



SHOEFLY

Table of Contents

01

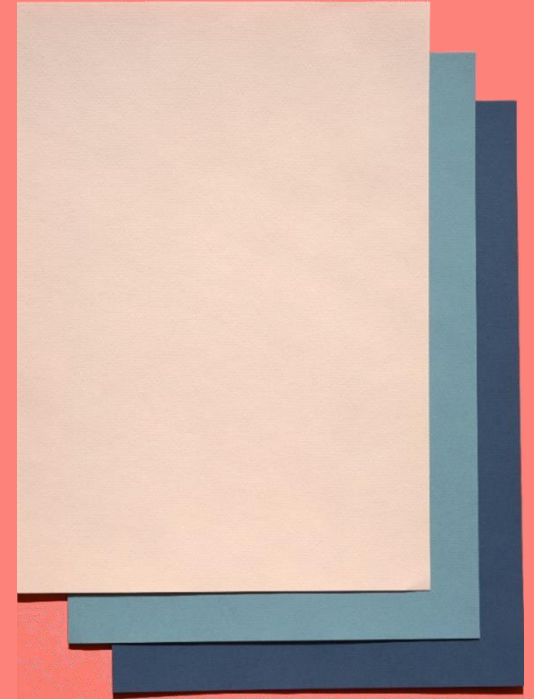
■ 소개 및 개요

02

■ 개발환경 및 기획

03

■ 프로젝트 구조, 담당파트 구현





01 소개 및 개요

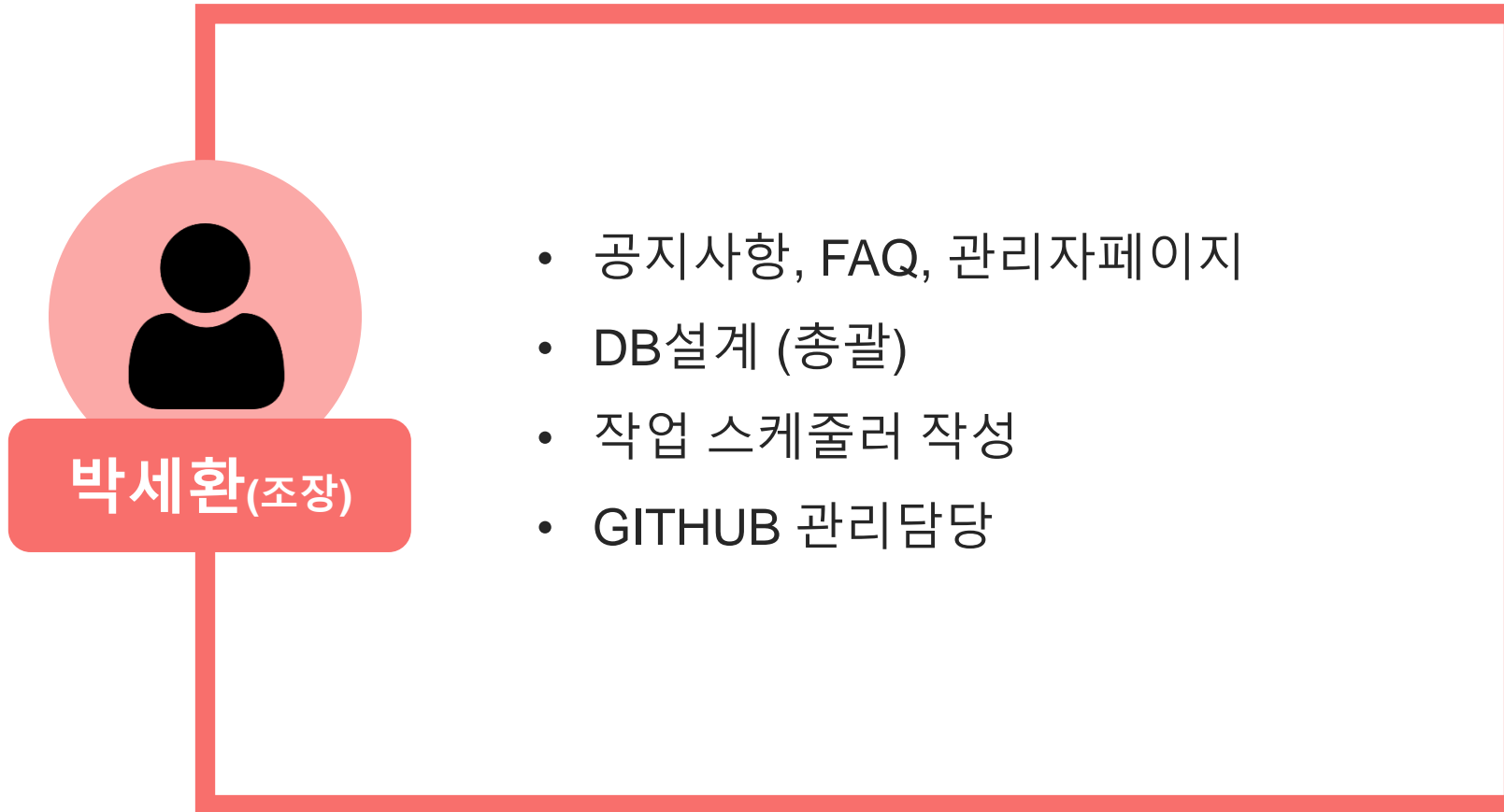
SHOEFLY란?



한정판 상품에 대해 안전한 거래를 제공합니다

- 거래를 위해 서로의 개인 정보를 공유할 필요가 없습니다. **SHOEFLY**가 안전한 거래를 도와드립니다.
- 철저한 검수를 통해 가품에 대한 불안감을 없애 드립니다.
- 거래 중인 상품들을 통해 시세 파악이 가능합니다.

프로젝트 역할





02 개발환경 및 기획

개발 환경

구현 언어



JSP



Servlets
& Jsp



DB 서버

ORACLE®



WAS



Apache Tomcat

서버 언어



사용 도구



spring®



eclipse



MyBatis

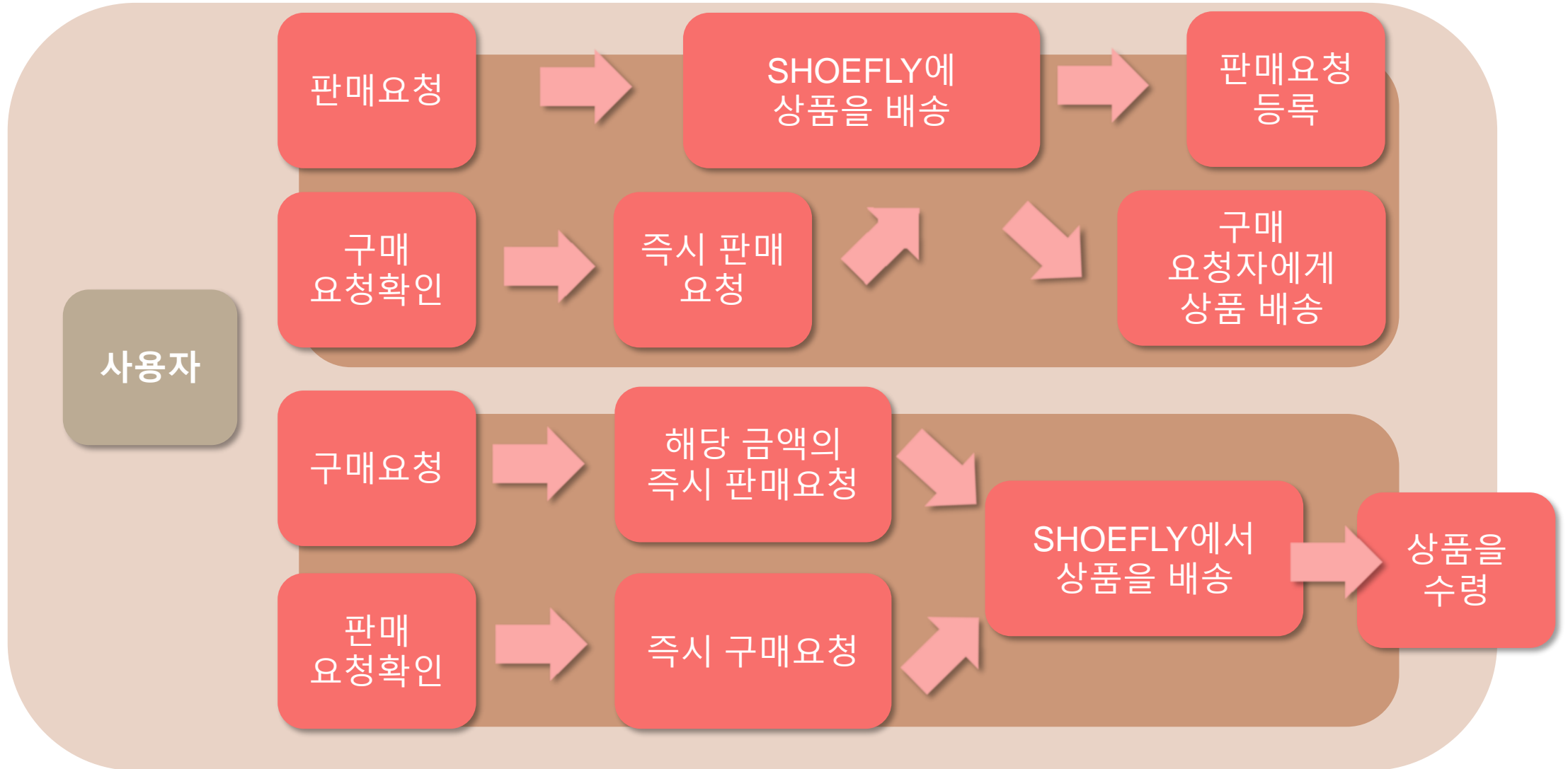


API



도로명주소

요구사항 명세서



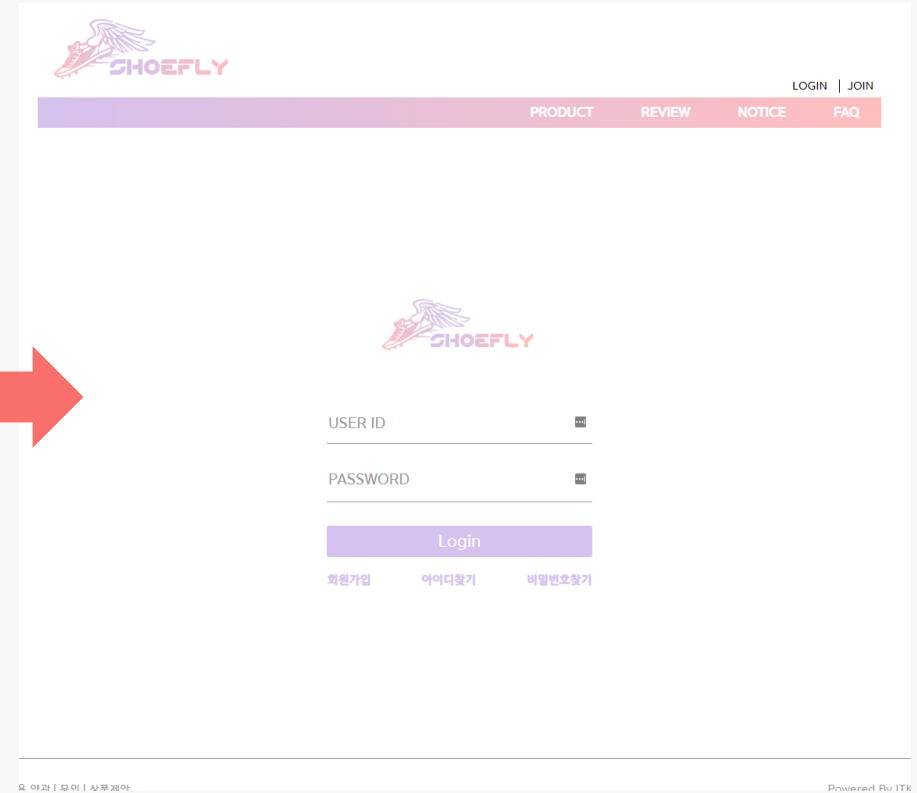
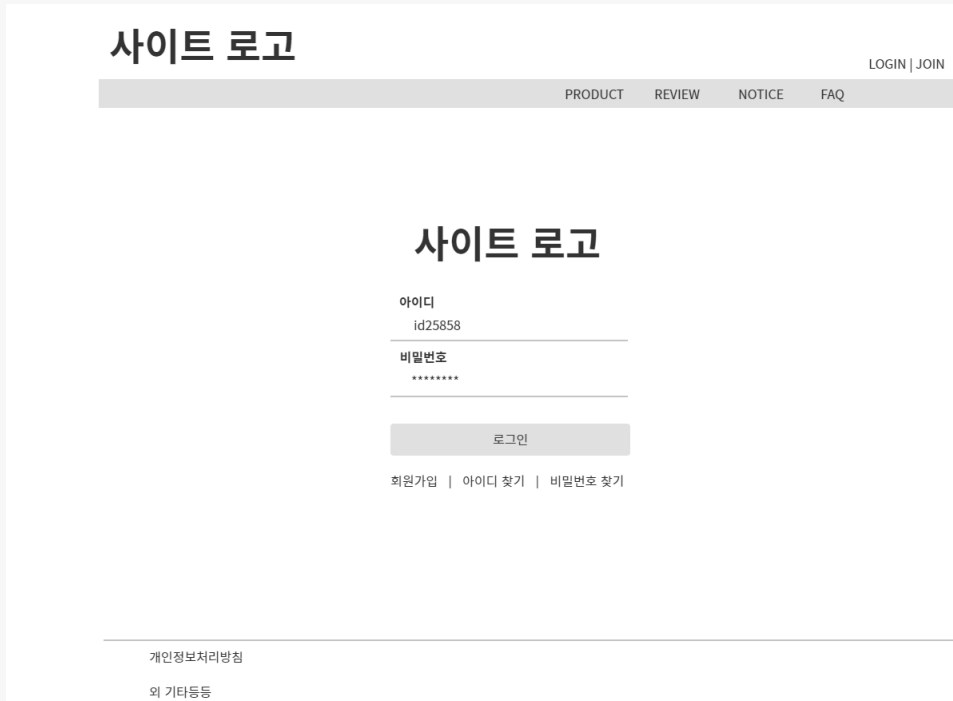
요구사항 명세서

1depth	2depth	3depth	4depth	기능(대분류)	기능(소분류)	외부api	구현여부
SHOEFLY				어디서든 회사로고 클릭시 메인화면으로 이동	로고이미지 제작해야함		O
				상단 우측의 로그인 메뉴를 통해 로그인 가능	로그인 성공시 우측 상단의 로그인링크가 로그아웃링크로변경		O
				상단 우측의 회원가입 메뉴를 통해 회원가입 가능	로그인 성공시 "유저id"님 반갑습니다 생성, 로그인->로그아웃 메뉴로 변경, 회원가입 ->마이페이지로 변경		O
				MOST POPULAR은 거래횟수 순으로 조회, RECOMANDED는 관리자가 설정			△
	비로그인	PRODUCT 메뉴 상품 1개 상세 페이지	상품리스트 조회	비 로그인이어도 상품 검색 가능	브랜드와 사이즈는 체크박스로, 가격은 최소 최대값 입력으로 조회 가능 , 구매 및 판매클릭시 로그인창으로 연결		O
			상품 조회	비 로그인이어도 상품 1개 페이지 보기 가능, 구매및 판매는 불가	사이즈선택시 상품의 즉시구매가 즉시판매가 조회가능 클릭시 로그인이 필요하다는 메시지 뜸		O
	로그인	로그인 페이지	로그인	로그인 성공시 session에 해당 유저 정보 올림			O
			아이디 찾기	가입 당시 이메일로 인증 받은 뒤 아이디를 알려줌			O
			비밀번호 찾기	가입 당시 아이디와 이메일로 인증 받은 뒤 비밀번호 재설정 가능			O
		로그아웃 페이지	로그아웃	session에서 제거			O
	회원	마이페이지	구매현황	구매했던 상품의 이미지, 제품명, 구입가격 조회 가능	배송완료 상태일 경우 후기작성 가능, 후기 미등록시 1회 후기등록 가능, 후기 등록 되어있을 시 후기보기 가능		O
			판매현황	판매했던 상품의 이미지, 제품명, 판매가격 조회 가능			O
			회원정보 변경	이름, 비밀번호 변경 가능, 주소록 등록가능	주소록 등록 수정 삭제 가능	주소api	O
			회원탈퇴	탈퇴해도 DB에 남아있음	delete가 아닌 update		O
		회원가입 페이지	아이디 정규식 검사	소문자와 숫자 조합으로 5자이상 10자이하 가능	key up 이벤트		O
			아이디 중복검사	DB상의 아이디 중복 여부 확인	중복아이디는 가입 불가 메시지 띄우기		O
			비밀번호 정규식 검사	영여 1이상, 숫자 1이상, 5글자 이상 15글자 이하, 특수문자 가능	key up 이벤트		O
			비밀번호 재확인	비밀번호와 비밀번호 확인 일치 여부 확인	DB통과 X		O
			이메일 정규식검사	ex) XX@XX.XX, YY@YY.YY.YY	key up 이벤트		O
			이메일 중복검사	DB상의 이메일중복 여부 확인	중복이메일은 가입 불가 메시지 띄우기		O
			이메일인증	인증코드 받고 인증 되어야 가입 가능	이메일인증 사용		O
	회원통합	조회	상품조회	상품명, 브랜드, 사이즈, 가격으로 조회가능	검색창에 미입력, 체크박스 미체크 시에도 조회 가능, 페이지처리		O
			판매요청서 작성	판매 희망 가격입력, 반송받을 수 있는 주소입력, 주소는 기존주소가능 신규주소 가능	"검수기준, 반송기준 등 규정에 동의합니다" 필수체크박스에 체크해야 판매 가능, 상세주소는 null이어도 가능	주소api	O
		구매	즉시판매	구매대기중인 상품에 대하여 즉시 판매 가능		주소api	O
			구매요청서 작성	판매 희망 가격입력, 상품 수령할 주소 입력, 주소는 기존주소가능 신규주소 가능	"검수기준, 반송기준 등 규정에 동의합니다" 필수체크박스에 체크해야 판매 가능, 상세주소는 null이어도 가능	주소api	O
			즉시구매	구매대기중인 상품에 대하여 즉시 구매 가능		주소api	O
			구매완료	구매가 완료되었을 때 페이지에서 구매정보 조회 가능			O
			결제	결제 가능	결제api사용(시간여유가 된다면구현)	결제api	X
		알림창	알림	판매 등록해둔 상품 판매시 알림 구매 등록해둔 상품 구매시 알림(최저가) 후기에 댓글이 달렸을 경우			X
			알림조회	우측 상단에서 알림이 도착했는지 확인가능 후기 작성시 제목입력, 내용입력, 이미지첨부 가능	"유저id님 반갑습니다" 왼쪽에 알림 아이콘 생성, 알림 수신했을때와 수신안했을때의 아이콘의 색을 다르게 함		X
		상품 후기	후기 게시물 작성	후기수정, 삭제 조회수 증가	로그인유저만 작성 가능, 제목과 내용은 필수입력, 이미지 첨부는 필수아님 후기 작성자만 후기의 수정과 삭제 가능 조회수 증가 / 쿠키를 사용하여 하루에 1회 증가만 가능		O
			후기 게시물 사진첨부	후기 게시물 작성시 사진 첨부 가능	사진첨부기능은 사용시 바로 화면에 표시됨		O
			후기 댓글 작성	상품 후기에 댓글달기 가능	본인의 게시물에도 댓글달기 가능		O
			후기 댓글 수정삭제	상품 후기 수정 삭제 가능	댓글작성자만 댓글의 수정과 삭제 가능		O
	관리자	NOTICE 메뉴	공지사항	관리자만 NOTICE 미리보기, 추가, 수정, 삭제 가능	페이지처리		O
			F&Q 메뉴	자주묻는 질문	페이지처리		O
		회원관리	가입/탈퇴 관리	회원의 가입/탈퇴 변경 가능	DB에는 남아있음		O
			주소록 관리	회원의 주소록 조회가능	팝업창으로 기능 구현		O
		후기관리	임시비밀번호	회원에게 임시비밀번호 발급 가능	팝업창으로 기능 구현, 이메일로 임시비밀번호 전송		O
			후기관리	후기 게시물 삭제/복구 가능	DB에는 남아있음		O
		거래관리	상품관리	구매 판매 가능한 상품 종류 조회 가능	상품번호, 상품명, 브랜드 등으로 조회 가능, 상품의 이미지 보기 가능		O
			결제된 상품 배송준비	결제된 상품 배송상태를 배송준비에서 배송중으로 변경하기	배송대기에서 배송중으로 처리 가능		△
			결제된 상품 배송완료	결제된 상품 배송상태를 배송중에서 배송완료로 변경하기	배송대기에서 배송중으로 처리 가능		△

작업 스케줄러

			7월																								8월			
			1주차				2주차							3주차							4주차						5주차			
			8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1	2	3	4
기획	프로젝트 기획		기획 및 프로토 타입 작성																											
	요구사항 명세		요구사항 명세서 작성																											
설계	테이블 설계		테이블 구조 설계																											
	데이터베이스 구현		테이블 작성					샘플 데이터 작성																						
작업	회원	로그인 / 회원가입			구현														CSS / JavaScript											
		마이페이지				구현														CSS / JavaScript										
	상품	상품 보기/검색			구현															CSS / JavaScript										
		회원 상품구매			구현																CSS / JavaScript									
		회원 상품판매			구현																CSS / JavaScript									
	서비스	상품후기			구현															CSS / JavaScript										
		관리자	회원관리			구현															CSS / JavaScript									
	상품관리				구현																CSS / JavaScript									
	공지사항				구현																CSS / JavaScript									
	FAQ				구현																CSS / JavaScript									
발표	PPT작성 및 준비																									PPT작성 및 준비				

스토리보드



스토리 보드를 작성하여 페이지 구현에 참고하였습니다.

(Oven : <https://ovenapp.io/project/iyquE9YK1540VxupmOqj1FcOoXf19ay5#Wlz7j>)

DB TABLE 구조

PK	AI	FKNull	Name	Type
✓	✓	+	FAQ_NO	NUMBER
		✓	MEMBER_ID	VARCHAR2(30)
		+	QUESTION	VARCHAR2(1000)
		+	ANSWER	VARCHAR2(4000)
		+	POSTDATE	DATE
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	MEMBER_ADDRESS_NO	NUMBER
		✓	MEMBER_NO	NUMBER
		+	NAME	VARCHAR2(100)
		+	ADDR1	VARCHAR2(100)
		✓	ADDR2	VARCHAR2(100)
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	NOTICE_NO	NUMBER
		✓	MEMBER_ID	VARCHAR2(30)
		+	TITLE	VARCHAR2(1000)
		+	CONTENT	VARCHAR2(4000)
		+	POSTDATE	DATE
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	ALARM_NO	NUMBER
		✓	MEMBER_ID	VARCHAR2(30)
		+	CONTENT	VARCHAR2(300)
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	MEMBER_NO	NUMBER
		+	MEMBER_ID	VARCHAR2(30)
		+	PW	VARCHAR2(64)
		+	NAME	VARCHAR2(20)
		+	EMAIL	VARCHAR2(45)
		+	REGDATE	DATE
		✓	MANAGER	NUMBER(1)
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	REVIEW_COMMENT_NO	NUMBER
		✓	REVIEW_NO	NUMBER
		✓	MEMBER_ID	VARCHAR2(30)
		+	CONTEXT	VARCHAR2(2000)
		+	STATE	NUMBER(1)

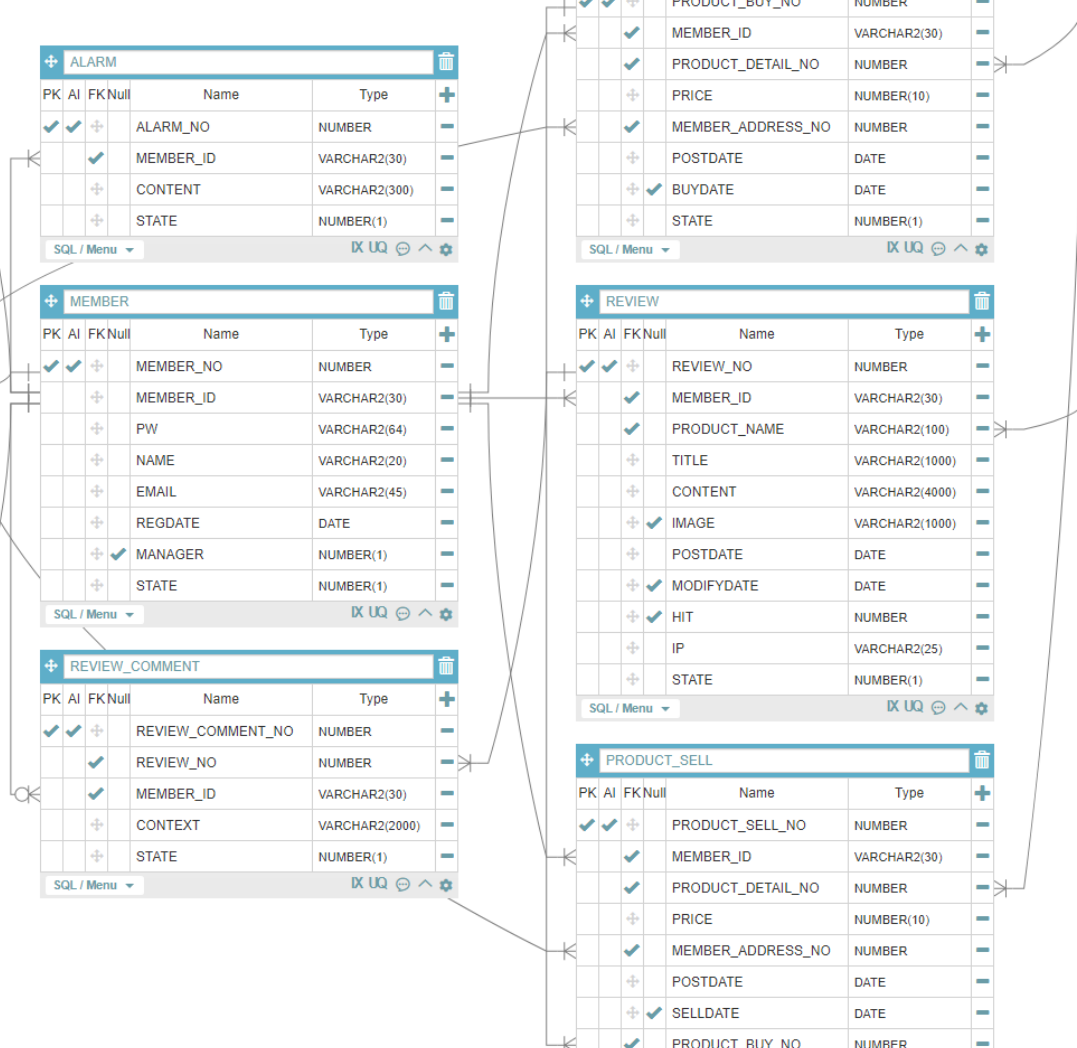
PK	AI	FKNull	Name	Type
✓	✓	+	PRODUCT_BUY_NO	NUMBER
		✓	MEMBER_ID	VARCHAR2(30)
		✓	PRODUCT_DETAIL_NO	NUMBER
		+	PRICE	NUMBER(10)
		✓	MEMBER_ADDRESS_NO	NUMBER
		+	POSTDATE	DATE
		✓	BUYDATE	DATE
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	REVIEW_NO	NUMBER
		✓	MEMBER_ID	VARCHAR2(30)
		✓	PRODUCT_NAME	VARCHAR2(100)
		+	TITLE	VARCHAR2(1000)
		+	CONTENT	VARCHAR2(4000)
		✓	IMAGE	VARCHAR2(1000)
		+	POSTDATE	DATE
		✓	MODIFYDATE	DATE
		✓	HIT	NUMBER
		+	IP	VARCHAR2(25)
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	PRODUCT_SELL_NO	NUMBER
		✓	MEMBER_ID	VARCHAR2(30)
		✓	PRODUCT_DETAIL_NO	NUMBER
		+	PRICE	NUMBER(10)
		✓	MEMBER_ADDRESS_NO	NUMBER
		+	POSTDATE	DATE
		✓	SELDATE	DATE
		✓	PRODUCT_BUY_NO	NUMBER
		+	STATE	NUMBER(1)

PK	AI	FKNull	Name	Type
✓	✓	+	PRODUCT_DETAIL_NO	NUMBER
		✓	PRODUCT_NAME	VARCHAR2(100)
		+	PRODUCT_SIZE	NUMBER(3)

PK	AI	FKNull	Name	Type
✓	✓	+	PRODUCT_NO	VARCHAR2(30)
		+	PRODUCT_NAME	VARCHAR2(100)
		✓	BRAND	VARCHAR2(30)
		+	PRICE	NUMBER(10)
		+	IMAGE	VARCHAR2(1000)
		+	STATE	NUMBER(1)



DB TABLE 구조 (회원)

회원 테이블

PK	AI	FK	Null	Name	Type	
✓	✓	+		MEMBER_NO	NUMBER	-
		+		MEMBER_ID	VARCHAR2(30)	-
		+		PW	VARCHAR2(64)	-
		+		NAME	VARCHAR2(20)	-
		+		EMAIL	VARCHAR2(45)	-
		+		REGDATE	DATE	-
		+	✓	MANAGER	NUMBER(1)	-
		+		STATE	NUMBER(1)	-

SQL / Menu IX UQ ☹ ^ ⚙

회원 주소 테이블

PK	AI	FK	Null	Name	Type	
✓	✓	+		MEMBER_ADDRESS_NO	NUMBER	-
		✓		MEMBER_NO	NUMBER	-
		+		NAME	VARCHAR2(100)	-
		+		ADDR1	VARCHAR2(100)	-
		+	✓	ADDR2	VARCHAR2(100)	-
		+		STATE	NUMBER(1)	-

SQL / Menu IX UQ ☹ ^ ⚙

회원이 여러 개의 주소를 가질 수 있도록 하기 위해
회원 주소 테이블로 분리하였습니다.

DB TABLE 구조 (상품)

상품 테이블

PRODUCT					
PK	AI	FK	Null	Name	Type
✓				PRODUCT_NO	VARCHAR2(30)
				PRODUCT_NAME	VARCHAR2(100)
			✓	BRAND	VARCHAR2(30)
				PRICE	NUMBER(10)
				IMAGE	VARCHAR2(1000)
				STATE	NUMBER(1)

상품 상세 테이블 (사이즈)

PRODUCT_DETAIL					
PK	AI	FK	Null	Name	Type
✓	✓			PRODUCT_DETAIL_NO	NUMBER
			✓	PRODUCT_NAME	VARCHAR2(100)
				PRODUCT_SIZE	NUMBER(3)

상품 사이즈 별로 상품 구매신청, 판매신청을 받기 위해
상품 상세 테이블을 만들었습니다.

DB TABLE 구조 (구매 신청, 판매 신청)

상품 구매신청 테이블

PRODUCT_BUY						
PK	AI	FK	Null	Name	Type	
✓	✓	+		PRODUCT_BUY_NO	NUMBER	-
		✓		MEMBER_ID	VARCHAR2(30)	-
		✓		PRODUCT_DETAIL_NO	NUMBER	-
		+		PRICE	NUMBER(10)	-
		✓		MEMBER_ADDRESS_NO	NUMBER	-
		+		POSTDATE	DATE	-
		+	✓	BUYDATE	DATE	-
		+		STATE	NUMBER(1)	-

상품 판매신청 테이블

PRODUCT_SELL						
PK	AI	FK	Null	Name	Type	
✓	✓	+		PRODUCT_SELL_NO	NUMBER	-
		✓		MEMBER_ID	VARCHAR2(30)	-
		✓		PRODUCT_DETAIL_NO	NUMBER	-
		+		PRICE	NUMBER(10)	-
		✓		MEMBER_ADDRESS_NO	NUMBER	-
		+		POSTDATE	DATE	-
		+	✓	SELDATE	DATE	-
		✓		PRODUCT_BUY_NO	NUMBER	-
		+		STATE	NUMBER(1)	-

회원이 주문신청을 넣을 경우 등록되는 테이블입니다.
 여러 테이블의 기본키를 참고하여 신청서에 필요한 정보들을 담을 수 있게 하였습니다.
 신청서의 진행도에 따라 STATE 값에 변화를 주었습니다.

DB TABLE 구조 (FAQ, NOTICE)

자주 묻는 질문 게시판

FAQ						
PK	AI	FK	Null	Name	Type	+
✓	✓	+		FAQ_NO	NUMBER	-
		✓		MEMBER_ID	VARCHAR2(30)	-
		+		QUESTION	VARCHAR2(1000)	-
		+		ANSWER	VARCHAR2(4000)	-
		+		POSTDATE	DATE	-
		+		STATE	NUMBER(1)	-
SQL / Menu						IX UQ ☰ ^ ⚙

공지사항 게시판

NOTICE						
PK	AI	FK	Null	Name	Type	+
✓	✓	+		NOTICE_NO	NUMBER	-
		✓		MEMBER_ID	VARCHAR2(30)	-
		+		TITLE	VARCHAR2(1000)	-
		+		CONTENT	VARCHAR2(4000)	-
		+		POSTDATE	DATE	-
		+		STATE	NUMBER(1)	-
SQL / Menu						IX UQ ☰ ^ ⚙

관리자만 작성할 수 있는 자주 묻는 질문 게시판과 공지사항 게시판 입니다.
페이징 기능만을 생각하고 테이블을 구현하였습니다.

DB TABLE 구조(후기)

후기 게시판

REVIEW						
PK	AI	FK	Null	Name	Type	
✓	✓	+		REVIEW_NO	NUMBER	-
		✓		MEMBER_ID	VARCHAR2(30)	-
		✓		PRODUCT_NAME	VARCHAR2(100)	-
		+		TITLE	VARCHAR2(1000)	-
		+		CONTENT	VARCHAR2(4000)	-
		+	✓	IMAGE	VARCHAR2(1000)	-
		+		POSTDATE	DATE	-
		+	✓	MODIFYDATE	DATE	-
		+	✓	HIT	NUMBER	-
		+		IP	VARCHAR2(25)	-
		+		STATE	NUMBER(1)	-

후기 댓글

REVIEW_COMMENT						
PK	AI	FK	Null	Name	Type	
✓	✓	+		REVIEW_COMMENT_NO	NUMBER	-
		✓		REVIEW_NO	NUMBER	-
		✓		MEMBER_ID	VARCHAR2(30)	-
		+		CONTEXT	VARCHAR2(2000)	-
		+		STATE	NUMBER(1)	-

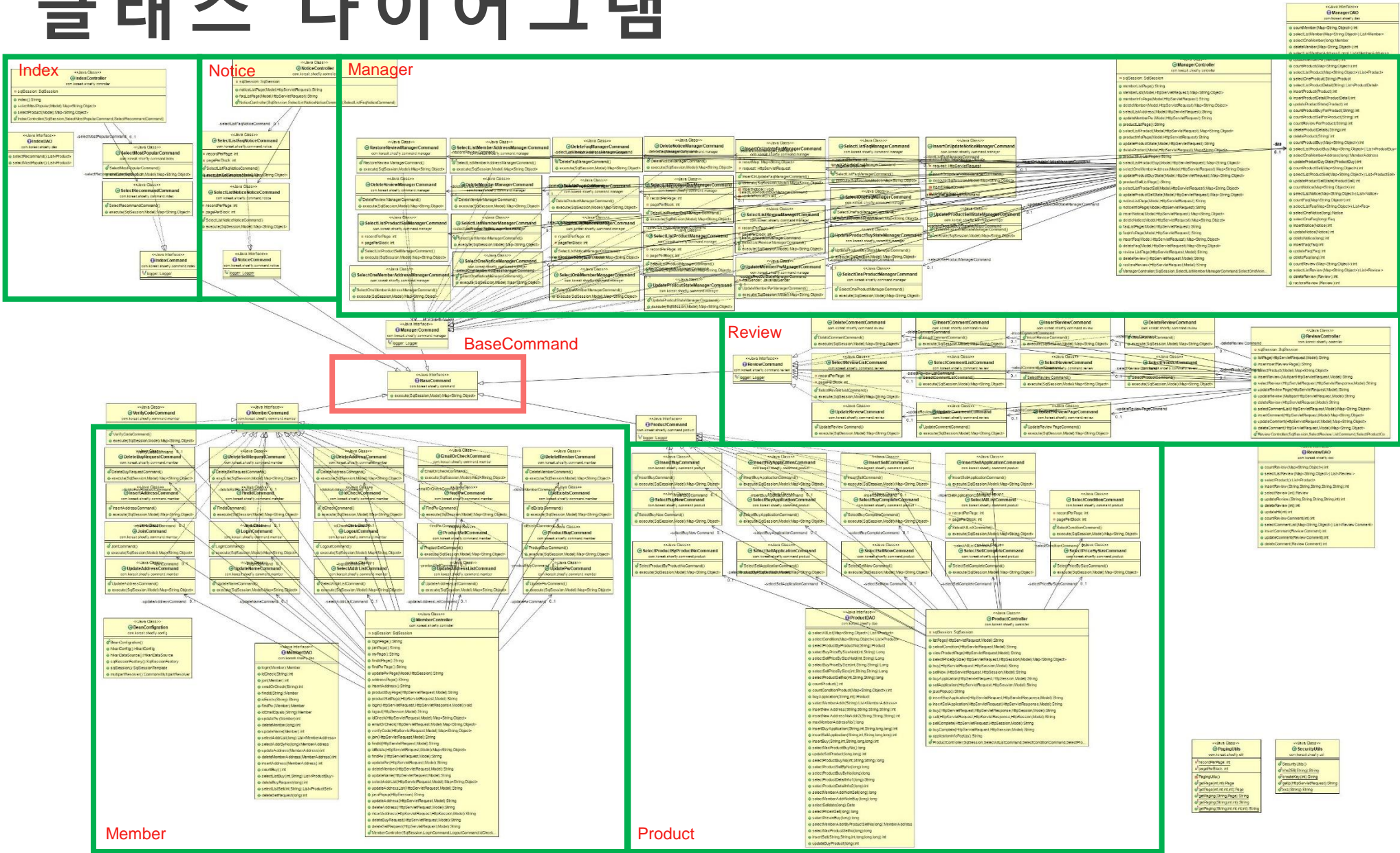
후기 게시판의 댓글은 대댓글이 없다고 상정하고 테이블을 구현하였습니다.




03

프로젝트 구조
당담파트 구현

클래스 다이어그램



메인 페이지 (INDEX)




PRODUCT

REVIEW


NOT

Most Popular


인기 상품



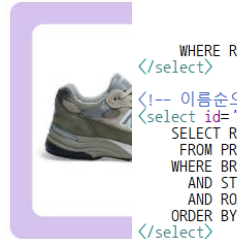
Jordan 1 Retro High OG Electro Orange



New Balance 530 White




New Balance x Aime Leon Dore 550 Navy Red




New Balance x WTAPS 992 Made in USA Olive

Recomanded Product


추천 상품




Nike Dunk Low SP Plum



Nike Dunk Low SP Veneer



Nike x Kasina Dunk Low Road Sign (kasina Exclusive)



Nike x Peaceminusone Air Force 1 Low Para-Noise

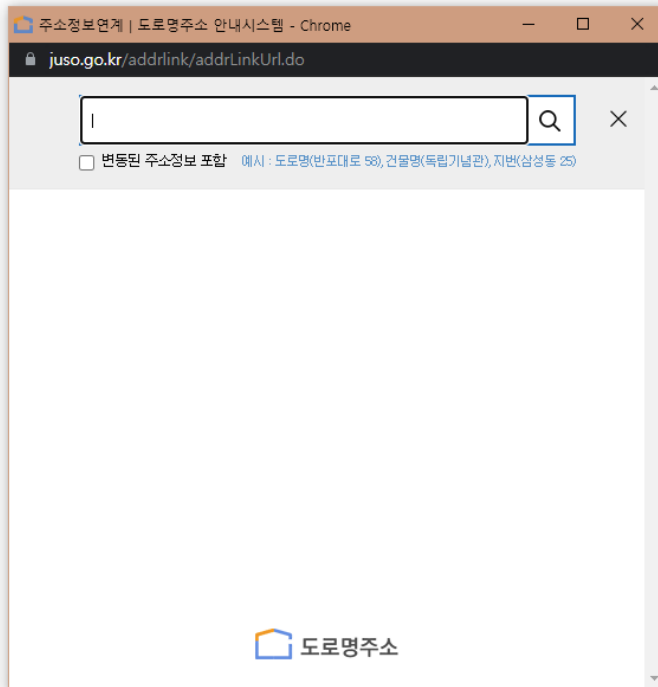
<!-- 거래량 순으로 4개 -->
<select id="selectMostPopular" resultType="com.koreait.shoefly.dto.Product">
SELECT ROWNUM RN, PRODUCT_NO AS PRODUCTNO, PRODUCT_NAME AS PRODUCTNAME, PRICE, BRAND, IMAGE, STATE, 거래량
FROM (SELECT P1.PRODUCT_NO, P1.PRODUCT_NAME, P1.PRICE, P1.BRAND, P1.IMAGE, P1.STATE, COUNT(C1.PRODUCT_DETAIL_NO) + COUNT(C2.PRODUCT_DETAIL_NO) AS 거래량
FROM PRODUCT_BUY C1
FULL OUTER JOIN PRODUCT_SELL C2
ON 1 = 2
LEFT OUTER JOIN PRODUCT_DETAIL P2
ON C1.PRODUCT_DETAIL_NO = P2.PRODUCT_DETAIL_NO
OR C2.PRODUCT_DETAIL_NO = P2.PRODUCT_DETAIL_NO
RIGHT OUTER JOIN PRODUCT P1
ON P1.PRODUCT_NAME = P2.PRODUCT_NAME
WHERE P1.STATE >= 0
GROUP BY P1.PRODUCT_NO, P1.PRODUCT_NAME, P1.PRICE, P1.BRAND, P1.IMAGE, P1.STATE
ORDER BY 거래량 DESC)
WHERE ROWNUM <= 5
</select>

<!-- 이름순으로 브랜드 4개 -->
<select id="selectRecomand" resultType="com.koreait.shoefly.dto.Product">
SELECT ROWNUM RN, PRODUCT_NO AS PRODUCTNO, PRODUCT_NAME AS PRODUCTNAME, BRAND, PRICE, IMAGE
FROM PRODUCT
WHERE BRAND = 'Nike' <!-- 현재 브랜드 Nike, Jordan, Adidas, New Balance -->
AND STATE = 0
AND ROWNUM <= 5
ORDER BY PRODUCT_NAME
</select>

메인 페이지의 상품 정렬 SQL문

- 인기상품 (가장 거래량이 많은)

회원 관리 (주소 API)



신규 주소등록

배송지

배송지를 입력해주세요.

주소

주소찾기

상세주소

추가

```

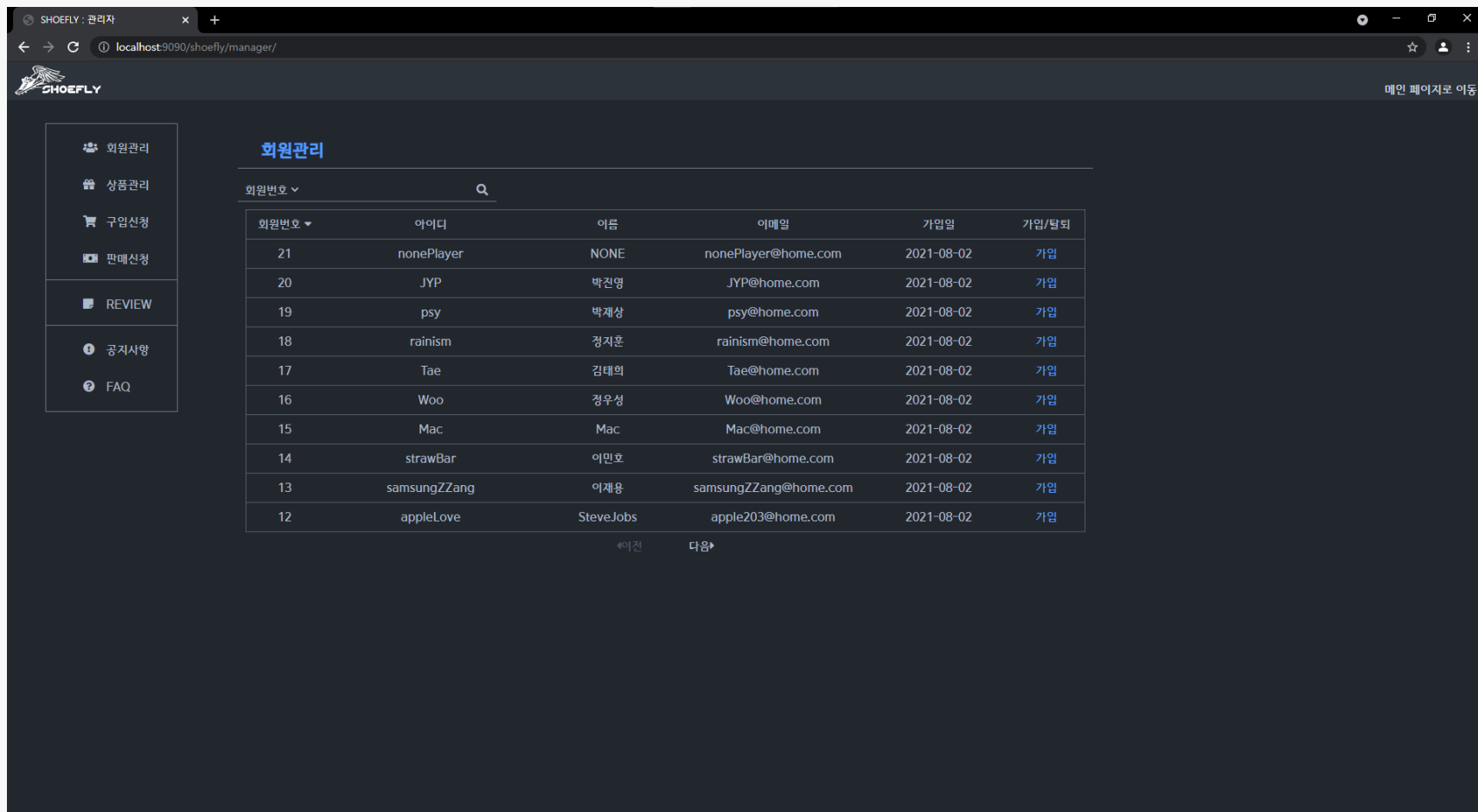
37 String emdNo = request.getParameter("emdNo");
38
39
40 </head>
41 <script language="javascript">
42 // opener관련 오류가 발생하는 경우 아래 주석을 해지하고, 사용자의 도메인정보를 입력합니다. ("주소입력화면 소스"도 동일하게 적용시켜야 합니다.)
43 //document.domain = "abc.go.kr";
44
45 /*
46  모의 해킹 테스트 시 팝업API를 호출하시면 jp가 차단 될 수 있습니다.
47  주소팝업API를 제외하시고 테스트 하시기 바랍니다.
48 */
49 function init(){
50   var url = location.href;
51   var confmKey = "U01TX0FVVEgyMDIxMTExMTUzMTEwMTQxNTI=";
52   var resultType = "3"; // 도로명주소 검색결과 화면 출력내용, 1 : 도로명, 2 : 도로명+지번+상세보기(관련지번, 관할주민센터), 3 : 도로명+상세보기(상세건물명), 4 : 도로명+지번+상세보기(관련
53   var inputYn = "<inputYn>";
54   if(inputYn != "Y"){
55     document.form.confmKey.value = confmKey;
56     document.form.returnUrl.value = url;
57     document.form.resultType.value = resultType;
58     document.form.action = "https://www.juso.go.kr/addrlink/addrLinkUrl.do"; //인터넷망
59     //document.form.action="https://www.juso.go.kr/addrlink/addrMobileLinkUrl.do"; //모바일 웹인 경우, 인터넷망
60     document.form.submit();
61   } else {
62     opener.jsoCallBack("<%=roadFullAddr%>","<%=roadAddrPart1%>","<%=addrDetail%>","<%=roadAddrPart2%>","<%=engAddr%>","<%=jibunAddr%>","<%=zipNo%>","<%=admCd%>","<%=rnMgtSn%>","<
63     window.close();
64   }
65 }
66 </script>
67 <body onload="init();">
68
69 <form id="form" name="form" method="post">
70   <input type="hidden" id="confmKey" name="confmKey" value=""/>
71   <input type="hidden" id="returnUrl" name="returnUrl" value=""/>
72   <input type="hidden" id="resultType" name="resultType" value=""/>
73   <!-- 해당시스템의 인코딩타입이 EUC-KR일경우에만 추가 START-->
74   <!--
75   <input type="hidden" id="encodingType" name="encodingType" value="EUC-KR"/>
76   -->
77   <!-- 해당시스템의 인코딩타입이 EUC-KR일경우에만 추가 END-->
78 </form>
79 </body>
80 </html>

```

주소API 구현

주소API를 통해 검색 결과를 주소와 상세주소에 표시하고 DB에 추가 할 수 있게 한다.

관리자 페이지



관리자 페이지 이동

- 관리자로 등록된 계정에 로그인
- 마이페이지 이동 시 관리자 페이지로 이동

관리자 페이지 요구사항

- 회원관리
- 상품관리
- 구입신청
- 판매신청
- 후기 게시판
- 공지사항
- 자주묻는 질문

관리자 페이지 (전체 테이블)

회원관리			
회원번호 ▾	<input type="text"/>		
회원번호 ▾	아이디	이름	이메일
21	nonePlayer	NONE	nonePlayer@home.com
20	JYP	박진영	JYP@home.com
19	psy	박재상	psy@home.com
18	rainism	청지훈	rainism@home.com
17	Tae	김태희	Tae@home.com
16	Woo	청우성	Woo@home.com
15	Mac	Mac	Mac@home.com
14	strawBar	이민호	strawBar@home.com
13	samsungZZang	이재용	samsungZZang@home.com
12	appleLove	SteveJobs	apple203@home.com
<div> ◀이전 다음▶ </div>			

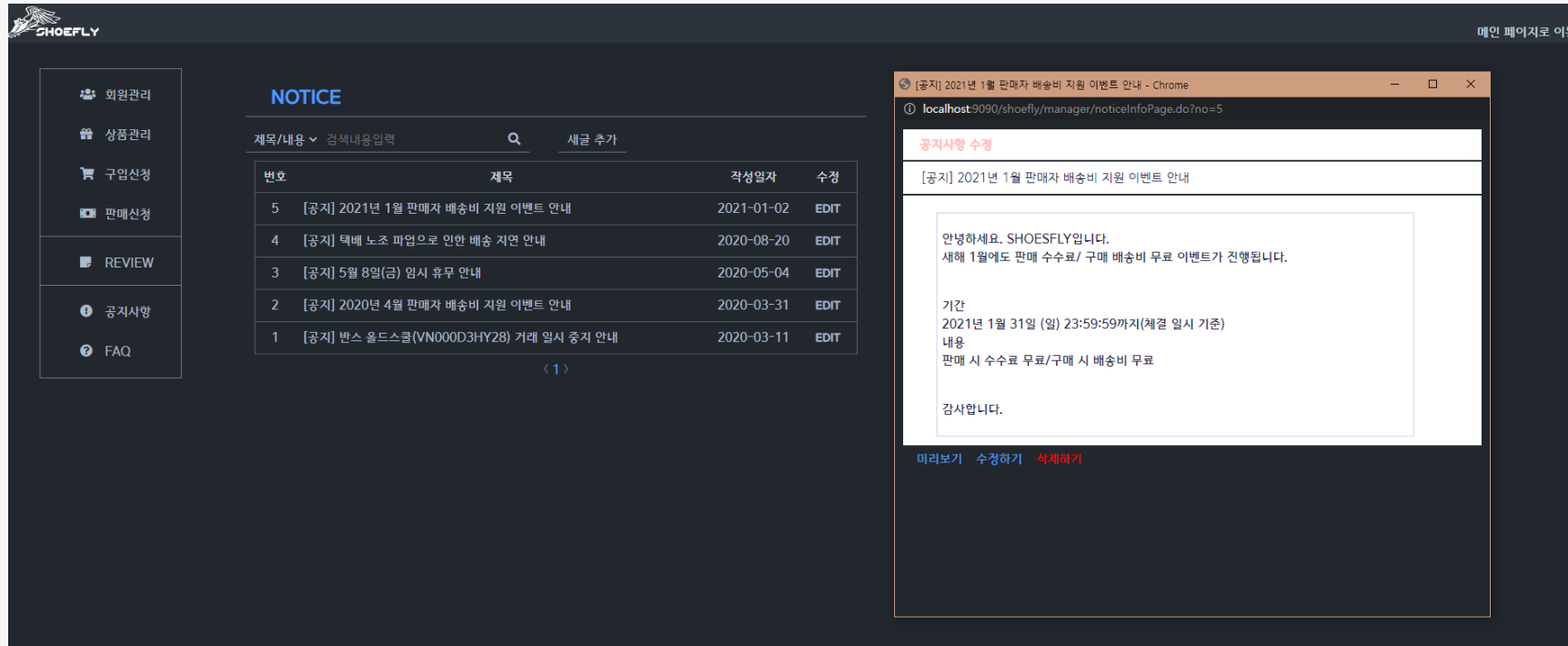
```
// MEMBER LIST LOAD
$.ajax({
  url: 'memberList.do',
  type: 'POST',
  data: 'order=' + order + '&isDesc=' + isDesc + '&column=' + $('#column').val() + '&query=' + $('#query').val()
    + '&page=' + page + '&startDate=' + $('#startDate').val() + '&endDate=' + $('#endDate').val(),
  dataType: 'json',
  success: function(data) {
    var member_tbody = $('#member_tbody');
    member_tbody.empty();
    if(data.result){
      $.each(data.list, function(index, member) {
        var tr = $('#tr').appendTo(member_tbody).attr('onclick', 'show_info(' + member.memberNo + ')');
        $('#td').text(member.memberNo).appendTo(tr);
        $('#td').text(member.memberId).appendTo(tr);
        $('#td').text(member.name).appendTo(tr);
        $('#td').text(member.email).appendTo(tr);
        $('#td').text(member.regdate).appendTo(tr);
        if(member.state == 0)
          $('#td style="color:#539bf5;"').text('가입').appendTo(tr);
        else
          $('#td style="color:red;"').text('탈퇴').appendTo(tr);
      })
    } else {
      $('#tr').append($('  | | | | | |
```

관리자 페이지 테이블 구성

- 관리 페이지에 따라 적절한 검색 구현
- 칼럼명을 선택함으로써 테이블 정렬

(AJAX) 테이블 로드

관리자 페이지 (공지사항, FAQ)



공지사항, FAQ 게시판 구현

- 새로운 글을 추가, 수정 할 때
미리보기를 통해 작성하는 글을 확인 가능

관리자 페이지 (공지사항, FAQ)

SHOEFLY 메인 페이지로 이동

회원관리

상품관리

구입신청

판매신청

REVIEW

공지사항

FAQ

구입신청

아이디 ▼

신청목록

주문번호	아이디	상품명	사이즈	가격	주소	구입신청일	거래확정일
9	user3	Jordan 1 Retro High OG Electro Orange	280	196000	확현	2021-03-17 03:00	취소
10	user4	Jordan 1 Retro High OG Electro Orange	270	196000	확현	2021-04-03 03:00	취소
11	user1	Adidas Yeezy Boost 350 V2 Zebra	240	235000	확현	2021-04-14 03:00	취소
12	user2	Adidas Yeezy Boost 350 V2 Zebra	250	250000	확현	2021-04-15 03:00	취소
13	user3	Adidas Yeezy Boost 350 V2 Zebra	260	198000	확현	2021-04-30 03:00	취소
14	user3	Adidas Yeezy Boost 350 V2 Zebra	270	196000	확현	2021-05-07 03:00	취소
15	user4	Adidas Yeezy Boost 350 V2 Zebra	280	196000	확현	2021-05-14 03:00	취소
16	user1	Adidas Yeezy Boost 350 V2 Zebra	290	200000	확현	2021-05-24 03:00	취소

◀이전 다음▶

구매확정

주문번호	아이디	상품명	사이즈	가격	주소	구입신청일	거래확정일
29	user4	Jordan 1 x Travis Scott Retro High OG SP Mocha	250	210000	확현	2020-12-31 03:00	2021-04-15 03:00

◀이전 다음▶

배송중

```
<update id="updateProductBuyState" parameterType="com.koreait.shoefly.dto.ProductBuy">
    UPDATE PRODUCT_BUY
    SET STATE = #{state}
    WHERE PRODUCT_BUY_NO = #{productBuyNo}
</update>
```

```
<update id="updateProductSellState" parameterType="com.koreait.shoefly.dto.ProductSell">
    UPDATE PRODUCT_SELL
    SET STATE = #{state}
    WHERE PRODUCT_SELL_NO = #{productSellNo}
</update>
```

```
> UpdateProductBuyStateManagerCommand.java
> UpdateProductSellStateManagerCommand.java
```

구입신청, 판매신청 관리 구현

- 구입신청, 판매신청 TABLE의 STATUS 수정 기능을 구현
- 유지보수를 위해 Java Class와 Sql문을 간소화하여 작성

Thank You

