**浙江大学城市学院**

**总体设计报告（ISO标准）**



组序： G23 ；

组长： 乔寒月 ；

组内成员： 李欣飏，吴智宏 ；

指导老师： 杨枨 ；

提交日期： 2019/3/16 ；

版本控制

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 版本/状态 | 作者 | 参与者 | 批准人 | 起止日期 | 备注 |
| 1.0 | 乔寒月 | 李欣飏、吴志宏 | 杨枨 | 起：  2019/4/16  止：  2019/4/21 |  |
| 1.1 | 乔寒月 | 李欣飏 | 杨枨 | 起：  2019/4/23  止：  2019/5/5 |  |
| 1.2 | 乔寒月 | 李欣飏 | 杨枨 | 起：  2019/5/21  止：  2019/5/26 |  |
| 1.3 | 乔寒月 | 李欣飏 | 杨枨 | 起：  2019/6/11  止：  2019/6/16 |  |

目录

[1.引言 4](#_Toc6595202)

[1.1编写目的 4](#_Toc6595203)

[1.2背景 4](#_Toc6595204)

[1.3定义 4](#_Toc6595205)

[1.4参考资料 4](#_Toc6595206)

[2.总体设计 5](#_Toc6595207)

[2.1需求规定 5](#_Toc6595208)

[2.1.1系统功能 5](#_Toc6595209)

[2.1.2系统性能 5](#_Toc6595210)

[2.1.2.1精度 5](#_Toc6595211)

[2.1.2.2时间特性要求 9](#_Toc6595212)

[2.1.2.3可靠性 9](#_Toc6595213)

[2.1.2.4灵活性 9](#_Toc6595214)

[2.1.3输入输出要求 10](#_Toc6595215)

[2.1.4数据管理能力要求 12](#_Toc6595216)

[2.1.5故障处理要求 12](#_Toc6595217)

[2.1.6其他专门要求 12](#_Toc6595218)

[2.2运行环境 12](#_Toc6595219)

[2.2.1设备 12](#_Toc6595220)

[2.2.2支持软件 12](#_Toc6595221)

[2.2.3接口 12](#_Toc6595222)

[2.2.4控制 13](#_Toc6595223)

[2.3基本设计概念和处理流程 13](#_Toc6595224)

[2.4结构 17](#_Toc6595225)

[2.4.1总体框图 17](#_Toc6595226)

[2.4.2HIPO图 17](#_Toc6595227)

[2.5系统模块 17](#_Toc6595228)

[2.6功能需求与系统模块的关系 17](#_Toc6595229)

[2.7人工处理过程 17](#_Toc6595230)

[2.8尚未解决的问题 17](#_Toc6595231)

[3．接口设计 17](#_Toc6595232)

[3.1用户接口 17](#_Toc6595233)

[3.2外部接口 22](#_Toc6595234)

[3.3内部接口 22](#_Toc6595235)

[4．运行设计 23](#_Toc6595236)

[4.1运行模块组合 23](#_Toc6595237)

[4.2运行控制 23](#_Toc6595238)

[4.3运行时间 23](#_Toc6595239)

[5．系统数据结构设计 23](#_Toc6595240)

[5.1逻辑结构设计要点 23](#_Toc6595241)

[5.2物理结构设计要点 23](#_Toc6595242)

[5.3数据结构与程序的关系 23](#_Toc6595243)

[5.4数据库设计 24](#_Toc6595244)

[6．系统出错处理设计 24](#_Toc6595245)

[6.1测试计划 24](#_Toc6595246)

[6.2出错信息 24](#_Toc6595247)

[6.3补救措施 24](#_Toc6595248)

[6.4系统维护设计 24](#_Toc6595249)

# 1.引言

## 1.1编写目的

目的：

说明对程序系统的设计考虑，包括程序系统的基本处理流程、程序系统的组织结构、模块划分、功能分配、接口设计、运行设计、安全设计、数据结构设计和出错处理设计等，为程序的详细设计提供基础。

预期读者：

软件制作小组成员：乔寒月、李欣飏、吴智宏；

软件用户代表：杨枨老师、信管1702 郭伊娜、统计1801 朱芳颖。

## 1.2背景

软件名称：承闲二手购物小程序；

任务提出者：李欣飏；

开发者：乔寒月、李欣飏、吴智宏；

用户：杨枨老师、信管1702 郭伊娜、统计1801 朱芳颖。

## 1.3定义

暂无。

## 1.4参考资料

张海藩、牟永敏，软件工程导论（第六版），清华大学出版社

乔寒月、李欣飏、吴智宏，项目计划书

乔寒月、李欣飏、吴智宏，可行性分析报告

# 2.总体设计

## 2.1需求规定

### 2.1.1系统功能

**用户功能：**

实名认证：用户输入有关本人的真实信息。

申诉系统：对于交易及评价有问题的用户可以进行申诉。

个人中心：包括我的资料、我的订单、发布商品、申诉、我的评价等内容。

关键字查询：通过关键字查询商品。

收藏：用户可以对喜欢的商品进行收藏。

购物车：用户可以将想要购买的商品加购。

提交订单：决定购买时要进行订单提交。

评价系统：收到商品可以进行评价。

查询订单信息：查询自己购买过的东西。

上传商品：对想要售出的商品进行上传。

修改商品信息：对已上传的商品信息进行修改。

下架商品：下架不想出售的商品。

### 2.1.2系统性能

#### 2.1.2.1精度

**数据字典**

|  |  |
| --- | --- |
| **名字** | **收货地址** |
| **别名** |  |
| **描述** | **收货地址** |
| **定义** | **收货地址=地址编号+地址内容+收货人+用户编号**  地址编号：id` int(11)  地址内容：dz` varchar(255)  收货人：name` varchar(255)  手机号：phone` varchar(255)  用户编号：userId` int(11) |
| **位置** | **收货地址表** |

|  |  |
| --- | --- |
| **名字** | **购物车** |
| **别名** |  |
| **描述** | **用户添加的购买信息** |
| **定义** | **购物车=购物车编号+数量+价格+商品编号+用户编号**  购物车编号：id` int(11)  数量：num` int(11)  价格：price` double  商品编号：goodsId` int(11)  用户编号：userId` int(11) |
| **位置** | **购物车表** |

|  |  |
| --- | --- |
| **名字** | **订单** |
| **别名** |  |
| **描述** | **用户购买信息时生成的订单** |
| **定义** | **订单=订单编号+收货地址+单价+生成时间+商品名+数量+用户手机号+总价+商品编号+用户编号**  订单编号：id` int(11)  收货地址：address` varchar(255)  单价：cost` double  生成时间：createTime` datetime  商品名：name` varchar(255)  数量：num` int(11)  用户手机号：phone` varchar(255)  总价：price` double  商品编号：goodsId` int(11)  用户编号：userId` int(11) |
| **位置** | **订单表** |

|  |  |
| --- | --- |
| **名字** | **商品信息** |
| **别名** |  |
| **描述** | **商品的各项信息** |
| **定义** | **商品信息=商品编号+用户编号+是否首页循环+单价+商品名+图片1+图片2+图片3+图片4+二级分类编号+数量**  商品编号：id` int(11)  用户编号：userid int(11)  是否首页循环：isLb` int(11)  单价：price` double  商品名：title` varchar(255)  图片1：url1` varchar(255)  图片2：url2` varchar(255)  图片3：url3` varchar(255)  图片4：url4` varchar(255)  二级分类编号：ppId` int(11)  数量：num` int(11) |
| **位置** | **商品信息表** |

|  |  |
| --- | --- |
| **名字** | **一级分类** |
| **别名** |  |
| **描述** | **商品分类信息** |
| **定义** | **一级分类=一级分类编号+一级分类名**  一级分类编号：id` int(11)  一级分类名：name` varchar(255) |
| **位置** | **一级分类表** |

|  |  |
| --- | --- |
| **名字** | **二级分类** |
| **别名** |  |
| **描述** | **商品分类信息** |
| **定义** | **二级分类=二级分类编号+二级分类名+缩略图+所属一级分类编号**  二级分类编号：id` int(11)  二级分类名：name` varchar(255)  缩略图：cpUrl` varchar(255)  所属一级分类编号：lbId` int(11) |
| **位置** | **二级分类表** |

|  |  |
| --- | --- |
| **名字** | **管理员** |
| **别名** |  |
| **描述** | **管理员个人信息** |
| **定义** | **管理员=管理员编号+管理员编号+管理员昵称+密码+管理员名字**  管理员编号：id` int(11)  管理员昵称：name` varchar(255)  密码：passWord` varchar(255)  管理员名字：realName` varchar(255) |
| **位置** | **管理员表** |

|  |  |
| --- | --- |
| **名字** | **意见反馈** |
| **别名** |  |
| **描述** | **用户反馈意见** |
| **定义** | **意见反馈=反馈编号+反馈人姓名+手机号+反馈内容+反馈时间**  反馈编号：id` int(11)  反馈人姓名：name` varchar(255)  手机号：phone` varchar(255)  反馈内容：content` varchar(255)  反馈时间：addTime` datetime |
| **位置** | **反馈表** |

|  |  |
| --- | --- |
| **名字** | **评价** |
| **别名** |  |
| **描述** | **买家用户对商品进行的评价** |
| **定义** | **评价=评价编号+评价时间+评价内容+商品编号+用户编号**  评价编号 id` int(11)  评价时间：addTime` datetime  评价内容 content` varchar(255)  商品编号 goodsId` int(11)  用户编号 userId` int(11) |
| **位置** | **评价表** |

|  |  |
| --- | --- |
| **名字** | **用户实名信息** |
| **别名** |  |
| **描述** | **用户实名认证时填写的个人信息** |
| **定义** | **用户信息=用户编号+小程序编号+姓名+性别+职业+身份证号**  用户编号 id` int(11)  小程序编号：openId` varchar(255)  姓名 username varchar(10)  性别 sex varchar(2)  职业 job varchar(20)  身份证号 personid int(18) |
| **位置** | **用户实名信息表** |

|  |  |
| --- | --- |
| **名字** | **收藏夹** |
| **别名** |  |
| **描述** | **用户收藏的商品信息** |
| **定义** | **收藏夹=收藏编号+用户编号+商品编号**  收藏编号：id` int(11)  用户编号：userId` int(11)  商品编号：goodsId` int(11) |
| **位置** | **收藏夹表** |

#### ER图



#### 2.1.2.2时间特性要求

用户打开界面和提交事务的平均响应时间低于1.5s。用户进行在线实时查询业务操作的数据处理时间应低于3s。

#### 2.1.2.3可靠性

系统应具有较高的可靠性，综合可靠性包括从实名认证到浏览商品信息到进行收藏、加购、订单提交过程正常运行的概率，用户实名信息不会主动泄露，用户交易出现问题及时解决。

#### 2.1.2.4灵活性

用户界面操作简单、灵活、统一易于使用。功能模块明确，符合用户思考方式，方便用户查找。小程序寄生于微信，打开关闭快速方便。

### 2.1.3输入输出要求

**用户功能：**

|  |  |  |  |
| --- | --- | --- | --- |
| 功能 | 输入 | 处理 | 输出 |
| 实名认证 | 输入相关要求的信息 | 功能：经过用户授权，可获得用户在微信支付认证的姓名以及身份证信息（非身份证的其他证件信息暂不提供授权）  验证方式：因为需要用户主动触发才能发起获取实名信息接口，所以该功能不由 API 来调用，需用 <button> 组件的点击来触发。且需要用户输入微信支付密码验证后，方可算授权。 | 显示是否认证成功  若未成功，小程序功能不对该用户开放 |
| 申诉系统 | 输入回复、申诉内容 |  | 显示提交申诉成功，审核后予以反馈 |
| 个人中心 | 点击个人中心 |  | 显示个人中心的内容 |
| 关键字查询 | 输入想要查询的内容：中文 英文 或数字符号 | 数据返回后台通过sql相关语句功能进行查找 | 显示用户想要查找的相关内容 |
| 收藏 |  | 增减数量  删除商品 | 删除商品。当收藏页面为空时，页面会变为空收藏页面的布局 |
| 购物车 |  | 购物车列表数据我们一般是通过请求服务器拿到的数据，  布局 wxml  计算总价  选择事件  全选事件  增减数量  删除商品 | * 单选、全选和取消，而且会随着选中的商品计算出总价 * 单个商品购买数量的增加和减少 * 删除商品。当购物车为空时，页面会变为空购物车的布局 |
| 提交订单 | 点击提交按钮进行提交 |  | 生成订单信息 |
| 评价系统 | 在相应的位置根据相应的要求输入评论 | 因为评论内容要嵌入到其它内容里面，也就是用户在某个内容上面提交的评论要属于这个内容，就是评论内容与被评论的内容之间要关联在一起。比如可以在接口里面用 post 属性表示评论所属的内容。评论之间还可以进行回复，一个人留了一条评论，另一个人可以回复这条评论，就是评论与回复之间也应该有个关联，比如在接口里用 parent 表示这条评论回复的是哪条评论  先在内容页面上显示一个评论列表，我们之间介绍过实施[内容列表的功能](https://ninghao.net/course/5457)，列表可以[无限滚动](https://ninghao.net/video/5474)加载新的列表项目。 | 提示是否评论成功  显示评论 |
| 查询订单信息 | 从个人中心查看历史订单 |  | 输出订单信息 |
| 上传商品 | 输入商品信息 | 上传要出售的商品信息 | 输出商品信息 |
| 修改商品信息 | 输入修改后的商品信息 |  | 输出商品信息 |
| 下架商品 | 删除商品 |  |  |

### 2.1.4数据管理能力要求

需要管理各类商品信息表和用户个人信息表

按照当下项目计划以及各项功能保守预计两方面内容将各占15—20M

### 2.1.5故障处理要求

对错误的操作给出错误提示，对未能正常完成的操作给出失败提示。

### 2.1.6其他专门要求

不会主动向第三方机构泄露用户信息；

寄生于微信，可以在ios及安卓机上使用。

## 2.2运行环境

### 2.2.1设备

安卓手机：系统要求：安卓4.0以上系统适用屏幕像素：通用

iPhone：系统要求：iPhone6.1-iPhone8.0以上系统适用屏幕像素：通用；

输入设备：一块触摸屏

输出设备：一块触摸屏

### 2.2.2支持软件

支持苹果端和安卓端的微信平台。

### 2.2.3接口

同其他系统的接口：微信小程序连接本地接口。

数据通信协议：TCP/IP协议。

### 2.2.4控制

控制方法：集中控制；

控制信号：数字信号。

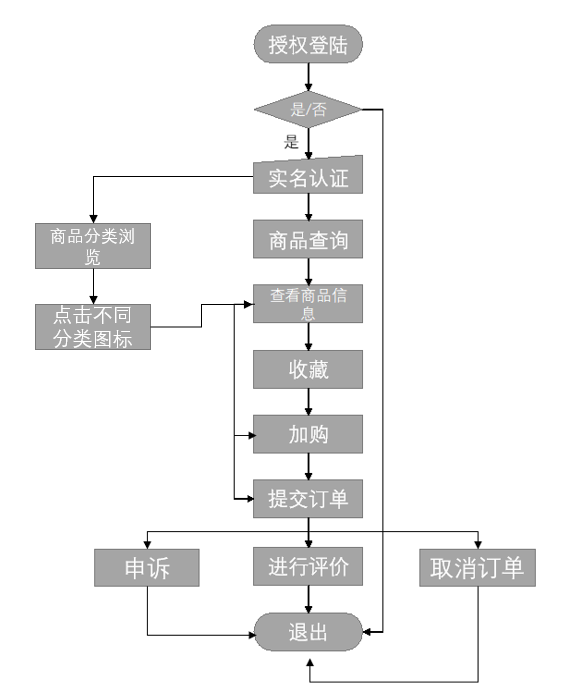
## 2.3基本设计概念和处理流程

基本设计概念：

设计简单、方便操作的用户操作流程。

处理流程：

购物流程：



上传商品流程：



下架商品流程：



申诉流程：



## 2.4结构

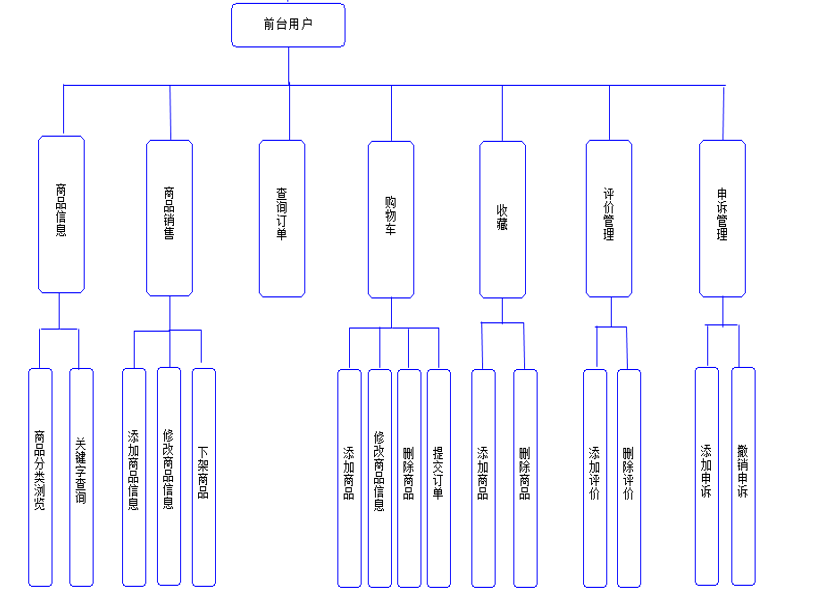
### 2.4.1总体框图





### 2.4.2HIPO图

H图：



IPO图：

## 2.5系统模块

首页模块

分类模块

购物车模块

我的模块

## 2.6功能需求与系统模块的关系

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 首页模块 | 分类模块 | 购物车模块 | 我的模块 |
| 实名  认证 |  |  |  |  |
| 申诉  系统 |  |  |  | √ |
| 关键  字查  询 | √ | √ |  |  |
| 收藏 | √ | √ | √ | √ |
| 购物  车 | √ | √ | √ |  |
| 提交/取消  订单 | √ | √ | √ |  |
| 评价  系统 | √ | √ | √ | √ |
| 上传/修改/下架  商品 |  |  |  | √ |

## 2.7人工处理过程

个人资料模块：个人资料填写。

实名认证模块：实名信息填写。

商品上传模块：商品信息填写。

评价管理模块：评价内容填写。

意见反馈模块：反馈内容审核。

## 2.8尚未解决的问题

各项功能的具体实现方法。

# 3．接口设计

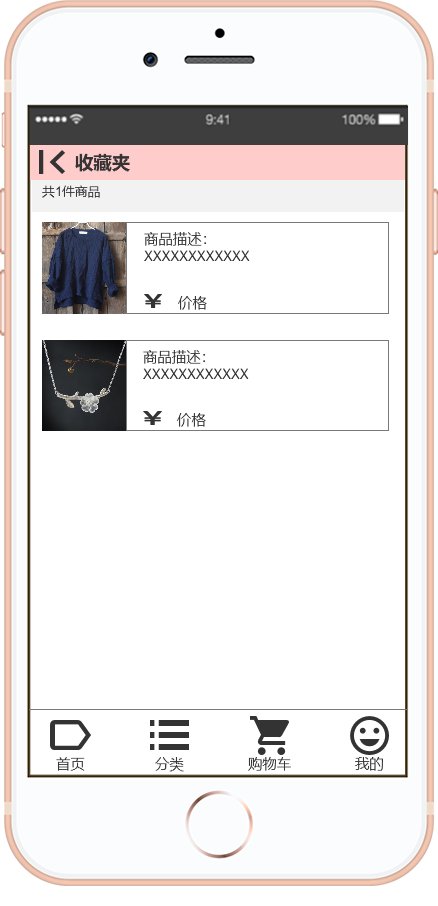
## 3.1用户接口

我们采用的是图形接口:图形用户接口采用了图形化的操作界面，用非常容易识别的各种图标来将系统各项功能、各种应用程序和文件，直观、逼真地表示出来。用户可通过鼠标、菜单和对话框来完成对应程序和文件的操作。图形用户接口元素包括窗口、图标、菜单和对话框，图形用户接口元素的基本操作包括菜单操作、窗口操作和对话框操作等。











提供给用户操作的硬件控制面板的定义是我们的主菜单，用户可以根据主菜单内显示的相关选项进行相应的操作。

## 3.2外部接口

软件与硬件之间的接口：

wx.request 小程序客户端与服务器端交互的接口，也用作网络请求；

wx.downloadFile 下载文件接口；

wx.uploadFile 上传文件接口；

本系统与各支持系统之间的接口关系：通过微信小程序获取用户信息接口。

## 3.3内部接口

跳转；

当鼠标点击跳到另一个页面：

wx.navigateTo；

把文件中的内容导出：

module.exports；

页面跳转：

wx.redirectTo；

显示弹窗：

wx.showToast；

使用wafer-node-sdk导出的mysql访问接口。

# 4．运行设计

## 4.1运行模块组合

首页、分类、购物车、个人中心。

## 4.2运行控制

登录:调用wx. getUserInfo获取用户信息

修改个人信息：调用UserInfo接口修改用户信息

浏览主界面：调用wx.showActionSheet接口显示操作菜单

查询：给商品表定义一个 keyword 数组类型字段，用于这种查询，在用户点击搜索后，把用户输入的「内容」作为查询条件添加到查询中，那么我们就会得到一个搜索结果列表。

购物车：

计算总价：总价 = 选中的商品1的 价格 \* 数量 + 选中的商品2的价格 \* 数量 + …

**选择事件：**点击时选中，再点击又变成没选中状态，其实就是改变 selected 字段。通过 data-index=”{{index}}” 把当前商品在列表数组中的下标传给事件。

**全选事件**

全选就是根据全选状态 selectAllStatus 去改变每个商品的 selected

**增减数量**

点击+号，num加1，点击-号，如果num > 1，则减1

**删除商品**

点击删除按钮则从购物车列表中删除当前元素，删除之后如果购物车为空，改变购物车为空标识hasList为false

生成订单：若交易成功则调用wx.showToast接口显示交易完成，否则显示交易失败

## 4.3运行时间

暂无。

# 5．系统数据结构设计

## 5.1逻辑结构设计要点

收货地址（地址编号，地址内容，收货人，用户编号）外码：用户编号

商品信息（商品编号，用户编号，是否首页循环，单价，商品名，图片1，图片2，图片3，图片4，二级分类编号，数量），外码：用户编号

订单信息（订单编号，收货地址，单价，生成时间，商品名，数量，用户手机号，总价，商品编号，用户编号），外码：用户编号,商品编号

用户（用户编号，小程序编号，姓名，性别，职业，身份证号）外码：小程序编号

评价（评价编号，评价时间，评价内容，商品编号，用户编号，评价时间），外码：用户编号，商品编号

收藏夹（收藏编号，用户编号，商品编号）外码：商品编号，用户编号

意见反馈（反馈编号，反馈人姓名，手机号，反馈内容，反馈时间）外码：商品编号

购物车（购物车编号，数量，价格，商品编号，用户编号）外码：商品编号，用户编号

一级分类（一级分类编号，一级分类名）

二级分类（二级分类编号，二级分类名，缩略图，所属一级分类编号）外码：所属一级分类编号

管理员（管理员编号，管理员昵称，密码，管理员名字）

## 5.2物理结构设计要点

商品信息表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id | 商品编号 | int | 11 | 否 | 主键 |
| userid | 用户编号 | int | 11 | 否 |  |
| isLb` | 是否首页循环 | int | 11 | 否 |  |
| price | 单价 | double |  | 否 |  |
| title` | 商品名 | varchar | 255 | 否 |  |
| url1` | 图片1 | varchar | 255 | 否 |  |
| url2` | 图片2 | varchar | 255 | 否 |  |
| url3 | 图片3 | varchar | 255 | 否 |  |
| url4 | 图片4 | varchar | 255 | 否 |  |
| ppId` | 二级分类编号 | int | 11 | 否 |  |
| num` | 数量 | int | 11 | 否 |  |

收货地址表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 地址编号 | **int** | **11** | **否** | **主键** |
| dz` | 地址内容 | varchar | **255** | **否** |  |
| name` | 收货人 | varchar | **255** | **否** |  |
| phone` | 手机号 | varchar | **255** | **否** |  |
| userId` | 用户编号 | int | **11** | **否** |  |

购物车表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 购物车编号 | int | **11** | **否** | **主键** |
| num` | 数量 | int | **11** | **否** |  |
| price` | 价格 | double |  | **否** |  |
| goodsId` | 商品编号 | int | **11** | **否** |  |
| userId` | 用户编号 | int | **11** | **否** |  |

订单表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 订单编号 | int | **11** | **否** | **主键** |
| address` | 收货地址 | varchar | **255** | **否** |  |
| cost` | 单价 | **double** |  | **否** |  |
| createTime` | 生成时间 | datetime |  | **否** |  |
| name` | 商品名 | varchar | **255** | **否** |  |
| num` | 数量 | int | **11** | **否** |  |
| phone` | 用户手机号 | varchar | **255** | **否** |  |
| price` | 总价 | double |  | **否** |  |
| goodsId` | 商品编号 | int | **11** | **否** |  |
| userId` | 用户编号 | int | **11** | **否** |  |

一级分类表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 一级分类编号 | int | **11** | **否** | **主键** |
| name` | 一级分类名 | varchar | **255** | **否** |  |

二级分类表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 二级分类编号 | int | 11 | **否** | **主键** |
| name` | 二级分类名 | varchar | 255 | **否** |  |
| cpUrl` | 缩略图 | varchar | 255 | **否** |  |
| lbId` | 所属一级分类编号 | int | 11 | **否** |  |

管理员表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 管理员编号 | int | **11** | **否** | **主键** |
| name` | 管理员昵称 | varchar | **255** | **否** |  |
| passWord` | 密码 | varchar | **255** | **否** |  |
| realName` | 管理员名字 | varchar | **255** | **否** |  |

意见反馈表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 反馈编号 | int | 11 | **否** | **主键** |
| name` | 反馈人姓名 | varchar | 255 | **否** |  |
| phone` | 手机号 | varchar | 255 | **否** |  |
| content` | 反馈内容 | varchar | 255 | **否** |  |
| addTime` | 反馈时间 | datetime | 255 | **否** |  |

评价表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 评价编号 | int | 11 | **否** | **主键** |
| addTime` | 评价时间 | datetime |  | **否** |  |
| content` | 评价内容 | varchar | 255 | **否** |  |
| goodsId` | 商品编号 | int | 11 | **否** |  |
| userId` | 用户编号 | int | 11 | **否** |  |
| addTimes` | 评价时间 | varchar | 255 | **否** |  |

用户表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 用户编号 | int | **11** | **否** | **主键** |
| openId` | 小程序编号 | varchar | **255** | **否** |  |
|  | 姓名 | Variable characters | 10 | **否** |  |
|  | 性别 | Variable characters | 2 | **否** |  |
|  | 职业 | Variable characters | 20 | **否** |  |
|  | 身份证号 | int | 18 | **否** |  |

收藏表结构

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **字段名** | **说 明** | **类 型** | **长度** | **可否为空** | **约束** |
| id` | 收藏编号 | int | **11** | **否** | **主键** |
| userId` | 用户编号 | int | **11** | **否** |  |
| goodsId` | 商品编号 | int | **11** | **否** |  |

## 5.3数据结构与程序的关系

服务器程序在对加入购物车/加入收藏夹/查看用户信息/查看订单信息/修改用户信息/评价/申诉进行操作时都需要对数据库中的所有表，进行联合查询、修改物理数据结构主要用于各模块之间函数的信息传递。接口传递的信息将是以数据结构封装了的数据。以参数传递或返回值的形式在各模块间传输，出错信息将送入显示模块中。

## 5.4数据库设计

|  |  |  |  |
| --- | --- | --- | --- |
| **编号** | **数据表名称** | **类 型** | **内容** |
| 1 | 商品信息 | 主表 | 记录商品信息 |
| 2 | 收货地址 | 辅助表 | 记录用户收货地址 |
| 3 | 购物车 | 辅助表 | 记录加入购物车的订单数量和价钱 |
| 4 | 订单信息 | 辅助表 | 记录订单商品详细信息 |
| 5 | 用户 | 主表 | 记录用户信息 |
| 6 | 评价 | 主表 | 记录评价信息 |
| 7 | 收藏夹 | 辅助表 | 记录收藏的商品详细信息 |
| 8 | 一级分类 | 辅助表 | 记录相关分类下的商品信息 |
| 9 | 二级分类 | 辅助表 | 记录相关分类下的商品信息 |
| 10 | 管理员 | 主表 | 记录管理员信息 |

# 6．系统出错处理设计

## 6.1测试计划

针对主要功能进行测试。

**用户功能：**

|  |  |  |
| --- | --- | --- |
|  | **操作成功** | **操作失败** |
| **实名认证** | **成功提示+用户信息表显示该用户信息** | **失败提示+用户信息表未显示该用户信息** |
| **申诉系统** | **成功提示+申诉表显示该申诉信息** | **失败提示+申诉表未显示该申诉信息** |
| **关键字查询** | **显示包含该关键字的所有商品内容** | **显示其他内容** |
| **收藏** | **收藏夹显示收藏商品+收藏夹表显示商品信息** | **收藏夹未显示收藏商品或收藏夹表未显示商品信息** |
| **购物车** | **购物车显示添加的商品信息+购物车表显示商品信息** | **购物车未显示添加的商品信息或购物车表未显示商品信息** |
| **提交订单** | **成功提示+订单表显示订单信息** | **失败提示+订单表未显示订单信息** |
| **评价系统** | **商品浏览界面显示评价信息+评价表显示评价信息** | **商品浏览界面未显示评价信息或评价表未显示评价信息** |
| **查询订单信息** | **显示所有订单信息** | **未显示所有订单信息** |
| **上传商品** | **商品浏览界面显示该商品+商品信息表显示商品信息** | **商品浏览界面未显示该商品或商品信息表未显示商品信息** |
| **修改商品信息** | **商品浏览界面显示该商品修改后的信息+商品信息表显示商品修改后的信息** | **商品浏览界面未显示该商品修改后的信息+商品信息表未显示商品修改后的信息** |
| **下架商品** | **商品浏览界面不显示该商品修改后的信息+商品信息表不显示商品修改后的信息** | **商品浏览界面显示该商品修改后的信息+商品信息表显示商品修改后的信息** |

## 6.2出错信息

由于输入信息不符合规范(如：商品数量为负数等)，称之为软错误

由于硬件方面的错误(如：网络传输超时、硬件出错等)，称之为硬错误；

对于一些关键的操作(如：删除操作) 应该提供提示确认机制；

对于数据、测试文档，都是要提供相应的保密措施设置。

|  |  |  |
| --- | --- | --- |
| 错误类型 | 子项 | 错误原因 |
| 数据库错误 | 连接 | 连接超时 |
| 连接断开 |
| 数据库本身 | 数据库代码出错 |
| 数据库溢出 |
| TCP连接错误 | 连接 | 连接超时 |
| 连接断开 |
| 其他TCP错误 | Socket自身错误 |
| 系统部分自定义错误 | 权限错误 | 管理员权限设置故障 |
| 输入错误 | 身份证号位数错误/为空 |
| 院系班级错误/为空 |
| 手机号码错误/为空 |
| 查找错误 | 为查找到符合要求的记录 |
| 链接错误 | 内部链接错误 | 网页链接错误 |
| 文件/图片链接错误 |
| 外部链接错误 | 页面休整或者其他错误无法打开 |
| 链接网址维护或其他错误无法打开 |

## 6.3补救措施

对于软错误，需要在添加/修改操作中及时对输入数据进行验证，分析错误的类型，并且给出相应的错误提示语句，传送到客户端上。

对于硬错误,错误类型比较少而且比较明确，所以可以在可能出错的地方中输出相应的出错语句，并将程序重置，最后返回输出阶段。

1. 后备技术：周期性的把磁盘信息记录在案。
2. 降效技术：限制用户数量。
3. 恢复及再启动技术：

Ⅰ.检查点设置与恢复；

Ⅱ.重新启动程序。

## 6.4系统维护设计

数据库维护：

定期备份数据库，定期检测数据库的一致性，定期查看操作日志等；

文件方面：对于下架商品等，定期删除相关文件，减少数据量。