系统/子系统设计(结构设计)说明(SSDD)

Qr code

Description automatically generated with low confidence

2021年6月

|  |  |
| --- | --- |
| 团队成员 | |
| 队长 | 宁子淳 |
|  | 柯俊哲 |
|  | 樊伟哲 |
|  | 梁浩 |
|  | 郑继凯 |

指导教师：余仲星

版本：1.0

Contents

[系统/子系统设计(结构设计)说明(SSDD) 1](#_Toc74849030)

[1. 引言 1](#_Toc74849031)

[1.1. 标识 1](#_Toc74849032)

[1.2. 系统概述 1](#_Toc74849033)

[1.3. 文档概述 1](#_Toc74849034)

[1.4. 基线 1](#_Toc74849035)

[2. 引用文件 1](#_Toc74849036)

[3. 系统级设计决策 1](#_Toc74849037)

[4. 系统体系结构设计 2](#_Toc74849038)

[4.1. 系统总体设计 2](#_Toc74849042)

[4.1.1. 概述 2](#_Toc74849048)

[4.1.2. 设计思想 3](#_Toc74849049)

[4.1.3. 基本处理流程 3](#_Toc74849050)

[4.1.4. 系统体系结构 5](#_Toc74849051)

[4.1.5. 功能需求与系统配置项的关系 6](#_Toc74849052)

[4.1.6. 人工处理过程 6](#_Toc74849053)

[4.2. 系统部件 7](#_Toc74849054)

[4.3. 执行概念 7](#_Toc74849055)

[4.4. 接口设计 7](#_Toc74849056)

[4.4.1 接口标识和图表 8](#_Toc74849057)

[5. 运行设计 18](#_Toc74849058)

[5.1. 系统初始化 18](#_Toc74849060)

[5.2. 运行控制 19](#_Toc74849061)

[5.3. 运行结束 19](#_Toc74849062)

[6. 系统出错处理设计 20](#_Toc74849063)

[6.1. 出错信息 20](#_Toc74849065)

[6.2. 补救措施 20](#_Toc74849066)

[7. 系统维护设计 20](#_Toc74849067)

[7.1. 检测点的设计 20](#_Toc74849069)

[7.2. 检测专用模块的设计 20](#_Toc74849070)

[8. 尚待解决的问题 20](#_Toc74849071)

[9. 需求的可追踪性 21](#_Toc74849072)

[10. 注解 21](#_Toc74849073)

[11. 附录 22](#_Toc74849074)

# 引言

## 标识

适用系统：所有可以连接因特网的系统

标题：“摸渔”网上购物平台

版本号：1.0

## 系统概述

本系统应该具有网上购物平台的基本功能，即对商品、用户、订单的新建、管理、删除功能，并且能够保存用户、商品、订单的基本信息。商家可以添加或者删除商品，查看商品详情和订单信息；顾客可以查看商品详情，订单详情，购买商品。本系统的使用者是有买或者卖需求的任意用户，要求基本性能比较健壮，同时便于操作，适合各个阶层的用户使用。

## 文档概述

本文档用于帮助开发人员了解工作进度，方便自己和他人阅读和改进代码，还能够帮助管理人员管理软件开发进程、提高软件的开发效率和质量；便于管理人员、开发人员和操作人员之间的协作、交流和了解，使软件开发活动更加科学有效。同时，便于用户了解软件的功能、性能等各项指标。无保密性要求。

本文档的阅读对象如下：

① 开发人员

② 测试阶段人员

③ 项目组及其他有权需要调用本文档的人员

## 基线

基线：GB/T 8567-2006

# 引用文件

GB/T 8567-2006 《计算机软件文档编制规范》

[美]Shari Lawrence Pfleeger [加]Joanne M.Atlee《Software Engineering 4th Edition 》

# 系统级设计决策

根据需求分析，该软件应该基本实现：

① 注册新用户：必须输入用户名、密码、确认密码，手机号或邮箱至少其中一项，并设置性别、头像，可选择设置个人简介，然后输入验证码。若不符合上述必填要求，则显示提示信息；若数据库中已存在当前注册的用户名，则显示提示信息；若用户名、密码规格不符合要求，则显示提示信息。

② 用户登陆：输入用户名、密码。若用户名不存在于数据库中，则显示提示信息；若用户名与密码不匹配，则显示提示信息。

③ 搜索商品：输入商品关键字，然后跳转搜索结果页面。若关键字匹配，则直接显示商品，并显示商品封面、店铺、销量、价格等信息，同时若商品过多则分页显示；若数据库中没有符合的商品，则给出相应提示。

④ 购买商品：点击商品进入商品详情页面，然后设置购买数量后点击立即购买，跳转到下单页面，确认订单、填写必要信息后即可完成下单。后台数据库同时做出相应变化。若可购买数量为0，则跳转下单页面失败。若在下单过程中数量为0或者商品忽然下架，则提示购买失败。

⑤ 成为商家：填写必要信息，如店铺名称、店铺地址等，即可成为商家。

⑥ 个人中心：可设置或修改昵称、性别、头像、个人简介等信息，并可更新手机号、邮箱、登陆密码、支付密码、收货地址等信息

⑦ 店铺页面：显示该店铺名称、logo等信息，并显示该店铺上架的商品等信息，可对商品进行分类、筛选、搜索等操作。对于用户，可浏览、搜索；对于持有该店铺的商家，可进行修改等操作。

⑧ 首页：包含注册、登录链接，适当显示一定量的爆款商品，可进行商品关键字搜索操作，可对商品类别进行初步搜索。

# 系统体系结构设计



## 系统总体设计



### 概述

#### 4.1.1.1 功能描述

功能：对店铺的商品信息和用户、商家的信息进行有效的管理；对商品的进存销等环节进行信息化管理；实现买家网上浏览商品，网上购物；实现数据库的录入、检索、用户信息查询、商品信息查询；店中所有的商品能够供用户随时浏览；用户的个人信息可以由用户自己修改，添加；商品信息只能由发布该商品的商家添加，修改；所有注册用户信息只能由后台管理人员查询。

性能：响应时间主要看用户的网速，所有链接采用Https，兼容现代化的浏览器（不考虑特别针对特别老旧的IE进行特殊优化，即不兼容IE 11及以下版本），可在PC端、Mac端、移动端等多端浏览，资源占用一般。

#### 4.1.1.2 运行环境

所有可以连接因特网的操作系统，且操作系统上必须有现代化的浏览器，如Chrome、FireFox、Safari等。

### 设计思想

#### 4.1.2.1 系统构思

一个购物网站，所有人都能使用，它包括了用户注册与登陆、商品搜索与购买等功能。

该系统运营的网上店铺与实体店铺相比减少了管理员的工作量，实现商品管理的数字化，大大提高了工作效率；用户足不出户即可进入商店浏览与购买所需商品，节省客户时间，并能迅速获得所需商品的全面信息，提高客户购买商品的效率；网店可以24小时营业，不受时间和地点的限制，既方便了客户，也增加了运营商的收益。

#### 4.1.2.2 关键技术与算法

① 关键技术：HTML、CSS、JavaScript、jQuery、Nginx、uWSGI、Django、SQLite。

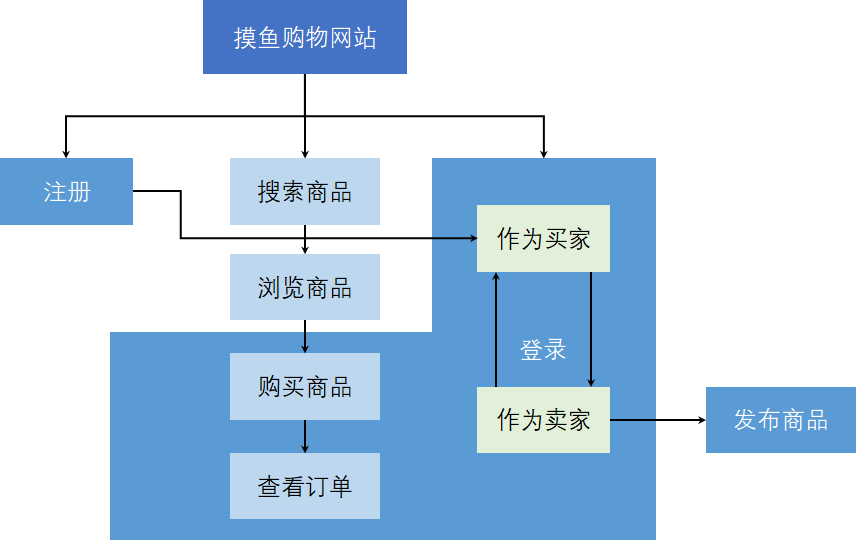
② 主要算法：存储、查找、对比算法。

#### 4.1.2.3 关键数据结构

最主要的是数据库的信息检索，采用B+树的数据结构。

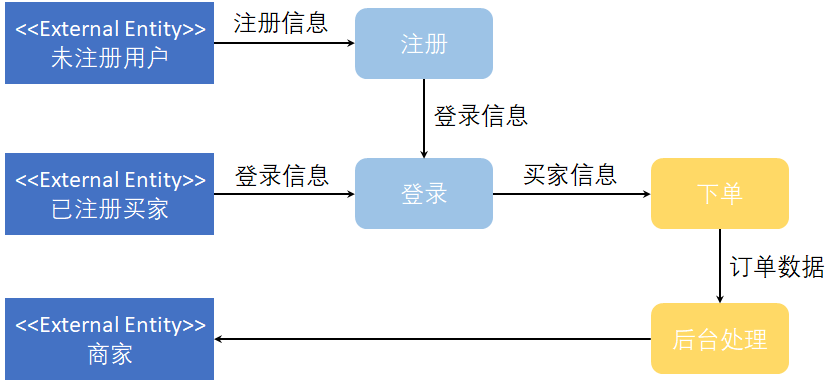
### 基本处理流程

#### 4.1.3.1 系统流程图

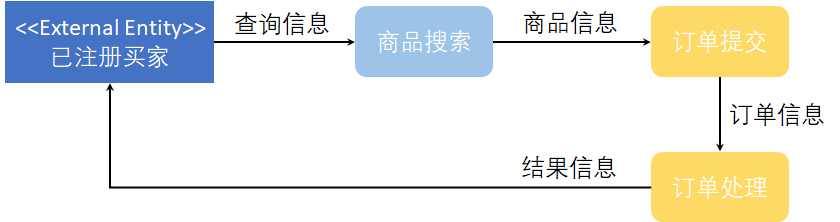


#### 4.1.3.2 数据流程图

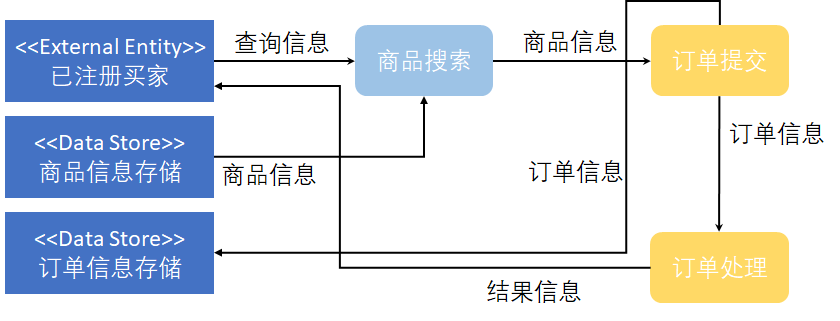
① 顶层数据流图：



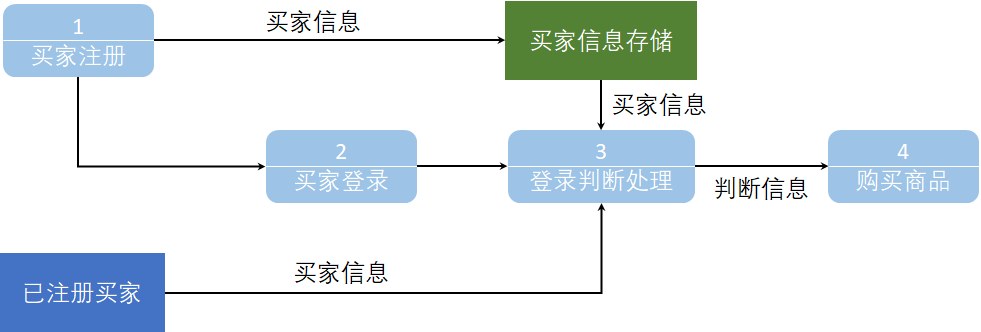
②第二层数据流图（前台购买）



③第二层数据流图（后台管理）



④第三层数据流图（登录信息）：



### 系统体系结构

#### 4.1.4.1 系统配置项

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 摸渔购物系统 | 子系统 | 子子系统 | 模块 | 标识符 | 功能 |
| 买家系统 | 信息系统 | 注册 | register | 用户注册 |
| 登录 | login | 用户登录 |
| 个人信息 | info | 浏览、修改、更新个人信息 |
| 成为商家 | seller\_reg | 个人用户注册成为卖家 |
| 商品系统 | 搜索 | search | 搜索商品 |
| 浏览 | browser | 浏览商品详情 |
| 购买 | purchase | 下单购买商品 |
| 用户订单系统 | order | 查看订单详情 |
| 卖家系统 | 信息系统 | 开设店铺 | store | 开设自己的店铺 |
| 店铺信息 | store\_info | 包括所在地、类别等 |
| 商品系统 | 发布商品 | release | 可设置单价、数量、详情等 |
| 编辑商品 | edit | 更新商品信息 |
| 下架商品 | off | 移除商品，使之只读 |
| 商家订单管理 | order\_ | 管理订单 |

#### 4.1.4.2 系统层次结构

各个配置项之间是并列的关系，但相似地方有所交集

#### 4.1.4.3 系统配置项设计

① 注册新用户：必须输入用户名、密码、确认密码，手机号或邮箱至少其中一项，并设置性别、头像，可选择设置个人简介，然后输入验证码。若不符合上述必填要求，则显示提示信息；若数据库中已存在当前注册的用户名，则显示提示信息；若用户名、密码规格不符合要求，则显示提示信息。

② 用户登陆：输入用户名、密码。若用户名不存在于数据库中，则显示提示信息；若用户名与密码不匹配，则显示提示信息。

③ 搜索商品：输入商品关键字，然后跳转搜索结果页面。若关键字匹配，则直接显示商品，并显示商品封面、店铺、销量、价格等信息，同时若商品过多则分页显示；若数据库中没有符合的商品，则给出相应提示。

④ 购买商品：点击商品进入商品详情页面，然后设置购买数量后点击立即购买，跳转到下单页面，确认订单、填写必要信息后即可完成下单。后台数据库同时做出相应变化。若可购买数量为0，则跳转下单页面失败。若在下单过程中数量为0或者商品忽然下架，则提示购买失败。

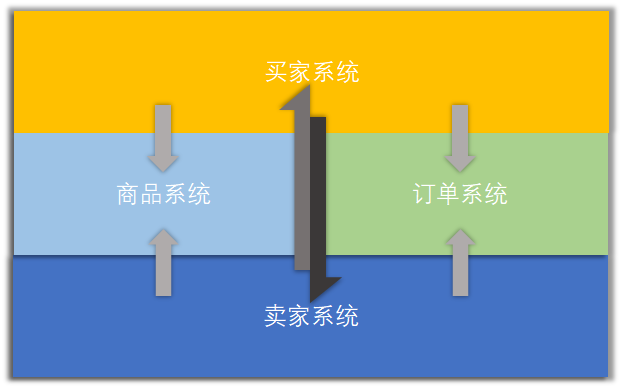
⑤ 成为商家：填写必要信息，如店铺名称、店铺地址等，即可成为商家。

⑥ 个人中心：可设置或修改昵称、性别、头像、个人简介等信息，并可更新手机号、邮箱、登陆密码、支付密码、收货地址等信息

⑦ 店铺页面：显示该店铺名称、logo等信息，并显示该店铺上架的商品等信息，可对商品进行分类、筛选、搜索等操作。对于用户，可浏览、搜索；对于持有该店铺的商家，可进行修改等操作。

⑧ 首页：包含注册、登录链接，适当显示一定量的爆款商品，可进行商品关键字搜索操作，可对商品类别进行初步搜索。

### 功能需求与系统配置项的关系

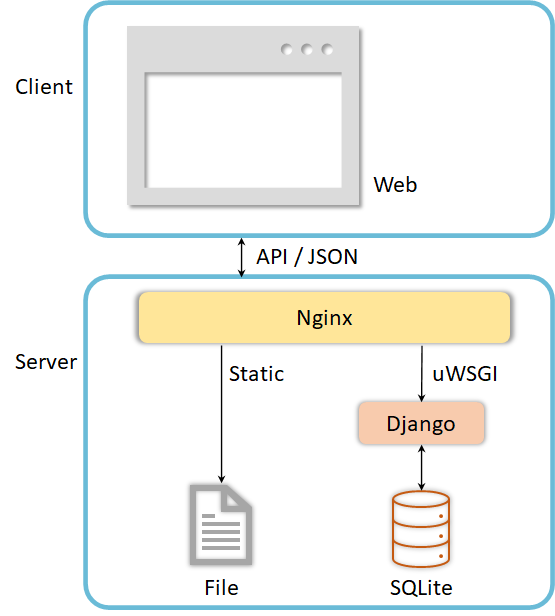


### 人工处理过程

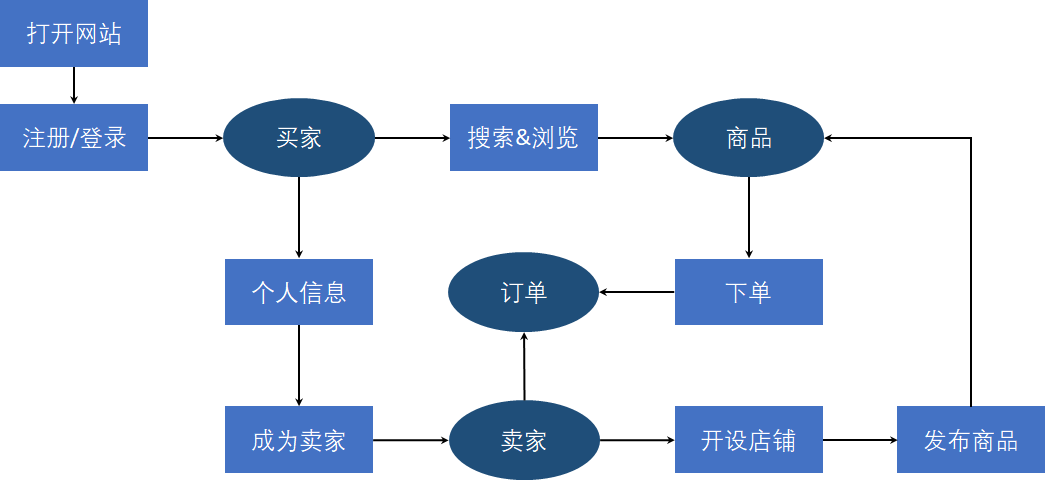
① 用户下单

② 商家接受订单

## 系统部件



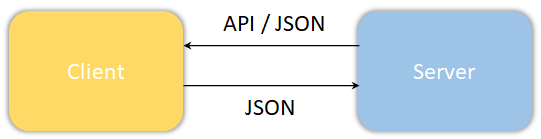
## 执行概念



## 接口设计

本条描述系统部件的接口特性，包括部件之间的接口及它们与外部实体之间的接口。

### 4.4.1 接口标识和图表



后端服务器和前端浏览器之间的API数据交换采用JSON数据格式。下面为各功能的接口信息。

#### 4.4.1.1 用户认证

**验证码**

接口地址：/api/user/captcha

请求方式：GET

说明：用户提交注册登录以及重置密码前，需要获取验证码。验证码只能使用一次，刷新验证码即重新获取

返回内容：image/png，cookie

**登录**

接口地址：/api/user/login

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| (user)或(uid, uname, phone, email)其中一项 | string | 是 | 传入user则自动判断类型 |
| password | string | 是 | 用户密码 |
| captcha | string | 是 | 验证码 |
| no\_captcha | bool | 否 | 是否关闭验证码（debug） |

返回参数：

|  |  |  |
| --- | --- | --- |
| HTTP状态码 | 返回json中，msg的值 | 说明 |
| 200 | 登录成功 | 在cookie中返回session，存储登录信息 |
| 403 | 请使用 POST 请求|无效参数|无此用户|密码错误|请输入验证码|验证码过期|验证码错误 |  |
| 500 | 服务器错误 | 下同 |

登录成功时，返回json信息

|  |  |
| --- | --- |
| 名称 | 说明 |
| uid | 用户ID |
| uname | 用户名 |
| phone | 手机号 |
| email | 邮箱 |
| gender | 0未设置 1男 2女 |
| intro | 个人介绍 |
| avatar | 用户头像文件相对于根目录的地址，或完整url |

1. {
2. "code": 200,
3. "msg": "获取成功",
4. "data": {
5. "uid": 1,
6. "uname": "ning",
7. "phone": null,
8. "email": "test@mrning.com",
9. "gender": 0,
10. "intro": "",
11. "avatar": "/media/users/default.jpg"
12. }
13. }

**登出**

接口地址：/api/user/logout

请求方式：GET / POST

请求参数：无，从session中获取用户信息

返回参数：

|  |  |  |
| --- | --- | --- |
| HTTP状态码 | 返回json中，msg的值 | 说明 |
| 200 | 已清除登录信息 | 服务器已清除session中存储的登录信息 |

1. {
2. "code": 200,
3. "msg": "已清除登录信息"
4. }

**注册**

接口地址：/api/user/register

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| uname | string | 是 | 用户名 |
| phone或email | string | 是 | 手机和邮箱中的至少一项 |
| password | string | 是 | 用户密码 |
| intro | string | 否 | 个人介绍 |
| address | string | 否 | 收货地址 |
| pay\_pwd | string | 否 | 支付密码 |
| avatar\_name | string | 否 | 要修改的默认头像文件名 |
| no\_captcha | bool | 否 | 是否关闭验证码（debug） |
| captcha | string | 是 | 密码先在前端加密一次，头像这里不用上传。 |

返回参数：

|  |  |  |
| --- | --- | --- |
| HTTP状态码 | 返回json中，msg的值 | 说明 |
| 200 | 注册成功 |  |
| 403 | 无效参数：缺少用户名或密码|无效参数：缺少手机或邮箱|用户ID已存在|用户名已存在|注册错误|请输入验证码|验证码错误|用户名不能全为数字|手机号格式不正确|用户名包含特殊字符 | uname不能为纯数字 |

1. {
2. "code": 200,
3. "msg": "注册成功"
4. }

**修改密码**

接口地址：/api/user/password

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| old\_password | string | 是 | 旧密码 |
| new\_password | string | 是 | 新密码 |

返回参数：

|  |  |  |
| --- | --- | --- |
| HTTP状态码 | 返回json中，msg的值 | 说明 |
| 200 | 成功地修改了密码 |  |
| 403 | 请使用 POST 请求|无效参数|未登录或登录超时|无此用户|旧密码错误 |  |

**上传头像**

接口地址：/api/user/avatar

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| avatar | file | 是 | 用户要求的新头像，建议前端先压缩一下 |

返回参数：

|  |  |  |
| --- | --- | --- |
| HTTP状态码 | 返回json中，msg的值 | 说明 |
| 200 | 成功地修改了头像 |  |
| 403 | 未登录或登录超时|未获取到avatar |  |

**用户信息**

接口地址：/api/user/info

请求方式：GET，查询用户信息

请求方式：POST，修改用户信息，提交啥修改啥

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| gender | string | 否 | 要修改的性别 |
| phone | string | 否 | 要修改的手机号 |
| email | string | 否 | 要修改的邮箱 |
| intro | string | 否 | 要修改的个人介绍 |
| address | string | 否 | 要修改的收货地址 |
| avatar\_name | string | 否 | 要修改的默认头像文件名 |
| pay\_pwd | string | 否 | 要修改的支付密码 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| uid | 用户ID |
| uname | 用户名 |
| phone | 手机号 |
| email | 邮箱 |
| gender | 0未设置 1男 2女 |
| intro | 个人介绍 |
| avatar | 用户头像文件相对于根目录的地址，或完整url |

1. {
2. "code": 200,
3. "msg": "获取成功",
4. "data": {
5. "uid": 1,
6. "uname": "ning",
7. "phone": null,
8. "email": "test@mrning.com",
9. "gender": 0,
10. "intro": "",
11. "address": null,
12. "avatar": "/media/users/default.jpg"
13. }
14. }

**邮件找回密码**

接口地址：/api/user/reset

请求方式：POST

第一次请求：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| email或phone | string | 是 | 恢复凭证 |
| captcha | string | 是 | 验证码 |
| no\_captcha | bool | 否 | 是否关闭验证码（debug） |

返回参数：

|  |  |  |
| --- | --- | --- |
| HTTP状态码 | 返回json中，msg的值 | 说明 |
| 200 | 已发送恢复码至邮箱|已发送恢复码至手机 |  |
| 403 | 验证码错误|找不到用户|发送邮件失败|发送短信失败 | 输入错误会使CAPTCHA失效；发送邮件失败时，返回错误类型。 |

1. {"code":403,
2. "data":"发送邮件失败：SMTPRecipientsRefused({'example@example.example': (550, b'Mailbox not found or access denied')})"
3. }

第二次及以后的请求：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| code | string | 是 | 向密保方式中发送的恢复码 |
| captcha | string | 是 | 验证码 |
| new\_password | string | 是 | 新密码 |

返回参数：

|  |  |  |
| --- | --- | --- |
| HTTP状态码 | 返回json中，msg的值 | 说明 |
| 200 | 修改成功 |  |
| 403 | 验证码错误|未获取恢复码或已失效|恢复码错误|参数错误 | 输入错误会使CAPTCHA失效 |

#### 4.4.1.2 店铺管理

**我的店铺**

接口地址：/api/user/shop

请求方式：GET

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| sid | 店铺ID |
| uid | 用户ID |
| sname | 店铺名 |
| saddr | 店铺物理地址 |
| avatar | 店铺头像 |
| create\_time | 创建时间 |
| update\_time | 更新时间 |

**开设店铺**

接口地址：/api/user/shop/new

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| sname | string | 是 | 店铺名 |
| saddr | string | 是 | 店铺地址 |
| avatar | file | 否 | 店铺头像 |
| avatar\_name | string | 否 | 店铺使用的默认头像名 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| sid | 店铺ID |
| uid | 用户ID |
| sname | 店铺名 |
| saddr | 店铺地址 |
| avatar | 店铺头像 |
| create\_time | 创建时间 |
| update\_time | 更新时间 |

**修改店铺信息**

接口地址：/api/user/shop/edit

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| sname | string | 否 | 店铺名 |
| saddr | string | 否 | 店铺地址 |
| avatar | file | 否 | 店铺头像 |
| avatar\_name | string | 否 | 店铺使用的默认头像名 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| sid | 店铺ID |
| uid | 用户ID |
| sname | 店铺名 |
| saddr | 店铺地址 |
| avatar | 店铺头像 |
| create\_time | 创建时间 |
| update\_time | 更新时间 |

#### 4.4.1.3 订单管理

**我的订单**

接口地址：/api/user/orders

请求方式：GET

返回参数：

|  |  |  |
| --- | --- | --- |
| 名称 | 说明 | |
| uid | 用户ID | |
| order\_num | 订单总数 | |
| orders | 订单数组，字段如下表所示 | |
| 名称 | | **说明** | |
| oid | | 店铺ID | |
| pid | | 用户ID | |
| pname | | 商品名 | |
| price | | 商品价格 | |
| product\_num | | 商品数量 | |
| sid | | 店铺ID | |
| sname | | 店铺名 | |
| status | | 状态(0,'等待确认'),(1,"等待发货"),(2,"等待买房确认"),(3,"已完成") | |
| deliverer | | 承运人 | |
| delivery\_number | | 运单号 | |
| create\_time | | 创建时间 | |
| update\_time | | 更新时间 | |

**订单详情**

接口地址：/api/user/order/<oid>

请求方式：GET

请求参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| oid | 订单ID |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| uid | 用户ID |
| oid | 店铺ID |
| pid | 用户ID |
| pname | 商品名 |
| price | 商品价格 |
| product\_num | 商品数量 |
| sid | 店铺ID |
| sname | 店铺名 |
| status | 状态(0,'等待确认'),(1,"等待发货"),(2,"等待买房确认"),(3,"已完成") |
| deliverer | 承运人 |
| delivery\_number | 运单号 |
| address | 收获地址 |
| create\_time | 创建时间 |
| update\_time | 更新时间 |

#### 4.4.1.4 商品管理

**新建商品**

接口地址：/api/product/new

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| pname | string | 是 | 商品标题 |
| price | float | 是 | 价格 |
| info | string | 是 | 描述 |
| tag | string | 是 | 分类标签 |
| cover | file | 否 | 商品图 |
| status | int | 是 | 剩余库存 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| pid | 商品ID |
| pname | 商品名 |
| sid | 店铺ID |
| sname | 店铺名 |
| price | 商品价格 |
| info | 商品数量 |
| volume | 销量 |
| tag | 分类标签 |
| cover | 商品封面图 |
| create\_time |  |
| update\_time |  |

**查看商品**

接口地址：/api/product/<int:pid>

请求方式：GET

请求参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| pid | 商品ID |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| pid | 商品ID |
| pname | 商品名 |
| sid | 店铺ID |
| sname | 店铺名 |
| price | 商品价格 |
| info | 商品数量 |
| volume | 销量 |
| tag | 分类标签 |
| cover | 商品封面图 |
| create\_time |  |
| update\_time |  |

**修改信息**

接口地址：/api/product/<int:pid>/edit

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| pname | string | 否 | 商品标题 |
| price | float | 否 | 价格 |
| info | string | 否 | 描述 |
| tag | string | 否 | 分类标签 |
| cover | file | 否 | 商品图 |
| status | int | 否 | 剩余库存 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| pid | 商品ID |
| pname | 商品名 |
| sid | 店铺ID |
| sname | 店铺名 |
| price | 商品价格 |
| info | 商品数量 |
| volume | 销量 |
| tag | 分类标签 |
| cover | 商品封面图 |
| create\_time |  |
| update\_time |  |

#### 4.4.1.5 订单相关

**购买产品**

接口地址：/api/product/<int:pid>/order

请求方式：POST

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| pid | int | 是 | 商品ID (In URL) |
| address | string | 是 | 收货地址 |
| quantity | string | 是 | 商品数量 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| oid | 订单号 |
| status | 订单状态 |

#### 4.4.1.6 搜索相关

**按标题**

接口地址：/api/search/name/<str:keyword>

请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| keyword | string | 是 | 关键词 |
| address | string | 否 | 地域筛选 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| keyword | 关键词 |
| item\_num | 搜索结果数量 |
| items | 搜索结果数组 |

其中，items中每个项目：

|  |  |
| --- | --- |
| 名称 | 说明 |
| pid | 商品ID |
| pname | 商品名 |
| sid | 店铺ID |
| sname | 店铺名 |
| price | 商品价格 |
| volume | 销量 |
| tag | 分类标签 |
| cover | 商品封面图 |

**搜店铺**

接口地址：/api/search/shop/<str:keyword>

请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| keyword | string | 是 | 关键词 |
| address | string | 否 | 地域筛选 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| keyword | 关键词 |
| item\_num | 搜索结果数量 |
| items | 搜索结果数组 |

其中，items中每个项目：

|  |  |
| --- | --- |
| 名称 | 说明 |
| sid | 店铺ID |
| sname | 店铺名 |
| avatar | 店铺头像 |
| uid | 老板ID |
| uname | 老板用户名 |

**根据标签查看商品**

接口地址：/api/product/tag/<str:tag>

请求方式：GET

请求参数：

|  |  |  |  |
| --- | --- | --- | --- |
| 名称 | 类型 | 必填 | 说明 |
| tag | string | 是 | 标签 |

返回参数：

|  |  |
| --- | --- |
| 名称 | 说明 |
| tag | 标签 |
| item\_num | 搜索结果数量 |
| items | 搜索结果数组 |

其中，items中每个项目：

|  |  |
| --- | --- |
| 名称 | 说明 |
| sid | 店铺ID |
| sname | 店铺名 |
| avatar | 店铺头像 |
| uid | 老板ID |
| uname | 老板用户名 |

# 运行设计



## 系统初始化

说明本系统的初始化过程。

系统初始化过程分为建立数据表、建立超级管理员账户两个步骤。

**建立数据表**

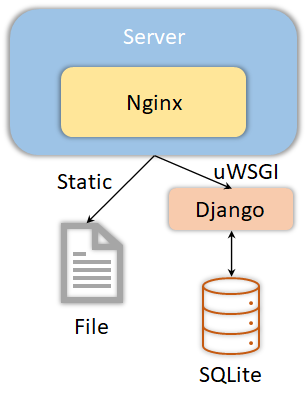
1. python3 manage.py makemigrations
2. python3 manage.py makemigrations user
3. python3 manage.py makemigrations forum
4. python3 manage.py makemigrations api
5. python3 manage.py makemigrations movie
6. python3 manage.py makemigrations photo
7. python3 manage.py migrate

**建立超级管理员账户**

1. echo "超级管理员设置"
2. read -p "用户名：" superuser
3. read -p "密码：" superpass
4. read -p "邮箱：" supermail
5. export DJANGO\_SUPERUSER\_PASSWORD=$superpass
6. python3 manage.py createsuperuser --noinput --username $superuser --email $supermail
7. unset DJANGO\_SUPERUSER\_PASSWORD
8. echo "完成"

## 运行控制

服务依赖于Nginx Web服务器，作为静态资源分发和反向代理，使用uWSGI作为Nginx和Django通信接口，使用SQLite作为后端数据库。因此，要使得服务运行，Nginx、uWSGI和Django同时运行。



**启动服务**

启动Nginx服务

service nginx start

启动uWSGI服务，运行Django

uwsgi --ini ./uwsgi.ini &

如果把TouchFishing写进了Linux服务中，可使用 service tf start 来启动服务

**重载服务**

重载Nginx配置

service nginx reload

**更新数据库**

python manage.py makemigrations

python manage.py migrate

如果把TouchFishing写进了Linux服务中，可使用 service tf restart 来重启服务

## 运行结束

说明本系统运行的结束过程。

**停止服务**

如果把 TouchFishing 写进了Linux服务中，可使用 service tf stop 来重启服务

# 系统出错处理设计



## 出错信息

**请求URL不存在**

如果请求的静态资源不存在，Nginx服务器返回404，如果请求的动态链接不存在，Django返回404。

**未经授权的访问**

如果用户请求了权限之外的内容，服务器返回HTTP状态码403。

**服务器崩溃**

如果因为某些原因，Django服务器无法响应用户的请求，Nginx服务器返回HTTP状态码500。

## 补救措施

**迁移**

如果服务器出现负载高、网络延时高等问题，可以采用纵向扩展服务器的方法，提升服务器的配置，使用内容分发网络来提升用户网络体验。也可以采用负载均衡技术，将请求分散到不同的服务器中。

**重新运行**

如果出现了系统故障，可以尝试重新启动系统来解决问题。

# 系统维护设计

说明为了系统维护的方便，在系统内部设计中作出的安排。



## 检测点的设计

在系统空闲时间，自动备份数据库内容，并以OOS、FTP等方式上传到异地，保证数据的安全性。

## 检测专用模块的设计

使用定时任务来执行bash脚本，从而实现服务器状态的检测和自动化的备份操作。

# 尚待解决的问题

未实现购物车功能

未实现订单评价功能

未实现真正的在线购买

# 需求的可追踪性

可以通过某商品的点击量和购买量来分析书籍的需求情况：

① 点击量高、购买量低：分析商品价格是否过高

② 点击量高、购买量高：分析商品是否可以适度涨价并多进货

③ 点击量低、购买量低：分析商品是否需要下架或进行促销活动

④ 点击量低、购买量高：分析商品是否被非法囤积

# 注解

缩略语：

|  |  |  |
| --- | --- | --- |
| CASE | 计算机辅助软件工程 | Computer Assistant Software Engineering |
| COM | 计算机操作手册 | Computer Operation Manual |
| CPM | 计算机编程手册 | Computer Programming Manual |
| CSCI | 计算机软件配置项 | Computer Software Configuration Item |
| DBDD | 数据库(顶层)设计说明 | Database Design Description |
| DID | 资料条目说明 | Data Item Description |
| DPMR | 开发进度月报 | Development Plan Month Report |
| DRD | 数据需求说明 | Data requirement Description |
| FAR | 可行性分析报告 | Feasibility analysis Report |
| HWCI | 硬件配置项 | Hardware Configuration Item |
| IDD | 接口设计说明 | Interface Design Description |
| IRS | 接口(软件)需求规格说明 | Interface Requirement Specification |
| IV＆V | 独立验证和确认 | Independent verification and validation |
| OCD | 运行概念说明 | Operation Conception Description |
| PDSR | 项目开发总结报告 | Project Development summary Report |
| SCCB | 软件配置控制委员会 | Software Configuration Control Board |
| SCM | 软件配置管理 | Software Configuration Manager |
| SCMP | 软件配置管理计划 | Software Configuration Manager Plan |
| SDD | 软件(结构)设计说明 | Software Design Description |
| SDF | 软件开发文件 | Software Development File |
| SFDD | 软件开发文档 | Software Development Document |
| SDL | 软件开发库 | Software Development Library |
| SDP | 软件开发计划 | Software Development Plan |
| SIP | 软件安装计划 | Software Installation Plan |
| SPS | 软件产品规格说明 | Software Product Specification |
| SQA | 软件质量保证 | Software Quality Assure |
| SQAP | 软件质量保证计划 | Software Quality Assure Plan |
| SRS | 软件需求规格说明 | Software Requirement Specification |
| SSDD | 系统/子系统设计(结构设计)说明 | System Subsystem Design Description |
| SSS | 系统/子系统需求规格说明 | System Subsystem Requirement Specification |
| STD | 软件测试说明 | Software Testing Descrition |
| STP | 软件测试计划 | Software Testing Plan |
| STR | 软件测试报告 | Software Testing Report |
| STrP | 软件移交计划 | Software Transfer Plan |
| SUM | 软件用户手册 | Software User Manual |
| SVD | 软件版本说明 | Software Version Description |
| SW | 软件 | Software |

# 附录

无