

<<“学院路”校园购物平台>>

**设计说明书**



北京航空航天大学

2011-09

版本变更历史

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 版本 | 提交日期 | 主要编制人 | 审核人 | 版本说明 |
| V1.1 | 2016.11.25 | 江涛 | 张阳 | 最初版本 |
| V1.2 | 2016.11.26 | 江涛 | 卢嵩 | 修订版本 |
| V1.3 | 2016.11.27 | 江涛 | 冯岩 | 修订版本 |
| V1.4 | 2016.11.28 | 江涛 | 江涛 | 修订版本 |
| V1.5 | 2016.11.29 | 江涛 | 韩旭 | 修订版本 |
| V1.6 | 2016.11.29 | 江涛 | 冯岩 | 最终版本 |
|  |  |  |  |  |

目 录

[<<“学院路”校园购物平台>> 1](#_Toc468226631)

[1. 范围 1](#_Toc468226632)

[1.1 标识 1](#_Toc468226633)

[1.2 系统概述 1](#_Toc468226634)

[1.2.1 软件功能及特性 1](#_Toc468226635)

[1.2.2 当前与计划运行现场 2](#_Toc468226636)

[1.3 文档概述 2](#_Toc468226637)

[2. 引用文档 2](#_Toc468226638)

[3. 需求概述 3](#_Toc468226639)

[4. 体系结构设计 4](#_Toc468226640)

[4.1 总体结构 4](#_Toc468226641)

[4.2 功能分配 5](#_Toc468226642)

[4.3 关键问题及解决方案 6](#_Toc468226643)

[5. 接口设计 6](#_Toc468226644)

[5.1 用户界面设计 6](#_Toc468226645)

[5.2 外部接口设计 7](#_Toc468226646)

[5.3 内部接口设计 9](#_Toc468226647)

[6. 数据结构设计 10](#_Toc468226648)

[6.1 公共数据结构设计 10](#_Toc468226649)

[7. 详细设计 13](#_Toc468226650)

[7.1.1 01登录模块 13](#_Toc468226651)

[7.1.2 02客户管理模块 17](#_Toc468226652)

[7.1.3 03商品信息浏览查询模块 18](#_Toc468226653)

[7.1.4 04客户购买模块 19](#_Toc468226654)

[7.1.5 05客户网上银行结算模块 21](#_Toc468226655)

[7.1.6 06客户售后评价模块 21](#_Toc468226656)

[7.1.7 07客户退货模块 23](#_Toc468226657)

[7.1.8 08商家商品信息管理模块 24](#_Toc468226658)

[7.1.9 09商家订单管理模块 25](#_Toc468226659)

[7.1.10 10商家配送管理模块 26](#_Toc468226660)

[7.1.11 商家商品退货处理管理模块 27](#_Toc468226661)

[7.1.12 商家售后评价处理管理模块 28](#_Toc468226662)

[7.1.13 商家客户信息管理模块 29](#_Toc468226663)

# 范围

## 标识

文档标识号：A2016-10-20-00-14

文档标题：“学院路”校园购物平台需求规格说明书

项目/产品中文全称：“学院路”校园购物平台

项目/产品英文全称："College Road" campus shopping platform

项目/产品英文简称：LMS\_1.0

项目/产品编码：201610100014

## 系统概述

### 软件功能及特性

本系统目的主要是为用户提供在线的商品信息检索和网上购物，通过吸引用户浏览商品，诱导用户达成网上购物，从而实现网站盈利。

本平台具有强大灵活的网上购物功能、会员功能和网站内容管理功能，可满足各种不同网上商店的需要。具体来说主要功能如下：对于一般浏览者：查看网站主页，商品信息查询，浏览公告信息；对于用户：注册新用户，登录系统，浏览公告，发表留言，添加修改和删除购物车的信息，提交订单；购物系统管理后台功能：管理员注册系统，管理员登录系统，用户管理系统，订单管理系统，商品管理系统，公告管理系统。

界面设计层面上，我们牢牢把握目标用户为在校大学生的特点，运用现代流行理念和元素，成熟的网络开发技术体现网站现代不失传统，大众不失前卫的特点，并做好各个栏目的导航，突出网站的重要职能和重要内容，使得网站能够同今日高密度信息与市场营销行业完美结合。

### 当前与计划运行现场

本软件的目标用户为在校大学生。首先计划在北京航空航天大学校内进行推广和试运营，根据用户反馈情况进行升级和服务器扩容后可以向更多高校进行推广。

## 文档概述

编写此文档的目的是进一步定制软件开发的细节问题，希望能使本软件开发工作更具体。为了使用户、软件开发者及分析和测试人员对该软件的初始规定有一个共同的理解，基于功能需求、性能需求和数据需求，它说明了本软件的各项模块的过程设计描述，设计功能模块的内部细节，包括算法和数据结构，为编写源代码提供必要的说明。具体而言，编写软件设计说明的目的是为所开发的软件提出：

a)体系结构设计

b)数据结构设计

c)接口设计

d)构建设计

e)软件确认测试的依据

保密性要求：相关开发人员均签署相关保密协议

# 引用文档

[1] 杨叶中.《图书馆管理系统开发计划》.2013,9  
[2] 朱蔷.《基于WEB的北航二手书交易平台》.2013,9

[3] 王裔鸣.《餐厅服务系统》.2013,9

[4] 王珊、萨师煊.《数据库系统概论》[M].高等教育出版社，2006,12

[5] Michael T.Goodrich.《算法分析与设计》[M].人民邮电出版社，2006,10

[6] 《[软件开发计划](https://github.com/sebuaa2016/Team5/blob/master/%E8%BD%AF%E4%BB%B6%E5%BC%80%E5%8F%91%E8%AE%A1%E5%88%92_SDP_1.0.docx)》2016,9

[7] 《“学院路”校园购物平台\_需求规格说明书》2016,11

# 需求概述

(1) 用户注册后，登陆到网上购物系统中，可以进入购物流程（或在结帐之前注册并登陆系统）。登录页面输入帐号和密码，均属于非空项，使用javascript进行录入项检测。

(2) 客户可以登录网上购物系统浏览和购物，当注册后，可以在客户所购商品总金额达一定数量时，从普通会员根据量值不同自动升级成为不同等级的VIP会员，并享受不同折扣优惠。

(3) 客户一次可以购买多种商品，当确认购买之后，将产生一张订单。

(4) 订单生成后，客户可在前台查询订单的处理状态：“T”表示订单上的商品已发出，“F”表示订单上的商品未发出。若订单不能在规定日期送达，客户可退单或做[其它](http://portal.vsharing.com/industry/1625.html)处理。

(5) 一个销售商可以提供多种商品，且一种商品可以由多家销售商提供。

(6)客户和商家可在系统发表留言或评论。

(7)新增需求客户可以在订单信息中查看商品的配送情况

# 体系结构设计

## 总体结构

C:\Users\150T\Pictures\软工.png

系统由3个大模块，13个小模块构成：

|  |  |  |
| --- | --- | --- |
| **序号** | **编号** | **名称** |
| 1 | 01 | 登录模块 |
| 2 | 02 | 客户个人信息管理模块 |
| 3 | 03 | 客户商品信息浏览、查询模块 |
| 4 | 04 | 客户购买模块 |
| 5 | 05 | 客户网上银行结算模块 |
| 6 | 06 | 客户售后评价模块 |
| 7 | 07 | 客户退货模块 |
| 8 | 08 | 商家商品信息管理模块 |
| 9 | 09 | 商家订单管理模块 |
| 10 | 10 | 商家配送管理模块 |
| 11 | 11 | 商家商品退货处理模块 |
| 12 | 12 | 商家售后评价处理模块 |
| 13 | 13 | 商家客户信息管理模块 |

## 功能分配

需求对应：

1. 用户注册和登陆：01登录模块
2. 商品浏览与搜索：03客户商品信息浏览、查询模块
3. 管理购物车：04客户购买模块
4. 支付功能：05客户网上银行结算模块
5. 评价商品：06客户售后评价模块
6. 退换货：07客户退货模块
7. 配送情况：10商家配送管理模块
8. 处理订单：09商家订单管理模块
9. 维护商品：08商家商品信息管理模块
10. 售后评价：12商家售后评价处理模块
11. 退换货管理：11商家商品退货处理模块

## 关键问题及解决方案

本购物网站的总体结构中最关键的问题在于网站高并发、大流量访问的处理。作为一个购物网站，面对着巨大的用户群体，同时包含了客户和商家，因而在同一时刻可能会有极大的访问流量，从而导致服务器反应缓慢、请求丢失，甚至崩溃的问题，解决方法如下：

1. 配置高性能的专用服务器以支持当前流量。
2. 优化数据库访问查询命令，并使用缓存技术，将动态数据存储到缓存文件中，动态网页直接调用。
3. 禁止外部的盗链，外部网站的图片或者文件盗链往往会带来大量的负载压力，因此应该严格限制外部对于自身的图片或者文件盗链，好在目前可以简单地通过refer来控制盗链，Apache自己就可以通过配置来禁止盗链，IIS也有一些第三方的ISAPI可以实现同样的功能。当然，伪造refer也可以通过代码来实现盗链，不过目前蓄意伪造refer盗链的还不多， 可以先不去考虑。
4. 使用不同主机分流主要流量。

# 接口设计

## 用户界面设计

界面设计总体决策：

参考当下的各大电商网站的界面布局，对应地修改为符合本站的内容

参考图片：



## 外部接口设计

1. 用来查询一条数据的私有接口

该方法有一个参数，参数是一个字符串，表示要向数据库发送一条SQL语句，这个SQL只是一个查询语句，方法的返回植是一个Hashtable，在Hashtable中以键值对的方式表示了从数据库中选出了第一行记录。用Hashtable中的get(“FieldName”)方法可以得到改行记录的某一个记录的值。

private Hashtable searchOneData(String sql)

{

Hashtable hash = new Hashtable();

ResultSet rs = selectRecord(sql);

Statement stmt = null;

try{

//取得列数和列名

ResultSetMetaData rsmd = rs.getMetaData();

int cols = rsmd.getColumnCount();

if(rs.next())

{

for(int i=1;i<=cols;i++)

{

String field= ds.toString(rsmd.getColumnName(i)); String value = ds.toString(rs.getString(i));

hash.put(field,value);

}

}

}catch(Exception e){

System.out.println("运行时出错："+e);

}

finally{

if(rs!=null)try{

stmt=rs.getStatement(); rs.close();

}

catch(Exception e){

System.out.println("关闭记录集rs时出错"+e);

}

if(stmt!=null)

try{

stmt.close();

}catch(Exception e){

System.out.println("关闭声明时statement出错"+e);

}

return hash;

}

## 内部接口设计

1. 增加记录的公共接口

该方法有一个参数，是java.util.Hashtable类，在调用该方法前，先用和hashtable的put方法将字段名和该条记录的值存入hashtable中，然后将这个hashtable作为参数传入insert方法中。 在insert方法的最后，调用ParentBean中的insertRecord方法，insertRecord可以根据传入的参数自动的生成增加记录的SQL语句并通过JDBC发送到数据库

Public int insert(Hashtable hash){

Int intID=makeID(“item”,”itemID”,””,true);

Vector vect=new Vector();

Vect.add(“………….”);

……………………

Return insertRecord(vect);

}

1. 删除记录的公共接口

该方法有一个参数id，该参数表示的是item表中的ID字段，ID字段是这个表的主键，用这个主键可以检索到表中的一条唯一的记录，通过这个主键，delete方法可以生成一条SQL语句，删除这一条记录。

public int delete(String id)

{

String sql="";

sql="delete from Item where ItemID="+id;

return deleteRecord(sql);

}

（3） 查询记录的公共接口

Public Vector getData(String sqlwhere, int page){

String sql=””;

Sql=”select \* from Item”;

If (!sqlwhere.equals(“”))

Sql+=sqlwhere;

Return getOnePage(sql,page,20);

}

（4）修改记录的接口

public int update(Hashtable hash)

{

Vector vect = new Vector();

vect.add("Book");

vect.add(addVector("Title",ds.toString((String)hash.get("NAME")),"CHAR"));

vect.add(addVector("Author",ds.toString((String)hash.get("SELLER")),"CHAR"));

vect.add(addVector("ISBN",ds.toString((String)hash.get("SUBNAME")),"CHAR"));

……………………

return updateRecord(vect);

}

# 数据结构设计

## 公共数据结构设计

使用了\_SESSION全局变量保存用户的登陆状态，id,以及用户名。

\_SESSION[login]保存登陆状态

\_SESSION[ID]保存已登陆用户的id

\_SESSION[nane]保存已登陆用户的名字

使用了常量

define("SELLER", "3");

define("CUSTOMER ", "2");

define("ADMIN ", "1");

以上三个为数据库用户的职位

define("LARGE", 1);

define("EXHAUSTED",0);

以上两个为商品库存的状态，充足或者不足

define("FEMALE", "1");

define("MALE", "0");

以上为用户的性别

class customer

{

public $name; //用户的名字，为public

public $sex; // 用户的性别, 为public

private $phone; // 用户的手机号, 为private

private $address; // 用户的地址，为private

private $id; // 用户的id, 为private

}

class item

{

public $name; //商品的名字，为public

public $price; //商品的价格，为public

public $describe; //商品的描述，为public

public $stock; //商品的库存，为public

public $picture; //商品的图片，为public

}

class order

{

public $no; //订单号，为public

public $company; //物流公司，为public

private $courierNo; //物流编号，为private

public $customer; // 收件人，为public

private $custmerPho; //收件人手机号，为private

public $sender; //发件人，为public

private $senderPho; //发件人手机号，为private

private $address; //目的地址，为private

}

数据库设计

|  |  |  |  |
| --- | --- | --- | --- |
| 表名称 | 设计者 | 审核者 | 完成日期 |
| Item | 江涛 | 冯岩 | 2016.11.26 |
| User | 江涛 | 卢嵩 | 2016.11.26 |
| Seller | 江涛 | 张阳 | 2016.11.26 |
| Admin | 江涛 | 韩旭 | 2016.11.26 |
| Order | 江涛 | 冯岩 | 2016.11.26 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 数据库的表结构 | 表名称 | 设计者 | 审核者 | 完成日期 |
| manager | 江涛 | 卢嵩 | 2016.11.26 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 字段代号 | 名称 | 类型 | 位数 | 值约束 | 索引或键 | 备注 | 缩写词 |
| 1 | AdminID | 文本 | 6 | 6位非空字符串 | 是 | 用户名 | AID |
| 2 | AdminPwd | 文本 | 16 | 6位以上，16位以下非空字符串 | 否 | 密码 | APWD |
| 3 | AdminName | 文本 | 10 | 10位或以下非空字符串 | 否 | 姓名 | AN |
| 4 | AdminSex | int | 1 | 1位数字 | 否 | 性别 | AS |
| 5 | AdminAddTime | 数字 | 10 | 10位unix时间戳 | 否 | 加入时间 | AT |
| 6 | AdminContact | 文本 | 20 | 不多于20位字符串 | 否 | 联系方式 | AC |

# 详细设计

### 01登录模块

#### 功能流程图



需要说明的问题：

1. 录入项检测使用javascript实现（各项必须非空）
2. 登录.jsp页面提交的数据只是用户名和密码

#### 功能描述

功能类型：查询数据

功能描述：提高系统的安全性

前提业务：无

后继业务：02 （管理模块）

功能约束：权限约束

约束描述：

操作权限：网站注册用户或商家

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 登录 | 点击登录按钮，提交数据到登录数据处理.jsp页面 |

#### 数据要求

1. 功能类型：数据查询
2. 数据描述：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名称** | **长度** | **录入方式** | **是否非空项** | **数据检验** | **默认显示** |
| 帐号 | 32 | 文本框 | Y | N |  |
| 密码 | 32 | password | Y | N |  |

#### 登录数据处理.jsp内部逻辑

登陆数据处理：

关键点两点：1，数据库连接；2，记录登陆信息及信息处理；

1. 数据库连接：

如下：

**public** **class** onlineshop\_system\_Conn **extends** Object{

**public** onlineshop\_system\_Conn(){

}

**private** Connection conn = **null**;

**private** ResultSet rs;

String re = "";

//设置你的数据库ip

String dbip = "127.0.0.1";

//设置你的数据库用户名和密码：

String use = "root";

String pass = "860409";

//设置您的数据库名

String dbName = "onlineshop\_system";

**public** java.sql.Connection getConn(){

**try**{

Class.*forName*("org.gjt.mm.mysql.Driver").newInstance();

String url ="jdbc:mysql://"+dbip+":3306/"+dbName+"?user="+use+"&password="+pass+"&useUnicode=true&characterEncoding=GBK" ;

conn= DriverManager.*getConnection*(url);

}

**catch**(Exception e){

e.printStackTrace();

}

**return** **this**.conn ;

}

}

1. 记录登陆信息及信息处理：

当用户点击“登陆”按钮之后，数据将提交到登陆数据处理.jsp页面。 取得帐号密码这两个从页面传入的值，然后跟数据库当中用户表中的账号和密码比较。如果正确的话，在JSP的SESSION中存入一个标记属性，表示当前已经有用户登陆了。处理完毕后，跳转到管理页面，如果失败，则提示登陆失败，并重新进入到登陆页面。

具体的逻辑如下：

<%

String username=ParamUtil.getString(request,"user\_ID");

//取得用户名

String password=ParamUtil.getString(request,"user\_password");

//取得密码

String erroMsg="";

//错误码

if(username!=null&&password!=null)

try

{

SkinUtil.login(request,response,user\_ID,user\_password);

//在JSP的session中存如一个标记属性，表示当前已经有用户登陆了

Session.setAttribute(“user\_ID”,user\_ID);

response.sendRedirect("login\_process.jsp");

}

catch(UserNotFoundException e)

{

erroMsg="错误的用户名和密码";

response.sendRedirect("login\_process.jsp?"+response.encodeURL(erroMsg));

}

else{

out.println("请填写好你的个人信息!");

}

%>

### 02客户管理模块

#### 功能流程图

C:\Users\150T\Pictures\11.png

需要说明的问题：

当选择相应的业务时，需要在当前页面显示；并且在管理页面内，默认显示商品浏览查询页面。

#### 功能描述

1. 功能类型：其他
2. 功能概述：总体归纳客户前端管理功能
3. 前提业务：登录模块（01）
4. 后续业务：03，05，06，07
5. 功能约束：权限约束
6. 约束描述：
7. 操作权限：注册客户

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 个人信息管理 | 点击按钮，将页面转入个人信息管理.jsp页面 |
| 2 | 商品浏览查询 | 点击按钮，将页面转入商品浏览查询.jsp页面 |
| 3 | 购物车 | 点击按钮，将页面转入购物车.jsp页面 |
| 4 | 售后评价 | 点击按钮，将页面转入售后评价.jsp页面 |
| 5 | 退货 | 点击按钮，将页面转入退货.jsp页面 |

#### 数据要求

功能类型：其他

#### 个人信息管理.jsp页面说明

客户可以在该页面中管理和修改自己的个人信息，包括地址、手机、密码等。

### 03商品信息浏览查询模块

#### 功能流程图

C:\Users\150T\Pictures\12.png

需要说明的问题：

1. 进入页面后会自动显示推荐的商品信息，不需要等待查询操作
2. 录入项检测使用javascript来实现（各项非空）；

#### 功能描述

1. 功能类型：查询数据
2. 功能概述：显示查询结果
3. 前提业务：无
4. 后继业务：04
5. 功能约束：没有约束
6. 操作权限：面向所有人

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 确定 | 点击按钮，提交数据到商品查询数据处理.jsp页面 |
| 2 | 购物车 | 点击按钮，将页面转入购物车.jsp页面 |

#### 数据要求

1. 功能类型：数据查询
2. 数据描述：

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名称** | **长度** | **录入方式** | **是否非空项** | **数据检验** | **默认显示** |
| 查询内容 | 200 | 文本框 | Y | N |  |

1. 商品信息查询的返回结果

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **长度** | **显示方式** | **数据类型** |
| 商品名称 | 20 | 文本 | string |
| 商品图片 |  | 图片 | bitmap |
| 产品介绍 | 100 | 文本 | string |
| 价格 | 10 | 文本 | int |
| 添加到购物车 |  | 按钮 | button |
|  |  |  | unsigned int |
|  |  |  | unsigned int |

### 04客户购买模块

#### 功能流程图

C:\Users\150T\Pictures\13.png

需要说明的问题：

在购物车页面中显示所有被添加到购物车的商品信息，列出用户的购物车的所有商品，并且每个商品列出商品名、所属的产品名、商品描述、是否有库存、购买数量、单价以及移除按钮。用户可以任意更改购买某个商品的数量，单击更新购物车按钮即可实现更新。用户可以单击移除按钮移除选择的商品，则可实现该商品从购物车中删除。同时在购物车的下方列出了总的价格。

#### 功能描述

（1）功能类型：修改数据和查询数据

（2）功能概述：显示购物车中商品信息，并可以修改购买数量，或从购物车中删除该商品。

（3）前提业务：

（4）后继业务：05

（5）功能约束：注册登录客户

（6）约束描述：

（7）操作权限：面向所有注册登录客户

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 增加 | 点击按钮，将对应商品数量增加1 |
| 2 | 减少 | 点击按钮，将对应商品数量减少1 |
| 3 | 结算 | 点击按钮，跳转到客户网上银行结算.jsp页面 |

#### 数据要求

数据描述：

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **长度** | **显示方式** | **数据类型** |
| 商品名称 | 20 | 文本 | string |
| 所属得到产品名 | 20 | 文本 | string |
| 商品描述 | 100 | 文本 | string |
| 是否有库存 | 2 | 文本 | string |
| 购买数量 | 10 | 文本框 | int |
| 单价 | 10 | 文本 | unsigned int |

### 05客户网上银行结算模块

#### 功能流程图

C:\Users\150T\Pictures\14.png

需要说明的问题：

用户选择支付方式之后，将跳转到对应的支付平台自己的页面完成进一步的支付操作，支付完成后将自动跳转回到本网站，并显示支付结果

#### 功能描述

（1）功能类型：其他

（2）功能概述：选择支付平台进行支付操作

（3）前提业务：04

（4）操作权限：注册登录客户

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 支付 | 点击按钮，将跳转到对应的支付平台页面 |
| 2 | 选择 | 点击radiobutton，将选中对应的支付平台 |

### 06客户售后评价模块

#### 功能流程图

C:\Users\150T\Pictures\15.png

#### 功能描述

（1）功能类型：添加数据

（2）功能描述：客户对购买过的商品添加评价

（3）前提业务：

（4）后继业务：

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录客户

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 支付 | 点击按钮，将跳转到对应的支付平台页面 |
| 2 | 选择 | 点击radiobutton，将选中对应的支付平台 |

#### 数据要求

1. 功能类型：数据增加。

（2） 数据描述：

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **字段名称** | **长度** | **录入方式** | **是否非空项** | **数据检验** | **默认显示** | **数据类型** |  |
| 售后评价 | 200 | 文本框 | N | N |  | string |  |
| 打分 | 1 | 下拉菜单 | Y | N |  | int[] |  |

### 07客户退货模块

#### 功能流程图

C:\Users\150T\Pictures\16.png

需要说明的问题：

客户在退货页面选择退货或者换货，并注明原因，点击提交按钮。申请将发送给商家确认，确认后用户将商品机会，商家收到后返还用户费用或寄回换货。

#### 功能描述

（1）功能类型：其他

（2）功能描述：客户选择退换货并注明原因

（3）前提业务：

（4）后继业务：

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录客户

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 输入 | 注明退换货原因 |
| 2 | 选择 | 点击radiobutton，选择退货或者换货 |
| 3 | 提交 | 点击按钮，提交退换货申请 |

### 08商家商品信息管理模块

#### 功能流程图

C:\Users\150T\Pictures\117.png

#### 功能描述

（1）功能类型：修改数据或添加删除数据

（2）功能描述：商家对自己所售商品信息进行修改，或添加、删除商品

（3）前提业务：

（4）后继业务：

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录商家

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 添加 | 点击按钮，添加新的商品信息 |
| 2 | 删除 | 点击按钮，将列表中已有的一件商品删除 |
| 3 | 修改 | 点击按钮，修改对应的商品信息 |

#### 数据要求

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **长度** | **显示方式** | **数据类型** |
| 商品名称 | 20 | 文本 | string |
| 所属得到产品名 | 20 | 文本 | string |
| 商品图片 |  | 图片 | bitmap |
| 商品描述 | 100 | 文本 | string |
| 是否有库存 | 2 | 文本 | string |
| 购买数量 | 10 | 文本框 | unsigned int |
| 单价 | 10 | 文本 | unsigned int |

### 09商家订单管理模块

#### 功能流程图

C:\Users\150T\Pictures\17.png

#### 功能描述

（1）功能类型：其他

（2）功能描述：商家对订单进行管理

（3）前提业务：

（4）后继业务：10 11 12

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录商家

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 配送 | 点击按钮，将跳转到商品配送管理.jsp页面 |
| 2 | 处理 | 点击按钮，将跳转到商品退货处理.jsp页面 |
| 3 | 客户 | 点击按钮，将跳转到客户信息管理.jsp页面 |

#### 数据要求

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **长度** | **显示方式** | **数据类型** |
| 订单号 | 20 | 文本 | string |
| 商品名称 | 20 | 文本 | string |
| 商品图片 |  | 图片 | bitmap |
| 商品描述 | 100 | 文本 | string |
| 购买数量 | 10 | 文本框 | unsigned int |
| 单价 | 10 | 文本 | unsigned int |
| 总价 | 10 | 文本 | unsigned int |

### 10商家配送管理模块

#### 功能流程图

C:\Users\150T\Pictures\18.png

#### 功能描述

（1）功能类型：添加数据

（2）功能描述：商家为订单添加配送信息

（3）前提业务：09

（4）后继业务：

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录商家

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 提交 | 点击按钮，将物流信息更新到订单信息中 |

#### 数据要求

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **长度** | **显示方式** | **数据类型** |
| 订单号 | 20 | 文本 | string |
| 物流公司 | 20 | 文本框 | string |
| 物流编号 | 20 | 文本框 | string |
| 收件人 | 4 | 文本框 | string |
| 收件手机号 | 11 | 文本框 | string |
| 发件人 | 4 | 文本框 | string |
| 发件手机号 | 11 | 文本框 | string |
| 目的地址 | 40 | 文本框 | string |

### 商家商品退货处理管理模块

#### 功能流程图

C:\Users\150T\Pictures\19.png

#### 功能描述

（1）功能类型：其他

（2）功能描述：商家确认并处理退换货申请

（3）前提业务：

（4）后继业务：

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录商家

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 提交 | 点击按钮，将处理结果更新到退换货状态中 |

### 商家售后评价处理管理模块

#### 功能流程图

C:\Users\150T\Pictures\20.png

#### 功能描述

（1）功能类型：添加数据

（2）功能描述：商家对客户的售后评价做出回复

（3）前提业务：

（4）后继业务：

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录商家

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 提交 | 点击按钮，将回复添加到售后评价回复中 |

### 商家客户信息管理模块

#### 功能流程图

C:\Users\150T\Pictures\21.png

#### 功能描述

（1）功能类型：其他

（2）功能描述：商家查看客户信息

（3）前提业务：

（4）后继业务：

（5）功能约束：权限约束

（6）约束描述：无

（7）操作权限：注册登录商家

#### 动作说明

|  |  |  |
| --- | --- | --- |
| **动作编号** | **动作名称** | **动作描述** |
| 1 | 详细 | 点击按钮，将显示对应客户的详细信息 |

#### 数据要求

|  |  |  |  |
| --- | --- | --- | --- |
| **字段名称** | **长度** | **显示方式** | **数据类型** |
| 客户姓名 | 20 | 文本 | string |
| 手机号 | 20 | 文本框 | string |
| 地址 | 20 | 文本框 | string |