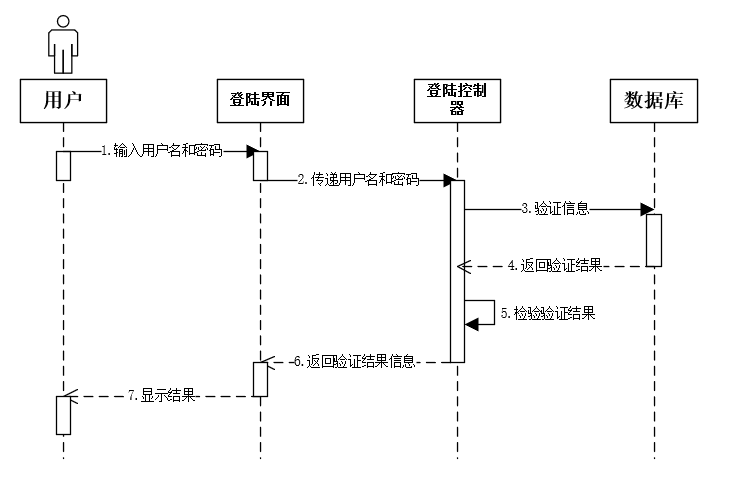
# 软件计划说明书

## 3.1登陆注册

### 3.1.1功能



登陆可分为用户登陆和管理员登陆，在登陆模块的描述中，将管理员和用户都看作系统的用户，他们根据自己的账户和密码登陆系统，才可进行以后的操作。若用户名和密码输入错误，不匹配，系统将会提示输入错误，并要求用户重新输入。新用户输入用户名密码进行注册。

### 3.1.2性能

用户的密码传输对安全性能要求很高，数据传输不能采用明文。系统对账户和密码的匹配并返回信息的时间不超过2s。

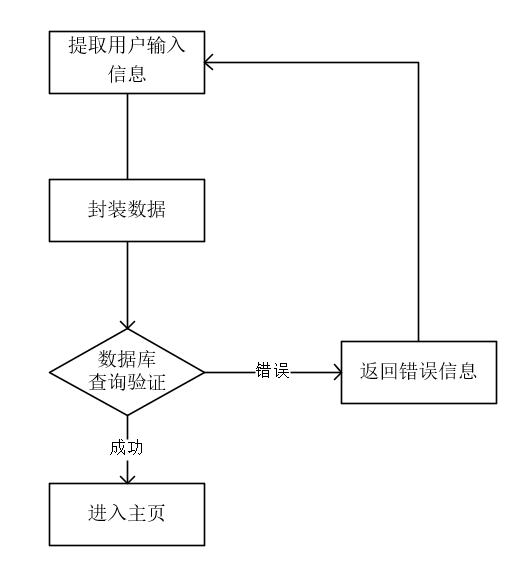
3.1.3输入项目

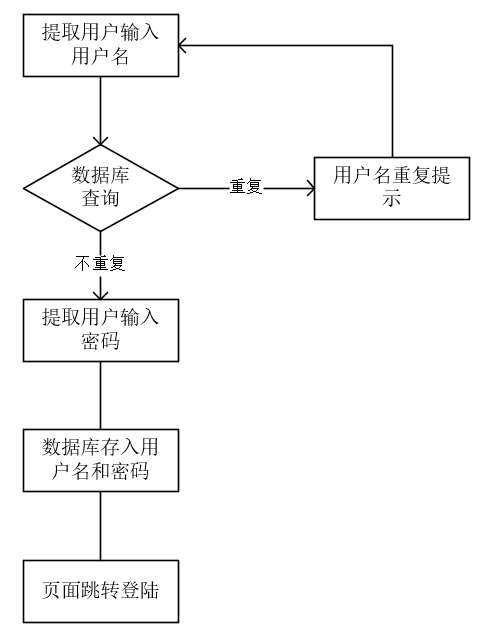
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 用户名 | Varchar |  |  |  | 主 |
| 密码 | Varchar |  |  |  |  |

### 3.1.4输出项目

根据数据库中存在的表中信息来查询，判断用户名、密码是否正确，正确则跳转到主页面，若用户名与密码不匹配则提示错误。

### 3.1.5流程图





### 3.1.6界面设计

从登陆界面可以点击注册进入注册界面。

从登陆界面可以点击忘记密码进入忘记密码的界面。

从注册界面可以返回登陆界面。

### 3.17接口设计

### 3.18算法设计

登陆时该模块通过读取前台信息，提取用户名及密码，将密码进行MD5加密，采用PreparedStatement生成SELECT语句解决SQL注入问题，到数据库查询信息，然后将结果返回给前台。

注册时该模块通过读取前台信息，提取用户名，采用PreparedStatement生成SELECT语句解决SQL注入问题，到数据库查询信息，然后将结果返回给前台，如果不重复再读取密码，将密码进行MD5加密，连同用户名存储进数据库

## 3.2浏览模块

### 3.2.1功能



浏览主页，查询商品



查看商品详细信息，举报商品



查看卖家详细信息

用户可以在主页浏览产品和寻找想要的产品；

可以通过关键字搜寻想要找的产品；

可以举报违规产品；

可以查看单独物品详细信息；

可以查看卖家所卖东西的记录；

可以查看东西的评价。

**3.2.2性能**

查询支持模糊查询；所有网页要求在5秒以内处理完成

**3.2.3输入项目**

所查询商品的关键信息

**3.2.4输出项目**

返回的商品关键信息

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 允许空 | 主/外键 |
| user\_id | int | 11 |  |  |
| goods\_name | varchar | 50 |  |  |
| description | Varchar | 255 |  |  |
| goods\_status | int | 11 |  |  |
| goods\_id | int | 11 |  | 主 |

### 3.2.5流程图



**3.2.6界面设计**

从首页在搜索框输入关键字，点击搜索，进入搜索页面

从首页或搜索页面，点击商品，进入商品详细信息页面

从商品详细信息页面，点击举报，进入举报页面

从首页或搜索页面或商品详细信息页面，点击卖家，进入卖家详细信息页面

**3.2.7接口设计**

**3.2.8算法**

3.3.1查看历史记录功能

3.3.1.1 功能

用户可以查看自己买东西的记录、用户可以查看自己卖东西的记录、用户可以查看卖家卖东西的记录。

3.3.1.2 性能

查询记录精确、单词查询任务要在3秒内完成，系统响应要在3秒以内。保证10台机器一起运行不会有高延迟。

3.3.1.3输入项目

账号、密码、点击相应的查看记录按钮。

3.3.1.4输出项目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| GoodsName | String | 64 |  |  |  |
| GoodsID | int | 4 |  |  |  |
| Number | Int | 4 |  |  |  |
| Description | String | 64 |  |  |  |

3.3.1.5流程图



图1 用户查看卖家或卖家记录流程图



图2 用户查看卖家记录流程图

3.3.1.6界面设计

见流程图

3.3.1.7接口设计

3.3.1.8算法设计

顾客点击对应的按钮，到数据库中查询是否有买/卖记录。如果没有记录，则返回没有记录的体型。如果有，后端根据用户名返回对应的记录。

3.4.1个人信息功能

3.4.1.1 功能

用户可以查看、修改自己的个人信息

3.4.1.2 性能

查看个人信息精确、单词查看个人信息任务要在3秒内完成，系统响应要在3秒以内。保证10台机器一起运行不会有高延迟。修改同步到数据库，修改精确，单词修改任务要在3秒的完成，系统响应要在3秒以内，保证式十台机器一起运行不会有高延迟，保证修改不发生冲突。

3.4.1.3输入项目

账号、密码、点击修改/查看按钮。

3.4.1.4输出项目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| User-name | String | 64 |  |  |  |
| User-ID | int | 4s |  |  |  |
| Phone | Int | 11 |  |  |  |
| E-mail | String | 64 |  |  |  |
| User-number | int | 10 |  |  |  |

3.4.1.5流程图



图3 用户查看修改个人信息流程图

3.4.1.6界面设计

见流程图

3.4.1.7接口设计

3.4.1.8算法设计

用户点击查看按钮，根据用户名到数据库中查看用户相关信息，由后台返回相应数据显示在前端。用户点击修改个人信息按钮，用户填入相应的数据再点击确认修改按钮，此时系统询问用户是否确认修改，如果点击确认，则修改同步至数据库，否则修改不生效。

## 3.10 悬赏功能

### 3.10.1 功能

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，系统会保存用户发布的二手信息。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。

### 3.10.2 性能

用户可以在该系统中发布所需要商品的信息，该功能的性能依赖于实现的算法。

### 3.10.3 输入项目

用户通过输入请求商品的名字和请求商品的类别，通过数据库封装在下面的表格中。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 是否为空 | 主/外键 |
| query\_id | int | 11 | 否 | 是 |
| user\_id | int | 11 | 否 | 否 |
| query\_name | varchar | 50 | 否 | 否 |
| category\_id | int | 11 | 否 | 否 |

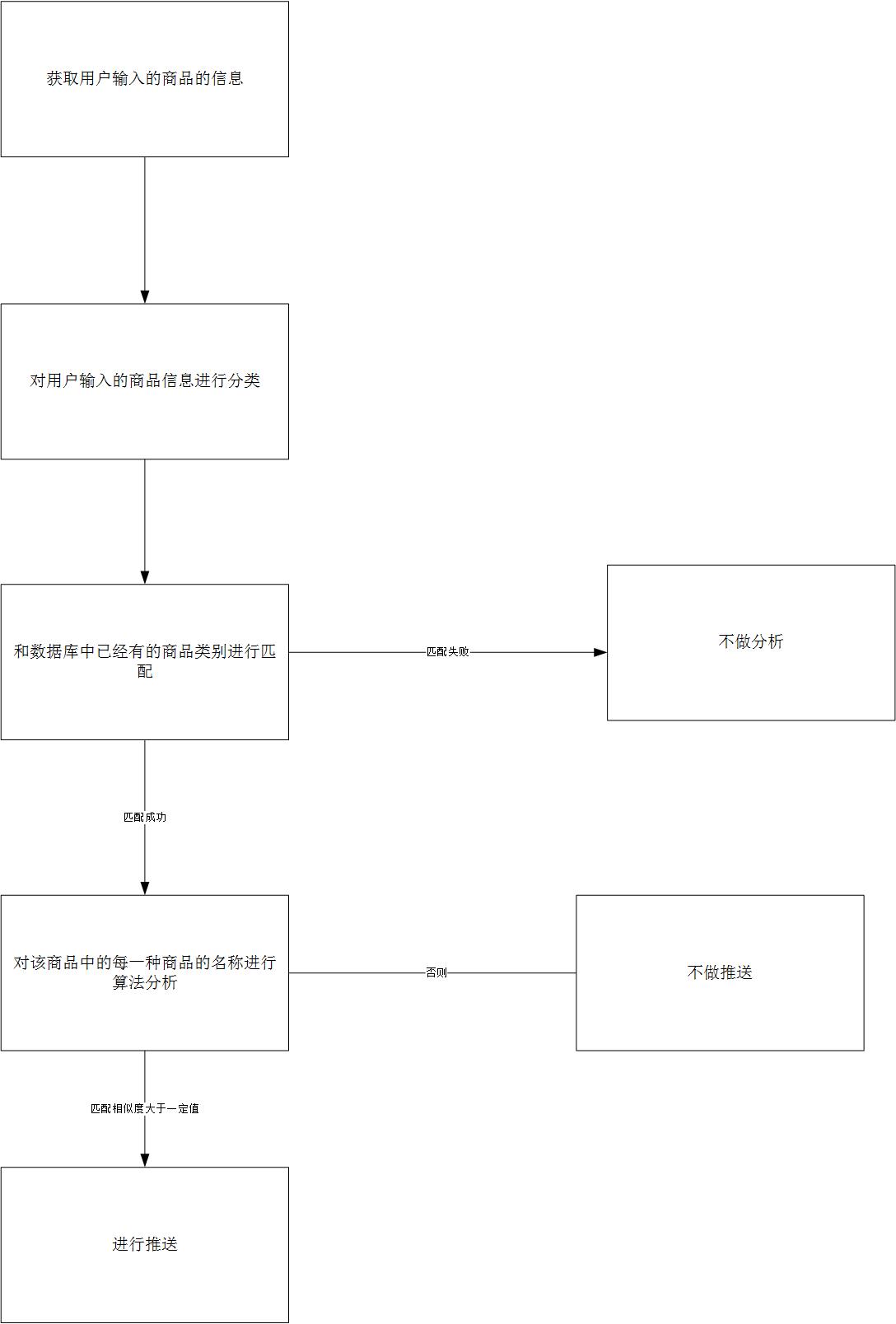
### 3.10.4 输出项目

系统根据输入的商品查看是否有相关的产品上架，如果有的话就给用户返回响应的商品

### 3.10.5 算法

该模块通过系统存储用户请求的商品分类，然后通过用户上架商品的分类进行匹配，如果是该分类的话，就在该分类中找寻是否有和请求相匹配的商品，通过聚类算法分析两种商品的匹配程度，当匹配程度大于一定值的时候，我们就认为这两种商品是同一种商品，之后系统会把两种商品认为是同一种商品，并推送给用户。

### s3.10.6 程序逻辑



### 3.11 审核功能

### 3.11.1 功能

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。

### 3.11.2 性能

审核需要系统管理员登录到该系统中，安全系数比较高，基本性能要求为系统响应时间不能超过1S.

### 3.11.3 输入项目

用户通过输入请求商品的名字和请求商品的类别，通过数据库封装在下面的表格中。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 是否为空 | 主/外键 |
| query\_id | int | 11 | 否 | 是 |
| user\_id | int | 11 | 否 | 否 |
| query\_name | varchar | 50 | 否 | 否 |
| category\_id | int | 11 | 否 | 否 |

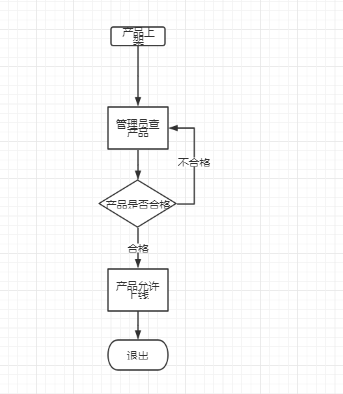
### 3.11.4 输出项目

管理员根据输入的商品查看是否有相关的产品上架，如果有的话就审核产品，若全部通过则审核成功。

### 3.11.5 算法

该模块通过上架的产品，管理员对其审核，若产品合格，则认为审核通过，如果产品不合格，则认为审核不通过。

### 3.11.6 程序逻辑



## 3.12管理员增删改查用户和产品的功能

### 3.12.1作为管理员可以对用户信息进行增删改查；

#### 3.12.1.1功能

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。

#### 3.12.1.2性能

响应时间不能超过1S，同时允许多个管理员进行操作。

#### 3.12.1.3输入项目

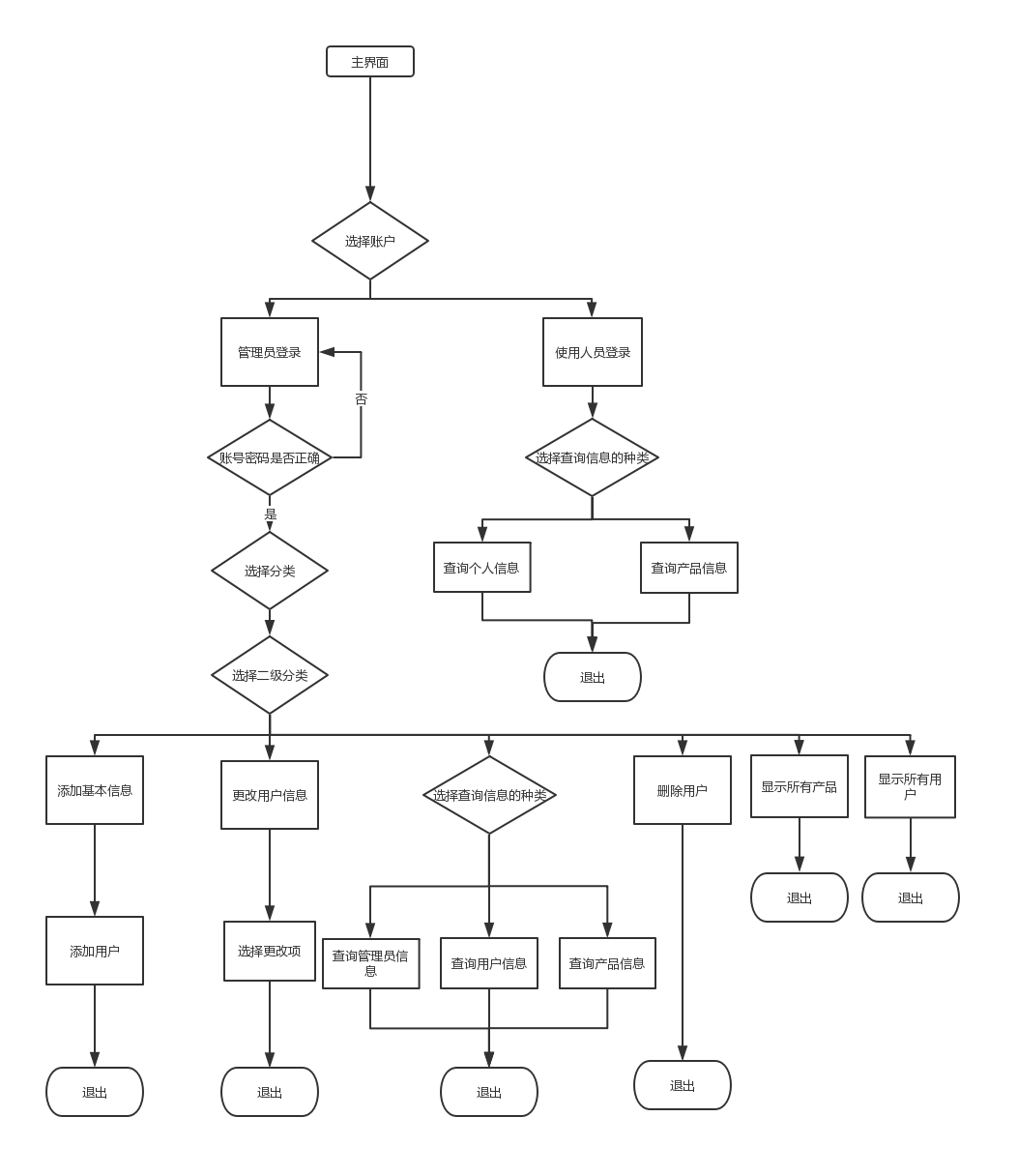
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 管理员账号 | Int | 11 |  | 不允许 | 主 |
| 管理员密码 | Varchar | 50 |  | 不允许 |  |

#### 3.12.1.4输出项目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 用户id | Int | 11 |  | 不允许 | 主 |
| 用户姓名 | Varchar | 50 |  | 不允许 |  |
| 用户密码 | Varchar | 50 |  | 不允许 |  |
| 电话 | Int | 11 |  | 不允许 |  |
| 邮箱 | Varchar | 50 |  | 不允许 |  |
| 用户状态 | Int | 11 |  | 不允许 |  |

根据用户的请求，管理员进行对用户信息的增加、删除、修改、以及查看。

#### 3.12.1.5流程图



#### 3.12.1.6界面设计

如上述流程图所示，点击主界面，选择账户类型，如果是管理员，进入管理员界面，进行对应的增删改查操作，比如进入到添加用户界面，弹出录入姓名、学号、电话等输入框，如果添加成功则跳转到显示用户界面。

#### 3.12.1.7接口设计

#### 3.12.1.8算法设计

管理员登陆系统后，获得用户的请求，比如获得用户的增加请求，系统给用户弹出信息录入界面，用户输入姓名、学号、电话等信息，如果输入非法字符，则提示输入错误，全部正确则录入成功，修改、删除、查看算法设计类似。

### 3.12.2作为管理员可以对被举报的违规产品进行删除

#### 3.12.2.1功能

管理员在登录系统后，可以进行对被举报的违规产品进行删除。

#### 3.12.2.2性能

响应时间不能超过1S，同时允许多个管理员进行操作。

#### 3.12.2.3输入项目

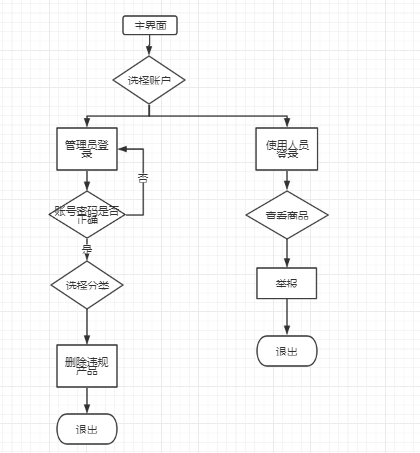
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 管理员账号 | Int | 11 |  | 不允许 | 主 |
| 管理员密码 | Varchar | 50 |  | 不允许 |  |

#### 3.12.2.4输出项目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 产品id | Int | 11 |  | 不允许 | 主 |
| 产品名字 | Varchar | 50 |  | 不允许 |  |
| 描述 | Varchar | 50 |  | 不允许 |  |
| 产品状态 | Int | 11 |  | 不允许 |  |
| 商品id | int | 11 |  | 不允许 |  |

用户举报违规产品后，管理员可以对其进行删除。

#### 3.12.2.5流程图



#### 3.12.2.6界面设计

如上述流程图所示，点击主界面，选择账户类型，如果是用户，进入用户界面，举报产品，接着跳入举报成功界面，管理员进入管理者界面，点击违规产品，将被举报的产品删除。

#### 3.12.2.7接口设计

#### 3.12.2.8算法设计

用户举报违规产品，管理员看到后，决定是否确实违规，如果着实违规，则删除产品，如果没有违规，则不删除该产品。

### 3.12.3作为管理员可以通知卖家产品下架

#### 3.12.3.1功能

管理员在登录系统后，可以通知卖家产品下架。

#### 3.12.3.2性能

响应时间不能超过1S，同时允许多个管理员进行操作。

#### 3.12.3.3输入项目

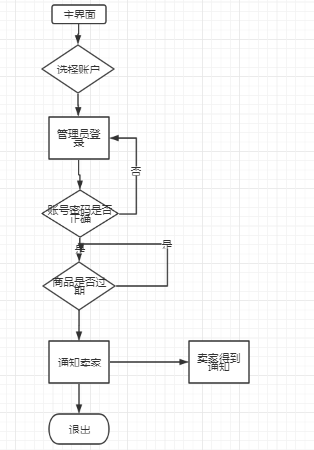
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 管理员账号 | Int | 11 |  | 不允许 | 主 |
| 管理员密码 | Varchar | 50 |  | 不允许 |  |

#### 3.12.3.4输出项目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 产品id | Int | 11 |  | 不允许 | 主 |
| 产品名字 | Varchar | 50 |  | 不允许 |  |
| 描述 | Varchar | 50 |  | 不允许 |  |
| 产品状态 | Int | 11 |  | 不允许 |  |
| 商品id | int | 11 |  | 不允许 |  |

用户举报违规产品后，管理员可以对其进行删除。

#### 3.12.3.5流程图



#### 3.12.3.6界面设计

如上述流程图所示，点击主界面，选择账户类型，管理员进入管理者界面，点击查看商品，如果商品过期，点击通知卖家商品过期，接着进入通知成功界面。

#### 3.12.3.7接口设计

#### 3.12.3.8算法设计

如果卖家的产品过期或库存没有，则管理者进入产品查看界面，点击对应的卖家，通知该产品下架，如果没有过期或库存依旧存在，则不做处理。

### 3.12.4作为管理员可以通知买家举报成功

#### 3.12.3.1功能

管理员在登录系统后，可以通知买家举报成功。

#### 3.12.3.2性能

响应时间不能超过1S，同时允许多个管理员进行操作。

#### 3.12.3.3输入项目

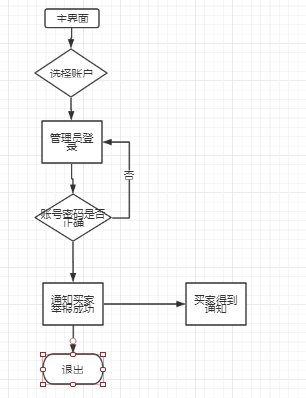
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 管理员账号 | Int | 11 |  | 不允许 | 主 |
| 管理员密码 | Varchar | 50 |  | 不允许 |  |

#### 3.12.3.4输出项目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 用户id | Int | 11 |  | 不允许 | 主 |
| 用户姓名 | Varchar | 50 |  | 不允许 |  |
| 用户密码 | Varchar | 50 |  | 不允许 |  |
| 电话 | Int | 11 |  | 不允许 |  |
| 邮箱 | Varchar | 50 |  | 不允许 |  |

用户举报违规产品后，管理员通知买家举报成功。

#### 3.12.3.5流程图



#### 3.12.3.6界面设计

如上述流程图所示，点击主界面，选择账户类型，管理员进入管理者界面，点击删除违规产品后，进入到通知买家界面，点击通知举报成功，买家得到通知。

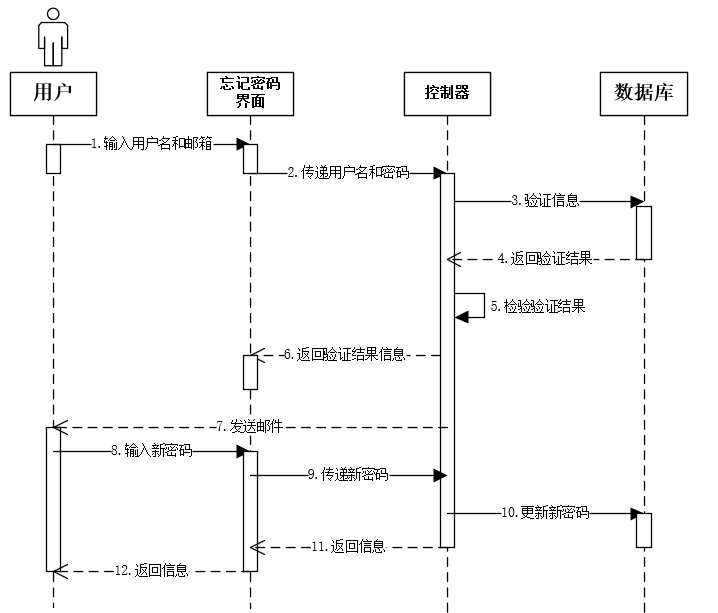
#### 3.12.3.7接口设计

#### 3.12.3.8算法设计

买家举报违规的产品后，管理者确定删除该违规产品后，点击通知买家，通知内容为举报成功消息，接着买家将得到通知。

## 3.13找回密码

### 3.13.1功能



用户可以在忘记密码界面输入用户名和事先绑定的邮箱，系统给指定邮箱发送更改密码的邮件，用户通过邮件给的链接进入密码更改页面，然后返回登陆界面。

### 3.13.2性能

发送邮件的时间不超过20s，更改后的密码同样需要进行加密传输与存储。

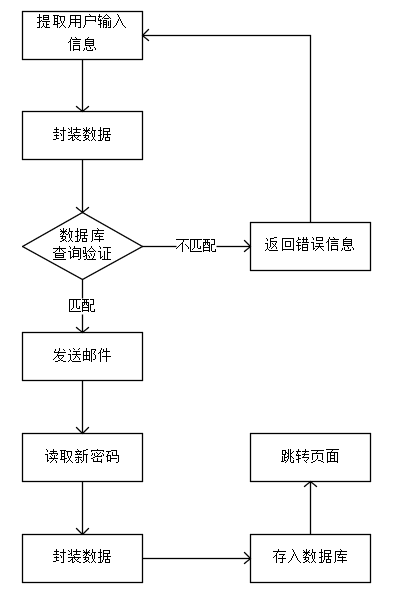
### 3.13.3输入项目

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 类型 | 长度 | 小数点 | 允许空 | 主/外键 |
| 用户名 | Varchar |  |  |  | 主 |
| 邮箱 | Varchar |  |  |  |  |
| 密码 | Varchar |  |  |  |  |

### 3.13.4输出项目

如果用户名和邮箱不能进行匹配则返回错误信息，反之返回发送邮件成功的信息；如果更改密码成功则返回成功信息并跳转页面。

### 3.13.5流程图



### 3.13.6界面设计

从忘记密码界面可以点击返回按钮返回到登陆界面。

### 3.13.7接口设计

### 3.13.8算法设计

通过邮箱给的POP3/SMTP/IMAP服务发送邮件。