 비밀번호 입력시 로그인 실패
- 회원에 따른 환영인사

1. 로그인 2. 회원 가입 3. 종료

```
번호 입력 > 1
id 입력 > hihi
password 입력 > byebye
알림 > 로그인 실패
```

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable



음식 기부 메뉴



일반 사용자

1. 음식 기부 2. 음식 요청 3. 신고 99. 종료

```
번호 입력 > 1
```

1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료

```
번호 입력 >
```



취약계층 사용자

1. 음식 기부 2. 음식 요청 3. 신고 99. 종료

```
번호 입력 > 1
```

1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 나눔 받기 5. 평점 작성 6. 평점 삭제 99. 종료

```
번호 입력 >
```



주요 기능

- 사용자에 따라 기능 변경

프로그램 화면

Program View



음식 요청 메뉴

일반 사용자

1. 음식 기부 2. 음식 요청 3. 신고 99. 종료

번호 입력 > 2

1. 조회(요청 전체) 2. 조회(내 지역) 3. 음식 등록 99. 종료

번호 입력 > _

취약계층 사용자

1. 음식 기부 2. 음식 요청 3. 신고 99. 종료

번호 입력 > 2

1. 조회(요청 전체) 2. 조회(내 요청) 3. 긴급 요청 등록 4. 긴급 요청 수정 5. 긴급 요청 삭제 99. 종료

번호 입력 > _



주요 기능

- 사용자에게 따라 기능 변경

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable



신고 메뉴

1. 음식 기부 2. 음식 요청 3. 신고 99. 종료

번호 입력 > 3

1. 조회(내가 한 신고) 2. 조회(내가 받은 신고) 3. 신고 등록 4. 신고 수정 5. 신고 삭제 99. 종료

번호 입력 > _

CRUD 기능 구현

Program View

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable



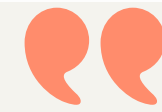
Create

```
-----
1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료
-----
번호 입력 > 4
※ 값이 없으면 '0'을 입력하세요 (→ null 처리됨)
식품명(필수 입력) > 우유
유통기한 (yyyy-MM-dd) > 20250601
날짜 형식이 잘못되었습니다. yyyy-MM-dd 형식으로 입력해주세요.
유통기한 (yyyy-MM-dd) > 2025-06-01
나눔 장소 > 건대
식품 수량 > 10개
숫자만 입력해주세요.
식품 수량 > 10
알림 > 1 개의 음식을 등록했습니다.
```



주요 기능

- 요구 형식 외 입력 예외처리
- null 가능한 입력값 0 입력시 null 처리



Read

```
-----
1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료
-----
번호 입력 > 2
=====식품 여러건 조회=====
FoodDTO(food_id=2, giver_id=2, name=우유, expiration_date=2025-06-01, registration_date=2025-05-01, place=건대, amount=10, is_sharing=true)
FoodDTO(food_id=1, giver_id=2, name=라면, expiration_date=2026-06-01, registration_date=2025-05-01, place=홍대, amount=10, is_sharing=true)
-----
```

```
-----
1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료
-----
번호 입력 > 3
음식명 > 쌀
알림 > 해당하는 데이터가 존재하지 않습니다.
```



주요 기능

- 존재하지 않는 데이터 조회 시 예외처리

CRUD 기능 구현

Program View

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable



Update

```
-----
1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료
-----
번호 입력 > 5
=====식품 여러건 조회=====
FoodDTO(food_id=1, giver_id=2, name=라면, expiration_date=2026-06-01, registration_date=2025-05-01, place=홍대, amount=10, is_sharing=true)
수정하고싶은 음식 id 입력 > 1
※ 값이 없으면 '0'을 입력하세요 (→ null 처리됨)
식품명(필수 입력) > 쌀
유통기한 (yyyy-MM-dd) > 2025-07-25
나눔 장소 > 건대
식품 수량 > 5
알림 > 1 개의 음식을 수정했습니다.
-----
1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료
-----
번호 입력 > 1
=====식품 여러건 조회=====
FoodDTO(food_id=1, giver_id=2, name=쌀, expiration_date=2025-07-25, registration_date=2025-05-01, place=건대, amount=5, is_sharing=true)
```



주요 기능

- 현재 로그인한 유저의 정보만 조회
- 요구 형식 외 입력 예외처리
- null 가능한 입력값 0 입력시 null 처리



Delete

```
-----
1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료
-----
번호 입력 > 6
=====식품 여러건 조회=====
FoodDTO(food_id=1, giver_id=2, name=라면, expiration_date=2026-06-01, registration_date=2025-05-01, place=홍대, amount=10, is_sharing=true)
FoodDTO(food_id=2, giver_id=2, name=우유, expiration_date=2025-06-01, registration_date=2025-05-01, place=건대, amount=10, is_sharing=true)
음식 id 선택 > 2
알림 > 1 개의 음식을 삭제했습니다
-----
1. 조회(식품 전체) 2. 조회(나눔 가능) 3. 조회(식품 이름) 4. 식품 등록 5. 식품 수정 6. 식품 삭제 99. 종료
-----
번호 입력 > 6
=====식품 여러건 조회=====
FoodDTO(food_id=1, giver_id=2, name=라면, expiration_date=2026-06-01, registration_date=2025-05-01, place=홍대, amount=10, is_sharing=true)
음식 id 선택 > 5
알림 > 선택한 음식이 존재하지 않습니다.
```



주요 기능

- 현재 로그인한 유저의 정보만 조회
- 존재하지 않는 데이터 삭제 시 예외처리

기대효과

Expected Effects

취약계층을 위한 음식 나눔 서비스
Food Sharing Refrigerator Service
for the Vulnerable

01

취약 계층 식품 지원

부족한 전체 식품과 신선 식품을 지원

02

음식물 쓰레기 감소

가정에서 식품의 포장 단위가 커,
버리는 것이 아까웠던 식품을 공유

03

서비스 확장

생필품, 의류, 생활 지원, 공공 의료 등
다양한 서비스 확장

04

공공기관, 지자체 연계 가능성

사회 공헌 서비스를 통해 사회 문제 해결