

## 一、实验目的

综合运用 Flask 网站开发方法，实现网站各项常用的功能。

## 二、实验环境

Windows 7,python 3.x ,PC

## 三、实验内容与步骤

1. 请运用本学期学习的 **Flask** 应用开发技术，开发一个小型网站。请撰写完整的项目开发报告，包括但不限于：项目背景，需求分析，开发环境（包括开发环境说明，配置说明，用到的技术和第三方库，目录结构等），网站架构，数据库设计，各模块详细设计（要包含重要函数，关键代码等），效果展示，网站使用说明，项目总结等。
2. 软件开发文档撰写可参考文章《软件开发文档介绍》：  
<https://cloud.tencent.com/developer/article/1353285>  
报告大纲可参考下文红字标示部分，文档规范参考文后模板。
3. 请提交本项目报告文档，汇报 **PPT**，并将完整的项目源代码压缩成 **rar** 格式文件提交。

# 琳琅-购物网站项目报告

## 目录

第1章	引言 .....	4
1.1	编写目的 .....	4
1.2	项目背景 .....	4
第2章	需求分析 .....	6
2.1	定义 .....	6
2.2	任务概述 .....	6
2.2.1	目标 .....	6
2.2.2	用户的特点 .....	7
2.2.3	假定和约束 .....	7
2.3	需求规定 .....	8
2.3.1	系统用例图 .....	8
2.3.2	分析类图 .....	8
2.3.3	组件图 .....	9
2.3.4	部署图 .....	10
2.4	对性能的规定 .....	11
2.4.1	精度 .....	11
2.4.2	时间特性要求 .....	11
2.4.3	灵活性 .....	11
2.4.4	其他专门要求 .....	12
2.5	运行环境规定 .....	12
2.5.1	设备 .....	12
2.5.2	支持软件 .....	12
2.5.3	接口 .....	12
第3章	概要设计 .....	13
3.1	系统构架 .....	13
3.1.1	系统总体布局 .....	13
3.1.2	系统总体结构设计 .....	13
3.2	模块划分 .....	13
3.2.1	用户管理模块 .....	13
3.2.4	评论管理系统 .....	14
3.2.5	功能 ER 图 .....	14
3.3	网站结构——三层架构 .....	14
3.4	数据设计 .....	15
第4章	详细设计（模块流程图，状态图，局部变量） .....	16
4.1	画面设计 .....	16

4.1.1 输入/输出设计（登录页面） .....	16
4.1.2 人机界面设计（管理页面） .....	16
4.2 代码设计 .....	16
4.3 模块流程图 .....	17
<b>第5章 数据库设计</b> .....	<b>18</b>
5.1. 数据实体-关系图 .....	18
5.2. 数据实体描述 .....	18
5.2.1. 数据库表 .....	18
5.2.2. 数据实体 .....	18
5.3. 实体关系描述 .....	19
5.4. 关系模型描述 .....	20
5.5. 数据库一致性设计 .....	21
<b>第6章 系统测试</b> .....	<b>24</b>
6.1 单元测试 .....	错误!未定义书签。
<b>第7章 效果和使用展示</b> .....	<b>29</b>
7.1 系统展示 .....	29
7.1.1 用户管理 .....	29
7.1.2 商品管理 .....	32
7.1.3 订单管理 .....	36
7.2 使用说明 .....	37
7.2.3 管理员使用说明 .....	37
7.2.3 用户使用说明 .....	38
<b>第8章 总结与展望</b> .....	<b>39</b>
1.1 项目总结 .....	39
1.2 项目展望 .....	39

# 第1章 引言

## 1.1 编写目的

本开发报告是基于针对购物交易网站日常业务深入调研基础上编制而成的，旨在详细记录并分析购物交易网站在功能、性能、安全性、接口等方面的需求。通过阅读本说明书，系统分析员和架构设计师能够深入理解用户的需求，并在此基础上进一步编制概要说明书，以完成后续的设计工作。测试员可以参考本说明书完成验收测试用例的设计工作。本说明书的预期读者是客户、系统分析员、架构设计师、测试员、项目经理。

## 1.2 项目背景

### (1) 网上购物系统的背景：

电子商务，就是在网上开展商务活动。它是以电子交易为手段，完成金融、物品、服务、信息等价值的交换，是快速而有效地进行各种商务活动的最新方法。电子商务的出现，改变了传统的一手交钱一手交货的模式，使得商业模式发生了很大的变化。

电子商务代表着未来贸易方式的发展方向，其应用和推广将给社会和经济带来极大的效益。具有强大生命力的现代电子交易手段已越来越被人们所认识，发达国家在这方面有很多年的发展历史，发展中国家也正在加紧制定总体发展战略，大力促进电子商务在国民经济各个领域的应用，力争在越来越激烈的国际竞争中占据有利的地位。因此，对于网上购物技术的研究和实现，以及搭建一个实用的网上商店是很重要的。

网上购物系统的意义：通过对在线购物系统的实现来详细理解在线购物的流程，安全机制，基于软件体系结构的软件开发方法，基于Flask Web开发技术。在了解了在线购物的流程的基础上，结合实际的在线购物模型，构思一个在线购物系统的模型，对系统的功能进行初步的定义，并且应用在实际的模型之上，验证模型的可用性。

本题目的实践意义在于对购物流程，安全机制的基础上，利用Python, sql, html等开发语言对在线购物系统进行模拟实现，最后通过在线的购物的一系列操作和业务规则来验证系统的可用性。

网上购物系统优点：

1、基本不需要占压资金。传统商店的进货资金少则几千元，多则数万元，而网上商店则不需要压资金。

2、24小时营业时间。网上商店延长了商店的营业时间，一天24小时、一年365天不停地运作，无须专人值班看店，都可照常营业。传统店铺的营业时间一般为8-12小时，遇上坏天气或者老板、店员有急事也不得不暂时休息。

3、销售规模不受地盘限制。传统商店有多大就只能摆放多少商品，生意大小常常被小店面积限制。而在网上，即便在地面上只有一个小商店，或者干脆就没有门面，开店的生意却可以照样做得很大。

4、不受店面空间的限制。哪怕只是街边小店，在网上却可以拥有百货大楼那么大的店面，只要投资者愿意，可以摆上成千上万种商品。目前国内最大的专业拍卖网站同时在线的商品要超过10万件——已超过一些大超市。

5、不受地理位置影响。不管客户离店有多远，也不管顾客是国内还是国外，在网上，客户一样可以很方便地找到并购买商品。这令消费群体突破了地域的限制，变得无限广阔了。该系统在功能上基本实现一个网上商店的应用

内容提纲

#### 一、电子商务系统可行性分析

1、经济可行性分析：系统是为方便人们网上购物的系统，客户端仅装有浏览器的电脑，即可使用系统，系统成本主要集中在系统软件的开发上，当系统投入运行后可以为人们节约大量的人力，时间。所带来的效益远远大于系统软件的开发成本。在经济上完全可行

2、技术可行性分析：本系统拟使用Python中的Flask框架作为开发后端的工具，使用HTML+CSS+JavaScript+Jinja2作为开发前端的工具。在IDEA集成开发环境中进行开发。使用MySQL作为数据库管理工具。

3、操作可行性分析：界面设计时充分考虑管理人员的习惯，使得操作简单；数据录入迅速、规范、可靠；统计准确；适应力强；容易扩充。用户在使用本系统的时候和一般的浏览网页一样，操作简单。

4、法律可行性分析：该平台的开发不会在社会上引起侵权或其它责任问题。因为该平台是作为课程设计与商业无关，又因为是自主开发设计，因此不会构成侵权，在法律上是可行的。

(2) 软件名称：琳琅-购物交易网站

(3) 投资单位：无

(4) 用户：广大互联网用户

(5) 开发单位：软件工程2班王俊伟、孙金玮开发小组

## 第2章 需求分析

### 2.1 定义

- (1) LinLang - Shopping Site: 琳琅-购物交易网站。
- (2) 登录功能: 如果是网上商城的用户, 购买或者访问个人资料界面需填写用户名、密码即可享受商城服务。
- (3) 注册功能: 顾客如果要使用更多功能首先要注册为网上商城的用户。注册时需填写登录邮箱、用户名、密码以及一系列个人信息即可。注册后可修改密码、邮箱、个人信息。
- (4) 浏览商品: 用户在未登录的情况下也可浏览商品, 但无法进行购买。
- (5) 购买功能: 顾客在对应的商品界面进行购买并生成对应的订单。
- (6) 支付功能: 用户在提交订单后对订单进行支付
- (7) 管理订单功能: 管理员可以查询、查看订单。
- (8) 管理商品功能: 管理员可以添加、查询、修改、删除商品。
- (9) 管理用户: 管理员可以对用户资料进行修改。

### 2.2 任务概述

#### 2.2.1 目标

##### (1) 商品管理

网站具备良好的商品管理功能, 具备对商品进行添加、修改、删除、查询的基本功能。

##### (2) 订单管理

网站具备良好的订单管理功能, 对于用户而言仅能查看到自己的订单并对订单进行支付和收货操作, 而对于管理员而言, 则可以查看所有的用户订单。同时管理员在订单界面可以进行发货操作。

##### (3) 用户管理

网站具有完备的用户管理功能, 具有用户登录, 用户注册, 用户修改个人资料, 用户修改密码等功能。同时网站对不同等级的用户进行了权限限制, 管

理员用户有权限查看所有用户信息以及修改用户资料的信息，普通用户只能查看到自己的全部信息以及修改自己的个人资料，而对于其余用户的敏感信息例如收货地址则无法查看。

#### (4) 商品检索

通过建立数据库，接受用户查询,利用信息检索算法，可以进行关键字模糊搜索。

#### (5) 其他功能

安全性: 用户登录是需要申请用户名跟密码，需要身份认证，交易时需要用到网银，对于网上交易的安全问题还是可以保障的。

可维护性: 当系统中发现并纠正一个故障或进行一次更改的时，随时都可以修改和维护。

### 2.2.2 用户的特点

琳琅-购物交易网站的用户包括网站管理员和客户两类。网站管理员负责管理客户信息、订单、商品以及物流工作，熟悉日常工作的业务规则。客户是广大互联网用户，普遍掌握了计算机的基础知识，能够利用浏览器作为工具完成网站购物操作。客户日常使用的浏览器主要包括 Firefox、Chrome 等。

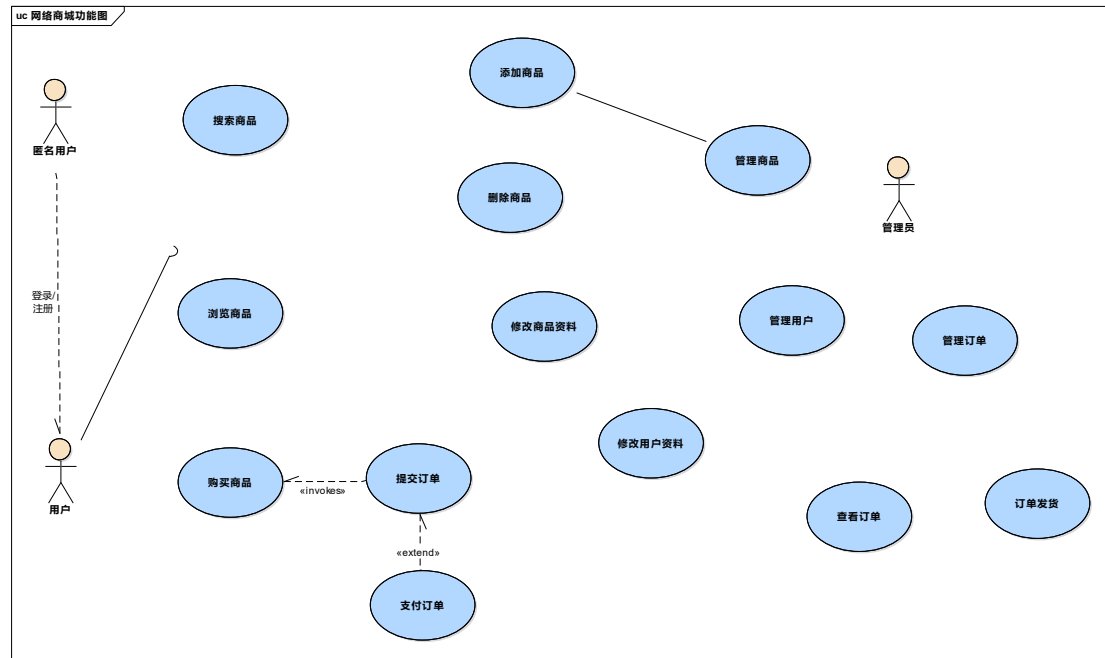
### 2.2.3 假定和约束

(1) 2022 年 6 月之前完成系统的分析、设计、编码和系统测试工作。

(2) 2022 年 7 月之前完成系统的部署和验收测试工作，系统正式上线运行。

## 2.3 需求规定

### 2.3.1 系统用例图



#### (1) 用户管理模块

注册: 游客通过本用例进行注册, 并成为网上商店的会员;

登录: 游客通过本用例登录, 从而可以用会员身份访问网站;

修改个人信息: 会员通过本用例进行个人信息管理——个人信息包含的很多子项均在此用例中统一修改, 故将其结合在同一用例中;

管理会员信息: 管理员可以通过本用例进行用户信息管理, 并且对某些用户进行封禁等操作。

#### (2) 库存管理模块

添加商品: 管理员通过本用例进行商品信息的录入;

删除商品: 管理员通过本用例进行上架商品的删除;

修改商品信息: 管理员通过本用例进行某条商品信息的修改。

#### (3) 订单管理系统

查看订单: 会员通过本用例查看所有与自己相关的订单;

取消订单: 会员通过本用例取消一个未完成的订单

提交订单: 会员通过本用例提交订单, 并且进入付款过程;

付款: 会员通过本用例付款, 此用例的具体实现需要与第三方支付系统进行交互。

### 2.3.2 分析类图

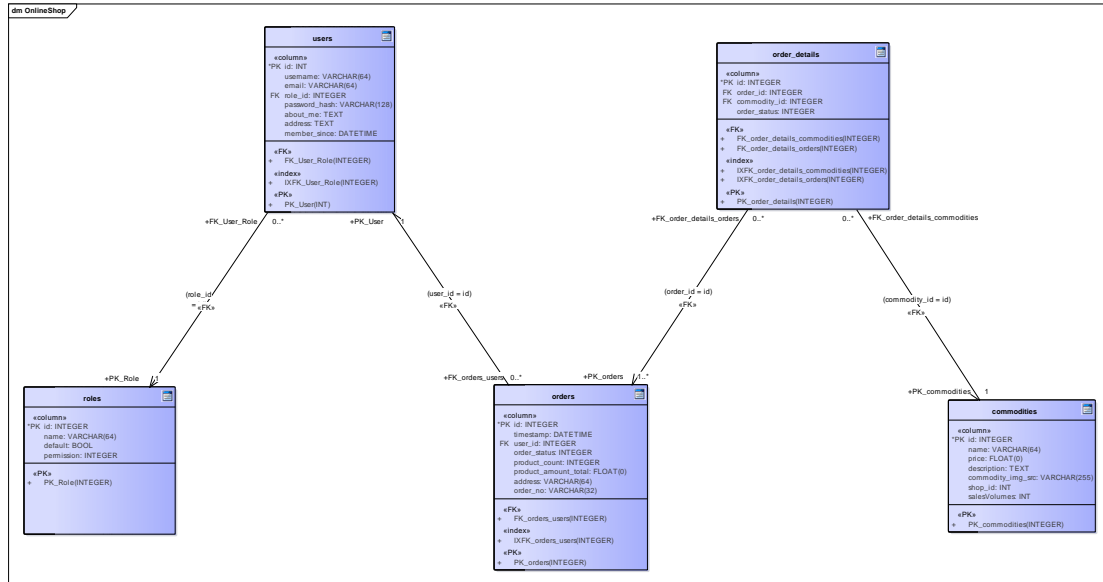
- (1) 用户页面、商品管理页面、订单管理页面, 用户信息管理页面四个控制类从属于业务动作层, 其中后三者为用户界面类的子类, 这些类接收页面的输入,



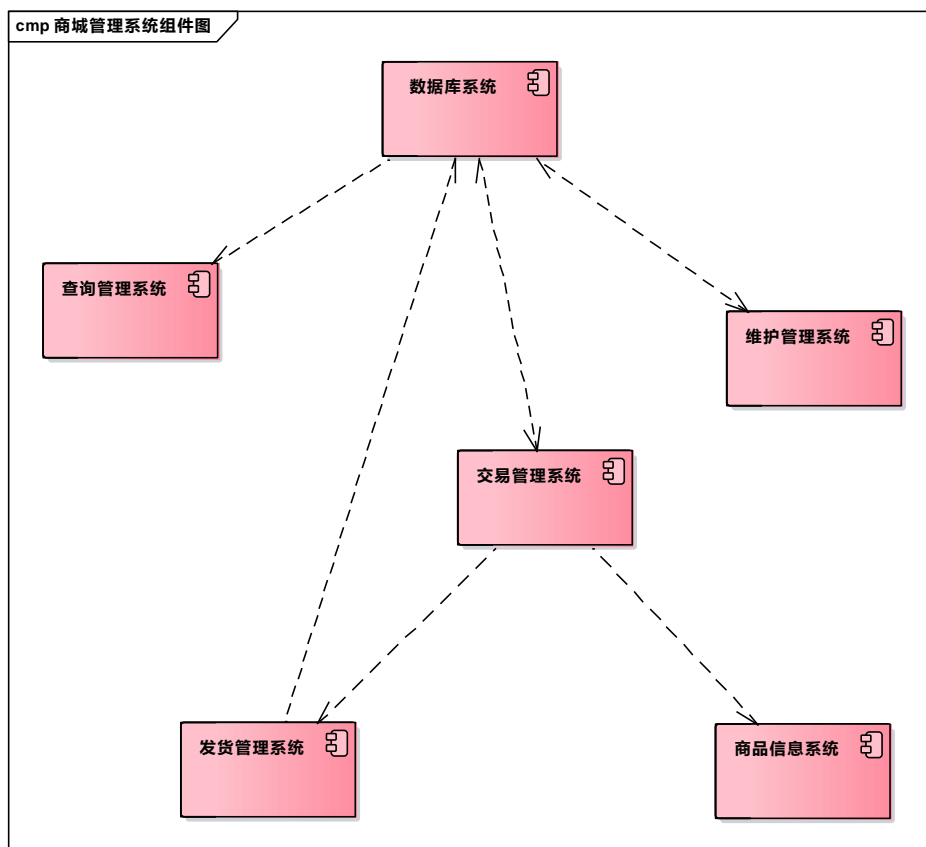
并且触发相应的业务动作

(2) 商品管理类、订单管理类, 用户信息管理类三个控制类属于业务实现层, 被业务动作层触发, 并完成相应的业务逻辑。

(3) 商品、订单和用户是三个系统范围内重要的实体类。



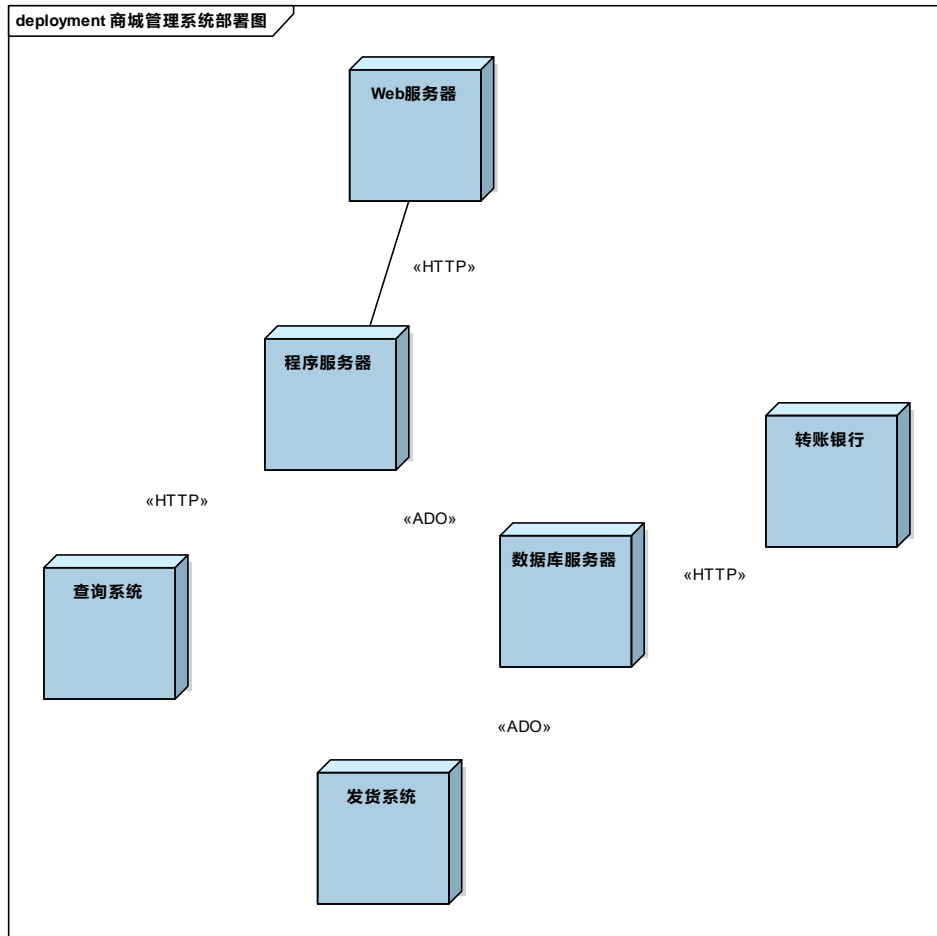
### 2.3.3 组件图



- (1) 本系统为满足扩展性的需求，对系统进行了分层设计。系统中有六个组件，是组成系统的现实表达。
- (2) 查询管理:系统用来处理用户的查询申请，在数据库里找寻匹配的数据，返回结果。
- (3) 交易管理:系统用来处理用户提交的订单和确认付款。
- (4) 维护管理:管理员对系统进行日常必要的维护和功能扩展。
- (5) 发货管理:系统根据确认的订单，提出发货指令;相关人员进行发货处理，并将结果反馈给系统。
- (6) 商品信息:关于商品介绍的详细信息。
- (7) 数据库系统: 用来存储各种相关的信息，实现对信息进行分类和查找等操作。

#### 2.3.4 部署图

- (1) 部署图包括系统运行时的硬件节点，以及在这些节点上运行的软件组件将在何处物理地运行，以及它们将如何彼此通信。
- (2) Web 服务器:提供网络协议，支持网页功能，网上操作。
- (3) 数据库服务器:负责数据存储。
- (4) 银行转账:确认用户付款，并将信息反馈给数据库。
- (5) 发货系统:根据用户订单，对用户发货。
- (6) 查询系统:用户在 Web 服务器的支持下，对数据库进行数据（商品）查询。
- (7) 程序服务器:处理所有相关数据，并进行归类。



## 2.4 对性能的规定

### 2.4.1 精度

密码字符不得有大小写偏差。

### 2.4.2 时间特性要求

- (1) 后台的网站管理的操作响应时间  $\leq 2S$ 。
- (2) 管理员管理订单、商品、物流操作的响应时间  $\leq 3S$ 。
- (3) 用户通过浏览器完成的购物操作的响应时间  $\leq 5S$ 。

### 2.4.3 灵活性

- (1) 琳琅-购物交易网站能够兼容主流的浏览器，包括 Firefox、Chrome 等。

- (2) 目前服务器中安装的是 32 位版本的 Linux 操作系统，今后在升级到 64 位服务器和 64 位版本的 Linux 操作系统之后，琳琅-购物交易网站仍然能够正常运行。

#### **2.4.4 其他专门要求**

- (1) 琳琅-购物交易网站应为用户提供登录的功能，以防止非法用户使用琳琅-购物交易网站进而破坏其中的数据。
- (2) 琳琅-购物交易网站应具有良好的可扩展性，允许未来方便地增加新的业务管理功能。
- (3) 琳琅-购物交易网站应具备连续运行 5 个月（约一个学期）的能力。

### **2.5 运行环境规定**

#### **2.5.1 设备**

- (1) 戴尔 PowerEdge SC430 塔式服务器 / Intel Pentium4 双核处理器 2.8GHz / 512M DDR2 内存 / 80GB 7200RPM SATA 硬盘。
- (2) 联想锋行 King2005 台式机 / AMD Athlon64 FX-57 处理器 2.8GHz / 1G DDR 内存 / 320GB 7200RPM IDE 硬盘。

#### **2.5.2 支持软件**

- (1) Red Hat Linux 9.0 操作系统。
- (2) Tomcat 应用服务器。
- (3) MySQL 5.0 关系数据库 DBMS。

#### **2.5.3 接口**

与购物卡管理中心通过 TCP 协议通信，数据包格式遵守《购物卡管理中心数据交换标准》定义的相关格式。

## 第3章 概要设计

### 3.1 系统构架

#### 3.1.1 系统总体布局

考虑到交易系统对存储功能、搜索功能和支付功能的要求,该系统采用基于 Web 浏览器/服务器模式的分布式运行模式。这种运行模式使用方便,可靠性、安全性高,可维护性、可扩展性、可变更性较好。同时,使用 Flask 框架设计网站。使用 MySQL 数据库。

#### 3.1.2 系统总体结构设计

总体设计的主要内容就是合理地对系统进行分解,将一个复杂的系统设计转化为若干个子系统。为了保证系统分解的准确性和合理性,该系统严格遵循系统分解的原则,力求合理调配商品资源,提升管理水平,提高服务质量,减少工作中的差错,充分实现数据的最小冗余和最大共享,在系统分析阶段得到的组织管理机构基础上,按照职能将系统划分为处理订单管理、用户管理、商品管理等三个相对独立的子系统,最终形成规范化的服务体系。

### 3.2 模块划分

#### 3.2.1. 用户管理模块

- (1) 注册: 游客通过本用例进行注册,并成为网上商店的会员;
- (2) 登录: 游客通过本用例登录,从而可以用会员身份访问网站;
- (3) 修改个人信息: 会员通过本用例进行个人信息管理——个人信息包含的很多子项均在此用例中统一修改,故将其结合在同一用例中;
- (4) 管理会员信息: 管理员可以通过本用例进行用户信息管理,并且对某些用户进行封禁等操作。

#### 3.2.2. 商品管理模块

- (1) 添加商品: 管理员通过本用例进行商品信息的录入;

- (2) 删除商品: 管理员通过本用例进行上架商品的删除;
- (3) 修改商品信息: 管理员通过本用例进行某条商品信息的修改。

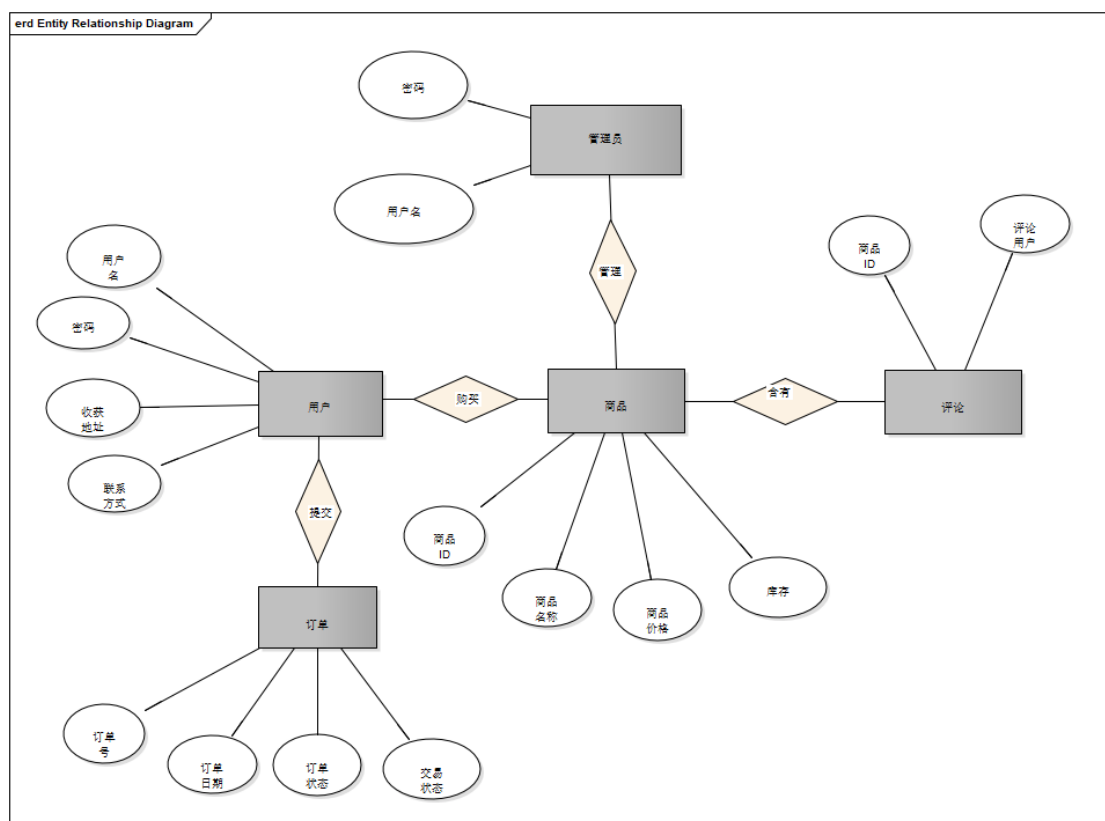
### 3.2.3. 订单管理模块

- (1) 查看订单: 会员通过本用例查看所有与自己相关的订单;
- (2) 提交订单: 会员通过本用例提交订单, 并且进入付款过程;
- (3) 付款: 会员通过本用例付款, 此用例的具体实现需要与第三方支付系统进行交互。

### 3.2.4 评论管理系统

- (1) 查看评论: 用户通过商品页面查看商品评论
- (2) 发表评论: 用户购买商品后可以发表评论

### 3.2.5 功能 ER 图



## 3.3 网站结构——三层架构

网站结构分为三个逻辑层:

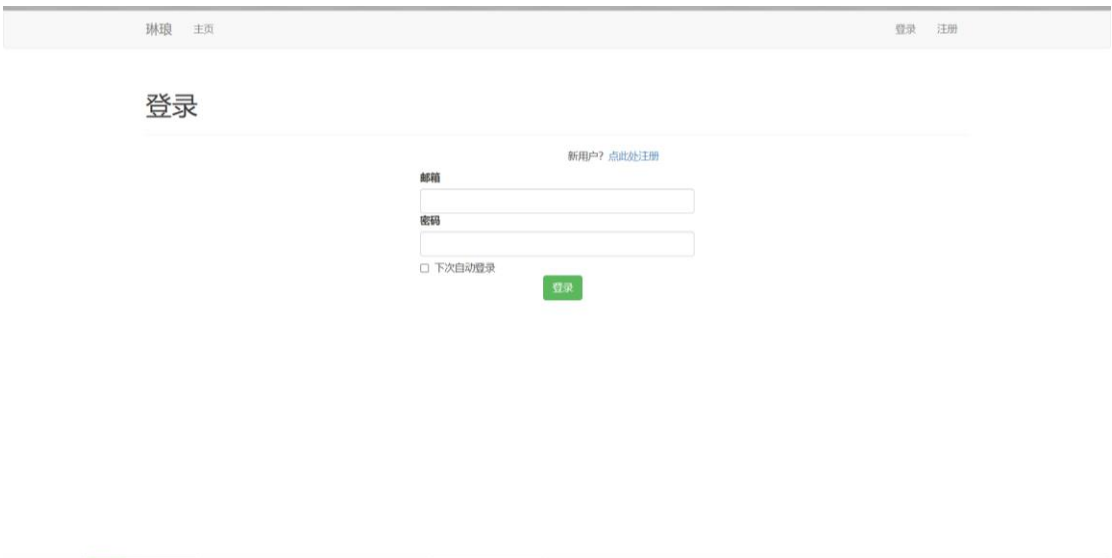
- (1) web层：它为客户端提供对应用程序的访问，它由HTML+BootStrap并由Jinja2引擎进行渲染
- (2) 业务逻辑层：业务逻辑层主要负责对数据层的操作。也就是说把一些数据层的操作进行组合..可以通过编程来实现对数据库的访问，最终得到想要的结果。
- (3) 数据访问层：数据访问层主要是对原始数据(数据库或者文本文件等存放数据的形式)的操作层，具体为业务逻辑层或表示层提供数据服务。

### **3.4 数据设计**

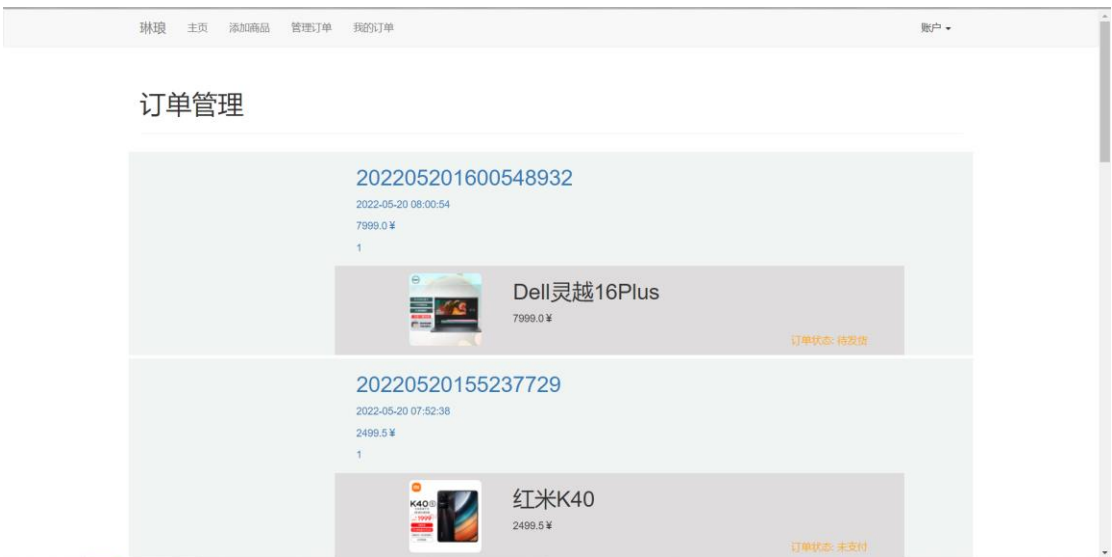
## 第4章 详细设计（模块流程图，状态图，局部变量）

### 4.1 画面设计

#### 4.1.1 输入/输出设计（登录页面）



#### 4.1.2 人机界面设计（管理页面）



### 4.2 代码设计

对用户，为了便于管理，采用分级权限：



ADDCART: 添加购物车

BUY: 购买

COMMENT: 评论

MERCHANT: 商家

MODERATE: 协管（即管理用户发言）

ADMIN: 管理员权限（拥有整个系统的权限）

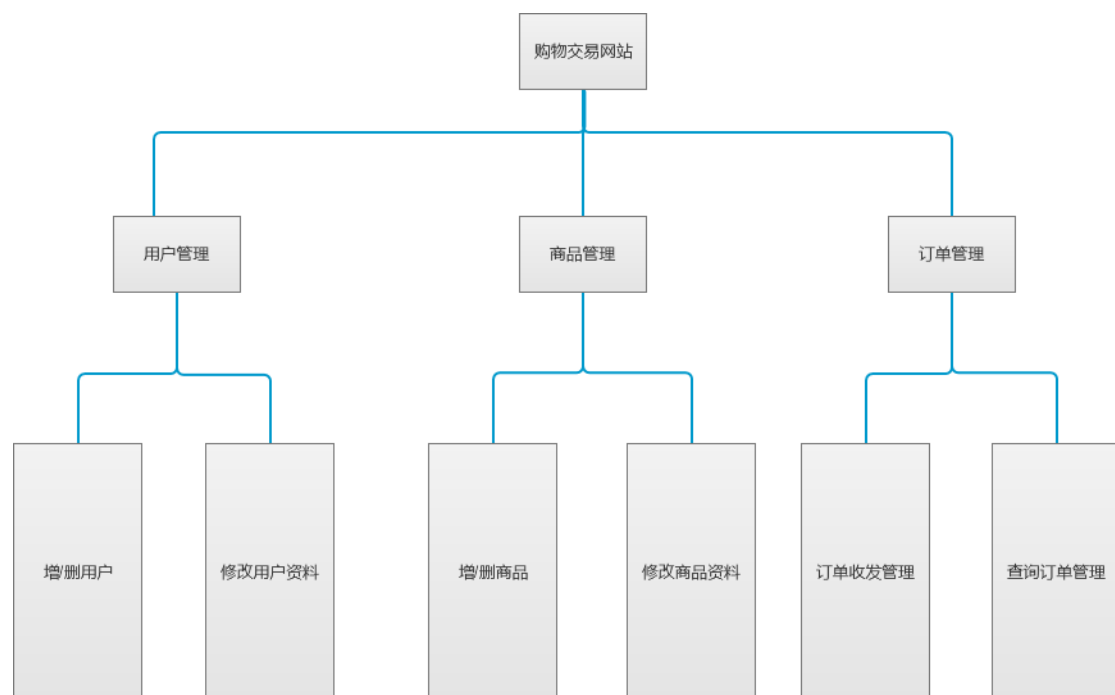
普通用户具有购买、添加购物车、评论的权限

管理员有所有权限。

(1) 对商品，采用整型变量作为 ID，由数据库自动管理

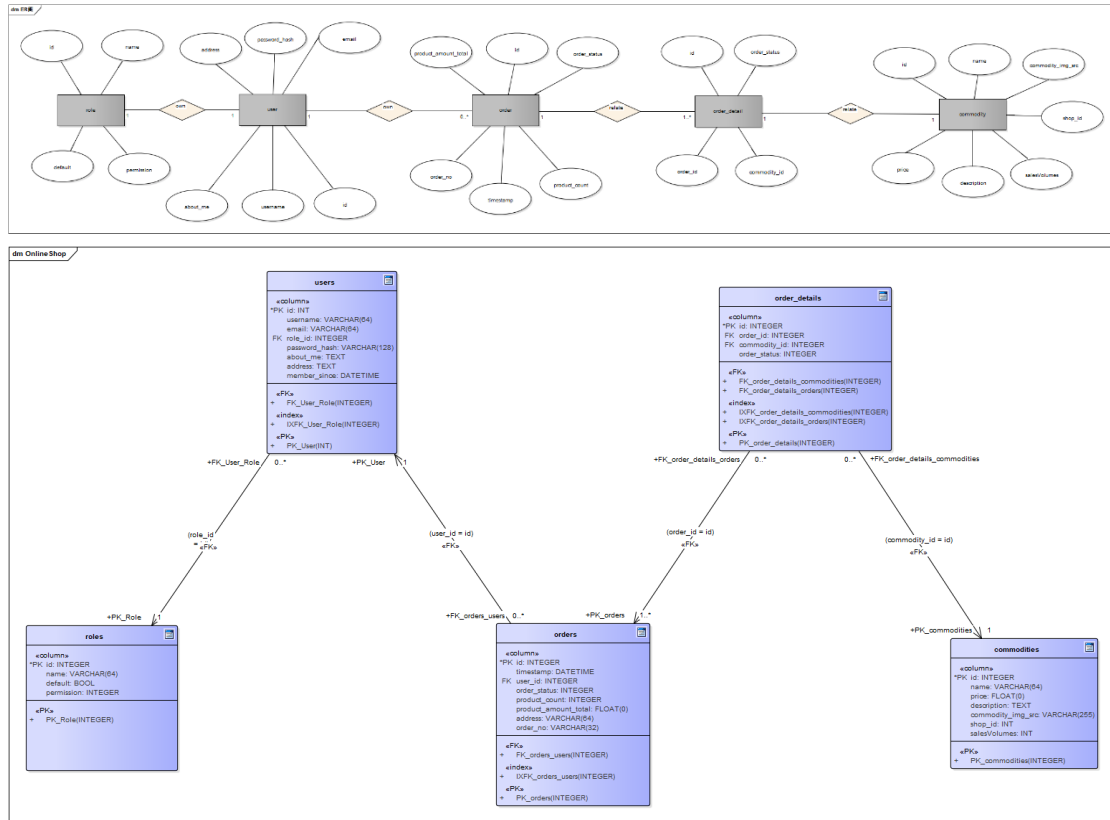
(2) 对订单，采用整型变量作为 ID，由数据库自动管理，订单编号则由购买时间+随机的四位数构成。

### 4.3 模块流程图



## 第5章 数据库设计

### 5.1. 数据实体-关系图



### 5.2. 数据实体描述

#### 5.2.1. 数据库表

1. 角色表: roles
2. 用户表: users
3. 商品表: commodities
4. 订单表: orders
5. 订单明细表: order\_details

#### 5.2.2. 数据实体

roles 表:

id(INTEGER): 数据项编号  
names(VARCHAR(64)): 角色名称  
default(BOOL): 表明是否为默认类型  
permission(INTEGER): 表示权限等级

users 表:

id(INTEGER) : 数据项编号  
username(VARCHAR(64)): 用户名称  
email(VARCHAR(64)): 用户邮箱  
role\_id(INTEGER): 角色 ID, 外键, 表明用户角色  
password\_hash(VARCHAR(128)): 密码哈希值  
about\_me(TEXT): 个人简介  
address(VARCHAR(64)): 个人地址  
member\_since(DATETIME): 注册时间

其余表详见 ER 图。

### 5.3. 实体关系描述

1. ( roles : users ) ( 1 : n )

一个用户只能对应一个角色, 一个角色可以对应多个用户。

2. ( users: orders ) ( 1 : n )

一个订单只能对应一个用户, 一个用户可以对应多个订单。

3. ( orders: order-details ) ( 1 : n )

一个订单只能对应一个用户, 一个订单可以对应多个订单明细项。

4. ( order-details: commodities ) (1: n)

一个订单明细只能对应一个商品, 一个商品可以对应多个订单明细项。

## 5.4. 关系模型描述

数据库表名: roles

主键: id

外键: 无

索引: 无

约束: 无

数据库表名: users

主键: id

外键: role\_id

索引: 无

约束: 无

数据库表名: orders

主键: id

外键: user\_id

索引: 无

约束: 无

数据库表名: order\_details

主键: id

外键: order\_id; commodity\_id

索引: 无

约束: 无

数据库表名: commodity

主键: id

外键: 无

索引: 无

约束: 无

## 5.5. 数据库一致性设计

(用标准 SQL 语言中创建表的语句描述)

### 1.Commodities 表

```
CREATE TABLE `commodities` (  
    `id` int NOT NULL AUTO_INCREMENT,  
    `name` varchar(64) NOT NULL,  
    `price` float NOT NULL,  
    `description` text,  
    `commodity_img_src` varchar(255) DEFAULT NULL,  
    `shop_id` int DEFAULT NULL,  
    `salesVolumes` int DEFAULT NULL,  
    PRIMARY KEY (`id`),  
    KEY `ix_commodities_name` (`name`),  
    KEY `shop_id` (`shop_id`),  
    CONSTRAINT `commodities_ibfk_1` FOREIGN KEY (`shop_id`)  
REFERENCES `shops` (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=21 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

### 2.roles 表

```
CREATE TABLE `roles` (  
    `id` int NOT NULL AUTO_INCREMENT,  
    `name` varchar(64) DEFAULT NULL,  
    `default` tinyint(1) DEFAULT NULL,  
    `permission` int DEFAULT NULL,  
    PRIMARY KEY (`id`),  
    UNIQUE KEY `name` (`name`),  
    KEY `ix_roles_default` (`default`)  
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

### 3.users 表

```
CREATE TABLE `users` (  
  `id` int NOT NULL AUTO_INCREMENT,  
  `username` varchar(64) DEFAULT NULL,  
  `email` varchar(64) DEFAULT NULL,  
  `role_id` int DEFAULT NULL,  
  `password_hash` varchar(128) DEFAULT NULL,  
  `about_me` text,  
  `address` text,  
  `member_since` datetime DEFAULT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `ix_users_email` (`email`),  
  UNIQUE KEY `ix_users_username` (`username`),  
  KEY `role_id` (`role_id`),  
  CONSTRAINT `users_ibfk_1` FOREIGN KEY (`role_id`) REFERENCES `roles`  
  (`id`)  
) ENGINE=InnoDB AUTO_INCREMENT=5 DEFAULT CHARSET=utf8mb4  
COLLATE=utf8mb4_0900_ai_ci;
```

### 4.orders 表

```
CREATE TABLE `orders` (  
  `id` int NOT NULL AUTO_INCREMENT,  
  `timestamp` datetime DEFAULT NULL,  
  `user_id` int DEFAULT NULL,  
  `order_status` int DEFAULT NULL,  
  `product_count` int NOT NULL,  
  `product_amount_total` float NOT NULL,  
  `address` varchar(64) NOT NULL,  
  `order_no` varchar(32) NOT NULL,  
  PRIMARY KEY (`id`),  
  UNIQUE KEY `ix_orders_order_no` (`order_no`),
```

```

KEY `ix_orders_timestamp` (`timestamp`),
KEY `user_id` (`user_id`),
CONSTRAINT `orders_ibfk_2` FOREIGN KEY (`user_id`) REFERENCES
`users` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=15 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;
5.order_details 表
CREATE TABLE `order_details` (
  `id` int NOT NULL AUTO_INCREMENT,
  `order_id` int DEFAULT NULL,
  `commodity_id` int DEFAULT NULL,
  `order_status` int DEFAULT NULL,
  PRIMARY KEY (`id`),
  KEY `commodity_id` (`commodity_id`),
  KEY `order_id` (`order_id`),
  CONSTRAINT `order_details_ibfk_1` FOREIGN KEY (`commodity_id`)
REFERENCES `commodities` (`id`),
  CONSTRAINT `order_details_ibfk_2` FOREIGN KEY (`order_id`)
REFERENCES `orders` (`id`)
) ENGINE=InnoDB AUTO_INCREMENT=15 DEFAULT CHARSET=utf8mb4
COLLATE=utf8mb4_0900_ai_ci;

```

## 第6章 系统测试

### 6.1 测试准备与说明

#### 6.1.1 编写目的

为软件开发项目管理者、软件工程师、系统维护工程师、测试工程师提供关于项目系统整体功能和性能的测试指导，同时也是用户确定软件是否完整测试的重要依据。

#### 6.1.2 测试目标

在用户使用软件之前，尽可能的发现软件中存在的错误和不合理之处，排除软件中潜在的错误，最终把高质量的软件系统交付给用户。系统的测试覆盖范围：功能、性能、UI、安全性、兼容性、容量。

#### 6.1.3 测试参考文档

GBT 15532-2008 计算机软件测试规范

GBT 9386-2008 计算机软件测试文档编制规范

#### 6.1.4 测试配置要求

运行系统：windows10

数据库：MySQL 数据库

#### 6.1.5 测试方法

系统的功能测试选用了手工测试，运用黑盒测试中的等价类划分、边界值分析、错误推测、因果图法。系统 UI 方面的测试包括：易用性测试、规范性测试、帮助设施测试、合理性测试、美观与协调性测试、独特性测试、快捷方式组合测试。系统的安全性、兼容性、安装与反安装、配置测试也是手工测试。单元测试采用的方法是白盒测试，功能测试采用黑盒测试。



## 6.2 测试策略

### 6.2.1 单元测试

首先依照系统、子系统和模块进行划分，但最终的单元必须是功能模块，或面向对象过程中的若干个类，单元测试是对功能模块进行正确性检验的测试工作，也是后续测试的基础。目的在于发现各模块内部可能存在的各种差错，因此需要从程序的内部结构出发设计测试用例。

编写代码，使用unittest对各个功能进行测试  
测试结果如下：

Coverage report: 54%						
coverage.py v6.4, created at 2022-05-23 11:15 +0800						
Module	statements	missing	excluded	branches	partial	coverage
app\__init__.py	30	0	0	0	0	100%
app\auth\__init__.py	3	0	0	0	0	100%
app\auth\forms.py	41	6	0	18	0	80%
app\auth\views.py	113	88	0	40	0	16%
app\decorators.py	14	3	0	2	0	69%
app\email.py	15	9	0	0	0	40%
app\main\__init__.py	6	1	0	0	0	83%
app\main\errors.py	8	3	0	0	0	62%
app\main\forms.py	36	7	0	14	0	74%
app\main\views.py	143	114	0	30	0	17%
app\models.py	189	17	0	50	11	87%
<b>Total</b>	<b>598</b>	<b>248</b>	<b>0</b>	<b>154</b>	<b>11</b>	<b>54%</b>
coverage.py v6.4, created at 2022-05-23 11:15 +0800						

### 6.2.2 集成测试

集成测试也叫组装测试或联合测试。通常，在单元测试的基础上需要将所有的模块按照设计要求组装成系统，这时需要考虑的问题如下：

- (1) 把各个模块连接起来，模块接口的数据是否会丢失；
- (2) 一个模块的功能是否会对另一个模块的功能产生不利的影响；
- (3) 各个子功能组合起来，能否达到预期要求的父功能；
- (4) 全局数据结构是否有问题；
- (5) 单元模块的误差累积起来，是否会放大，从而达到不能接受的程度。我们在组装时可参考采用一次性组装方式或增值式组装方式；

### 6.2.3 系统测试

系统测试目的是在于验证软件的功能和性能及其他特性是否与用户的要求一致，主要是以下类型的测试；

(1) 功能测试：验证系统功能是否符合其需求规格说明书，核实系统功能上是否完整，没有冗余和遗漏的功能。

测试范围：验证每个模块的业务功能是否实现

测试目标：登录注册功能，用户信息修改功能，用户商品购买功能，管理员添加删除商品功能，管理员管理订单功能

技术：黑盒测试，边界测试，等价类划分法测试

开始标准：代码完成后，功能完成，设计测试用例

完成标准：测试用例通过并且高级缺陷全部解决

(2) 用户界面测试：测试用户界面是否具有导航性、美观性、行业或公司的规范性、是否满足设计中要求的执行功能、详细介绍如下

其中，Web 测试通用方法可以参考《Web 测试检查点总结》

测试目标：登录窗口，注册窗口，用户资料界面，用户资料更改界面，管理员资料更改界面

网站主页，用户商品详情页面，我的订单界面，管理员商品详情界面，管理员添加商品界面，管理员修改商品界面，管理员订单界面。

技术：web 测试通用方法

工具与方法：手工测试，目测

开始标准：界面开发完成

完成标准：UI 符合可接受标准，能够保证用户界面的友好性，易操作性，符合，用户的操作习惯

(3) 性能测试：测试相应时间、事务处理效率和其他时间敏感的问题。

测试目标：测试多用户，管理员同时登录的情况

技术：web 测试通用方法

工具与方法：手工测试，目测

开始标准：前后端开发完毕

完成标准：支持多用户同时在线与管理在线

(4) 兼容性测试：测试软件在不同平台上使用的兼容性。

测试目标：edge 浏览器，火狐浏览器，qq 浏览器，360 浏览器

技术：web 测试通用方法，白盒方法，黑盒方法

开始标准：前后端开发完毕

完成标准：支持多种浏览器的正常运行

## 6.3. 验收测试

用户新增或修改内容，以待用户反馈问题确认：

## 6.4 通过测试的标准

一般有“基于测试用例”和“基于缺陷密度”两种评比准则，在这里我们采用前者。

- (1) 功能性测试用例通过率达到 100%
- (2) 非功能性测试用例通过率达到 95%
- (3) 没有高于优先级 3 以上的问题

备选通过方法：根据实际情况共同讨论确定本测试阶段是否结束。（详细的系统测试通过标准可参考《系统测试各阶段准入准出规则》）

## 6.5 测试用例

用于软件测试的账号信息如下

## 6.6 测试关注点

### 6.6.1 文本输入框

- (1) 检查空数据；
- (2) 检查过长数据（超出空间本身的长度和数据库中改字段所允许的长度）；
- (3) 检查特殊字符，尤其是数据库中不允许的字符，甚至回车字符、空格字符等；
- (4) 检查字符类型，比如应该输入数字的文本框输入英文字符；
- (5) 中文字符的处理；
- (6) 对于日期时间型数据，检查格式正确性以及时间日期的合理性。比如开始时间不能晚于结束时间等；

### 6.6.2 下拉列表

- (1) 列表数据是否正确、完整；
- (2) 下拉列表与其他空间的联动关系；
- (3) 是否允许多选；

#### 6.6.3 增加数据

- (1) 数据个数的上限;
- (2) 重复数据处理, 尤其是键值的重复;
- (3) 相关表格的更新;
- (4) 检查多次使用 back 键的情况, 在有 back 的地方, back 回到原页面, 再 back 重复多次, 看是否会出错;

#### 6.6.4 修改数据

- (1) 不能破坏数据库数据的关联和完整;
- (2) 重复数据处理, 尤其是键值的重复;
- (3) 修改登录用户本身信息时对系统的影响;
- (4) 修改正在使用的数据;
- (5) 检查多次使用 back 键的情况, 在有 back 的地方, back, 回到原页面, 再 back, 重复多次, 看是否会出错。

#### 6.6.5 删除数据

- (1) 不能破坏数据库数据的关联和完整;
- (2) 删除正在使用的数据;
- (3) 删除登录用户本身;

#### 6.6.6 查询数据

- (1) 多条件组合查询的正确性;
- (2) 多次连续查询正确性;

#### 6.6.7 数据导入导出

- (1) 导入数据格式要求不应太严格, 提示明确;
- (2) 导出数据不应乱码;

#### 6.6.8 数据接入与处理

- (1) 数据接入方式是否全部可用, 数据是否能正确接入;
- (2) 数据处理方式是否全部可行;
- (3) 数据的动态监测是否正确无误;

#### 6.6.9 其他

- (1) 对网络故障的提示;
- (2) 同一用户多次登录;
- (3) 内存使用情况;
- (4) 压力测试, 系统承受能力, 多用户同时登录使用。

## 第7章 效果和使用展示

### 7.1 系统展示

#### 7.1.1 用户管理

##### 1.用户登录界面：

琳琅 主页

登录 注册

登录

新用户? [点此处注册](#)

邮箱

密码

☐ 下次自动登录

登录

简介：用户在此输入账号和密码即可登录账户，未注册的用户可点击右上角注册按钮或是注册链接进行登录。

##### 2.用户注册界面：

琳琅 主页

登录 注册

注册

邮箱

用户名

密码

再次输入密码

送货地址

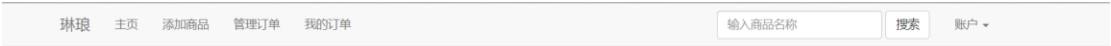
个人介绍

注册

用户在此输入必要的信息并且按照规定的格式便可完成注册。

Hello, Stranger ! 欢迎来到琳琅购物商城

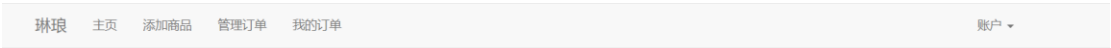
在用户完成登录后，首页的欢迎标语会换成如下



Hello, nikoYoung(Administrator) ! 欢迎来到琳琅购物商城

3.查看用户信息

1) 管理员权限



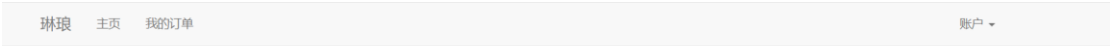
用户资料

- 用户名:  
James2
- 个人介绍:  
AHNUer
- 邮箱:  
wangjunwei@ahnu.edu.cn
- 注册时间:  
2022-04-18 12:35:34
- 送货地址:  
安徽师范大学花津校区

修改资料(ADMIN)

2) 普通用户

1.查看他人的用户资料:



用户资料

- 用户名:  
nikoYoung(Administrator)
- 个人介绍:  
I am the king of the kings.

2.查看自己的用户资料

## 用户资料

- 用户名:  
James2
- 个人介绍:  
AHNUer
- 邮箱:  
wangjunwei@ahnu.edu.cn
- 注册时间:  
2022-04-18 12:35:34
- 送货地址:  
安徽师范大学花津校区

编辑资料

修改密码

## 4. 修改用户信息界面

### 1) 管理员修改界面

## 编辑用户资料

邮箱

1992592417@qq.com

角色

Administrator ▾

用户名

nikoYoung(Administrator)

送货地址

安徽师范大学花津校区

个人介绍

I am the king of the kings.

修改

### 2) 普通用户修改界面

## 编辑用户资料

用户名

James2

送货地址

安徽师范大学花津校区

个人介绍

AHNUer

修改

## 5.修改密码界面

琳琅 主页 添加商品 管理订单 我的订单 账户 ▾

更改密码

旧密码

新密码

再次输入密码


更改密码

## 7.1.2 商品管理

### 1.主页

琳琅 主页 添加商品 管理订单 我的订单 输入商品名称 搜索 账户 ▾

Hello, nikoYoung(Administrator) ! 欢迎来到琳琅购物商城




小米 K40S  
全新旗舰手机  
骁龙870 三星 E4 旗舰直屏  
全新升级的 VC 散热 全新升级的 67W 旗舰快充 全新升级的 OIS 光学防抖 不顾一切的全新设计 这个 K40S，天生就是个狠角色

¥1999

立即购买


- 红米K40
- 2499.5 ¥
- 经典的骁龙870 经典的三星 E4 旗舰直屏 全新升级的 VC 散热 全新升级的 67W 旗舰快充 全新升级的 OIS 光学防抖 不顾一切的全新设计 这个 K40S，天生就是个狠角色



RTX3050显卡  
11代酷睿处理器  
20000转固态硬盘  
轻薄便携商务 学生办公利器


¥7999.0

- Dell灵越16Plus
- 7999.0 ¥
- 英特尔酷睿16英寸轻薄全能本设计师 i7-11800H 16G 512G RTX3050 3K屏 灰蓝



华为手环6  
标准版  
限时优惠 40 元  
¥229


- 华为 HUAWEI 手环6 标准版
- 249.0 ¥
- 运动手环 智能手环 炫彩全面屏/2周续航/96种运动 曜石黑





华硕 (ASUS) RTX3060

¥3499.0

- 华硕 (ASUS) RTX3060
- 3499.0 ¥
- 电脑电竞游戏光追独立显卡 DUAL-RTX3060-O12G-V2







简介：主页按照销量降序展示商品，并用弹性布局自动分布。并且用分页机制来控制单个页面商品数为16个。主页上还有搜索框可以进行关键词模糊搜索（%关键字%）。



- 超旗舰 超频版 电竞游戏设计智能学习 电脑独立显卡

- 6799.0 ¥
- 是否支持PCI-E4.0: 支持PCI-E4.0散热 风扇: 三风扇接口: DP显卡型号: RTX 3080显存类型: GDDR6X显存容 量: 10GB显存位宽: 320bit
- 芯片组: NVIDIA 散热风扇: 涡轮风扇 显存位宽: 256bit 质保期: 3年接口: HDMI, DP显卡型号: 其他显存容量: 8GB显存类型: GDDR6
- 车 全消光黑铁灰 700C\*430MM XS适 合 155-165cm

- 雅马哈(YAMAHA) PSR- F51 儿童成 年初学入门娱乐演奏电子琴61键
- 649.0 ¥
- 品牌: 雅马哈 (YAMAHA) 商品名 称: 雅马哈PSR- F51商品毛重: 4.7kg 音色数: 101-200种示范曲数: 51-100 首复音数: 32个颜色: 黑色键盘数: 61 键

- Hi nova 9
- 2569.0 ¥
- 5G全网通 120Hz高刷 后置5000万超感 光主摄 支持66W疾速快充 8+128GB梦 幻星河手机 标配无充

- Adidas阿迪达斯卡塔尔世界杯足球
- 195.0 ¥
- 逐梦之旅5号比赛训练成人4号儿童小学 生足球zuqlu 5号【世界杯款】H57799

- Adidas阿迪达斯足球尤文图斯
- 129.0 ¥
- 尤文图斯C罗5号成人儿童比赛训练耐磨 用球 白黑【5号】GT3917

Hello, nikoYoung(Administrator) ! 欢迎来到琳琅购物商城

- 索尼 (SONY) 摄像机高清4K
- 16220.0 ¥
- 高清4K 专业广播级手持便携式摄影机 直播 会议 录课 采访 抖音快手 HXR- NX80摄像机 (HDMI)

- 雅迪 (yadea) 电动车
- 2098.0 ¥
- 小飞侠新国标48V12AH锂电可提取电池 电动车成人踏板电动自行车电瓶车黑色

### 3. 商品详情页

## 商品资料



- 商品名:  
红米K40
- 商品价格:  
2499.5
- 商品介绍:  
经典的骁龙870 经典的三星 E4 旗舰直屏 全新升级的 VC 散热 全新升级的 67W 旗舰快充 全新升级的 OIS 光学防抖 不顾一切的全新设计 这个 K40S，天生就是个狠角色
- 销量:  
2

修改商品资料(ADMIN)

删除商品(ADMIN)

购买

简介：此界面详细介绍商品的信息以及有购买按钮，当以管理员身份打开页面时会显示修改按钮和删除按钮。

### 4. 添加商品页面（管理员权限）

## 注册商品

商品名

价格

商品介绍

图片地址

添加商品

### 5.修改商品页面（管理员权限）

## 修改商品

商品名

红米K40

价格

2499.5

商品介绍

经典的骁龙870  
经典的三星 E4 旗舰直屏

图片地址

https://img14.360buyimg.com/n0/jfs/t1/110466/25/20655/

保存修改

删除商品按钮（管理员权限）

天生就是个狠角色

• 销量:

2

修改商品资料(ADMIN)

删除商品(ADMIN)

购买

7.1.3 订单管理

1.订单管理界面（需要管理员权限）

琳琅 主页 添加商品 管理订单 我的订单

账户


订单管理

202205201600548932

2022-05-20 08:00:54

7999.0

1



Dell灵越16Plus

7999.0


订单状态: 待发货

20220520155237729

2022-05-20 07:52:38

2499.5

1



红米K40

2499.5

订单管理，点击对应订单项可以查看明细，

琳琅 主页 添加商品 管理订单 我的订单

账户

订单明细

购买者: nikoYoung(Administrator)

配送地址: 安徽师范大学花津校区

订单编号: 202205201600548932

订单创建时间: 2022-05-20 08:00:54

订单金额: 7999.0

商品总数: 1

订单状态: 已支付



Dell灵越16Plus

7999.0

发货

订单状态: 待发货

订单明细，在此界面可以查看订单的详细信息，并且管理员可以对订单进行发货操作，对应的用户可以在此界面进行收货操作。

## 2.我的订单界面（普通用户权限）



在此可以查看用户已订购的订单，以及点击对应的订单项查看详细信息，并对订单进行支付以及收货操作。

## 7.2 使用说明

### 7.2.3 管理员使用说明

- 1.权限说明：管理员拥有整个系统最高的权限,可以对商品、用户、订单做出增删改的操作。
- 2.管理员申请：管理员身份是以邮箱来确认，最初的管理员是根据config文件中的FLASK\_ADMIN环境变量来赋予，在这之后，其余的管理员身份可以由最初的管理员来赋予。
- 3.商品管理：以具有管理员身份的账号登陆后，在首页导航栏会显示添加商品按钮，而在商品详情页会多显示修改信息按钮和删除商品按钮，点击对应按钮即可执行对应操作。
- 4.用户管理：以具有管理员身份的账号登陆后，通过以下链接形式访问用户详情页：<http://127.0.0.1:5000/user/id>，其中id为对应用户的id，在打开的用户界面中由修改用户资料的按钮，可对用户信息进行修改。
- 5.订单管理：以具有管理员身份的账号登陆后，点击导航栏中的订单管理，可进入订单管理界面，在此界面中，按照时间降序排列了所有订单项，点击订单项即可进入订单详情页，在此界面可进行订单收货、订单发货等操作。

### 7.2.3 用户使用说明

- 1.注册说明：用户在进入网站后，点击导航栏中注册按钮，即可进入注册界面，在此界面按照规定的格式输入必要的信息，点击注册按钮即可完成注册。
- 2.登录说明：用户进入网站后，点击导航栏中登录按钮，进入登录界面，输入正确的账号信息即可登录。登录界面还有“记住我”登陆选项，勾选后登录状态会被保存七天，登陆后的七天内会再次访问网站会自动登录。
- 3.购买说明：用户在登陆后，在首页可查看商品，点击商品可进入商品详情页，在此界面有购买按钮，点击此按钮会生成一个订单，修改好信息后点击提交，界面会自动跳转至生成好的订单界面，在此界面完成支付，订单状态会自动转换成待发货，等待管理员发货，用户收到货之后在此界面点击收货按钮即可完成整个购买流程。
- 4.订单说明：用户购买商品时会生成对应订单，在首页导航栏点击我的订单按钮可查看用户所购买的按时间降序排布的所有的订单。

## 第8章 总结与展望

### 1.1 项目总结

本项目大体可分为前端和后端部分。前端部分使用了HTML+CSS+Jinja2来编写设计并且采用了Bootstrap框架。后端部分采用Python语言开发、使用MySQL作为数据库，使用了Flask框架来完成服务端的编写。

整个开发过程中，学习Flask框架以及一些相关知识耗时约为1个月，正式开始编写到大体完成耗时约两周。通过这个项目我们开发小组收获了很多知识，包括：

- 1) 通过网上商城这个实战项目的开发，不仅了解到了一个项目的业务逻辑，而且掌握了实现相关业务功能的方法。
- 2) 通过这个实战项目，了解到了模块化开发项目的基础结构的搭建，以及项目文件的管理方式。
- 3) 通过这个实战项目，运用封装的接口api文档实现了客户端服务器之间的交互知识。
- 4) Flask框架的大部分基础知识。
- 5) 通过浏览器向服务器请求数据，然后用Jinja2渲染页面。
- 6) 通过运用MySQL增删查改数据，温习了SQL的知识。
- 7) 通过编写HTML界面，学习了BootStrap的知识。
- 8) 通过版本控制，学习了Git的一些相关知识。

### 1.2 项目展望

受限于工期和精力，项目仍有些许部分没有完善，包括但不限于：

- 1) 购物车功能
- 2) 评论功能
- 3) 更详细的订单管理
- 4) 商品分类功能
- 5) 店铺功能
- 6) 更精美的界面设计

在今后的时间里，我们会尽力去完成这个系统，使其更完善。