

소프트웨어 시스템 개발 PBL 모듈2 최종 보고서

구 분	세부내용		
팀 명	팔색조 2팀		
팀 구 성	성명	학번	학과
팀 장	김시라	20150970	컴퓨터학과
팀 원	오유민	20150253	컴퓨터학과

□ 목차 (요약)

1. 핵심 기능 및 특징 (구현 완료된 기능)
2. 설계 (기존 설계에서 변경/추가된 사항만)
3. 구현방법 (사용 기술, 알고리즘 등)
4. 요구사항 만족도 (진도표)

□ 핵심 기능

세부 내용	
핵심 기능	<input type="checkbox"/> [1. 일반 상품 CRUD / 경매 상품 CRUD]

및 특징

- 일반/경매 등록 form

Sale Registration

DOGS 종류를 선택하십시오.

종류

품목

삼살개

포메라니안

시베리안허스키

시츨

➔

Auction Sale Registration

DOGS

삼살개

Item Title	<input type="text"/>
Seller	admin
Image	<input type="text" value="파일 선택"/> 선택된 파일 없음
Title	<input type="text"/>
start Price	<input type="text" value="0"/>
Start Date	<input type="text"/>
Last Date	<input type="text"/>
Auction Name	<input type="text"/>
Att 1	<input type="text"/>
Att 2	<input type="text"/>

➔

Check Your Registration Information


Item ID	1039
Product ID	11
Category ID	DOGS
Item Title	삼살개,삼살개 경매입니다
Seller	
Start Price	1000
Start Date	18-06-22
Last Date	18-06-30
Att1	cute
Att2	
Att3	
Att4	
Att5	
Quantity	3

category에 따른 product 값 선택받고 사용자에게 세부사항을 입력 받는다. 등록 전에 입력한 정보가 맞는지 확인하는 form을 보여준다.

- 등록된 아이템 조회

터키시 앙고라			
Image	Item	가격	
	<u>auctionTestDog</u> attr1 / //	0.0	
	<u>asdf</u> attr1 / adsf /	1234.0	





product name터키시 앙고라

Item Title auctionTestDog

attribute1 attr1

attribute2

attribute3

PRICE 0.0

Quantity 1

SELLER admin

- 등록된 아이템 수정 및 삭제

1035	0	광고 중	일반	삭제	수정
1038	0	광고 신청	경매	삭제	수정
1029	0	광고 중	경매	삭제	수정
1030	1234	광고 중	일반	삭제	수정



General Item 수정

1026

Item Title	<input type="text" value="auctionTestDog"/>
Image	
list Price	<input type="text" value="1234.0"/>

MyPage 나의 판매 내역에서 내가 등록한 아이템 리스트를 보여주며, 아이템을 수정 및 삭제가 가능하다.

▶ [2. 경매 상품 입찰 / 낙찰 서비스]

경매 아이템의 상세보기에서 입찰참여하기 버튼을 통해 입찰페이지로 이동.

item title	이구마나,auctionTestReptiles
attribute1	attr1
attribute2	attr2
attribute3	
START_PRICE	5555
START_DATE	2018-06-20 00:00:00
END_DATE	2018-06-29 00:00:00
Quantity	1
SELLER	admin

신고하기

경매 참여하기

입찰페이지 이동버튼

입찰 부분 데이터 CRUD

- update는 반드시 화면에 표시되는 현재 낙찰가보다 높은 금액이어야 한다(최초 입찰가는 사용자가 정한다).

현재 입찰가 : 12345 원

Ranking	User ID	Bid Price
1	j2ee	12345

현재 경매에 참여한 인원 : 1 명
현재 내 입찰가 : 12345 원

입찰취소 버튼 :

Update 버튼(입찰 참여)

Delete 버튼

localhost:8112 내용:

현재 낙찰가보다 높은 금액을 입력하세요!

update 제한 alert 창

localhost:8112 내용:

입찰을 취소하시겠습니까?(해당 아이템에 대한 입찰이 모두 취소됩니다.)

delete 확인 alert 창

낙찰은 두가지 방식으로 전개된다.

- 판매자가 강제로 경매를 종료
- 사용자가 경매를 등록할 때 정한 마감일이 되면 종료

product name 이구아나

Item Title 이구아나,auctionTestReptiles

attribute1 attr1

attribute2 attr2

attribute3

START_PRICE 5555

START_DATE 2018-06-20 00:00:00

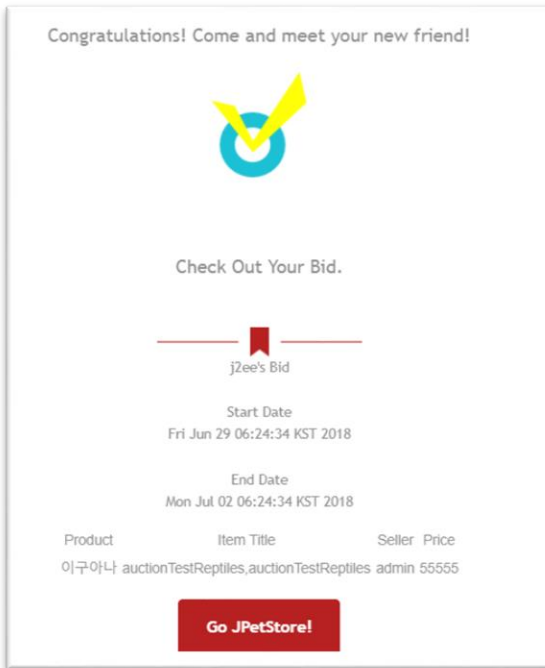
END_DATE 2018-06-29 00:00:00

Quantity 1

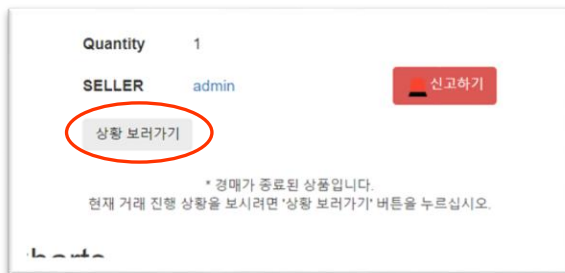
SELLER admin

강제 낙찰 종료 버튼
(경매 아이템 상세보기
페이지, 다른 유저들에게
는 입찰 참여 버튼으로
보이고 판매자에게만 낙
찰 버튼으로 보임)

낙찰이 종료되면 입찰자에게 3일 이내로 구매해달라는 메일이 전달된다.
만약 3일 이내로 구매하지 않으면 다음 차순위 낙찰자에게 구매 기회가 넘어간다.
구매는 입찰 페이지에서 진행할 수 있다(현재 낙찰자에게만 버튼이 보임).



아이템 낙찰 알림 메일



Ranking	User ID	Bid	Price
1	j2ee	12345	

현재 경매에 참여한 인원 : 1 명
현재 내 입찰가 : 12345 원

현재 거래중인 고객 : j2ee
현재 거래 금액 : 12345

[Add to Cart](#)

▶ [3. Q & A 게시판]

- 질문 보기 및 등록

질문 게시판

번호	제목	작성자	작성일
1	흠나	test2	18/00/26
2	궁금한 거 있어요!	j2ee	18/00/25
3	나이가 궁금해요	aaaa	18/00/24



애완동물에 대해 궁금한 점을 질문해주세요.

질문자ID	sira
제목	궁금한 거 있어요!
질문 내용	정말 귀여운가요?

상세보기 페이지의 질문 게시판에서 질문하기 버튼을 클릭하여 질문을 등록한다.

- 답변 보기 및 등록

Item Id : 1017

글쓴이	j2ee	날짜	18/00/25
제목	궁금한 거 있어요!		
얼마나 귀여운가요?			

1	영형 귀여워요	sira	18/00/26
---	---------	------	----------



localhost:8080 내용:

답변 등록이 완료되었습니다.

답변하기

질문 --- 귀여워요!

해당 질문을 클릭하면 질문의 내용과 그에 달린 답변들을 확인할 수 있다.
또한 내가 그 질문에 대해 답변하는 것도 가능하다.

Quantity 1

SELLER admin

Add to Cart



Report Black List 블랙리스트 신고하기

신고자 ID	j2ee
<input type="checkbox"/> 부적절한 홍보 게시물 <input type="checkbox"/> 음란성 또는 부적절한 내용 <input type="checkbox"/> 허위/착오 <input type="checkbox"/> 안전공격/영악행위 <input type="checkbox"/> 불법적인 유출 <input type="checkbox"/> 같은 내용의 반복(도배) <input type="checkbox"/> 지역감정 조장 <input type="checkbox"/> 기타 게시물과 관계없는 내용	

신고하기 전에 잠깐!

신고하시면 해당 게시물은 삭제될 수 있으며, 글쓰기 또한 금지될 수 있습니다.
허위 신고의 경우 신고자의 활동에 제한을 받게 되니 이점 유의해 주시기 바랍니다.



Black List 신고가 완료되었습니다.

BlackList ID	admin
신고 사유	부적절한 홍보 게시물

▶ [4. 블랙리스트]

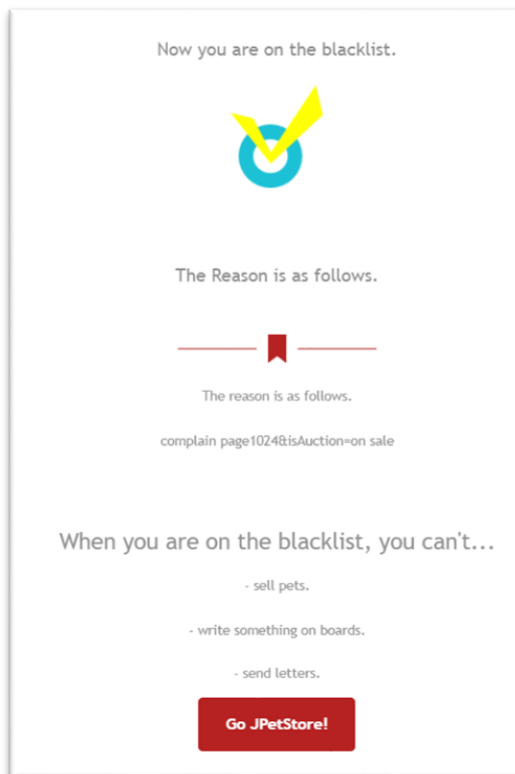
-신고하기

아이템 상세보기 페이지의 신고하기 누르면 판매자 신고가 가능하다.
중복 신고는 허용하지 않으며, 자기 자신을 신고할 수도 없다.
로그인을 해야 하는 서비스이다.

같은 페이지에 대한 신고횟수가 10회 초과하면 그 페이지가 블라인드 처리가 되고,
그 게시물을 작성한 판매자는 블랙 리스트에 올라서 판매를 할 수 없게 된다.
블랙리스트에 오른 판매자에게는 즉시 블랙리스트에 올랐다는 메일을 전송해준다.



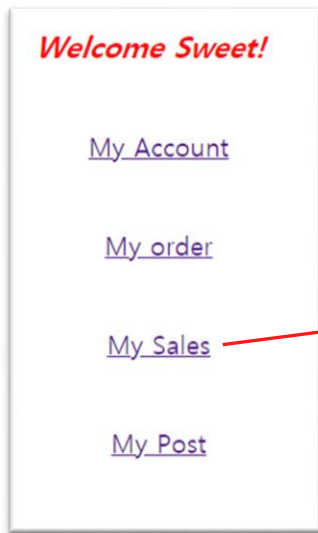
블라인드로 막힌 페이지와 블랙리스트에 오른 유저의 쓰기 권한이 제한된 모습



신고로 인해 블랙리
스트에 오른 사용자
의 메일 주소로 전
송된 메일(제한되는
활동을 설명).

▶ [5. 마이 페이지]

회원과 관련된 내역들을 조회할 수 있는 페이지
- 계정관리, 주문 내역, 입찰 내역, Q&A와 review, 판매 게시물



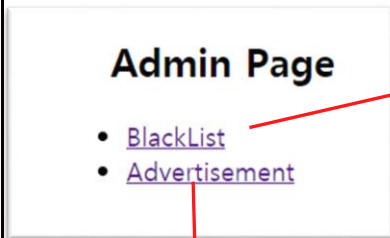
판매중		판매완료	
글번호	item	가격	광고
1	1026	1234	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>
2	1027	1234	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>
3	1028	12	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>
4	1032	1234	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>
5	1033	1234	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>
6	1034	0	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>
7	1035	0	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>
8	1038	0	승인 대기 경매 <input type="button" value="삭제"/> <input type="button" value="수정"/>
9	1029	0	광고 중 경매 <input type="button" value="삭제"/> <input type="button" value="수정"/>
10	1030	1234	광고 중 일반 <input type="button" value="삭제"/> <input type="button" value="수정"/>

1 2 Next >>

사용자가 올린 판매중인 or 판매 완료된 아이템들 조회 가능

▶ [6. 관리자 페이지]

매핑 url 주소 /shop/admin.do, 관리자 이외의 계정이 로그인 시도시 /shop/index.do로 이동.



No.	아이디	신고당한 페이지	리스트에 오른 날짜	전체 선택
1	sira	/shop/viewItem.do?itemId=1024&isAuction=on	20180626	<input type="checkbox"/>

j2ee님이 신고당한 이유

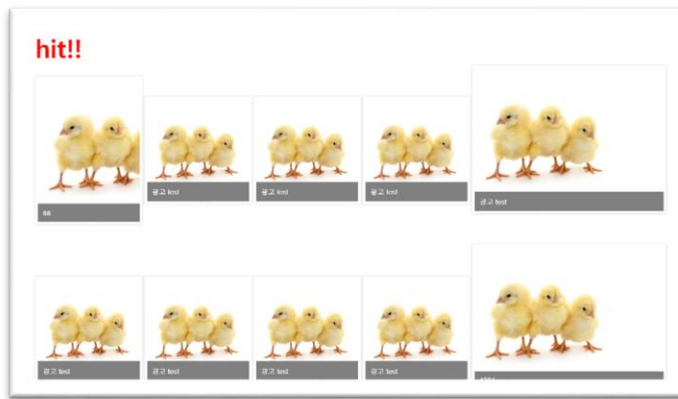
No.	신고당한 페이지	이유
1	/shop/viewItem.do?itemId=1024&isAuction=on	-
2	/shop/viewItem.do?itemId=1024&isAuction=on	-
3	/shop/viewItem.do?itemId=1024&isAuction=on	-
4	/shop/viewItem.do?itemId=1024&isAuction=on	-
5	/shop/viewItem.do?itemId=1024&isAuction=on	-
6	/shop/viewItem.do?itemId=1024&isAuction=on	-
7	/shop/viewItem.do?itemId=1024&isAuction=on	-
8	/shop/viewItem.do?itemId=1024&isAuction=on	-
9	/shop/viewItem.do?itemId=1024&isAuction=on	-
10	/shop/viewItem.do?itemId=1024&isAuction=on	-

No.	아이디	아이템 페이지	신청 기간	전체 선택
1	admin	1037	2018-06-14 13:50:16 ~ 2018-06-30 13:50:16	<input type="checkbox"/>
2	admin	1038	2018-06-23 14:48:23 ~ 2018-06-30 14:48:23	<input type="checkbox"/>
3	admin	1038	2018-06-23 14:49:00 ~ 2018-06-30 14:49:00	<input type="checkbox"/>

전체 선택으로 편하게 관리가 가능.

관리자가 광고 승인/거부, 블랙리스트 제외시키기 등을 할 때, 관련 사용자에게 메일이 전달된다. 게시물에 대한 신고가 타당한지 알 수 있도록 관리자는 블라인드 처리된 게시물에 접근이 가능하도록 했다.

▶ [7. 광고]



메인에 광고가 뜬다. 로그인한 사용자에게는 가입할 때 입력한 FavCategory와 카테고리가 일치하는 아이템 광고를 띄워준다(추적 광고).

광고 신청은 MyPage -> MySales에서 진행.

광고 신청에서는 3가지 상태가 있음(신청/승인 대기/광고중).

광고

광고 중

[광고 신청](#)

[광고 신청](#)

이미지 첨부 : [파일 선택](#) [선택된 파일 없음](#)

ad Title =

광고 시작 날짜

광고 끝나는 날짜

광고 신청이 정상적으로 완료되었습니다!

입금 계좌: xxxxxxx-xx-xxxxxxx

* 입금 기한은 신청 후 3영업일로, 그 안에 입금되지 않을 시 신청이 취소됩니다.

예) 2018년 5월 22일 화요일 신청 시, 5월 25일 금요일까지 입금.
2018년 5월 25일 금요일 신청 시, 5월 30일 수요일까지 입금.

* 입금하신 후 관리자의 승인을 거쳐야 정상적으로 광고가 게재됩니다. 승인은 입금 후 이틀 안에 진행됩니다.

예) 2018년 5월 25일 입금 시, 최대 5월 27일 내로 승인.

[확인](#)

[판매중](#)

[판매완료](#)

글번호	item	가격	광고	
1	1023	3000	광고 중	일반
2	1025	300	승인 대기	일반
3	1018	0	광고 신청	경매

아이템에 대한 광고 상태

광고 신청 후 3일 이내 입금하지 않으면 광고 승인이 자동으로 거부된다.

▶ [8. 쪽지]

- 쪽지 조회 및 보내기



사용자간 쪽지를 주고 받을 수 있다.
ajax를 이용하여 단어 자동 완성 기능을 구현하였다.

- 쪽지 검색

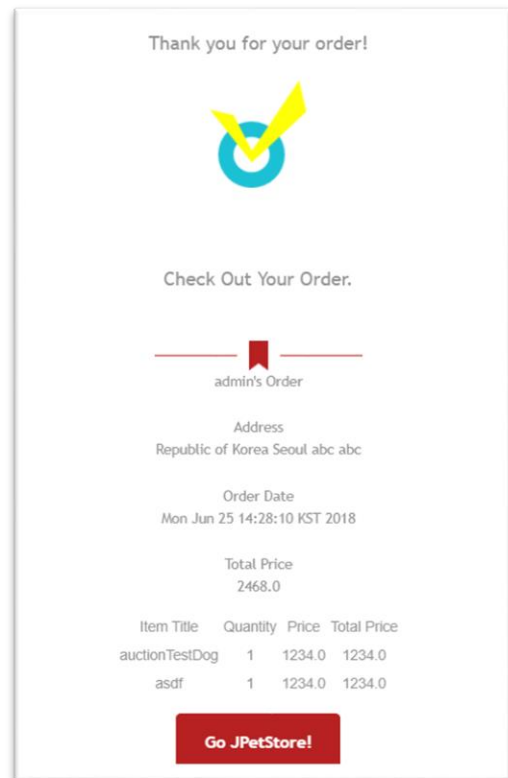
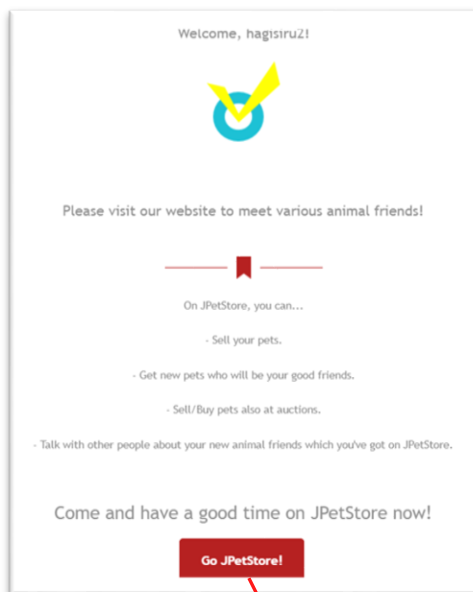
쪽지를 찾고 싶을 때 쪽지를 검색하는 것도 가능하다. 해당 단어가 들어 있는 모든 쪽지를 검색해서 보여준다.

▶ [9. 메일]

S&Y PetStore에서 사용자에게 특정 이벤트에 대해 알림 메일을 보낼 수 있다.

- 회원가입, 구매시 구매 내역, 블랙리스트 오름/제외됨, 광고 승인/거부, 경매에서 낙찰 알림.

주로 사용자가 알아야 하는 내용을 메일로 보내준다.



회원 가입 메일과 구매 내역 메일

S&Y PetStore로 이동

▶ [10. 판매자에 대한 후기]

게시물 목록						
번호	제목	글쓴이	날짜	조회수	추천수	
1		sira	2018-06-24	1	0	

◀ 1 2 ▶

제목 ▼ 검색 검색 검색초기화

글쓰기



후기 등록

✎ 제목

👤 sira

🏠 비밀번호

✎ 내용

📎 파일 선택

취소 등록

- 판매자에 대한 후기 조회 및 후기 남기기

	<p>상세보기 페이지의 판매자 버튼을 누르면 판매자에 대한 후기를 남길 수 있는 포스팅 게시판이 나온다. 이 곳에서 판매자의 후기를 조회할 수 있고 사용자가 직접 후기도 등록할 수 있다.</p>

□ 설계

부 내용		
DB 추가/변경	ITEM TABLE 수정	<p>- ISAUCTION : 경매 여부를 나타내는 필드로 일반 판매는 0, 경매 판매는 1이었던 기존 설정 방식을 경매 ON/OFF 로 변경</p> <p>- ISAD : 기존의 ISBANNER를 변경</p>

ITEM	
ITEMID	varchar2(10 char) (NN)
PRODUCTID	varchar2(10 char)
CATID	varchar2(10 char)
LISTPRICE	number(10,2)
UNITCOST	number(10,2)
SHILLER	varchar2(10 char)
STATUS	varchar2(10 char)
ATTR1	varchar2(80 char)
ATTR2	varchar2(80 char)
ATTR3	varchar2(80 char)
ATTR4	varchar2(80 char)
ATTR5	varchar2(80 char)
ISAUCTION	varchar2(10 char)
ISAD	varchar2(10 char)
IMAGE	varchar2
ITEMTITLE	varchar2(80 char)

-IMAGE

: 사용자로부터 이미지를 입력받아서
ITEMID_사용자이미지파일.jsp 식의 형태로 파일명 저장

- ITEMTITLE

: 아이템 특징을 살리기 위한 TITLE 도입

AUCTION TABLE 수정

AUCTION	
AUCTIONID	varchar2(10 char) (NN)
ITEMID	varchar2(10 char) (NN)
SELLER	varchar2(80 char) (NN)
START_PRICE	number
START_DATE	date
END_DATE	date
ORDERID	varchar2(10 char)
IMAGE	varchar2(30 char)
AUCTIONNAME	varchar2(30 char)
STATUS	varchar2(80 char)

- AUCTIONNAME

: 경매의 특징을 강조하기 위한 경매 이름 도입

LETTER TABLE 추가

LETTER	
LETTERID	varchar2(80 char) (NN)
SENDER	varchar2(80 char)
RECEIVER	varchar2(80 char)
TITLE	varchar2(100 char)
CONTENT	varchar2(1000 char)
DATE_TIME	date
RECEIVED_DATE	date

사용자간의 원활한 의사소통을 위한 쪽지 기능 구현을
위한 테이블

- LETTERID : 쪽지의 번호 (PK)

- SENDER : 보내는 사람 (FK)

- RECEIVER : 받는 사람 (FK)

- TITLE : 쪽지 제목

- CONTENT : 전달할 내용

- DATE_TIME : 쪽지는 전송한 시각

- RECEIVED_DATE : 수신자가 쪽지를 확인한 시각

	<p>COMPLAIN TABLE 추가</p> <pre> COMPLAIN COMPLAINID varchar2(10 char) (NN) BLACKID varchar2(10 char) BLACKREASON varchar2(80 char) PAGE varchar2(80 char) DATETIME date COMPLAINANT varchar2(10 char) </pre>	<p>신뢰있는 판매를 위한 판매자 신고 기능을 위한 테이블</p> <ul style="list-style-type: none"> - COMPLAINID : 신고 ID (PK) - BLACKID : 신고 당한 사람의 ID - BLACKREASON : 신고 사유 - PAGE : 신고된 페이지 - DATETIME : 신고 시간 - COMPLAINANT : 신고자

□ 구현 방법

세부 내용	
구현 방법	<p>▶ [1. 일반 상품 CRUD / 경매 상품 CRUD]</p>

```

@Transactional
public void updateGeneralItemAll(GeneralItem generalItem) {
    generalItemDao.updateGeneralItem(generalItem);
    generalItemDao.updateQuantity(generalItem);
}

@Transactional
public void updateAuctionItemAll(AuctionItem auctionItem) {
    auctionItemDao.updateAuctionItem(auctionItem);
    auctionItemDao.updateAuctionInfo(auctionItem);
    auctionItemDao.updateQuantity(auctionItem);
}

@Transactional
public void deleteItemAll(String itemId) {
    itemDao.deleteInventoryFromTable(itemId);
    itemDao.deleteQuestionFromTable(itemId);
    itemDao.deleteItemFromTable(itemId); //마지막에 실행하지 않으면 무결성 오류
}

@Transactional
public void deleteAuctionInfoAll(String auctionId) {
    itemDao.deleteBidFromTable(auctionId);
    itemDao.deleteAuctionFromTable(auctionId);
}

```

트랜잭션을 이용하여 일반 아이템의 경우 ITEM TABLE과 INVENTORY TABLE, 경매 아이템의 경우는 ITEM, INVENTORY, AUCTION TABLE를 동시에 INSERT, UPDATE 처리.

DELETE는 해당 ITEM에 관련된 게시물들과 입찰가격들을 함께 DELETE 처리.

▶ [2. 경매]

- 외부 API 이용하여 하이차트 구현

```

$(document).ready(function() {
    var list = [];
    <c:forEach items = "${bidListChart}" var="item">
        list.push("${item.bid_date}");
    </c:forEach>

    var data = [];
    <c:forEach items = "${bidListChart}" var="item">
        data.push(Number("${item.bid_price}"));
    </c:forEach>

    var title = {
        text: 'Auction Progress'
    };
    var subtitle = {
        text: '경매 상황'
    };

```

▶ [3. Q & A 게시판]

- ViewItemContorller에서 해당 ITEMID에 따른 질문 내역을 리스트 객체주어
jsp에서 <c:forEach>문을 통해 전달 받음.

```

//Question List
List<Question> questionList = this.petStore.getListQuestion(itemId);

if (isAuction.equals("off")) {
    GeneralItem item = this.gItemDao.getGeneralItem(itemId);
    model.put("item", item);
    model.put("product", item.getProductId());
    model.put("questionList", questionList);

    return "tiles/GeneralItem";
}

```

- ANSWER의 경우에는 해당 questionId를 입력받아 조회.

```

@Override
public List<Answer> getAnswerList(String questionId){
    return answerDao.getAnswer(questionId);
}

```


▶ [8. 쪽지]

```
...
function searchajax() {
    $("#searchword")
        .keyup(
            function() {
                var words = $("#searchword").val();
                if (words != '') {
                    $
                        .ajax({
                            type : 'POST',
                            url : '/jpetstore/shop/searchAjax.do',
                            data : {
                                searchword : words
                            },
                            dataType : 'json',
                            success : function(result) {
                                if (result.accountList.length > 0) {
                                    var str = ''
                                    for (var i = 0; i < result.accountList.length; i++) {
                                        str += '<a href="#" class="list-group-item selected">'
                                            + result.accountList[i].user_id
                                            + '</a>';
                                    }
                                    $("#results").html(str);
                                } else {
                                    $("#results").html("");
                                }
                            },
                            error: function(e){
                                console

```

- ajax 이용한 단어 자동 검색 기능 구현.

- DB에서 단어 검색시 LIKE 조건을 이용하여 데이터 찾기.

```
<!--검색 세는것 -->
<select id="seLectBySearchWord" parameterType="Map" resultMap="Letter">
    SELECT
        *
    FROM
        letter
    where (sender like '%'||#{loginID}||'%' or receiver like '%'||#{loginID}||'%' )a

</select>
<!--전체쪽지수 세는것 -->
<select id="countALLLetter" resultType="int"
    parameterType="map">
    SELECT COUNT(*) count
    FROM letter
    WHERE
        sender like '%'||#{loginID}||'%' or receiver like '%'||#{loginID}||'%'
    -

```

▶ [9. 메일]

```
<bean id="mailSender" class="org.springframework.mail.javamail.JavaMailSenderImpl">
  <property name="host" value="smtp.gmail.com" />
  <property name="port" value="587" />
  <property name="username" value="springTest45@gmail.com" />
  <property name="password" value="jejudo12!@" />
  <property name="javaMailProperties">
    <props>
      <prop key="mail.transport.protocol">smtp</prop>
      <prop key="mail.smtp.auth">true</prop>
      <prop key="mail.smtp.starttls.enable">true</prop>
      <prop key="mail.debug">true</prop>
    </props>
  </property>
  <property name="defaultEncoding" value="utf-8" />
</bean>
```

- SMTP를 이용하여 메일 전송.

- 자바에서 지원하는 JavaMailSender를 이용.

```
try {
    MimeMessage message = mailSender.createMimeMessage();
    MimeMessageHelper messageHelper = new MimeMessageHelper(message, true, "UTF-8")

    messageHelper.setFrom(setfrom);
    messageHelper.setTo(tomail);
    messageHelper.setSubject(title);
    messageHelper.setText(sb.toString(),true);
    System.out.println("ad1");
    mailSender.send(message);
    System.out.println("ad2");
} catch (Exception e) {
    System.out.println("ad3");
    e.printStackTrace();
}
```