

# 景德镇陶瓷大学科技艺术学院

## 本科生毕业设计（论文）

题目：基于 HTML5 的社区服务 webapp

学 号： 201330457115

姓 名： 丁文成

系 别： 工程系

专 业： 13 计算机科学与技术

完成日期： 2017.5.25

指导老师： 徐远纯

分类号\_\_\_\_\_

学校代码\_\_\_\_\_

UDC \_\_\_\_\_ 密级 \_\_\_\_\_

学 号 201330457115

景德镇陶瓷大学科技艺术学院

本科生毕业设计（论文）

题目：基于 HTML5 的社区服务 webapp

丁文成

指导老师： 徐远纯 教授 科技艺术学院 工程系

申请学位级别： 工学学士 专业名称： 13 计算机科学与技术

论文提交日期： 2017 年 5 月 25 日

论文答辩日期： 2017 年 5 月 27 日

学位授予单位 景德镇陶瓷大学科技艺术学院

答辩委员会主席： \_\_\_\_\_

论文评阅人： \_\_\_\_\_

2017 年 5 月 27 号

## 摘 要

社区服务是改革开放以来，逐渐新起的一个行业。随着社会经济的持续性发展，人们的物质生活的水平也日益提高，社区的地位和作用也受到了人们的重视。众所周知，现在中国正处于人口老龄化时期，这也使得拓展社区服务发展建设成为了社会发展的必经之路。

本毕业设计在开始之初，是想通过完成程序来实现对于满足社区家庭需求，同时操作简便，快捷的社区服务 webapp。这个 webapp 不仅能体现出网上预订社区服务的便捷性，更能充分的收集和利用人力资源。避免了传统的名片联系方式在查找上的困难，也防止了纸质材料的浪费，为社区服务提供了新的平台。这个 webapp 利用了以下的技术：html5，css3，javascript 技术以及 php 技术来完成，实现了社区人员注册，社区人员登录，预约社区服务等。

**关键词：**社区服务    webapp    HTML5

## Abstract

Community service is a new industry since the reform and opening to the outside world. With the continuous development of social economy, people's material life level is also increasing, and the status and role of the community has also been paid attention to. As we all know, China is now in an age of population aging, which has made it possible for the development of community services to become the only way for social development.

At the beginning of this graduation project, I want to complete the procedure to meet the needs of the community and family, while operating simple and efficient community service webapp. This webapp can not only reflect the convenience of online booking community service, but also fully collect and utilize human resources. It avoids the difficulty of finding the traditional business card contact way, also prevents the waste of paper materials and provides a new platform for community service. The webapp uses the following technologies: HTML5, CSS3, JavaScript technology, and PHP technology to achieve community registration, community login, and community service.

**Key words: community   service   webapp   HTML5**

目 录

摘 要..... I

Abstrac..... II

第一章 绪 论..... 1

    1.1 研究背景与意义..... 1

    1.2 国内外研究现状分析..... 1

第二章 软件开发主要技术介绍..... 2

    2.1 B/S 技术..... 2

    2.2 HTML5 技术..... 2

    2.3 CSS3 技术..... 2

    2.4 JS 技术..... 3

    2.5 PHP 技术..... 4

第三章 系统分析..... 5

    3.1 可行性分析..... 5

    3.2 软件功能需求..... 5

        3.2.1 系统用例图..... 5

    3.3 系统性能需求..... 7

第四章 系统设计..... 8

    4.1 总体设计..... 8

        4.1.1 系统分层介绍..... 8

        4.1.2 数据库设计..... 9

    4.2 详细设计..... 11

        4.2.1 用户中心模块设计..... 11

        4.2.2 社区服务商品模块设计..... 12

        4.2.3 后台管理模块设计..... 11

第五章 系统实现与测试..... 12

    5.1 系统实现..... 12

        5.1.1 用户中心模块的实现..... 12

        5.1.2 订单功能的实现..... 15

        5.1.3 购物车功能的实现..... 19

        5.1.4 后台功能的实现..... 20

    5.2 系统测试..... 30

        5.2.1 测试概述..... 30

        5.2.2 模块单元测试..... 31

结 论..... 33

致 谢..... 34

---

参考文献.....	35
-----------	----

## 第一章 绪 论

### 1.1 研究背景与意义

近年来国内外的移动互联网开发技术取得了较大的发展,随着 HTML5 与 CSS3 的研究开发,大批 webapp 项目开发成功并且投入市场使用,电商,教育,医疗,美容等行业都先后的推出了自己的 webapp 平台。其中阿里巴巴,京东等电子商城更是 webapp 领域的先行者,给后来者提供了许多的借鉴方式,带动了中国开发 webapp 应用的潮流。

本文以社区服务为题材,研究并完成了智能社区 app,以全新的 HTML5 技术为开发技术,拟为社区家庭开发社区服务 webapp,实现了基于社区生活的在线预约,在线购买等功能。该 webapp 的开发对于社区服务数字化体系的创建具有下列众多重要的意义:

1、传统的社区服务呼叫方式都是采用电话方式联系,特别是针对电话查询来说,查询速度慢,效率低,而且电话号码也十分容易丢失,造成无法预订服务。本文涉及到社区服务 app 实现预约的信息化、电子化,容易联系,维护效率和查询效率高。

2、社区服务 app 可有助于利用网络商城等,丰富社区服务的形式和内容,为用户提供多种形式内容丰富的家政服务。

3、解决了当代年轻人忙于工作或者不会做家务,没有时间照顾老人和孩子的急迫问题。

4、加快发展社区服务,加强基层社区组织建设,优化社区环境,增加了社区居民内心的归属感,幸福感。

### 1.2 国内外研究现状分析

在欧美经济发达国家,由于人均收入与个人可支配经济都很富裕,其商业形态更加围绕个人化,个人在生活周边的消费是极高的,在这些国家,围绕个人周边生活的社区商业差不多占整个国家的 60%以上。而在中国,这个份额还不足 30%。足以见得中国的社区服务在中国的发展还处于落后阶段,具备了极大的发展前景。

从国外社区服务的发展过程和专家学者的研究结论来管中窥豹,社区服务已经开始从纸质名片向电子商务发展。在全世界都推进信息化生活的今天,中国的社区服务平台也在探索新的目标和新的发展方向,他们都希望可以完成社区服务的网络化平台。

近年来,成功运营的社区服务应用很多,比如:叮咚小区 2014 年 3 月上线,2014 年年初叮咚小区成功融资五千万人民币,目前在上海已做大规模的宣传推广。这也说明了社区服务平台得到了越来越多的好评,并且市场需求供小于求。所以本文借助这样的市场情况,打造了这个社区助手 webapp,为社区 O2O 服务增加更多的线下资源,利用线上咨询预订模式,使社区人民的生活变得更方便,更快捷,更效率。

## 第二章 软件开发主要技术介绍

### 2.1 B/S 技术

在浏览器没有普及之前，那时候桌面应用从很早的 DOS 应用程序的开发，一般只能用做数字的计算，到后来的桌面应用开发的，由于当初技术的不成熟，只能开发简单的应用和功能，后来随着数据库技术和桌面应用开发语言的不断普及，C/S 架构的软件也曾经有着不可磨灭的影响。但是，随着 C/S 架构的使用过程，对资源占用和安装的不方便性，对数据管理的实时性要求，与此同时，IE 浏览器的普及以及对于 IE 浏览器的研发，令 B/S 架构的管理系统在使用环境，使用效率，占用存储等众多方面的优势得到迅速的蔓延和推广。

### 2.2 HTML5 技术

HTML，超文本标记语言，是一种为“个人设计的网页可在其他电脑上呈现”设计的语言。它在实际开发的主要用途有两点：一是页面信息的结构化处理，给人带来层次感和美感；二是用来描述文档的外观和语义。

自从有了网页和 IE 浏览器后，人们就开始研发一种创建网站的语言，这个时候 html 语言就应运而生了。这种语言本源就是为了设计适合 IE 浏览器结构的网页，这种网页，起先只能够在电脑上呈现出静态的形式，后来经过人们的不断开发，出现了 HTML6.0，XHTML 等版本，网站的形式也从最先的单一的静态网站变化到现在的多彩多样的动态网站。可以说，就目前为止，在世界电脑上运行最多的语言就是 HTML 了。

一小部分人在 2004 年成立了一个叫做 Web Hypertext Application Working Group（WHATWG，即 Web 超文本应用技术工作组）的组织，他们制定了新的 HTML5 语言规范，并且针对之前最薄弱的 Web 应用的环节开发出新的功能。也就是这个时候，web 2.0 开始出现。当大多数的网站由静态网站逐渐转化成动态网站的时期，它也随之提供着更多的特性。现在 HTML5 所拥有的某些效果已经超越早期标准的效果，其中之一就是动画制作。其主要的性能优点包括：

- （1）SEO 得到优化；
- （2）图片下载更迅速，特点突出在移动用户；
- （3）移动 webapp 容易开发；
- （4）精美的动画特效；

### 2.3 CSS3 技术

层叠样式表也称为字符串样式表。CSS 是 W3C 定义和维护的一种标准，它是将结构化样式添加到结构化文档（如 HTML 文档或 xml 应用程序）中的一种计算机语言。开发人员可以使用 CSS 技术来有效地控制页面布局，字体，颜色，背景和其他效果在网页制作过程中。同时，如果你想改变某个特定页面的不同部分，或者修改一些页面的外观和格式，



用不同的页面，就可以成功实现对相应代码的简单修改。

CSS 在网页建造中最突出的作用就是可以将文件的结构与文件的显示分割开来。这种分隔方式好处有很多：

- (1) 加强文件的可读性；
- (2) 增加文件结构的灵活性；
- (3) 作者和读者可以自己决定文件的显示；
- (4) 文件的结构简单明了。

CSS3 的出现，使得代码变的更简洁，易懂，页面结构也变得更加合理，性能与框架都得到了质的飞跃，而且 CSS3 数据也精简实用，许多以前 CSS2 不能实现的效果，现在利用 CSS3 都能够完成，类似于 CSS2 要用图片做效果，而现在利用 CSS3 编写代码就可以直接实现图片效果。

CSS3 现在已经广泛作用于微信页面开发，WebApp 开发等。

## 2.4 JS 技术

JavaScript 是作用在网页上的开发语言，虽然 JAVASCRIPT 技术跟 java 很类似，但是它只是一个脚本语言，用于特效制作以及一些前台验证和时间选择、文本编辑等使用，网页开发中，人们给静态网页添加动态交互模块时用的最多的就是它，所以，如何使用 javascript 是前台美工必须掌握的语言。同时 Java Script 也被用于诸如服务器等的不同的接口上。Ecma 国际在 Java Script 的基础上制定了 ECMAScript 新标准，使最初的弱类，基于原的特点等从 JS 中淘汰。现在完整的 Java Script 语言的实现概念实际上由三个部分组成：ECMAScript，文档对象模型，以及浏览器对象模型。

JavaScript 脚本语言的基本特点如下：

(1) 解释性：JavaScript 不同于一些编译性的程序语言，例如 C、C++等，它是一种解释性的程序语言，它的源代码不需要经过编译，而是在程序运行过程中被逐行地解释。它与 HTML 标识结合在一起，从而方便用户的使用操作。

(2) 基于对象：JavaScript 是一种基于对象的语言。这意味着它能运用自己已经创建的对象。因此，许多功能可以来自于脚本环境中对象的方法与脚本的相互作用。

(3) 事件驱动：JavaScript 可以直接对用户或客户输入做出响应，无须经过 Web 服务程序。它对用户的响应，是以事件驱动的方式进行的。所谓事件驱动，就是指在主页中执行了某种操作所产生的动作，此动作称为“事件”。比如按下鼠标、移动窗口、选择菜单等都可以视为事件。当事件发生后，可能会引起相应的事件响应。

(4) 跨平台：JavaScript 依赖于浏览器本身，与操作环境无关，只要能运行浏览器的计算机，并支持 JavaScript 的浏览器就可正确执行。

(5) 安全性: JavaScript 是一种安全性语言, 它不允许访问本地的硬盘, 并不能将数据存入到服务器上, 不允许对网络文档进行修改和删除, 只能通过浏览器实现信息浏览或动态交互。这样可有效地防止数据的丢失。

## 2.5 PHP 技术

PHP(Hypertext Preprocessor——超文本预处理器)是一种 HTML 内嵌式的脚本语言。PHP 的语法大部分兼容了 C、Java、perl, 并增加了 PHP 所特有的语法结构, 它简洁易懂, 实用性高, 只需要程序员掌握一点点编程知识就可以创建一个动态的 WEB 网页。PHP 可以用来管理动态网站后台、数据库, 甚至能够建立一个大型的电商网站。它支持许多数据库, 包括 MySQL、Oracle、和 Microsoft SQL Server 等等。

PHP 是开源的编辑语言, 不用花上一分钱, 就可以自由的获得源码, 甚至你能从中加入你自己想要的特色语言, 目前我们可以在 PHP 官方网站下载。Apache 和 MYSQL 也是完全免费开源的。他们之间可以相互使用, 通常快速搭建一套动态网站可以利用 PHP+MYSQL 或者 APACHE+PHP。而且就目前来看, 这种搭配可以完全免费, 且比后者的效率要高所以目前大多数的主机系统都配有 APACHE+PHP+MYSQL 程序。

PHP 具有如下的优点:

1. 学习简单: 使用 PHP 编写程序只需要通过短暂的学习就可以编写了。
2. 数据库连接方便: PHP 现在可以连接大多数的主流数据库。PHP 与 MYSQL 是现在绝佳的组合。可以通过改变连接数据库的代码来适应各种数据库。
3. 扩展性强: PHP 发展至今, 已经进入了一个蓬勃成长的时期, 更多的功能被挖掘出来, 所以它具有许多的发展方向。
4. 可以进行面向对象编程: PHP 提供了基于 WEB 的编程工作所需要的类和对象, 并且 PHP 支持构造器、提取类等。

那么 PHP 是怎样实现动态的功能呢? 还是让我们看看它的执行示意图吧! 如图 2.1 所示:

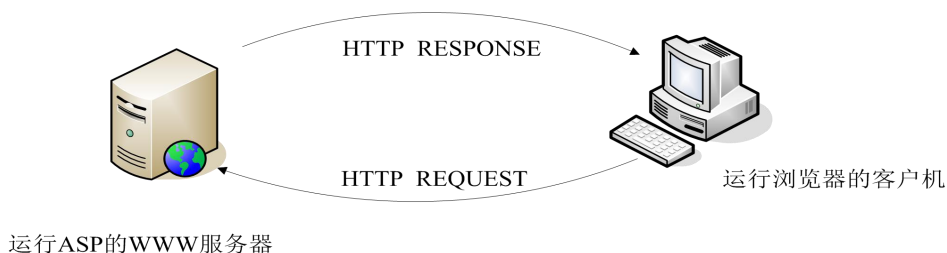


图 2.1 PHP 执行过程简图

当一个客户端浏览器向服务器端请求一个“.php”文件时, 服务器会将这个 PHP 文件仔细的读一遍, 并加以编译执行, 最后送出标准的 HTML 格式文件给客户, 这也就克服了浏览器不兼容的问题。

## 第三章 系统分析

在对该软件进行设计之前，首先要对该软件进行需求分析。在软件工程中，需求分析具有十分重要的意义。在软件开发之初，设计人员需要通过需求分析来明确将要实现的软件的整体和细节功能，一份准确而详实的软件需求分析可以为软件设计提供良好的设计思路和方法。下面将对软件的功能需求分析和用例分析进行描述。

### 3.1 可行性分析

本文的设计的系统将会分别从经济的可行性，技术的可行性以及操作的可行性等方面进行全面的可行性的分析。

#### 1、经济可行性

由于系统开发的工具和软件都是可以免费使用的，因此系统开发的费用很少，甚至只需要少许的费用就可以开发出本系统，也可以忽略不计；再加上通过系统的实施可以提高手机用户文件的私密性，最大程度的保护用户的信息安全，因此，从这些方面可以说经济上是可行的。

#### 2、技术可行性

技术的可行性是适应计算机的软件和硬件的配置的发展在现在拥有的技术条件的基础之上，是否能够满足发展的要求。由于本系统开发基于 PHP+HTML5 技术，在一般的计算机上就能够满足开发本系统所需要的软硬件条，因此技术上也是可行的。

#### 3、操作可行性

作为在校大学生来讲，都是受过良好教育，具有高素质的人才，都能十分容易的使用手机进行各种操作，因此本文设计的利用 PHP+HTML5 技术进行设计的社区服务系统，其操作是完全可行的。

### 3.2 软件功能需求

开发一个系统的关键在于我们要根据以往的研究，并做了充分的需求分析。因为许多先前的理论或实践研究，如果是合理的对于设计者来说就是好的。可采用的方式，如问卷调查或自身体验等，以明确网站或系统的真正需求，通过系统提供的样例确认客户真正需要的东西。做重要的客户系统的关键是可以从需求分析出发，只需要了解用户的兴趣和关注，就可以得出系统的功能设计以及需要什么功能。好的需求分析不仅可加快系统开发的进程，而且也可以提高效率系统开发，为系统早期投入市场，主动为客户提供方便，或增加或改善管理提供好处和便利。

#### 3.2.1 系统用例图

基于 HTML5 的社区服务系统的用例抽象，可以确定如下用例：未注册用户可以实现的是用户注册；注册用户可以实现登录、搜索、加入购物车、付款等，后台管理人员可以实现的是登录、商品管理、订单管理、权限管理、用户管理、文章管理等。

用户注册实现功能用例图：

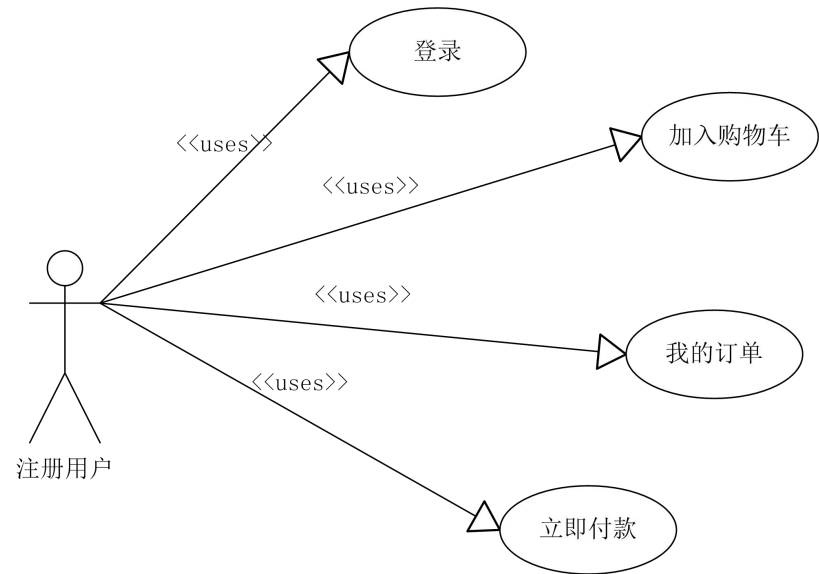


图 3.1 注册用户实现功能用例图

后台管理功能用例图：

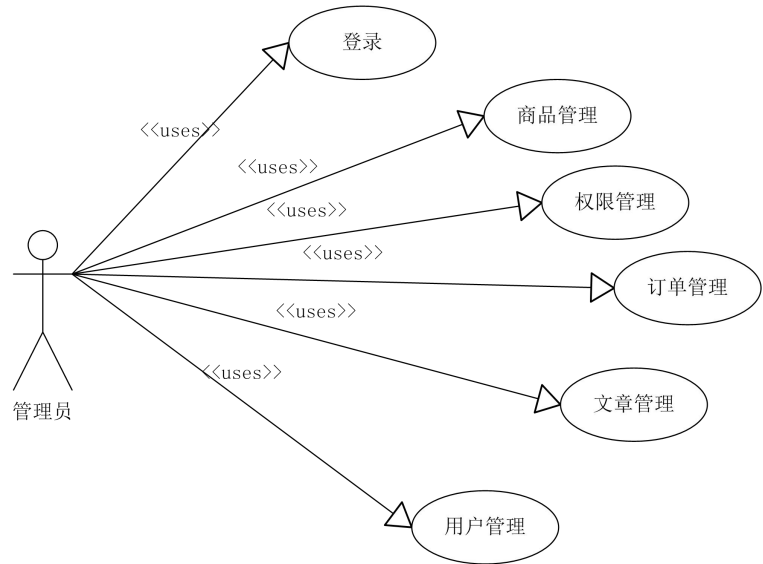


图 3.2 后台管理功能用例图

用户修改信息用例描述：

表 3.1 基本信息修改用例描述

用例名称：基本信息修改
用例简述：当前登录用户编辑修改个人基本信息。
用例参与者：已注册用户
用例事件基本流程： 1. 用户进入“个人资料”界面； 2. 系统以可编辑的方式显示当前登录用户的个人信息，包括：昵称、qq、性别、出生年月、个性签名； 3. 用户向系统发出“修改个人基本信息”的请求； 4. 用户编辑修改自己的个人基本信息，可以修改的信息包括：昵称、qq、性别、出生年月、个性签名； 5. 用户向系统提交修改后的个人基本信息； 6. 系统确认并保存用户提交的信息； 7. 用例结束。

表 3.2 密码修改用例描述

用例名称：密码修改
用例简述：当前登录用户修改自己的登录密码。
用例参与者：已注册用户
用例事件基本流程： 1. 用户向系统发出“修改密码”请求； 2. 系统显示相应界面，要求输入旧密码、新密码、确认新密码； 3. 用户输入旧密码、新密码和确认新密码； 4. 用户向系统提交密码修改信息； 5. 系统确认并保存用户提交的信息； 6. 用例结束

表 3.3 用户头像修改用例描述

用例名称：用户头像修改
用例简述：当前登录用户修改自己的头像图片。
用例参与者：已注册用户
用例事件基本流程： 1. 用户向系统发出“修改头像”请求； 2. 系统显示原有的头像图片； 3. 用户上传新的头像图片； 4. 用户向系统提交已上传的头像图片； 5. 系统确认并保存用户提交的信息； 6. 用例结束

### 3.3 系统性能需求

#### 1、响应时间

一般来说，基于网站开发出来的应用程序响应时间很大程度上受制于计算机的硬件配置和系统版本。因此，用户的计算机设备的硬件配置和系统版本都会对应用程序的响应时间有一定的影响。

#### 2、安全及稳定性

应用程序为互联网应用，用户必须要有权限方可使用系统，因此，可以防止信息泄露的风险。

#### 3、界面清晰友好，使用方便

一个应用程序的成功，程序界面必然要结构合理，突出重点功能，操作方便，HTML5 的出现，为 WEBAPP 的界面优化提供了良好的技术支持。

## 第四章 系统设计

### 4.1 总体设计

本文设计的社区服务系统包括前台展现和后台管理两部分。前台展现实现如下的主要功能：注册与登录、首页、旅游产品分类、购物车、订单以及个人中心等。后台管理实现如下的主要功能：管理员登录、用户管理、商品管理以及订单管理等。并确定每个模块的功能，制出系统组织结构图。

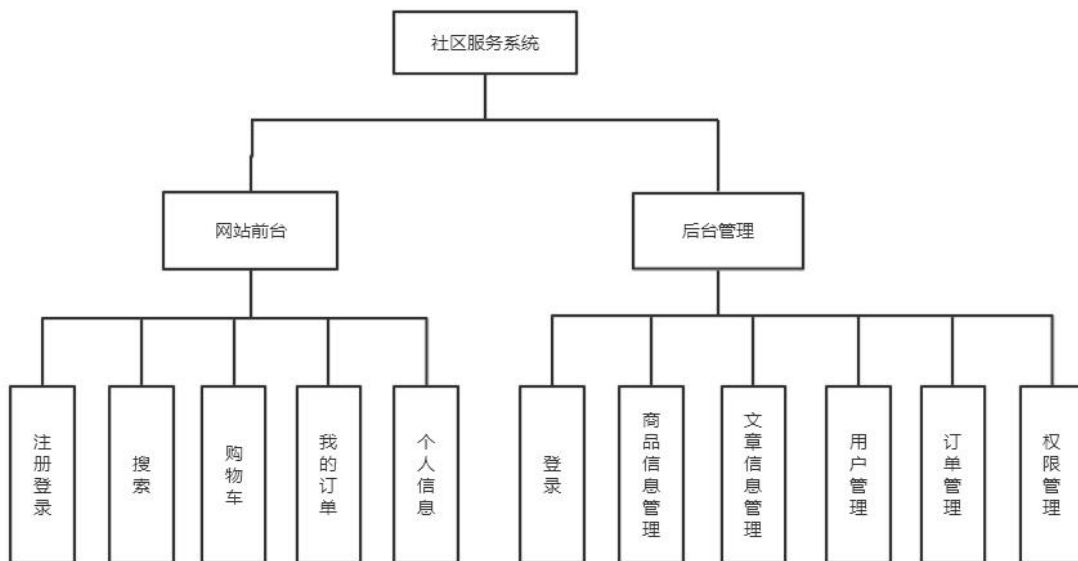


图 4.1 系统组织结构图

#### 4.1.1 系统分层介绍

系统的分层与系统文件结构相对应，分框架入口文件（index.php）、框架公共资源文件目录（Public）、插件包文件目录（plugin）、上传文件目录(upload)、系统核心实现模块目录（protected）。

系统进行应用初始化之前的调用流程如下：

- (1). 加载公共入口文件
- (2). 项目初始化 init
- (3). 项目预编译
- (4). URL 分析 Dispatcher
- (5). 获取模块和操作名
- (6). 项目执行 exec
- (7). 执行控制器的操作
- (8). 调用模型
- (9). 获取数据 find
- (10). 输出视图

项目在加载公共文件再进行项目初始化，之后主要操作是根据 MVC 模型的流程进

行执行的，控制器（Controller）定义了应用程序的行为；并负责对来自视图的用户要求进行解释，并把这些要求映射成相应的行为，这些行为由模型（Model）负责实现；视图（View）用来组织模型的内容。它从模型那里获得数据并指定这些数据如何表现。当模型变化时，视图负责维持数据表现的一致性。视图同时将用户要求告知控制器（Controller）。

#### 4.1.2 数据库设计

表 4.1 shope\_user 表字段属性

字段	字段名称	数据类型	主键	允许为空	备注
user_id	用户的主键	int	是	否	唯一
username	名字	varchar	否	是	
password	密码	varchar	否	是	
email	邮箱	varchar	否	否	
mobile	手机号	int	否	否	
name	昵称	varchar	否	是	
sex	性别	varchar	否	是	
birthdate	生日	date	否	是	
avatar	头像	varchar	否	否	

表 4.2 shope\_goods 表字段属性

字段	字段名称	数据类型	主键	允许为空	备注
goods_id	产品主键	int	是	否	唯一
cate_id	产品分类	varchar	否	是	
brand_id	品牌	varchar	否	是	
goods_name	产品名称	varchar	否	是	
now_price	当前价格	decimal	否	是	
original_price	原先价格	decimal	否	是	
goods_image	产品主图	varchar	否	是	
goods_content	产品简介	varchar	否	是	
created_date	创建时间	date	否	是	
status	状态	varchar	否	是	

表 4.3 shope\_goods\_cate 表字段属性

字段	字段名称	数据类型	主键	允许为空	备注
cate_id	产品分类的主键	int	是	否	唯一
parent_id	产品分类上级 ID	int	否	是	
cate_name	产品分类名称	varchar	否	是	
seq	首页状态	varchar	否	是	

表 4.4 shope\_goods\_album 表字段属性

字段	字段名称	数据类型	主键	允许为空	是备注
id	主键	int	是	否	唯一
goods_id	产品 ID	varchar	否	是	
image	图片路径	varchar	否	是	

表 4.5 shope\_feedback\_message 表字段属性

字段	字段名称	数据类型	主键	允许为空	备注
----	------	------	----	------	----

id	评论的主键	int	是	否	唯一
name	名称	int	否	是	
good_id	产品 id	int	否	是	
user_id	用户 id	int	否	否	外键
content	内容	text	否	否	
time	时间	varchar	否	是	
fid	父 id	int	否	是	外键

表 4.6 shope\_order 表字段属性

字段	字段名称	数据类型	主键	允许为空	备注
id	订单的主键	varchar	是	否	唯一
order_id	订单编号	varchar	否	否	
user_id	用户 id	int	否	是	外键
shipping_method	购买方式	varchar	否	是	
price	价格	decimal	否	否	
goods_amount	数量	int	否	否	
payment_method	支付方式	text	否	是	外键
created_date	生成时间	varchar	否	是	
order_status	状态	int	否	是	
payment_date	支付日期	int	否	是	

表 4.7 shope\_order\_log 表字段属性

字段	字段名称	数据类型	主键	允许为空	备注
id	消费的主键	int	是	否	唯一
time	时间	varchar	否	是	
content	内容	text	否	是	
money	金钱	decimal	否	否	
user_id	用户 id	int	否	否	外键
shopid	消费 id	text	否	是	
is_re	购买方式	int	否	是	
order_id	订单 id	int	否	是	外键

表 4.8 shope\_order\_goods 表字段属性

字段	字段名称	数据类型	主键	允许为空	备注
goodname	商品名称	varchar	否	否	
goodpicture	商品图片	varchar	否	否	
price	价格	text	否	否	
number	数量	decimal	否	否	
total	小计	int	否	否	

## 4.2 详细设计

### 4.2.1 用户中心模块设计

用户模块需要的功能有：用户注册功能、用户登录功能、用户信息修改功能。

用户注册功能，要求用户填写相应的信息，有名字、邮箱、密码等相应的属性字段。

用户登录功能，要求用户正确填写名字和登录密码。

用户信息修改功能，用户可修改的属性字段有邮箱、手机、昵称、性别、生日、用户头像等。



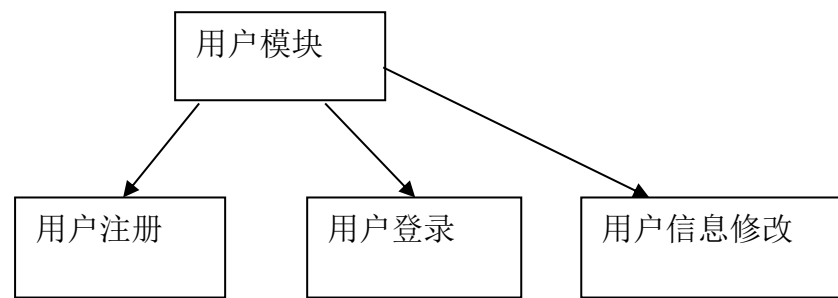


图 4.4 用户中心模块

4.2.2 社区服务商品模块设计

社区服务商品模块设计需要的功能有：搜索服务商品信息、订单功能、购物车功能、评论产品。

搜索服务商品信息，可以根据服务商品分类来搜索商品。

订单功能需要包括的字段属性有：收件人信息、配送方式、商品名称、商品价格、数量、金额。

购物车功能需要包括的字段属性有商品名称、商品图片、价格、数量、金额总计。

评论产品功能需要包括的字段属性有，评论人、评论时间、评论内容。

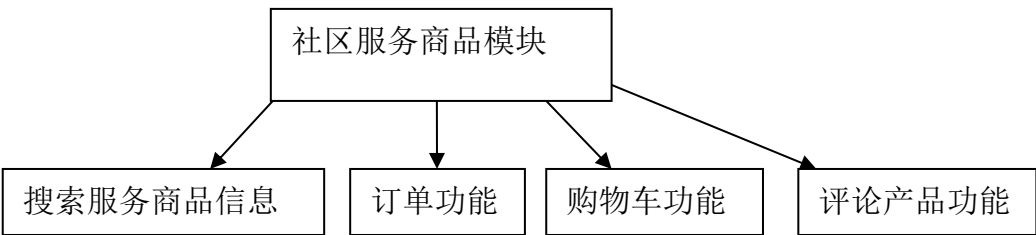


图 4.5 社区服务商品模块

4.2.3 后台管理模块设计

后台模块主要负责前台模块的维护，用户信息管理、社区服务产品管理、订单管理、公告或资讯管理、系统管理。

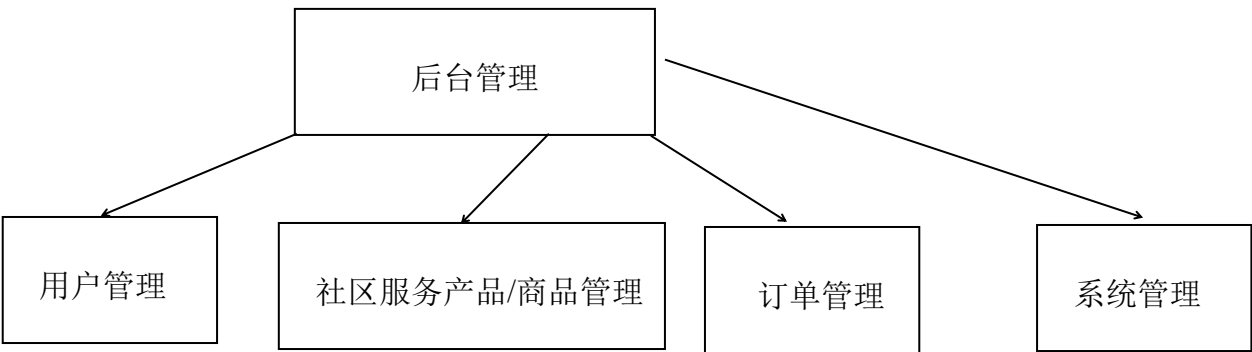


图 4.6 后台管理模块

## 第五章 系统实现与测试

### 5.1 系统实现

#### 5.1.1 用户中心模块的实现

在系统中的前端入口，类的属性有我的收藏、购买记录、我的购物车、我的订单、基本信息修改，具体如下：

用户注册实现：

```
public function action_register()
{
    if(request('step') == 'submit')
    {
        $data = array
        (
            'username' => trim(request('username', '', 'post')),
            'password' => trim(request('password', '', 'post')),
            'repassword' => trim(request('repassword', '', 'post')),
            'email' => trim(request('email', '', 'post')),
            'captcha' => strtolower(trim(request('captcha', ''))),
        );

        $user_model = new user_model();
        $verifier = $user_model->verifier($data, array('mobile' => FALSE));
        if(TRUE === $verifier)
        {
            if($user_model->register($data)) $this->prompt('success', '恭喜您，注册成功！请您务必牢记您的用户名和邮箱.', url('user', 'index'));
            $this->prompt('error', '注册失败！请稍后重新尝试.');
```

效果如图所示：

注册新用户

已注册了账号? 请直接 [登陆](#)

用户名:

test02

用户名可以包含字母、数字或下划线, 须以字母开头, 长度为5~16个字符

邮箱:

@

请填写您常用的电子邮箱地址, 邮箱可用来重置密码, 接收订促销信息, 订单状态等

设置密码:


密码可包含字母、数字或特殊符号, 长度为6~32个字符


确认密码:

\*\*\*\*\*

确认您所设置的密码

验证码:

 点击刷新



☒ 已阅读完并同意 "用户注册协议"

立即注册

图 5.1 注册页面

用户登录实现:

```
public function action_login()
{
    $client_ip = get_ip();
    if(request('step') == 'submit')
    {
        $error_model = new request_error_model();
        if($is_captcha = $error_model->check($client_ip,
$GLOBALS['cfg']['captcha_user_login']))
        {
            $captcha = strtolower(trim(request('captcha', '', 'post')));
            if(empty($_SESSION['CAPTCHA']) || $_SESSION['CAPTCHA'] !=
$captcha)
            {
                unset($_SESSION['CAPTCHA']);
                $this->prompt('error', '验证码不正确', url('user', 'login'), 5);
            }
        }

        $username = trim(request('username', '', 'post'));
```

```

$password = request('password', '', 'post');

$user_model = new user_model();
if($user = $user_model->find(array('username' => $username, 'password'
=> $password)))
{
    if(request('stay'))        $user_model->stay_login($user['user_id'],
$user['password'], $client_ip);
        $user_model->set_logged_info($client_ip,        $user['user_id'],
$user['username'], $user['avatar']);
        $redirect        =        isset($_SESSION['REDIRECT'])        ?
$_SESSION['REDIRECT'] : url('user', 'index');
        unset($_SESSION['REDIRECT']);
        $this->prompt('success', '登录成功', $redirect, 3);
    }
else
{
    $error_model->incr_err($client_ip);
    $this->prompt('error', '用户名或密码错误', url('user', 'login'), 5);
}
}
else
{
    if($cookie = request('USER_STAYED', null, 'cookie'))
    {
        $user_model = new user_model();
        if($user_model->check_stayed($cookie, $client_ip)) jump(url('user',
'index'));
    }
    $error_model = new request_error_model();
    $this->captcha        =        $error_model->check($client_ip,
$GLOBALS['cfg']['captcha_user_login']);
    $oauth_model = new oauth_model();
    $this->oauth_list = $oauth_model->get_enable_list();
    $this->compiler('login.html');

```

```
    }  
}  
效果如图所示:
```



图 5.2 登录页面

5.1.2 订单功能的实现

订单模块是本系统设计最核心的模块之一，也是最总要的模块之一，用户通过注册成会员之后，在平台预订或购买社区服务商品之后，会生成相应的订单信息，其主要实现过程如下：

```
public function action_freight()  
{  
    $user_id = $this->is_loggedin();  
    //购物车信息  
    $cart = json_decode(stripslashes(request('CARTS', null, 'cookie')), TRUE);  
    if(!$cart) die(json_encode(array('status' => 'error', 'msg' => '无法获取购物车数  
据')));  
    $goods_model = new goods_model();  
    if(!$cart = $goods_model->get_cart_items($cart)) die(json_encode(array('status'  
=> 'error', 'msg' => '购物车商品数据不正确')));  
    //收件人信息  
    $csn_id = (int)request('csn_id', 0);  
    $consignee_model = new user_consignee_model();  
    if(!$consignee = $consignee_model->find(array('id' => $csn_id, 'user_id' =>  
$user_id)))  
    {  
        die(json_encode(array('status' => 'error', 'msg' => '收件人地址不存在')));  
    }  
}
```

```

    }
    //计算运费
    $shipping_id = (int)request('shipping_id', 0);
    $shipping_model = new shipping_method_model();
    $amount = $shipping_model->check_freight($user_id, $shipping_id,
    $consignee['province'], $cart);
    if(FALSE === $amount) die(json_encode(array('status' => 'error', 'msg' => '计
    算运费失败')));

    echo json_encode(array('status' => 'success', 'amount' => sprintf('%.2f',
    $amount)));
    }

    public function action_submit()
    {
        $user_id = $this->is_logged_in();
        //检查购物车信息
        $cart = json_decode(stripslashes(request('CARTS', null, 'cookie')), TRUE);
        if(!$cart) die(json_encode(array('status' => 'error', 'msg' => '无法获取购物车数
        据')));

        $goods_model = new goods_model();
        if(!$cart = $goods_model->get_cart_items($cart)) die(json_encode(array('status'
        => 'error', 'msg' => '购物车商品数据不正确')));
        //检查收件人信息
        $csn_id = (int)request('csn_id', 0);
        $consignee_model = new user_consignee_model();
        if(!$consignee = $consignee_model->find(array('id' => $csn_id, 'user_id' =>
        $user_id))) die(json_encode(array('status' => 'error', 'msg' => '无法获取收件人地址信息')));
        //检查配送方式
        $shipping_id = (int)request('shipping_id', 0);
        $shipping_map = vcache::instance()->shipping_method_model('indexed_list');
        if(!isset($shipping_map[$shipping_id])) die(json_encode(array('status' => 'error',
        'msg' => '配送方式不存在')));
        //检查运费
        $shipping_model = new shipping_method_model();
        $shipping_amount = $shipping_model->check_freight($user_id, $shipping_id,
        $consignee['province'], $cart);
        if(FALSE === $shipping_amount) die(json_encode(array('status' => 'error',
        'msg' => '无法获取运费')));
        //检查付款方式

```

```
$payment_id = (int)request('payment_id', 0);
$payment_map = vcache::instance()->payment_method_model('indexed_list');
if(!isset($payment_map[$payment_id]))
{
    $payment_id = current($payment_map);
    $payment_id = $payment_id['id'];
}
//创建订单
$order_model = new order_model();
$data = array
(
    'order_id' => $order_model->create_order_id(),
    'user_id' => $user_id,
    'shipping_method' => $shipping_id,
    'payment_method' => $payment_id,
    'goods_amount' => $cart['amount'],
    'shipping_amount' => $shipping_amount,
    'order_amount' => $cart['amount'] + $shipping_amount,
    'memos' => trim(strip_tags(request('memos', "))),
    'created_date' => $_SERVER['REQUEST_TIME'],
    'order_status' => 1,
);

if($order_model->create($data))
{
    $order_goods_model = new order_goods_model();
    $order_goods_model->add_records($data['order_id'], $cart['items']);
    $order_consignee_model = new order_consignee_model();
    $order_consignee_model->add_records($data['order_id'], $consignee);
    setcookie('CARTS', null, $_SERVER['REQUEST_TIME'] - 3600, '/');
    $res = array('status' => 'success', 'order_id' => (string)$data['order_id']);
}
else
{
    $res = array('status' => 'error', 'msg' => '创建订单失败, 请稍后重试');
}
echo json_encode($res);
}
```

效果如图所示:

<

确认订单

收件人

test 15800000000  
北京市 市辖区 西城区 test  
111111

>

配送方式

普通快递

▼

包裹清单

育儿嫂

¥888.00 × 1

留言备注

填写您对订单需要备注的内容，没有请留空

订单合计

商品合计:

¥888.00

运费:

¥0.00

共计:

¥888.00

提交订单

图 5.3 订单页面

5.1.3 购物车功能的实现

社区服务的购物车、是注册的会员购买了商品之后，就直接放到购物车里面，在购物车里面可以增加，修改，删除商品信息。  
实现过程如下：

```
class cart_controller extends general_controller
{
    public function action_list()
    {
        $cookie = request('CARTS', null, 'cookie');
        if($cookie)
        {
            $cookie = json_decode(stripslashes($cookie), TRUE);
            $goods_model = new goods_model();
            if($cart = $goods_model->get_cart_items($cookie))
            {
                $res = array('status' => 'success', 'cart' => $cart);
            }
            else
            {
                $res = array('status' => 'error');
            }
        }
        else
```



```

    {
        $res = array('status' => 'nodata');
    }
    echo json_encode($res);
}
}

```

效果如图所示：



图 5.4 购物车页面

#### 5.1.4 后台功能的实现

##### 1) 社区服务商品管理

后台对社区服务商品/产品的管理，查询服务商品/产品的信息，实现代码如下：

```

public function action_index()
{
    if(request('step') == 'search')
    {
        $cate_id = (int)request('cate_id', 0);
        $brand_id = (int)request('brand_id', 0);
        $sign_id = request('sign_id', "");
        $status = request('status', "");
        $kw = request('kw', "");

        $where = 'WHERE 1';
        $binds = array();
    }
}

```

```

        if(!empty($cate_id))
        {
            $where .= ' AND cate_id = :cate_id';
            $binds[':cate_id'] = $cate_id;
        }

        if(!empty($brand_id))
        {
            $where .= ' AND brand_id = :brand_id';
            $binds[':brand_id'] = $brand_id;
        }

        $sign_map = array('newarrival', 'recommend', 'bargain');
        if(isset($sign_map[$sign_id])) $where .= " AND {$sign_map[$sign_id]} = 1";

        if($status != "")
        {
            $where .= ' AND status = :status';
            $binds[':status'] = $status;
        }

        if($kw != "")
        {
            $where .= ' AND goods_name LIKE :kw';
            $binds[':kw'] = '%'.$kw.'%';
        }

        $goods_model = new goods_model();
        $total = $goods_model->query("SELECT COUNT(*) as count FROM
{$goods_model->table_name} {$where}", $binds);
        if($total[0]['count'] > 0)
        {
            $fields = 'goods_id, goods_name, goods_sn, now_price, stock_qty,
created_date, newarrival, recommend, bargain, status';
            $limit = $goods_model->set_limit(array(request('page', 1),
request('pernum', 10)), $total[0]['count']);

            $sort_id = request('sort_id', 0, 'post');
            $sort_map = array('goods_id DESC', 'created_date DESC', 'created_date
ASC', 'now_price DESC', 'now_price ASC');
            $sort = isset($sort_map[$sort_id])? $sort_map[$sort_id] : $sort_map[0];

```

```
$sql = "SELECT {$fields} FROM {$goods_model->table_name} {$where}
ORDER BY {$sort} {$limit}";
```

```

    $results = array
    (
        'status' => 'success',
        'list' => $goods_model->query($sql, $binds),
        'paging' => $goods_model->page,
    );
}
else
{
    $results = array('status' => 'nodata');
}

echo json_encode($results);
}
else
{
    $cate_model = new goods_cate_model();
    $this->cate_list = $cate_model->indexed_cate_tree();
    $brand_model = new brand_model();
    $this->brand_list = $brand_model->indexed_list();
    $this->compiler('goods/goods_list.html');
}
}
```

对应的效果图如下图：

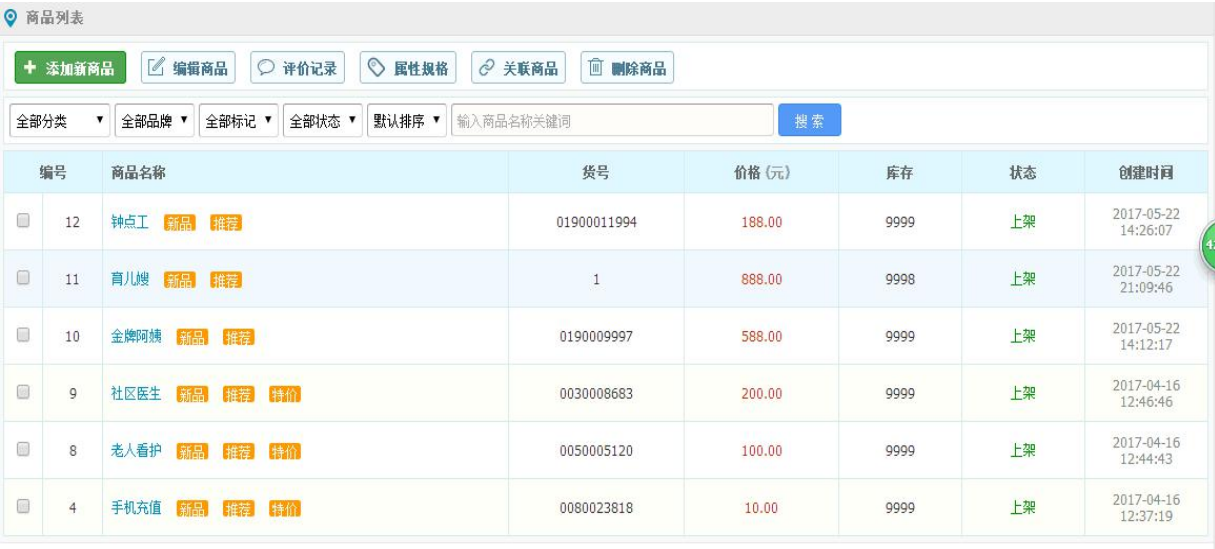


图 5.5 商品管理页面

2) 订单管理模块

订单模块的后台管理主要负责社区服务商品的订单管理、对订单进行审核，发货，实现代码如下：

```
public function action_index()
{
    if(request('step') == 'search')
    {
        $where = 'WHERE 1';
        $binds = array();

        $user_id = (int)request('user_id', 0);
        if(!empty($user_id))
        {
            $where .= ' AND a.user_id = :user_id';
            $binds[':user_id'] = $user_id;
        }

        $order_status = request('order_status', '');
        if($order_status != "")
        {
            $where .= ' AND a.order_status = :order_status';
            $binds[':order_status'] = $order_status;
        }

        $start_date = request('start_date', '');
        if($start_date != "")
        {
            $where .= ' AND a.created_date >= :start_date';
            $binds[':start_date'] = strtotime($start_date);
        }

        $end_date = request('end_date', '');
        if($end_date != "")
        {
            $where .= ' AND a.created_date <= :end_date';
            $binds[':end_date'] = strtotime($end_date);
        }

        $order_id = request('order_id', '');
        if($order_id != "")
        {
            $where .= ' AND a.order_id = :order_id';
```

```

        $binds[':order_id'] = $order_id;
    }

    $order_model = new order_model();
    $total = $order_model->query("SELECT COUNT(*) as count FROM
{$order_model->table_name} AS a {$where}", $binds);
    if($total[0]['count'] > 0)
    {
        $limit = $order_model->set_limit(array(request('page', 1), request('pernum',
10)), $total[0]['count']);

        $sql = "SELECT a.id, a.order_id, a.order_status, a.order_amount,
a.created_date,
                                b.receiver, b.province, b.city, b.borough, b.address, b.zip,
b.mobile
                                FROM {$order_model->table_name} AS a
                                INNER JOIN
{$GLOBALS['mysql']['MYSQL_DB_TABLE_PRE']}order_consignee AS b
                                ON a.order_id = b.order_id
                                {$where}
                                ORDER BY id DESC {$limit}
                                ";

        $list = $order_model->query($sql, $binds);
        $status_map = $order_model->status_map;
        foreach($list as &$v)
        {
            $v['order_status'] = $status_map[$v['order_status']];
            $v['created_date'] = date('Y-m-d H:i:s', $v['created_date']);
        }

        $results = array
        (
            'status' => 'success',
            'list' => $list,
            'paging' => $order_model->page,
        );
    }
    else
    {
        $results = array('status' => 'nodata');
    }
}

```

```
        echo json_encode($results);
    }
    else
    {
        $order_model = new order_model();
        $this->status_map = $order_model->status_map;
        $this->compiler('order/order_list.html');
    }
}
```

效果图如下：



图 5.6 订单管理页面

查看订单详细的代码实现如下：

```
public function action_view()
{
    $condition = array('order_id' => request('id', ''));
    $order_model = new order_model();
    if($order = $order_model->find($condition))
    {
        $vcache = vcache::instance();
        $payment_map = $vcache->payment_method_model('indexed_list');
        $shipping_map = $vcache->shipping_method_model('indexed_list');
        $order['payment_method_name'] = $payment_map[$order['payment_method']]['name'];
        $order['shipping_method_name'] = $shipping_map[$order['shipping_method']]['name'];

        $consignee_model = new order_consignee_model();
        $this->consignee = $consignee_model->find($condition);
        $this->order = $order;
        $this->status_map = $order_model->status_map;
        //用户信息
        $user_model = new user_model();
```

```

        $this->user = $user_model->find(array('user_id' => $order['user_id']));
        //商品列表
        $order_goods_model = new order_goods_model();
        $this->goods_list = $order_goods_model->get_goods_list($condition['order_id']);
        //发货列表
        $this->carrier_list = $vcache->shipping_carrier_model('indexed_list');
        $shipping_model = new order_shipping_model();
        $shipped_list = $shipping_model->find_all($condition, 'dateline DESC');
        if(!empty($shipped_list))
        {
            foreach($shipped_list as $v)
            {
                $this->carrier_list[$v['carrier_id']]['name'] = $v['carrier'];
            }
        }
        $this->shipped_list = $shipped_list;
        //日志列表
        $log_model = new order_log_model();
        if($this->log_list = $log_model->find_all($condition, 'dateline DESC'))
        {
            $this->admin_list = $vcache->admin_model('indexed_list');
            $this->operate_map = $log_model->operate_map;
        }

        $this->compiler('order/order.html');
    }
    else
    {
        $this->prompt('error', '无法找到相应的订单记录');
    }
}

```

效果如下图:



图 5.7 订单后台页面

3) 系统管理

系统管理主要包含管理员管理、角色管理等。

管理员新增的实现如下：

```
public function action_index()
{
    $admin_model = new admin_model();
    $this->results = $admin_model->find_all(null, 'user_id DESC', '*',
array(request('page', 1), 15));
    $this->paging = $admin_model->page;
    $this->compiler('admin/admin_list.html');
}

public function action_add()
{
    if(request('step') == 'submit')
    {
        $data = array
        (
            'username' => trim(request('username', '')),
            'password' => trim(request('password', '')),
            'repassword' => trim(request('repassword', '')),
            'email' => trim(request('email', '')),
            'name' => trim(request('name', '')),
            'created_date' => $_SERVER['REQUEST_TIME'],
        );
    }
}
```



```
$admin_model = new admin_model();
$verifier = $admin_model->verifier($data);
if(TRUE === $verifier)
{
    $data['password'] = md5e($data['password']);
    $data['hash'] = sha1(uniqid(rand(), TRUE));
    unset($data['repassword']);
    if($user_id = $admin_model->create($data))
    {
        $role_ids = request('role_ids', null, 'post');
        if(!empty($role_ids) && is_array($role_ids))
        {
            $admin_role_model = new admin_role_model();
            foreach($role_ids as $v)
            $admin_role_model->create(array('user_id' => $user_id, 'role_id' => $v));
        }
        $this->clear_cache();
    }
    $this->prompt('success', '添加管理员成功', url($this->MOD.'/admin',
'index'));
}
else
{
    $this->prompt('error', $verifier);
}
}
else
{
    $role_model = new role_model();
    $this->role_list = $role_model->find_all(null, null, 'role_id, role_name');
    $this->compiler('admin/admin.html');
}
}
```

效果图：

管理员列表

+ 添加新管理员

 编辑

 删除

编号	用户名	电子邮箱	姓名称呼	上次登录日期	上次登录IP	创建日期	
	2	admin0	1157354193@qq.com	丁文成	2017-05-22	::1	2017-05-22
	1	admin	admin@qq.com	未设置	2017-05-22	::1	2017-03-25

图 5.8 管理员管理页面

管理员的编辑实现如下：

```
public function action_edit()
{
    if(request('step') == 'submit')
    {
        $user_id = request('id');
        $condition = array('user_id' => $user_id);
        $data = array
        (
            'username' => trim(request('username', '')),
            'password' => trim(request('password', '')),
            'repassword' => trim(request('repassword', '')),
            'email' => trim(request('email', '')),
            'name' => trim(request('name', '')),
        );

        $rule_slices = array();
        $admin_model = new admin_model();
        $user = $admin_model->find($condition);
        if($user['username'] == $data['username'])
        {
            $rule_slices['username'] = FALSE;
            unset($data['username']);
        }
        if(empty($data['password']) && empty($data['repassword']))
        {
            $rule_slices['password'] = $rule_slices['repassword'] = FALSE;
            unset($data['password']);
        }

        $verifier = $admin_model->verifier($data, $rule_slices);
        if(TRUE === $verifier)
        {
            if(isset($data['password'])) $data['password'] =
md5e($data['password']);
            unset($data['repassword']);
            $admin_model->update($condition, $data);

            $admin_role_model = new admin_role_model();
            $admin_role_model->delete($condition);
            $role_ids = (array)request('role_ids');
            if(!empty($role_ids))
            {
```

```

        $admin_role_model = new admin_role_model();
        foreach($role_ids as $v)
$admin_role_model->create(array('user_id' => $user_id, 'role_id' => (int)$v));
        }

        $this->clear_cache();
        $this->prompt('success', '更新管理员成功', url($this->MOD.'/admin',
'index'));
    }
    else
    {
        $this->prompt('error', $verifier);
    }
}
else
{
    $user_id = (int)request('id', 0);
    $condition = array('user_id' => $user_id);
    $admin_model = new admin_model();
    if($rs = $admin_model->find($condition))
    {
        $admin_role_model = new admin_role_model();
        $rs['role_ids'] = $admin_role_model->find_all($condition, null,
'role_id');

        if(!empty($rs['role_ids']))
        {
            $rs['role_ids'] = array_column($rs['role_ids'], 'role_id');
        }
        $this->rs = $rs;
        $role_model = new role_model();
        $this->role_list = $role_model->find_all(null, null, 'role_id,
role_name');

        $this->compiler('admin/admin.html');
    }
    else
    {
        $this->prompt('error', '未找到相应的数据记录');
    }
}
}
}

```

效果图如下：

 编辑管理员: [1]

用户名	<div>admin</div> <div>可以包含字母、数字或下划线，须以字母开头，长度为5-16个字符</div>
重设密码	<div><div>点击重新设置密码</div><div>如需重设密码请点击以上按钮，否则密码保持不变</div></div>
姓名称呼	<div>丁文成</div>
电子邮箱	<div>admin@qq.com</div>
上次登录日期	<div>2017-05-22 08:49:18</div>
上次登录IP	<div>::1</div>
创建日期	<div>2017-03-25 07:18:00</div>

保存并更新

重置表单

图 5.9 管理员信息管理页面

5.2 系统测试

5.2.1 测试概述

软件测试是检测程序是否存在错误的过程。规格测试是基于研究和发展的过程中,每段时期和程序结构 and 设计了一些情况下(即,信息和预计的结果),用这部分的情况下同时运行,发现程序错误在过程按照一定的顺序一个接一个的项目。在测试过程中应遵循以下原则:

1. 及时对软件测试。

不可以说软件测试单单是当做一个完全单独的模块,并需要将其应用于软件开发的整个阶段。

2. 测试用例必须有工作人员的输入和猜想输出这两个方面。

在测试之前应根据测试需要选择测试用例(测试用例),为了学院项目,所以不仅需要测试输入数据,还需要输入数据的预期输出。

3. 开发人员不要检测本人的代码。

开发人员应该尽可能不要测试本人的编写代码,应该独立创建相关的研究和开发人员,以避免测试自己的代码。如果条件允许的话,最好可以开发一个新的测试团队。这样的话他们之间不会相互干扰。

4. 设计测试用例时,输入是否合理的条件我们都应该去测试它。

合理的条件意味着可以测试程序,在正常的数据输入的情况下,不合理的输入条件意味着输入条件是错误的,可能会导致问题的变化,所以我们需要测试。在测试时,系统必须验证软件非法处理命令的功能。测试程序中,通常我们都输入不合理的输入条件,而不是测试合理的输入条件,这样可以获得更多的错误来修改并完善我们的程序。

5. 足够的注意力测试集群现象。

在需要检查的程序当中,如果我们已经发现了大量的错误 bug,那有可能还有更多的错误是我们目前没有发现的,许多测试已经证明错误集群性的存在。根据这一规则,应重点关注测试错误聚类过程部分,提高投资的效率测试。

6. 严格执行测试方案,避免一些随机性测试结果。

测试需要有测试方案,方可对系统进行全面而有计划性的测试,如果随机测试,有些问题不太容易查到,所以需要制定测试计划,来完成系统的测试。

软件测试的目的一是为了发现错误,二是为了提高系统可用性、可靠性等。例如,我们测试的目的是发现尽可能多的错误,测试将需要直接的一部分程序相对复杂,位置在过去经常出现异常。如测试的目的是为了给广大用户提供一个高质量的测试评估,测试应该在实际应用中面临经常使用的假设。

软件测试是为了发现程序中的 bug 而执行的过程。一个优秀的用例测试可以一直找到没有 bug 生成为止。

测试不是仅仅发现 BUG 这一点这么简单,而是为了需要寻找问题的根源。但通过分析一些错误的原因和错误的分布特征,可以帮助项目的用户,让他们找出当前使用的应用软件的一部分。

如果找不到错误,这样的测试不是无用的,一个好的测试程序测试质量评估的一种方法。

5.2.2 模块单元测试

本系统的测试主要采用黑盒测试方法,比如用户登录的测试用例如下:

表 5.1 用户登录方法测试用例表

用例编号	Test-001		
方法测试	用户登录方法 Login() 测试		
用例目的	测试用户登录方法时候正确运行		
前提条件	已有相应的方法参数 userName 和 password		
输入/动作	数据	预期	实际结果
输入正确的参数类型	输入正确的方法参数 userName:admin;password:admin	后台打印输出字符串“success”	后台打印输出字符串“success”
	输入正确的参数 userName: admin 错误的参数 password: a	后台打印输出字符串“密码错误”	后台打印输出字符串“密码错误”
	输入错误的用户名 userName: a 正确的密码 password: admin	后台打印输出字符串“用户名错误”	后台打印输出字符串“用户名错误”

登录截图



图 5.10 登录测试页面

表 5.2 评论内容列表方法测试用例表

用例编号	Test-002		
方法测试	评论内容列表方法测试		
用例目的	测试评论内容列表方法是否输出评论列表内容		
前提条件	数据库有相应数据		
输入/动作	数据	预期	实际结果
	输入第 1 条评论的主键 Id	后台打印输出第一条评论的内容	后台打印输出第一条评论的内容
	输入第 2 条评论的主键 Id	后台打印输出第二条评论的内容	后台打印输出第二条评论的内容
	输入第 3 条评论的主键 Id	后台打印输出第三条评论的内容	后台打印输出第三条评论的内容

图 5.11 评论测试页面



## 结 论

经过这个项目的磨练，我觉得要做好一个程序，最为重要的关键点就是要做好系统分析，我们开发整个程序开发之初，就应该明确好程序的功能需求和数据库，不然我们会在开发的过程中遇见不断的意外，然后不断的修改出错的问题，这也会给整个的开发周期带来巨大的麻烦，可能到最后也不能达到我们预期的效果。所幸的是老师明确的说明了他会在我们做完程序之后按照我们所做的程序来书写毕业设计任务书，大大的减少了我们的开发任务与周期，使得我们的设计可以按时按我们预期的完成。

本文设计的档案助手首先阐述了网站开发的背景和国内外的研究现状及存在的问题；其次以社区服务开发建设为主线，介绍开发网站的相关技术、对网站进行需求分析、总体设计以及重要功能模块设计的实现，本网站采用 B/S 三层结构，并在 HTML5、CSS3、JS 以及 PHP 技术开发下完成的社区服务 webapp，实现用户注册登录，购买物品，修改个人信息等功能。在完成这些功能的同时，本文主要解决了以下几个方面的关键问题：采用 HTML5 技术来设计，可实现在手机端进行使用，极大方便了移动端的使用。

## 致 谢

大学的学习生活总是感觉短暂的，总感到知识还没有学够，但大学毕业就在我们面前，即毕业设计即将的结束，意味着我们的大学生涯即将结束。毕业设计是我们在大学所学知识的一次总结，更是对我们过去所学知识的提炼和升华，通过毕业设计，我们认识到大学学习过程中还有那些知识没有掌握，那些知识需要我们进入深入研究。这次毕业设计让我学到了很多新的知识，研究了一些新的以前自己不熟悉的领域，使我处理问题的能力得到了一定的提升，同时也为今后的继续深造或踏入社会大门提供了很好的锻炼机会。

这次程序设计能够顺利完成，首先我要衷心感谢我的老师对我的帮助和支持，老师对我设计过程中的帮助和指导使我受益颇多，同时做设计过程中也得到许多同学们的帮助，在此同样表示由衷的感谢！因为我们的知识是有限的，所以本文将不可避免的会出现一些错误和不足之处，还请老师指正。



## 参考文献

- [1]Thomas A.Powell 编著 . Web 设计大全[M]. 机械工业出版社, 2001.
- [2]郝悍勇, 孙久文, 杜建成编著 . 疯狂站长之 PHP[M]. 中国水利水电出版社 , 2005.
- [3]施红芹, 谢军, 彭海明 编著 . 动态网页程序设计[M]. 清华大学出版社 , 2002.
- [4]杨文龙, 姚淑珍, 吴云 编著 . 软件工程[M]. 电子工业出版社, 2003.
- [5]李雁翎, 陈光 编著 . Mysql 基础与应用[M]. 清华大学出版社 , 2001.
- [6]宣小平, 但正刚, 张文毅编著. PHP 数据库系统开发实例导航[M]. 人民邮电出版社 , 2002.
- [7]Valduriez P.Parallel Database Systems:Open Problems and New Issues.Distributed and Parallel Databases[J]. Vol. 1, No. 2, 1993.
- [8]Li jianzhong, Srivastava, Rotem D. CMD:A Multi-dimensional Declustering Method for Parallel Database Systems[J]. In Proceedings of VLDB, 1992.
- [9]宋彦浩编著. PHP 建网技术[M]. 中国水利水电出版社, 2003.
- [10]Douglas E.Comer 编著 . Internet 技术基础 (英文版) [M]. 机械工业出版社 . 2002.
- [11]向柳波, 杨彬. 浅谈 HTML5 在电子地图中的应用[J]. 江西理工大学学报. 2016,
- [12]Lubber P, Greco F. HTML5 Web Sockets:A quantum leap in scalability for the web[J]. Jay View, 2016.
- [13]刘华星, 杨庚. HTML5 下一代 Web 开发标准研究[J]. 计算机技术与发展, 2015.
- [14]王玲. PHP 动态网站编程[M]. 北京:清华大学出版社, 2015.
- [15]孙丹鹤, 王振. 基于 HTML5 的视频网站播控系统[J]. 电脑知识与技术, 2016.