

Primavera

새로운 생명의 시작

'개발의 봄' 조
성공하면 혁명이고 실패하면 반역이다

조장 : 권예빈
조원 : 김현수, 손민정, 정대윤



Index

1. 프로젝트 착수

- 1) 주제 및 선정 이유
- 2) 팀 소개
- 3) 작업 흐름
- 4) 개발 환경 및 기술 스택

2. 설계 및 분석

- 1) 요구사항 분석
- 2) 논리적&물리적 모델링
- 3) MVC 패턴
- 4) 프로젝트 구조

3. UI 및 기능 구현

- 1) UI 화면구현
- 2) 기능&코드리뷰

4. 시연 및 마무리

- 1) 각 팀원별 시연
- 2) 느낀 점 & 개선방안



01_프로젝트 착수

Primavera_봄의 시작

1_선정이유

전 세계에서 1년 간 제조 되는 의류량

약 1000억 벌

전 세계에서 1년간 버려지는 의류량

약 9200 만 톤

의류 제조에 드는 세계 플라스틱 생산량 비중

20%

전세계 폐수 중 의류제조 과정 배출 비중

20%

합성 섬유 폴리에스터 분해 시간

200년

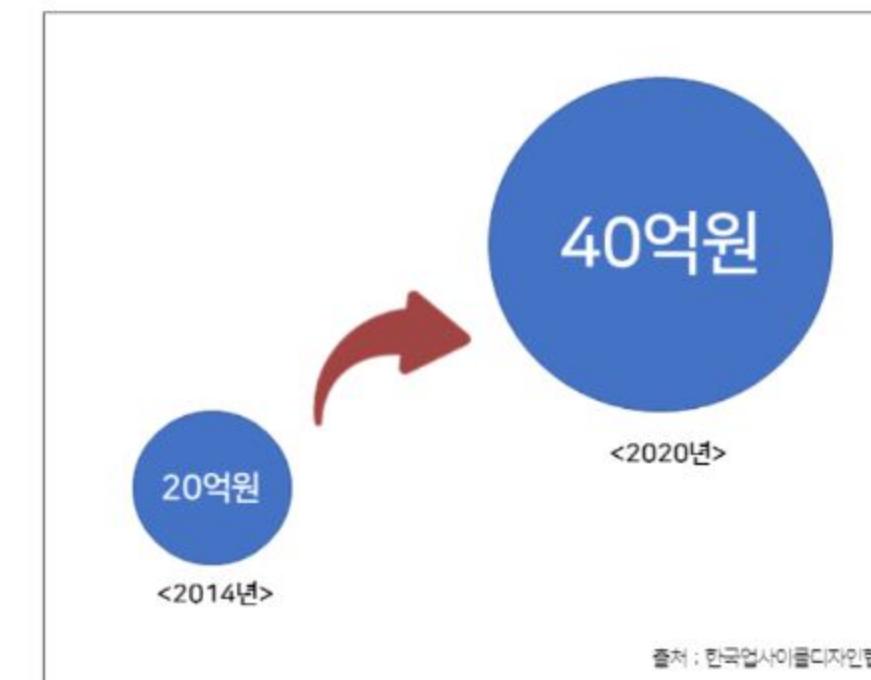
한국 현옷 배출량

세계 5위

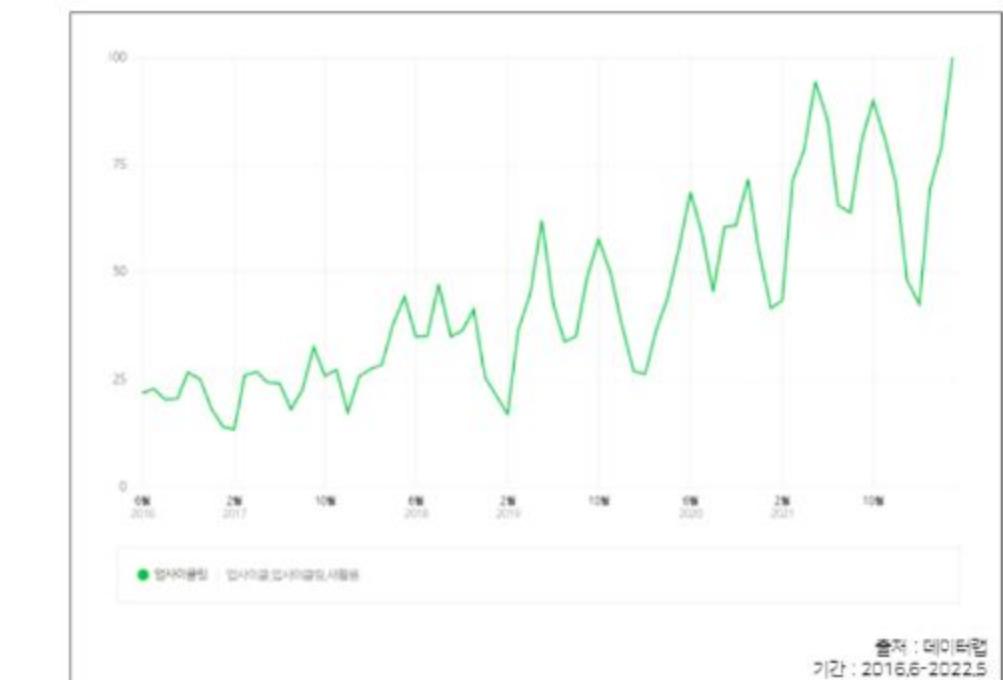
선정 이유

일상에서 흔히 보이는 의류의 산업적 과잉생산과 가정에서의 재활용 부재로 인해 발생하는 의류 폐기물이 지구온난화의 10%를 차지한다는 사실을 알게되면서 이 문제에 대한 인식을 높이고 대응책을 모색하기 위해 선정하게 되었습니다.

국내 업사이클링 시장 규모



'업사이클링' 월간 검색 빈도 수



1_주제 & 회사소개



주제

의류 폐기물을 재활용하여 탄소 배출량을 줄이는 업사이클링의 환경적 가치와 중요성.

소개

'Primavera'는 '진정한 봄'이라는 뜻의 이탈리아어로 새로운 생명의 시작과 자연의 활기를 뜻합니다. 저희 'Primavera'는 심각한 지구 온난화와 기후 변화로 인한 피해를 줄이고자 탄소 배출량을 줄일 수 있는 폐의류 업사이클링을 통해 환경과 자연의 피해를 줄이고 이들의 원래 모습을 되찾기 위한 길을 걷고 있습니다

2_ 팀 소개



권예빈

조장 각하_ISFP

- 총괄
- 로그인 및 회원가입
- 회원/관리자 페이지
- 회원 관리

starting0809@gmail.com

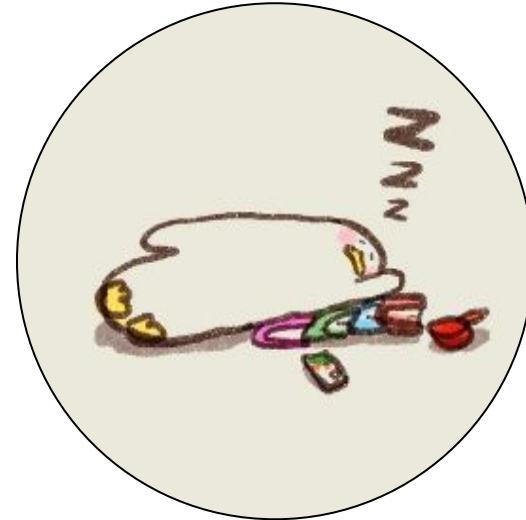


김현수

분위기 메이커_INFJ

- 메인 페이지 UI
- NOTICE 페이지
UI 및 기능 구현
- GUIDE 페이지 UI

kjwgustn123@naver.com



손민정

팀의손_ESFP

- 수거 신청 관리
- PICK UP 페이지 UI
- Q&A 페이지 UI 및 기능 구현
- STORE 페이지 UI

sonmj0424@naver.com



정대윤

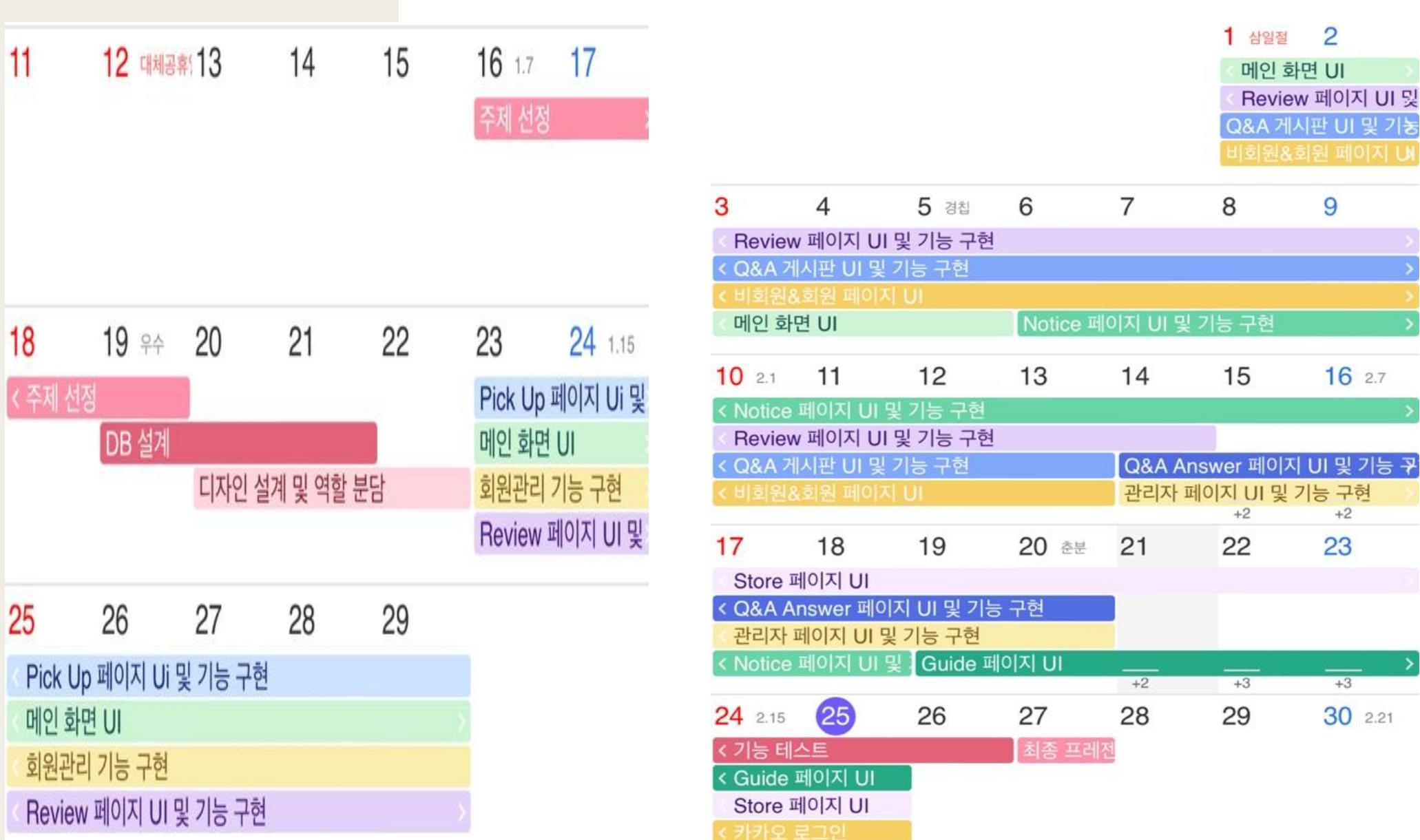
가보자고_INTP

- REVIEW 페이지
UI 및 기능 구현
- STORE 페이지
UI 및 기능 구현

eodbs1774@gmail.com

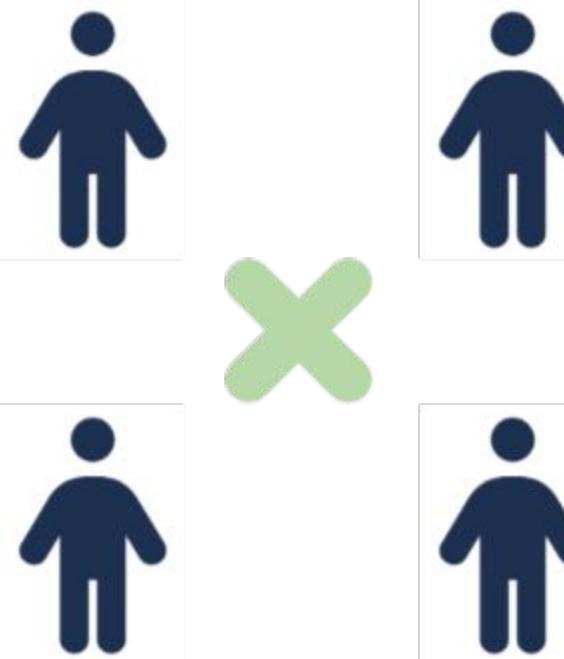
3_작업 흐름

총 프로젝트 기간 : 24.02.15~24.03.27(약 6주)



! 프로젝트 시작	2024년 2월 15일
프로젝트 주제 선정	2024년 2월 16일 ~ 2024년 2월 19일
DB 설계	2024년 2월 19일 ~ 2024년 2월 21일
디자인 설계 및 역할 분담	2024년 2월 20일 ~ 2024년 2월 22일
메인 화면 UI	2024년 2월 23일 ~ 2024년 3월 5일
Pick Up 페이지 UI 및 기능 구현	2024년 2월 23일 ~ 2024년 2월 29일
회원관리 기능 구현	2024년 2월 23일 ~ 2024년 2월 29일
Review 페이지 UI 및 기능 구현	2024년 2월 23일 ~ 2024년 3월 14일
Q&A 페이지 UI 및 기능 구현	2024년 3월 1일 ~ 2024년 3월 13일
회원/비회원 페이지 UI	2024년 3월 1일 ~ 2024년 3월 13일
Notice 페이지 UI&기능 구현	2024년 3월 6일 ~ 2024년 3월 18일
Q&A Answer 페이지 UI 및 기능 구현	2024년 3월 14일 ~ 2024년 3월 20일
관리자 페이지 UI 및 기능 구현	2024년 3월 14일 ~ 2024년 3월 20일
Store 페이지 UI 및 기능 구현	2024년 3월 15일 ~ 2024년 3월 25일
Guide 페이지 UI	2024년 3월 19일 ~ 2024년 3월 25일
카카오 로그인	2024년 3월 21일 ~ 2024년 3월 25일
! 프로젝트 마감	2024년 3월 26일

3_작업 흐름



(주) - 전체 회의

(부) - 특정 기능 및 디자인 구현 등 필요에 따라
팀원 간 협력 및 소통 후 전체 회의



팀원 Commit&Push

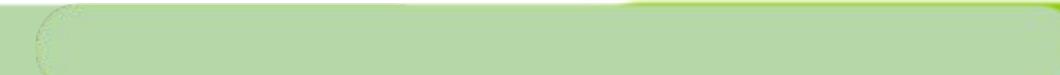
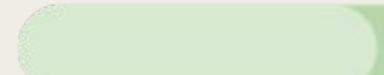
-> 다 함께 결과물 확인 후 팀장이 Merge, 각 팀원이 Pull

09:00 10:00

16:30

17:30

18:00



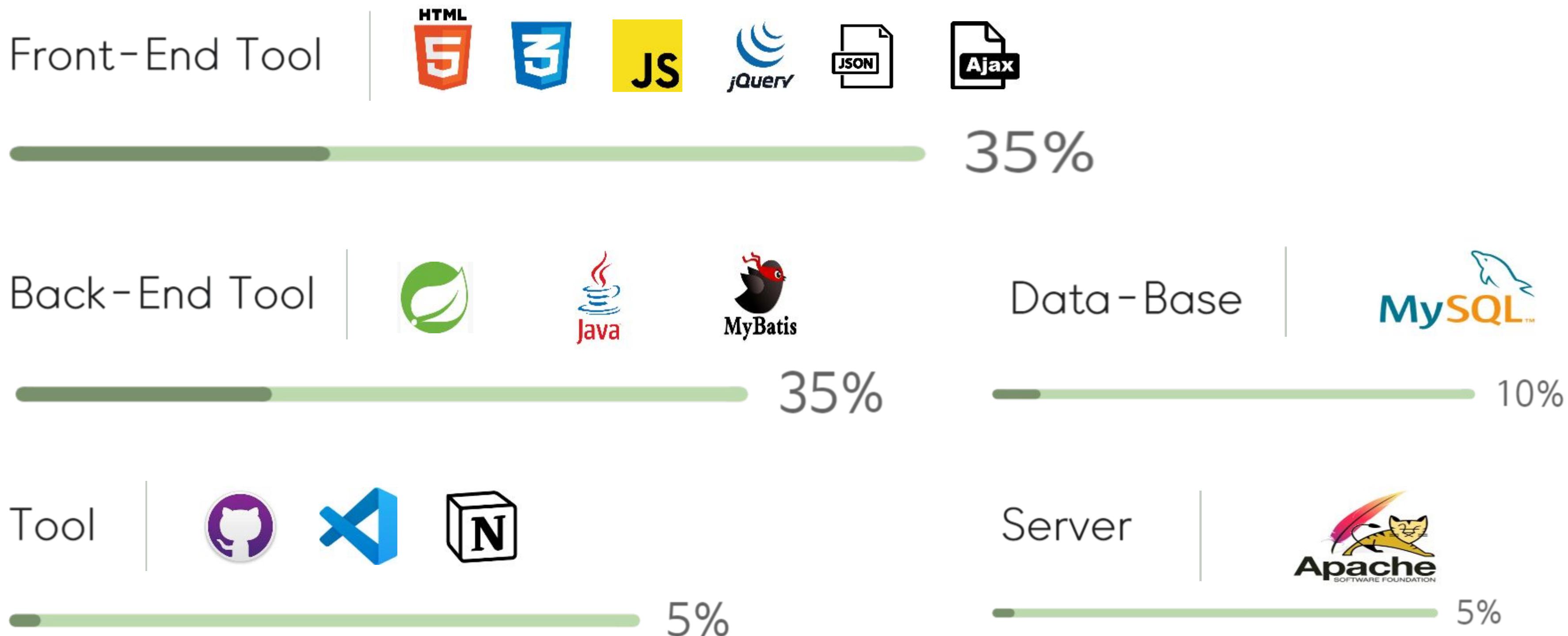
질문 및 변경사항 공유

각자 맡은 역할 수행

Commit 후 작업 결과 공유 및 모니터링
내일 할 일 계획 및 의견 공유

GitHub Push&Pull

4_ 개발환경 및 기술 스택





02_설계 및 분석

Primavera_봄의 시작

1_ 요구 사항분석

▽ 분류	▽ 카테고리	Aa 요구사항명	☰ 요구사항 상세	▽ 중요도	<input checked="" type="checkbox"/> 완성
회원관리	기능	회원가입	중복 아이디, 이메일, 휴대폰 번호 회원 가입 불가능	상	<input checked="" type="checkbox"/>
회원관리	기능	회원 정보 수정	비밀번호 확인 후 정보 수정 가능하도록 수정 시에도 중복되지 않도록	상	<input checked="" type="checkbox"/>
카카오 회원관리	기능	회원 탈퇴	비밀번호 확인 과정을 거친 후 회원 탈퇴 가능하도록	상	<input checked="" type="checkbox"/>
회원관리	기능	카카오 로그인	카카오 로그인 가능	중	<input checked="" type="checkbox"/>
카카오 회원관리	기능	강제 정보 수정	pickUp 신청 시도 시 정보 수정 창으로 넘어가서 전화번호, 비밀번호를 설정하도록	중	<input checked="" type="checkbox"/>
카카오 회원관리	기능	카카오 회원 탈퇴	비밀번호를 설정하지 않은 상태로 회원 탈퇴를 시도할 때 바로 탈퇴되도록	중	<input checked="" type="checkbox"/>
회원관리	기능	로그아웃	로그아웃 후 첫 메인 페이지로 돌아가도록	상	<input checked="" type="checkbox"/>
회원관리	기능	아이디 찾기	이름과 이메일을 입력받아 두 정보가 일치하는 회원의 아이디를 도출하도록	상	<input checked="" type="checkbox"/>
회원관리	기능	비밀번호 찾기	존재하는 아이디일 경우 이메일 입력 가능하도록, 회원의 아이디와 이메일이 일치하는 경우 이메일 전송 기능이 구현되도록, 이메일로 전송된 인증번호와 회원이 입력한 인증번호가 같을 때 비밀번호 변경이 가능하도록	상	<input checked="" type="checkbox"/>
회원관리	기능	로그인 실패 메세지	존재하지 않는 회원, 아이디/비밀번호 불일치 로그인 시도 등 경우에 따라 경고 메세지가 달라지도록	상	<input checked="" type="checkbox"/>
회원메뉴	기능	내 review	회원이 myPage에서 본인이 적은 review들을 모아볼 수 있도록	상	<input checked="" type="checkbox"/>
회원메뉴	기능	내 Q&A	회원이 myPage에서 본인이 적은 review들을 모아볼 수 있도록	상	<input checked="" type="checkbox"/>
회원메뉴	기능	진행상황	회원이 신청한 pickUp의 진행상황을 모아볼 수 있도록	상	<input checked="" type="checkbox"/>
관리자메뉴	기능	Q&A 게시판 관리	관리자가 답변하지 않은 Q&A 리스트를 모아서 볼 수 있도록	상	<input checked="" type="checkbox"/>
관리자메뉴	기능	전체 회원 조회	관리자를 포함한 모든 유저 정보를 볼 수 있도록	 상	<input checked="" type="checkbox"/>
관리자메뉴	기능	수거 신청 현황 관리	 열기 신청 완료, 수거 완료, 수거 물품 확인, 입금 진행 중, 입금 완료 총 5단계로 진행 상황에 따라 관리자가 업데이트를 할 수 있도록	상	<input checked="" type="checkbox"/>
관리자메뉴	기능	전체 수거 신청 정보 보기	주문번호, 이름, 수거일, 수거 무게 순으로 전체 정보가 선택 정렬 가능하도록	상	<input checked="" type="checkbox"/>
myPage 페이지	비기능	카드 형식으로 화면 분리	로그인, 회원가입 창이 분리되어 카드 형식으로 뒤집히도록	상	<input checked="" type="checkbox"/>
myPage 페이지	비기능	권한에 따라 배경 이미지 변경	MEMBER의 경우와 ADMIN의 경우 마이페이지 메뉴 뒤에 보이는 배경 사진이 달랐으면 좋겠으나, 메뉴들의 위치, 보이는 형식은 같게	상	<input checked="" type="checkbox"/>

1_ 요구 사항분석

▼ 분류	▼ 카테고리	Aa 요구사항명	☰ 요구사항 상세	▼ 중요도	<input checked="" type="checkbox"/> 완성
메인페이지	비기능	최종	요즘 보여주는 국내기업의 느낌보다는 외국기업의 홈페이지의 무드가 느껴졌으면 좋겠습니다!!	중	<input checked="" type="checkbox"/>
메인페이지	비기능	PDF 파일	저희가 생각하는 디자인에 대한 글들은 pdf파일에 있어서 그걸 기반으로 만들어 주시기 바랍니다!	중	<input checked="" type="checkbox"/>
메인페이지	비기능	글자 효과	첫 상단에 나오는 환경문구에 이동 느낌을 주었으면 좋겠습니다 그리고 목록과 내용 이동관련 글자들에 밀 줄 효과도 추가해주세요 !!	중	<input checked="" type="checkbox"/>
메인페이지	비기능	이동가능 버튼	우측에 관련 부분마다 바로 이동가능한 버튼 설정추가	하	<input checked="" type="checkbox"/>
메인페이지	비기능	이후 디자인	회사소개, 추구방향/ 이용가이드 소개, 서비스가능구역 / 마지막 하단부 저희 기업관련 정보들을 기제해주세요	중	<input checked="" type="checkbox"/>
메인페이지	비기능	디자인	상단 회사명과 주요 목록들을 고정 시키고 제일 상단은 저희기업에서 보낸 영상을 보여줄수있게	하	<input checked="" type="checkbox"/>
Notice 게시판	기능	페이지 이동 관련	다른 게시판관련 페이지와 동일하게 한페이지의 게시물개수를 10개로 맞춰주시고 전과후를 이동하는 기능을 탑재해주세요	중	<input checked="" type="checkbox"/>
Notice 게시판	기능	게시판	관리자가 공지사항을 등록했을시 등록 날짜 순으로 정렬	중	<input checked="" type="checkbox"/>
Notice 게시판	기능	비회원과 회원	게시판을 볼때 제목을 누르면 상세내용과 파일을 볼수있게 설정	중	<input checked="" type="checkbox"/>
Notice 게시판	기능	게시판 등록	게시물 등록시 파일 업로드와 제목과 내용의 길이 limit 설정	중	<input checked="" type="checkbox"/>
Notice 게시판	기능	관리자로그인/회원, 비회원	관리자일 경우 게시판을 등록하고 잘못 등록시 수정or삭제or목록으로 이동하는 아이콘과 설정	중	<input checked="" type="checkbox"/>
Notice 게시판	비기능	디자인	모든 페이지와 같은 공통을 주는 이미지를 두었으면 한다	하	<input checked="" type="checkbox"/>
Guide 페이지	비기능	슬라이드 형식의 순서	다른 페이지의 느낌과 다르게 가로로 이동하는 모션을 주었으면 좋겠다	중	<input checked="" type="checkbox"/>
Guide 페이지	비기능	배경사진의 중요도	저희 회사의 모티브를 반영하여 가이드 순서 마다 배경의 느낌을 다르게 주었으면 한다.	중	<input checked="" type="checkbox"/>
Guide 페이지	비기능	명확한 순서 전환	가이드의 title과 번호가 기제되었으면 한다.	중	<input checked="" type="checkbox"/>
Guide 페이지	비기능	가이드 순서	가이드라인 순서를 이미지와 내용을 이용해 보기 쉽게 하도록	하	<input checked="" type="checkbox"/>

1_ 요구 사항분석

▽ 분류	▽ 카테고리	Aa 요구사항명	≡ 요구사항 상세	▽ 중요도	☑ 완성
PickUp 페이지	기능	페이지 이동 관련	비회원의 경우 Pick up 버튼 클릭 시 login 페이지로 이동	중	<input checked="" type="checkbox"/>
PickUp 페이지	기능	수거 신청 관리	수거 신청 가능하도록 만들어주세요	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	회원 고정 사항	아이디, 이름, 전화번호는 변경 불가로 회원 정보가 고정값으로 들어가도록	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	권한	회원의 경우에만 pick up 신청이 가능하도록 해주세요	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	신청 시 필수값	아이디, 이름, 전화번호, 주소, 수거일, 수거 신청 무게 필수값으로 받아야 함	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	신청 시 주소_우편번호	api 사용해서 주소 신청	중	<input checked="" type="checkbox"/>
PickUp 페이지	기능	수거일자 제한	수거일은 최소 당일 +2일 / 최대 다음 해 말일로 신청 가능하도록 제한	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	신청 수정 및 삭제	신청 후 진행 상황이 신청 확인이 된 이후에는 수정 및 삭제가 불가능하도록	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	신청 리스트	관리자가 모든 신청 리스트 확인 가능하도록 / 각 회원이 자신의 신청 확인 가능하도록	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	수거 신청 현황 관리	관리자-관리자페이지에서 수거 신청 현황 확인 가능하며 주문번호에 따른 진행 상황 변경 가능하도록	상	<input checked="" type="checkbox"/>
PickUp 페이지	기능	회원의 수거 신청 진행 상황 열기	회원-마이페이지에서 자신의 신청번호 및 진행상황 확인 가능하도록	상	<input checked="" type="checkbox"/>
PickUp 페이지	비기능	디자인	각 단계 별 이미지 슬라이드 및 버튼 클릭 시 간략한 설명 보이도록	중	<input checked="" type="checkbox"/>
PickUp 페이지	비기능	이용약관 및 개인정보 수집 동의	수거 신청 위한 이용약관 및 개인정보 수집 동의 받아야 함	상	<input checked="" type="checkbox"/>
PickUp 페이지	비기능	신청 정보 확인 및 수정 디자인	신청 정보를 볼 수 있는 페이지와 수정 페이지가 품을 비슷한 느낌을 주게 해주세요	중	<input checked="" type="checkbox"/>

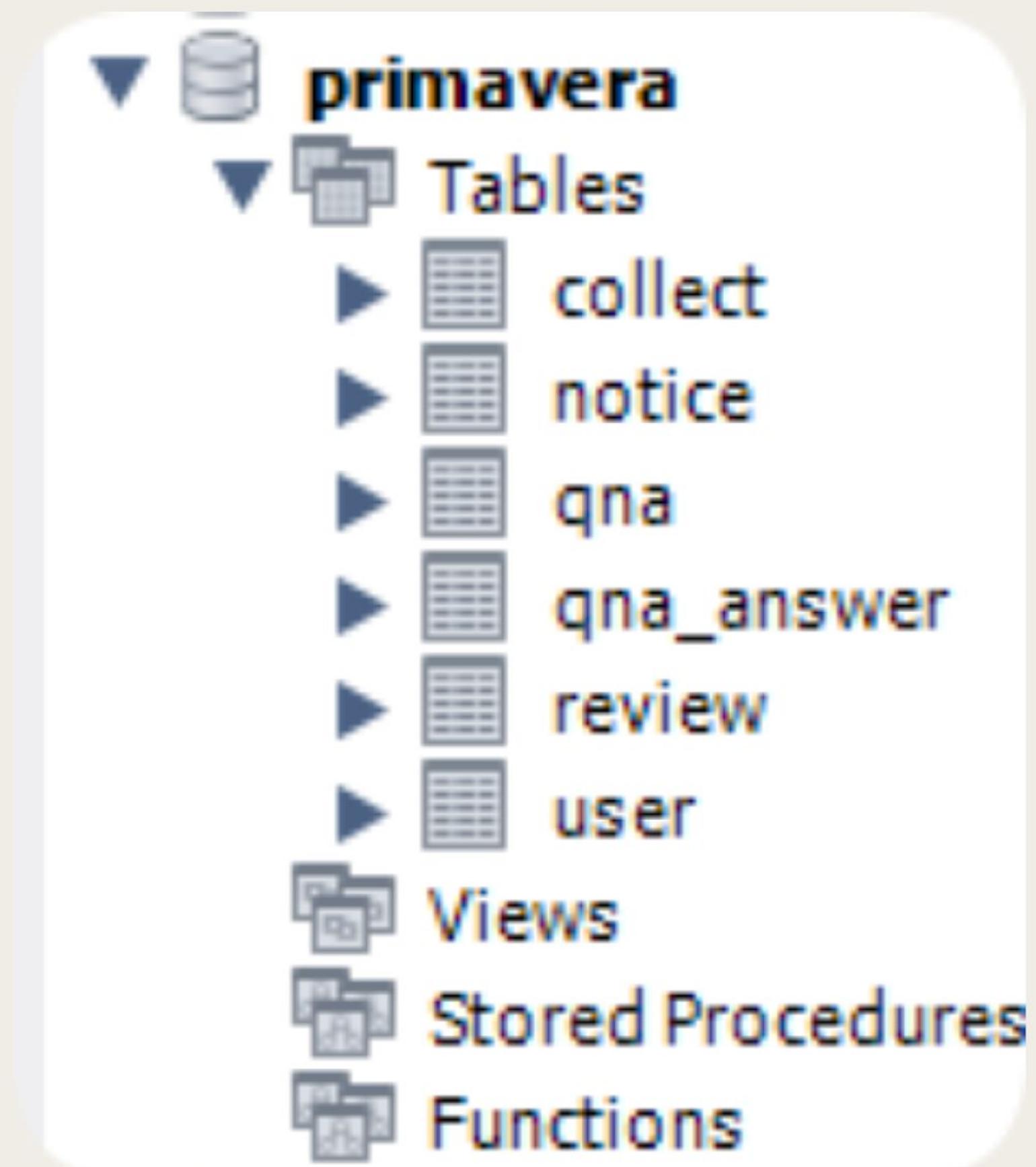
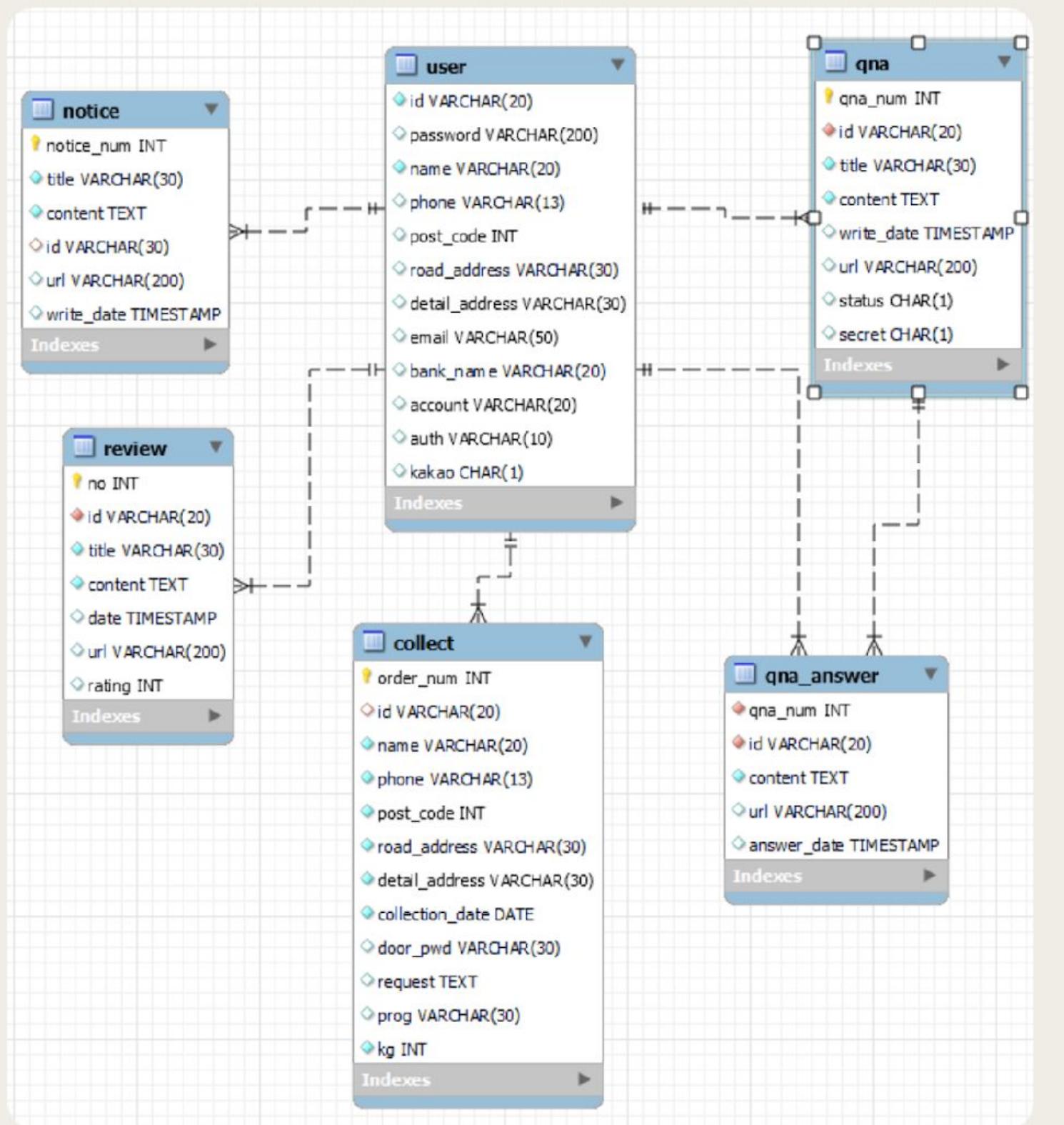
1_ 요구사항분석

▽ 분류	▽ 카테고리	Aa 요구사항명	☰ 요구사항 상세	▽ 중요도	<input checked="" type="checkbox"/> 완성
Q&A 게시판	기능	글쓰기 권한	Q&A 등록은 로그인 된 아이디만 가능하도록	상	<input checked="" type="checkbox"/>
Q&A 게시판	기능	비밀글	비밀글 선택 가능하고 글을 게시한 당사자의 비밀번호가 일치할 시에만 확인 가능하도록	상	<input checked="" type="checkbox"/>
Q&A 게시판	기능	비회원 비밀글 열람	비회원이 비밀글 열람하려고 할 경우 로그인 페이지로 이동하도록	중	<input checked="" type="checkbox"/>
Q&A 게시판	기능	수정, 삭제 권한	글 확인 시 작성한 당사자에게만 수정, 삭제 버튼이 보이도록 하고 아닐 경우 목록 버튼만 생성	상	<input checked="" type="checkbox"/>
Q&A 게시판	기능	관리자 글 확인 권한	관리자의 경우 비밀글 상관없이 모든 글을 열람할 수 있도록	중	<input checked="" type="checkbox"/>
Q&A 게시판	기능	관리자 댓글 권한	관리자가 글 확인 시 목록, 삭제, 댓글 버튼 생성하도록 하고 관리자만 댓글 작성 가능하도록	상	<input checked="" type="checkbox"/>
Q&A 게시판	기능	댓글 관리	관리자는 자신이 쓴 댓글 수정, 삭제 가능하도록	상	<input checked="" type="checkbox"/>
Q&A 게시판	기능	검색 기능	전체, 제목, 아이디로 검색 가능하도록	중	<input checked="" type="checkbox"/>
Q&A 게시판	기능	페이지	한 페이지에 게시물 10개씩 보이도록 하며 가장 최근에 작성한 게시물이 가장 상위에 오도록 진행(검색 시에 도 이와 동일하게)	상	<input checked="" type="checkbox"/>
Q&A 게시판	기능	댓글 답변 상태	관리자가 댓글 작성 할 시 Q&A 리스트에서 답변상태 Y로 변경(댓글 삭제 시 다시 N으로 돌리도록)	상	<input checked="" type="checkbox"/>
Q&A 게시판	기능	작성 게시물 확인	회원은 마이페이지에서 자신이 작성한 Q&A 확인 가능하도록 / 관리자는 관리자 페이지에서 답변이 달려있지 않은 게시물 리스트 보고 제목 클릭 시 해당 게시물로 넘어갈 수 있도록	상	<input checked="" type="checkbox"/>
Q&A 게시판	비기능	디자인	다른 게시판들과 비슷한 디자인이 비슷하도록 / 상단에 들어가는 사진이 깨끗한 자연의 느낌을 주었으면 좋겠다	상	<input checked="" type="checkbox"/>
Q&A 게시판	비기능	댓글 작성/수정 페이지 디자인	관리자가 댓글 작성 및 수정 시 한 화면에 기존 게시물 내용이 보였으면 좋겠음	중	<input checked="" type="checkbox"/>

1_ 요구 사항분석

▽ 분류	▽ 카테고리	Aa 요구사항명	☰ 요구사항 상세	▽ 중요도	<input checked="" type="checkbox"/> 완성
Store 페이지	비기능	제품 소개	각 품목 별 사진 3개와 상품명, 상품 상세 특징이 기재되었으면	상	<input checked="" type="checkbox"/>
Store 페이지	비기능	제품 상세 설명	사진과 상품명만 먼저 보이고 상품의 상세 특징은 가려져있다가 보였으면	중	<input checked="" type="checkbox"/>
Store 페이지	기능	지도(kakao map API)	사용자와 상호 작용할 수 있는 지도 기능, 상점이 표시되어야 함, 길 찾기 기능도 있었으면 좋겠음	상	<input checked="" type="checkbox"/>
Store 페이지	기능	지도 마커 관련 기능	3개의 마커 필요, 마커마다 해당 상점의 정보가 오버레이로 표시되어야 함	상	<input checked="" type="checkbox"/>
Review 게시판	비기능	디자인	통일성을 위해 최대한 다른 게시판들과 비슷한 디자인으로 설정해야 함	중	<input checked="" type="checkbox"/>
Review 게시판	기능	게시글 목록 기능	글 목록이 출력되며 페이징 기능을 통해 페이지를 이동할 수 있어야 함. 게시글 클릭시 해당 글의 상세페이지로 이동 할 수 있는 기능 추가	상	<input checked="" type="checkbox"/>
Review 게시판	기능	정렬기능	 열기 사용자가 드롭 박스로 글 목록의 정렬 기준(최신 순, 별점 많은 순, 별점 적은 순)을 설정하여 글 목록을 정렬하는 기능 추가, 기본 정렬 값은 최신 순으로 설정. 단, 최신순 정렬에서는 최종 수정 일이 아닌 최초 작성일을 기준으로 해야함.	중	<input checked="" type="checkbox"/>
Review 게시판	기능	검색 기능	사용자가 드롭박스로 검색 유형(작성자, 제목, 내용, 제목 OR 내용)을 선택한 뒤 텍스트를 입력해 해당 텍스트 관련 글을 찾을 수 있는 기능 추가요망	중	<input checked="" type="checkbox"/>
Review 게시판	기능	페이징 기능	페이징 기능을 추가해 사용자가 페이지를 이동 할 수 있는 기능 추가. 페이지에 이전, 다음 버튼을 추가하여 한 페이지씩 이동 가능하도록 하고 첫 페이지 이거나 마지막 페이지일 경우 이전 or 다음 버튼을 누르면 해당 항목의 첫 페이지로 이동하도록 설정. 페이징 기능이 정렬 기능이나 검색 기능을 사용하면 해당 글 목록에서만 페이징 기능이 동적으로 작동되도록 설정하여 정렬 결과나 검색 결과에 해당하는 글 목록에서만 페이징 기능이 작동하도록 설정해야 함	상	<input checked="" type="checkbox"/>
Review 게시판	기능	게시글 작성 기능	사용자가 로그인 후 게시글을 작성할 수 있고 게시글은 리뷰게시글이니까 별점과 이미지 추가 기능이 들어가는 것이 필요 할 듯 제목과 내용은 텍스트로 제한이 있어야 함, 별점은 다섯 개의 별점 아이콘을 클릭하면 해당 별점 아이콘에 해당하는 별점이 주어지게 설정, 이미지 파일을 첨부시 해당 버튼을 누르면 이미지 파일에 한해 첨부 할 수 있으며 첨부한 이미지는 바로 확인할 수 있어야 함	상	<input checked="" type="checkbox"/>
Review 게시판	기능	게시글 수정 기능	수정 기능은 로그인 한 사용자와 게시글 작성자와 일치 되어야 이용할 수 있어야 함, 수정 후 해당 버튼 누르기 전까지는 품 제출이 되면 안됨 이미지 수정 기능도 있으면 좋을 듯 이미지 수정, 삭제시 바로 확인 할 수 있었으면 좋겠음	상	<input checked="" type="checkbox"/>
Review 게시판	기능	게시글 삭제 기능	삭제 기능은 로그인 한 사용자와 게시글 작성자와 일치하거나 로그인한 사용자가 관리자일 경우에만 활성화되어야 함 절대!!!!!!	상	<input checked="" type="checkbox"/>

2_논리적 모델링 : ER Diagram / 물리적 모델링



3_ MVC 패턴

Model

```
model
|- dao
|  |- CollectDAO.java
|  |- NoticeDAO.java
|  |- QnaAnswerDAO.java
|  |- QnaDAO.java
|  |- ReviewDAO.java
|  |- UserDAO.java
|- vo
|  |- Collect.java
|  |- Notice.java
|  |- Paging.java
|  |- PagingQna.java
|  |- Pagingseven.java
|  |- Qna.java
|  |- QnaAnswer.java
|  |- Review.java
|  |- User.java
```

Service

```
service
|- CollectService.java
|- NoticeService.java
|- QnaAnswerService.java
|- QnaService.java
|- ReviewService.java
|- UserService.java
```

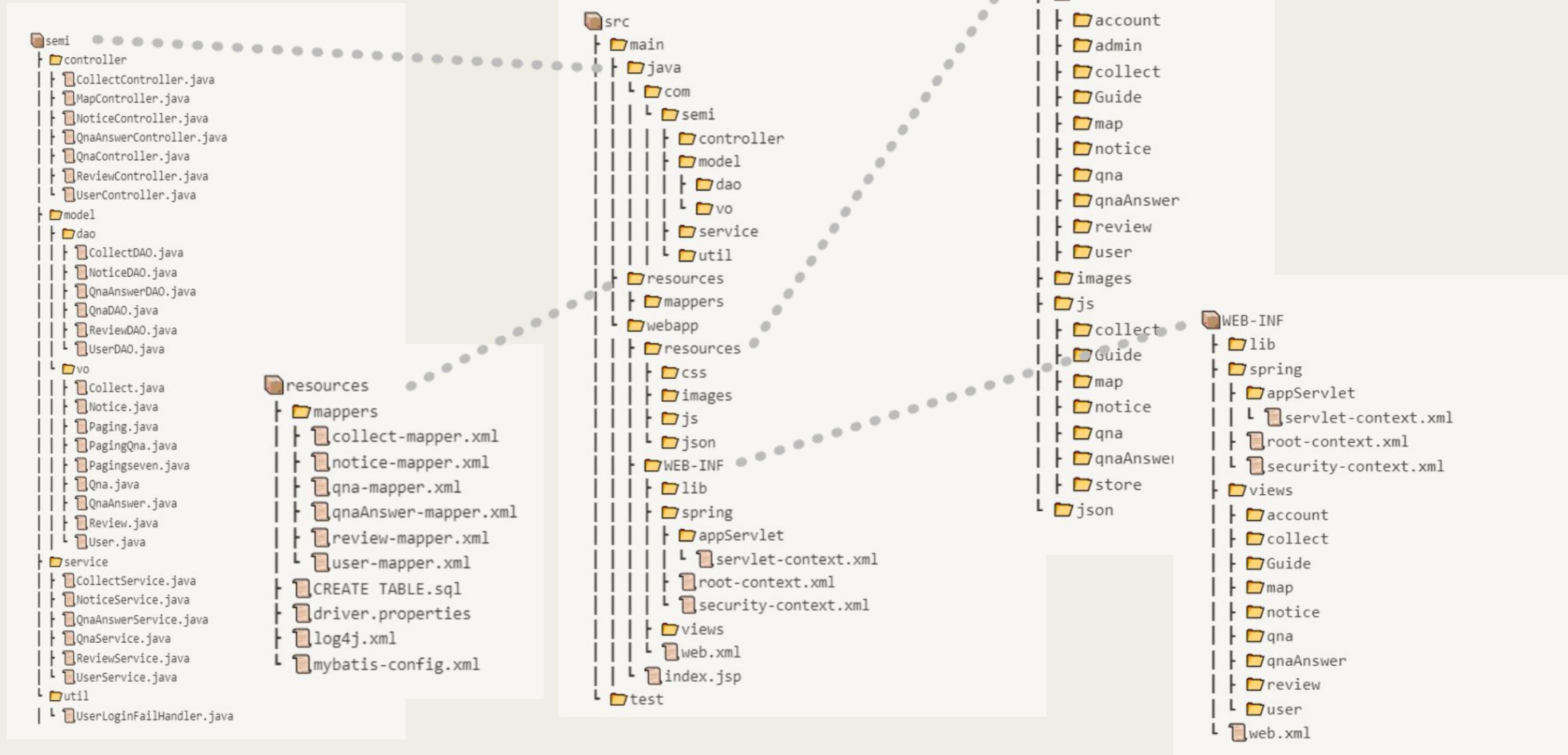
Controller

```
controller
|- CollectController.java
|- MapController.java
|- NoticeController.java
|- QnaAnswerController.java
|- QnaController.java
|- ReviewController.java
|- UserController.java
```

View

```
views
|- account
|  |- changePassword.jsp
|  |- findId.jsp
|- collect
|  ...
|- ...
|  |- registerCollect.jsp
|  |- updateCollect.jsp
|  |- viewCollect.jsp
|- Guide
|  |- Guide.jsp
|- map
|  |- mainMap.jsp
|- notice
|  |- noticeList.jsp
|  |- noticeView.jsp
|  |- noticeWrite.jsp
|- qna
|  ...
|  |- listQna.jsp
|  |- updateQna.jsp
|  |- viewQna.jsp
|- qnaAnswer
|  |- insertQnaAnswer.jsp
|  |- listQnaAnswer.jsp
|  |- updateQnaAnswer.jsp
|- review
|  |- list.jsp
|  |- view.jsp
|  |- write.jsp
|- user
|  |- myPage.jsp
|  |- register.jsp
|- ...
|- updateUser.jsp
```

4_ 프로젝트 구조





03_UI 및 기능 구현

Primavera_봄의 시작



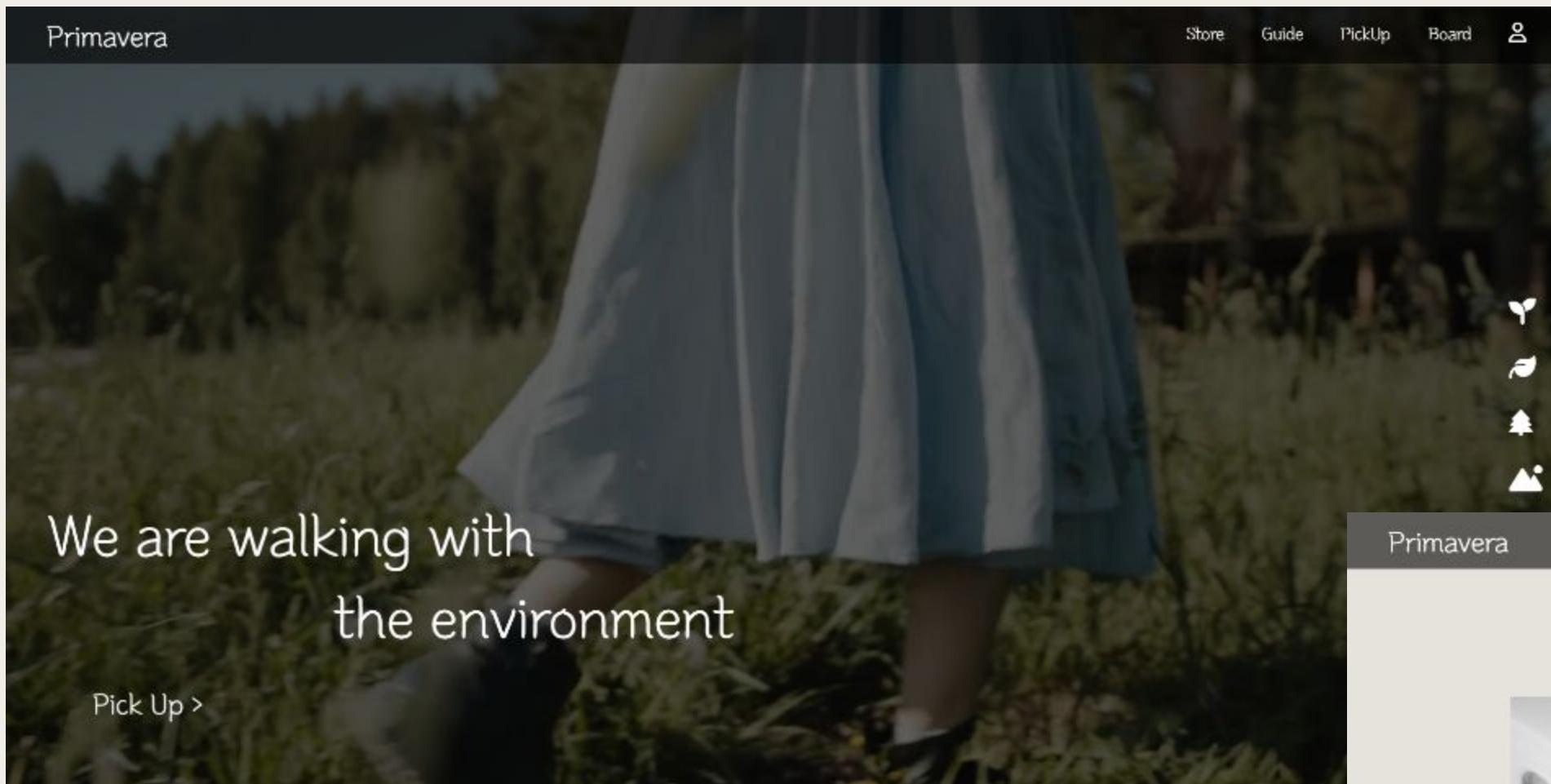
Main - Guide&Notice

배고픈 개발자 김현수



Welcome!
Main_Page

Main_Page - IMG



header + MainSection2 -

- header + MainSection1

Primavera

Store Guide PickUp Board

Company_introduce

'Primavera'는 '진정한 봄'이라는 뜻의 이탈리아어로 새로운 생명의 시작과 자연의 활기를 뜻합니다.

서회 'Primavera'는 심각한 지구 온난화와 기후 변화로 인한 피해를 줄이고자 탄소 배출량을 줄일 수 있는 폐의류 업사이클링을 통해 환경과 자연의 피해를 줄이고 이들의 원래 모습을 되찾기 위한 길을 걷고 있습니다.

Main_Page - IMG

header + MainSection3 -

Primavera

Store Guide PickUp Board ☰

Join our Prima

커뮤니티에 가입하여 더 많은 최신정보들과 유저들간의 소통을 원한다면

[Login](#)

[Join Membership](#)

Primavera
Let's change the environment

p : 052)XXXX-XXXX
T : 010-XXXX-XXXX
E : Primavera@com.kh.kr

새로운 생명의 시작과 자연의 활기
the beginning of a new life and the exuberance of nature

서피 회사는 이론에 끌려온 회사입니다(번호 CSC242153)

instagram facebook twitter YouTube

Since 2024

CEO : 이드워드 김
Personal Information Management Officer :
서비스 장 | 소리아 손 | 세일즈 김

본사 & 도어주소 : 서울 강남구 테헤란로 14길 6

이용약관 개인정보처리방침

App Store

Primavera

User_Guide

지속 가능한 옷은 이미 옷장에 있는 옷입니다.
프리마베라는 버려지는 의류에 새로운 가치를 부여합니다.

Sustainable clothes are clothes you already have in your closet. Primavera gives new value to discarded clothing.

> You want more details, don't you?

Map

• The service area is all over Seoul.

Details of user guides and maps

Primavera

Store Guide PickUp Board ☰

Map

• The service area is all over Seoul.

Details of user guides and maps

- header + Section4&Footer

MainPage _ JS

Primavera

We are walking with the environment

Pick Up >

Primavera

User_Guide

지속 가능한 옷은 이미 옷장에 있는 옷입니다.
프리파베라는 버려지는 의류에 새로운 가치를 부여합니다.

Sustainable clothes are clothes you already have in your closet. Primavera gives new value to discarded clothing.

> You want more details, don't you?

Map

The service area is all over Seoul.

Details of user guides and maps

```
/ 스크롤 시 애니메이션 효과 JS /
```

```
$(window).on("scroll", function () {
  var value = $(window).scrollTop();
  console.log("valueY", value);
  if (value > 789) {
    $(".h1").css("animation", "out1 1.8s ease-out forwards");
  } else {
    $(".h1").css("animation", "slide1 1.8s ease-out ");
  }
  if (value > 1578) {
    $(".h2").css("animation", "out2 1.8s ease-out forwards");
  } else {
    $(".h2").css("animation", "slide2 1.8s ease-out ");
  }

  if (value >= 1905) {
    $(".container1 h1").css("animation", "out4 1.5s ease-out forwards");
  } else {
    $(".container1 h1").css("animation", "slide4 1.5s ease-out ");
  }
  if (value >= 1905) {
    $(".container2 h1").css("animation", "out5 1.5s ease-out forwards");
  } else {
    $(".container2 h1").css("animation", "slide5 1.5s ease-out ");
  }
  // // =====
  if (value >= 1905) {
    $(".Floor p:first-child").css("animation", "out6 1.9s ease-out forwards");
  } else {
    $(".Floor p:first").css("animation", "slide6 1.9s ease-out ");
  }
  if (value >= 1905) {
    $(".Floor p:last-child").css("animation", "out7 1.9s ease-out forwards");
  } else {
    $(".Floor p:last-child").css("animation", "slide7 1.9s ease-out ");
  }
});
```



Guide_Page

GuidePage - IMG

Primavera

Store Guide PickUp Board ☰

Primavera

Store Guide PickUp Board ☰

01

수기 신청하기 웹 사이트를 통해 간편하게 의류 수기 신청을 할 수 있습니다.

사이트 내 [상단바 > PickUp] 또는 [마인 첫 메이지] > [Pick Up]을 통해 신청이 가능합니다.

수기 신청 주소가 회원 정보와 다른 경우 신청 품에서 변경 가능합니다.

'수거일'에 수거를 원하는 일정을 선택해 주세요.

다음으로 '수기 신청 무게' 항목에 수기 신청할 의류의 무게를 선택해 주세요.

* 프리미버라의 수거 서비스는 5kg 이상부터 20kg 이하인 경우에만 이용 가능합니다.

'수기 신청 하기' 버튼을 누르면 신청이 완료됩니다.



Guide

Primavera

Store Guide PickUp Board ☰

04

수기로 의류를 접수하거나 단단 제품들을 오프라인 스토어를 찾으시면 만날 수 있습니다.

> Offline store

Copyright © Primavera. All Rights Reserved.

Primavera

Store Guide PickUp Board ☰

02

수기 신청한 날짜이내에 수거물을 네이버시티 프리미버라 배달 노드에 차별 배송합니다.

당일 배송된 주소 지도를 기반으로 이동하는 노드 배달 시간은 충분히 확보 됩니다.

약속된 수기일 오후 8시 이전까지 수거물을 끝없이 놓아두시면 구타드립니다!

수거물 확인 시 신청 품에 입력하신 번호로 수거 확인 버튼을 활동합니다. (수거 완료)



Primavera

Store Guide PickUp Board ☰

03

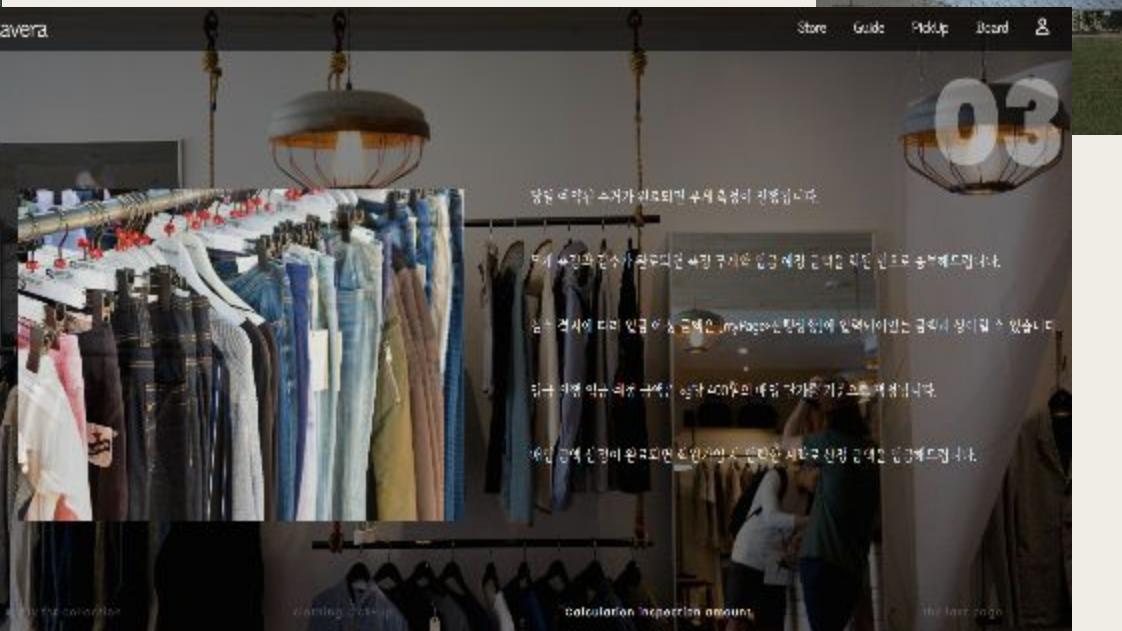
점포 수거일 수거가 원로되어 유기 속성이 저발집니다.

수기 배송과 접수가 원로되길 속장 주제와 같은 배송 금액을 적용 안으로 충분해드립니다.

수기 전자에 따라 일정 이용 금액은 MyPage 신청 품에 일련번호 있는 글자와 상이될 수 있습니다.

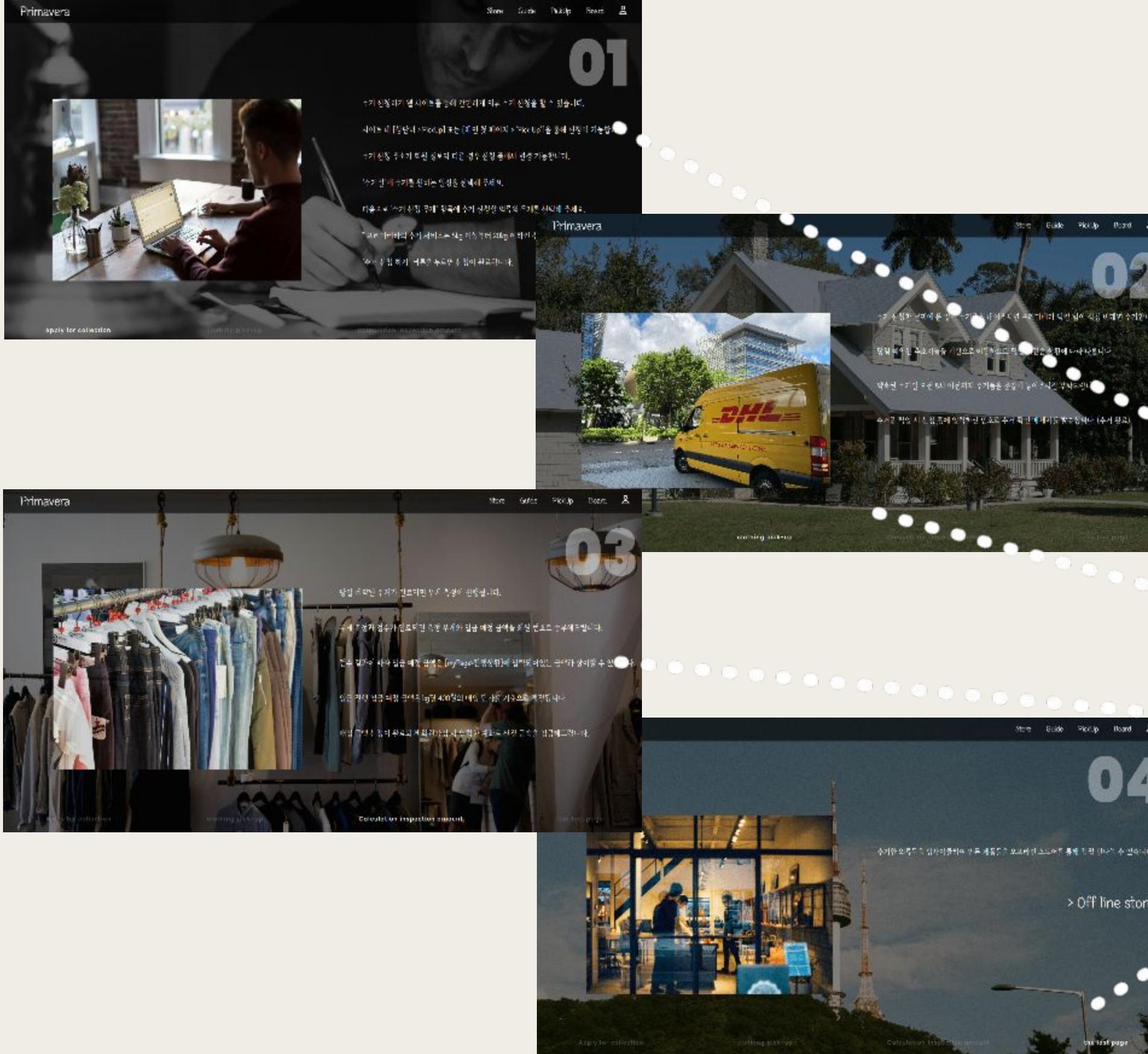
수기 전에 의류 리본 부착은 500~4000원의 배송 추가를 기준으로 예상됩니다.

제작 단계 신경이 요구되면 조언과 함께 제작 단계로 신청 금액은 일부 예상됩니다.



Primavera

Guide-JS



```
(function ($) {
  "use strict";

  //Page cursors

  document
    .getElementsByTagName("body")[0]
    .addEventListener("mousemove", function (n) {
      (t.style.left = n.clientX + "px"),
      (t.style.top = n.clientY + "px"),
      (e.style.left = n.clientX + "px"),
      (e.style.top = n.clientY + "px"),
      (i.style.left = n.clientX + "px"),
      (i.style.top = n.clientY + "px");
    });
  var t = document.getElementById("cursor"),
    e = document.getElementById("cursor2"),
    i = document.getElementById("cursor3");
  function n(t) {
    e.classList.add("hover"), i.classList.add("hover");
  }
  function s(t) {
    e.classList.remove("hover"), i.classList.remove("hover");
  }
  s();
  for (
    var r = document.querySelectorAll(".hover-target"), a = r.length - 1;
    a >= 0;
    a--
  ) {
    o(r[a]);
  }
  function o(t) {
    t.addEventListener("mouseover", n), t.addEventListener("mouseout", s);
  }
  $(document).ready(function () {
    /* 마우스 enter시 이벤트&삭제 show&remove 이벤트 */
    $(".case-study-name:nth-child(1)").on("mouseenter", function () {
      $(".case-study-name.active").removeClass("active");
      $(".case-study-images li.show").removeClass("show");
      $(".case-study-images li:nth-child(1)").addClass("show");
      $(".case-study-name:nth-child(1)").addClass("active");
    });
    $(".case-study-name:nth-child(2)").on("mouseenter", function () {
      $(".case-study-name.active").removeClass("active");
      $(".case-study-images li.show").removeClass("show");
      $(".case-study-images li:nth-child(2)").addClass("show");
      $(".case-study-name:nth-child(2)").addClass("active");
    });
    $(".case-study-name:nth-child(3)").on("mouseenter", function () {
      $(".case-study-name.active").removeClass("active");
      $(".case-study-images li.show").removeClass("show");
      $(".case-study-images li:nth-child(3)").addClass("show");
      $(".case-study-name:nth-child(3)").addClass("active");
    });
    $(".case-study-name:nth-child(4)").on("mouseenter", function () {
      $(".case-study-name.active").removeClass("active");
      $(".case-study-images li.show").removeClass("show");
      $(".case-study-images li:nth-child(4)").addClass("show");
      $(".case-study-name:nth-child(4)").addClass("active");
    });
    $(".case-study-name:nth-child(1)").trigger("mouseenter");
  })(jQuery);
```



Notice

Notice _ List UI & 기능

Notice

번호	제목	내용	작성일
1	asdasdasd	asdasdasd	24-03-21
2	사진 없어(수정)	수우저엉	24-03-21
3	사진 없어	사진 없어	24-03-21
4	사진 있어	사진 있어	24-03-21
5	asd	asdasd	24-03-21
6	asdasd	asdasd	24-03-21
7	내가 누군지 아니	ㅁㄴㅇㅁㄴㅇ	24-03-21
8	나야나	123123	24-03-20
9	누구인가	ㅁㄴㅇㅁㄴㅇ	24-03-20
10	첫번째	asd	24-03-20

Previous 1 2 Next

게시글 등록

Security ADMIN <〉 USER & a non-member
- JSP에 Security를 사용하여 조건으로 관리자와
그외의 사람들에 대한 게시물 등록 버튼 보이는 유무

NoticeList.JSP

```
<div class="container">
    <h1>Notice</h1>
    /* Security => 사용시기 회원이시 관리자일때 (ADMIN) */
    <c:if test="#{user != 'anonymousUser' && user.auth == 'ADMIN'}">
        <a href="/notice/write" class="btn btn-outline-warning"
            id="admit">게시글 등록</a>
    </c:if>
</div>
<table border="1" class="table">
    <thead>
        <tr>
    </thead>
    <tbody>
        <c:forEach items="${noticeList}" var="notice" varStatus="status">
            <tr>
                <td>${status.count + paging.pageSize * (paging.page-1)}</td>
                <td><a href="/notice/noticeView?noticeNum=${notice.noticeNum}">${notice.title}</a>
                </td>
                <td>${notice.content}</td>
                <td><fmt:formatDate value="${notice.writeDate}"
                    pattern="yy-MM-dd" /></td>
            </tr>
        </c:forEach>
    </tbody>
</table>
<nav>
    <ul class="pagination">
        <li class="page-item ${paging.prev ? '' : 'disabled'}">
            <c:choose>
                <c:when test="${paging.startPage == 1}">
                    <a class="page-link" href="/notice/list?page=${paging.startPage=1}">Previous</a>
                </c:when>
                <c:otherwise>
                    <a class="page-link" href="/notice/list?page=${paging.startPage-1}">Previous</a>
                </c:otherwise>
            </c:choose>
        </li>
        <c:forEach begin="${paging.startPage}" end="${paging.endPage}" var="page">
            <li class="page-item"><a class="page-link ${paging.page== page ? 'active' : ''}" href="/notice/list?page=${page}">${page}</a></li>
        </c:forEach>
        <li class="page-item ${paging.next ? '' : 'disabled'}">
            <c:choose>
                <c:when test="${paging.endPage < 10}">
                    <a class="page-link" href="/notice/list?page=${paging.endPage=paging.endPage}">Next</a>
                </c:when>
                <c:otherwise>
                    <a class="page-link" href="/notice/list?page=${paging.endPage+1}">Next</a>
                </c:otherwise>
            </c:choose>
        </li>
    </ul>
</nav>
```

Notice_List UI & 기능



1. Select 로직

번호	제목	내용	작성일
1	asdasdasd	asdasdasd	24-03-21
2	사진 없어(수정)	수우저엉	24-03-21
3	사진 없어	사진 없어	24-03-21
4	사진 있어	사진 있어	24-03-21
5	asd	asdasd	24-03-21
6	asdasd	asdasd	24-03-21
7	내 가 누군지 아니	ㅁㄴㅇㅁㄴㅇ	24-03-21
8	나야나	123123	24-03-20
9	누구인가	ㅁㄴㅇㅁㄴㅇ	24-03-20
10	첫번째	asd	24-03-20

Previous 1 2 Next

2. VO와 JSP에서 처리하여 Paging 기능 설정

```
@GetMapping("/list")
public String list(Model model, Paging paging) {
    model.addAttribute("noticeList", service.selectAll(paging));
    model.addAttribute("paging", new Paging(paging.getPage(), service.total()));
    return "/notice/noticeList";
};

private int page = 1;
private int offset = 0;
private int limit = 10;
private int pageSize = 10;
private int endPage;
private int startPage;
private int total;

private String searchType;
private String searchKeyword;

private int sorts;

private boolean prev;
private boolean next;

private String sort = "dateDesc";

public Paging(int page, int total) {
    this.page = page;
    this.total = total;

    int totalPages = (int) Math.ceil((double) total / this.limit);

    this.startPage = ((page - 1) / pageSize) * pageSize + 1;
    this.endPage = Math.min(startPage + pageSize - 1, totalPages);

    this.prev = this.startPage > 1;
    this.next = this.endPage < totalPages;
```

Notice_View UI & 기능 NoticeView.JSP = ! ADMIN

Notice

title : "디지몬 어드벤쳐"

content :

태일아 나 배고파~~~~~ㅠㅠ

첨부파일 : b6d378fa-841d-40f2-a42d-414da509a381_carbon (1).png

Security ADMIN
< User & a non
- member

목록

- INSERT로 추가된 유의사항을 불러오는 로직

```
@GetMapping("noticeView")
public void view(String noticeNum, Model model) {
    Notice vo = service.select(Integer.parseInt(noticeNum));
    model.addAttribute("vo", service.select(Integer.parseInt(noticeNum)));
}
```

```
<sec:authentication property="principal" var="user" />
<c:choose>
    <c:when test="${user == 'anonymousUser' || user.auth == 'MEMBER'}">
        <div class="container2">
            <h1>Notice</h1>
            <form action="updateNotice" method="post"
                enctype="multipart/form-data">
                <div class="form-group" id="usertitle">
                    <label class="form-control" name="title" value="${vo.title}"
                        id="titleinput">title : "${vo.title}"</label>
                </div>
                <div class="form-group" id="usercontext">
                    <label>content : </label>
                    <textarea class="form-control" cols="1000" rows="100"
                        name="content" style="resize: none" id="content_input"
                        readonly>${vo.content}</textarea>
                </div>
                <c:if test="${vo.url != null}">
                    <a href="/upload/notice/${vo.url}" id="fileupload" for="file" download>
                        첨부파일 : ${vo.url}</a>
                </c:if>
                <c:if test="${vo.url == null}">
                    <a href="#" id="fileupload" style="cursor:default;"></a>
                </c:if>
                <a href="/notice/list" id="list">목록</a>
            </form>
        </div>
    </c:when>
```

Notice_View UI & 기능

게시물 정보

title 디자운 어드벤처

태일아 나 배고파 ~~~~

content

파일 업로드 : b6d378fa-841d-40f2-a42d-414da509a381_carbor

파일 선택 선택된 파일 없음

수정 삭제 목록

중복 방지를 위한 업로드 >



```
public String noticeFileUpload(MultipartFile file) throws IllegalStateException, IOException {  
    UUID uuid = UUID.randomUUID();  
    String filename = uuid.toString() + "_" + file.getOriginalFilename();  
    File copyFile = new File(path + filename);  
    file.transferTo(copyFile);  
  
    return filename;  
}
```

```
<c:otherwise>  
    <c:if test="${user.auth == 'ADMIN' || user.auth ==! 'ADMIN' }">  
        <div class="container">  
            <h1>게시물 정보</h1>  
            <form action="updateNotice" method="post"  
                  enctype="multipart/form-data" onsubmit="return validate()">  
                <input type="hidden" name="noticeNum" value="${vo.noticeNum}">  
                <input type="hidden" name="url" value="${vo.url}">  
                <div class="form-group" id="title">  
                    <label>title</label> <input class="form-control" name="title" value="${vo.title}" id="title_input" maxlength="30" placeholder="제목을 입력하세요.(최대 30자)">  
                </div>  
                <div class="form-group" id="context">  
                    <label>content</label>  
                    <textarea class="form-control" row="10" name="content" style="resize: none;" id="content_input">${vo.content}</textarea>  
                </div>  
                <div class="form-group" id="file">  
                    <a href="/upload/notice/${vo.url}" for="file" download>파일 업로드 : ${vo.url}</a> <input class="form-control" type="file" id="file" name="file" accept="/*">  
                </div>  
                <div class="type">  
                    <button type="submit" class="btn btn-outline-warning">수정</button>  
                    <a class="btn btn-outline-danger" href="/notice/deleteNotice?noticeNum=${vo.noticeNum}">삭제</a> <a href="/notice/list" class="back">목록</a>  
                </div>  
            </form>  
        </div>  
    </c:if>  
</c:otherwise>
```

NoticeView.JSP == ADMIN

Security ADMIN <

Notice_View UI & 기능

UpDate 로직 >

게시물 정보

title

내일아나 배고파 ~~~~~

content

파일 업로드 : b6d378fa-841d-40f2-a42d-414da509a381_carbor

파일 선택 선택된 파일 없음

FileUpload 로직 >

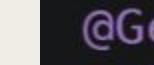


```
@PostMapping("/upload")
public String upload(MultipartFile file) throws IllegalStateException, IOException {
    noticeFileUpload(file);
    return "redirect:/";
}
```

Delete 로직 >



```
@PostMapping("updateNotice")
public String update(Notice n) throws IllegalStateException, IOException {
    if (!n.getFile().isEmpty()) {
        if (n.getUrl() != null) {
            File file = new File(path + n.getUrl());
            file.delete();
        }
        String url = noticeFileUpload(n.getFile());
        n.setUrl(url);
    }
    service.update(n);
    return "redirect:/notice/list";
}
```



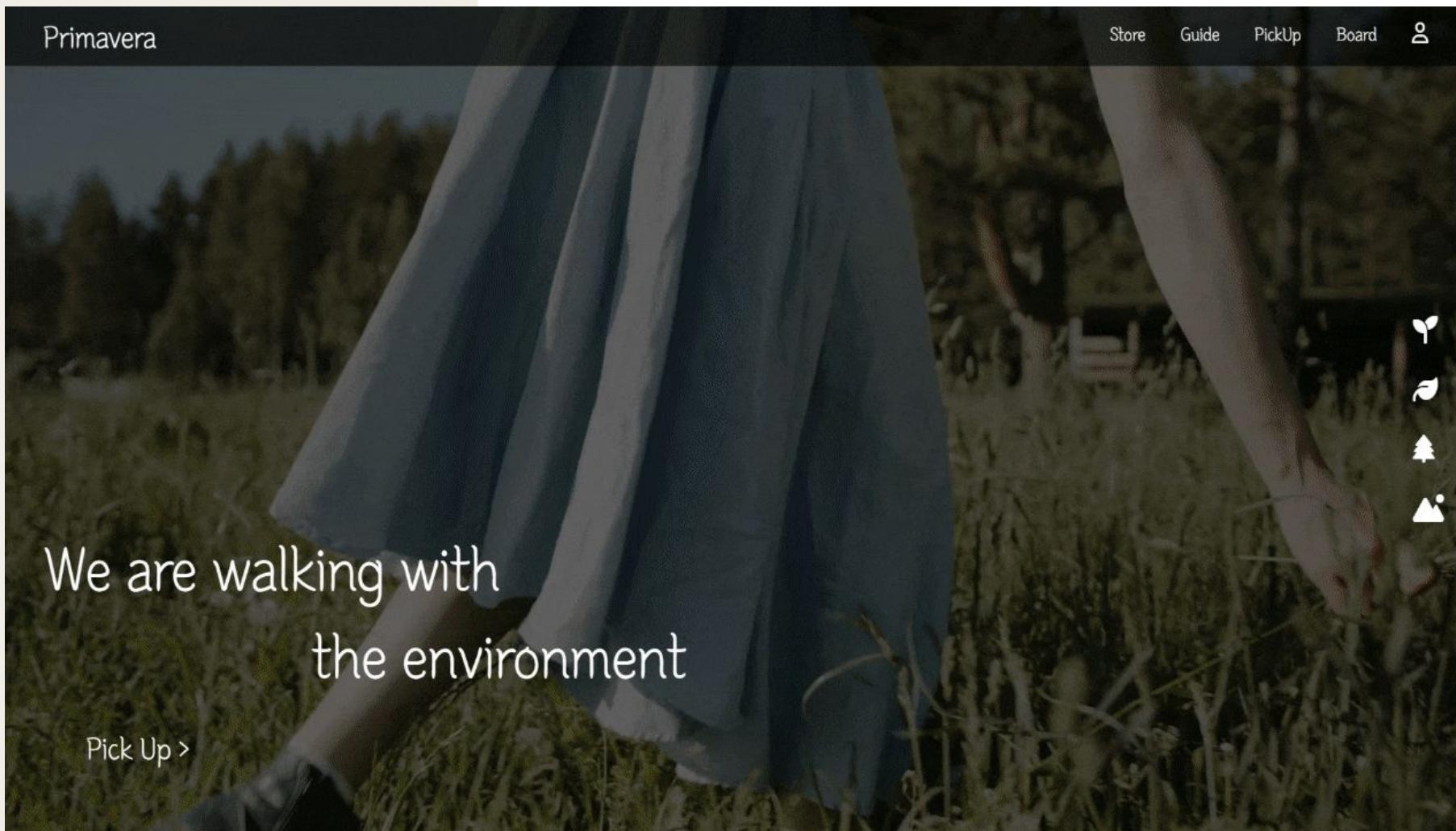
```
@GetMapping("deleteNotice")
public String delete(String noticeNum) {
    int parsingNo = Integer.parseInt(noticeNum);

    Notice n = service.select(parsingNo);
    if (n.getUrl() != null) {
        File file = new File(path + n.getUrl());
        file.delete();
    }
    service.delete(parsingNo);
    return "redirect:/notice/list";
}
```



Review / Store

REVIEW PAGE



```
● ○ ●

<table class="table">
    <thead>
        <tr>
            <th>PHOTO</th>
            <th>번호</th>
            <th>제목</th>
            <th>작성자</th>
            <th>별점</th>
            <th>작성일/수정일</th>
        </tr>
    </thead>
    <tbody>
        <c:forEach items="${list}" var="review" varStatus="status">
            <tr>
                <td class="image-cell"><c:if test="${not empty review.url}">
                    
                </c:if></td>
                <td>${paging.total - (paging.page - 1) * 10 - status.index}</td>
                <td><a class="review-link"
                    href="/review/view?no=${review.no}">${review.title}</a></td>
                <td>${review.id}</td>
                <td>
                    <div class="star-rating">
                        <c:forEach begin="1" end="5" var="i">
                            <i
                                class="${i <= review.rating ? 'fas fa-star' : 'far fa-
                                star'}"></i>
                        </c:forEach>
                    </div>
                </td>
                <td><fmt:formatDate value="${review.date}"
                    pattern="yyyy-MM-dd" /></td>
            </tr>
        <c:forEach>
    </tbody>
</table>
return go(f, seed, [])
}
```

REVIEW PAGE

```
<nav>
    <ul class="pagination">
        <li class="page-item ${paging.page > 1 ? '' : 'disabled'}"><a
            class="page-link"
            href=<c:url value='/review/list'>
<c:param name='page' value='${paging.page - 1}' />
<c:param name='sort' value='${paging.sort}' />
<c:param name='searchType' value='${searchType}' />
<c:param name='searchKeyword' value='${searchKeyword}' />
</c:url>">Previous</a>
        </li>

        <c:forEach var="pageNum" begin="${paging.startPage}"
            end="${paging.endPage}">
            <li class="page-item ${pageNum == paging.page ? 'active' : ''}">
                <a class="page-link"
                    href=<c:url value='/review/list'>
<c:param name='page' value='${pageNum}' />
<c:param name='sort' value='${paging.sort}' />
<c:param name='searchType' value='${searchType}' />
<c:param name='searchKeyword' value='${searchKeyword}' />
</c:url>">${pageNum}</a>
            </li>
        </c:forEach>

        <li class="page-item ${paging.page < totalPages ? '' : 'disabled'}">
            <a class="page-link"
                href=<c:url value='/review/list'>
<c:param name='page' value='${paging.page + 1}' />
<c:param name='sort' value='${paging.sort}' />
<c:param name='searchType' value='${searchType}' />
<c:param name='searchKeyword' value='${searchKeyword}' />
</c:url>">Next</a>
        </li>
    </ul>
</nav>
```

```
@GetMapping("/review/list")
public String list(Model model,
    @RequestParam(value = "page", defaultValue = "1") int page,
    @RequestParam(value = "sort", defaultValue = "dateDesc") String sort,
    @RequestParam(value = "searchType", required = false) String searchType,
    @RequestParam(value = "searchKeyword", required = false) String searchKeyword) throws
UnsupportedEncodingException {

    int total = service.total(searchType, searchKeyword);
    Paging paging = new Paging(page, total);
    paging.setSort(sort);

    int totalPages = (int) Math.ceil((double) total / paging.getLimit());

    if (page < 1 || page > totalPages) {
        page = 1;
        paging = new Paging(page, total);
    }

    List<Review> reviewList = service.selectPage(paging, searchType, searchKeyword);
    model.addAttribute("list", reviewList);
    model.addAttribute("paging", paging);
    model.addAttribute("totalPages", totalPages);

    Authentication authentication = SecurityContextHolder.getContext().getAuthentication();
    boolean isLoggedIn = authentication != null && authentication.isAuthenticated()
        && !(authentication instanceof AnonymousAuthenticationToken);
    model.addAttribute("isLoggedIn", isLoggedIn);
    model.addAttribute("searchType", searchType);
    model.addAttribute("searchKeyword", searchKeyword);

    return "review/list";
}
```

REVIEW PAGE

The screenshot shows a review page from a mobile application. At the top, there is a navigation bar with links for 'Store', 'Guide', 'PickUp', 'Board', and a user icon. Below the navigation bar, there is a large image of a green lawn. In the center, there is a search bar with a dropdown menu set to '작성자' and a text input field containing '검색어를 입력하세요'. A green button labeled '검색' is next to the input field. Below the search bar, there is a section titled 'REVIEW' with a green button. The main content area displays a table with the following columns: PHOTO, 번호, 제목, 작성자, 별점, and 작성일/수정일. The first row of the table is highlighted, showing a photo of a bird perched on a branch, the number 52, the title '세번째 리뷰쓰기', the author 'eodbs21', a 5-star rating, and the date '2024-03-26'.

```
<select id="count" parameterType="map" resultType="int">
    SELECT COUNT(*) FROM review
    <where>
        <choose>
            <when test="searchType == 'author'">
                id LIKE CONCAT( '%', #{searchKeyword}, '%' )
            </when>
            <when test="searchType == 'title'">
                title LIKE CONCAT( '%', #{searchKeyword}, '%' )
            </when>
            <when test="searchType == 'content'">
                content LIKE CONCAT( '%', #{searchKeyword}, '%' )
            </when>
            <when test="searchType == 'titleContent'">
                (title LIKE CONCAT( '%', #{searchKeyword}, '%' ) OR
                content LIKE
                CONCAT( '%', #{searchKeyword}, '%' ))
            </when>
        </choose>
    </where>
</select>
```

REVIEW PAGE

Primavera

Store Guide PickUp Board ☰

REVIEW

최신순

작성자 검색어를 입력하세요 검색

PHOTO	번호	제목	작성자	별점	작성일/수정일
	52	세번째 리뷰쓰기	eodbs21	★★★★★	2024-03-26
	51	두번째 리뷰 쓰기	eodbs21	★★★☆☆	2024-03-26
	50	리뷰 처음 쓰기	eodbs31	★★★★☆	2024-03-26

```
<select id="selectPage" parameterType="map"
    resultMap="reviewMap">
    SELECT * FROM review
    <where>
        <if test="searchType != null">
            <choose>
                <when test="searchType == 'author'"> AND id LIKE CONCAT( '%', #{searchKeyword}, '%' )
                </when>
                <when test="searchType == 'title'"> AND title LIKE CONCAT( '%', #{searchKeyword}, '%' )
                </when>
                <when test="searchType == 'content'"> AND content LIKE CONCAT( '%', #{searchKeyword}, '%' ) </when>
                <when test="searchType == 'titleContent'"> AND (title LIKE CONCAT( '%', #{searchKeyword}, '%' )
                    OR content LIKE CONCAT( '%', #{searchKeyword}, '%' )) </when>
            </choose>
        </if>
    </where>
    ORDER BY
    <choose>
        <when test="paging.sort == 'ratingDesc'"> rating DESC, no DESC </when>
        <when test="paging.sort == 'ratingAsc'"> rating ASC, no DESC </when>
        <otherwise> no DESC </otherwise>
    </choose>
    LIMIT #{paging.offset}, #{paging.limit}
</select>
```

REVIEW PAGE

PHOTO	번호	제목	작성자	별점	작성일/수정일
	53	네번째 리뷰	eodbs12	★★★☆☆	2024-03-26
	52	세번째 리뷰쓰기	eodbs21	★★★★★	2024-03-26

localhost:8080/review/write

```
function previewImage(event) {
    var reader = new FileReader();
    reader.onload = function () {
        var output = document.getElementById('image_preview');
        output.style.backgroundImage = 'url(' + reader.result + ')';
        output.style.display = 'block';
    };
    reader.readAsDataURL(event.target.files[0]);
}

function validateForm() {
    var title = document.getElementById("title_input").value;
    var content = document.getElementById("content_input").value;
    if (!title && !content) {
        alert("제목과 내용이 입력되지 않았습니다.");
        return false;
    } else if (!title) {
        alert("제목이 입력되지 않았습니다.");
        return false;
    } else if (!content) {
        alert("내용이 입력되지 않았습니다.");
        return false;
    }
    return true;
}

document.addEventListener("DOMContentLoaded", function () {
    const stars = document.querySelectorAll("#rating i");
    updateStars(1);

    stars.forEach((star, index) => {
        star.addEventListener("click", () => {
            const rating = index + 1;
            document.getElementById("rating-input").value = rating;
            updateStars(rating);
        });
    });

    function updateStars(rating) {
        stars.forEach((star, index) => {
            if (index < rating) {
                star.classList.remove("far");
                star.classList.add("fas");
            } else {
                star.classList.remove("fas");
                star.classList.add("far");
            }
        });
    }
});
```

REVIEW PAGE

Primavera

REVIEW

최신순 ▾

작성자 ▾ 검색어를 입력하세요 ▾ 검색

리뷰 작성

PHOTO	번호	제목	작성자	별점	작성일/수정일
	54	다섯번째 리뷰	eodbs12	★★★★★	2024-03-26
	53	네번째 리뷰	eodbs12	★★★★★	2024-03-26
	52	세번째 리뷰쓰기	eodbs21	★★★★★	2024-03-26
	51	두번째 리뷰 쓰기	eodbs21	★★★★★	2024-03-26

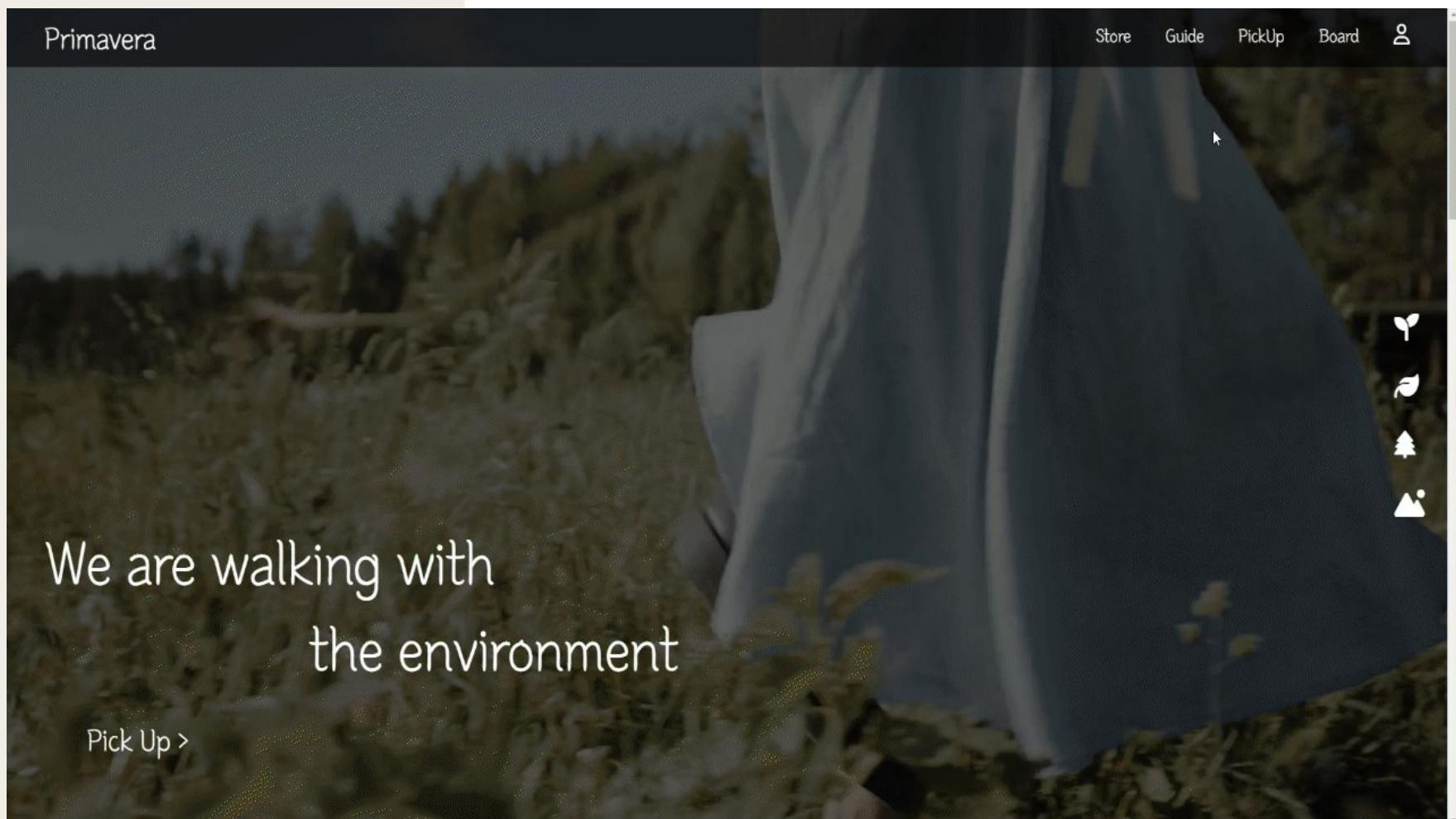
```
@GetMapping("/review/view")
public String view(String no, Model model) {
    Object principal = SecurityContextHolder.getContext().getAuthentication().getPrincipal();
    String currentLoggedInUserId = "";
    if (principal instanceof UserDetails) {
        currentLoggedInUserId = ((UserDetails) principal).getUsername();
    } else {
        currentLoggedInUserId = principal.toString();
    }
    model.addAttribute("currentUserId", currentLoggedInUserId);
    model.addAttribute("vo", service.select(Integer.parseInt(no)));
    return "review/view";
}

@PostMapping("/updatereview")
public String update(Review review, @RequestParam("file") MultipartFile file,
                    RedirectAttributes redirectAttributes) throws IllegalStateException, IOException {
    if (!file.isEmpty()) {
        String newUrl = fileUploadreview(file);
        review.setUrl(newUrl);
    } else {
        File oldFile = new File(path + review.getUrl());
        if (oldFile.exists()) {
            oldFile.delete();
        }
        review.setUrl(null);
    }
    service.updatereview(review);
    return "redirect:/review/view?no=" + review.getNo();
}

var inputFile = document.getElementById('fileInput');
if (inputFile) {
    inputFile.value = "";
}

```

STORE PAGE



```
markers.forEach(function(markerInfo, index) {
  var marker = new kakao.maps.Marker({
    map : map,
    position : markerInfo.position
  });
  var overlay = new kakao.maps.CustomOverlay({
    content : markerInfo.content,
    map : map,
    position : marker.getPosition()
  });
  overlay.setMap(null);
  daum.maps.event.addListener(marker, 'mouseover', function() {
    overlay.setMap(map);
    setTimeout(function() {
      overlay.setMap(null);
    }, 5000);
  });
}
);
function closeOverlay(overlay) {
  overlay.setMap(null);
}
var zoomControl = new kakao.maps.ZoomControl();
map.addControl(zoomControl, kakao.maps.ControlPosition.RIGHT);
</script>
<script>
document.addEventListener("DOMContentLoaded", function() {
  var observer = new IntersectionObserver(function(entries) {
    entries.forEach(entry => {
      if(entry.isIntersecting) {
        entry.target.classList.add('animate');
      }
    });
  });
  document.querySelectorAll('.title').forEach(item => {
    observer.observe(item);
  });
});
</script>
```



Store / Pick Up / Q&A

1_ Store

상품 상세 설명



Quilting Card Wallet

가죽 소재를 활용하여 업사이클링한
퀼팅 카드 지갑

* 아코디언 4단 분리 지갑

* 카드, 지폐, 동전, 명함 등 분리 수납 가능

* 모던 블랙 컬러



Accordion Card Holder



Pink Edge Half-Wallet

```


<h2>Frame Shoulder Bag</h2>
  <div class="size">
    <h3>인조 가죽 소재를 활용하여 업사이클링한 가죽 숄더 백</h3>
  </div>
  <div class="color">
    <p>
      * 내부 섹션 2개와 내부 포켓 3개 보유 <br> <br>
      * 탈부착 가능한 체인 스트랩<br><br>
      * 접이식 플랩, 자석 스냅 버튼
    </p>
  </div>

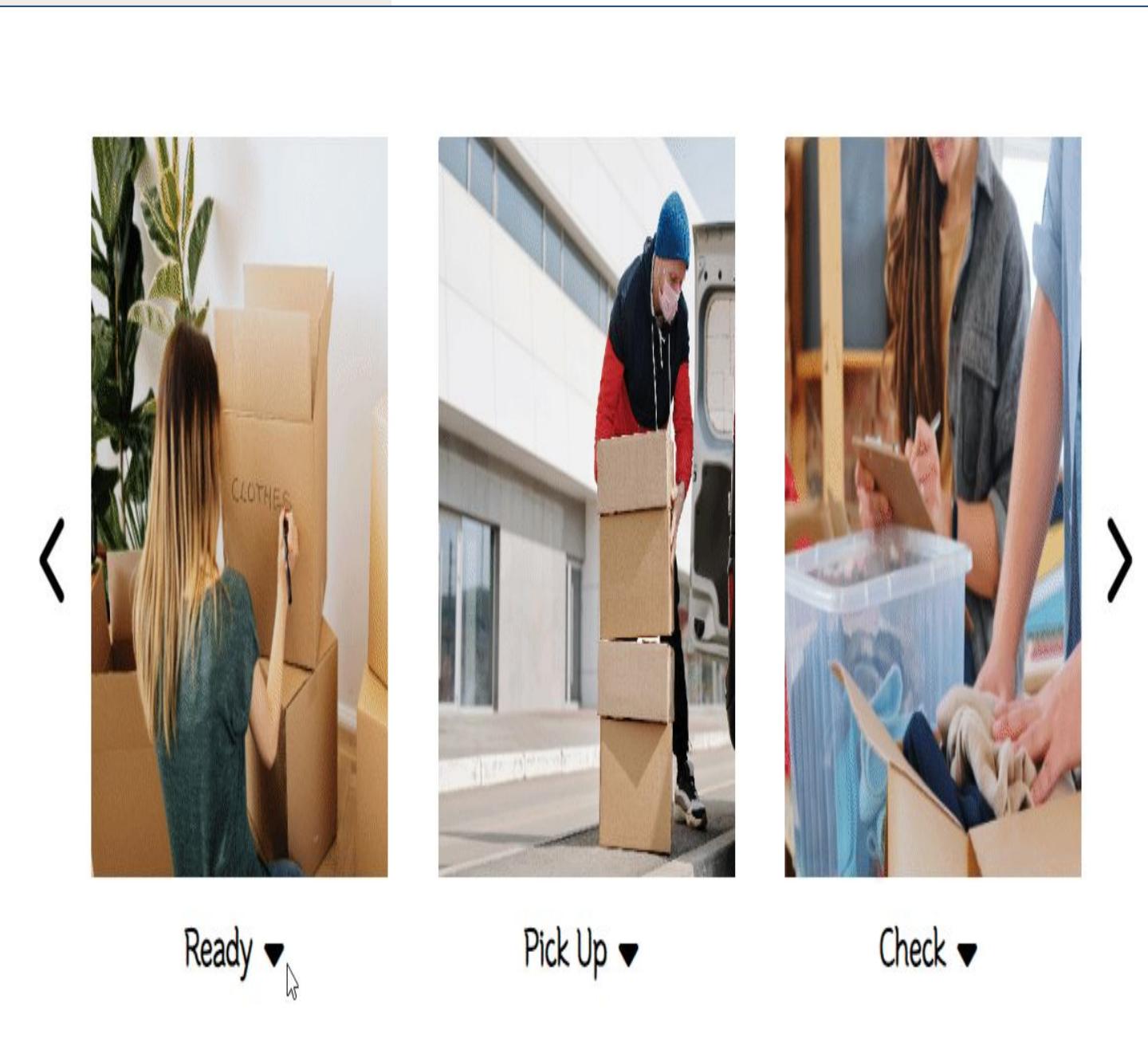

```

* 내부 섹션 2개와 내부 포켓 3개 보유
 * 탈부착 가능한 체인 스트랩
 * 접이식 플랩, 자석 스냅 버튼

</div>

2_ Pick Up

창 숨기기



```

● ● ●
<div class="image-content">
  <div class="card" style="width: 18rem;">
    
    <div class="card-body" id="card-body">
      <div id="ready_icon">
        <p class="card-text">Pick Up</p> &ampnbsp&ampnbsp
        <i class="fa-solid fa-caret-down baa"></i>
      </div>
      <div id="hiddenContent">
        <span>[STEP 2]<br>신청 날짜에 맞춰 픽업팀이 비대면 수거</span>
      </div>
    </div>
  </div>
</div>

```

● ● ●

CSS

```

.show {
  display: block;
}

.hide {
  display: none;
}

```

```

$(document).ready(function(){
  $('.baa').each(function() {
    $(this).click(function() {
      let layer = $(this).parent().next();
      layer.toggle(
        function() {layer.addClass('hide')},
        function() {layer.addClass('show')}
      );
    });
  });
});

```

js

2_ Pick Up

수거 신청 날짜 지정 및 무게 제한

Primavera

*아이디 *변경 불가 항목입니다.
user001

*이름 유저

*전화번호 01011112222

*주소 12345 우편번호 찾기
서울특별시 kh 4층 Q 강의실

수거일

*수거 신청 무게 5~20kg 신청 가능

공동 현관 비밀번호
공동 현관 비밀 번호가 필요한 경우, 입력해주세요

요청사항
요청사항을 있으신 경우, 입력해주세요

* 수거 신청을 위해 [이용약관](#) 및 [개인정보 수집](#)에 동의합니다.
 네, 동의합니다.

수거 신청 하기

jsp

```

<div class="collectDate">
    <p>*수거일</p>
    <input type="text" name="collectionDate" id="collectionDate" />
</div>

<div class="kg">
    <p>*수거 신청 무게</p>
    <input type="number" name="kg" id="kg" placeholder="5~20kg 신청 가능" min="5" max="20" />
</div>

```

js

```

$(function(){
    $("#collectionDate").datepicker({
        dateFormat: 'yy-mm-dd',
        minDate: 0 + "2D",
        maxDate: new Date('2025-12-31'),
        changeMonth:true,
        changeYear:true,
        nextText : '다음 달',
        prevText : '이전 달'
    });
});

```

2_ Pick Up 이용약관&정보수집 동의 Modal

Primavera

*아이디 변경 불가 항목입니다.

user001

*이름

유저

*전화번호

01011112222

*주소

12345

우편번호 찾기

서울특별시

kh 4층 Q 강의실

* 수거 신청을 위해 [이용약관](#) 및 [개인정보 수집](#)에 동의합니다.
 네, 동의합니다.

수거 신청 하기

jsp

```

<div id="agree_check">
  <p id="agree_text">
    * 수거 신청을 위해 <a href="#pop_info_1" class="btn_open"> 이용약관 </a>
    및 <a href="#pop_info_2" class="btn_open"> 개인정보 수집 </a>에 동의합니다.
  </p>
  <input type="checkbox" name="agree" id="agree" value="yes" />네, 동의합니다.
</div>

<div id="pop_info_1" class="pop_wrap">
  <div class="pop_inner">
    <jsp:include page="/WEB-INF/views/collect/agree1.jsp"></jsp:include>
    <hr>
    <button type="button" class="btn_close">닫기</button>
  </div>
</div>

<div id="pop_info_2" class="pop_wrap">
  <div class="pop_inner">
    <jsp:include page="/WEB-INF/views/collect/agree2.jsp"></jsp:include>
    <hr>
    <button type="button" class="btn_close">닫기</button>
  </div>
</div>

```

2_ Pick Up 이용약관&정보수집 동의 Modal

The screenshot shows a modal window overlaid on a dark background. The modal has a white header with the text "Primavera". Inside, there are several input fields and labels:

- *아이디:** user001 (with a note: "변경 불가 항목입니다.")
- *수거일:** 2024-03-30
- *이름:** 유저
- *수거 신청 무게:** 6
- *전화번호:** 01011112222
- 공동 현관 비밀번호:** (Input field placeholder: "공동 현관 비밀 번호가 필요한 경우, 입력해주세요")
- *주소:** 12345 (with a link: "우편번호 찾기")
- 요청사항:** (Input field placeholder: "요청사항을 있으신 경우, 입력해주세요")
- 서울특별시:** kh 4층 Q 강의실
- 동의합니다:** A checkbox labeled "네, 동의합니다." with a note below: "* 수거 신청을 위해 이용약관 및 개인정보 수집에 동의합니다."
- 수거 신청 하기:** A black button at the bottom of the modal.

js

```

var target = document.querySelectorAll('.btn_open');
var btnPopClose = document.querySelectorAll('.pop_wrap .btn_close');
var targetID;

for (var i = 0; i < target.length; i++) {
    target[i].addEventListener('click', function() {
        targetID = this.getAttribute('href');
        document.querySelector(targetID).style.visibility = 'visible';
    });
}

for (var j = 0; j < target.length; j++) {
    btnPopClose[j].addEventListener('click', function() {
        this.parentNode.parentNode.style.visibility = 'hidden';
    });
}

```

agree1.jsp

modal_css

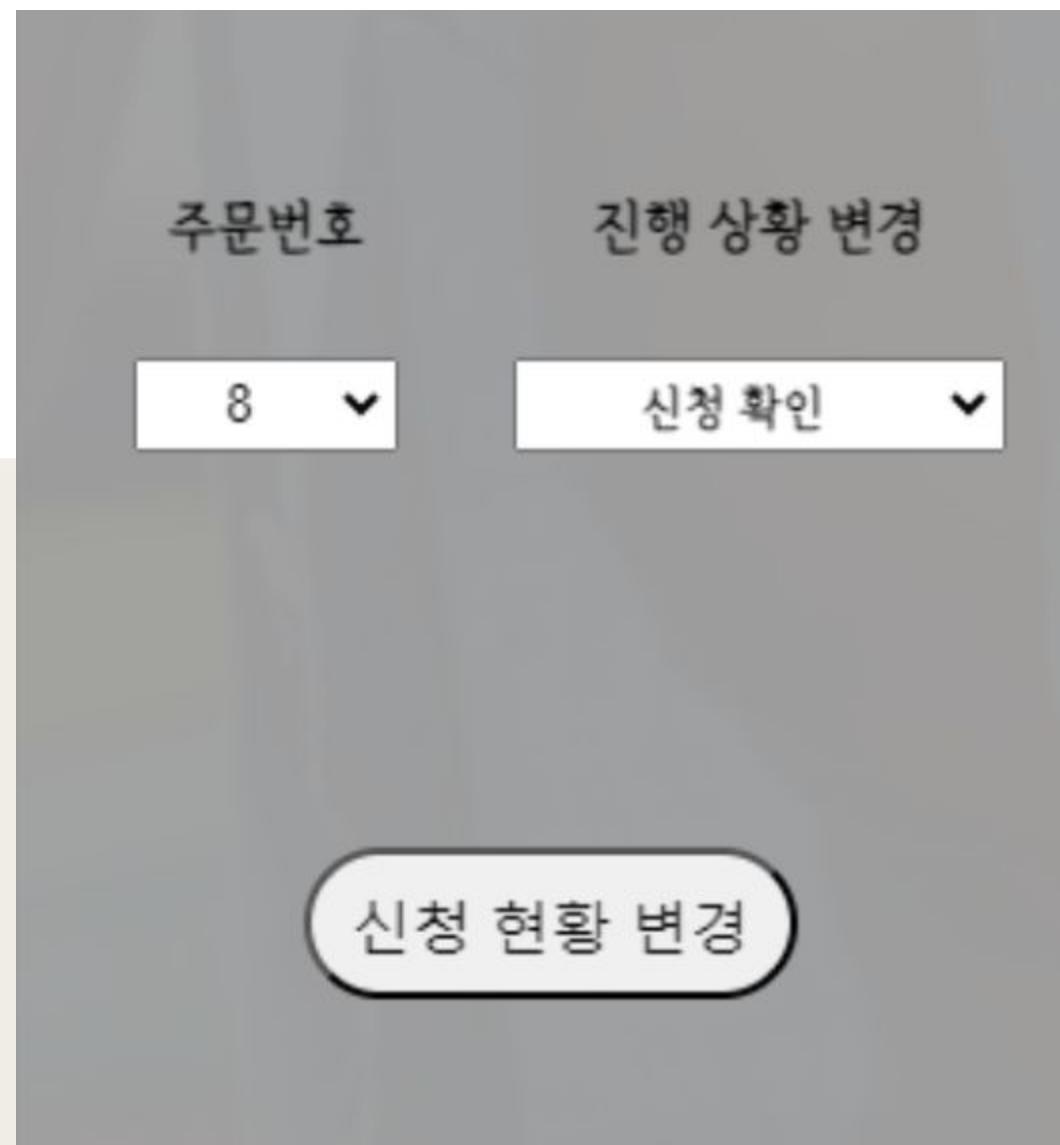
```

.modal-body{
  max-height: calc(99vh - 500px);
  overflow-y: auto;
}

```

2_ Pick Up

진행 상황 변경



```


<table border=1 class="progress">
        <tr>
            <th class="proghth">주문번호</th>
            <th class="proghth">진행 상황 변경</th>
        </tr>
        <tr>
            <td>
                <select name="orderNum" id="orderNum">
                    <c:forEach items="${progress}" var="item">
                        <option value="${item.orderNum}">${item.orderNum}</option>
                    </c:forEach>
                </select>
            </td>
            <td>
                <select name="prog" id="prog">
                    <option value="신청 원료">신청 원료</option>
                    <option value="신청 확인">신청 확인</option>
                    <option value="수거 완료">수거 완료</option>
                    <option value="수거 물품 확인">수거 물품 확인</option>
                    <option value="입금 진행중">입금 진행중</option>
                    <option value="입금 완료">입금 완료</option>
                </select>
            </td>
        </tr>
    </table>
    <div>
        <input type="button" id="btn" value="신청 현황 변경">
    </div>
</div>


```

jsp


```

$("#btn").click(()=>{
    $.ajax({
        type:"post",
        url:"/update_ok",
        data: $("#update").serialize(),
        success: function(result){
            $("#prog").html(result.prog);
            window.location.reload();
        },
        error: function(){
            console.log("에러 발생!");
        }
    })
})

```

js


```

@RequestBody
@PostMapping( "update_ok" )
public String updateProgress(Collect collect) {
    service.updateProgress(collect);
    return "collect";
}

```

controller

2_ Pick Up

수거 신청 변경&삭제 권한

수거 신청 현황 관리

주문번호	아이디	이름	전화번호	진행상황	수거량(kg)	금액(원)
1	user01	손민정	01012345678	신청 확인	6	2400
4	admin	매니저	01012345670	신청완료	6	2400
6	user01	손민정	01012345678	신청완료	6	2400
7	user01	손민정	01012345678	신청완료	9	3600
8	user01	손민정	01012345678	신청 확인	8	3200

Previous 1 Next

주문번호 진행 상황 변경

8 신청 확인

[신청 현황 변경](#)

```

<c:choose>
    <c:when test="${collect.prog eq '신청외류'}">
        <div id="btn">
            <div class="button-list">
                <button id="updateBtn"
                    onclick="location.href='/updatePageorderNum=${collect.orderNum}';">
                    신청 시험 수정
                </button>
                <button class="reverse dark"
                    onclick="location.href='/deleteCollect?orderNum=${collect.orderNum}';">
                    신청 취소
                </button>
            </div>
        </div>
    </c:when>
</c:choose>

```

진행 상황

주문번호	진행상황	수거일	수거 신청 무게	금액
8	신청 확인	2024-03-31	8	3200

아이디, 이름, 전화번호 아이디 전화번호

주소 주소

수거일, 수거 신청 무게 수거일, 수거 신청 무게

공동 현관 비밀번호, 요청사항 공동 현관 비밀번호, 요청사항

[신청사항수정](#) [신청취소](#)

<진행상황 변경 전>
<진행상황 변경 후>

3_ Q&A

Q&A

번호	제목	아이디	작성일	답변상태	Q&A등록
13	비밀글10	user01	24-03-20 15:39	N	
12	비밀글9	user01	24-03-26 12:46	Y	
11	비밀글8	user01	24-03-20 15:39	N	
10	비밀글7	user01	24-03-20 15:39	N	
9	비밀글6	user01	24-03-20 15:39	N	
8	비밀글5	user01	24-03-20 15:38	N	
7	비밀글4	user01	24-03-20 15:38	N	
6	비밀글3	user01	24-03-20 15:37	N	
5	비밀글2	user01	24-03-20 15:37	N	
4	비밀글1	user01	24-03-20 15:36	N	

제목
아이디
전체 | Previous 1 2 Next | 조회

controller

```

@GetMapping("listQna")
public String showFilm(Model model, @RequestParam(value = "page", defaultValue = "1") int page, String
select, String keyword){
    int total = service.total();
    Paging paging = new Paging(page, total);
    paging.setKeyword(keyword);
    paging.setSelect(select);

    List<Qna> list = service.showAllQna(paging);

    model.addAttribute("list", list);
    model.addAttribute("paging", paging);

    return "qna/listQna";
}

```

jsp

```

<div id="findQna">
    <form action="listQna" method="get">
        <select name="select">
            <option value="allFind">전체</option>
            <option value="titleFind">제목</option>
            <option value="idFind">아이디</option>
        </select>
        <input type="text" name="keyword">
        <input type="submit" value="조회" id="searchOk">
    </form>
</div>

```

mapper

```

<select id="showAllQna" parameterType="Paging" resultMap="qnaMap">
    SELECT qna_num, title, id, write_date, status, secret
    FROM qna
    <where>
        <choose>
            <when test="select == 'allFind'">
                title LIKE CONCAT('%',#{keyword},'%')
                OR id LIKE CONCAT('%',#{keyword},'%')
            </when>
            <when test="select == 'titleFind'">
                title LIKE CONCAT('%',#{keyword},'%')
            </when>
            <when test="select == 'idFind'">
                id LIKE CONCAT('%',#{keyword},'%')
            </when>
        </choose>
    </where>
    ORDER BY qna_num DESC
    LIMIT #{offset}, #{limit}
</select>

```

3_ Q&A

Q&A				
번호	제목	아이디	작성일	답변상태
13	비밀글10	user01	24-03-20 15:39	N
12	비밀글9	user01	24-03-26 12:46	Y
11	비밀글8	user01	24-03-20 15:39	N
10	비밀글7	user01	24-03-20 15:39	N
9	비밀글6	user01	24-03-20 15:39	N
8	비밀글5	user01	24-03-20 15:38	N
7	비밀글4	user01	24-03-20 15:38	N
6	비밀글3	user01	24-03-20 15:37	N
5	비밀글2	user01	24-03-20 15:37	N
4	비밀글1	user01	24-03-20 15:36	N

Previous 1 2 Next

[전체] [비밀글] [조회]

Q&A				
번호	제목	아이디	작성일	답변상태
3	비밀글	user01	24-03-20 15:36	N

Previous 1 2 Next

```

<c:forEach begin="${paging.startPage}" end="${paging.endPage}" var="page">
    <li class="page-item">
        <c:choose>
            <c:when test="${(paging.select != null) && (paging.keyword != null)}">
                <a class="page-link ${paging.page== page ? 'active' : ''}" href="/listQna?select=${paging.select}&keyword=${paging.keyword}&page=${page}" id="page_num">${page}</a>
            </c:when>
            <c:otherwise>
                <a class="page-link ${paging.page== page ? 'active' : ''}" href="/listQna?page=${page}" id="page_num">${page}</a>
            </c:otherwise>
        </c:choose>
    </li>
</c:forEach>

```

jsp

3_ Q&A

비밀글

```
<div class="insertQna" id="secret">
    <label id="secretCheck">비밀글</label>
    <input type="checkbox" name="secret"
        id="secret_input" value="Y">
</div>
```

jsp

Q&A 등록

*아이디	user01	비밀글 <input checked="" type="checkbox"/>
*제목	비밀글입니다.	
*내용	비밀글 작성입니다.	
 <input type="button" value="등록"/> <input type="button" value="취소"/>		

```
@PostMapping("pwdCheck")
public String pwdCheck(String password, String qnaNum, String idCheck){
    Object principal = SecurityContextHolder.getContext().getAuthentication().getPrincipal();
    UserDetails userDetails = (UserDetails)principal;

    if(bce.matches(password, userDetails.getPassword())&&(userDetails.getUsername().equals(idCheck))){
        return "redirect:/viewQna?qnaNum=" + qnaNum;
    }else {
        return "redirect:/listQna";
    }
}
```

controller

Q&A

번호	제목	아이디	작성일	답변상태
14	비밀글입니다	user01	24-03-26 19:24	N

비밀글입니다

비밀글

비밀번호

회원 비밀번호

확인 닫기

비밀글7

3_ Q&A 답변

Q&A

Q&A 목록				
번호	제목	아이디	작성일	답변상태
13	확인용입니다	user01	24-03-26 20:01	N
12	비밀글입니다	user01	24-03-26 19:24	N
11	비밀글8	user01	24-03-20 15:39	N

번호	제목	아이디	작성일	답변상태
13	확인용입니다	user01	24-03-26 20:02	Y
12	비밀글입니다	user01	24-03-26 19:24	N
11	비밀글8	user01	24-03-20 15:39	N

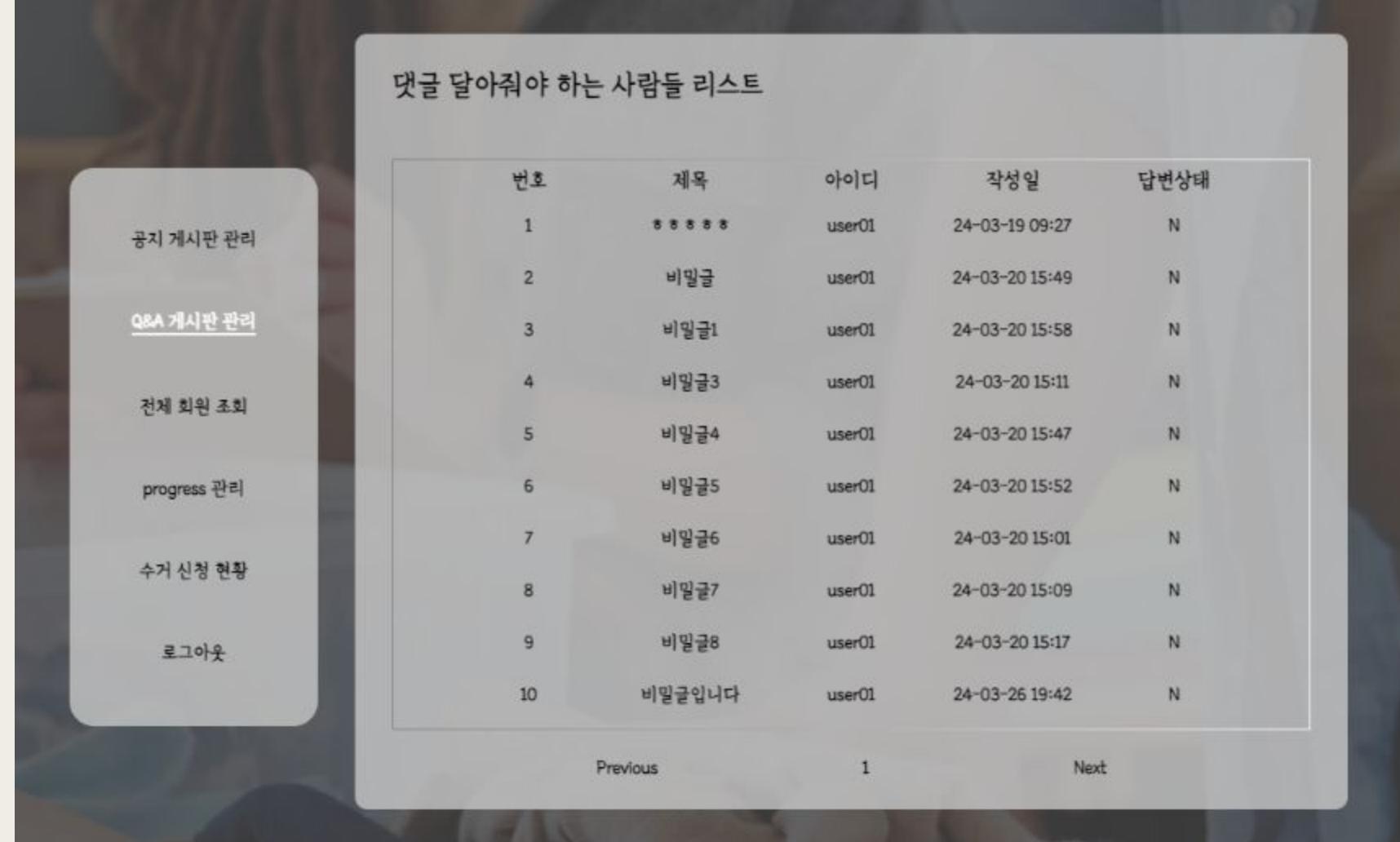


controller

```

@PostMapping("insertAnswer")
public String insertAnswer(QnaAnswer qnaAnswer) throws IllegalStateException, IOException {
    if(!qnaAnswer.getFile().isEmpty()) {
        String url = fileUploads(qnaAnswer.getFile());
        qnaAnswer.setUrl(url);
    }
    service.insertQnaAnswer(qnaAnswer);
    service.updateStatus(qnaAnswer.getQnaNum());
    return "redirect:/viewQna?qnaNum="+qnaAnswer.getQnaNum();
}

```



번호	제목	아이디	작성일	답변상태
1	*****	user01	24-03-19 09:27	N
2	비밀글	user01	24-03-20 15:49	N
3	비밀글1	user01	24-03-20 15:58	N
4	비밀글3	user01	24-03-20 15:11	N
5	비밀글4	user01	24-03-20 15:47	N
6	비밀글5	user01	24-03-20 15:52	N
7	비밀글6	user01	24-03-20 15:01	N
8	비밀글7	user01	24-03-20 15:09	N
9	비밀글8	user01	24-03-20 15:17	N
10	비밀글입니다	user01	24-03-26 19:42	N



controller

```

@GetMapping("listStatus")
public String listStatus(Model model, @RequestParam(value = "page", defaultValue = "1") int page) {
    int total = service.totalStatus();
    Paging paging = new Paging(page, total);
    List<Qna> listStatus = service.listStatus(paging);
    model.addAttribute("listStatus", listStatus);
    model.addAttribute("paging", paging);
    return "qnaAnswer/listQnaAnswer";
}

```

3_ Q&A 사진 수정



```
<div class="updatePhoto">
    <c:choose>
        <c:when test="${qna.url != null}">
            
            <div id="photoDelete"><i class="fa-solid fa-circle-xmark"
                onclick="photoDelete()"></i></div>
            <input type="hidden" value="${qna.delImg}" name="delImg" id="delImg" />
        </c:when>
    </c:choose>
    <div class="updateAnswer" id="img">
        <div id="img_input">
            <label for="file" id="photo">수정 사진</label>
            <input class="form-control" type="file" id="file" name="file"
                accept="image/*" onchange="imgShow(event); onclick="labelText(); hidden">
            <div id="image_container"></div>
        </div>
    </div>
</div>
```

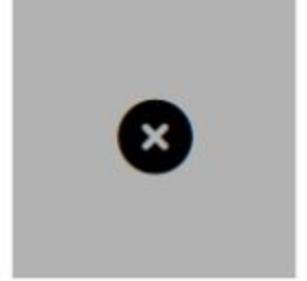
jsp

질문 수정

*제목
비밀글8

*내용
비밀글입니다


수정 사진


수정


수정

js

```
function labelText(){
    var text = document.querySelector("#photo");
    text.style.display = "none";
}

function photoDelete(){
    const qnaPhoto= document.querySelector("#qnaPhoto");
    const photoDeleteBtn = document.querySelector("#photoDelete");

    qnaPhoto.style.visibility = 'hidden';
    photoDeleteBtn.style.display = 'flex';
    delImg.value = true;
}
```

controller

```
public String update(Qna qna) throws IllegalStateException, IOException {
    if(!qna.getFile().getOriginalFilename().equals("")) {
        String url = fileUpload(qna.getFile());
        qna.setUrl(url);
    } else if(qna.isDelImg()) {
        File file = new File(path+qna.getUrl());
        file.delete();
        qna.setUrl(null);
    }
    service.update(qna);
    return "redirect:/viewQna?qnaNum="+qna.getQnaNum();
}
```

3_ Q&A

수정/삭제 권한

The diagram illustrates three different user scenarios for managing Q&A posts, each mapped to a portion of a shared JSP code snippet.

Non-member/User case: The UI shows a single "목록" (List) button. The corresponding JSP code handles users who are not anonymous or are not administrators.

```

<div id="button">
    <c:choose>
        <c:when test="${user == 'anonymousUser' || (qna.id != user.id && user.auth != 'ADMIN')}">
            <button value="목록" id="btnlist">
                <a href="listQna" id="listQna">목록</a>
            </button>
        </c:when>
        <c:when test="${user != 'anonymousUser' && user.auth == 'ADMIN' && qna.id != user.id}">
            <button value="목록" id="btnlist">
                <a href="listQna" id="listQna">목록</a>
            </button>
            <button value="삭제" id="btndelete">
                <a href="/deleteQna?qnaNum=${qna.qnaNum}" id="deleteQna">삭제</a>
            </button>
            <button value="댓글" id="btnAnswer">
                <a href="writeAnswer?qnaNum=${qna.qnaNum}" id="AnswerQna">댓글</a>
            </button>
        </c:when>
        <c:otherwise>
            <c:if test="${qna.id == user.id}">
                <button value="목록" id="btnlist">
                    <a href="listQna" id="listQna">목록</a>
                </button>
                <button value="수정" id="btnupdate">
                    <a href="updateQna" id="updateQna">수정</a>
                </button>
                <button value="삭제" id="btndelete">
                    <a href="/deleteQna?qnaNum=${qna.qnaNum}" id="deleteQna">삭제</a>
                </button>
            </c:if>
        </c:otherwise>
    </c:choose>
</div>

```

Admin case: The UI shows "목록", "삭제", and "댓글" buttons. The JSP code handles users who are not anonymous and are administrators.

Author case: The UI shows "목록", "수정", and "삭제" buttons. The JSP code handles users who are the author of the post.

The JSP code uses conditional logic (`<c:choose>`, `<c:when>`, `<c:otherwise>`) to determine which buttons are displayed based on the user's role and the post's author.

.jsp

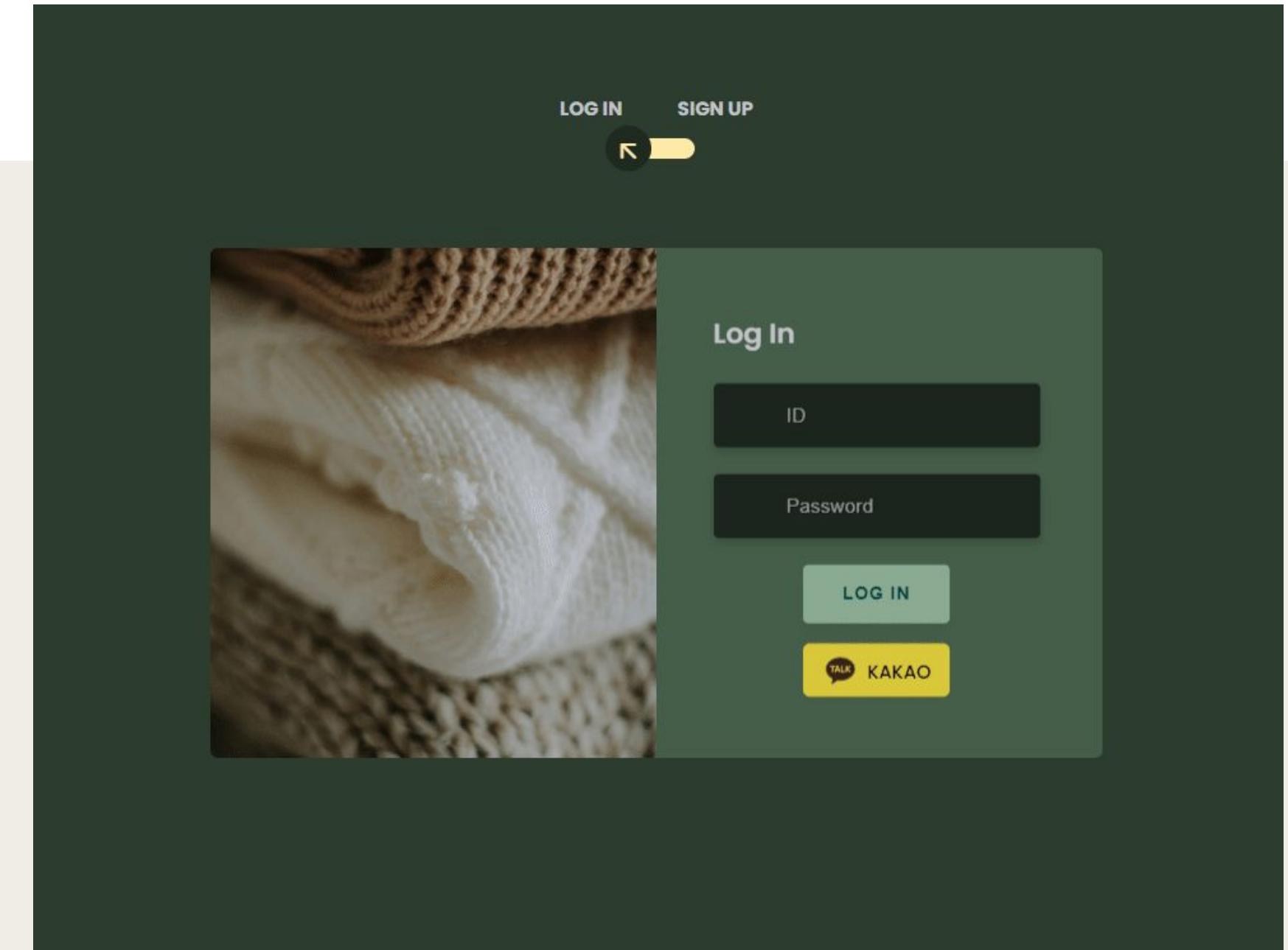
A photograph of a person with long dark hair, seen from behind, walking away on a narrow dirt path through a vast, rolling green field. The person is wearing a light-colored, button-down shirt and dark trousers. The landscape is hilly and covered in lush green grass.

MyPage / Manage User

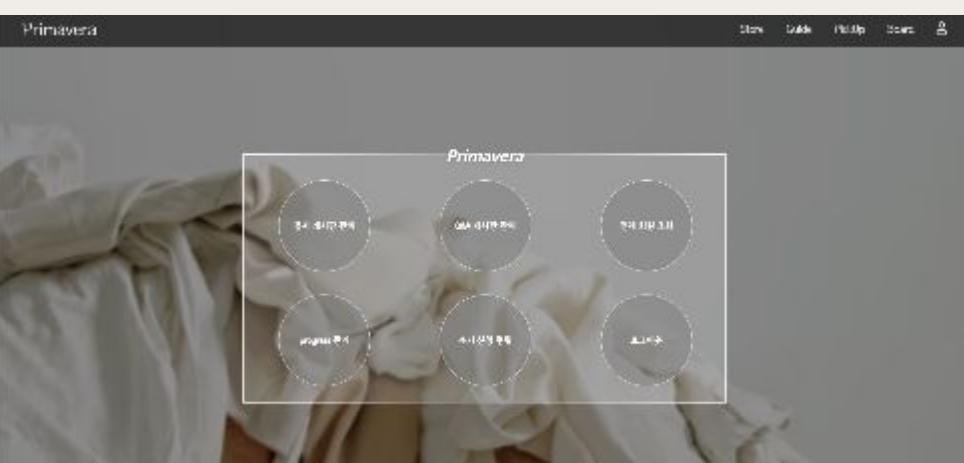
myPage

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<%@ taglib prefix="sec" uri="http://www.springframework.org/security/tags" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>

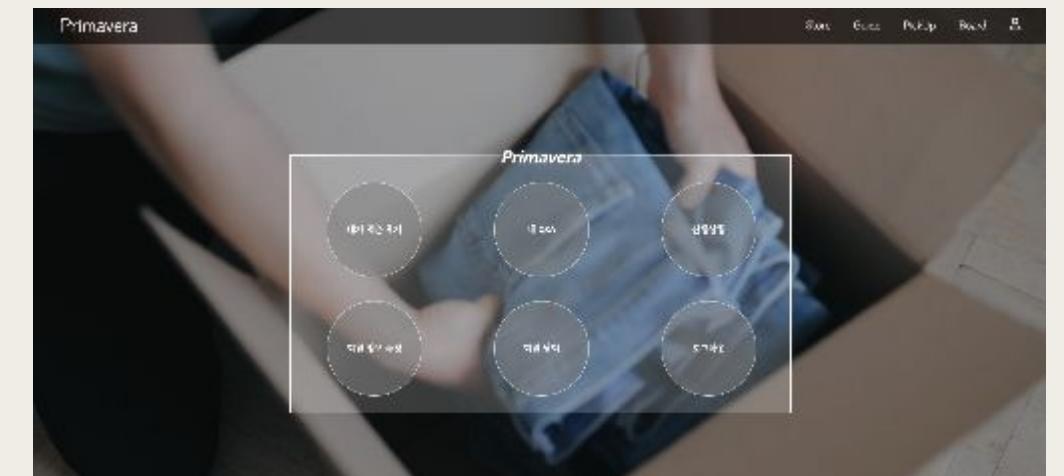
<body>
    <sec:authentication property="principal" var="user" />
    <c:choose>
        <c:when test="${user == 'anonymousUser'}">
            <main>
                <div class="card-front">
                    ...
                </div>
                <div class="card-back">
                    ...
                </div>
            </main>
        </c:when>
    </c:otherwise>
        <c:if test="${user.auth == 'ADMIN'}">
            ...
        </c:if>
        <c:if test="${user.auth == 'MEMBER'}">
            <c:choose>
                <c:when test="${user.password != null}">
                    ...
                </c:when>
                <c:otherwise>
                    ...
                </c:otherwise>
            </c:choose>
            <c:choose>
                <c:when test="${user.kakao == 'Y' && user.password == null}">
                    ...
                </c:when>
                <c:otherwise>
                    ...
                </c:otherwise>
            </c:choose>
        </c:otherwise>
    </c:choose>
</body>
</html>
```



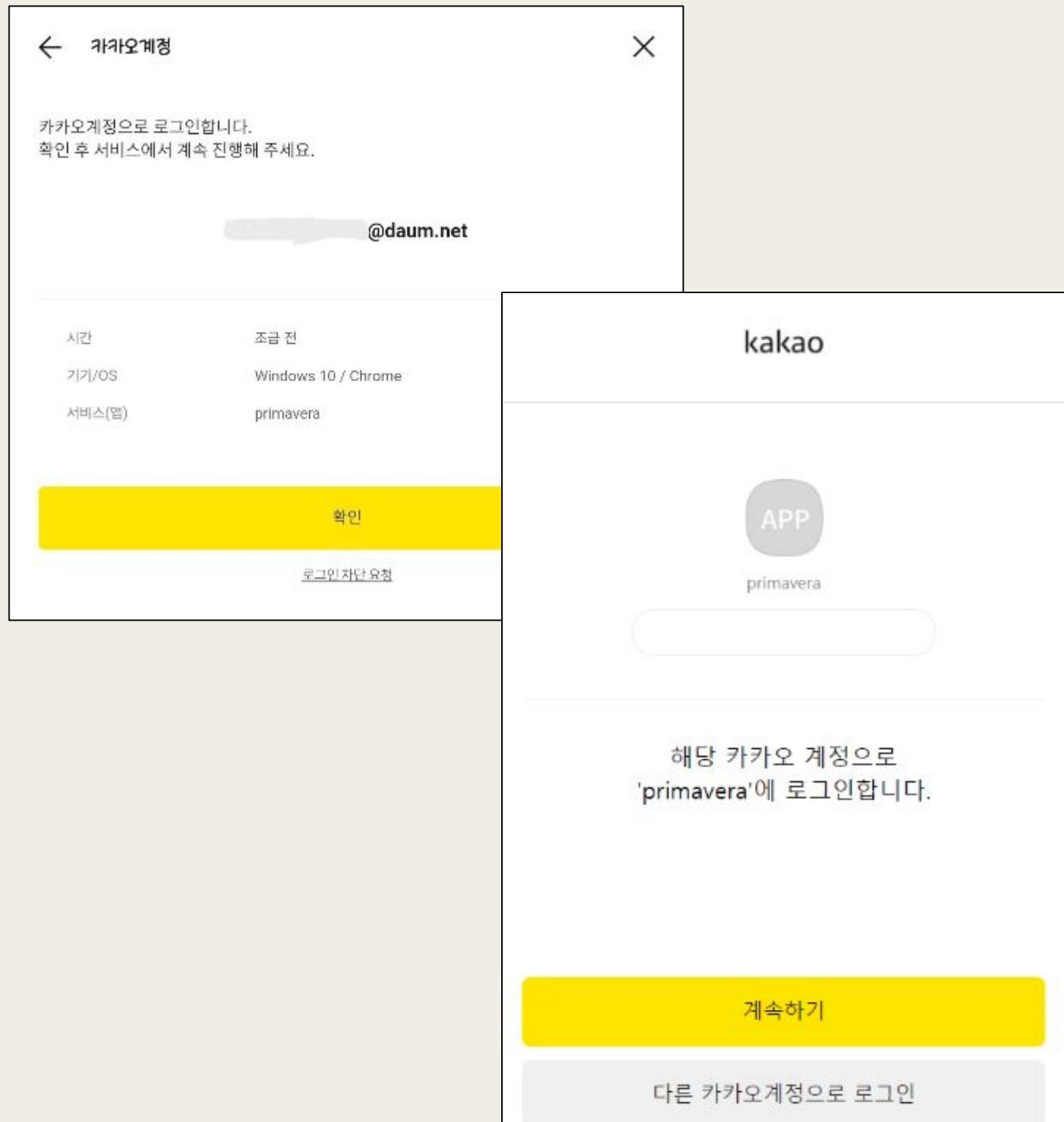
관리자 myPage



회원 myPage



KAKAO LOGIN



```
@GetMapping("/kakaologin")
    public String Kakaologin(@RequestParam(value = "code", required = false) String code) throws
Throwable {
    String access_Token = service.getAccessToken(code);
    User userInfo = service.getUserInfo(access_Token);
    session.invalidate();
    session.setAttribute("kakaoName", userInfo.getName());
    session.setAttribute("kakaoId", userInfo.getId());
    return "user/myPage";
}
```

```
public String getAccessToken(String authorize_code) throws IOException {
    String access_Token = "";
    String refresh_Token = "";
    String reqURL = "https://kauth.kakao.com/oauth/token";
    try {
        URL url = new URL(reqURL);
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
        conn.setRequestMethod("POST");
        conn.setDoOutput(true);

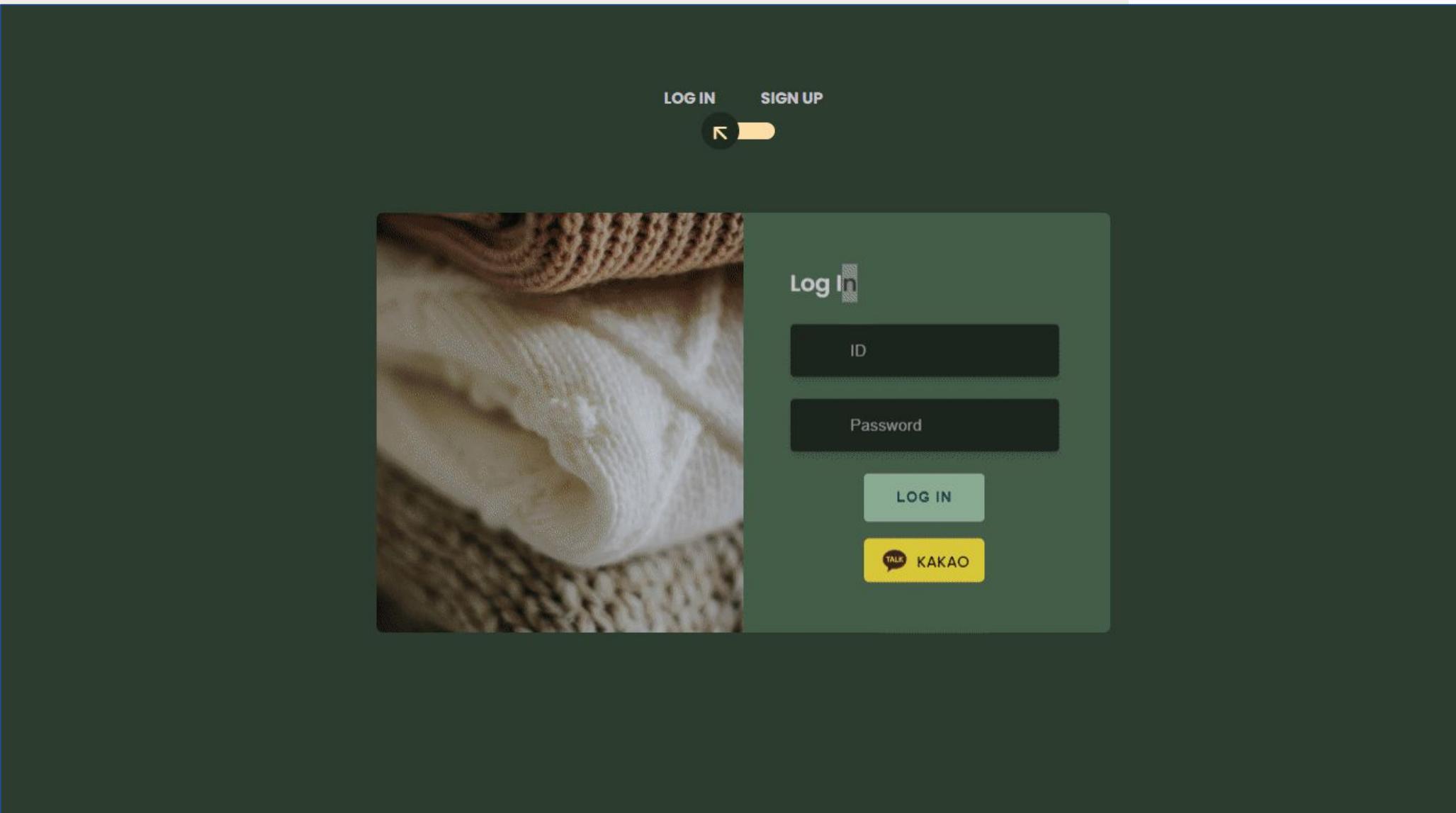
        BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(conn.getOutputStream()));
        StringBuilder sb = new StringBuilder();
        sb.append("grant_type=authorization_code");
        sb.append("&client_id=2aa4500445d4b2a832d23e0d819fb737");
        sb.append("&redirect_uri=http://localhost:8080/kakaologin");
        sb.append("&code=" + authorize_code);
        bw.write(sb.toString());
        bw.flush();

        int responseCode = conn.getResponseCode();

        BufferedReader br = new BufferedReader(new InputStreamReader(conn.getInputStream()));
        String line = "";
        String result = "";
        while ((line = br.readLine()) != null) {
            result += line;
        }
        JsonParser parser = new JsonParser();
        JsonElement element = parser.parse(result);

        access_Token = element.getAsJsonObject().get("access_token").getAsString();
        refresh_Token = element.getAsJsonObject().get("refresh_token").getAsString();
        br.close();
        bw.close();
    } catch (IOException e) {
        e.printStackTrace();
    }
    return access_Token;
}
```

KAKAO LOGIN



```
public User getUserInfo(String access_Token) {
    HashMap<String, Object> userInfo = new HashMap<String, Object>();
    String reqURL = "https://kapi.kakao.com/v2/user/me";
    try {
        URL url = new URL(reqURL);
        HttpURLConnection conn = (HttpURLConnection) url.openConnection();
        conn.setRequestMethod("GET");
        conn.setRequestProperty("Authorization", "Bearer " + access_Token);
        int responseCode = conn.getResponseCode();
        BufferedReader br = new BufferedReader(new InputStreamReader(conn.getInputStream()));
        String line = "";
        String result = "";
        while ((line = br.readLine()) != null) {
            result += line;
        }
        JsonParser parser = new JsonParser();
        JsonElement element = parser.parse(result);
        JsonObject properties = element.getAsJsonObject().get("properties").getAsJsonObject();
        String nickname = properties.getAsJsonObject().get("nickname").getAsString();
        String id = element.getAsJsonObject().get("id").getAsString();
        userInfo.put("name", nickname);
        userInfo.put("id", id);
    } catch (IOException e) {
        e.printStackTrace();
    }
}
```

FIND ID

```
● ● ●  
@PostMapping("/findId")  
public String findId(User user, Model model) {  
    model.addAttribute("finder", user);  
    if (service.findId(user) != null) {  
        model.addAttribute("user", service.findId(user));  
        return "account/findIdResult";  
    } else {  
        return "account/findFail";  
    }  
}
```

해당 사용자가 존재하지 않습니다.

사용자님의 아이디는 **user03**입니다.

[아이디 찾기](#) [회원가입](#)

[해당 아이디로 로그인](#) [해당 아이디로 비밀번호 재설정](#)

아이디 찾기

이름

이메일

[아이디 찾기](#)

[비밀번호 찾기](#) | [로그인](#) | [회원가입](#)

이메일 인증 버튼 활성화 조건

```
<script type="text/javascript">
...
idDupCheck = false;

$('#id').keyup(() => {
    const id = $('#id').val();

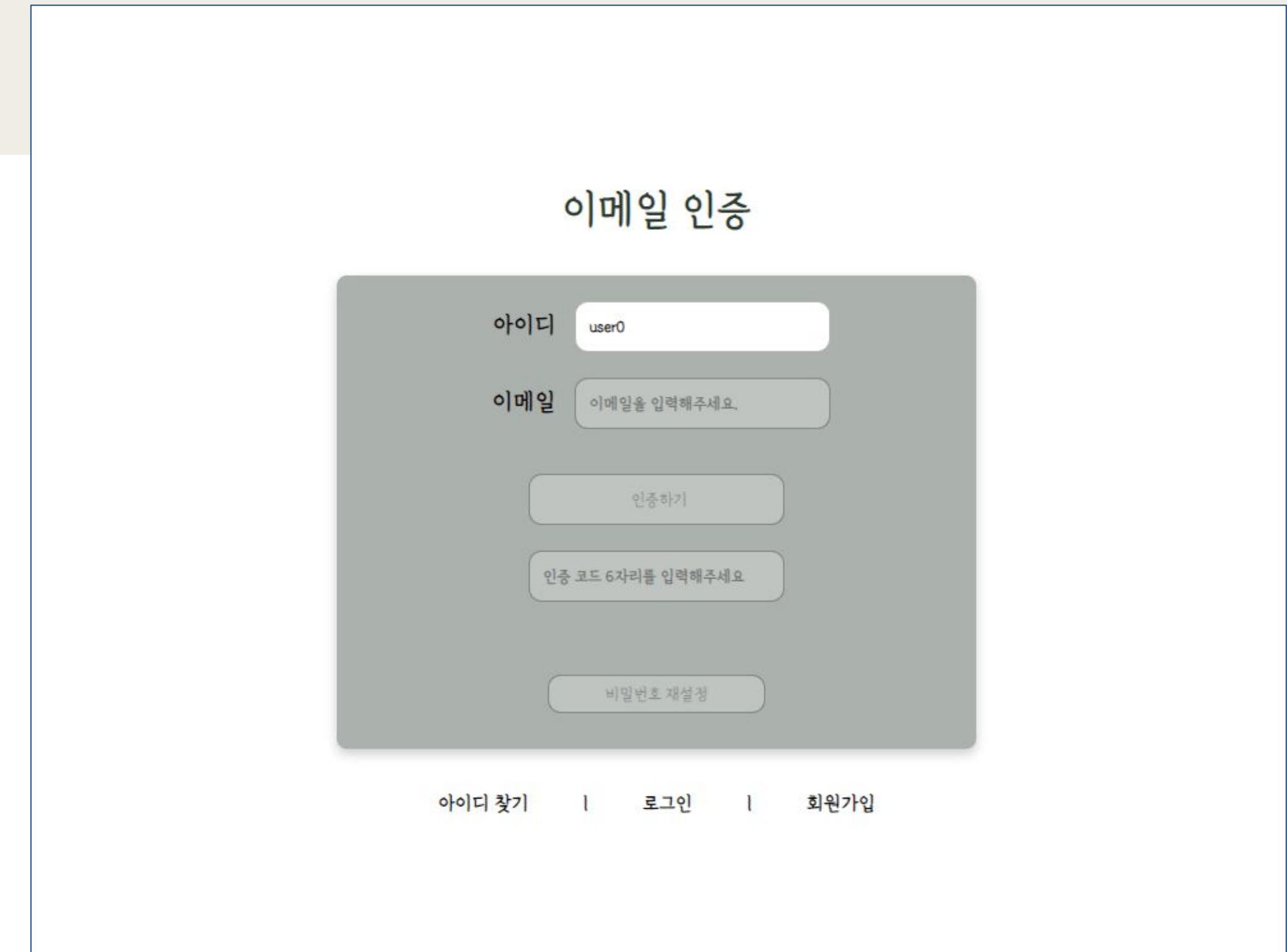
    $.ajax({
        type: "post",
        url: "/check",
        data: "id=" + id,

        success: function (result) {
            if (result) {
                // 아이디가 존재할 경우 이메일 인증 버튼 활성화
                $('#email').attr('disabled', false);
                idDupCheck = true;
            } else {
                // 아이디가 존재하지 않을 경우 이메일 인증 버튼 비활성화
                $('#email').attr('disabled', true);
                idDupCheck = false;
            }
        }
    })
});

$('#email').keyup(() => {
    const id = $('#id').val();
    const email = $('#email').val();

    $.ajax({
        type: "post",
        url: "/checkUser",
        data: $('#frm').serialize(),

        success: function (result) {
            if (result) {
                $('#emailAuth').attr('disabled', false);
            } else {
                $('#emailAuth').attr('disabled', true);
            }
        }
    })
...
</script>
```



Email Authentication

localhost:8080 내용:

인증 코드가 입력한 이메일로 전송 되었습니다.

확인

```
$("#emailAuth").click(function () {
  const email = $("#email").val();

  $.ajax({
    url: './EmailAuth',
    data: {
      email: email
    },
    type: 'POST',
    dataType: 'json',
    success: function (result) {
      $("#authCode").attr("disabled", false);
      code = result;
      alert("인증 코드가 입력한 이메일로 전송 되었습니다.");
    }
});
```

비밀번호 변경 인증 이메일입니다. 받은편지함 X

 primavera240327@gmail.com
나에게 ▾

인증 코드는 732155입니다.
해당 인증 코드를 인증 코드 확인란에 기입하여 주세요.

```
@ResponseBody
@PostMapping("/EmailAuth")
public int emailAuth(String email) {

    Random random = new Random();
    int checkNum = random.nextInt(888888) + 111111;

    String setForm = "primavera240327@gmail.com";
    String toMail = email;
    String title = "비밀번호 변경 인증 이메일입니다.";
    String content = "인증 코드는 " + checkNum + "입니다. " + "<br>" + "해당 인증 코드를 인증 코드 확인란에
                    기입하여 주세요.";

    MimeMessage message = mailSender.createMimeMessage();
    try {
        MimeMessageHelper helper = new MimeMessageHelper(message, true, "utf-8");
        helper.setFrom(setForm);
        helper.setTo(toMail);
        helper.setSubject(title);
        helper.setText(content, true);
        mailSender.send(message);
    } catch (MessagingException e) {
        e.printStackTrace();
    }
    return checkNum;
}
```

SIGN UP

회원가입

아이디

비밀번호

비밀번호 확인

이름

휴대폰번호

이메일

동의합니다.
 네, 동의합니다.

* 수거 신청을 위해 [이용약관](#) 및 [개인정보 수집에 동의합니다.](#)

```
$('#userPhone').keyup(() => {
    const phone = $('#userPhone').val();
    $.ajax({
        type: "post",
        url: "/checkphone",
        data: "phone=" + phone,
        success: function (result) {
            if (result) {
                $('#phoneCheckresult').text("전화번호 사용 불가").css("color", "black");
                phoneDupCheck = true;
            } else if (phone == '') {
                $('#phoneCheckresult').html(' ');
            } else if(!result && !phoneCheck) {
                $('#phoneCheckresult').text("전화번호 사용 가능").css("color", "white");
                phoneDupCheck = false;
            }
        }
    })
})
```

```
function validate(){
    if(idCheck){
        userId.focus();
        return false;
    } else if (idDupCheck){
        userId.focus();
        return false;
    } ...
} else if(emailCheck){
    email.focus();
    return false;
} else if (emailDupCheck){
    email.focus();
    return false;
}
return true;
}
```

PWD CHECK

비밀번호 인증

My Review

My Q&A

My Pick Up

회원 정보 수정

회원 탈퇴

로그아웃

정보를 안전하게 보호하기 위해
비밀번호를 다시 한 번 확인합니다.

비밀번호가 타인에게 노출되지 않도록 항상 주의해주세요.

아이디 : user01

기존 비밀번호

비밀번호 확인

```
@GetMapping("/userUpdate")
public String update() {
    Object principal = SecurityContextHolder.getContext().getAuthentication().getPrincipal();
    User userDetails = (User) principal;
    if (userDetails.getPassword() == null) {
        return "user/updateUser";
    }
    return "user/updateCheck";
}
```

```
@PostMapping("/updateCheck")
public String updateCheck(String password, Model model) {

    Object principal = SecurityContextHolder.getContext().getAuthentication().getPrincipal();
    User userDetails = (User) principal;

    model.addAttribute("pwd", password);

    if (bcpe.matches(password, userDetails.getPassword()) ||
password.equals(userDetails.getPassword())) {
        model.addAttribute("user", userDetails);
        return "user/updateUser";
    } else {
        return "redirect:/userUpdate";
    }
}
```

UPDATE USER

회원 정보 수정

아이디	user04
비밀번호
이름	사용자
휴대폰 번호	01044445555
이메일	user04@gmail.com

```
@ResponseBody  
@PostMapping("/editphone")  
public int editphone(String phone) {  
    Object principal = SecurityContextHolder.getContext().getAuthentication().getPrincipal();  
    User user = (User) principal;  
  
    User editphone = service.phoneCheck(phone);  
  
    boolean check = false;  
  
    if (editphone != null) {  
        check = user.getPhone().equals(editphone.getPhone());  
    } else {  
        check = false;  
    }  
  
    if (check) {  
        return 2;  
    } else if (editphone == null) {  
        return 1;  
    }  
    return 3;  
}  
  
$( '#userPhone' ).keyup((e) => {  
    let phone = $( '#userPhone' ).val();  
    const regExp = /^01(0|1|[6-9])[0-9]{3,4}[0-9]{4}$/;  
  
    $.ajax({  
        type: "post",  
        url: "/editphone",  
        data: "phone=" + phone,  
        success: function (result) {  
            if (regExp.test(phone)){  
                $('#phonereg').text("");  
                switch(result){  
                    case 1:  
                        $('#phoneCheckresult').text("전화번호 사용 가능");  
                        phoneDupCheck = false;  
                        phoneCheck = true;  
                        break;  
                    case 2:  
                        $('#phoneCheckresult').text("기존 전화번호와 동일합니다.");  
                        phoneDupCheck = false;  
                        phoneCheck = true;  
                        break;  
                    case 3:  
                        $('#phoneCheckresult').text("전화번호 사용 불가");  
                        phoneDupCheck = true;  
                        phoneCheck = false;  
                        break;  
                }  
            } else if (!regExp.test(phone)){  
                $('#phoneCheckresult').text("전화번호 사용 불가");  
                $('#phonereg').text("유효한 전화번호를 입력해주세요.");  
                phoneCheck = false;  
                phoneDupCheck = false;  
            } else if (phone === ""){  
                $('#phoneCheckresult').html(' ');  
                phoneDupCheck = false;  
                phoneCheck = false;  
            }  
        }  
    });  
});
```

ORDER BY

전체 수거 신청 정보 보기

주문번호 순 ▾

주문번호	아이디	이름	전화번호	우편번호	도로명 주소	상세 주소	수거일	수거 신청 무게	금액	공동 현관 비밀번호	요청사항
2	user01	사용자	01011111111	12345	12387	12538	2024-03-31	14	5600		
3	user01	사용자	01011111111	12345	12387	12538	2024-04-10	7	2800		
4	user02	사용자	01022222222	9845162	984516	984516	2024-03-30	7	2800		
5	user02	사용자	01022222222	9845162	984516	984516	2024-04-19	8	3200		
6	user02	사용자	01022222222	9845162	984516	984516	2024-04-08	17	6800		
7	user03	사용자	01033333333	12345	12345	12345	2024-03-28	6	2400		
8	user03	사용자	01033333333	12345	12345	12345	2024-03-31	8	3200		

Previous 1 Next

```
$( "#sorts" ).change(() => {
    var sort = $('#sorts option:selected').val();
    location.href = "/showAllCollect?sorts=" + sort;
});
```

```
● ● ●
<select name="sorts" id="sorts">
    <option value="1">주문번호 순</option>
    <option value="3" <c:if test="#{param.sorts == '3'}">selected</c:if>> 이름 순 </option>
    <option value="8" <c:if test="#{param.sorts == '8'}">selected</c:if>> 수거일 순 </option>
    <option value="11" <c:if test="#{param.sorts == '11'}">selected</c:if>> 수거 무게 순 </option>
</select>
```

```
<select id="showAllCollectororderby" resultMap="collectMap">
    SELECT order_num, id, name, phone, post_code, road_address,
    detail_address, collection_date, door_pwd, request, kg
    FROM collect
    <if test="sorts != 0">
        ORDER BY #{sorts}
    </if>
    LIMIT #{offset}, #{limit}
</select>
```



04_시연 및 마무리

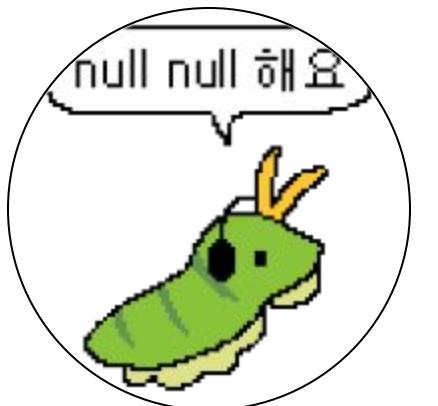
Primavera_봄의 시작

느낀 점



권예빈

백지부터 사이트를 채우는 게 처음에는 막막했는데, 팀원들과 협업하여 사이트의 구색을 갖춰가는 게 즐거웠습니다. 함께 작업하면서 다른 팀원들의 코드를 읽다보니 코드 보는 눈도 키울 수 있었고, 기능 구현을 위해 검색하고 적용하는 과정에서 알게 된 것들도 많고, 실시간으로 적용되는 게 눈에 보이다보니 코드 공부에 더 흥미를 느끼게 된 것 같습니다. 그리고 개발 과정보다 유지보수 과정이 훨씬 오래 걸린다는 것을 체감하면서, 초기에 요구사항을 정의하는 것이 중요하다는 것을 알게 되었습니다.



정대윤

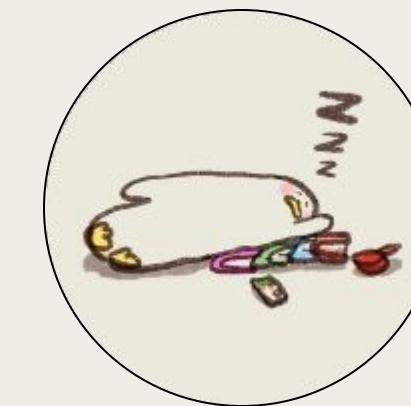
프로젝트를 진행하면서 데이터가 어디서 어디로 어떻게 이동하는지 다시 되새겨볼 수 있는 시간이었고, 오류가 나면 검증을 어떻게 해야 어디가 문제인지 찾을 수 있는지 좀 더 알게 되었습니다.

다른 분들과 프로젝트를 진행하면서 꾸준한 소통과 피드백을 통한 빠른 수정의 중요성도 느꼈습니다. 좀 더 발전할 수 있었던 굉장히 좋은 시간이었다고 생각합니다.



김현수

수업을 통해 배운 내용을 토대로 프로젝트에 임하는 가운데 방대한 정보들이 UI와 기능 구현을 통해 하나하나씩 맞춰지면서 정리된 게 강사님께서 말씀하신 대로 성장한 것 같아서 좋았던 것 같다!



손민정

적용해보고 싶었던 기능을 직접 찾아보고 다양한 코드에 많이 노출되면서 자연스럽게 공부할 수 있어 좋았고 팀원들과 소통하면서 같이 힘을 합쳐 만들어낸 결과물이라는 생각에 뿌듯하였습니다.

비록 막힌 부분도 있었지만 프로젝트를 통해 부족한 점이 어느 부분인지 깨닫게 되어 이를 보완하며 공부해야겠다고 생각하였고 이후 프로젝트에서는 어떤 점들을 개선해야 할지 캐치할 수 있어 좋았습니다. 꾸준히 코드 공부를 하고 혼자서 페이지 구현을 할 수 있을 정도로 실력을 키우고 싶다고 생각하였습니다.

좋았던 점

매일 진행된 회의와 피드백으로 원활한 의사소통

notion을 통한 세부적인 프로젝트 관리

다양한 의견 수용

정확한 역할 분담 및 신속한 수행

개선할 점

전체적인 UI 디자인 통일

경로, 함수명 지정 형식 통일

기능 구현에 대한 요구사항 정의서

header.jsp 분리

에러 페이지 디자인



Thank you