真实姓名：王先坤

用户名：141341690465005

所属服务站：潞安职业技术培训学校奥鹏远程教育学习中心

指导教师：谭智勇

**图书销售管理信息系统设计**

【摘要】：

随着互联网的蓬勃发展，传统的线下图书销售模式已经满足不了当今社会更加庞大的供给需求，线上图书销售体系慢慢地成为趋势，因此对于线上图书销售管理信息系统的设计与开发具有很强的现实意义。本文设计与开发是以图书销售系统为中心，整理线下销售体系需求，然后利用JAVA语言和MySQL数据库建立了一套线上销售与后台管理流程，完成用户系统，图书管理系统，订单系统以及消息功能，实现了线上销售的信息化，敏捷化。其对于提高图书销售管理水平有了极好的促进作用。

**【关键词】：**图书销售管理系统、前后端分离、JSP、MySQL、JSON、JQuery

【正文】：

一、绪论

（一）研究背景

随着人们生活水平的提高，购买需求也日益巨增，线下实体销售系统已经无法满足这样膨胀的的市场需求，高效的销售模式也是势在必行。科技进步也带动消费升级，互联网经过二十多年的发展也慢慢融进我们的生活，从PC机，平板电脑，智能手机以及到穿戴设备，互联网技术正在慢慢地改变我们的生活。电子商务，线上购物也慢慢生成人们生活中必不可少的部分。

在这种互联网的大背景下，互联网图书销售系统需求也是应势而生。图书具有易于包装和运输等特点，并且不需提供调试、安装、维修等售后服务。在图书印刷质量得到保障的情况下，基本不会产生图书退货。介于图书产品的固有特殊，非常适合这种线上的销售模式。

（二）研究意义

通过互联网实现图书销售体系，能够有效地减少图书销售过程的中间环节。实现出版商、代理商以及顶层销售商与读者之间直接的信息交换与沟通。由于中间环节的减少，销售商能够以更大的折扣力度提升图书销售竞争力。读者在图书销售商折扣基础上，获得更加优惠的销售价格。图书销售商在给予较大折扣基础上，仍然能够获取较之传统销售渠道更大的销售利润。并且在较大折扣情况下，能够提升图书销售量，从而创造更大的销售利润。

二、相关理论概述

（一）JSP技术

JSP(Java Server Pages)，它是一种基于Java的技术语言，用于生成跨平台的动态网页的技术，由于它的跨平台特性，项目移值也是非常便捷。此语言是由Sun Microsystems公司创立、很多开源技术公司参与一起建立的一种动态网页技术标准，由于它的标准化，它在在动态网页的建设中有着不可被动摇的地位。JSP它为创建高度动态的Web应用创造了一个独特的开发环境，这种WEB技术由于其优异的特性倍受大家喜爱并认可，很快成为当下最流行的技术语言。

（二）MySQL数据库技术

MySQL，它是瑞典公司开发的一个款小型的关系型数据库管理系统。MYSQL被广泛应用于WEB程序开发中。其速度快、体积小、成本低，并且是开放源码的。

（三）B/S模式简介

B/S，即Browser/Server，作为WEB兴起后的一种网络结构类型，本文提及的JAVA其实是此类型的一个典型的代表，本文的图书销售系统是以JAVA技术为核心，并且J2EE框架作为平台体系，基本于JRE运行环境的软件系统。WEB浏览器是客户端最主要的应用软件，主要代表有火狐和谷歌和微软的IE，但是JAVA这种通用的脚本特性不用在乎浏览器兼容差异，可以说这种模式统一了客户端。JAVA的这种体系统可以使我们将技术研发重点集中于服务器，使得开发更专一，一步解决所有问题，同时能使不同的用户，在不同的地点，不论什么接入网络，都有同样的操作体验。通过称稳定的服务系统，合理分配用户权限，非常有效的保护了本地数据库安全，以及访问安全。JAVA这种跨平台的出现与普及，其安全性及可持续也得到广大用户的认可，系统也更加稳定。

三、系统需求分析

（一）系统功能性需求

对于线下的图书消费流程，简概如下：用户选书，用户下单结帐，图书出库。通对线下图书商店的销售逻辑理解，我们将线上销售体系会有两大平台，分别为前台和后台，前台实现店铺的基本功能，如用户选书，用户下单等功能，其它包括用户操作导航，用户登录与注册，新书展示，销售排行，下单购买等。后台实现图书出库功能，即用户下单确定后，给用户发货。除此之后，后台还要实现图书，用户，订单的运营，即管理书集，新书上架，用户权限，还有订单处理等等。当然图书管理系统还有最重要的业务，就是要实现用户级别。我们将用户分为三类，即普通用户，注册用户，和管理员，不同的用户在前台和后台两大平台，呈现的功能也会有所不同。比如普通用户只能浏览前台系统，只有注册用户才有购买图书的权限，当然更高级别的管理员会拥有所有权限。

1. 前台功能需求

前台功能功能主要分导航，用户帐户模块，图书展示，用户下单，还会辅助支持个性化推荐，新书推荐，搜索等，以提高用户的购书体验。

导航模块，它是我们书店的地图，方便用户快速浏览与操作，导航上有“首页”，“书库”，“购物车”，“我的订单”，“用户登录与注册”，“售后”

首页，顾名思义，当用户在操作中找不到方向，可以直接通过首页功能回到网站初始状态，首页是图书商店的入口，可以方便找到用户喜好的图书，其中包括很多子模块，如新书推荐，售销排行榜，最近活跃用户等

书库，这里用户查书线上图书仓库里所有可以售卖的图书，支持书名，分类，发行时间等不同维度的搜索方式，方便用户以最短的时间找到自己喜欢的图书。

购物车，此模块用户展示已经下单的图书列表，用户可以随时进行批量下单并支付，只有注册用户才有此功能

订单列表，此功能用户展示用户已经支付的订单，用于查看所有购买的书籍。

用户登录与注册，此功能是个动态模块，当用户注册并登录后，只会显示当前登录的用户信息，用户未登录时，会引导用户注册或登录书店，并进行购买图书。

售后，此模块为反馈模块，方便用户在操作异常，或者有购买疑问时，通过邮件的方式通知管理员，是一个用户与管理沟通的入口。

2.后台管理功能需求

图书管理，此为此系统的重点需求，支持对图书入库，出库，下线，编辑等操作，并支持不同维度的搜索，便于管理员方便的管理书籍。

用户管理，首先它于给管理员对注册用户提查找与浏览用户信息，其次是对用户进行管理操作，如冻结有户，屏蔽用户，但管理员只能修改它状态，不能修改其基本信息，以保护用户隐私，管理员可以配合搜索，很快能找到异常用户，对于冻结。

订单管理，用来管理员管理用户生成的购买的图书清单，并对清单用效性进行管理，可以修改订单的状态或者拒绝订单，并通知购买用户。同时能够搜索订单，找到异常订单等。但是管理员不能删除用户订单信息。

反馈管理，这个是用于查看用户反馈建议等，相当于一个收件箱，收集用户反馈的信息

。

经过对网上图书销售系统的整理，我们的在管理模板做了如下模块，如后台登录模块，负责用户系统，图书管理模块负责图书系统，订单管理模块负责用户与管理的交互，公告模块负责通知广告播，同时还设计了退出后台管理系统的退出模块，以保护管理信息安全，防止他们越权误操作。通过以上功能模块，足以满足网上销售流程的完整，极大的方便管理员高效完成作业

（二）系统的其它需求

1.系统的安全性

系统的设计应当保证用户的个人信息安全性，不能外流，以保障使用者的个人隐私安全和商店的企业信息安全，从而保障客户的合法权益不受侵害。所以我们在设计此系统的过程中采用了用户权限管理的机制，通过合理设置不同用户的使用权限，使用户不能进行越权管理和使用。同时要求该系统需要具备较高的稳定性，后台数据库要求具备较高的准确性和可恢复性，防止硬件故障造成不必要的损失

2.系统的可维护性

由于用户会在系统设计完成并投入使用后，因为业务的不断增加，需求也会不断的增多，这样会对系统有更高的要求，因此我们要求该系统应该具备非常良好的可升级性和可扩展性，通够满足用户不断提高的用户客观需求，以确保用户在有新需求提出的时候，能够非常准确、及时、方便的扩展系统功能，同时能对比其它同类网上系统，能预估未来需求。

3.系统可移值性

由于每个人对于计算机的操作水平差异，系统的设计时要充分考虑适用性。因而子系统的设计应该能够达到操作方法简单、使用方便，人机交互用户界面友好等特征，对于复杂的操作，提供相应的使用手册。

4.系统的运行速度需求

系统的运行速度是普通用户用来评价系统主要性能的一项重要的指标。对于用户来说，都要求系统的响应时间尽量短，界面有响应，在进行数据更新处理时的迅速快，数据转换以及传送的时间尽量短，后台响应快，总之，用户操作后能够立即回应。所以，我们一定要在保证系统安全的前提下，保护响应的敏捷性，以防止大量用户购买排队情况，影响销售体验。

5.平台的用户可操作性

（三）系统设计可行性分析

1.技术可行性

按照该类子系统总体目标来衡量所需的技术手段是否具有，一般能从计算机硬件以及软件的使用性能要求、大坏境前提、操作管理人员技术水平以及数量等方面去顾全以及分析。考顾及到系统实施的适用性及复杂性，在软件方面我们选用了JSP技术手段来展开开发管理服务平台的设计，并使用SQLserver2012数据仓库存储数据资料。同时在硬件方面，可供抉择的空问也很大，硬件的配置越高系统运行会快，更佳流畅。于此同时，我们考虑到当今的软理件的普及程充以及家用或者商用电脑硬件的整体软硬件配置水平，此系统在当前环境下是可行的。JSP技术和SQLserver2012技术经过十多年的发展，经历互联网的考验，无论从安全性、可用性或者稳定性上来说，都是毋庸置疑的，因此，软件和硬件上都是可行的。

2.经济可行性

管理软件的的经济可行性，主要体现在生产中产生的经济效益成本与开发设计所需要的人力以及投资成本相比，是否相适宜，不仅如此，我们不光要看商业效益，还要看软件能否给用户带来更多的经济效益和使用成本。我们开发的这套系统就是为了解决客户和商家的问题，从在线销售系统的使用者或者图书店主来说，它不仅难提高工作效率，节省开销，还能方便操作，能够应付未来的更多多变的需求。本系统从开发成本来说，只是个人的独立设计，可以省略很多成本费用，同时也要提高个人动手能力，也能辅助其它平台，得到技术上的提升。对于客户来说，增加购书的便捷性，还有可能得到图书销售量数据，购书也不再盲目。

3.操作可行性

互联网的普及，Internet网络互联，这种新型的产业形式天天在影响着我们的工作和生活。不仅如此多企业为了扩大他们的生产规模，不仅将他们的市场遍布国内，同时也影响着部分国外市场，市场越大，客户越多，面临的线下销售模式早已满足不了他们的日益增长的需求。在线销售图书对于这种大范围的客户管理，有着无以伦比的优势，无论在国内网，多大的客户，只要网络传输到的地方，都能很快的解决问题。

在线图书管理销售系统，使用简单，不需要做太多的技术培训也能够迅速容易上手，加上模块的精细化，类别化，以及高度可扩展化，方便迎接未来的挑战，不仅对商业，还是用户都是很方便的。经上所述，从实施上来看也是完全可行的。

四、系统概要设计

（一）系统整体架构设计

考虑到项目的可移植性与可扩展性，我们采用前后台分离的技术架构，后端目前采用JAVA语言和MySQL数据库提供数据服务，前端采用HTML和Jquery图形操作框架，关系层用JSON数据格式定义前后端数据接口规范。在保持接口规范不变的情况下，可以随时升级前端和后端的技术类型。比如前端可以升级为交互更强的Vue和React界面框架，后端可以升级为时下更为火热的NodeJS等。具体结构如图4.1所示：

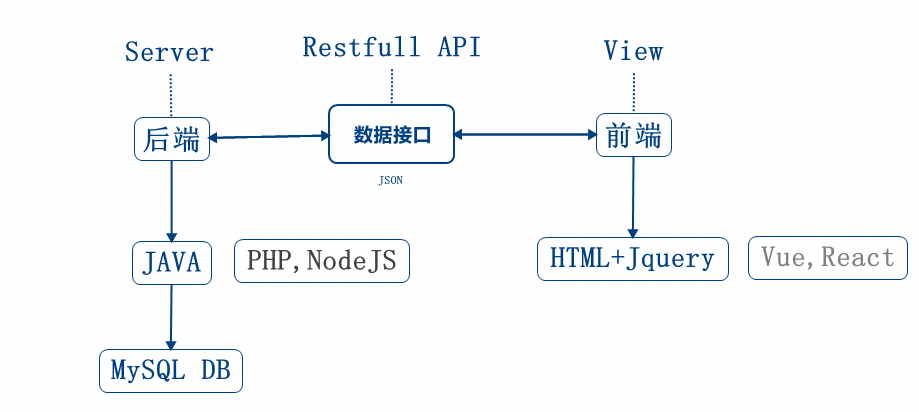


图4.1前后端架构图

在线图书销售管理系统分为前台和后台两大体系，每个体系的功能如图4.2所示：



图4.2前台和后台功能结构图

（二）系统数据库设计

1.数据库结构化设计

通过了解满足线下销售业务逻辑与流程，网上销售做了部分变化，比如增加了用户系统，也是我们的读者系统。根据用户类型，我们做了出此划分，将用户分为游客和注册用户；同时将订单分为单张详细订单和总订单（用于我们总结销售情况），普通单用户不仅可以购买单本图书，还可以购买一系列书籍，所以单用户会产生一对一订单和一对多订单，从而一个列表的数据类型可以对应多张订单。

针对图书销售的重点，即书的管理，包括书籍的信息描述，可以包含书名，作者，摘要信息，价格，封面图，还有最关键的ISBN信息，用于标识唯一一部书籍，当然还有书的状态信息，比如下架，和删除等等。除此之外，还有会员系统，描述会特征的用户名，邮箱，用户ID，还有权限相关的密码以及用户锁定状态等。最后是书与用户关联关系，即订单关系，它描术了谁谁谁，买了什么书，所以订单还要包括数据有，订单号（订单的唯一标记），购买数量，收件地址，还有最重要的用户ID和ISBN号，用于关联订单所属的用户以及订单对应的图书详情。当然还要支持消息系统，用于显示管理推送给网站的消息内容，可以修改和删除。

根据上述的数据设置需求，我们系统就可以简明的析出以下几种结构体，即用户，图书，和订单关联实体，然后不同的结构体都拥有查看详情功能，方便查看更明细内容。以上结构体之间关系的E-R图如图4.3所示。

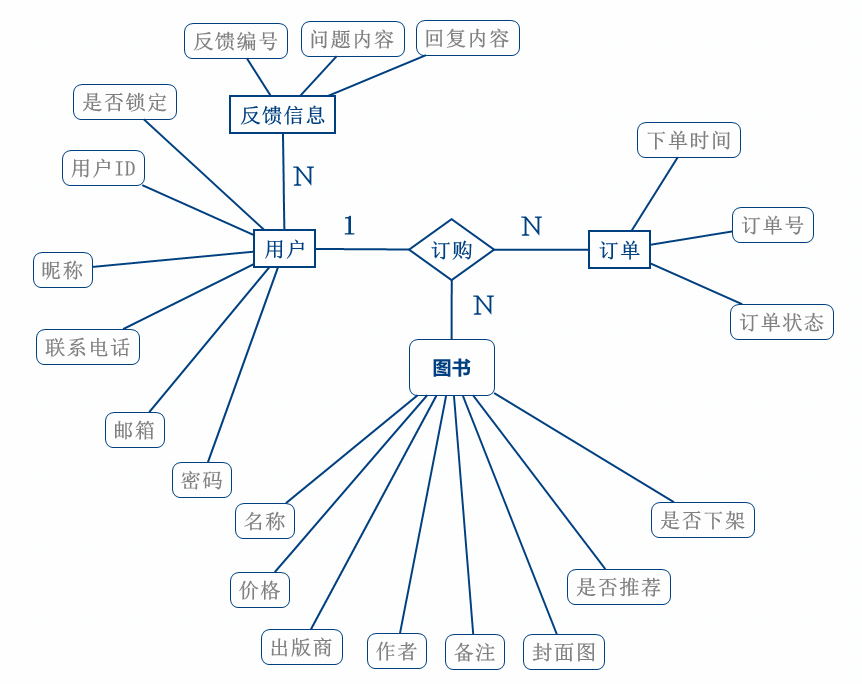


图4.3实体关系E-R图

2.数据库表的设计

首选打开并启动SQL Server2012服务，通过Navicat的图形界面连接数据库，并新建一个数据库名为db\_shops。根据上述实休关系图，我们先创建三张表，分别名为：db\_books, db\_users, db\_order，db\_books为图书表，存用于存储图书基本信息，db\_users为用户表，用于存储注册用户信息和管理员信息，db\_order为订单表，用于存储用户下的订单信息。辅助建立db\_feedback表用来存储用户反馈建议信息。具体的表内容字段如下。

表4.1图书信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 数据类型 | 长度 | 是否允许为空 | 备注 |
| Book\_id | varchar | 10 | 否 | 主键，唯一ID |
| Book\_isbn | varchar | 10 | 否 | ISBN编号 |
| Book\_name | varchar | 20 | 否 | 名称 |
| Book\_type | varchar | 20 | 否 | 分类 |
| Book\_publisher | varchar | 20 | 否 | 图书出版社 |
| Book\_author | varchar | 20 | 否 | 图书作者 |
| Book\_intro | text | 16 | 是 | 说明 |
| Book\_price | money | 8 | 否 | 单价 |
| Book\_publishdate | varchar | 10 | 是 | 发行日期 |
| Book\_pic | varchar | 30 | 是 | 封面图 |
| Book\_count | Int | 4 | 否 | 库存数量 |
| Book\_createdate | datetime | 8 | 否 | 记录创建日期 |
| Book\_isnew | Int | 4 | 否 | 是否是新书 |
| Book\_isrecommend | Int | 4 | 是 | 是否推荐 |
| Book\_isdeleted | Int | 4 | 是 | 是否已经删除、下架 |

表4.2用户信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 数据类型 | 长度 | 是否允许为空 | 备注 |
| User\_id | Int | 10 | 否 | 主键，唯一ID |
| User\_name | varchar | 18 | 否 | 用户名 |
| User\_level | Int | 4 | 否 | 用户级别标识,0为超管，1为普通用户，2,为登录用户 |
| User\_nickname | varchar | 18 | 否 | 真实姓名 |
| User\_pass | varchar | 18 | 否 | 密码 |
| User\_address | text | 60 | 否 | 收件地址 |
| User\_sex | Int | 6 | 是 | 性别 |
| User\_telphone | varchar | 12 | 是 | 联系电话 |
| User\_email | varchar | 30 | 否 | 邮件 |
| User\_locked | Int | 4 | 是 | 是否被禁用 |

表4.3订单信息主表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 数据类型 | 长度 | 是否允许为空 | 备注 |
| Order\_id | Bigint | 4 | 否 | 订单唯一ID,主键 |
| User\_id | Int | 10 | 否 | 用户ID |
| Order\_shops | Varchar | 500 | 否 | 与订单关联的书集描述信息 |
| Order\_address | varchar | 60 | 否 | 订单收件地址 |
| Order\_nickname | varchar | 18 | 否 | 订单收件人 |
| Order\_tel | varchar | 12 | 否 | 订单联系人电话 |
| Order\_createdate | varchar | 20 | 否 | 订单创建时间 |
| Order\_state | Int | 4 | 否 | 订单状态 |
| Order\_desc | varchar | 200 | 否 | 订单备注信息 |
| Order\_price | money | 8 | 否 | 总价 |

表4.6用户反馈信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 名称 | 数据类型 | 长度 | 是否允许为空 | 备注 |
| Feedback\_id | VarChar | 4 | 否 | 主键 |
| Feedback\_content | Varchar | 500 | 否 | 反馈内容 |
| Feedback\_reply | Varchar | 500 | 是 | 反馈回复内容 |
| Feedback\_createdate | datetime | 8 | 是 | 创建日期 |
| Feedback\_isdeleted | Int | 4 | 是 | 是否删除 |
| Feedback\_nickname | VarChar | 18 | 是 | 反馈方的昵称 |
| Feedback\_telphone | varchar | 12 | 否 | 订单联系人电话 |
| User\_id | Int | 10 | 是 | 反馈作者id |

（三）Restfull API的设计

Rest是Representational State Transfer（表现层状态转移）的缩写，API是应用程序接口的缩写，它是前端与后端完全分离的最佳实贱，并且定了前端与后端程序的交互规范。

1. 前台的Restfull API

前台API主要定义的接口有商品查询，订单查询，反馈信息查询，用户下单，用用户登录，用户注册等，具体定义如下：

商品展示相关接口

（1）推荐图书‘get/’‘/web/book/suggest’

（2）销售排行榜‘get/’‘/web/book/top/’

（3）图书库查询‘get/’‘/web/book/all/{$name}&{$publish\_date}&{$type}

用户相关接口

用户注册‘post/’‘/web/user/register/’

data: user\_name&password&password\_confirm&email&address&telphone

响应：{status:200,user\_id, token}

用户登录‘post/’‘/web/user/login’

data:user\_name&password

响应：{status:200,user\_id, token}

订单相关接口

（1）查看用户所有订单‘get/’‘/web/order/list/{$name}&{$order\_id}’

说明：支持按商品名称和订单号查询

响应：{status:200,list:[] }

（2）用户下单‘post/ ‘‘web/order/create/{$shop\_ids}&{$shop\_counts}’

说明：支持传和多个图书以及每本图书的数量

响应：{status:200,order\_id }

其它接口

（1）用户反馈　‘post/’ ‘web/feedback/{$content}&{$user\_id}&{$token}

说明：注册用户和普通用户都可以反馈信息给管理员

（2）查看反馈‘get/’‘web/feedback/list’

1. 后台主要RestAPI

后台所有的API接口都要传入user\_id和token以验证用户的权限，只有权限值（level为0）为管理员时，RestAPI才有数据返回，否则返回错误，并跳至后台登录页。

商品相关管理接口

1. 添加商品‘post/’‘admin/book/add/{$name}&{$price}……{$user\_id}&{$token}’
2. 删除商品　‘post/’‘admin/book/delete/{$book\_id}……{$user\_id}&{$token}’
3. 修改商品　‘post/’‘admin/book/modify/{$book\_id}&{$name}&{$price}……{$user\_id}&{$token}’
4. 搜索商品　‘get/’ ‘admin/book/list/{$name}&{$price}……{$user\_id}&{$token}’
5. 推荐商品至首页‘post/’ ‘admin/book/modify/{$book\_id}&suggest=1&{$user\_id}&{$token}’

订单相关管理接口

查看所有订单　‘get/’ ‘admin/order/list/{$name}&{$order\_id}……{$user\_id}&{$token}’

（1）确认订单并发货‘post’‘admin/order/mody/status=1’

说明：提交订单状态为2标记为待出库（0:未处理，4:拒绝，1:完成交易）

（2）拒绝订单　‘post’‘admin/order/mody/status=4’

用户相关接口

1. 查询所有用户　‘get/’ ‘admin/user/list/{$name}&{$user\_id}
2. 修改用户状态（禁用、开启）‘get/’ ‘admin/user/modify/{$user\_id}&{$status }

其它接口

1. 管理员登录‘post/’ ‘admin/user/login{$name}&{$password}’
2. 管理员注销‘post/’ ‘admin/user/logout’
3. 查看反馈信息‘get/’‘admin/feedback/list’
4. 删除反馈信息‘post/’‘admin/feedback/delete/{$feedback\_id}

五、系统详细设计及实现

（一）系统业务流程设计

1.前台基本业务流程

任何软件项目开发前，都需要进行需求分析，通过对线上其它电商平台的调研，我们将线上图书销售平台分为前台和后台两大系统，前台实现图书内容展现，用户交互，后台实现图书管理，用户及订单管理等。

前台系统主要分以下几个功能模块：

（1）首页内容模块

大首页内容模块是一个在线图书销售系统的入口，它罗列了网站导航，所有图书，新书推荐，销售排行，提供了用户登录和注册，售后入口。

（2）新书模块，它是一个自动运营模块，系统通过发布时间信息，自动化展示的数据内容，无需维护的数据，可以方便用户快速了解最新动态

（3）销售排行模块，此模块和新书模块类似，也是一个自动运营模块，系统会根据图书的销售量，将销售量靠前的书集展示出来，销售排行对购书者购书提供了有益的参考，同时吸引消费者。

（4）书库模块，它是在管理系统的图书仓库，所有未下架的图书都可以在这个模块找到。目前支持通过书名，出版社，图书作者等多个维度检索，还支持分类检索，缩小检索范围，方面购书者在短时间内找到自己想要的书籍，用户可以通过检索到的结果，随时将找到的书添加至购物车。

（5）登录与注册模块，主要是用户进入网站的入口，只有登录用户才有购书的权利。所用用户参于购，必须先要注册成为网站会员才可以，建立起用户关系，然后登录后进行用户与图书的销售关系绑定，才能完成订单的生成。对于用户特征信息登录后，管理员是无权修改的，管理员只能对期进行锁定操作。当然，对于注册用户，管理员通过了后才可以使用，网站才能为他提供服务。与此同时，用户要真正完成订单流程，填写正确的联系方式也是非常有必要的，不然是无法进行售后收货的。

（6）购物车模块，是方便用户购物的一个辅助工具，过去的网购只能对一件商品下单，有了购物车，可以随便将看到的商品放入车中，并能够随时从车中移除，大大方便了用户。同时通过将车中所有的书籍一并发放至收银行台，一并结帐，简化了很多操作流程。在结帐前，还可以一键清空购物车，为下次重新购物做准备。

（7）我的订单模块，它用来方便用户随时查看历史订单，也可以方便用用户跟踪正在进行的订单，了解订单的处理进展。订单模块支持按订单号维度搜索订单。订单是结帐的凭证，在购物车下单后，即生成购物商品的唯一清单信息。

（8）订单结算，它是一最重要的模块，以上所有模块都是为此功能服务，当用户将图书添加至购车，通过“结算”进入此流程。如图：5.1

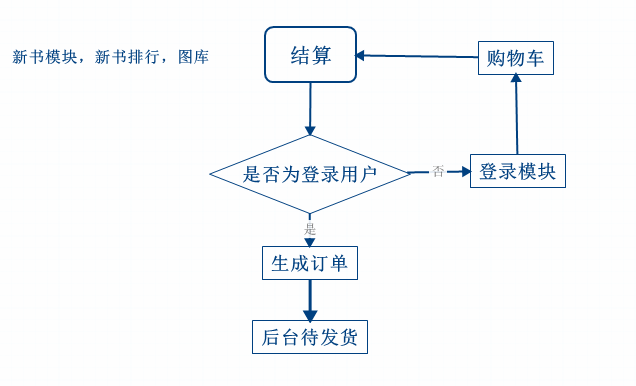


图5.1订单结算流程图

2.后台基本业务流程

后台主要分以下功能划块：

（1）登录模块，登录模块是校验管理员权限的中间模块，图书管理员通过这个唯一入口进系统，服务端通过校验用户名和密码信息，认证为管理员身份，是则进入后台管理页面，不是则返回到网站首页。

（2）图书管理模块，此为系统的最为核心一部分，它用来对图书基本信息进行一系列操作管理，其功能包括新书的入库，图书的基础信息的编辑（如封面，书名，版号等等），书的下架，精编推荐等行装，当然还包括基本的搜索功能，如按名书，ISBN，入库时间范围，等。便于图书管理员的查询。

（3）订单管理，它主要是用于管理员查看和修改订单状态，方便反馈给下单的网站会员，但是图书管理员绝对不允许修改订单的基础信息，否则未导致不可预估的后果。

（4）用户管理，此模块主要用于图书管理员对注册会员各种权限操作，如锁定用户，当然还支持搜索用户，发现异常用户等，管理员无权修改用户基本信息，但是可以辅助帮会员重置密码信息，便于解决日常问题，比如会员忘记密码等等。

（5）登录模块，其实它是一个不可见的中间模块，每当管理员发送接口请求时，都要校验user\_id和token信息，一但校验不是管理员身分，系统自动跳转至登录模块，要求进行权限认证，认证通过后才可以进行后续操作，当然如果通过前台系统登录后，如果是管理员身份，也可以直接通过前台进行后台，而不必要再次经过后台登录模块登录。

（6）退出模块，主要是清空登录权限，防止陌生人误操作造成安全隐患。

后台业务流程如图5.2所示：

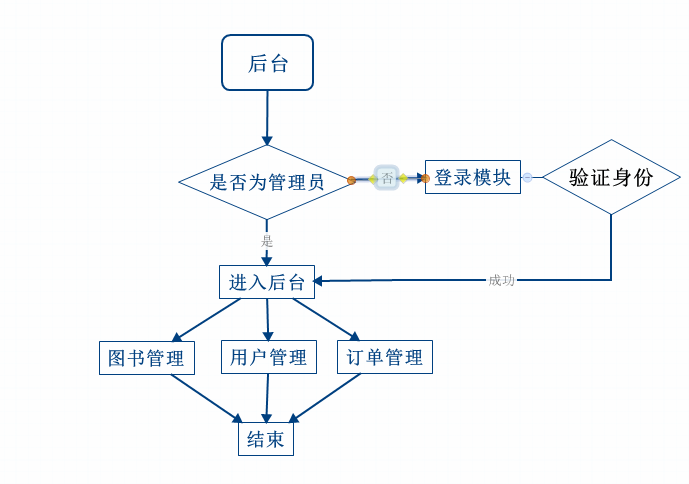


图5.2后台业务流程图

（二）系统前台功能及模块的实现

1.首页模块模块

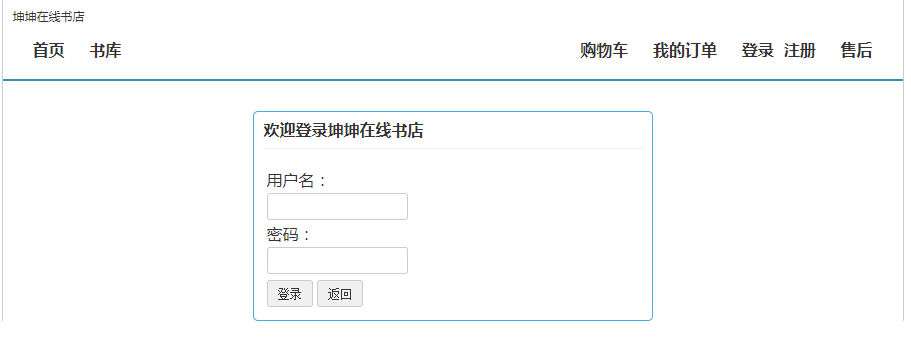
首页是一个站点地图，它有导航，提供登录注册，订单，购物车等二级入口，还有推荐图书，热点图功能。如下图：

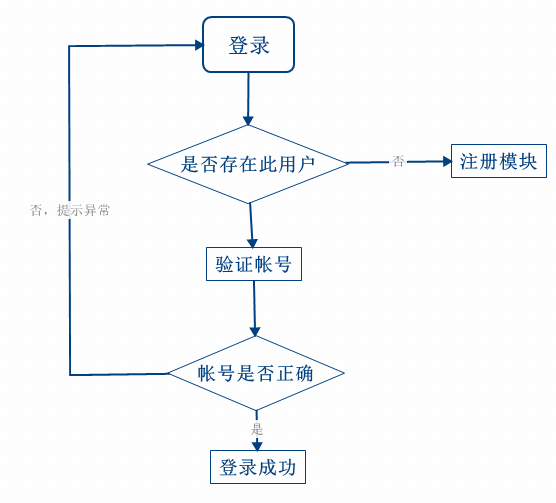


2.用户注册与登录页

普通会员功能，当用户首次访问网上图书超市时时，如有购买意图，先要进行用户注册，然后成为会员用户。注册成功后才可以进行后续的购买操作。我们的在线图书销售系统采用会员权限制度进行管理，每个想购买图书的用户必须先要注册并成为本网站的会员用户，然后才可以进行后续操作，对于普通游客用户只有浏览的权限，无法购买书籍。

用户登录如下图：





当管理员用户登录时，登录后还要验证是否为管理员，如果为管理员时，网站导航栏还要显示管理员入口，方便管理员直接进入，省去从管理登录页登录的麻烦，如下图：



注册模块功能和登录流程类似，注册前判读数据库是否存在同用户名，如果存在，则跳转登录，否则进行后续验证，表单信息填写正确后，则完成注册并跳至网站首页。

如下图：



图5.3用户注册流程图

3.购物车模块

购物车模块用于临时保存在用户在图书网站上所挑选的图书，它不依赖后台API，数据全是通过前端JS脚本存储在本地浏览器缓存中。当用户点击“购买”时，会在这个缓存中追加数据，用户可以有选择的清空选择的项目，也可以清除所有项目，这个和线下购物有些同样的操作体验。用户确定好要购买的图书，直接点击“结算”后提交并生成订单，同时从购物车清空已经购选的书集。



图5.4购物车模块

4.我的订单模块

我的订单询模块，此功能此模块是为了方便用户查看订单信息的执行状态，用户通过购物车结算并下单后，所产生的订单信息会展示在此模块中，当然历史订单也会这此模块展示。用户可以根据查看订单的状态，了解管理员的处理进展。订单模块支持按照订单号搜索订单。

5.书库模块

此模块前文已经简要提及，当会员在首页无法找到喜欢的图书时，可以在图书库中检索出自己喜欢的图书。用可以从首页导航“书库”，或者从首页的图书“搜索”框中搜索进入。书库右方有图书分类，用户可以根据自己的兴趣爱好进行分类查找，然后通过“书名”，“作者”等不同维度抽像搜索，精确的找到自己喜欢的图书。当用户查到自己喜爱的图书，可以随时通过检索结果中的“购买”按纽进行下单添加至购物车，并进行结算。

图5.5图书分类页面

（三）系统后台功能及模块的实现

1.图书管理模块

图书管理模块是此系统最为复杂的模块，它除了要实现图书的基本功能，如新书入库，书籍编辑，旧书下架，还要支持图书推荐，是否新书等，同时支持图书按照书名，日期，ISBN等多维度的条件搜索。

（1）图书信息的添加

点击操作添加图书信息时，首先要确保入库书集不能重复，这个就要首先判断输入的图书唯一代号（即ISBN号）在数据库中是不能够重复，即是唯一的，当用户输入了重复的ISBN信息时，界面要相应的给予错误提示，告之管理员已存在同名书籍，并重新修改后再提交。但唯一的图书入库后，回到图书列表页，刷新列表，显示最新入库的书籍，继续等待新书入库。

如图：



（2）修改图书信息

修改书籍，业务功能基本同添加模块，也要进行入库书籍标识鉴别，如果库中不存在的书籍也应该提醒管理员，不允许修改。具体流程，首先要确定和判断ISBN的参数值，当ISBN的参数值不为空时，就通过ISBN查询图书的详细信息，并将相应的信息显示显示在我们规定好的表单里，对于普通数据直接通过文本框显示，对于boolean类的数据进行逻辑处理，显示单选框，用户提交后，调用restfull api. /admin/book/modify，实现对ISBN相应数据的更新。当然，出现异常ISBN或者数据库服务异常，还要给予错误提示，以友好的方式通知管理员。

（3）删除除图书信息

删除图书信息相对业务简单，完成ISBN的校验时，如果校验通过，则将ISBN数据中的删除标记置为真，否则不处理或者给予管理员异常提醒。删除时不建议将数据删除，如果删除了数据，对于用户还未完成的订单就无法查询到订单关联的图书信息，也对后续恢复数据造成困扰。

（4）图书的查询与分页

在完成以上增添改时，最终会落到列表页，此模块就是方便整理并显示所有图书信息，包括已经删除的信息。它可以将数据库中保存的图书信息，按照书名、作者、出版社、入库时间、删除状态等以列表的形式展现出来，同时能够通过以上维度进行搜索操作。在每一条列表信息后面还要支持各种操作功能，方便管理员有针对的进行“添加”“修改”和“删除”操作。当然点击标题文本，还能快速地查看书籍详情以及销售量等等。

图书管理的页面如图5.8所示。



图5.8图书管理页面

2.用户管理界面

用户的管理功能除了能够浏览和查询用的基本信息，同时还能对购物有不良记录的用户进行禁用操作，以禁止此用户扰乱正的图书管理秩序，同时具有提醒用户功能，以警示用户。



图5.7用户管理界面

3.订单管理模块

此模块涉及用户，以及图书商品，还有用户与商品的关联关系。主要功能是通过订单ID信息，反馈出与其关联的用户信息和商品的详细信息，当然还包括了订单的状态信息（如发货，拒绝等），对于订单，管理员只有的查看的权限，对于已经订单信息是不可以随便进行修改的，否则会产生不可补救的后果。

对于订单的操作功能要做如下控制，比如已经完结的订单（已发货）或者已经拒绝（商品下架）的订单，只支持“详情”操作，不允许二次发货或者拒绝，只有待处理的订单才有发货和拒绝两个操作功能。具体展示如下图：



4.反馈管理模块

此功能为售后模块，用于处理注册用户和普通用户问题，与用户进行线上交流。用户可以通过前台系统提交问题，管理员在后台系统处理回复，此时前台用户在反馈入口也能即时的看到管理员的回复内容，实现交流。目前暂不支持二次回复功能，已经回复的问题，不再支持回复操作。



图5.9公告管理页面

5.部分Java服务程序

（1）数据库连接模块

package utils;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class DBUtil {

// 数据库连接路径

private static final String URL = "jdbc:mysql://127.0.0.1:3306/db\_shops?characterEncoding=utf8&useSSL=false";

private static final String NAME = "root";

private static final String PASSWORD = "root";

private static Connection conn = null;

static{

try {

// 加载驱动程序

Class.forName("com.mysql.jdbc.Driver");

// 获取数据库的连接

conn = DriverManager.getConnection(URL, NAME, PASSWORD);

} catch (ClassNotFoundException e) {

e.printStackTrace();

} catch (SQLException e) {

e.printStackTrace();

}

}

...

}

（2）Restfull API基本实现

通过HttpServlet对象获取request参数，根据参值进行数据查询或保存操作，如下：

// GET /admin/book/list/?name

@Override

public void doGet(HttpServletRequest request, HttpServletResponse response) {

String param = request.getParameter("name");

Integer key = (param == null) ? null : Integer.valueOf((param.trim()));

boolean json = false;

String accept = request.getHeader("accept");

if (accept != null && accept.contains("json")) json = true;

String payload = "";

Object list = null;

if (key == null) {

list = this.dao.query(null);

if (json) payload = JSONObject.fromObject(list);

}

// Otherwise, return the specified Novel.

else {

HashMap map = new HashMap();

map.put(0, key);

list = this.dao.query(map);

}

this.sendResponse(response, payload);

}

（3）数据DAO的增，添，改，查

/\*\* 添加图书信息

\*

\* @param book

\* @throws Exception

\*/

public void addBook( Book book ) throws Exception{

// 首先拿到数据库的连接

Connection con = DBUtil.getConnection();

String sql="insert into tb\_books"

// isbn、书名、图书价格、图书作者、出版社

+ "(BOOK\_isbn, BOOK\_name, BOOK\_price, BOOK\_author, BOOK\_publisher,"

+ "BOOK\_type, BOOK\_pic, BOOK\_intro) "

+ "values("

+ "?,?,?,?,?,?,?,?)";

PreparedStatement psmt = con.prepareStatement(sql);

// 先对应SQL语句，给SQL语句传递参数

psmt.setString(1, book.isbn);

psmt.setString(2, book.getBookName());

psmt.setFloat(3, book.price);

psmt.setString(4, book.author);

psmt.setString(5, book.publisher);

psmt.setString(6, book.type);

//执行SQL语句

psmt.execute();

}

修改图书：

/\*\*

\* 更新图书信息

\*/

public void update(Book book) throws SQLException{

// 首先拿到数据库的连接

Connection con=DBUtil.getConnection();

String sql="update tb\_books "

+ "set BOOK\_isbn = ?, BOOK\_name = ?, BOOK\_price = ?, BOOK\_author = ?"

+ ",BOOK\_publisher = ?, BOOK\_type = ?, BOOK\_intro = ?"

+ "where ID = ?";

// 预编译sql语句

PreparedStatement psmt = con.prepareStatement(sql);

// 先对应SQL语句，给SQL语句传递参数

psmt.setString(1, book.isbn);

psmt.setString(2, book.getBookName());

psmt.setFloat(3, book.price);

psmt.setString(4, book.author);

psmt.setString(5, book.publisher);

psmt.setString(6, book.type);

psmt.setString(7, book.intro);

psmt.setInt(9, book.ID);

// 执行SQL语句

psmt.execute();

}

六、系统测试

（一）系统测试环境搭建

测试环境包括硬件环境、软件环境两部分。

系统测试的硬件环境如下所示：

硬件环境：CPU 类型：Inter(R) Core(TM) i5-3470 CPU @3.20 GHZ ；

CPU个数：4个

CPU一级缓存：4\*64kbytes

CPU二级缓存：4\*1024kbytes

总线类型：FSB

内存类型：DDR3

内存容量：8GB

硬盘容量：500GB

系统测试的软件环境如下所示：

软件环境：

操作系统：Windows win7

网络服务器：Tomcat5.5；

JDK: java version "1.8.0\_181"

数据库：SQL Server 2012；

前端框架：Jquery1.8

浏览器环境：InternetExplorer10, GoogleChrome 58.0.3029.110, Firefox39.0

（二）部分功能测试

1.前台功能测试

测试目的，验证软件功能的可用性，功能稳定性以及在不同客户端环境的兼容性。

测试内容：

（1）测试前台页面在不同浏览器环境下的展示表现，有没有出现较大的布局异常

1. 后台功能测试

测试目的：验证后台服务的功能可用性，准确性

（三）测试结果

一个功能健全的图书销售系统的设计与开发的基本实现，数据库字段在操作中也能正确更新，满足了现有需求，读取过程中未现异常卡顿情况，系统功能基本符合预期。

经过上述对于系统的功能测试，并得以下总结：

（1）该系统的软件界面响应速度时间小于1秒（目前是单用户测试）。

（2）软件能够处理各种异常状况，并且给予友好的提示，体验较佳。反复操作不出现明显故障，表现出很好的可靠性和健壮性，并在低配环境表现稳重。

综上所述，基于线上图书销售系统的设计与开发的系统完全符合预期，可以进行软件验收。

总结

本文主要介绍了在线图书销售系统的设计和实现，通过对线下销售的流程总结并系统的进行需求分析，通过JAVA, SQLServer技术设计，并详细的进行各项功能测试以及子模块验证，完成了该系统。本文取得的重要成果有：

（1）能够运用现在学习到的技术解决现实生活问题，并能改善生活

（2）通过对线下图书销售进行业务流程分析，根据当下国内环境，提出可行性方案进行系统的功能需求和性能需求评估，并根据所学知识点完成流程设计、功能设计、时序图设计和数据库设计，最终确定前台和后台架构。

（3）图书销售系统，前台主要实现了会员注册与登录，图书展台与推荐、购物车、用户通知、订单查询等业务。其中前台通过推荐，新书上架，分类浏览，销售排行榜功能极大的吸引了用户，有利于提供图书销售水平，并在不同模块都添加了增加购物车入口，很大程度的方便了用户，提升操作体验。后台主要是图书管理，用户管理，订单管理，三大核心功能，让图书线上销售业务清晰。同时增加用户锁定功能，方便管理员维户图书管理秩序。在图书管理上优化多维度搜索，便于管理员快捷查找，提高图书维护效率。还在订单功能，实现了不同维度状态，方便快捷的检索出异常订单，便于管理员跟踪订单状态。当然还辅助增加公告等功能，提供整全网站的交互体验，方便图书管理员与用户的互动。

（4）通过对软件系统的功能测试，500个用户登录平均响应时间为2.512。此软件在超过压力，低硬件设备条件下仍然能够保持良好的运行，表现了良好稳定和健壮性。

【参考文献】

[1]赵娜.图书销售系统流程分析——以无锡某出版社为例[J].中外企业家,2020(17):99.

[2]余兵兵.关联规则挖掘在图书销售系统设计中的使用[J].科学技术创新,2019(01):86-87.

[3]孙加兵,施阳.网上图书销售系统的分析与设计[J].微处理机,2018,39(01):32-35.

[4]李小为,张健.中小型书店信息管理系统的设计与实现[J].电脑知识与技术,2016,12(36):14-16.

[5]况昊.网上书店详细设计与开发[J].现代商贸工业,2016,37(13):176-179.

[6]胡能发.基于UML图书销售系统分析与设计[J].电脑知识与技术,2015,11(26):67-68.

[7]郭健,任永功.云计算环境下的关联挖掘在图书销售中的研究[J].计算机应用与软件,2014,31(11):50-53.

[8]杨巍.基于VFP的图书销售管理系统的设计与开发[J].中国管理信息化,2014,17(21):32-33.

[9]杨晨,韩旭.基于ASP的网上图书销售系统的设计与实现[J].电子世界,2014(15):119.

[10]姜静,尹秀叶.书店购销管理系统设计与实现[J].电脑知识与技术,2013,9(26):5908-5910.

[11]刁宇飞.网上图书销售系统的设计与实现[J].电子技术与软件工程,2013(16):84.