

基于 PHP 微信端的订房系统

专业：通信工程 学生：林淑敏 指导老师：林培杰/陈晓昀

摘要

现代经济的发展和互联网的技术的进步，同时在 2009 年以后移动终端智能机的发展和普及，使得人类的生产和生活发生了很大的改变，电子商务是这时代发展下最大的宠儿。电子商务英文 ELECTRONIC COMMERCE。这是两个含义：一是以电子作为媒介实现，二是商业贸易活动。一般指的是电子互联网的手段进行的商业活动的方式，实现产品的销售和盈利，随着时代的进步，电子商务的使用空间和方式越来越多样化，使用方式越来越广泛。

在移动互联网发展的今天，微信在 2011 年诞生了，截止到 2016 年末，微信用户达到 7 亿用户，微信在 2013 年推出公众号，在微信如此多用户的基数下，微信公众号的使用越来越频繁。在微信公众号的基础上诞生了，微官网、微支付、微活动等适合商业的线上线下的微信推广方式。微信公众号中的微支付和 1bs 登陆可以给企业增加了微商城的销售方式。这种方式最主要的特点，减少了 APP 推广的安装成本，在手机使用过程中不占用过多内存。免费的占有微信的免费流量，京东在微信上面获得大量流量就是很好的例子。为商城已经成为了电商很好的模式。

作为当代大学生也深深地感受到电子商务带来的便利。不过，在现有的市场上却很少有针对大学生存在的电子商务的产品。针对学生考研、考证等考试地点在外地或者离自己所在地较远的情况下提供考试地点或者学校附近的酒店的汇总，通过价格、房源、距离等的横纵对比让学生们选择最好的订房方案，解决了学生党对于价格等方面的痛点的解决。但是互联网产品一直在是错中成长，利用现有的资源我们目前想针对考研的学生做个订房系统。看看市场反应在升级。

本设计采用 PHP 语言作为开发语言，利用 ThinkPHP 框架、MySQL 数据库、结合微信开发文档，通过了解微信开发文档的开发说明，获取个人微信信息，实现微信授权登录。本系统主要实现的功能是：用户通过微信实现用户的快速登录注册、用户通过该系统，进行在线选择指定城市，指定酒店，选择好房间，在线订房。后台管理员实现包括对城市的及其分类下的酒店编辑管理，发布到前台页面。订单模块实现用户在线订房生成订单信息，减库存等，后台管理员看到订单。支付模块用到微信支付，通过微信开发者文档的研究，提供的接口，以及申请的微信号，进行订单的支付功能实现。短信模块通过申请短信接口，阅读接口所需要的参数，对用户订房下单中使用的手机号进行发送短信消息提醒。以及之后用户可以在个人中心中查询到自己订单的消息查看，搜索等。形成订房系统的整个流程。达到在线订房目的。

关键字：订房系统 微信端 PHP 程序语言 MySQL 数据库 Apache

目 录

1 绪论	3
1.1 课题研究背景及意义	3
1.2 国内外的发展现状	3
1.3 课题研究的内容	4
1.4 组织结构	5
2 相关技术	6
2.1 B/S 模式	6
2.2 开发环境 Apache 服务器	7
2.3 PHP 语言	8
2.4 MySQL 数据库	9
3 系统分析与设计	9
3.1 整体需求分析	10
3.2 系统设计	12
4 微信端订房系统的实现	18
4.1 微信授权登录	18
4.2 商品分类模块	18
4.3 订单模块	19
4.4 支付模块	20
4.5 短信提醒模块	21
4.6 搜索模块	22
5 订房系统测试	24
5.1 订房系统测试步骤	24
5.2 主要测试用例	24
结论	27
参考文献	28
附录一	29

1 绪论

1.1 课题研究背景及意义

电子商务是从 1990 年起步，直到 2000-2009 年，已经进入稳定期，经过这么多年的发展，国外的亚马逊、易贝以及国内的淘宝、天猫、京东等著名电商品牌，带领着人类开始习惯、大力使用电商购物。完成生活必须的衣食住行。外出旅游、出差在今时今日已经是一种常见的活动，传统的电话订房已经满足不了需求，大家可以通过大众点评、美团、携程订房，已经养成习惯。对于酒店来说更希望有自己的订房系统，可以增加回头客，减少订房验证的步骤。对于中小型的酒店来说希望有一种节约成本，方便的方式完成自主订房系统。微信成为用户量大，使用人数多，依托微信公众号的订房系统更加容易推广，开发成本也远远低于 APP。那么专门针对学生考研的考点选取合适的酒店，用市面上常有的软件通常要进行价格、环境、位置的考量。然而我们的产品就是为了加快学生完成订单的过程。

1.2 国内外的发展现状

1.2.1 国外发展现状

上世纪二战结束后，苏美争霸全球，军备、科技竞赛成为量大强国的主流。世界强国美国在当时开始了计算机系统研究，为互联网的诞生提供了技术硬件支持。经过几十年的发展，互联网已经演变成全球化、商业化，发展之迅猛、应用之广是历史上前所未有的。

在美国，1999 年，整体网民数量已经达到了 1.1 亿，在如此大的互联网技术下，更多的公司开始研究互联网，直到今日诞生了 Google、Facebook 等国际大型互联网公司，促进了互联网技术的发展。也促进了 B2C 的普及和发展^[1]。

亚马逊几乎成为全球电子商务的代表、主流。以世界最长河流特殊命名的电商界老大，在互联网领域成功的抛弃了传统的零售企业的经营方式。追溯亚马逊的发展历史不难得到，亚马逊在该领域中的发展速度以及发展的劲头之大，但是相对来说资金消耗也比较多、迅速。20 世纪末，亚马逊成立，并且也像其他店家一样迎来成立之后的第一位客人。亚马逊也不出意外的在三年之后，通过互联网的销售方式，与传统行业行业形成了很大的区别，更是拥有了传统行业无法媲美的客户数，轻轻松松的将传统行业的同行零售者甩出几条街。就在这时，全球

的网上购物革命就这样开始了。

在亚马逊成立之初的三年内的的发展进程中，随着业务的壮大，物流的配送已经成为一个大问题，随之，就成立了物流配送中心。随着经营的发展，亚马逊根据市场需要调整了经营的方式从原来只出售图书到互联网上零售的业务改变，开启了多元化发展。当时，亚马逊已经越来越突出自身的企业文化、企业品牌和企业自身的优势，在服务以及用户体验上一直不断的优化、完善。在用户体验上，当用户注册后服务器会记忆 ip 地址，并且记录用户访问的数据、页面去判断该客户的喜好，去推荐相应的喜好设计的网页，还推出特色服务，比如餐馆介绍、选择送礼的推荐等。就因为亚马逊的坚持，顺应市场、信息科技的发展相结合，勇于创新，电子商务的发展才会变得如此通畅。

1.2.2 国内研究现状

目前在中国的电子商务的领域内，首当其冲的会想到淘宝。淘宝已经是中国最大电子商务大零售平台。随着亚马逊的零售方式在美国崛起并且发展迅猛，中国的阿里巴巴也在 21 世纪初成立了。阿里巴巴旗下的淘宝网零售方式有 B2C（平台对用户）、C2C（用户对用户）两大板块。淘宝成功的打败了当时很火的易趣并且荣登上中国第一的位置。业务一直在提升的同时加之智能手机的出现，对电子商务的发展就是锦上添花。在 2011 年的时候，淘宝依据市场发展适时的推出双 11 活动，活动的交易额均创历史之最。在往后的这几年里，每年的双 11 交易额的数额均是成番的增长。随着这几年互联网、网上支付等方式的快速发展，加快了电子商务在中国发展的速度，甚至已经赶超创始这个电子商务领域的亚马逊。并在不断的提升用户的购物体验。

显而易见，在创始电子商务巨头的亚马逊与目前世界上最大的电商平台：淘宝的发展始终离不开用户在购物过程中的用户体验的重视。网上购物改变了传统零售方式中信息不对称的劣势，我们拿起手机就可以轻易的对想要购买的物品进行货比三家，在时间上也节约了很多甚至包括体力。虽然目前还会有很多实体业者对于电子商务还有很多保留的意见，但是电子商务的出现对于消费者来说相对是好事，虽然会出现很多假货、劣质产品。但是无论在时间成本、还是金钱成本都是最大程度上的方便了用户。

1.3 课题研究的内容

本课题研究主要内容是基于 PHP 微信端的订房系统，微信的注册登录模块，是本软件的基础模块，实现用户的快速注册功能及登入功能。房源展示是本系统的基本支撑模块，不仅可以查看房间的情况还有房源数量。下单支付模块是一定要的，其主要实现的功能是完成电商本质的交易。还有包括后台对用户管理、订单管理也是不可缺少的，对用户购买的物品、手机、姓名等的管理和记录。在实现以上功能的基础上，还需要对 UI 界面进行美化、优化，加强用户操作习惯，能简单明了的看清楚页面并且能快速的找到自己想要的酒店。简单地说就是：缩短消费者下单付款的步骤^[2]。

1.4 组织结构

本课题主要研究的内容是：基于 PHP 微信端的订房系统，通过微信公众平台的用户基数搭建微信端的订房系统，相比 APP 省去了用户 APP 下载和推广的步骤，节约了一定的成本。借助微信的支付接口，提升了该订房系统在付款时的便捷度。其开发主要是去除繁杂的流程，旨在让用户快速找到自己所在考点的酒店，尽快订单^[3]。

本文由以下几个部分构成：

第一章 绪论

本章主要是基于 PHP 微信端的订房系统题目深入调查，了解电子商务、互联网等相关发展背景、趋势以及该方面国内外的现状，本次选题基于 PHP 微信端的订房系统的意义，然后对微信端的订房系统的主要内容进行概述。

第二章 相关技术

介绍基于 PHP 微信端的订房系统的所涉及到的相关技术，包括开发需要的环境 Apache、编程语言 PHP 超文本预处理器语言、MySQL 数据库等。

第三章 系统的分析与设计

对基于 PHP 微信端的订房系统的自身的需求与用户的需求进行对比分析，来确定我们最终需要的功能模块。并确定本系统的思维导图、流程图、原型用来更好的体现我们的需求，并对需求分析所设计的系统总体和各个模块的功能特点进行介绍。同时对本系统主要的七张 MySQL 数据库表内的内容进行简要列举说明。

第四章 微信端订房系统的实现

对基于 PHP 微信端的订房系统的各个模块所实现的功能进行详细的介绍，主要是技术层面和产品层面的说明。

第五章 真机调试

对完成的微信端的订房系统进行测试，并形成测试用例，进行简单的举例说

明。

第六章 总结

对本次的毕业设计和毕业论文相结合，进行一个比较全面的总结，包括这次毕业设计的不足。发展空间等。

2 相关技术

本章主要介绍基于 PHP 微信端的订房系统的设计的相关技术，包括了 PHP 开发的环境 Apache 和编程使用的 PHP 语言、MySQL 数据库等相关的开发技术。

2.1 B/S 模式

随着互联网技术的发展，文件/服务器的单机时代已经与我们渐行渐远，取而代之的客户机/服务器模式，而今时今日最常见的就是 B / S 模式，即浏览器/服务器的模式。该模式伴随着互联网的发展，在 C / S 模式的基础上做出了部分的改进模式^[4]。现在在 B / S 模式下，用户只需要在电脑上随便安装一个通用的浏览器，就可以轻轻松松的访问 UI 界面。简单的说就是该软件程序只在服务器端运行，用户用浏览器就可以访问服务器，访问到自己想要的数据库。B / S 模式下，大部分的事务逻辑在后端实现，很少的由前端来完成，因此很好的降低了客户端的负荷，使用户的总成本得到有效的控制。通常 B / S 模式包含的组件有安装了浏览器(Browser)的各种终端硬件设备(电脑、手机、IPad 等)、WEB 应用提供的各种网络服务器(Server)、连接浏览器(Browser)和服务器(Server)的互联网。在实际的应用过程中，为了提高数据库的并发处理能力，我们会将服务器

端拆分为 WEB 应用服务器和数据库服务器，从而 B / S 模式由浏览器、WEB 服务器、数据库服务器构成了三层次的服务结构。B/S 模式的结构如图 2-1 所示。

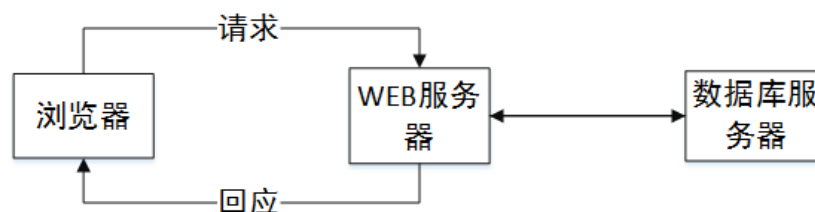


图 2-1 B/S 模式结构图

B / S 模式的优点：

(1) 用户无需安装客户端。我们不用为每一个客户端配备一个客户端软件，只要客户端具备通用的浏览器即可。并且这些浏览器可以是运行在任意操纵系统（微软、Linux、Mac OS 等）上的任意浏览器（Internet Explore、火狐、Google Chrome 等）^[5]。

(2) 适用面比较广。B / S 模式不需要专门的网络环境就能使用，并且能够在任意时间任意地方浏览相关内容。

(3) 系统的维护和升级相对比较简单。不需要逐个升级客户端，只需要对服务器进行重装，能够把维护的成本降到最低。

B / S 模式的缺点：

(1) 功能相对比较固定，很难像 C / S 模式一样可以实现一些特有的功能。

(2) B / S 模式比较注重流程，运行过程中会对权限进行多次的校验，因此响应速度没有 C / S 模式快。

(3) 由于它的适用面广、建立在广域网上，所以安全性很难得到保障。此外基于 B / S 模式的 Software 开发采用的是网络多层结构，层与层之间的互相透明，由下层为上一层提供可靠的服务，我们只需要集中精力完善每一个层的功能，而不需要关注其他更多的细节，因此 B / S 模式极大程度上的给系统开发和维护提供了便利。B / S 模式特别适用于结构复杂、地域分布广、机构众多的场景^[6]。

综上所述，本文基于 B / S 模式设计本系统系统，用户只需要打开浏览器就能登录后台管理系统。用户在登录进入系统的时候，就可以对商品、订单等进行查看、处理。这种基于 B / S 模式的管理系统达到了方便后台管理员的使用。

2.2 开发环境 Apache 服务器

apache Web 应用服务器是世界上非常流行的一个网页服务器，由于它的源代码的开源特性，所以会有很多人来修正原本的缺陷或者开发新的功能、新的特性等等。并且能够在不同的 OS 中使用^[7]。更因为它能跨多平台而且安全性高这两个优点，许多用户相对来说比较青睐于它，使得它已经成为 web 服务器端中最受欢迎的一种。Apache web 应用服务器安全性能高、反应速度快，并且可扩展性强，能够通过接口，即 API 把一些解释器如 Perl / Python 等，编译到服务器中。

Apache web 服务器包含如下几个特点：

- (1) 文件的配置过程十分简单，容易操作，可直接修改配置文件信息来修改 Apache；
- (2) 完全兼容最新的通信协议 HTTP / 1.1 和一些通用网关接口；
- (3) 支持市面上主流的操作系统；
- (4) 监视服务器状态，可以根据自己的需要来定制日志。

2.3 PHP 语言

PHP（即 Hypertext Preprocessor 的缩写，中文名为“超文本预处理器”）是一种在服务器端执行为大家普遍使用的开源的脚本语言。它的独特的语法集合了 Perl 语言、C 语言和 Java 的优点以及本身的一些创新语法，学习起来相对来说比较容易，因此，应用的范围十分广泛^[8]。

用 PHP 做出的动态页面和另外一些编程语言相比，能够更快速地执行动态网页。PHP 把程序导入到通用的标记语言 HTML 文档中去，再去执行程序，这样执行效率就会比较高；PHP 支持执行编译后代码，它有加密以及优化代码的效果，使程序运行起来速度会更快。

PHP 语言具有以下几个主要的优势：

(1) PHP 语言是免费提供的，且 PHP 的开源代码也可以唾手可得，用户可以在 PHP 的官方网站上免费下载使用，还可以将下载下来的源代码根据自己的需要去增删改。

(2) 可以跨平台运行，PHP 作为一种服务器端的脚本，能够运行在 Windows、Android、

Linux、Macos 和 UNIX 等多种操作系统平台，同时也能够与 Apache、IIS 和 Nginx 等多种 web 服务器兼容。

(3) 市面上主流的数据库都可以与 PHP 语言相结合相适应，比如 MySQL、Oracle 等，都有专用的接口函数来实现对他们的访问。

(4) 在开发过程中，由于 PHP 能够嵌入到 HTML 语言中，相对于其他的编程

语言，PHP 更容易编写，且刚接触开发的人容易上手。同时，具有占用系统资源少、效率高的特点，使开发过程变得更加便捷快速。

(5) PHP 拥有很多成熟的框架，如 ThinkPHP、CanPHP 和 Yii 等，多半是基于 MVC 架构的模式。

2.4 MySQL 数据库

MySQL 数据库是当前使用最广泛、并且是免费开源的数据库软件，更是一个跨平台的关系型的数据库系统。作为一种关联数据库管理系统，MySQL 把相关的数据放在不一样的表里面，并非把相关的数据统一放在大的库里，因此，它的灵活性比较高而且运行的速度也比较快。也采用双授权的政策，分为商业版、社区版两种！PHP 主要使用性能相对比较卓越的社区版。

MySQL 除了具备其他一般数据库软件拥有的功能以外，还拥有以下几个优点：

(1) MySQL 支持很多种不同的存储引擎，如 MyISAM、BDB、Archive 和 Merge 等，使用者能根据各个引擎的特点来选择相对适合自己存的储引擎。

(2) MySQL 提供了用于管理、检查、优化数据库操作的管理工具，更支持大型的数据库。可处理成千上万条记录的大型数据库。

(3) 占用的空间小，由于用了压缩、优化等技术，因此，它的运行速度快且安装也极其简便快捷；

(4) MySQL 提供了 JDBC、ODBC 和 TCP / IP 等多种数据库连接方式。

3 系统分析与设计

基于微信的订房系统的设计要实现用户端的微信登录模块、商品分类模块、支付模块、短信提醒模块、订单模块、搜索模块等功能。并在这些功能基础上给用户提供良好的界面交互和用户体验等需求^[9]。以下本着对上述的原则就系统整体的需求进行分析，并进行系统设计，系统程序设计流程图如图 3-1 所示：

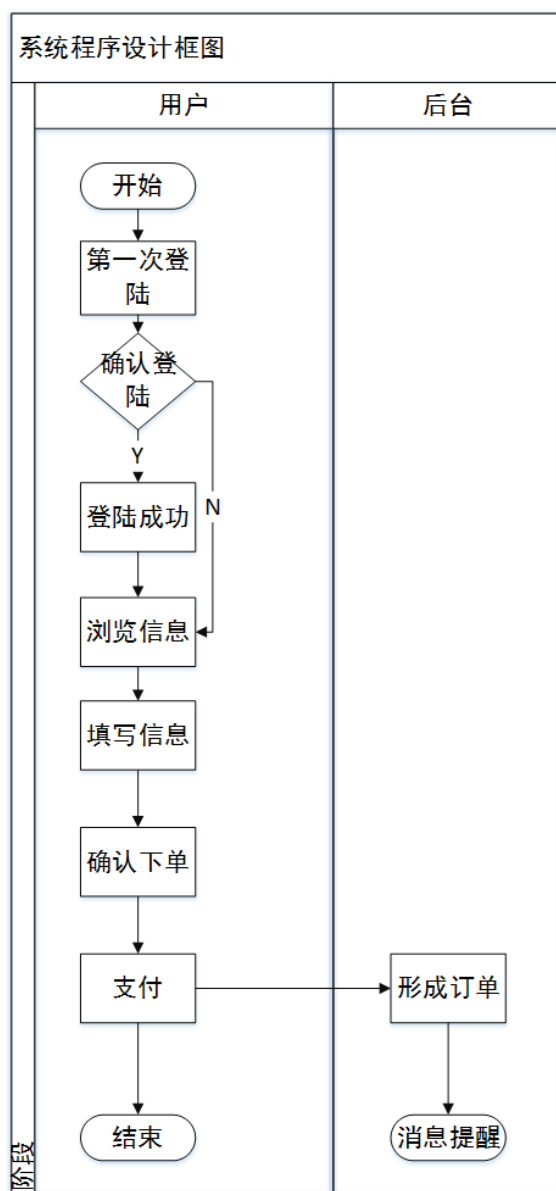


图 3-1 系统程序设计流程图

3.1 整体需求分析

3.1.1 功能性需求分析

订房系统主要模块关系图如图 3-2 所示。微信订房系统主要包含微信登录模

块、商品展示模块、支付模块、短信提醒模块、订单模块、会员管理、商品管理、库存管理、订单管理

(1) 微信登录：调用微信 API 授权登陆，降低使用门槛、方便快捷。

(2) 商品展示：先选城市，再选学校，最后选酒店；一个学校多个酒店，单个酒店可处于多个学校附近。处理数据关系有一对一、一对多、多对多三种情况，而商品分类采用多对多关系，也是最难最复杂。

(3) 支付模块：可以通过申请微信支付接口，对接支付功能，实现线上支付，便捷、安全、易用。

(4) 短信提醒模块：通过短信 API，实现登陆绑定的手机号码，自动发送短信。

(5) 订单模块：实现买家购物下单的功能。

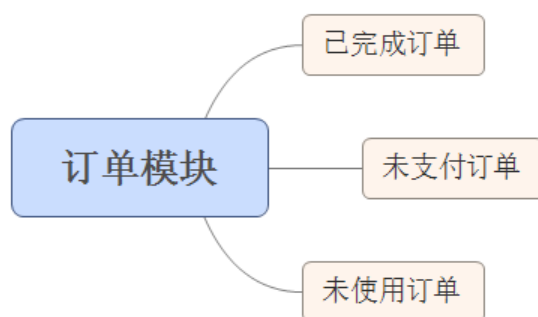


图 3-2 订单模块图

(6) 商品管理：后台实现商品管理，前台商品展示。商品种类及商品属性可以自由定义，能满足产品行业特点。

1) 商品列表：对添加的产品进行编辑、修改、删除、排序等操作。

添加新商品：

2) 酒店分类即考点：采用多级分类，适用于酒店等行业，可以把不同产品线的产品分类属性添加到系统中。

3) 学校名称：对酒店所属学校进行设置。

4) 城市：对所属学校城市设置。

(7) 库存管理：对对应酒店商品库存进行设置，退房后自动恢复库存。

(8) 订单管理：顾客在前台提交了订单之后，可以在其会员口内查询订单的处理进程，网上商城系统的后台订单处理包括订单审核、财务处理、订单入住等内容。

- 1) 订单列表：在此可以对订单进行操作，如查询、撤销、修改等。
- 2) 待入住订单。

3.1.2 性能性需求分析

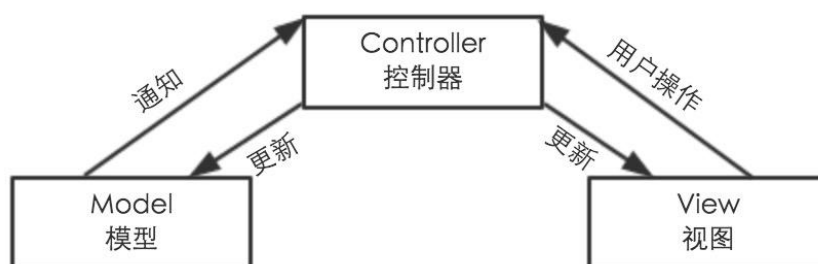
微信订房客户端为了达到良好的用户体验并实现功能性需求，本 APP 性能性指标如下：

- (1) 登陆指标：在网络正常的情况下，登录的成功率要达到 99%。
- (2) 操作指标：用户在界面操作时不能出现明显的卡顿状况。
- (3) 考点搜索：在 0.3 秒的时间内完成对输入的考点名称的搜索。
- (4) 订单功能指标：订单的内容不能出现对丢失。

3.2 系统设计

3.2.1 UI 界面设计

UI 界面设计主要是用来优化软件界面的视觉效果以及加强用户操作过程中的交互体验，并按照是市面上常用的设计规范来培养、顺应用户的操作习惯。这一层使用了 MVC 模式^[10]，MVC 模式图如图 3-3 所示。这样的模式用户在使用过程中所存储的实际数据不会因为界面的操作动作而被修改，在一定程度上保障了系统数据的安全性，换种说法也就是加强了用户交互体验。因此，本订房系统的界面首页视图上直接显示用户刚需以及订房过程中迫切需要的城市选择以及搜索功能，学校用图表去显示该城市的学校考点，页面视图简单明了，用户可以



快速搜索、寻找到自己想要的物品。

图 3-3 MVC 模式图

3.2.2 MySQL 数据库的设计

(1) 关于 MySQL 数据库的说明

MySQL 是一个小型关系型数据库管理系统，开发者为瑞典 MySQL 的 AB 公司，可以处理拥有上千万条记录的大型数据、支持常见的 SQL 语句规范、可移植行高，安装简单小巧、具有良好的运行效率，相对其他大型数据库调试、管理，优化简单^[11]。这些特点受到中小型网站的青睐。数据库会有多中的使用的方向和场景，开发人员一般将数据模型区分为两个层次，一个为关系数据模型，另一个则为概念模型。其中概念模型常被用与开发工作人员与产品用户的交互中，而关系数据库模型却是由概念模型发展转变而来的，根据计算机系统的判断对数据进行处理后再建立模型^[12]。

(2) MySQL 数据库的逻辑结构设计

数据库设计上，习惯命名方式为表前缀_模块名称（或者其它程序员看得懂英文）作为数据表名称。前缀的使用优点是可以防止表名称和数据库关键词重名，开发过程如果需要其他模块，可以通过修改前缀就可以达到数据表的重新利用。缺点是导致数据表名字过长。

(3) MySQL 数据库表的设计

数据库设计是程序开发的第一步，对今后程序的开发起者引导作用，表字段的命名是否合理规范，字段长度的设置是否符合今后程序设计中实际存储的大小、字段的类型是否符合实际开发需求、字段的主键是否设置，外键的设置，索引的设置等，关乎改表数据量大之后达到快速搜索的目的。整个过程都需要根据前期的需求分析，达到合理规划数据库的目的。本订房系统的需要设计七张数据库表。主要的数据表以及存储的信息如下所示：

本系统含有的 7 张表分别是：gwshop_admin_role（后台用户角色表）、gwshop_system_module（后台模块表）、gwshop_admin（后台登录用户表）、gwshop_goods_category（酒店分类表）、gwshop_goods_images（酒店相册表）、gwshop_goods（酒店表）、gwshop_order（订单表）。

1) gwshop_admin_role（后台用户角色表）：

后台用户角色表是用来存放后台管理人员的设置的各种管理角色权限的详细情况，其中包括已注册用户的账号和密码以及 ID。该表的结构如表 3-1 所示：

表 3-1 gwshop_admin_role 的结构

字段名	类型	长度	主键	是否为空	备注
role_id	SMALLINT	6	是	是	角色 id
role_name	VARCHAR	30	否	否	角色名称

act_list	TEXT	0	否	否	权限列表
role_desc	VARCHAR	255	否	否	角色描述

2) gwshop_system_module (后台模块表):

后台模块表是用来存放网站后台菜单的。这里面包括菜单的分类、命名、方法、控制器、图标、等级。该表的结构如表 3-2 所示:

表 3-2 gwshop_system_module 的结构

字段名	类型	长度	主键	能否为空	备注
mod_id	SMALLINT	6	是	否	模块 ID
module	ENUM	0	否	能	模块名
level	TINYINT	1	否	能	等级
ctl	VARCHAR	20	否	能	控制器
act	VARCHAR	30	否	能	方法
title	VARCHAR	20	否	能	菜单名称
visible	TINYINT	1	否	能	是否显示
parent_id	SMALLINT	6	否	能	父类菜单
orderby	SMALLINT	6	否	能	排序
icon	VARCHAR	30	否	能	图标

3) gwshop_admin (后台登录用户表):

后台登录用户表用来存放后台管理员用户名、密码等信息,甚至还包括该用户的登陆时间、添加该用户的时间以及是否启用改用户等信息。的该表的结构如表 3-3 所示:

表 3-3 gwshop_admin 的结构

字段名	类型	长度	主键	能否为空	备注
id	SMALLINT	5	是	否	管理员 ID
user_name	VARCHAR	60	否	能	用户名
email	VARCHAR	60	否	能	邮箱
password	VARCHAR	32	否	能	密码
add_time	INT	11	否	能	添加时间

last_login	INT	11	否	能	最后登录时间
last_ip	INT	11	否	能	最后一次登录 ip
status	TINYINT	1	否	能	用户状态
role_id	SMALLINT	5	否	能	角色 ID

4) gwshop_goods_category (酒店分类表):

酒店分类表用来存放各个城市内所包含的酒店名称、图片、以及对各个酒店进行排序、是否显示该酒店等的信息，该表的结构如表 3-4 所示：

表 3-4 gwshop_goods_category 的结构

字段名	类型	长度	主键	能否为空	备注
id	SMALLINT	20	是	否	类型编号
type_name	VARCHAR	20	否	能	类型名称
type_desc	VARCHAR	20	否	能	类型描述
parent_id	SMALLINT	20	否	能	父类 id
sort_order	TINYINT	16	否	能	分类排序
status	TINYINT	16	否	能	发送状态
type_img	VARCHAR	20	否	能	分类图片
is_show	INTEGER	16	否	能	是否显示
cat_group	TINYINT	16	否	能	分类分组
type_img_b	VARCHAR	16	否	能	备用图片

5) gwshop_goods (酒店表):

酒店表主要用于存放商家的商品信息，包括商品的名称、相册、概述、以及酒店各个类型的价格、剩余房间时数等。该表的结构如表 3-5 所示：

表 3-5 gwshop_goods 的结构

字段名	类型	长度	主键	能否为空	备注
id	MEDIUMINT	8	是	否	商品编号
goods_sn	VARCHAR	60	否	能	商品货号
goods_name	VARCHAR	120	否	能	商品名称
cat_id	SMALLINT	8	否	能	酒店分类

goods_brief	VARCHAR	255	否	能	商品概述
goods_desc	TEXT	0	否	能	商品详述
goods_thumb	VARCHAR	255	否	能	微缩图片
goods_img	VARCHAR	255	否	能	实际大图
add_time	INT	10	否	能	添加时间
end_time	INT	10	否	能	结束时间
spec_type	SMALLINT	5	否	能	规格类型
goods_sales	INT	10	否	能	商品销量
store_count	SMALLINT	5	否	能	库存数量
is_on_sale	TINYINT	3	否	能	是否上架
is_recommend	TINYINT	1	否	能	是否推荐
single_stock	SMALLINT	5	否	能	单人间库存
double_stock	SMALLINT	5	否	能	双人间库存
single_price	DECIMAL	10	否	能	单人价格
double_price	DECIMAL	10	否	能	双人价格
ds_price	DECIMAL	10	否	能	大床房价格
goods_school	VARCHAR	255	否	能	考点
goods_addre	VARCHAR	255	否	能	酒店地址
ss					

6) gwshop_goods_images (酒店相册表):

酒店相册表用来存放各个酒店的照片信息，通过绑定 ID 即可。该表的结构如表 3-6 所示:

表 3-6 gwshop_goods_images 的结构

字段名	类型	长度	主键	能否为空	备注
img_id	VARCHAR	20	是	否	图片 id
goods_id	VARCHAR	20	是	否	商品 id
image_url	VARCHAR	20	否	能	图片地址

7) gwshop_order (订单表):

订单表用来存放用户下单的酒店、订单号、支付编号、支付时间、生成时间爱你、以及改用户的姓名、联系电话、同住人的姓名等关于该订单的信息。该表的结构如表 3-7 所示:

表 3-7 gwshop_order 的结构

字段名	类型	长度	主键	能否为 空	备注
id	MEDIUMINT	8	是	否	订单编号
order_sn	VARCHAR	20	否	能	订单号
user_id	MEDIUMINT	8	是	否	用户编号
order_status	SMALLINT	5	否	能	订单状态
pay_status	TINYINT	1	否	能	支付状态
order_amount	DECIMAL	1	否	能	应付款金额
add_time	INT	10	否	能	订单生成时间
confirm_time	INT	10	否	能	订单确认时间
pay_time	INT	10	否	能	订单支付时间
goods	TEXT	0	否	能	商品列表信息
pay_note	VARCHAR	255	否	能	付款备注
mobile	VARCHAR	60	否	能	手机号
is_delete	SMALLINT	1	否	能	是否删除的订单
name	VARCHAR	20	否	能	姓名
live_name	VARCHAR	1	否	能	同住人信息

4 微信端订房系统的实现

4.1 微信授权登录

进入该订房系统的公众号，该服务号已经拥有微信的高级接口，此接口已经默认拥有授权作用域即 scope 参数。点击在线选房进去网站，从技术层面上，该网站通过微信已经提供好了的 API 接口，获取到微信 openid，将 openid 放入 cookie，每次执行操作对 cookie 进行判断是否存在，不存在则重新发出 API 接口请求（手机锁屏后 cookie 消失），从产品层面上，这将大大提高用户体验的舒适度，减少用户注册登陆的流程。微信授权登录页面如图 4-1 所示：



图 4-1 微信授权登录界面图

4.2 商品分类模块

用户登录成功后，进入首页，用户根据自己的考点进行选择，先选城市，再选学校，最后选酒；从技术层面上：一个学校多个酒店，单个酒店可处于多个学校附近。处理数据关系有一对一、一对多、多对多三种情况，而商品分类采用多对多关系，也是最难最复杂的^[13]。从产品层面上：通过这样多个方面、角度的对比，给用户提供了更多的选择，让用户一进网站就可以一目了然网站的功能，直奔自己的目标查看！增加了用户的体验效果。商品分类界面如图 4-2 所示：



图 4-2 商品分类界面图

4.3 订单模块

订单模块主要是通过用户选择城市，选择相应的房间信息，填写订房人的信息，电话等联系方式。用户提交这些信息，程序上我们通过 jquery 获取用户填写的信息，

以及房间名称，价格等信息，如下 jQuery 代码即页面获取信息并进行初步判断，最后将获取表单 id，通过 submit() 方法进行提交到后台 php 程序端处理。

```
<script> $(document).ready(function() {
    $("#submitBtn").click(function() {
        var name = $("input[name='name']").val();
        if(name == '') {
            alert('请输入姓名');return false;}
        $("#subForm").submit(); } }</script>
```

后台通过 I 方法获取信息并插入数据库订单表：

```
$data['name'] = I("name");
$data['age'] = I("age");
$data['mobile'] = I("mobile");
$res = M("Order")->add($data);
```

另一种是通过 ajax 异步传到 PHP 后台，PHP 后台程序将这些信息获取到，保

存到 Order 订单表^[14]。订单界面如图 4-3 所示。

图 4-3 订单界面图

4.4 支付模块

微信支付需要申请，申请成功后登陆微信商户平台，获取微信商户 ID 及微信商户秘钥（用于调取微信支付）用户下单后通过将 openid，APPID 商户 ID，商户秘钥，订单号及订单金额，通过制定加密方式，发送 API 请求，调取支付窗口进行支付，用户支付成功后微信返回支付成功通知在指定页面处理该订单信息（修改订单状态及支付时间）。加快支付速度同时也需要保证支付环境的安全^[15]。

步骤 1：网页授权获取用户 openid，步骤 2：使用统一支付接口，获取 prepay_id，这一步要提供商户号（微信公众号中可查）订单表该订单的订单号，查询出订单中的订单价格，以及订单支付成功后的通知页面，支付成功后是否需要改变订单状态等等。步骤 3：使用 jsapi 调起支付，一般这个步骤是微信官方提供的代码。我们只需要引入，并且提供其需要的参数即可。前端调起支付的代码官方也有提供事例，我们只需要根据自己页面加入代码 js 即可。支付页面如图 4-4 所示：



图 4-4 支付界面图

4.5 短信提醒模块

短息提醒主要是通过向相应的运营商申请注册短信提醒功能，运营商会给我们一个账号，通过账号密码登录，可以获取到 PHP 实现该短信功能的代码，以及 appid 等参数，最后移植到订房系统中，比如提交订单后短息提醒等。

短信提醒主要通过一下代码：

```
$url="http://utf8.sms.webchinese.cn/?Uid=".$messinfo2['regId']."&Key=".$messinfo2['regKey']."&smsMob=".$messinfo2['phone']."&smsText=".$smscontent2."";
```

```
$res=$this->Get($url);
```

其中\$url 是该运营商发送短信的链接，其中的 uid、key 是运营商提供的参数，我们注册成为会员后就可以拿到。Smsmod、smsText 即要发送的手机号、内容等。通过\$this->Get() 封装好的模拟表单提交的方法，就可以将改链接传入，从而实现用户订房后提醒。

4.6 搜索模块

前端用 html 脚本语言，添加一个<input type="text" name="type-name">的表单抓关键字，把抓到的关键字，构造 MySQL 数据库的表，抓取数据库，用 like 匹配 `select * from `gwshop_goods_category` where `type-name` like ` '%$keyword%'``；注意%是通配符，表示若干字符。这里主要搜索考试地点。搜索页面如图 4-5 所示：



图 4-5 搜索考点图

5 订房系统测试

软件测试是为了发现程序在编写过程中，人为的语法或者是漏敲代码甚至是对需求了解不清晰的非人为的逻辑错误。本章对于微信端的订房系统的主要订房、支付等各个模块进行测试，确保系统的可靠性。

5.1 订房系统测试步骤

本订房系统主要测试步骤如下：

- (1) 搭建微信公众平台；
- (2) 测试工作者分别在微信公众号的客户端以及网站后台的服务器对前后台的功能交换的进行调试；
- (3) 测试工作者对测试结果进行详细记录，分析错误产生的原因；
- (4) 对测试结果进行修改；
- (5) 再次对修改的结果进行检查，再全方面的测试各个功能，以防因修改的代码影响到其他模块，如此反复；
- (6) 对本次订房系统的测试文档进行整理。

5.2 主要测试用例

5.2.1 考点分页显示测试

(1) 测试步骤：

1) 在“城市”分类下的考点下后台管理人员通过后台管理输入 9 个考点的信息；

2) 在微信公众号的客户端上查看“城市”类别下的商品。

(2) 预期结果：

“城市”考点总数为 9，每页考点显示为 8 个考点，所以，在“城市”类别下的考点赢分为两页进行显示查看。

(3) 测试结果：

“城市”类别下的第 1 页显示了 8 个考点，再点击查看更多时，第二页的页面显示出一个考点，说明本次测试结果与预期的猜想的结果一致，证明并无出错。

(4) 错误分析：无。

5.2.2 搜索功能测试

(1) 测试步骤:

- 1) 在网站首页的搜索框输入想要查询的考点地点，比如福州外国语学校；
- 2) 点击搜索，对输入框的内容进行搜索。

(2) 预期结果:

页面将会显示福州外国语学校，点击进去将会查看到考点附近的酒店。

(3) 测试结果:

点击搜索时，页面并无变化，搜索无效。搜索考点测试结果如图 5-1 所示:



图 5-1 搜索考点测试结果如图

(4) 错误分析:

- 1) 功能设计的时候是用 form 表单提交搜索的功能，可能表单用的标签没用 submit 或者提交的方法错误或者没写。
- 2) PHP 端控制器写的方法根据搜索提交的信息，查询的条件有误（包含查询的字段条件，模糊查询，精确查询等是否有误）。

结论

随着现代经济的发展，科学技术和信息技术也随之飞速发展，基于互联网等电子商务类商务软件已经成为了人们日常生活中必不可少的工具。人们可以随时随地的在网络上进行购物，买到自己想要的物品。在任意时间、任意地点进行订购，很大程度上节约了我们的时间成本。而且在网络上可以很轻易的对所要的商品进行对比，打破了传统行业的信息不对称问题！也降低了用户的降低成本。因此本篇论文在分析了国内外互联网、电子商务的发展现状的基础上，并根据用户的需求分析设计了一款基于 PHP 微信端的订房系统。

通过深入的研究、了解互联网的发展、现状和开发技术，本设计主要包括以下内容：

（1）通过在微信搭建的平台上，去开发我们的订房系统，减少用户下载 APP 的繁琐。通过微信登录降低用户的注册门槛，提高用户数量。

（2）用户通过选取考试地点所在的城市以及所在的学校，挑选酒店。用户可以横纵对比各个酒店的环境以及价格来挑选自己想要的酒店。

（3）本订房系统主要针对学生考研订房专门设置的，订房的流程和最终目的就是让学生尽快的找到离自己考试地点近适合自己的酒店，并顺利的完成订单。所以流程简单、明了。目的清晰。

（4）本系统的支付方式选择微信支付，直接调用微信的支付接口，让用户可以进行快速的订单。

（5）短信提醒，当用户支付完成之后，会短信提醒用户下单成功。用这种正式、官方的形式让用户切实的感受到自己真的订单成功的感受。

本次基于 PHP 微信端的订房系统取得了一定的小成果，但仍然存在一些缺陷和不足，如不能退款、评论等功能。在今后的工作和生活中不断的去学习感受用户体验加上深入学习新的技术，继续完善近日的订房系统，以此可以开启今后的工作之路。

参考文献

- [1] 沈凤池.电子商务网站设计与管理[M].北京:北京大学出版社,2006:33-38.
- [2] 温昱.软件架构设计:程序员向架构师转型必备[M].北京:电子工业出版社,2012:51-55.
- [3] 戴有炜.WindowsServer 2008 R2 网络管理与架站[M].北京:清华大学出版社,2011:32-34.
- [4] 翟广宇.基于 C/S 结构的酒店管理系统[J].兰州工业高等专科学校学报,2004,4(4):21-25.
- [5] 周杨.基于 B/S 模式的社交网络系统的设计与实现[D].郑州:解放军信息工程大学信息工程学院,2013.
- [6] 李华明.基于 PHP 和 MySQL 的网上购物系统设计与实现[D].成都:电子科技大学计算机科学与工程学院,2014.
- [7] 张宴.实战 Nginx:取代 Apache 的高性能 Web 服务器[M].北京:电子工业出版社,2010:36-37.
- [8] W.Jason Gilmore.PHP 与 MySQL 程序设计[M].北京:人民邮电出版社,2011:69-70.
- [9] 郭欣.构建高性能 Web 站点(精选版)[M].北京:电子工业出版社,2012:58-61.
- [10] PAOLUZZI L, SINGH AS, PRICE DK, et al. Influence of genetic variants in UGT1A1 and UGT1A9 on the in vivo glucuronidation of SN-38[J]. J Clin Pharmacol, 2004, 44(8):854-860.
- [11] Jerome Kanter, John J. Walsh. Toward More Successful Project Management[J]. Information Systems Management, 2004, 21(2):16-21.
- [12] 张丽娟.基于企业信息系统集成的电子商务架构研究[D].昆明:昆明理工大学信息工程与自动化学
院,2002.
- [13] 文广.基于 PHP 的汉唐网上书店的设计与实现[D].成都:电子科技大学计算机科学与工
程学院,
2011.
- [14] 张璨.基于 WEB 的制造业采购管理系统的设计与实现[D].大连:大连理工大学计算机科
学与技术学
院,2016.
- [15] 沈藺.公安局内务管理系统设计与实现[D].成都:电子科技大学信息与软件工程学院,2014.

附录一

部分关键代码及解释：

微信授权登录

```
public function ind() {
    $appid = 'wxa074a81164ea0074';
    $secret = '6c341a3aad3cda56341224846f33c841';
    //回调地址
    $code= $_GET['code'];
    $uri = C('site_url').'/index.php';//当前页面完整 url
    if(!$SESSION['access_token']) {
        if(!$code) {
            $uri=urlencode($uri);
            $url="https://open.weixin.qq.com/connect/oauth2/authorize?appid="
            . $appid. "&redirect_uri=". $uri. "&response_type=code&scope=snsapi_base&
            state=1&backurl=". $url. "#wechat_redirect";//
            header("Location: ".$url); exit;
        }else{
            $this->c_url($url, true);
            $url="https://api.weixin.qq.com/sns/oauth2/access_token?appid={$a
            ppid}&secret={$secret}&code={$code}&grant_type=authorization_code";
            $userinfo = json_decode($this->c_url($url, true), true);
            $access_token = $userinfo['access_token'];
            $_SESSION['access_token'] = $access_token;
            $_SESSION['openid'] = $userinfo['openid'];
            $url2
            =
            "https://api.weixin.qq.com/sns/userinfo?access_token={$access_token}&
            openid={$userinfo['openid']}&lang=zh_CN";
            $info = json_decode($this->c_url($url2, true), true);
        }
    }
}
```

```

public function c_url($url,$https=false) {
    $ch = curl_init();
    curl_setopt($ch,CURLOPT_URL,$url);
    curl_setopt($ch,CURLOPT_RETURNTRANSFER,1);    //关闭直接输出
    curl_setopt($ch,CURLOPT_TIMEOUT ,5);        //最大执行 5 秒
    if($https) {
        curl_setopt($ch, CURLOPT_SSL_VERIFYPEER, false);
        curl_setopt($ch, CURLOPT_SSL_VERIFYHOST, false);
    }
    $content = curl_exec($ch);
    curl_close($ch);
    return $content;
}

```

商品分类模块

商品列表:

```

public function goods() {
    $school_id = I("school_id");
    $where[' goods_school'] = array(' like', '%'.$school_id.' %');
    $where[' is_on_sale'] = 1;
    $info = M("Goods")->where($where)->select();
    foreach($info as &$val) {
        $val[' images'] =
M("Goods_images")->where(array("goods_id"=>$val[' id']))->select();
    }
    $this->assign(' info',$info);
    $this->display();
}

```

订单模块

```

public function addOrder() {
    $id = I("id");
    $type = I("type");
    $info = M("Goods")->where(array("id"=>$id))->find();
}

```

```

        $this->assign('info', $info);
        if($type == 1 && $info['single_stock'] == 0){
            die("<script> alert(' 库存已不足 ');history.go(-1);
</script>");
        }else if($type == 2 && $info['double_stock'] <= 1){
            die("<script> alert(' 库存已不足 ');history.go(-1);
</script>");
        }else if($type == 3 && $info['double_stock'] <= 0){
            die("<script> alert(' 库存已不足 ');history.go(-1);
</script>");
        }
        $this->display();
    }

    public function overOrder() {
        $order = $_SESSION['order_info'];
        $goods = json_decode($order['goods'], true);
        $this->assign('order', $order);
        $this->assign('goods', $goods);
        $this->display();
    }

    //订单提交方法
    public function add() {
        $id = I("goods_id");
        $type = I("type");
        $info = M("Goods")->where(array("id"=>$id))->find();
        if($type == 1 && $info['single_stock'] == 0){
            die("<script> alert(' 库存已不足 ');history.go(-1);
</script>");
        }else if($type == 2 && $info['double_stock'] <= 1){
            die("<script> alert(' 库存已不足 ');history.go(-1);
</script>");
        }else if($type == 3 && $info['double_stock'] <= 0){
            die("<script> alert(' 库存已不足 ');history.go(-1);
</script>");
        }
    }

```

```

    }
    if($type == 1) {
        $data['order_amount'] = $info['single_price'];
    }else if($type == 2) {
        $data['order_amount'] = $info['double_price'];
    }else if($type == 3) {
        $data['order_amount'] = $info['ds_price'];
    }
    $goods = M("Goods")->field("id , goods_name , goods_brief ,
mobile , goods_address")->where(array("id"=>$id))->find();
    $goods['type'] = $type;
    $goods['price'] = $data['order_amount'];
    $goods = json_encode($goods);
    $data['name'] = I("name");
    $data['age'] = I("age");
    $data['mobile'] = I("mobile");
    $data['live_name'] = I("live_name");
    $data['pay_note'] = I("note");
    $data['order_sn'] = date("YmdHis").rand(1000, 9999);
    $data['add_time'] = time();
    $data['type'] = $type;
    $data['goods'] = $goods;
    $data['status'] = 100;
    if($data['mobile']) {
        $_SESSION['order_info'] = $data;
        $res = M("Order")->add($data);
    }
    $money = $data['order_amount'];
    $sn = $data['order_sn'];
    $openid = $_SESSION['openid'];
    $site = C('site_url');

```

支付模块

//=====步骤 1: 网页授权获取用户 openid=====

```

//通过 code 获得 openid
$jsApi = new \JsApi_pub();
if (!isset($_GET['code'])) {

    $redirect_url=urlencode(C('site_url')."/index2.php?m=Home&c=index
&a=add&money={$money}&sn={$sn}");

    $url = $jsApi->createOAuthUrlForCode($redirect_url);
    Header("Location: $url");
    exit;
}

$money = $_GET['money'];
$sn = $_GET['sn'];
//使用 jsapi 接口
//$openid = 'oG7KGwJcCoAWHzZJTACiY4LL95M8';
//===== 步骤 2：使用统一支付接口，获取
prepay_id=====

//使用统一支付接口
$unifiedOrder = new \UnifiedOrder_pub();
$unifiedOrder->setParameter("openid","$openid");//
$unifiedOrder->setParameter("body","宝尚订房");//商品描述
//自定义订单号，此处仅作举例
$unifiedOrder->setParameter("out_trade_no",$sn);//商户订
号
$unifiedOrder->setParameter("total_fee",$money*100);//总金
额
//$unifiedOrder->setParameter("total_fee",1);//总金额
$notifyUrl = $site."/index.php/Pay/notifyUrl";
//$notify_url=C('site_url')."/wxpay/index.php/Wap/Pay/wx_notif
y_url";
$unifiedOrder->setParameter("notify_url",
$notifyUrl);//通知地址
$unifiedOrder->setParameter("trade_type","JSAPI");//交易类
型

//非必填参数，商户可根据实际情况选填
$prepay_id = $unifiedOrder->getPrepayId();

```

```

//=====步骤 3: 使用 jsapi 调起支付=====
$jsApi->setPrepayId($prepay_id);
$jsApiParameters = $jsApi->getParameters();
$this->assign('jsApiParameters', $jsApiParameters);
$this->display('to_pay');
}

public function notifyUrl() {
    $postStr = $GLOBALS["HTTP_RAW_POST_DATA"];
    //$this->logger($postStr);
    $postObj = simplexml_load_string($postStr, 'SimpleXMLElement',
LIBXML_NOCDATA);
    $out_trade_no = strval($postObj->out_trade_no);
    $money = strval($postObj->total_fee);
    //$attach = strval($postObj->attach);
    $this->logger($out_trade_no);
    $order =
M("Order")->where(array('order_sn'=>$out_trade_no))->find();
    if($order['pay_status'] == 200) { //订单状态: 100-待付款; 200-
待发货; 101-已取消; 210-待收货; 220-待评价; 222-完成
    //    $this->logger('ok2');
    } else {
        if($postObj->result_code == 'SUCCESS') {
            //支付成功改变订单状态
            $order_save['order_status'] = 200;
            $order_save['pay_status'] = 1;
            $order_save['money_paid'] = $money; //写入实际付款金额
            $order_save['pay_time'] = time(); //写入实际付款金额

M("Order")->where(array('order_sn'=>$out_trade_no))->save($order_save
);

            $type = $order['type'];
            $goods_id = $order['goods_id'];
            if($type == 1) {

```

```

M("Goods")->where(array('id'=>$goods_id))->setDec('single_stock',
1);

    }else if($type == 2){

M("Goods")->where(array('id'=>$goods_id))->setDec('double_stock',
2);

    }else if($type == 3 && $info['double_stock'] <= 0){

M("Goods")->where(array('id'=>$goods_id))->setDec('double_stock',
1);

    }

    $goods = json_decode($order['goods'], true);
    if($order['age'] == 2){
        $age = "女士";
    }else{
        $age = "先生";
    }

    $note = "尊敬的{"$order['name']} {"$age}, 您好! 您在我宝尚
订房平台上预订的 {"$goods['goods_name']} 的标间已成功, 地址
{"$goods['goods_address']}, 宝尚在这里祝您用房愉快! ";
    $this->msgSend($order['mobile'], $note);
}
}
}

```

前端调起微信支付 jquery:

```

<script type="text/javascript">
    var jsApiParameters = '{jsApiParameters}';
    alert(jsApiParameters);
    //调用微信 JS api 支付
    function jsApiCall()
    {
        WeixinJSBridge.invoke(
            'getBrandWCPayRequest', jsApiParameters,
            function(res) {

```

```

        WeixinJSBridge.log(res.err_msg);
        alert(res.err_code+res.err_desc+res.err_msg);
        alert(res.err_msg);
        if(res.err_msg ==
"get_brand_wcpay_request:ok") {
            //alert(' 支付成功!!! ');

location.href="{:C(' site_url')}/wx/Order_index.html";
        }else{
            history.back(-1);
        }
    }
)
}
function callpay()
{
    if (typeof WeixinJSBridge == "undefined") {
        if( document.addEventListener ) {
            document.addEventListener(' WeixinJSBridgeReady',
jsApiCall, false);
        }else if (document.attachEvent){
            document.attachEvent(' WeixinJSBridgeReady',
jsApiCall);
            document.attachEvent(' onWeixinJSBridgeReady',
jsApiCall);
        }
    }else{
        //alert(' ok3' );
        jsApiCall();
    }
}
</script>
短信提醒模块
    //验证码

```

```

public function msgSend($mobile , $note){

    //$checkReg=M("shopsys")->where(' id=1')->getField(' checkReg');//
    会员短信提醒功能

    $messinfo2['regId']='zh1111';
    $messinfo2['regKey']='dee8274965cd9835569b';
    $messinfo2['phone'] = $mobile;
    //echo $messinfo2['regKey'];exit;
    $smscontent2 = '测试';
    $smscontent2=urlencode($smscontent2);
    $url="http://utf8.sms.webchinese.cn/?Uid=".$messinfo2['regId']
    ."&Key=".$messinfo2['regKey']."&smsMob=".$messinfo2['phone']."&smsTex
    t=".$smscontent2.""";
    $res=$this->Get($url);
    echo $res;exit;
}

public function Get($url){
    if(function_exists('file_get_contents')){
        $file_contents = file_get_contents($url);
    }else{
        $ch = curl_init();
        $timeout = 5;
        curl_setopt ($ch, CURLOPT_URL, $url);
        curl_setopt ($ch, CURLOPT_RETURNTRANSFER, 1);
        curl_setopt ($ch, CURLOPT_CONNECTTIMEOUT, $timeout);
        $file_contents = curl_exec($ch);
        curl_close($ch);
    }
    return $file_contents;
}

```