



Jasmine

오승용 최영권

목차

1. 팀원소개

2. 기획의도

3. 개발 플랫폼 및 사용 기술

4. 프로젝트소개

5. ERD 및 UML

6. 주요소스코드

7.후기

팀원소개

팀원소개



『1』 팀원소개



- DB설계
- DB데이터 관리
- 게시판
(페이징,답글,이미지업로드포함)
- 장바구니 기능
- 상품 검색 기능
- 주문기능
- 메인페이지

오승용



- DB설계
- DB데이터 관리
- 카카오 소셜 로그인 API 연동
- 로그인 페이지
- 배송추적 API
- 결제 API
- 주소API연동
- 회원가입
- 마이페이지 회원정보 수정
- 마이페이지 주문관리
- 마이페이지 회원관리
- 마이페이지 주문내역
- 상품 상세 페이지
- 타일즈 프레임워크 적용
- AWS 인프라 구축

최영권

기획의도

기획의도



『2』
기획의도



책을 사랑 하는 사람들 위한 서비스

Jasmine

개발 플랫폼 및 사용 기술

사용 기술

사용 기술 및 개발 플랫폼



사용 기술 및 개발 플랫폼

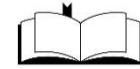
OS/WAS	Window 10 64-bit/Ubuntu 18.04 64-bit/ Tomcat 8.0.45
DATABASE	Oracle 11g
IDE 및 기타 TOOL	Eclipse / Visual Studio Code / GIT / SQL Developer / StarUML / Notion / Slack / Oven / Google Office / AWS EC2
Programming Language	JAVA / Javascript / JSP / Servlet / JQuery / HTML 5 / CSS3
Framework	Spring / Mybatis / Bootstrap / Tiles / Maven
Application Programming Interface	구글 차트 / Kakao Login API / iamPort 결제API / 스마트 택배 배송조회 API

프로젝트 소개

— 1 — 1, 2



『4』 프로젝트 소개



피드

신간, 베스트 셀러, 추천상품 등
고객에게 필요한 도서 정보 제공
및 검색 기능 서비스



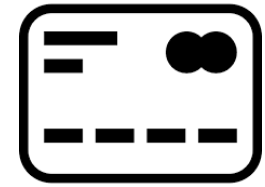
장바구니

상품 등을 담아 놓을 수
있는 가상의 저장 공간
서비스



사용자 인증

사용자 중심의 인터페이
스 편의성 및 소셜 로그인
제공 서비스



결제

상품에 대한 신용카드, 실시간
계좌이체, 가상계좌 등 온라인
결제 기능을 제공하는 서비스



『4』 프로젝트 소개



문의

고객의 불편함을 해소 할 수 있는 문의 게시판 서비스



데이터 시각화

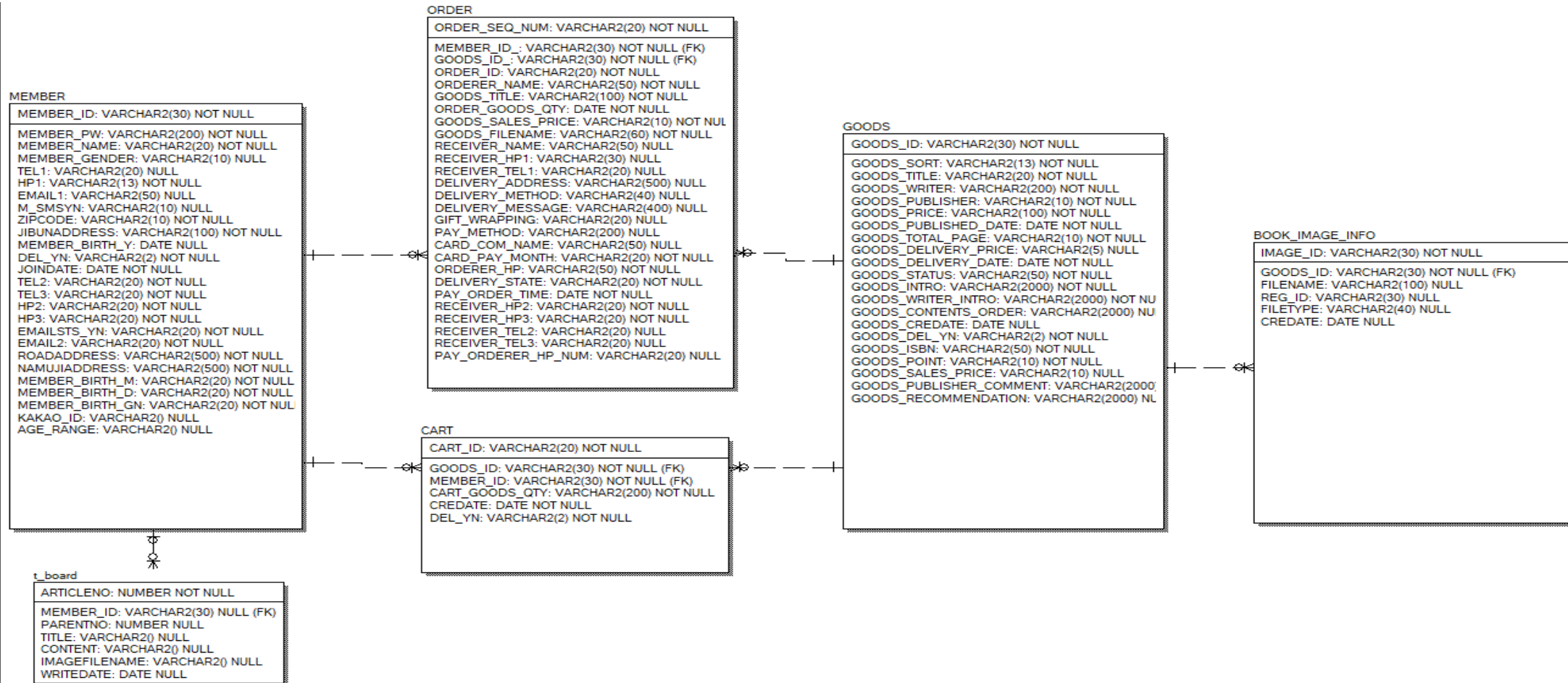
고객의 데이터를 분석하여 니즈를 파악할 수 있는 기능



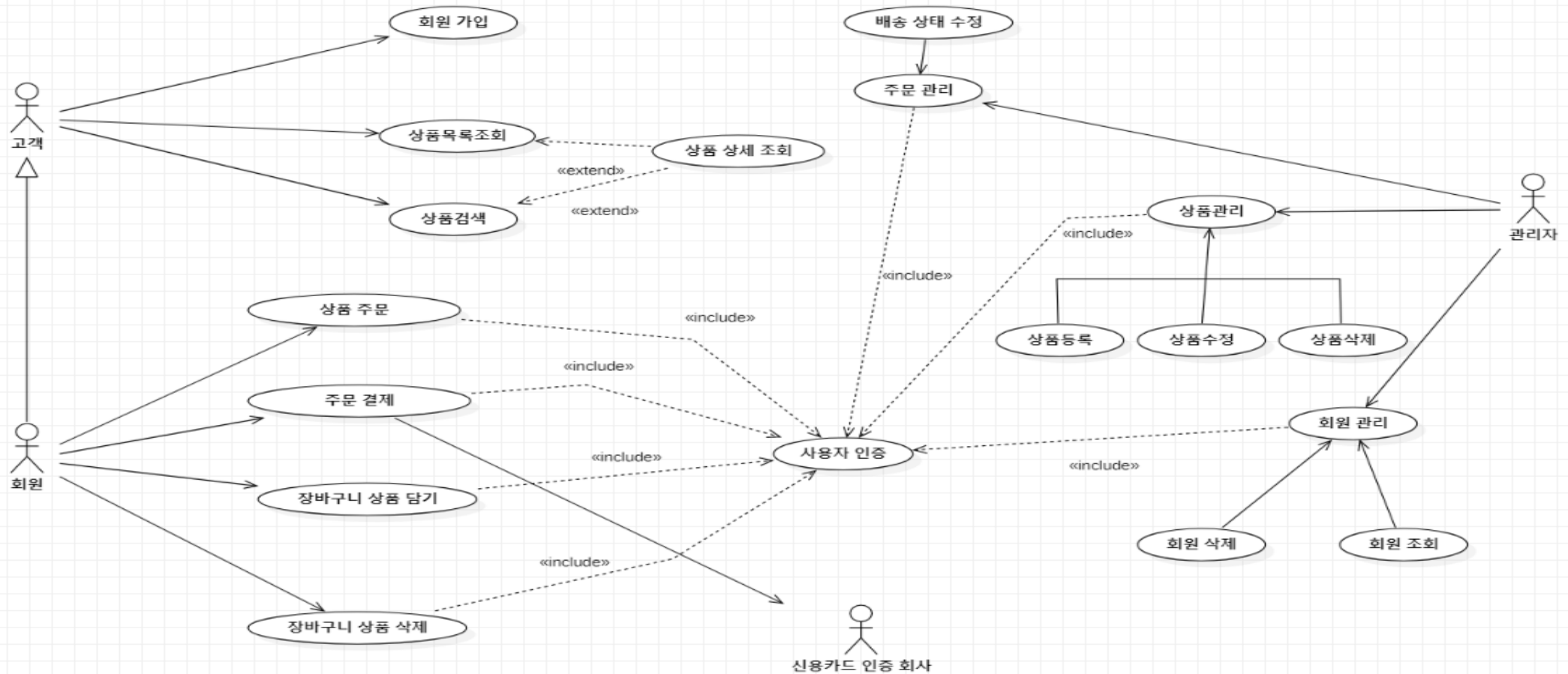
관리자 페이지

고객이 구매한 상품을 조회하고 새로운 상품을 등록할 수 있는 기능

ERD 및 UML



ERD 및 UML



주요 소스 코드

1. 2. 3. 4. 5.

소스코드

프로젝트 URL : <http://jasmine62.shop>

소스코드 : <https://github.com/osy0907/JasmineProject>

메인페이지

```
package js.bs.main;

import java.util.List;

@Controller("mainController")
@EnableAspectJAutoProxy
public class MainController extends BaseController {
    @Autowired
    private GoodsService goodsService;

    @RequestMapping(value="/main/main.do", method= {RequestMethod.POST, RequestMethod.GET})
    public ModelAndView main(HttpServletRequest request, HttpServletResponse response) throws ServletException, IOException {
        HttpSession session;

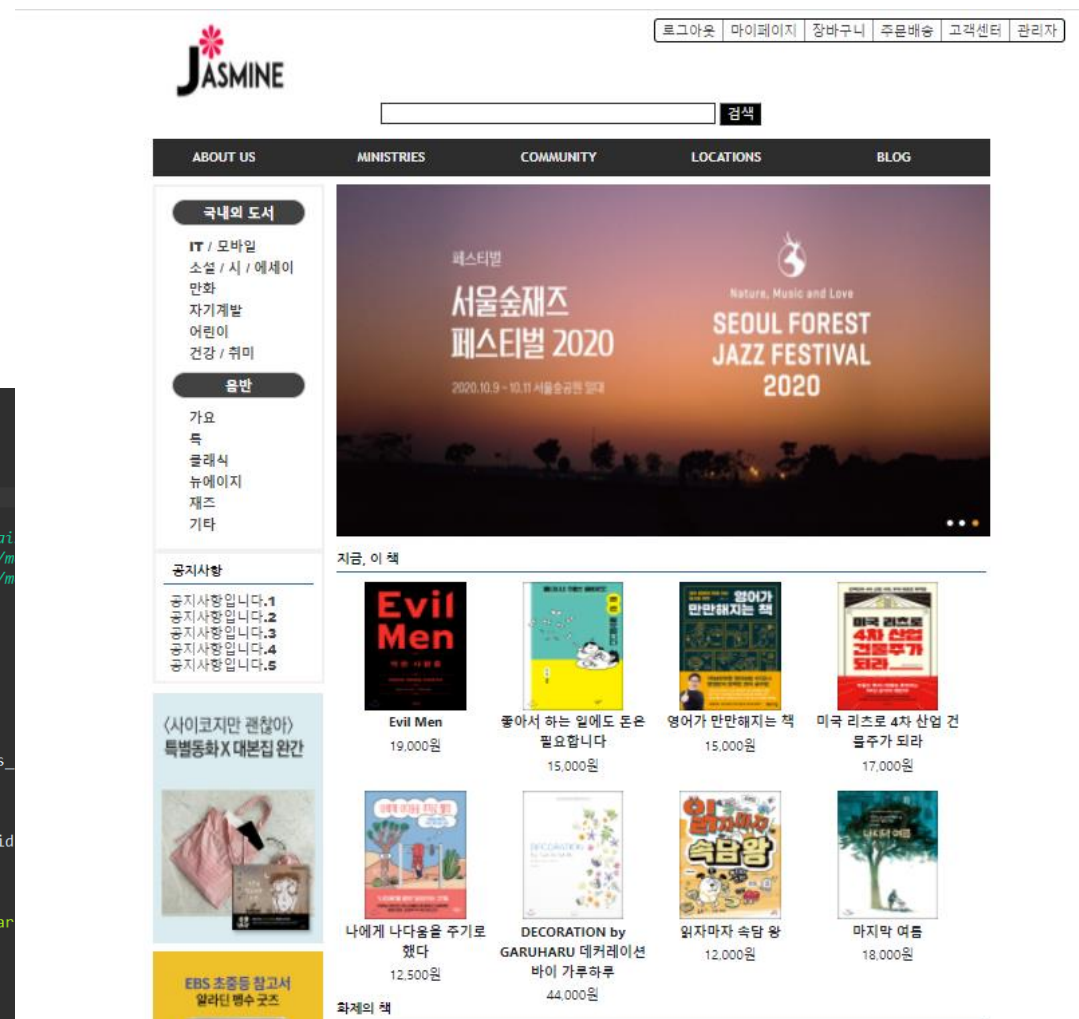
        ModelAndView mav = new ModelAndView();
        String viewName=(String)request.getAttribute("viewName");
        mav.setViewName(viewName);

        session=request.getSession();
        session.setAttribute("side_menu", "user");

        Map<String, List<GoodsVO>> goodsMap = goodsService.listGoods();
        mav.addObject("goodsMap", goodsMap);

        return mav;
    }
}
```

```
7 <c:set var="contextPath" value="${pageContext.request.contextPath}" />
8 <%
9     request.setCharacterEncoding("UTF-8");
10 %>
12 <div id="ad_main_banner">
13     <ul class="bjqs">
14         <li></li>
15         <li></li>
16         <li></li>
17     </ul>
18 </div>
19 <div class="main_book">
20     <c:set var="goods_count" value="0" />
21     <h3>지금, 이 책</h3>
22     <c:forEach var="item" items="${goodsMap.bestseller }">
23         <c:set var="goods_count" value="${goods_count+1}" />
24         <div class="book">
25             <a href="${contextPath}/goods/goodsDetail.do?goods_id=${item.goods_id}">
26                 
27                 
28                 <div class="title">${item.goods_title}</div>
29                 <div class="price">
30                     <fmt:formatNumber value="${item.goods_price}" type="number" var="price">
31                         ${price}원
32                     </fmt:formatNumber>
33                 </div>
34             </div>
35         </div>
36     </c:forEach>
37     <c:if test="${goods_count==25 }">
38         <div class="book">
39             <font size=20> <a href="#">more</a></font>
40         </div>
41     </c:if>
42 </c:forEach>
```



소스코드

프로젝트 URL : <http://jasmine62.shop>

소스코드 : <https://github.com/osy0907/JasmineProject>

장바구니


```
@RequestMapping(value="/myCartList.do",method = RequestMethod.GET)
public ModelAndView myCartMain(HttpServletRequest request, HttpServletResponse response) throws Exception {
    String viewName=(String)request.getAttribute("viewName");
    ModelAndView mav = new ModelAndView(viewName);
    HttpSession session=request.getSession();
    MemberVO memberVO=(MemberVO)session.getAttribute("memberInfo");
    String member_id=memberVO.getMember_id();
    cartVO.setMember_id(member_id);
    Map<String ,List> cartMap=cartService.myCartList(cartVO);
    session.setAttribute("cartMap", cartMap); //장바구니 목록 화면에서 상품 주문 시 사용하기 위해서 장바구니 목록을 세션에 저장한다.
    //mav.addObject("cartMap", cartMap);
    return mav;
}

@RequestMapping(value="/addGoodsInCart.do",method = RequestMethod.POST,produces = "application/text; charset=utf8")
public @ResponseBody String addGoodsInCart(@RequestParam("goods_id") int goods_id, // --> 전송된 상품 번호를 받는다.
    HttpServletRequest request, HttpServletResponse response) throws Exception{
    HttpSession session=request.getSession();
    memberVO=(MemberVO)session.getAttribute("memberInfo");
    String member_id=memberVO.getMember_id();

    cartVO.setMember_id(member_id);
    //카트 등록전에 이미 등록된 제품인지 판별한다.
    cartVO.setGoods_id(goods_id);
    cartVO.setMember_id(member_id);
    boolean isAlreadyExisted=cartService.findCartGoods(cartVO);
    System.out.println("isAlreadyExisted:"+isAlreadyExisted);
    if(isAlreadyExisted==true){
        return "already_existed";
    }else{
        cartService.addGoodsInCart(cartVO);
        return "add_success";
    }
}
```

```
//console.log(cart_goods_qty);
$.ajax({
    type : "post",
    async : false, //false인 경우 동기식으로 처리한다
    url : "${contextPath}/cart/modifyCartQty",
    data : {
        goods_id:goods_id,
        cart_goods_qty:cart_goods_qty
    },

    success : function(data, textStatus) {
        //alert(data);
        if(data.trim()=='modify_success'){
            alert("수량을 변경했습니다!!");
        }else{
            alert("다시 시도해 주세요!!");
        }
    },
    error : function(data, textStatus) {
        alert("에러가 발생했습니다."+data);
    },
    complete : function(data, textStatus) {
        //alert("작업을완료 했습니다");
    }
}); //end ajax
```

MINISTRIES		COMMUNITY		LOCATIONS		BLOG	
구분	상품명	정가	판매가	수량	합계	주문	
<input checked="" type="checkbox"/>	 좋아서 하는 일에도 돈은 필요합니다	15000원	12,150원(10%할인)	<input type="text" value="1"/> 변경	12,150원	<div>주문하기</div> <div>나중에 주문</div> <div>리스트 넣기</div> <div>삭제</div>	
총 상품수	총 상품금액	총 배송비	총 할인 금액	최종 결제금액			
1개	12,150원	<div>+</div> 0원	<div>-</div> 0원	<div>=</div> 12,150원		<div>주문하기</div> <div>쇼핑 계속하기</div>	

소스코드

프로젝트 URL : <http://jasmine62.shop>

소스코드 : <https://github.com/osy0907/JasmineProject>

상품주문 결제 API

```
// alert("총주문 금액:"+total_order_goods_price);
order_goods_qty=h_order_goods_qty.value;
//order_total_price=h_order_total_price.value;

orderer_name=h_orderer_name.value;
receiver_name=i_receiver_name.value;
hp1=i_hp1.value;
hp2=i_hp2.value;
hp3=i_hp3.value;

tel1=i_tel1.value;
tel2=i_tel2.value;
tel3=i_tel3.value;

receiver_hp_num=hp1+"-"+hp2+"-"+hp3;
receiver_tel_num=tel1+"-"+tel2+"-"+tel3;

delivery_address="우편번호:"+i_zipcode.value+"<br>"+
"도로명 주소:"+i_roadAddress.value+"<br>"+
"[지번 주소:"+i_jibunAddress.value+"]<br>"+
"i_namujiAddress.value;
delivery_addr = i_roadAddress.value + i_jibunAddress.value;
delivery_message=i_delivery_message.value;

var r_pay_method = frm.pay_method;

for(var i=0; i<r_pay_method.length;i++){
    if(r_pay_method[i].checked==true){
        pay_method=r_pay_method[i].value
        if(pay_method=="신용카드"){
            var i_card_com_name=document.getElementById("card_com_name").value;
            var i_card_pay_month=document.getElementById("card_pay_month").value;
            card_com_name=i_card_com_name.value;
            card_pay_month=i_card_pay_month.value;
            pay_method+="<br>"+
            "카드사:"+card_com_name+"<br>"+
            "할부개월수:"+card_pay_month;
            pay_orderer_hp_num="해당없음";

        }else if(pay_method=="휴대폰결제"){
            var i_pay_order_tel1=document.getElementById("pay_order_tel1").value;
            var i_pay_order_tel2=document.getElementById("pay_order_tel2").value;
            var i_pay_order_tel3=document.getElementById("pay_order_tel3").value;
            pay_orderer_hp_num=i_pay_order_tel1.value+"-"+
            i_pay_order_tel2.value+"-"+
            i_pay_order_tel3.value;
            pay_method+="<br>"+"결제휴대폰번호:"+pay_orderer_hp_num;
            card_com_name="해당없음";
            card_pay_month="해당없음";
        } //end if
        break;
    } // end for

    var IMP = window.IMP; // 생략해도 괜찮습니다.
    IMP.init("imp15084481");

    IMP.request_pay({ // param
        pg: "html5_inicis",
        pay_method: "card",
        merchant_uid: "Jasmine_" + new Date().getTime(),
        name: goods_title,
        amount: 100,
        buyer_email: "gildong@gmail.com",
        buyer_name: receiver_name,
        buyer_tel: receiver_hp_num,
        buyer_addr: delivery_addr,
        buyer_postcode: "01181"
    }, function (rsp) { // callback
        if (rsp.success) {
            alert("결제 성공");
        } else {
            alert("결제 실패");
        }
    })
}
```

신용카드

KG 이니시스

이용약관

전자금융거래 이용약관

약관보기

현대카드

삼성카드

KB국민

비씨(페이북)

신한카드

하나카드

롯데카드

NH농협

씨티카드

UnionPay

그외카드

pay

CHAI

N Pay

L.pay

PAYCO

그외결제

하나카드 2~6개월 무이자할부

TEST결제창으로 익일 자동취소처리됩니다

KG 이니시스

상품명

총아서 하는 일에도 돈은 필요...

상품가격

100 원

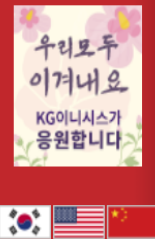
제공기간

별도제공기간없음

결제금액

100 원

다 음



소스코드

프로젝트 URL : <http://jasmine62.shop>

소스코드 : <https://github.com/osy0907/JasmineProject>

고객센터 - 문의게시판

```
public ResponseEntity<Map<String, Object>> addReply(MultipartHttpServletRequest multipartRequest,
HttpServletResponse response) throws Exception {
    multipartRequest.setCharacterEncoding("utf-8");
    Map<String, Object> articleMap = new HashMap<String, Object>();
    Enumeration<String> enu = multipartRequest.getParameterNames();
    while(enu.hasMoreElements()){
        String name = (String)enu.nextElement();
        String value = multipartRequest.getParameter(name);
        articleMap.put(name, value);
    }

    String imageFileName = upload(multipartRequest);
    articleMap.put("imageFileName", imageFileName);

    String articleNO = (String)articleMap.get("articleNO");
    String message;
    ResponseEntity resEnt = null;
    HttpHeaders responseHeaders = new HttpHeaders();
    responseHeaders.add("Content-Type", "text/html; charset=utf-8");
    try {
        boardService.modArticle(articleMap);
        if(imageFileName != null && imageFileName.length() != 0) {
            File srcFile = new File(ARTICLE_IMAGE_REPO + "\\temp\\" + articleNO + imageFileName);
            File destDir = new File(ARTICLE_IMAGE_REPO + "\\article\\" + articleNO);
            FileUtils.moveFileToDirectory(srcFile, destDir, true);

            String originalFileName = (String)articleMap.get("originalFileName");
            File oldFile = new File(ARTICLE_IMAGE_REPO + "\\article\\" + articleNO + originalFileName);
            oldFile.delete();
        }
        message = "<script>";
        message += " alert('글을 수정했습니다.');";
        message += " location.href='"+multipartRequest.getContextPath()+"/board/viewArticle.do?articleNO="+articleNO+"';";
        message += "</script>";
        resEnt = new ResponseEntity(message, responseHeaders, HttpStatus.CREATED);
    } catch (Exception e) {
        File srcFile = new File(ARTICLE_IMAGE_REPO + "\\temp\\" + articleNO + imageFileName);
        srcFile.delete();
        message = "<script>";
        message += " alert('오류가 발생했습니다. 다시 수정해주세요.');";
        message += " location.href='"+multipartRequest.getContextPath()+"/board/viewArticle.do?articleNO="+articleNO+"';";
        message += "</script>";
        resEnt = new ResponseEntity(message, responseHeaders, HttpStatus.CREATED);
    }
}
```

```
@Override
@RequestMapping(value="/board/addReply.do", method = RequestMethod.POST)
@ResponseBody
public ResponseEntity addReply(MultipartHttpServletRequest multipartRequest,
HttpServletResponse response) throws Exception {
    multipartRequest.setCharacterEncoding("utf-8");
    Map<String, Object> articleMap = new HashMap<String, Object>();
    Enumeration<String> enu = multipartRequest.getParameterNames();
    while(enu.hasMoreElements()){
        String name = (String)enu.nextElement();
        String value = multipartRequest.getParameter(name);
        articleMap.put(name, value);
    }

    String imageFileName = upload(multipartRequest);
    HttpSession session = multipartRequest.getSession();
    int parentNO = (Integer)session.getAttribute("parentNO");
    session.removeAttribute("parentNO");
    MemberVO memberVO = (MemberVO)session.getAttribute("memberInfo");
    String id = memberVO.getMember_id();

    System.out.println("id :: " + id);
    System.out.println("parentNO :: " + parentNO);

    articleMap.put("parentNO", parentNO);
    articleMap.put("id", id);
    articleMap.put("imageFileName", imageFileName);

    String message;
    ResponseEntity resEnt = null;
    HttpHeaders responseHeaders = new HttpHeaders();
    responseHeaders.add("Content-Type", "text/html; charset=utf-8");
    try {
        int articleNO = boardService.addReply(articleMap);
        if(imageFileName != null && imageFileName.length() != 0) {
            File srcFile = new File(ARTICLE_IMAGE_REPO + "\\temp\\" + articleNO + imageFileName);
            File destDir = new File(ARTICLE_IMAGE_REPO + "\\article\\" + articleNO);
            FileUtils.moveFileToDirectory(srcFile, destDir, true);
        }
    }
}
```

고객 센터

문의사항

번호	작성자	제목	작성일
1	SWS	12312	2020. 8. 21
2	lee	하고싶은말이많은데~~	2020. 8. 21
3	SWS	[답변] dsad	2020. 8. 21
4	SWS	[답변] 24434	2020. 8. 21

1 2 3 4 5 6 7 8 9 10 next

글쓰기

소스코드

프로젝트 URL : <http://jasmine62.shop>

소스코드 : <https://github.com/osy0907/JasmineProject>

카카오 로그인 API

```
// 요청을 통해 얻은 JSON 타입의 Response 메시지 읽어오기
BufferedReader br = new BufferedReader(new InputStreamReader(conn.getInputStream()));
String line = "";
String result = "";

while ((line = br.readLine()) != null) {
    result += line;
}
System.out.println("response body > " + result);

// GSON 라이브러리에 포함된 클래스로 JSON 파싱 객체 생성
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();

System.out.println("access_token : " + access_Token);
System.out.println("refresh_token : " + refresh_Token);

br.close();
bw.close();
} catch (IOException e) {
    e.printStackTrace();
}
return access_Token;
}

public HashMap<String, Object> getUserInfo(String access_Token){
    // 요청하는 클라이언트마다 가진 정보가 다를 수 있기에 HashMap타입으로 선언
    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("POST");

        // 요청에 필요한 Header에 포함될 내용
        conn.setRequestProperty("Authorization", "Bearer " + access_Token);
        int responseCode = conn.getResponseCode();
        System.out.println("responseCode : " + responseCode);

        BufferedReader br = new BufferedReader(new InputStreamReader(conn.getInputStream()));

        String line = "";
        String result = "";

        while ((line = br.readLine()) != null) {
            result += line;
        }
        System.out.println("response body" + result);

        JsonParser parser = new JsonParser();
        JsonElement element = parser.parse(result);

        // properties와 kakao_account는 '값'이 json객체 타입이라서 값을 다시 json객체로 받아야 함
        JsonObject properties = element.getAsJsonObject().get("properties").getAsJsonObject();
        JsonObject kakao_account = element.getAsJsonObject().get("kakao_account").getAsJsonObject();

        // id는 '값'이 json객체 타입이 아니기 때문에 getAsString으로 바로 값 가져옴.
        String kakao_id = element.getAsJsonObject().get("id").getAsString();
        String nickname = properties.getAsJsonObject().get("nickname").getAsString();
        String email = kakao_account.getAsJsonObject().get("email").getAsString();
        System.out.println("email > " + email);
        String memail[] = email.split("@");
        System.out.println("email1 > " + memail[0]);
        System.out.println("email1 > " + memail[1]);
        String email1 = memail[0];
        String email2 = memail[1];
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

카카오계정 하나로 충분합니다.

Kakao의 모든 서비스뿐 아니라 Melon, Daum 등 다른 다양한 서비스까지 이제 카카오계정으로 이용해 보세요!



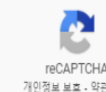
kakao

카카오계정 (이메일 또는 전화번호)

비밀번호

☒ 로그인 상태 유지

☐ 로봇이 아닙니다.



로그인

또는

QR코드 로그인

회원가입

카카오계정 | 비밀번호 찾기

소스코드

프로젝트 URL : <http://jasmine62.shop>

소스코드 : <https://github.com/osy0907/JasmineProject>

배송추적 API

```
public List<DeliveryInfo> (String t_key,String t_code,String t_invoice) {
    List<DeliveryVO> Delivery = new ArrayList<DeliveryVO>();
    try {
        URL obj = new URL("https://info.sweettracker.co.kr/api/v1/trackingInfo?t_key="+t_key+"&t_code="+t_code+"&t_invoice="+t_invoice);
        HttpURLConnection conn = (HttpURLConnection)obj.openConnection();
        conn.setDoOutput(true);
        conn.setRequestMethod("GET");

        BufferedReader in = new BufferedReader(new InputStreamReader(conn.getInputStream(),"UTF-8"));
        String line;
        String resultLine = "";
        while ((line=in.readLine())!=null) {
            resultLine += line;
        }
        in.close();
        System.out.println(resultLine);
        if (resultLine != null) {

            JsonParser parser = new JsonParser();
            JsonElement element = parser.parse(resultLine);

            String itemName = element.getAsJsonObject().get("itemName").getString();
            JsonArray trackingDetails = element.getAsJsonObject().get("trackingDetails").getAsJsonArray();
            int size = trackingDetails.size();

            for (int i = 0; i < size; i++) {
                DeliveryVO deliVO = new DeliveryVO();
                String timeString = trackingDetails.get(i).getAsJsonObject().get("timeString").getString();
                String where = trackingDetails.get(i).getAsJsonObject().get("where").getString();
                String telno = trackingDetails.get(i).getAsJsonObject().get("telno").getString();
                String kind = trackingDetails.get(i).getAsJsonObject().get("kind").getString();
                System.out.println("timeString >> " + timeString);
            }
        }
    } catch (Exception e) {
        e.printStackTrace();
    }
}
```

```
<title>스마트택배 배송조회</title>
<link rel="stylesheet" href="${contextPath}/resources/css/bootstr
<link rel="stylesheet" href="${contextPath}/resources/css/bootstr
<link rel="stylesheet" href="${contextPath}/resources/css/trackin
/head><body>
    <form method="get" action="${contextPath}/mypage/selectDelivery
        <input type="hidden" class="form-control" id="t_key" name="
            value="2U0kPL0iU1w0eFG8agdiSg">
        <div class="container-fluid">
            <div class="panel-body">
                <div class="col-md-12 header-title">택배 배송조회</div>
                <div class="col-md-12">
                    <div class="form-group row">
                        <label for="invoice" class="col-xs-4 label-
                        <input type="text" class="form-control" nam
                            placeholder="운송장 번호">
                    </div>
                    <div class="form-group row">
                        <label for="company" class="col-xs-4 label-
                        <select class="form-control" name="t_code">
                            <option value="01" selected="selected">
                            <option value="04">CJ대한통운</option>
                            <option value="05">한진택배</option>
                            <option value="06">로젠택배</option>
                            <option value="08">롯데택배</option>
                            <option value="53">농협택배</option>
                            <option value="08">롯데택배</option>
                            <option value="18">건영택배</option>
                            <option value="22">대신택배</option>
                            <option value="24">CVSnet 편의점택배</opti
                            <option value="12">EMS</option>
                            <option value="99">롯데글로벌로지스</option>
                            <option value="42">CJ대한통운 국제특송</opti
                            <option value="21">Fedex</option>
                            <option value="13">DHL</option>
                        </select>
                    </div>
                </div>
            </div>
        </div>
    </div>
```

택배 배송조회

운송장번호

운송장 번호

택배사

우체국택배

조회하기

시간	현재 위치	배송 상태
2020-08-18 15:19:00	용인우체국	집하완료
2020-08-18 15:26:00	용인우체국	접수
2020-08-18 18:08:00	용인우체국	발송
2020-08-18 18:48:00	성남우편집중국	도착
2020-08-18 20:11:00	용인우체국	발송
2020-08-18 20:39:00	성남우편집중국	도착
2020-08-18 21:16:00	성남우편집중국	발송
2020-08-18 23:55:00	중부권광역우편물류센터	도착
2020-08-19 04:43:00	중부권광역우편물류센터	발송
2020-08-19 06:57:00	천안우편집중국	도착
2020-08-19 07:39:00	천안우편집중국	발송
2020-08-19 07:40:00	천안우체국	도착

관리자 페이지 상품등록 조회

```
@RequestMapping(value="/adminGoodsMain.do",method={RequestMethod.POST,RequestMethod.GET})
public ModelAndView adminGoodsMain(@RequestParam Map<String, String> dateMap,
    HttpServletRequest request, HttpServletResponse response) throws Exception {

    String viewName=(String)request.getAttribute("viewName");
    ModelAndView mav = new ModelAndView(viewName);
    HttpSession session=request.getSession();
    session=request.getSession();
    session.setAttribute("side_menu", "admin_mode"); //마이페이지 사이트 메뉴를 설정한다.

    String fixedSearchPeriod = dateMap.get("fixedSearchPeriod");
    String section = dateMap.get("section");
    String pageNum = dateMap.get("pageNum");
    String beginDate=null,endDate=null;

    String [] tempDate=calcSearchPeriod(fixedSearchPeriod).split(",");
    beginDate=tempDate[0];
    endDate=tempDate[1];
    dateMap.put("beginDate", beginDate);
    dateMap.put("endDate", endDate);

    Map<String,Object> condMap=new HashMap<String,Object>();
    if(section== null) {
        section = "1";
    }
    condMap.put("section",section);
    if(pageNum== null) {
        pageNum = "1";
    }
    condMap.put("pageNum",pageNum);
    condMap.put("beginDate",beginDate);
    condMap.put("endDate", endDate);
    List<GoodsVO> newGoodsList=adminGoodsService.listNewGoods(condMap);
    mav.addObject("newGoodsList", newGoodsList);

    String beginDate1[]=beginDate.split("-");
    String endDate2[]=endDate.split("-");
    mav.addObject("beginYear",beginDate1[0]);
    mav.addObject("beginMonth",beginDate1[1]);
    mav.addObject("beginDay",beginDate1[2]);
    mav.addObject("endYear",endDate2[0]);
    mav.addObject("endMonth",endDate2[1]);
    mav.addObject("endDay",endDate2[2]);

    mav.addObject("section", section);
    mav.addObject("pageNum", pageNum);
    return mav;
}

@RequestMapping(value="/addNewGoods.do",method={RequestMethod.POST})
public ResponseEntity addNewGoods(MultipartHttpServletRequest multipartRequest, HttpServletResponse
    response) throws Exception {
    String imageFileName=null;

    Map newGoodsMap = new HashMap();
    Enumeration enu=multipartRequest.getParameterNames();
    while(enu.hasMoreElements()){
        String name=(String)enu.nextElement();
        String value=multipartRequest.getParameter(name);
        newGoodsMap.put(name,value);
    }

    HttpSession session = multipartRequest.getSession();
    MemberVO memberVO = (MemberVO) session.getAttribute("memberInfo");
    String reg_id = memberVO.getMember_id();

    List<ImageFileVO> imageFileList =upload(multipartRequest);
    if(imageFileList!= null && imageFileList.size()!=0) {
        for(ImageFileVO imageFileVO : imageFileList) {
            imageFileVO.setReg_id(reg_id);
        }
        newGoodsMap.put("imageFileList", imageFileList);
    }

    String message = null;
    ResponseEntity resEntity = null;
    HttpHeaders responseHeaders = new HttpHeaders();
    responseHeaders.add("Content-Type", "text/html; charset=utf-8");
    try {
        int goods_id = adminGoodsService.addNewGoods(newGoodsMap);
        if(imageFileList!=null && imageFileList.size()!=0) {
            for(ImageFileVO imageFileVO:imageFileList) {
                imageFileName = imageFileVO.getFileName();
                File srcFile = new File(CURR_IMAGE_REPO_PATH+"\\temp\\"+imageFileName);
                File destDir = new File(CURR_IMAGE_REPO_PATH+"\\goods_id");
                FileUtils.moveFileToDirectory(srcFile, destDir,true);
            }
        }
        message= "<script>";
        message += " alert('상품등록 추가되었습니다.');
```

ADMINISTRIES

COMMUNITY

LOCATIONS

BLOG

상품 조회

☒등록일로조회 ☐상세조회

2020년 8월 27일 이전 1주 1개월 3개월 까지 조회

전체 조회

조회한 기간 : 2020년 04월 27일 ~ 2020년 08월 27일

상품번호	상품명	저자	출판사	상품가격	입고일자	출판일
494	김미경의 리부트	ds	sda	13500	2020-08-24	2020-08-04
493	김미경의 리부트	ds	sda	13500	2020-08-23	2020-08-12
492	sad	dsa	dsa	123	2020-08-23	2020-08-23
491	김미경의 리부트	ds	sda	13500	2020-08-23	2020-08-06
401	좋아서 하는 일에도 돈은 필요합니다	이랑	창비	13500	2020-08-18	2018-10-02
400	Evil Men	제임스 도즈	오월의 봄	17100	2020-08-18	2020-08-18
402	영어가 만만해지는 책	벤벡	채인지업	13500	2020-08-18	2018-10-02
449	SLAM DUNK ILLUSTRATIONS	이노우에 타케히코	대원	35100	2020-07-30	2020-07-30
450	코믹 어느 날 공주가 되어버렸다 4	클루토스	캐롯툰	12600	2020-07-30	2020-07-30
451	여명기 女命記	팀 송명기	위즈덤하우스	28800	2020-07-30	2020-07-30

1 2 3 4 5 6 7 8 9 10 다음

상품 등록

소스코드 : <https://github.com/osy0907/JasmineProject>

23 / 28

후기

고, 1

프로젝트 후기



오승용

프로젝트를 하면서 '협업'이 무엇인지를 배우는 좋은 계기가 되었습니다. 서로 맡은 부분은 최선을 다하면서 모르는 부분은 서로 코드리뷰를 통해 학습하며 서로의 정보지식을 공유하면서 서로서로 발전하고 공부가 될 수 있는 시간이 되었습니다. 반면, 어느 부분이 부족한지를 팀 프로젝트를 하면서 다시 한 번 생각 하게 되었습니다. 프로젝트를 진행하면서 모르는 부분 막히는 부분이 있을 때 어떻게 문제를 해결해 나가야할까 라는 문제해결능력의 중요성에 대해 느꼈고 또 많이 배웠다고 생각합니다. 그리고, ComputerScience의 기초 특히 그 중에서도 네트워크와 자료구조에 대한 지식을 탄탄히 다지는것에 대한 중요성을 느꼈습니다.



최영권

개발에서 가장 중요한 부분 중 하나가 협업이라고 느끼게 되었습니다. 프로젝트를 진행하면서 1달이라는 기간동안 개인이 지치거나 힘들어도 팀원들이 서로 이끌어가고 묵묵히 할 일을 하는 것을 보면서 자극이 많이 되었고 또한 누군가 못하는 일을 다른 사람이 해결해주고 서로 지식을 쌓아가면서 많이 발전했다고 생각합니다.