# 软件需求规格说明书

# 1范围

## 1.1标识

标识号：YH-ZQ-QP-122

名称：校园二手交易系统

缩略名：UstSystem

版本号：1.0.0

发布号：1.0.0.181220\_Alpha

## 1.2系统概述

对于用户而言，该系统是为了方便校园学生进行低价的有价值的物品转移。

校园二手交易系统可以更好的帮助用户买家查询到该系统的所有物品，并且可以浏览该物品的大概信息，以低廉的价格来获得对自己有用的物品。

校园二手交易系统可以更好的帮助用户卖家很好的出售自己无用的东西来补贴自己生活。

## 1.3文档概述

在信息化高速发展的今天，价值利用这样的名词正主导着人们的生活和发展, 有必要为物质需求增长的人们开发校园二手交易系统。提高物品利用率，从而提高管理效率。本文档具体对校园二手交易系统的软件需求等进行基本分析，及需求，故在此针对本系统编写此文档，

## 1.4基线

通过系统功能有效的解决快速准确获得书籍的途径、确定该系统基本功能，本文档的最终解释权在本小组手中，请勿随意更改。本文档的设计基线是《GJB438B-2009 军用软件开发文档通用要求》。

# 2 引用文档

[1]GJB438B-2009 军用软件开发文档通用要求

[2]Y .Daniel Liang著李娜译 ,JAVA语言程序设计．北京：机械工业出版社 2012

[3]刘先锋 ,数据库系统原理与应用.武汉：华中科技大学出版社2012

[4]谢希仁,计算机网络（第五版）．北京：电子工业出版社 2012

# 3 需求

## 3.1 所需的状态和方式

## 3.2 需求概述

## 3.2.1 目标

　在如今的大学校园中，伴随着学生的购买能力的提高和每年的升学和毕业，存在许多各种类型的二手商品。目前，二手商品交易成为了当代大学生课余生活的一大热门。以我校为例，每年都要举办的“跳蚤”市场，可见大学生对二手商品交易的需求，然而这种方式有很多局限性和偶然性，远远无法满足广大学子交易的需求。

建立一个校园二手交易平台可以大大方便在校的学生，方便了同学，也营造了节约光荣，浪费可耻的校园文化氛围。最主要的是，它也可以通过网络将自己不用的东西放在网上,也可在网上找到自己需要的东西,物美价廉,达到双赢。

该系统的主要功能是可以让学生对自己不需要的产品进行上架，而别的同学可以通过该系统找到自己喜欢的产品。

该系统的优势的可以让学生对自己想要的但是系统上没有的产品进行申请，系统会根据上架的产品对申请的产品进行匹配，如果匹配高的话，就把该商品推送给该学生。

### 3.2.2 运行环境

服务器软件 Apache Tomcat v8.5 Server

服务器操作系统 linux

Jdk版本 1.8

Mysql版本 5.7

### 3.2.3 用户特点

用户主要是在校大学生

### 3.2.4 关键点

关键功能：本项目是向用户推荐指定的商品是通过用户向系统提交自己想要的商品，从而系统可以给用户进行特定或者相似商品的反馈。

关键算法：通过聚类算法对用户的请求进行分类，然后当有新的相关产品提交的时候可以通过分析该产品的属性，看他属于哪种分类的请求，之后给该分类下的请求用户发送通知。

### 3.2.5 约束条件

开发时间限制为15天

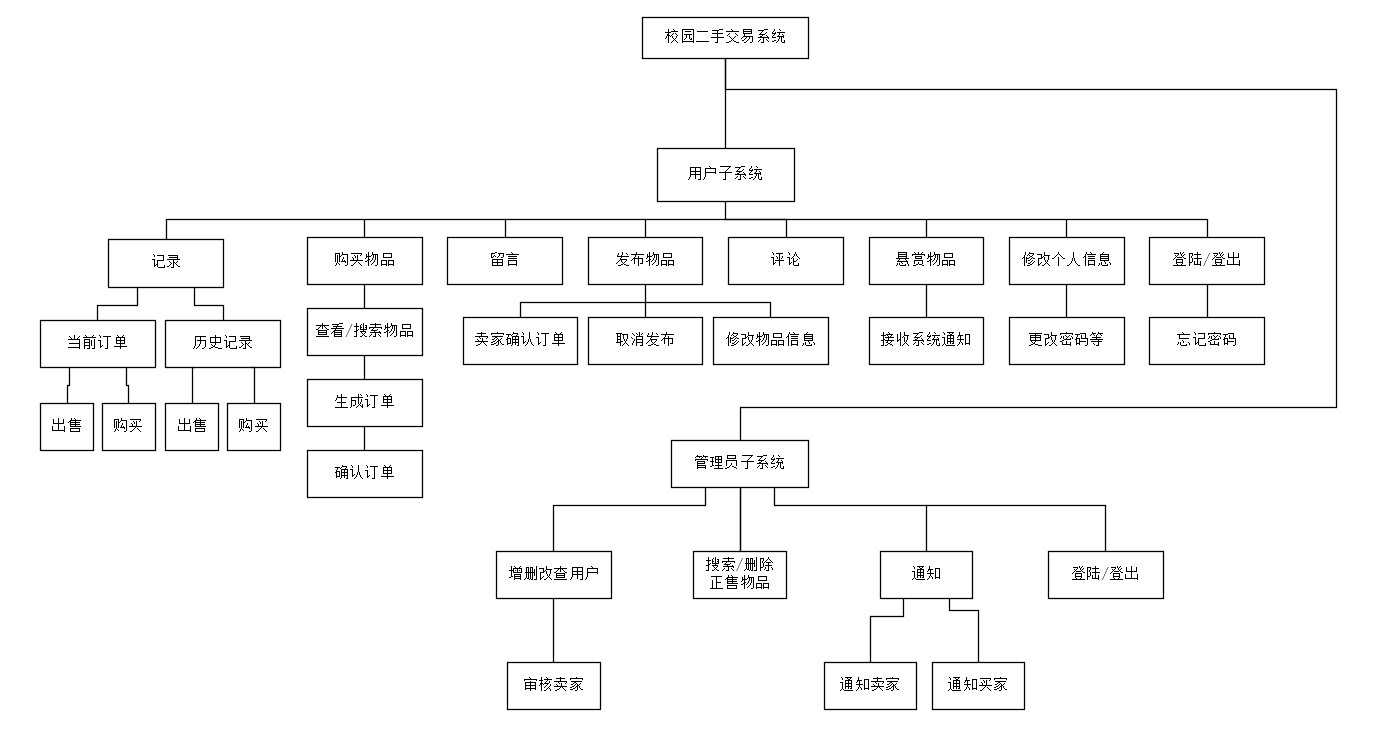
所采用的开发框架为springboot框架

开发的数据库为mysql

用户上架的产品不能是违反当地法律的违规产品

## 3.3需求规格

### 3.3.1软件系统总体功能/对象结构



### 3.3.2软件子系统功能/对象结构

### 3.3.3描述约定

### 3.4.1注册登陆

#### 3.4.1.1说明和优先级

用户第一次进入网站后可以选择进行注册，通过输入唯一的账户和密码进行注册；拥有账户的用户可以通过输入账户和密码进行登陆，进入主页。优先级为高。

#### 3.4.1.2激励/响应序列

刺激：用户申请注册账户。

响应：系统为新用户注册账户并跳转页面。

刺激：用户输入正确的账户和密码，并登陆。

响应：系统匹配账户和密码，跳转页面。

刺激：用户输入不正确的账户和密码，并登陆。

响应：系统匹配账户和密码，提示用户账户或密码错误。

#### 3.4.1.3功能需求

|  |  |
| --- | --- |
| User.Login | 用户输入正确的账号、密码可以登录到校园二手交易系统 |
| User.Login.Failure | 用户输入错误的账号、密码，系统提示顾客输入错误，重新输入 |
| User.Signup | 用户输入自定义账户系统检测重复，然后输入密码和确认密码，然后进行注册 |

### 3.4.2浏览功能

#### 3.4.2.1说明和优先级

用户可以在主页浏览产品和寻找想要的产品，优先级为高；

用户可以通过关键字搜寻想要找的产品，优先级为高

用户可以举报违规产品，优先级为低；

用户可以查看单独物品详细信息，优先级为高；

用户可以查看卖家所卖东西的记录，优先级为中；

用户可以查看东西的评价，优先级为低；

#### 3.4.2.2激励/响应序列

刺激：用户请求进入主页。

响应：系统向用户返回主页并显示首页的产品。

刺激：用户输入关键字向系统请求包含关键字的产品信息

响应：系统向用户返回包含关键字的产品信息

刺激：用户向系统请求举报违规产品

响应：系统询问用户是否确定举报

刺激：用户选择确定举报

响应：系统向用户返回举报成功信息

刺激：用户选择取消举报

响应：系统返回原来的产品界面

刺激：用户选择一样物品，向系统请求物品的详细信息

响应：系统向用户返回物品的详细信息

刺激：用户选择一个卖家，向系统请求卖家的记录

响应：系统向用户返回卖家的记录

刺激：用户向系统请求商品的评价

响应：系统向用户返回商品的评价信息

#### 3.4.2.3功能需求

|  |  |
| --- | --- |
| User.Customer.Login | 用户输入正确的账号、密码和验证码登入到系统的首页界面 |
| User.Customer.Login.Failure | 用户输入错误的账号、密码或验证码，系统提示输入错误消息并要求用户重新输入 |
| User.Customer.Browse | 用户向系统请求访问主页，系统向用户返回首页并显示产品 |
| User.Customer.Serachh | 用户输入关键字搜寻带有关键字的产品 |
| User.Customer.Report  User.System.Report.Confirm  User.Customer.Report.Confirm.Yes  User.Customer.Report.Confirm.No | 用户向系统举报违规产品  系统询问用户是否确定举报  用户确定举报  用户取消举报 |
| User.Customer.Detail | 用户选择一件商品查看详细信息 |
| User.Customer.SellerRecords | 用户选择一个卖家查看卖家记录 |
| User.Customer.Comment | 用户选择一件商品查看商品评价 |

### 3.4.3查看历史记录

#### 3.4.3.1用户可查看所卖东西记录

##### 3.4.3.1.1说明和优先级

用户可以在用户页面查看自己卖东西的记录，优先级为低。

##### 3.4.3.1.2激励/响应序列

刺激：用户请求查看自己卖东西的记录

响应：系统向用户展示卖东西的记录

刺激：用户没有卖出过东西

响应：系统提示用户没有卖出过东西

##### 3.4.3.1.3功能需求

|  |  |
| --- | --- |
| User.Customer.CheckSoldHistory | 登录系统的用户可以点击按钮请求查看自己卖东西的记录 |
| User.Customer.CheckSoldHistory.Notice | 如果自己没有卖过任何东西，则提示用户没有卖过任何东西 |
| User.Customer.CheckSoldHistory.show | 系统向用户展示卖东西的记录 |

#### 3.4.3.2用户可查看所买东西记录

##### 3.4.3.2.1说明和优先级

用户可以在用户页面查看自己买东西的记录，优先级为低。

##### 3.4.3.2.2激励/响应序列

刺激：用户请求查看自己买东西的记录

响应：系统向用户展示买东西的记录

刺激：用户没有买过东西

响应：系统提示用户没有买过东西

##### 3.4.3.2.3功能需求

|  |  |
| --- | --- |
| User.Customer.CheckBuyHistory | 登录系统的用户可以点击按钮请求查看自己买东西的记录 |
| User.Customer.CheckBuyHistory.Notice | 如果自己没有买过任何东西，则提示用户没有卖过任何东西 |
| User.Customer.CheckBuyHistory.show | 系统向用户展示买东西的记录 |

#### 3.4.3.3用户可浏览卖家记录

#### 3.4.3.3.1说明和优先级

用户可以在浏览叶敏查看卖家卖东西的记录，优先级为低。

#### 3.4.3.3.2激励/响应序列

刺激：用户请求查看卖家卖东西的记录

响应：系统向用户展示卖家卖东西的记录

刺激：卖家没有卖东西的记录

响应：系统提示用户卖家没有卖出过东西

#### 3.4.3.3.3功能需求

|  |  |
| --- | --- |
| User.Seller.CheckSoldHistory | 登录系统的用户可以在浏览页面点击按钮请求查看卖家卖东西的记录 |
| User.Seller.CheckSoldHistory.Notice | 如果卖家没有卖过任何东西，则提示卖家没有卖过任何东西 |
| User.Seller.CheckSoldHistory.show | 系统向用户展示卖东西的记录 |

### 3.4.4个人信息修改

#### 3.4.4.1用户可修改自己的个人信息

##### 3.4.4.1.1说明和优先级

用户可以在用户页面修改自己的个人信息，优先级为低。

##### 3.4.4.1.2激励/响应序列

刺激：用户请求修改自己的个人信息

响应：系统向用户提供修改自己个人信息的页面

刺激：用户确认修改自己的个人信息

响应：修改生效

刺激：用户取消修改自己的个人信息

响应：修改无效

##### 3.4.4.1.3功能需求

|  |  |
| --- | --- |
| User.Customer.CheckInfor | 用户登录后可以请求查看自己的个人信息 |
| User.Customer.CheckInfor.show | 系统向用户展示个人信息 |

#### 3.4.4.2用户可以查看个人信息

##### 3.4.4.2.1说明和优先级

用户可以在用户页面查看自己的个人信息，优先级为低。

##### 3.4.4.2.2激励/响应序列

刺激：用户请求查看自己的个人信息

响应：系统向用户提供显示自己个人信息的页面

##### 3.4.4.2.3功能需求

|  |  |
| --- | --- |
| User.Customer.modifyInfor | 用户登录后可以请求修改自己的个人信息 |
| User.Customer.modifyInfor.Confirm | 用户修改完个人信息之后需要向系统确认是否修改 |
| User.Customer.modifyInfor.no | 用户取消修改，修改不起效 |
| User.Customer.modifyInfor.yes | 用户确认修改，修改生效。 |

### 3.4.5 用户发布产品和取消产品功能

#### 3.4.5.1 说明和优先级

当用户去发布产品的时候，将提供给用户发布产品界面，并且在其中填写相关信息，优先度较低。用户可以在发布产品的时候和自己的发布产品界面取消自己发布的产品，优先级较高。

#### 3.4.5.2 激励/响应序列

刺激： 用户申请发布产品

响应： 系统回应用户响应，并跳转到发布界面

刺激： 用户申请取消自己发布的产品

响应： 系统回应用户响应，并取消已发布的产品

刺激： 用户申请修改自己已发布的产品的信息

响应： 系统收到用户的请求，并且把用户修改的信息显示到界面上

#### 3.4.5.3 功能需求

|  |  |
| --- | --- |
| User.Seller.Applaction | 用户申请发布产品 |
| User.Seller.Responed | 系统回应用户响应 |
| User.Seller.Cancel | 用户取消发布产品 |
| User.Seller.Confirm | 用户确认发布产品 |
| User.Seller..Modify | 用户修改产品信息 |
| User.Seller.Show | 系统收到用户修改信息，并将修改后的信息显示到页面上 |

### 3.4.6购买产品功能

#### 3.4.6.1说明和优先级

当顾客想要交易某个物品后，还想继续浏览物品，可以把之前的心仪的物品加到购物车里，选择完毕之后确认生成订单，卖家会受到信息通知，并联系买家。

#### 3.4.6.2刺激响应序列

刺激：点击添加入购物车

响应：该物品加入购物车数量加一

刺激：确认生成订单

响应：系统减少物品数，并通知卖家买家的信息

刺激：删除产品

响应：购物车产品数量减少至0时，该物品消失

刺激：取消订单

响应：购物车清空

#### 3.4.6.3功能需求

|  |  |
| --- | --- |
| Buy.Add | 购物车物品数量添加 |
| Buy.Ensure | 系统生成订单 |
| Buy.cancel | 购物车清空 |
| Buy.Receive | 系统通知卖家买家信息 |

### 3.4.7信息功能

#### 3.4.7.1说明和优先级

当顾客在购物过程中需要与商家进行交流时，可以通过这个功能进行留言，双方可通过回复各自留言进行交流。优先级为中。

#### 3.4.7.2 刺激/响应序列

刺激：顾客向商家留言

响应：系统将顾客留言内容发送给商家

刺激：商家回复顾客留言

响应：系统将商家回复内容发送给顾客

刺激：顾客回复商家留言

响应：系统将顾客回复内容发送给商家

#### 3.4.7.3功能需求

|  |  |
| --- | --- |
| Message.Customer.Message | 在顾客提出留言申请，系统为顾客提供留言功能，将留言发送给对应商家。 |
| Message.Business.Respond | 商家在接受到顾客留言提出回复申请，系统为商家提供回复功能，并将回复发送给对应顾客。 |
| Message.Customer.Respond | 顾客在接受到商家留言提出回复申请，系统为顾客提供回复功能，并将回复发送给对应商家。 |

### 3.4.8 评论功能

#### 3.4.8.1 说明和优先级

当用户购买商品后可以对卖家的商品进行评价，用户也可对曾经购买过的商品的评价进行修改。

#### 3.4.8.2 激励/响应序列

刺激：顾客购买商品

响应：系统提示顾客可以对商品进行评价

刺激：顾客可以对以前的评价进行修改

响应：系统提示顾客可以进行相关的修改

#### 3.4.8.3 功能性需求

|  |  |
| --- | --- |
| Return.Display | 系统提示顾客可以对商品进行评价 |
| Return.Update | 顾客可以对以前的评价进行修改 |

### 3.4.9用户申请成为卖家

#### 3.4.9.1描述和优先级

该功能针对用户。

不是卖家的用户可以向系统管理员提交申请，申请成为卖家。

只有经过审批后具有卖家资格的用户才有权利发布想要出售的商品。

优先级为中。

#### 3.4.9.2刺激/响应序列

刺激：用户点击申请成为卖家按钮

响应：系统跳转到申请界面

刺激：用户在申请界面二次确认后确定提交

响应：系统生成关于该用户的申请信息，等待管理员审核，同时提示用户等待申请审核。

#### 3.4.9.3功能性需求

|  |  |
| --- | --- |
| ApplyFor.Submit | 用户可以提交申请，想要成为卖家 |
| ApplyFor. Submit.Success | 申请后系统告知用户已申请，等待审核 |
| ApplyFor.Notice | 在管理员审核之后，系统将通过或不通过的结果告知用户。 |

### 3.4.10 悬赏功能

#### 3.4.10.1 说明和优先级

当用户发现没有需要的二手商品的时候，可以在该系统上进行发布悬赏信息，如果有人在之后发布相应的商品的时候，系统可以根据关键词进行匹配，然后将相关的商品信息推送给该用户。优先级为高。

#### 3.4.10.2 刺激和响应序列

刺激：用户进行悬赏信息的发布。

响应：系统将该发布的信息进行保存在数据库中

刺激：系统收到新的发布商品

响应：系统将相匹配的商品通知给用户

#### 3.4.10.3 功能需求

|  |  |
| --- | --- |
| Reward.publish | 用户向系统发布自己的需求信息 |
| Reward.save | 系统将信息保存在数据库中 |
| Reward.save.getNews | 系统收到新的发布信息 |
| Reward.info | 系统将匹配的商品通知给用户 |

### 3.4.11 审核功能

#### 3.4.11.1 说明和优先级

作为一个用户，当他想从买家变成卖家的时候，他会向系统发出申请，而管理员可以收到用户发出的申请，管理员可以对用户发出的申请作出审核，来决定该用户时候可以成为一个用户。优先级为高。

#### 3.4.11.2 刺激和响应序列

刺激：用户向系统申请成为卖家

响应：管理员对申请进行审核确定

#### 3.4.11.3 功能需求

|  |  |
| --- | --- |
| Identify.recieve | 系统收到用户发出的申请 |
| Identify.makesure | 管理员对申请进行审核确定 |

### 3.4.12管理员增删改查用户和产品的功能

#### 3.4.12.1作为管理员可以对用户信息进行增删改查；

##### 3.4.12.1.1说明和优先级

管理员在登录系统后，可以进行对用户信息的增加、删除、修改、以及查看。优先级为高。

##### 3.4.12.1.2刺激/响应序列

刺激：用户请求管理员添加自己到二手交易系统。

响应：系统向用户请求需要注册的个人信息，包括姓名、学号、密码。

刺激：用户请求删除自己，或者用户违规

响应：系统弹出警示并提示用户确认

刺激：用户请求修改自己的信息

响应：系统向用户展示需要修改的信息

刺激：用户请求查看自己的信息

响应：系统展示出该用户的信息

##### 3.4.12.1.3功能性需求

|  |  |
| --- | --- |
| Add User | 管理员在登录系统后，可以添加一个用户 |
| Delete User | 管理员在登录系统后，可以添加一个用户 |
| Edit User | 管理员在登录系统后，可以修改一个用户的信息 |
| View User | 管理员在登录系统后，可以添加一个用户的信息 |

#### **3.4.12.2作为管理员可以对被举报的违规产品进行删除**；

##### 3.4.12.2.1说明和优先级

管理员在登录系统后，可以对被举报的违规产品进行删除。优先级为高。

##### 3.4.12.2.2刺激/响应序列

刺激：有违规的产品出现在二手交易系统中，并被用户举报。

响应：管理员删除掉违规的产品。

##### 3.4.12.2.3功能性需求

|  |  |
| --- | --- |
| Delete Illegal Products | 有违规的产品出现在二手交易系统中，并被用户举报, 管理员删除掉违规的产品 |

##### 3.4.12.3作为管理员可以通知卖家产品下架；

##### 3.4.12.3.1说明和优先级

管理员在登录系统后，可以通知卖家产品下架。

##### 3.4.12.3.2刺激/响应序列

刺激：用户的产品在二手交易系统过期。

响应：管理员通知卖家产品下架。

##### 3.4.12.3.3功能性需求

|  |  |
| --- | --- |
| Notify Seller of Products Off Shel | 用户的产品在二手交易系统过期后，管理员通知卖家产品下架 |

##### 3.4.12.4作为管理员可以通知买家举报成功；

##### 3.4.12.4.1说明和优先级

管理员在登录系统后，可以通知买家举报成功。

##### 3.4.12.4.2刺激/响应序列

刺激：用户举报了违规的产品。

响应：管理员通知买家举报成功。

##### 3.4.12.4.3功能性需求

|  |  |
| --- | --- |
| Inform The Buyer to Report Success | 用户举报了违规的产品后，管理员通知买家举报成功 |

#### **3.4.12.5作为管理员可以搜索已经发布的产品**；

##### 3.4.12.5.1说明和优先级

管理员在登录系统后，可以搜索已经发布的产品

##### 3.4.12.5.2刺激/响应序列

刺激：用户上传了产品。

响应：管理员可以搜索已经发布的产品。

##### 3.4.12.5.3功能性需求

|  |  |
| --- | --- |
| Search For Products That Have Already Been Released | 用户上传了产品后，管理员可以搜索已经发布的产品 |

### 3.4.13找回密码

#### 3.4.13.1说明和优先级

用户登陆时点击忘记密码，系统将发送消息给用户邮箱，用户通过邮箱内给定的链接更改密码。优先级为高。

#### 3.4.13.2激励/响应序列

刺激：用户点击忘记密码。

响应：系统发送更改密码的邮件给用户的邮箱。

刺激：用户通过邮件内链接进入更改密码页面并更改密码。

响应：提示用户密码更改成功。

#### 3.4.13.3功能需求

|  |  |
| --- | --- |
| User.ForgetPassword | 系统发送更改账户密码邮件给用户的邮箱。 |
| User.ChangePassword | 用户输入新密码，系统更改用户密码并提示更改成功。 |

## 3.5 外部接口需求

该项目的找回密码需要外部邮箱接口

## 3.6 CSCI内部接口需求

## 3.7 CSCI内部数据需求

此需求是定义CSCI内部数据的需求，内部数据库和数据文件的需求，如果所有的设计由开发者决定，这里只要说明即可。如果关于内部数据的所有决策都待留设计时考虑，那么应在此处如实陈述。

## 3.8 适应性需求

此需求规定CSCI和安装数据有关的需求以及不同操作下可能不同的操作参数需求。对于本项目，对于开发地点没有要求，开发环境为配有git、eclipse的电脑十台，操作参数要求每个人数据库一致、每个开发人员对应的git分支无误、设计函数接口一致。

## 3.9 保密性需求

也叫做安全性需求，开发人员在学校实验室开发，无意外人身伤害风险。对于机房电源、空调设备也无危险，本项目为网页版校园二手交易系统，所以本项目无任何直接风险。

## 3.10保密性和私密性需求

本需求指明保密性和私密性的CSCI需求，包括CSCI运行的保密性、提供的保密性或私密性的类型和程度、CSCI必须经受的保密性、CSCI必须遵守的保密性、认可准则。对于本项目，开发期间对于10名开发人员内部可见，包含数据库访问、git仓库上项目上传和更改、文档使用等。开发完毕后系统使用人员可以拥有自己的权限，但对于内部数据库访问、代码、文档修改和查看没有访问权。

## 3.11 CSCI环境需求

硬件设备：

1) PC机最低配置：2G Hz以上CPU；1G以上内存；1G以上硬盘空间；

软件设备：

1) JDK 1.6或以上版本；

2) Eclipse 4.2或以上版本。

3) git 2.18.0.或以上版本

网速要求：

100k/s或以上

## 3.12计算机资源需求

### 3.12.1计算机硬件需求

1. 该系统的操作将通过如下的Web浏览器来完成：基于Tridend内核、Gecko内核、Webkit内核的浏览器如Internet explorer 9及以上版本、Safari 3.2及以上版本、Firefox 38.0.5及以上版本、Opera15及以上版本、Chrome 30.0.1599.101及以上版本。
2. 该系统将运行在一个服务器中，该服务器运行当前由政府批准的Tomcat Server、Apache HTTP Server和Internet Information Services。
3. 该系统将运行在一个服务器中，该服务器的硬件平台为CPU：PentiumⅢ 500以上处理器；内存：64MB以上；硬盘：10Ｇ以
4. windows 7及以上版本操作系统的计算机，CPU:Intel Pentium II 350/Intel 赛扬 300/AMD K62以上处理器；内存：64MB以上；硬盘：6.4G以上 10台

### 3.12.2计算机硬件资源利用需求

1. 最大许可使用的处理器能力：4M
2. 存储器能力：内存64M以上，硬盘10G以上
3. 输入输出设备能力：100kps以上
4. 通信、网络设备能力：1M带宽以上

### 3.12.3计算机软件需求

1. 操作系统：要求window7及以上操作系统
2. 数据库管理系统：MYSQL
3. 通信/网络软件协议：HTTP，FTP，UDP
4. 开发软件:eclipse

## 3.12.4计算机通信需求

1. 数据传输速率：1M带宽以上
2. 要求的系统使用时间：2周以上
3. 响应时间限制：1S以内
4. 传送/接收数据容量：100M以下

## 3.13软件质量因素

1. 功能性：全部实现所需功能
2. 可靠性：生产正确、一致的结果
3. 可维护性：容易更正
4. 可移植性：MVC框架保证了可以移植
5. 可测试性：容易测试充分
6. 易用性：易于学习、使用
7. 灵活性：敏捷开发保证了可以适应需求变化

## 3.14设计和实现的约束

1. JDK版本：1.7
2. 数据库：MYsql
3. 要求采用HTML、JavaScript、CSS、jquery框架等技术编写前端程序。
4. 要求采用SpringBoot MVC框架编写后端程序。
5. 要求采用AxureRP进行用户界面设计。

## 3.15数据

## 3.16操作

## 3.17故障处理

a.多人登陆账户可能导致系统阻塞

多个用户同一时间对同一个商品进行买卖操作时会产生错误

b.多人登陆时提示人数过多稍后登录

多个用户同时操作商品时提示该商品暂时无法购买

## 3.18算法说明

## 3.19有关人员需求

## 3.20 安全设施需求

1. 当系统奔溃时，系统能把用户最末尚未保存的数据进行自动保存；
2. 允许用户进行数据备份和恢复，防止数据丢失带来的损失；
3. 系统能记录所有运行时发生的错误，包括本机错误和网络错误。将错误信息存为日志，此外日志保存用户的操作信息。

## 3.21 安全性需求

1. 每个用户设置的用户名不会相同；
2. 每个用户在第一次登录后，必须更改他的最初登录密码，最初的登录密码不能重用；
3. 每个用户设置的密码有较高的复杂性；
4. 系统应保证用户无法对数据进行直接修改，只能通过系统的接口进行，而接口的调用应该被用户的权限严格限制;
5. 任何恶意的数据修改都会被系统阻止，并记录；
6. 系统对重要数据进行非对称加密，如密码和重要的参数
7. 每个用户只拥有自己的权限，只能访问属于自己的数据；
8. 用户的登录受计算机系统访问控制策略的限制。

## 3.22 其他需求

1. 系统安装方便，易于维护；
2. 支持撤销/恢复操作；
3. 支持数据处理运行功能。

## 3.23 包装需求

## 3.24 需求的优先次序和关键程度

### 3.24.1 对用户重要的属性

1. 有效性
2. 工作日期间内，早上8到晚上6点系统的有效性至少达到99.5%，晚上6点到凌晨12点，系统的有效性达99.95%。
3. 高效性
4. 在系统预计的峰值负载下，至少25%的CPU能力和系统内存被保留用于备用。
5. 在过多用户同时在线时，性能不会下降超过20%。
6. 灵活性
7. 需要一个至少6个月产品支持经验的系统维护人员，在大于一个小时的时间内为系统增加一个新的模块，包括代码修改和测试。
8. 完整性
   1. 对涉及更改数据和查看个人数据的用户进行身份校验，严格区分顾客的权限、网店的权限、网店的权限和网购平台的权限；
   2. 拥有对应权限的账户可以修改对应的数据，系统会阻止其他越过权限的用户的操作；
   3. 系统能够撤回错误操作。
9. 可靠性
10. 由于软件故障引发的数据错误不超过0.01%。
11. 健壮性
12. 如果用户在保存数据前系统编辑器发生故障退出，下次用户恢复页面时，编辑器能恢复故障发生前1分钟对编辑文字所做的全部修改；
13. 系统具有一定的容错和抗干扰能力，对于非硬件故障或者非通信故障，系统有一定的忍受能力；
14. 当输入的数据无效或者丢失，并且该数据不是很重要，可以使用缺省的参数值进行数据的写入。
15. 易用性
16. 未使用过该系统的用户可以在5分钟内上手该系统的所有基本操作的布局，并能在10分钟内掌握所有的功能操作；
17. 系统能保证从用户那里获取的数据是被控制的，确保数据的完整性和安全性；
18. 系统对用户的操作具有提示功能。

### 3.24.2对开发者重要的属性

1. 可维护性
2. 程序维护人员可以在20小时或以内，对现有系统进行修改；
3. 函数调用嵌套层次不超过3层；
4. 每个软件模块都有必要的注释，并对向外提供的接口提供完整详细的注释，重要模块的注释和源代码语句比例至少为1:3。
5. 可测试性
6. 一个模块的最大循环复杂度不超过20。
7. 可移植性
   1. 底层的功能应该以通用的模块形式提供调用，当运行环境改变时，可以方便的进行移植。
8. 可重用性
   1. 重用系统种通用的模块，提供给各个子系统，当需要对系统进行扩展时，也可以通过重用的通用模块进行实现。

# 4.合格性规定

|  |  |
| --- | --- |
| 需求 | 规定 |
| 登陆，注册，找回功能 | 演示，审查 |
| 浏览功能 | 演示，审查 |
| 历史记录功能 | 演示，审查 |
| 个人信息修改功能 | 演示，审查 |
| 用户发布产品和取消产品功能 | 演示，审查 |
| 购买产品功能 | 演示，审查 |
| 信息功能 | 演示，审查 |
| 评价功能 | 演示，审查 |
| 用户申请成为卖家 | 演示，审查 |
| 悬赏功能 | 演示，审查 |
| 管理员审核用户能否成为卖家功能 | 演示，审查 |
| 管理员增删改查用户和产品功能 | 演示，审查 |
| 外部接口需求 | 测试，分析 |
| 安全性需求 | 分析 |
| 计算机资源需求 | 测试 |
| 软件质量因素 | 测试，分析 |
| 设计和实现约束 | 审查 |
| 人员需求 | 验收审查 |
| 培训需求 | 验收审查 |
| 软件保障需求 | 测试，分析 |
| 验收、交付和包装需求 | 审查 |
| 安全性需求 | 分析 |
| 计算机资源需求 | 测试 |
| 软件质量因素 | 测试，分析 |

# 5.需求可追踪性

在3中一对需求经行详细划分，按此逐层追踪。

# 6.尚未解决的问题

暂无