



湖南石油化工职业技术学院
Hunan Petrochemical Vocational Technology College

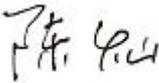
学生毕业设计成果说明书

设计题目： 基于 Web 的购物系统
专业名称： 移动应用开发
班级名称： 移动应用开发 3171
学生姓名： 陈灿
指导教师： 夏维
责任领导： 何飞红

二零二零年五月

学生毕业设计成果说明书真实性承诺书

本人郑重承诺：我所递交的毕业设计材料，是本人在指导老师的指导下独立进行完成的；除文中已经注明引用的内容外，不存在有作品（产品）剽窃和抄袭他人成果的行为。对本设计的共同完成人所做出的贡献，在对应位置已以明确方式标明。若被查出有抄袭或剽窃行为，或由此所引起的法律责任，本人愿意承担一切后果。

学生（确认签字）： 签字日期：2020-5-20

指导教师关于学生毕业设计成果说明书真实性审核承诺书

本人郑重承诺：已对该生递交的毕业设计材料中所涉及的内容进行了仔细严格的审核，其成果是本人在的指导下独立进行完成的；对他人成果的引用和共同完成人所做出的贡献在对应位置已以明确方式标明。不存在有作品（产品）剽窃和抄袭他人成果的行为。若查出该生所递交的材料有学术不端的行为，或由此所引起的法律责任，本人愿意承担一切责任。

指导教师（确认签字）： 签字日期：2020-5-24

目 录

| | |
|--------------------|----|
| 一、成果简介..... | 1 |
| (一) 设计背景..... | 1 |
| (二) 设计意义..... | 1 |
| 二、设计思路..... | 1 |
| (一) 系统目标及系统功能..... | 1 |
| (二) 功能设计..... | 2 |
| 1、系统功能流程..... | 2 |
| 2、用户登入..... | 2 |
| 3、用户注册..... | 3 |
| 4、便利店主页..... | 4 |
| 5、添加商品页..... | 4 |
| 6、个人中心页..... | 5 |
| 三、设计过程..... | 6 |
| (一) 数据库设计..... | 6 |
| 1、用户信息表..... | 6 |
| 2、商品信息表..... | 6 |
| 3、订单信息表..... | 7 |
| 4、商品分类表..... | 8 |
| (二) 系统模块设计..... | 8 |
| 1、登入界面..... | 8 |
| 2、注册界面..... | 9 |
| 3、首页界面..... | 10 |
| 4、添加界面..... | 14 |
| 5、商品详情页..... | 16 |
| 6、个人中心页..... | 18 |
| 四、成果特点..... | 23 |
| 五、收获与体会..... | 24 |
| (一) 收获..... | 24 |

| | |
|--------------|----|
| (二) 体会 | 25 |
| 参考文献 | 27 |

基于 web 开发的购物系统

一、成果简介

(一) 设计背景

随着互联网的发展，网络购物、电商已成为家喻户晓的平台，这使得人民在家足不出户就可以购买到自己心怡产品，以前只能通过电视上的广告或者电话才能购买而且种类有限，因此随着电商互联网的发展，提高了人民的工作效应。

电子商务指利用简单、快捷、低成本电子通讯方式，买卖双方通过网络进行各种商贸活动的一种商业交易模式。随着互联网的发展，以及人们消费观念和生活方式的改变，这种新型的商业模式逐渐融入了人们的生活中。通过企业的门户网站，人们可以足不出户的寻找自己所需的物品，通过对不同虚拟商城的访问，“顾客”可以找出性价比最高的商品，自己有极大地选择空间；通过下订单，你可以很快得到自己想要的物品，这给顾客节省很多时间和精力；对于厂家来说，可以极大地降低库存风险，可根据客户需要，按需采购和生产，大大地提高了效益。

(二) 设计意义

为了提高网上购物中网店与顾客之间的互动,与移动平台相结合,设计了基于 Web Service 的网上购物系统。该系统以 Mysql 为数据库服务器,以 Mybatis、spring、springmvc 框架为设计模式,在 Java EE 应用中发布和使用 Web 服务,从客户确认网购到客户接收商品过程中的每一个环节,都利用网络环境实现,考虑到终端客户购物的多样性,实现了在线购物和查看商品详情、删除商品等功能,该购物系统具有良好的互动性、灵活性和扩展性,满足了网上购物的需求。

二、设计思路

(一) 系统目标及系统功能

本系统的主要目标是实现商家在线管理商品,用户查看商品。这样方便用户的购买,可以让用户进行购物的一系列操作,比传统的购买模式更加方便。

下面分别介绍系统功能:

用户登入：用户可以在线登入。

用户注册：用户在线注册。

便利店主页：查看所有商品，商品分类等。

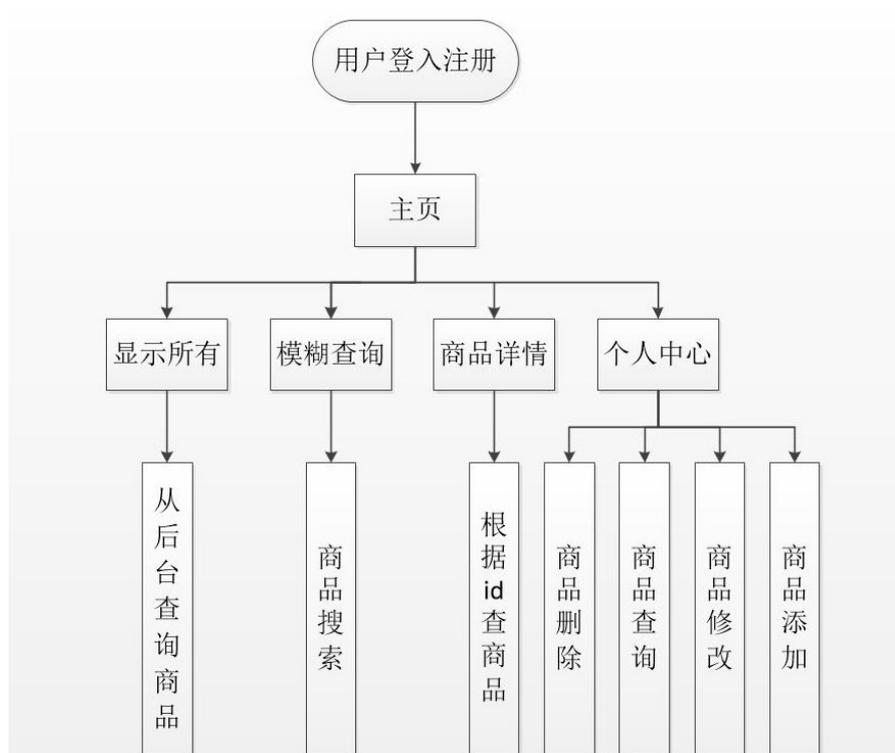
商品详情页：用户点击商品跳转至详细介绍。

添加商品页：实现用户添加商品。

个人中心页：实现删除、修改、查询商品。

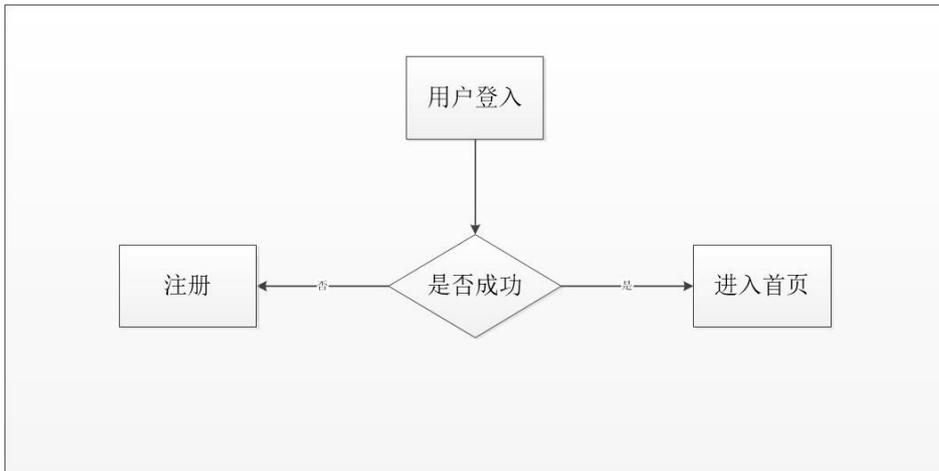
(二) 功能设计

1、系统功能流程



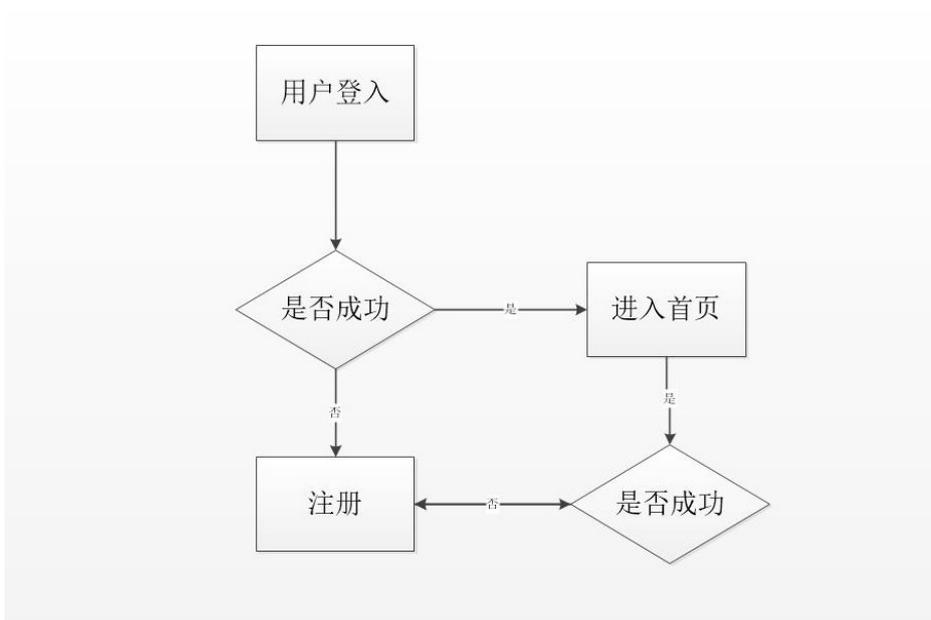
2、用户登入

用户输入手机号和密码后，后台就会进行查询，如果有就会进入首页，没有就会去注册。



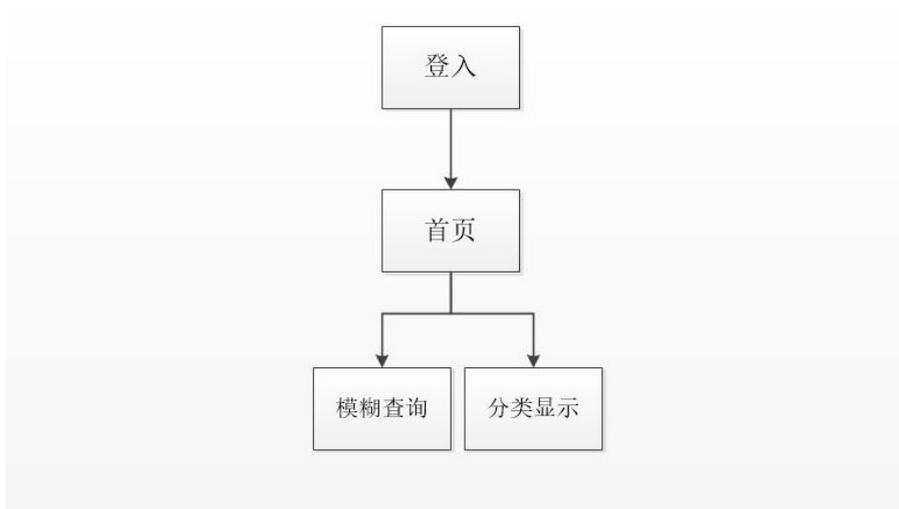
3、用户注册

当用户登入不成功或点击注册时，用户输入该输入的信息实现注册，后台进行添加用户信息。



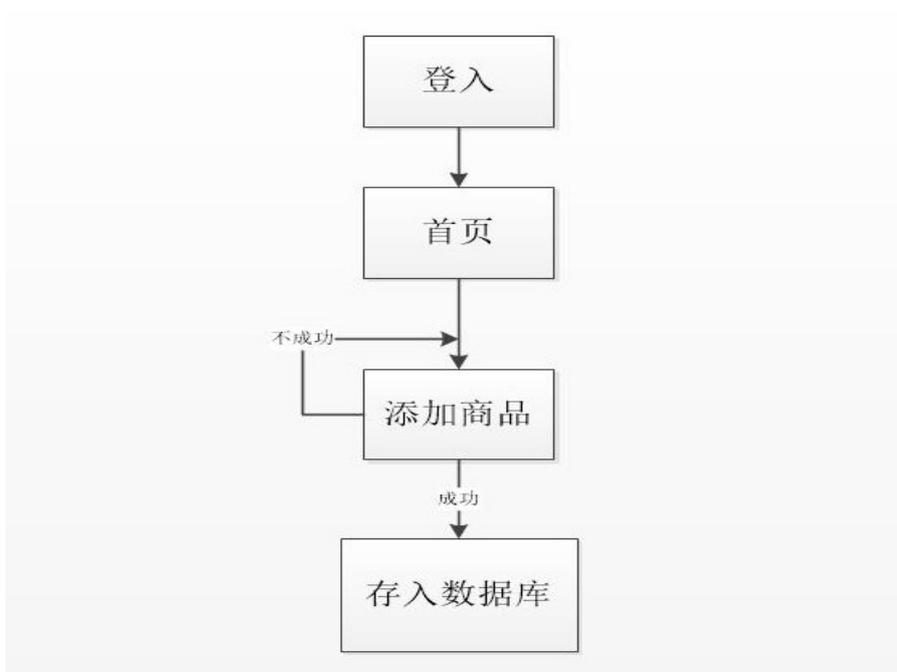
4、便利店主页

- 1) 在用户登入成功进入首页时，就可以看到所有商品，用户可以点击商品查看。
- 2) 用户在首页可以进行模糊查询，查询想要的商品。
- 3) 主页设有分类显示。



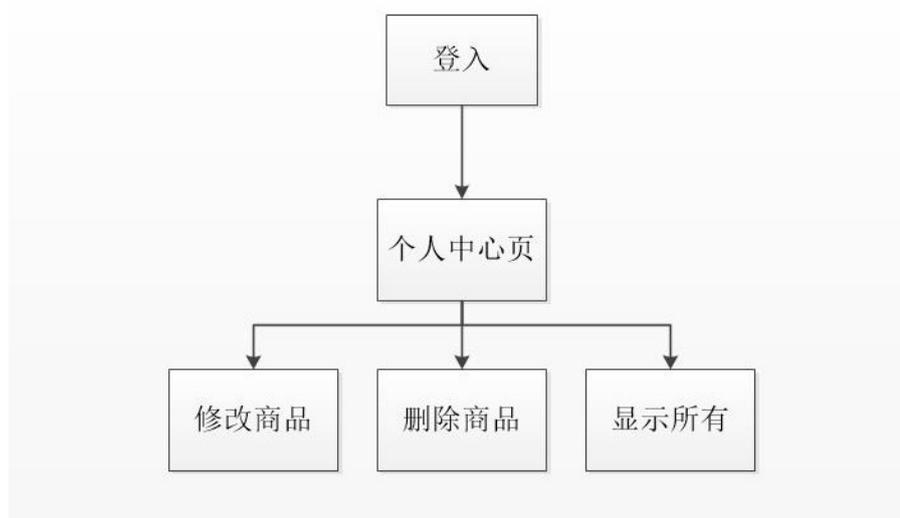
5、添加商品页

实现添加商品功能，把需要展示的商品添加进数据库，在主页、个人中心页等页面进行展示。



6、个人中心页

- 1) 实现商品查所有功能。
- 2) 根据 id 修改商品。
- 3) 删除商品。



三、设计过程

(一) 数据库设计

随着互联网的发展，电商、快递等各行各业的转型，发展离不开互联网，这时候就需要一个类似储存卡的物品储存数据，而这个东西就是数据库，数据库设计中概念结构设计尤为重要，库概念结构设计中比较常用的方法是实体——联系方法，也就是 E-R 方法 (Entity-Relation Approach)。E-R 图是用来描述数据之间联系的有效方法，有实体、属性、联系 3 种基本成分。

本系统根据数据库设计原则，遵循数据设计规范化理论，按照数据设计的步骤进行详细设计。主要有商品表，商品详情表，用户表，订单表，用户表跟商品表存在一对多关系，商品跟商品详情存在一对一关系，商品详情与订单表存在一对多，订单表与用户表存在多对多的关系。

1、用户信息表

用户信息表主要存储用户基本信息，它包括用户名、用户密码、联系电话等信息。

用户信息表

| 字段名 | 中文名 | 字段类型 | 字段长度 | 是否为空 |
|-----------|-------|---------|------|----------|
| Id | 用户 id | Int | 11 | Not null |
| shoujihao | 手机号 | Int | 255 | |
| pwd | 用户密码 | Varchar | 255 | |

2、商品信息表

商品信息表主要存储商品信息，它包括商品 id、商品名称、商品图片、商品价格等，用户可以直接管理

商品信息表

| 字段名 | 中文名 | 字段类型 | 字段长度 | 是否为空 |
|---------|-------|---------|------|----------|
| Id | 商品 id | Int | 11 | Not null |
| Title | 商品名称 | Varchar | 255 | |
| Img | 商品图 | Varchar | 255 | |
| Zms | | Varchar | 255 | |
| Price | 商品价格 | Double | 10 | |
| H1 | | Varchar | 255 | |
| H2 | | Double | 255 | |
| Spxqing | 商品详情图 | Varchar | 255 | |
| Spfl | 商品分类 | Varchar | 255 | |

3、订单信息表

储存用户添加的订单，，它包括商品 id、价格、数量等。

订单信息表

| 字段名 | 中文名 | 字段类型 | 字段长度 | 是否为空 |
|-------|-------|---------|------|----------|
| Id | 订单 id | Int | | Not null |
| Price | 价格 | Int | 255 | |
| Title | 商品名称 | Varchar | 255 | |
| Count | 商品数量 | Int | 11 | |

4、商品分类表

商品分类表主要是存在关联关系，对商品信息表进行关联。

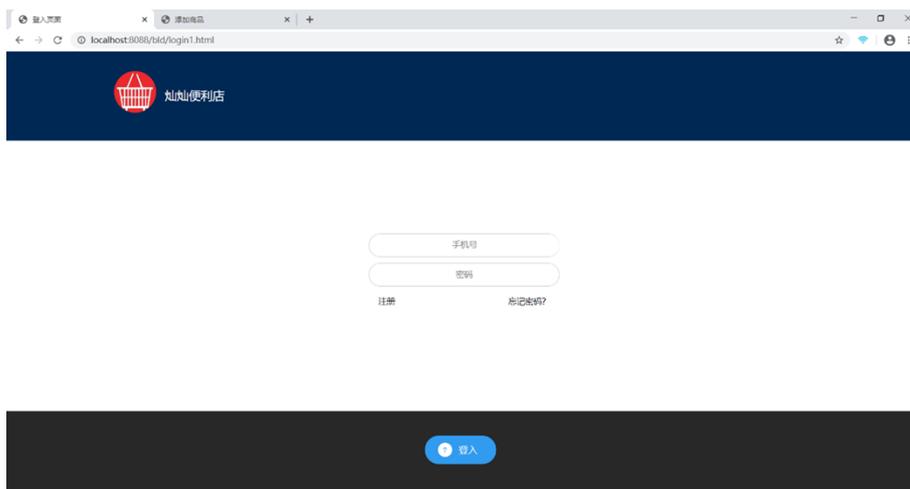
商品分类表

| 字段名 | 中文名 | 字段类型 | 字段长度 | 是否为空 |
|------|---------|---------|------|----------|
| Id | 商品分类 id | Int | | Not null |
| Spfl | 商品分类 | Varchar | 255 | |

(二) 系统模块设计

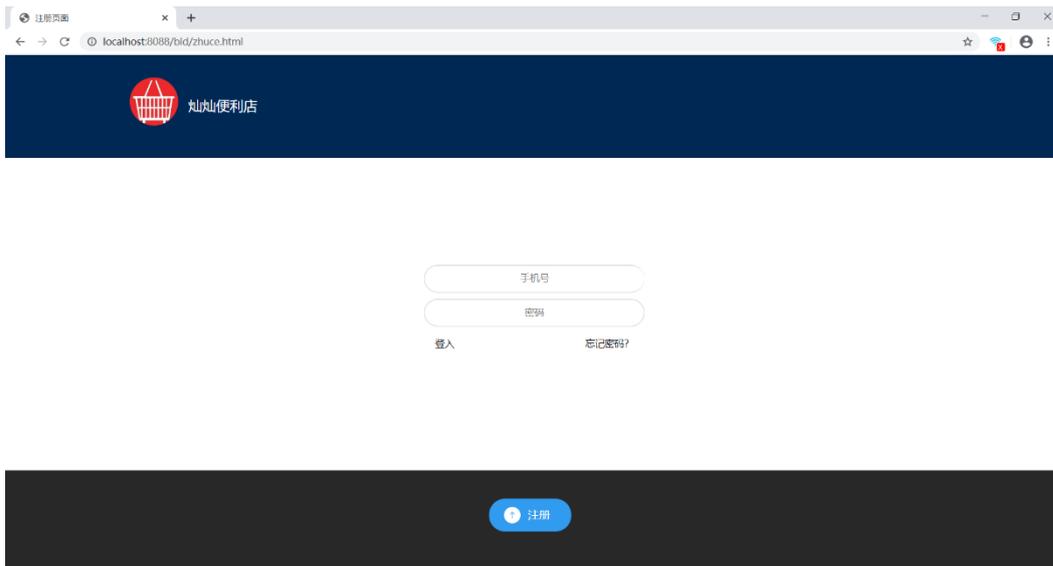
1、登入界面

用户进入首先看到的是登入界面，为方便进入系统，登入只用作交互，用户登入成功后进入首页，登入不成功说明数据库没有此用户，这是就会去注册界面进行注册。另外，登入界面有很好的交互动作登入按钮。



用户登入时，当用户输入手机号密码密码后点击登入，这样就会进入后台查询是否有此手机号的用户，利用@RequestMapping("/login")中的地址访问后台，调用后台的public String login(user u,HttpServletResponse hp)方法的user login = us.login(u);进行查询，利用 json 格式数据转发 hp.getWriter().print(new Gson().toJson("success"));若有则进入首页，没有则进入注册页面进行注册。

2、注册界面



用户登入不成功时则进入注册页面，用户输入该输入的信息，点击注册，利用@RequestMapping("/zhuce")的/zhuce地址访问后台执行方法public String zhuce(user u,HttpServletResponse hp)，在接口层写好的insert方法把前台页面传进来的对象放进去int i = us.insert(u);当这个i不等于1时则注册成功利用HttpServletResponse对象把 json 格式的数据返回到前台页面则进入首页，不成功就停留在注册页面。

3、首页界面

用户进入首页即显示所有商品和模糊查询，在页面的顶部会有零食、日用品、全部商品、酒水铺、个人中心、登入注册按钮以及搜索框样式，旁边有返回顶部、店铺首页、以及联系客服等功能。实现这些功能主要依靠一些css、js、jquery、bootstrap、其他插件等插件实现。



用户登入成功进入首页，ajax 通过@RequestMapping("/selectzy")访问后台里面的 public void selectzy(HttpServletRequestResponse hp)方法调用 shangpinService 的 selectAll 方法来查询所有商品，查询成功后通过 json 数据格式 hp.getWriter().print(new Gson().toJson(sa))发送到前台界面，主页 ajax 接收数据使用 jquery 的 html 元素进行展示到页面显示。

其主要代码如下：

- 1) @RequestMapping("/selectzy")ajax 的访问地址。
- 2) HttpServletResponse hp;解决中文乱码的问题。
- 3) List<shangpin> sa = ss.selectAll();利用 service 的 selectAll 方法进行查询所有商品。
- 4) new Gson()用 Gson 转发查询出的数据到页面进行显示。

5) 后端代码如下:

```
@RequestMapping("/selectzy")
public void selectzy(HttpServletResponse hp) {
    hp.setContentType("text/html");
    hp.setCharacterEncoding("utf-8");
    System.out.println("主页查询进来啦");
    List<shangpin> sa = ss.selectAll();
    try {
        hp.getWriter().print(new Gson().toJson(sa));
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

@RequestMapping("/chaolianjie")
public void chaolianjie(Model m,HttpServletResponse hp) {
    hp.setContentType("text/html");
    hp.setCharacterEncoding("utf-8");
    List<shangpin> list = ss.selectAll();
    System.out.println("超链接过来"+this.id);
    try {
        hp.getWriter().print(new Gson().toJson(list));
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

6) 前端代码如下:

```
$(function() {
    $("#div2").mouseover(function() {
        $("#div1").fadeIn();
    })
})
```

```

$("#div3").hover(function() {
    $("#div5").slideToggle();
})
$("#auto").click(function() {
    alert("123")
    $.ajax({
        url:"http://localhost:8088/bld/chaolianjie",
        type:"post",
        dataType:"json",
        success:function(data) {
            for (var i = 0; i < data.length; i++) {
                var id = data[i].id;
                var title = data[i].title;
                var price = data[i].price;
                var img = data[i].img;
                $("#image_"+i).attr("src",img);
                $("#title_"+i).html(title);
                $("#jiage_"+i).html(price);
                $("#div_span_"+i+" span").html(id);
            }
        },
        error:function(data) {
        }
    })
})
$("#auto").trigger("click");

```

在顶部，用户输入想要的商品在输入框，比如小米锅巴的锅巴两字，这时 ajax 异步请求访问后台方法将进行模糊查询，将查询到的所有锅巴两字的数据利用 Gson 转发到前台 ajax 里进行接收，再通过实时显示技术把查询到的商品名称在搜索框下的下拉框进行显示，当用户点击搜索出来的商品名称时就会进入商品详细界面来介绍商品。

其主要的代码如下：

- 1) @RequestMapping("/sosuo") 鼠标触控搜索框时进入的后台。
- 2) List<shangpin> list = ss.select(p); 模糊查询用户输入的数据。
- 3) new Gson() 转发到前台。
- 4) 前台使用 jquery 的 html 方法实现数据的显示

后台代码如下：

```
@RequestMapping("/sosuo")
public void select(shangpin p, HttpServletResponse hp) {
    hp.setContentType("text/html");
    hp.setCharacterEncoding("utf-8");
    System.out.println("模糊查询");
    List<shangpin> list = ss.select(p);
    try {
        hp.getWriter().print(new Gson().toJson(list));
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

前台代码如下：

```
$("#sos").mouseover(function() {
    $("#ol").css("display", "block");
    $("#ol span").html("");
    var ss=$("#souseo").val();
    $.ajax({
        url:"http://localhost:8088/bld/sosuo",
        type:"post",
        data:{"title":ss},
        dataType:"json",
        success:function(data) {
            for (var i = 0; i < data.length; i++) {
                var s=$("#s"+i).html();
                $("#s"+i).html(s+data[i].title);
            }
        },
        error:function(data) {
```

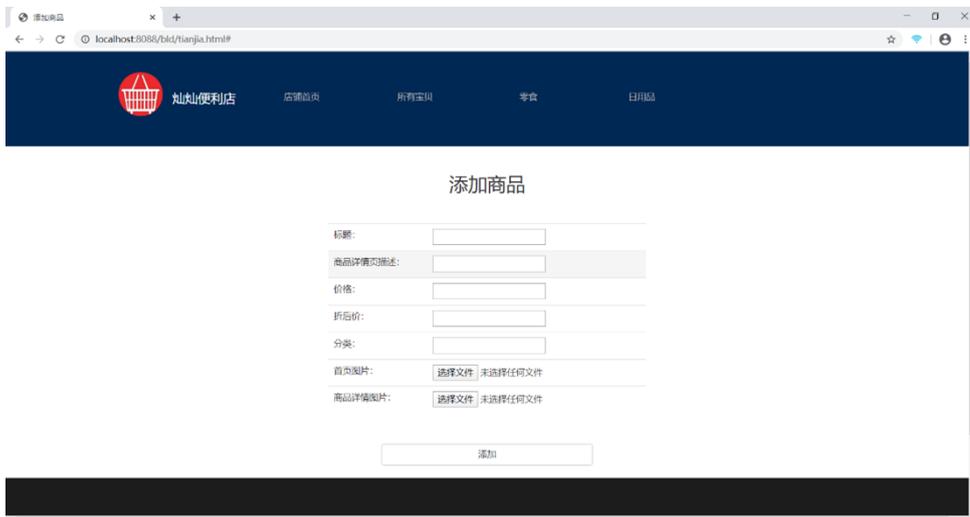
```

    }
  })
})
$("#sos").mouseout(function() {
    $("#ol").css("display", "none");
})
})

```

4、添加界面

用户登入成功后，进入首页页面进行显示所有商品，在个人中心处点击添加商品进入商品添加页，用户依次输入商品的标题，商品详细描述、价格、折后价、分类、图片等，点击添加按钮进行添加，在此 Ajax 就把数据传送到后台，使用 sql 的 insert 语句进行添加。



其主要代码如下：

1) @RequestMapping("/tianjia"); Ajax 访问的后台地址。

2) int i = ss.insert(s); 利用 service 的 insert 方法把数据传入。

3) new Gson().toJson(); 利用 Gson 技术返回给 Ajax。

4) 后端代码如下

```

@RequestMapping("/tianjia")
public String tianjia(shangpin s, HttpServletResponse hp) throws
IOException {
    hp.setContentType("text/html");
    hp.setCharacterEncoding("utf-8");
    System.out.println("添加进来了");
    int i = ss.insert(s);
}

```

```

        if (i!=0) {
            hp.getWriter().print(new Gson().toJson("添加成功"));
        }
        return null;
    }
}

```

5) 前端代码如下

```

<script type="text/javascript">
    $(function() {
        $("#btn").click(function() {
            var formdata=new FormData($("#from")[0]);
            var biaoti1 = $("#biaoti").val();
            var spxq1 = $("#spxq").val();
            var jiage1 = $("#jiage").val();
            var zhehoujial = $("#zhouhoujia").val();
            var fenlei1 = $("#fenlei").val();
            var shouyetup1 = $("#shouyetup").val();
            var spxqtup1 = $("#spxqtup").val();
            $.ajax({
                url:"http://localhost:8088/bld/tianjia",
                dataType:"json",
                type:"post",

                data:{"title":biaoti1,"h1":spxq1,"h2":jiage1,"price":zhehoujial,

                "spfl":fenlei1,"shouyetup":shouyetup1,"spxqtup":spxqtup1,formdata},
                cache: false,
                processData: false,
                contentType: false,
                success:function(data) {
                    for (var i = 0; i < data.length; i++) {
                        alert(data);
                    }
                },
                error:function(data) {
                    alert("添加失败")
                }
            })
        })
    })
</script>

```

在页面的交互就是是否添加成功使用 alert 弹窗 rang 用户知道这个商品是否添

加成功，成功则弹窗添加成功，不成功则弹窗添加不成功

5、商品详情页

用户登入成功后进入首页，呈现在眼前是所有商品，当用户点击某个商品时，Ajax把商品的id传入后台进行查询，根据id查询，查询出来在详情页面进行展示。



其代码如下所示：

1) 后台代码：

```
@RequestMapping("/selects")
public void selects(HttpServletRequestResponse hp) {
    hp.setContentType("text/html");
    hp.setCharacterEncoding("utf-8");
    System.out.println("商品详情");
    System.out.println(id);
    HashMap<String, Integer> h = new HashMap<String, Integer>();
    h.put("hashid", this.id);
    List<shangpin> list = ss.selectId(h);
    for (shangpin sp : list) {
        System.out.println(sp.getTitle());
    }
}
```

```

try {
    hp.getWriter().print(new Gson().toJson(list));
} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
}

```

2) 前端代码

```

<script type="text/javascript">
    $(function() {
        $("#auto").click(function() {
            $.ajax({
                url:"http://localhost:8088/bld/selects",
                dataType:'json', //服务器返回json格式数据
                type:'post', //HTTP请求类型
                success:function(data) {
                    for (var i = 0; i < data.length; i++) {
                        var id = data[i].id;
                        var h1 = data[i].h1;
                        var h2 = data[i].h2;
                        var price = data[i].price;
                        var img = data[i].img;
                        var spxqimg = data[i].spxqimg;
                        $("#title").html(h1);
                        $("#img").attr("src",img);
                        $("#price").html(price);
                        $("#h2").html(h2);
                        $("#spxqimg").attr("src",spxqimg);
                    }
                },
                error:function(data) {
                }
            }
        )
    }
}

```

```

    });
});
$("#auto").trigger("click");
});
</script>

```

6、个人中心页

用户登入成功后，在首页可以点击个人中心进入个人中心页显示，当进入中心页时，后台进行查询所有商品，因为这个页面是管理页，可以对所有商品进行修改和删除，用户点击某个商品旁边的删除按钮则访问后台的删除商品方法，通过 ajax 把需要删除商品的 id 传入后台。



把 id 传入后台后调用删除方法，之后再吧值传进 xml 文件中的 sql 语句进行删除基础操作。

其代码如下所示：

后台代码：

```

@RequestMapping("/deleteone")
public void deletesp(Integer id, HttpServletResponse hp) {
    hp.setContentType("text/html");
}

```

```

hp.setCharacterEncoding("utf-8");
System.out.println("删除"+id);
int i = ss.deletesp(id);
try {
    if (i!=0) {
        hp.getWriter().print(new Gson().toJson("删除成功"));
    }else {
        hp.getWriter().print(new Gson().toJson("删除失败"));
    }

} catch (IOException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}

}

```

前台代码:

```

<script type="text/javascript">
    $(function() {
        $("#auto").click(function() {
            $.ajax({
                url:"http://localhost:8088/bld/chaolianjie",
                type:"post",
                dataType:"json",
                success:function(data) {
                    for (var i = 0; i < data.length; i++) {
                        var id = data[i].id;
                        var title = data[i].title;
                        var price = data[i].price;
                        var img = data[i].img;
                        $("#image_"+i).attr("src",img);
                    }
                }
            });
        });
    });

```

```

        $("#title_"+i).html(title);
        $("#jiage_"+i).html(price);
        $("#div_span_"+i+" span").html(id);
        $("#xianshi").empty();
        var html = '<tr>
            +<td id="id" style="text-align:
center;">#{id}</td>
            +<td style="text-align:
center;">#{title}</td>
            +<td style="text-align:
center;">#{jiage}</td>
            +<td style="text-align: center;"><a
class="btn btn-default" href="#">修改</a>&nbsp;&nbsp;&nbsp;<a id="shanchu"
onclick="shanchu()" class="btn btn-danger" href="#">删除</a></td>
            +</tr>';
        html = html.replace("#{id}", data[i].id);
        html =
html.replace("#{title}", data[i].title);
        html =
html.replace("#{jiage}", data[i].price);
        $("#xianshi").append(html);
    }
},
    error:function(data) {
    }
})
})
$("#auto").trigger("click");

})
function shanchu() {
    var msg="确定要删除吗? ";
    var id=$("#id").html();

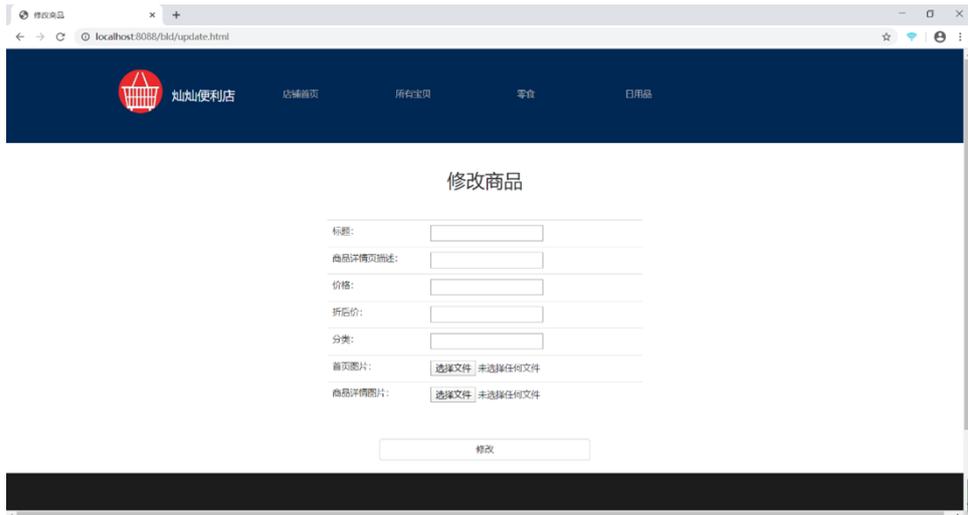
```

```

if(confirm(msg)) {
    //对话框确认执行的操作
    var name=$(this).prev();
    var newname=name.html();
    alert(id+newnamenm);
    $.ajax({
        url:"http://localhost:8088/bld/deleteone",
        dataType:"json",
        data:{"id":id},
        type:"post",
        success:function(data) {
            window.location.href="grzx.html";
        },
        error:function(data) {
        }
    })
} else {
    //反之操作
}
}
</script>

```

当然，用户选择修改时也是同一种道理，把 id 传入后台进行查询，利用 ssm 框架把查询到的值保存，再进入修改页面，用户依次输入商品的标题，商品详细描述、价格、折后价、分类、图片等，点击修改，这时再进行后台的访问，访问 uodate 方法，进入 xml 文件把用 sql 的 update 语句进行更新修改。



前台代码如下:

```

<script type="text/javascript">
    $(function() {
        $("#btn").click(function() {
            var formdata=new FormData($("#form")[0]);
            var biaoti1 = $("#biaoti").val();
            var spxq1 = $("#spxq").val();
            var jiage1 = $("#jiage").val();
            var zhehoujial = $("#zhehoujia").val();
            var fenlei1 = $("#fenlei").val();
            var shouyetup1 = $("#shouyetup").val();
            var spxqtup1 = $("#spxqtup").val();
            $.ajax({
                url:"http://localhost:8088/bld/update",
                dataType:"json",
                type:"post",

                data:{"title":biaoti1,"h1":spxq1,"h2":jiage1,"price":zhehoujial,

                "spfl":fenlei1,"shouyetup":shouyetup1,"spxqtup":spxqtup1,formdata},
                cache: false,
                processData: false,
                contentType: false,
            });
        });
    });

```

```

        success:function(data) {
            for (var i = 0; i < data.length; i++) {
                alert(data);
            }
        },
        error:function(data) {
            alert("修改失败")
        }
    })
})
})
</script>

```

后台代码如下：

```

@RequestMapping("/selects")
public void selectids(HttpServletRequestResponse hp) {
    hp.setContentType("text/html");
    hp.setCharacterEncoding("utf-8");
    System.out.println("商品详情");
    System.out.println(id);
    HashMap<String, Integer> h = new HashMap<String, Integer>();
    h.put("hashid", this.id);
    List<shangpin> list = ss.selectId(h);
    for (shangpin sp : list) {
        System.out.println(sp.getTitle());
    }
    try {
        hp.getWriter().print(new Gson().toJson(list));
    } catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```

```

    }
    @RequestMapping("/selects")
    public void update(shangpin s,HttpServletResponse hp) throws IOException
    {
        hp.setContentType("text/html");
        hp.setCharacterEncoding("utf-8");
        System.out.println("添加进来了");
        int i = ss.update(s);
        if (i!=0) {
            hp.getWriter().print(new Gson().toJson("修改成功"));
        }
    }
}

```

四、成果特点

互联网始于 1969 年美国的阿帕网。通常 internet 泛指互联网，而 internet 则特指因特网。这种将计算机网络互相联接在一起的方法可称作“网络互联”，在这基础上发展出覆盖全世界的全球性互联网络称互联网，即是互相连接一起的网络结构。互联网并不等同万维网，万维网只是一建基于超文本相互链接而成的全球性系统，且是互联网所能提供的服务其中之一。

Java 是由 Sun Microsystems 公司于 1995 年 5 月推出的 Java 面向对象程序设计和 Java 平台的总称。由 James Gosling 和同事们共同研发，并在 1995 年正式推出，经过二十多年的发展，Java 继承了许多框架，spring mybatis 等，Java web 的出现是可以用纯 Servlet 开发和 jsp 开发，架构与 mvc 总共有三层如下、业务逻辑层、数据持久层、表现层，而 spring 的出现主要是用来替代更加重量级的企业级技术，提供了更加轻量级和简单的编程模型，它增强了老式的 Java 对象 pojo 的功能，使得 spring 具备企业级 Java 规范才有的功能。

本购物系统使用 ssm 框架就是 spring、springmvc、mybatis 框架进行整合完成开发的，简单的设计主要体现在方便上，使得用户直接注册登入进入商品显示，用户自己也是卖家，使用的 Ajax 是一种独立于 Web 服务器软件的浏览器技术，允许客

客户端脚本发送 HTTP 请求 (XMLHTTP)，使用 Ajax 的最大优点，就是能在不更新整个页面的前提下维护数据。Spring 就像是整个项目中装配 bean 的大工厂，在配置文件中可以指定使用特定的参数去调用实体类的构造方法来实例化对象。也可以称之为项目中的粘合剂。Spring 的核心思想是 IoC (控制反转)，即不再需要程序员去显式地 `new` 一个对象，而是让 Spring 框架帮你来完成这一切。SpringMVC 在项目中拦截用户请求，它的核心 Servlet 即 DispatcherServlet 承担中介或是前台这样的职责，将用户请求通过 HandlerMapping 去匹配 Controller，Controller 就是具体对应请求所执行的操作。SpringMVC 相当于 SSH 框架中 struts。mybatis 是对 jdbc 的封装，它让数据库底层操作变的透明。mybatis 的操作都是围绕一个 sqlSession 实例展开的。mybatis 通过配置文件关联到各实体类的 Mapper 文件，Mapper 文件中配置了每个类对数据库所需进行的 sql 语句映射。在每次与数据库交互时，通过 sqlSession 拿到一个 sqlSession，再执行 sql 命令。当用户登入注册、进入首页、添加页面、商品详情、个人中心页，页面发送 Ajax 请求给控制器，控制器调用业务层处理逻辑，逻辑层向持久层发送请求，持久层与数据库交互，后将结果返回给业务层，业务层将处理逻辑发送给控制器，控制器再调用视图展现数据。使用户完成一系列流程。

五、收获与体会

(一) 收获

随着毕业日子的到来，毕业设计也接近了尾声。经过几周的奋战我的毕业设计终于完成了。在没有做毕业设计以前觉得毕业设计只是对这几年来所学知识的单纯总结，但是通过这次毕业设计发现自己的看法有点太片面。毕业设计不仅是对前面所学知识的一种检验，而且也是对自己能力的一种提高，通过这次毕业设计使我明白了自己原来知识还比较欠缺。自己要学习的东西还太多，以前老是觉得自己什么东西都会，什么东西都懂，有点眼高手低。通过这次毕业设计，我才明白学习是一个长期积累的过程，在以后的工作、生活中都应该不断地学习，努力提升自己知识和综合素质。

这次毕业设计中也使我们的同学关系更进一步了，同学之间互相帮助，有什么不

懂的大家在一起商量，听听不同的看法对我们更好的理解知识，所以在这里非常感谢帮助我的同学。

虽然系统设计的基本功能都已实现，但还是存在系统不稳定等多个问题尚待解决，这个网站项目是我自己开发的，但也得到老师和同学很大的帮助，我在做的过程中也遇到了很多问题，有的是知识存储不足，有的是考虑不周到，之所以能顺利完成基本功能离不开老师和同学们的帮助，事实证明，开发一套完整系统需要一个团队，因为各方面都会想的周到，所以最重要的是细心，一个人不一定能做到面面俱到。通过这次毕业设计我收获到应该注意编程的基本习惯和开发系统应注意流程。

总之，不管学会的还是学不会的的确觉得困难比较多，真是万事开头难，不知道如何入手。最后终于做完了有种如释重负的感觉。此外，还得出一个结论：知识必须通过应用才能实现其价值！有些东西以为学会了，但真正到用的时候才发现是两回事，所以我认为只有到真正会用的时候才是真的学会了。

（二）体会

在此要感谢我的指导老师夏维对我的悉心指导，感谢老师给我的帮助。在设计过程中，我通过查阅大量有关资料，与同学交流经验和自学，并向老师请教等方式，是自己学到了不少知识，也经历了不少艰辛，但收获同样巨大。在整个设计中我懂得了许多东西，也培养了我独立工作的能力，树立了对自己工作能力的信心，相信会对今后的学习工作生活有非常重要的影响。而且大大提高了动手的能力，使我充分体会到了在创造过程中探索的艰难和成功时的喜悦。虽然这个设计做的也不太好，但是在设计过程中所学到的东西是这次毕业设计的最大收获和财富，使我终身受益。

虽然在完成毕业设计的过程中有过失落，有过烦恼、有过悲伤，但在此期间跟同学老师的关系更加融洽了，有什么不懂的就问，因此在这里我要非常感谢帮助过我的同学。

在毕业设计完成过程里，我更明白了要做好人生规划的重要性，做任何事情所要有态度和心态，明白了做学问要一丝不苟，出现错误要通过正确途径去解决。

我相信，在以后的道路中我一定会铭记三年来带给我的每一份欢乐与汗水，将它

们绘制成只属于我的风画卷。

参考文献

- [1]贾晓芳, 沈泽刚。 Java Web 应用开发中的常见乱码形式及解决方法[J]. 软件导刊, 2017, (04) : 214-216.
- [2]周建儒。 基于 Java Web 的智能交通管理系统的设计与仿真[J]. 信息技术, 2017, (03) : 144-146.
- [3]胡蕊蕊。 Java Web 程序运用中泛型 DAO 的作用[J]. 电子测试, 2017, (04) : 46+49.
- [4]周昱晨, 罗向阳, 张豪。 基于 Java Web 的小型企业自动化办公系统设计与研究[J]. 计算机时代, 2017, (03) : 30-33+36.
- [5]张泽, 雷光波。 基于 Java Web 的创新创业管理系统设计与实现[J]. 软件导刊, 2017, (02) : 113-115.
- [6]苏航。 接口的意义及在 java web 三层架构中的作用分析[J]. 西部皮革, 2016, (24) : 19.
- [7]赵璞。 Java Web 程序开发中字符乱码的原因与解决办法[J]. 电脑编程技巧与维护, 2016, (20) : 24+27. [34]张伟, 刘延军。 基于 Java Web 的钢管企业局域网信息共享平台[J]. 现代冶金, 2015, (04) : 60-63. [42]杨宝林。 基于 Java Web 的学生成绩信息管理系统[D]. 吉林大学, 2015. [53]周兵。 Java Web 开发中的中文乱码问题分析及解决方案[J]. 电脑知识与技术, 2014, (29) : 6884-6888. [57]陈冈。 ANT 和 OpenSSH 在 Java Web 数据安全传输中的应用[J]. 电脑开发与应用, 2014, (08) : 47-49. [71]韩江。 基于 SSH2 架构的 Java Web 通用权限管理的设计[J]. 智能计算机与应用, 2014, (02) : 114-116. [80]鞠凤娟。 基于 Java Web 的新闻发布系统的设计与实现[J]. 信息通信, 2013, (07) : 97-98.
- [8]范成臣。 浅析在 JAVA WEB 应用程序中使用 Servlet 技术[J]. 数字技术与应用, 2013, (09) : 121. [87]程艳艳, 尹绍红。 浅析 Struts 框架下的 Java Web 应用[J]. 科技创新导报, 2013, (17) : 34-35.