

# Java 程序设计实践（Web 篇）

## 实习报告

班 级： 10041612

姓 名： 白云鹏

学 号： 1004161221

指导教师： 牛云云

提交日期： 2018.07.24

# 目录

1 目的.....	- 1 -
2 主要内容.....	- 1 -
2.1 开发背景.....	- 1 -
2.2 技术介绍.....	- 1 -
2.3 需求分析.....	- 2 -
2.4 系统设计.....	- 3 -
2.5 系统实现.....	- 4 -
3 感想与总结.....	- 4 -
4 附录.....	- 6 -
4.1 代码.....	- 6 -
4.2 实现效果.....	- 16 -

## 1 目的

通过一周左右的 Java Web 的学习，在网站的搭建上，从前端到后端都有了一个比较初步的了解。此次实习的任务主要是使用 JSP 与 Servlet 技术实现简易的网上购物商城。在开发过程中学习使用 Html、CSS、JS 等来实现网页的基本界面，使用 Java 来组建其后端服务部分。同时在开发过程中对于数据库，服务器等的配置搭建也有所涉猎，为毕业之后的工作打好一定的基础。

## 2 主要内容

### 2.1 开发背景

#### 2.1.1 Java 语言

java 是一种高级的面向对象的程序设计语言。不同的平台有不同的 JVM，而 java 字节码不包含任何与平台相关的信息，不直接与平台交互，而是通过 JVM 间接与平台交互。因此其具有:(1)平台无关性(2)安全性(3)面向对象(4)异常处理

#### 2.1.2 Html 语言

网页的本质就是超级文本标记语言，通过结合使用其他的 Web 技术，可以创造出功能强大的网页。因而，超级文本标记语言是万维网编程的基础，也就是说万维网是建立在超文本基础之上的。

#### 2.1.3 CSS

层叠样式表是一种用来表现 HTML 或 XML 等文件样式的计算机语言。CSS 不仅可以静态地修饰网页，还可以配合各种脚本语言动态地对网页各元素进行格式化。

#### 2.1.4 JavaScript

JavaScript 一种直译式脚本语言，是一种动态类型、弱类型、基于原型的语言，内置支持类型。它的解释器被称为 JavaScript 引擎，为浏览器的一部分，广泛用于客户端的脚本语言，最早是在 HTML 网页上使用，用来给 HTML 网页增加动态功能。

#### 2.1.5 mysql

MySQL 是一种关系数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性。

### 2.2 技术介绍

#### 2.2.1 JSP 技术

JSP 技术可以分为三大部分，即 JSP 指令，JSP 脚本元素和变量以及 JSP 动作。在编写

JSP 上，可以直接将相应的 JSP 组成成分添加到相应的 html 元素中，在加载的时候就会执行其中的动作，而且客户端是无法读取到相应的 JSP 元素的，也保证了其相应的安全性。区别于其他很多的语言在平台的基础上也非常的完善，但是相应的，这也增加了 JSP 的复杂性，同时 JSP 脚本的运行也是建立在 Class 类的基础上的，在内存使用方面劣势也比较明显。

### 2.2.2 Servlet 技术

项目在前后端的交互上，使用了比较经典的 Servlet 技术，Servlet 程序在服务器端运行，动态地生成 Web 页面。在编写上，Servlet 技术允许使用 Java 语言强大而广泛的包作为其编写基础，由此在编写方面上，封装了很多基础的操作，比如在获取客户端请求上，只需重载相应的方法就可以接受相应方式的请求。在服务方面，Servlet 也包含了很多常用的操作，比如在基本的 cookie 和 session 的操作上，之需要调用相应的 API 就可以进行对应的设置以及读取，即便是在一些新兴的语言中，功能也又可能没有这样完善，比如在新兴的 Go 语言中缺乏对 Session 的相关处理。在数据处理方面，Servlet 也能够对系统进行一些基本的操作，比如在导入相关的包之后就可以与数据库进行连接。其处理流程在逻辑上也比较简单，即前端发送请求至相应的 Servlet，Servlet 生成响应内容并将其传给 Server，最后 Server 再将响应返回给客户端从而完成一次请求。而且其在不同的平台之上也能够流畅运行，即所谓的一次编写，到处运行。

### 2.2.3 Session 技术

由于 Http 协议的无状态性，而客户与服务器进行交互的时候，往往都需要保存其前态，由此，诞生了 Cookie 与 Session 技术，相比于 Cookie 技术，Session 技术讲所需要的数据保存在服务器上，Cookie 技术则将数据保存在客户端上，如果客户端将 Cookie 技术禁用，那么 Cookie 技术则就将失效。而 Session 技术则无此后患。当然在实现上，Session 技术也相对比较复杂，在 Cookie 技术中，客户端每次发送请求时都将 Cookie 放在 Http 的协议头中进行发送，而 Session 在创立时，仅为客户端分配一个 Session ID 对应的数据则根据 Session ID 进行查找，减小客户端消耗的同时也增大了客户端的消耗。

## 2.3 需求分析

### 2.3.1 查看关键商品

客户在进入系统之后，首先会跳转到首页，而在首页中，则应该向客户展示比较关键的商品，在这里使用了简单的轮播图作为其展示方式。

### 2.3.2 查看所有商品

当然客户不能只查看比较关键的商品,也要能浏览所有的商品,由此则需要为其制作相应的预览界面,在此界面上应留有商品的简略信息供用户进行选择。

### 2.3.2 查看商品的详细信息

对于每一个商品,都应该为其生成相应的商品详情页,介绍其具体的商品详情,但是由于时间的限制,详情页制作的相对较为简易。

### 2.3.3 查询相应的商品

在商品较多的情况下查询商品则显得比较重要,这里将商品首先进行分类,用户在查询的时候,选择对应的分类,服务器端返回分类下所有的商品。

### 2.3.4 添加相应商品至购物车

对于每一个商品,如果客户有购买意愿,那么此时就应该将商品添加至个人的购物车,此时就需要为商品添加相应的功能根据客户的 Session 来将商品添加至个人的购物车。

### 2.3.5 生成订单

将购物车中的商品进行结算,生成相应的订单。

### 2.3.6 注册与登录

在注册上逻辑则较为简单,只需要保证非空元素的必须值,以及关键元素不可重复的属性等,在注册成功后,需要将注册信息保存在数据库以供使用。在登陆前首先需要查看客户是否有相应的 Session,在登录的时候则需要先验证用户名以及密码的对应关系,如果验证成功,那么就需要为客户注册相应的 Session。

## 2.4 系统设计

在系统设计上,主要分为前端显示界面,后端的业务处理方面以及公用的实体类。

前端显示界面上,比较关键的就是首页的设计,因为在首页的编写与其他界面的编写有较大的区别,而且在编写过程中也比较复杂,而在其他的界面设计上,其中有很多的重复页面元素,在编写的过程中则相对而言比较容易。在这里主要的前端文件有:index.jsp(首页),head.jsp(界面的公共顶端部分),showDetail.jsp(商品详情页),lookCosmetic(浏览商品页)

而在后端的服务方面,则编写了很多的 Servlet 来进行对请求的处理,这里相对复杂的则为各式各样的服务,比较关键的服务都应有相应的 Servlet 类,在刚开始的注册登录方面则需要为其分别编写 handleRegister 以及 handleLogin 两个 Servlet,在浏览和查询上则有 QueryAllRecords 以及 SearchByCondition 两个 Servlet,在添加购物车以及生成订单上则有 PutGoodSToCar 以及 HandleBuyGoods 两个 Servlet 为其进行处理。

## 2.5 系统实现

在系统实现中，选取了三个我认为比较关键的技术实现：

### 2.5.1 登录处理

1. 从客户端请求中获取相应的用户名以及密码
2. 检验二者是否为空，如为空，打印错误信息，结束。
3. 检验用户是否已经被分配 Session，如果已经分配，打印错误信息，结束。
4. 为用户分配 Session。结束。

### 2.5.2 购物车处理

1. 从客户请求中获取相应的商品信息。
2. 检验用户是否被分配 Session，如果没有，跳转至登录界面。
3. 获取 Session，将相应商品添加到对应的实体类中。

### 2.5.3 查询商品处理

1. 从客户请求中获取相应的商品类别信息。
2. 从数据库中根据相应的商品类别信息获取类别码。
3. 从数据库中查询对应类别码所包含的产品，回送前端进行打印。

## 3 感想与总结

经过 Java web 的学习，首先对于网站的搭建有了初步的了解，首先需要配置 Apache/nginx 等对端口进行监听，不断捕获客户请求，接下来需要建立数据库的连接以及启动服务。在服务过程中，前端首先接受请求并转发至相应的后端接口进行处理，后端在处理结束后给予前端反馈，最后向用户提交服务。

在开发过程中，由于自己知识水平的狭隘，经常遇到了很多难以解决的问题，在项目的初始构建阶段总是认为思路简单，实现起来也不会困难，但是在真正编写的过程中却进度缓慢，我才明白“纸上觉来终得浅，绝知此事要躬行。”这句话的含义，在以后的学习过程中，即使是简单的问题也不容小觑，真真正正的解决以后才能有所进步。

在测试阶段，主要是在前端的测试上，由于 JSP 页面需要不断的重启服务来生成网页，导致测试时浪费了很多时间，这主要是因为编写过程中没有将页面渲染部分与核心内容部分加以分离，在以后的编写过程中，不断的对代码进行分类整理，不断对项目进行重构，保证高效开发。

项目的功能比较局限，很多经常的操作没有添加，我认为这主要是因为自己没有合理安排时间，在前期的时候没有把握时间对所学知识进行拓展，导致在后期开发比较困难，进展

较为缓慢。在以后的学习过程中不应浪费一分一秒，对于所学知识要及时加以复习与拓展，从而减少在开发过程中遇到的困难。保证开发的高效。将多余的时间反作用于项目，不断完善项目功能，保证项目尽量的完善。

## 4 附录

### 4.1 代码

lookCosmetic.jsp:

```
1. <%@ page contentType="text/html; charset=GB2312"%>
2. <%@ page import="java.sql.*"%>
3. <HTML>
4. <HEAD>
5. <%@ include file="head.jsp"%>
6. <link href="css/style2.css" rel="stylesheet" type="text/css"/>
7. </HEAD>
8. <style>
9. .ul05{
10.     padding-left:0px;
11.     list-style: none;
12.     text-align:center;background-color:rgb(48,50,51);height:30;
13. }
14. .ul05 li{margin-left:0px;
15.     float: left;
16.     width: 500px;
17.     display:inline; height:30;
18. }
19. .ul05 a{
20.     font-size: 20px;
21.     font-weight: 700;color:white;
22.     text-decoration: none;
23. }
24. </style>
25. <BODY>
26. <ul class="ul05">
27. <li style="background-color:rgb(48,50,51);">
28. <A style="text-decoration: none;">Plane</A>
29. </li>
30. </ul>
31. <br>
32.
33. <div class="kk1">
34.
35. <div class="newBox">
36. <ul>
```



```
37.
38. <li>
39. <a>
40. <form action="showDetail.jsp" method="post" name="form1" id="xijie">
41. <input type='hidden' name='xijie' value="plane1001">
42. 
43. <br>plane1001</a><br>
44. <span class="lj_h1"><a>大疆（DJI）无人机 御 Mavic Air 便携可折叠 4K 超清航拍 旅
    行无人机（曜石黑）</a></span><br>
45.     ¥4999.00
46.     </form>
47. </li>
48.
49. <li>
50. <a>
51. <form action="showDetail.jsp" method="post" name="form2" id="xijie">
52. <input type='hidden' name='xijie' value="plane1002">
53. 
54. <br>plane1002</a><br>
55. <span class="lj_h1"><a>大疆（DJI）无人机 晓 Spark 掌上智能高清航拍无人机 自拍神
    器（白色）</a></span><br>
56.     ¥2599.00
57.     </form>
58. </li>
59.
60. <li>
61. <a>
62. <form action="showDetail.jsp" method="post" name="form3" id="xijie">
63. <input type='hidden' name='xijie' value="plane1003">
64. 
65. <br>plane1003</a><br>
66. <span class="lj_h1"><a>特洛（Tello） 无人机 遥控飞机 益智无人机 小型迷你 高清航
    拍无人机</a></span><br>
67.     ¥699.00
68.     </form>
69. </li>
70.
71. </ul>
72. </div>
73. </div>
74.
```

```
75. <br>
76.
77. <ul class="ul05">
78. <li style="background-color:rgb(48,50,51);">
79. <a style="text-decoration: none;">Hand</a>
80. </li>
81. </ul>
82.
83. <br>
84. <div class="kk1">
85. <div class="newBox">
86. <ul>
87. <li>
88. <a>
89. <form action="showDetail.jsp" method="post" name="form4" id="xijie">
90. <input type='hidden' name='xijie' value="hand1001">
91. 
92. <br>hand1001</a><br>
93. <span class="lj_h1"><a>大疆（DJI）云台 如影 Ronin-MX 三轴手持云台系统 专用摄影
    摄像器材</a></span><br>
94. ¥8699.00
95. </form>
96. </li>
97.
98. <li>
99. <a>
100. <form action="showDetail.jsp" method="post" name="form5" id="xijie">
101. <input type='hidden' name='xijie' value="hand1002">
102. 
103. <br>hand1002</a><br>
104. <span class="lj_h1"><a>大疆（DJI）云台 如影 Ronin 2 三轴手持摄影云台 专业稳定
    器 手持套装</a></span><br>
105. ¥32999.00
106. </form>
107. </li>
108.
109. </ul>
110. </div>
111. </div>
112.
113. <br>
114.
```

```
115. <ul class="ul05">
116. <li style="background-color:rgb(48,50,51);">
117. <A style="text-decoration: none;">Other</A>
118. </li>
119. </ul>
120.
121. <br>
122. <div class="kk1">
123. <div class="newBox">
124. <ul>
125. <li>
126. <a>
127. <form action="showDetail.jsp" method="post" name="form6" id="xijie">
128. <input type='hidden' name='xijie' value="other1001">
129. 
130. <br>other1001</a><br>
131. <span class="lj_h1"><a>大疆（DJI）无人机 Goggles 飞行眼镜 竞速版 FPV 智能体感
    控制 & 多功能双肩包 套装</a></span><br>
132. ¥4198.00
133. </form>
134. </li>
135.
136. </ul>
137. </div>
138. </div>
139.
140.
141. <script type="text/javascript">
142. function SendForm()
143. {
144.     document.getElementsByName("form")[0].sumbit;
145.     alert("1");
146. }
147.
148. </script>
149.
150. <script language="javascript">
151. function fsubmit(obj){
152. obj.submit();
153. }
154. </script>
155. <%@ include file="back.jsp"%>
156. </BODY>
```

157. </HTML>

lookOrderForm.jsp:

```

1. <%@ page contentType="text/html; charset=GB2312" %>
2. <jsp:useBean id="loginBean" class="mybean.data.Login" scope="session" />
3. <%@ page import="java.sql.*" %>
4. <HTML><HEAD><%@ include file="head.jsp" %></HEAD>
5. <BODY background=image/back.jpg>
6.     <div align="center">
7.         <% if (loginBean == null) {
8.             response.sendRedirect("login.jsp");//重定向到登陆页面
9.         }
10.        else {
11.            boolean b = loginBean.getLogname() == null ||
12.            loginBean.getLogname().length() == 0;
13.            if (b)
14.                response.sendRedirect("login.jsp");//重定向到登陆页面
15.        }
16.        Connection con;
17.        Statement sql;
18.        ResultSet rs;
19.        try { Class.forName("com.mysql.jdbc.Driver");
20.        }
21.        catch (Exception e) {}
22.        try { String url = "jdbc:mysql://127.0.0.1/web";
23.            String user = "root";
24.            String password = "123456";
25.            con = DriverManager.getConnection(url, user, password);
26.            sql = con.createStatement();
27.            String cdn = "SELECT id,mess,sum FROM orderform where lognam
e='"+ loginBean.getLogname() + "'";
28.            System.out.println(cdn);
29.            rs = sql.executeQuery(cdn);
30.            out.print("<table border=2>");
31.            out.print("<tr>");
32.                out.print("<th width=100>" + "订单号");
33.                out.print("<th width=100>" + "信息");
34.                out.print("<th width=100>" + "价格");
35.            out.print("</TR>");
36.            while (rs.next()) {
37.                out.print("<tr>");
38.                    out.print("<td>" + rs.getString("id") + "</td>");
39.                    out.print("<td>" + rs.getString("mess") + "</td>");
40.                    out.print("<td>" + rs.getString("sum") + "</td>");

```

```
41.         out.print("</tr>");
42.     }
43.     out.print("</table>");
44.     con.close();
45. }
46. catch(SQLException e) {
47.     out.print(e);
48. }
49. %>
50. </div>
51. <%@ include file="back.jsp"%>
52. </BODY>
53. </HTML>
```

QueryAllRecord.java:

```
1. package myservlet.control;
2.
3. import java.io.IOException;
4. import java.sql.Connection;
5. import java.sql.DriverManager;
6. import java.sql.ResultSet;
7. import java.sql.SQLException;
8. import java.sql.Statement;
9.
10. import javax.servlet.ServletConfig;
11. import javax.servlet.ServletException;
12. import javax.servlet.http.HttpServlet;
13. import javax.servlet.http.HttpServletRequest;
14. import javax.servlet.http.HttpServletResponse;
15. import javax.servlet.http.HttpSession;
16.
17. import com.sun.rowset.*;
18.
19. import mybean.data.DataByPage;
20.
21. public class QueryAllRecord extends HttpServlet {
22.     CachedRowSetImpl rowSet = null;
23.
24.     public void init(ServletConfig config) throws ServletException {
25.         super.init(config);
26.         try {
27.             Class.forName("com.mysql.jdbc.Driver");
28.         } catch (Exception e) {
29.             System.out.println("111"+e);
```

```
30.     }
31. }
32.
33. public void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
34.     request.setCharacterEncoding("gb2312");
35.     String idNumber = request.getParameter("fenleiNumber");
36.     if (idNumber == null)
37.         idNumber = "0";
38.     int id = Integer.parseInt(idNumber);
39.     HttpSession session = request.getSession(true);
40.     Connection con = null;
41.     DataByPage dataBean = null;
42.     try {
43.         dataBean = (DataByPage) session.getAttribute("dataBean");
44.         if (dataBean == null) {
45.             dataBean = new DataByPage();
46.             session.setAttribute("dataBean", dataBean);
47.         }
48.     } catch (Exception exp) {
49.         dataBean = new DataByPage();
50.         session.setAttribute("dataBean", dataBean);
51.     }
52.     String url = "jdbc:mysql://127.0.0.1/web";
53.     try {
54.         Class.forName("com.mysql.jdbc.Driver");
55.         con = DriverManager.getConnection(url, "root", "123456");
56.         Statement sql = con
57.             .createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,
58.                 ResultSet.CONCUR_READ_ONLY);
59.         ResultSet rs = sql
60.             .executeQuery("SELECT * FROM cosmeticform where id like
61.                 " + id);
62.         rowSet = new CachedRowSetImpl();
63.         rowSet.populate(rs);
64.         dataBean.setRowSet(rowSet);
65.         con.close();
66.     } catch (SQLException exp) {
67.         System.out.println(exp);
68.     } catch (ClassNotFoundException e) {
69.         // TODO Auto-generated catch block
70.         e.printStackTrace();
71.     }
```

```
72.         response.sendRedirect("byPageShow.jsp");
73.     }
74.
75.     public void doGet(HttpServletRequest request, HttpServletResponse response)
76.         throws ServletException, IOException {
77.         doPost(request, response);
78.     }
79. }
```

handleLogin.java:

```
1.  package myservlet.control;
2.
3.  import java.io.IOException;
4.  import java.io.PrintWriter;
5.  import java.sql.Connection;
6.  import java.sql.DriverManager;
7.  import java.sql.ResultSet;
8.  import java.sql.SQLException;
9.  import java.sql.Statement;
10.
11. import javax.servlet.RequestDispatcher;
12. import javax.servlet.ServletConfig;
13. import javax.servlet.ServletException;
14. import javax.servlet.http.HttpServlet;
15. import javax.servlet.http.HttpServletRequest;
16. import javax.servlet.http.HttpServletResponse;
17. import javax.servlet.http.HttpSession;
18.
19. import mybean.data.Login;
20.
21. public class HandleLogin extends HttpServlet {
22.     public void init(ServletConfig config) throws ServletException {
23.         super.init(config);
24.         try {
25.             Class.forName("com.mysql.jdbc.Driver");
26.         } catch (Exception e) {}
27.     }
28.
29.     public String handleString(String s) {
30.         try {
31.             byte bb[ ] = s.getBytes("iso-8859-1");
32.             s = new String(bb);
33.         }
```

```
34.         catch (Exception ee) {}
35.         return s;
36.     }
37.
38.     public void doPost(HttpServletRequest request, HttpServletResponse response)
        throws ServletException, IOException {
39.         Connection con;
40.         Statement sql;
41.         String logname = request.getParameter("logname").trim();
42.         String password = request.getParameter("password").trim();
43.         logname = handleString(logname);
44.         password = handleString(password);
45.         String url = "jdbc:mysql://127.0.0.1/web?"
46.             + "user=root&password=123456&characterEncoding=gb2312";
47.         boolean boo = (logname.length() > 0) && (password.length() > 0);
48.
49.         try {
50.             con = DriverManager.getConnection(url);
51.             String condition = "select * from user where logname = '" + logname
52.                 + "' and password= '" + password + "'";
53.             sql = con.createStatement();
54.             if (boo) {
55.                 ResultSet rs = sql.executeQuery(condition);
56.                 boolean m = rs.next();
57.                 if (m == true) {
58.                     // 调用登录成功的办法
59.                     success(request, response, logname, password);
60.                     RequestDispatcher dispatcher = request
61.                         .getRequestDispatcher("loginshow.jsp");
62.                     dispatcher.forward(request, response);
63.                 } else {
64.                     String backNews = "您输入的用户名不存在, 或者密码不正确";
65.                     fail(request, response, logname, backNews);
66.                 }
67.             } else {
68.                 String backNews = "请输入用户名和密码";
69.                 fail(request, response, logname, backNews);
70.             }
71.         } catch (SQLException exp) {
72.             String backNews = "" + exp;
73.             fail(request, response, logname, backNews);
74.         }
75.     }
```



```
76.     }
77. }
78.
79. public void doGet(HttpServletRequest request, HttpServletResponse response)
80.     throws ServletException, IOException {
81.     doPost(request, response);
82. }
83.
84. public void success(HttpServletRequest request,
85.     HttpServletResponse response, String logname, String password) {
86.     Login loginBean = null;
87.     HttpSession session = request.getSession(true);
88.     try {
89.         loginBean = (Login) session.getAttribute("loginBean");
90.         if (loginBean == null) {
91.             loginBean = new Login();
92.             session.setAttribute("loginBean", loginBean);
93.             loginBean = (Login) session.getAttribute("loginBean");
94.         }
95.         String name = loginBean.getLogname();
96.         if (name.equals(logname)) {
97.
98.             loginBean.setBackNews(logname + "已经登录了");
99.             loginBean.setLogname(logname);
100.        }else{
101.            loginBean.setBackNews(logname+"登录成功");
102.            loginBean.setLogname(logname);
103.        }
104.    } catch (Exception ee) {
105.        loginBean = new Login();
106.        session.setAttribute("loginBean", loginBean);
107.        loginBean.setBackNews(logname + "登录成功");
108.        loginBean.setLogname(logname);
109.    }
110. }
111.
112. public void fail(HttpServletRequest request, HttpServletResponse response,
113.     String logname, String backNews) {
114.     response.setContentType("text/html; charset=UTF-8");
115.     try {
116.         PrintWriter out = response.getWriter();
```

```
117.         out.println("<html><body>");
118.         out.println("<h2>" + logname + " logname Or password error<br>"
    + "</h2>");
119.         out.println("<a href=login.jsp>login.jsp</a><br>");
120.         out.println("<a href=index.jsp>index.jsp</a>");
121.         out.println("</body></html>");
122.     } catch (IOException exp) {
123.     }
124. }
125. }
```

## 4.2 实现效果



图 1 首页效果图

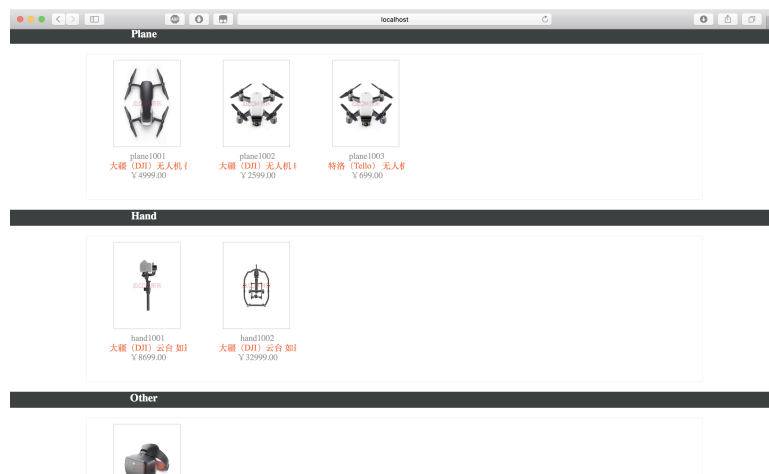


图 2 浏览商品效果图



图 3 查询商品效果图



图 4 查询订单效果图

# 个人项目总结

自我评定表					
班级	10041612	姓名	白云鹏	学号	1004161221
项目名称	基于 Servlet 的网上商城		项目成员	白云鹏	
完成情况	<input checked="" type="checkbox"/> 全部完成 <input type="checkbox"/> 大部分完 <input type="checkbox"/> 少部分完成 <input type="checkbox"/> 没有思路		自我评价	良好	
代码规范度	1.在代码的缩进方面，部分 JSP 代码在标签的缩进上有所欠缺。 2.在类的结构上，耦合性比较强，后端中没有将数据库操作抽出集成。 3.在类的命名上则比较规范，基本上都使用了驼峰命名法。 4.在包的封装上，将实体类与控制类分离。 5.异常捕捉上比较完善。				
项目中遇到的主要问题及解决办法	1.在学习过程中，很多操作都是基于 windows 操作系统而言的，而在开发的过程中则使用 mac os 系统，由于系统的差异，很多问题都需要自力解决，比如在 mysql 的配置上，不同的操作系统在安装方法、配置方式上都有所区别。这时只能通过搜索引擎进行相关查找。 2.在编写代码的过程中，有很多函数的使用方法，其功能都不了解，只能通过查询函数或者相关功能进行尝试。 3.在数据库的操作上，虽然基本语句都已经学习，但是在实际使用的过程中仍有所遗漏，在使用的时候不断复习所学的语句。 4.在项目构建方面，在开始时配置 tomcat 失败，通过询问同学，进行搜索解决。 5.在 JDBC 连接数据库时，由于版本的不同导致连接失败，首先通过修改语句判断编写是否有误，最后将问题定位在版本上，更新版本后解决。				
项目体会	1.在学习的过程中，不能轻易对待看似简单的问题，具体到代码实现上可能会出现意想不到的困难。 2.在开发的过程中，遇到困难不能主观臆断，要根据具体的问题进行分析，不断缩小问题可能出现的范围，准确定位问题。 3.在开发的过程中，不能只以为孤军奋斗，往往需要总结前人的经验教训或者向同学进行请教，快速解决问题才是目的所在。 4.养成良好的调试习惯，比如在前端的开发中，需要不断的清除缓存来保证页面的实时刷新。 5.编写代码的过程中要留有注释，否则重新阅读时会造成很大的障碍。 6.将不同的操作进行分离，相似的操作进行整合，减少系统的耦合性，增强稳定性。				
自我评定	1.在编写代码的过程中，遇到的基本问题能够独立进行解决，困难的问题能够通过搜索或者询问得到解决，保证项目的完成。我认为这是值得保持的。 2.在开发过程中，由于系统的耦合，导致代码较为脆弱，总是需要修改多个文件，这是值得改进的地方。 3.在测试过程中，没有固定的测试集导致测试效率较低。 4.在代码的编写规范上有所进步。				

