

全日制本科生毕业论文

`

**题 目：****基于B/S结构的交流论坛的初探**

**学 院： 计算机与信息科学学院**

**专业年级： 计算机科学与技术（师范）**

**学生姓名： 杨路 学号：20130516207**

**指导教师： 蒋明宇 职称： 实验师**

**2017 年 5 月 7 日**

基于B/S结构的交流论坛的初探

计算机与信息科学学院 计算机科学与技术（师范）专业 2013级

指导教师 蒋明宇

**摘 要：**随着信息化社会的到来，为了适应各个机构或者组织的交流沟通以及信息共享的需求，BBS论坛系统应运而生。本文介绍了什么是B/S结构以及基于B/S结构的BBS论坛系统开发的研究背景和意义及其开发前景，详细介绍了本系统所采用的JAVA相关技术。根据BBS论坛系统的功能的基本需求，进行了较为详细的分析与探讨。

本系统主要采用JetBrains公司的IntelliJ IDEA开发，基于Spring全家桶及FreeMarker架构实现的，数据库采用MySQL对象关系型数据库。系统的使用者主要有三种角色：游客方面，游客可以查看及搜索不同板块的帖子；用户方面，用户可以发帖及回帖；管理员方面：除了拥有用户的功能，还可以实施对用户及帖子的管理，以便维护一个和谐健康的网络环境。

**关键词：**JAVA、B/S、Spring、对象关系数据库、论坛

**Abstract：**Along with the advent of the age of information. To meet the needs of communication and information sharing of social organizations, internet forums were born at the right moment. The paper introduces what B/S architecture is and the background of internet forums and the values of its development. Besides, it also presents some relative technologies of Java that mainly use in the internet forums system. According to the basic demands of the system, the paper provides some detailed analysis and discussion.

The system was developed by JetBrains company’s IntelliJ IDEA, and was made of Spring and Freemarker architecture. What’s more, it depends on MySQL as its database. There are three roles of this system: for the tourists, they are able to search and check posts on BBS; for the user, they are able to start and replay a topic on BBS; for the administrator, they possess supreme authority. Besides those authority above, they have right to manage users and topics, so that we will have a healthier and harmonious internet environment.

**Key words：**JAVA ; B/S ; Spring ;  [object-relational database](http://www.baidu.com/link?url=09fXCoxIR6gCZ0cg7P-CpuBBYNcSdOueFaZ811Tj0r82d5hkM770hjCXVEFQ7oOtXWN-vAwpcuIJOCgKJkwc0vSCWJ6sacZ74OyLGKo1569KF7AE8GrrAsn-8IXbRXCF" \t "https://www.baidu.com/_blank) ; online forum

1 引言

1.1 背景及意义

论坛英文全称Bulletin Board System(电子公告牌系统)，英文缩写BBS，它是基于B/S结构的一种为广大网民提供交流平台的电子信息服务系统。随着互联网的普及，现代社会的信息呈爆炸式增长。如何有效率的传递以及获取信息成为了制约互联网发展的一道障碍。此外现实生活中的交流沟通难以逾越时间和空间上的局限性并且受众面小达不到广泛传播的目的。而论坛的出现无疑为解决这个问题提供了一种有效的方案。

在我国最早的BBS论坛是北京长城站，建于1991年。由于当时互联网并未广泛应用，论坛每天的访问量只有十几人。1994年的春天，曙光BBS论坛——建成了。接着由于互联网的发展，各类论坛如雨后春笋般涌现。论坛的功能愈加完善，内容范围也愈加丰富。如今论坛系统将朝着以下两个方面发展：论坛系统将朝着即时性方向发展，信息是有时效性的，因此即时的信息将更加有价值。现在动态网站使用Ajax技术，在不需要刷新页面就可以更新数据，从而提升论坛内容响应的即时性 [1] ；论坛系统将朝着图形化方向发展，致力于提升用户体验。论坛已经成为了人们平时互动沟通的常用选择之一。它可以用于公告，在线聊天群组讨论，联系，社区内通信，找工作等，利用网络实时，便捷的特点来方便人们的交流，联系和娱乐。通过论坛的方式，增强个人与个人，团体与团体，个人与团体之间的交流联系，在互联网上实现信息的传递，提高办事效率。因此论坛建设在互联网应用上的地位显而易见，它已成为现代人沟通和获取信息的重要组成部分，从而倍受人们的重视[2] 。网络论坛和互联网上的其他信息一样，具有范围广，内容庞杂，动态变化性强等特点，但是两个最重要的特点是交互性和时效性。交互性是指论坛用户能够参加到论坛信息的交流过程中来，可以在论坛中发布自己的信息并且可以得到其他用户的反馈，这是网络论坛信息最基本和最重要的特征。一位用户发布信息，往往有很多人回应，可以开展多人讨论，这是一个多向交流信息的过程，它使人们能够便捷地进行交流。由于互联网信息传播的快捷与方便以及网络论坛的交互性，使得人们能够及时的从论坛中获取某一技术和行业的发展动态与最新的进展，专业技术论坛中信息的时效性更强，往往一些最新的技术信息都可以从论坛中获取。

1.2 B/S结构简介

  B/S结构，即Browser/Server(浏览器/服务器)结构，是随着Internet技术的兴起，对C/S结构的一种变化或者改进的结构。在这种结构下，用户界面完全通过WWW浏览器实现，一部分事务逻辑在前端实现，但是主要事务逻辑在服务器端实现，形成所谓3-tier结构。B/S结构，主要是利用了不断成熟的WWW浏览器技术，结合浏览器的多种Script语言(VBScript、[JavaScript](http://lib.csdn.net/base/javascript" \t "_blank" \o "JavaScript知识库)…)和ActiveX技术，用通用浏览器就实现了原来需要复杂专用软件才能实现的强大功能，并节约了开发成本，是一种全新的软件系统构造技术。Browser客户端，Web App服务器端和DB端构成所谓的三层架构。B/S架构的系统无需特别安装，只要有浏览器即可。B/S架构中，显示逻辑交给了web浏览器，业务处理逻辑放在了Web App上，这样就避免了庞大的客户端，减少了客户端的压力。优点：（1）客户端无需安装，有浏览器即可。（2）B/S架构可以放在广域网上，通过一定的权限控制实现多客户访问的目的，交互性较强。（3）B/S架构无需升级多个客户端，升级服务器即可。缺点：（1）无法完美兼容不同浏览器。（2）表现要达到CS架构的程度要花费不少精力。（3）在速度和安全性上需要花费巨大的设计成本。

1.3 开发工具及技术

1.3.1 开发工具

此次设计主要采用的编辑器为IDEA 全称 IntelliJ IDEA，是java语言开发的集成环境，IntelliJ在业界被公认为最好的java开发工具之一，尤其在智能代码助手、代码自动提示、重构、J2EE支持、各类版本工具(git、svn、github等)、JUnit、CVS整合、代码分析、 创新的GUI设计等方面的功能可以说是超常的。IDEA是[JetBrains](http://baike.baidu.com/item/JetBrains" \t "_blank)公司的产品它的旗舰版本还支持HTML，CSS，PHP，MySQL，Python等。免费版只支持Java等少数语言。

1.3.2 Spring Boot

Spring Boot是Spring社区较新的一个项目。该项目的目的是帮助开发者更容易的创建基于Spring的应用程序和服务，让更多人的人更快的对Spring进行入门体验，让Java开发也能够实现Ruby on Rails那样的生产效率。并为Spring生态系统提供了一种固定的、约定优于配置风格的框架。

1.3.3 FreeMarker

FreeMarker是一个用Java语言编写的模板引擎，它基于模板来生成文本输出。FreeMarker与Web容器无关，即在Web运行时，它并不知道Servlet或HTTP。它不仅可以用作表现层的实现技术，而且还可以用于生成XML，JSP或Java文件等。

2 需求分析

2.1 用户需求

交流论坛方便了不同地域的网民分享自己的知识、经验、技术、见解、感悟。基于安全性考虑，本系统应该能进行登录和注册。未登录的用户为游客，只有浏览和查询帖子的权利；登录的用户可以发帖回帖和点赞收藏帖子等功能；管理员有最大的权限，可以对用户和帖子进行管理，以维护一个和谐绿色的上网环境。

2.2 功能需求

2.2.1 用户注册需求

没有论坛账号的用户可以通过注册获得属于自己的账号密码，从而可以有更多的权限。

2.2.2 用户登录功能需求

当用户试图对帖子进行操作时（发帖、回帖、点赞、收藏），弹出提示框，提升需要用户登录，登录成功之后回到主页。

2.2.3 板块分类需求

用户可以通过选择板块的分类来浏览对应的类型的帖子，节省用户自己查找相关帖子的时间。

2.2.4 发帖回帖需求

已登录的用户有发帖回帖的权限，可以在任意板块发帖回帖。

2.2.5 个人中心管理功能

用户登录系统之后可以对自己的个人信息进行编辑，例如昵称，签名，头像等。

2.2.6 后台管理功能

管理员账号可进入后台系统对帖子进行删除修改加精置顶等操作，可对用户进行拉黑的操作，即删除此账号。

3 系统设计

3.1 系统说明

论坛交流系统包括两个基本部分：一个BBS论坛前台（用户），一个论坛后台管理（管理员）。

用户注册成为会员，登录论坛系统后可以发表帖子，回复帖子。用户注册是为了保证用户在系统中的信息安全性，及个性化设置。用户登录之后才拥有相应的权限，例如发表主题、收藏、点赞、回复、修改个人设置。以管理员账号登录论坛系统之后可选择进入后台系统，可以对所以会员进行管理，包括权限的设置，用户角色的设置及删除账号，还可以查看删除帖子及对其进行置顶加精。

3.2 系统功能介绍

3.2.1 基本功能

网站的前台（用户界面）组成由以下几个子系统组成：

1、主页，显示欢迎界面，提供书籍的分类，最新的活动、最新上架的书籍。

2、分类系统，根据用户的选择，显示书籍的分类。

3、查询系统，通过书名或作者名来查询自己需要的书籍。

4、书籍详情，显示书籍的详细信息，如简介、数量、价格等。

5、购物车，与超市的购物车同理，将想要购买的书籍放入购物车，最后再处理购物车里的书籍。

6、注册系统，处理新用户的注册。

7、登录系统，处理老用户登录、购书时的确认和处理用户和管理员的登录。

8、订单详情，显示用户的购买的书籍情况。

如图3-1。

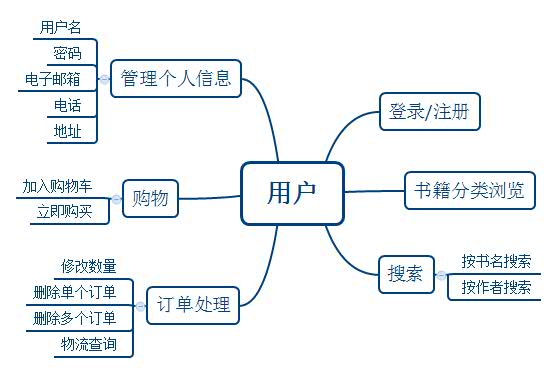


图3-1 网站前台功能图

网站的后台（管理员界面）组成由以下几个子系统组成：

1、图书管理,对所有的图书进行增、删、修改库存等。

2、订单处理，对用户所提交的订单进行处理，即发货。

3、管理员信息修改，对管理员的用户名或密码进行修改。

如图3-2。

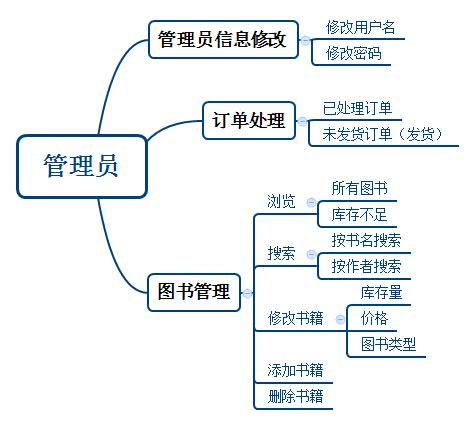


图3-2 网站后台功能图

3.2.2 功能模块列表

各功能模块明细，如表3.1；管理个人信息模块，如表3.2；分类浏览书籍功能模块，如表3.3；信息查询模块，如表3.4；购物车模块，如表3.5；前台订单处理，如表3.6；管理员信息修改功能模块，如表3.7；图书管理功能模块，如表3.8；后台订单处理功能模块，如表3.9。

表3.1 各功能模块明细

|  |  |  |  |
| --- | --- | --- | --- |
| **序号** | **模块类别** | **功能模块** | **备注** |
| 1 | 客户端 | 管理个人信息 | 需要管理端用户验证 |
| 2 | 分类浏览书籍 | 无需验证 |
| 3 | 查询书籍 | 无需验证 |
| 4 | 购物 | 需要验证 |
| 5 | 前台订单处理(结账) | 需要用户模块共同工作 |
| 6 | 管理端 | 图书管理 | 需要管理端用户验证 |
| 7 | 后台订单处理 | 需要管理端用户验证 |
| 8 | 管理员信息修改 | 需要管理端用户验证 |

表3.2 管理个人信息模块

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 用户登录 | 用户输入用户名和密码进行验证 |
| 2 | 用户注册 | 注册用户以登录系统 |
| 3 | 用户信息修改 | 编辑并保存用户注册信息，包括用户名、密码、电子邮箱、电话、地址的修改 |
| 4 | 用户注销 | 单击安全退出即可退出当前登录 |

表3.3 分类浏览书籍功能模块

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 查看同类书籍 | 单击某类书籍，在页面中只会显示这一类的书籍 |
| 2 | 查看全部书籍 | 在分类浏览图书页面，单击图书，查看全部书籍 |

表3.4 信息查询模块

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 按书名查询 | 输入书名，点击搜索进行查询 |
| 2 | 按作者查询 | 输入作者，点击搜索进行查询 |

表3.5 购物车模块

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 商品放入购物车 | 浏览查询商品时单击加入购物车标识即可添加到购物车 |
| 2 | 浏览购物车 | 单击网页右上角的购物车图标，进入购物车 |
| 3 | 取消购物车中商品 | 从购物车中将商品删除 |
| 4 | 编辑购物车的商品 | 更改数量 |
| 5 | 清空购物车 | 将购物车所有商品删除 |
| 6 | 支付订单 | 单击确认支付按钮，即可进行支付 |

表3.6 前台订单处理

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 查看订单状态 | 查看订单状态，确认是否发货 |
| 2 | 结账 | 用户购买完成，结账，需要把购买的商品信息和交易信息保存到数据库 |
| 3 | 取消订单 | 用户可以取消没有处理完的订单 |

表3.7 管理员信息修改功能模块

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 管理员登录 | 管理员输入用户名和密码系统进行验证 |
| 2 | 修改用户名 | 输入欲修改的用户名，单击修改即可 |
| 3 | 修改密码 | 输入欲修改的密码，单击修改即可 |
| 4 | 删除管理员用户 | 单击删除即可删除用户 |
| 5 | 注销登录 | 单击安全退出按钮，即可退出当前登录 |

表3.8 图书管理功能模块

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 查看所有图书 | 单击所有图书按钮即可显示所有图书 |
| 2 | 查看库存不足图书 | 单击库存不足按钮,即可查看所有库存为零的图书 |
| 3 | 按书名搜索图书 | 在搜索框键入书名，单击搜索，即可看到搜索结果 |
| 4 | 按作者搜索图书 | 在搜索框键入作者，单击搜索，即可看到搜索结果 |
| 5 | 修改图书信息 | 可以修改图书的价格、库存、分类，修改完后，单击修改按钮即可保存信息 |
| 6 | 删除图书 | 单击删除按钮删除图书，删除后该图书将不再存在 |
| 7 | 添加图书 | 单击添加图书按钮，进入添加图书页面，添加书籍 |

表3.9 后台订单处理功能模块

|  |  |  |
| --- | --- | --- |
| **序号** | **功能列表** | **功能明细** |
| 1 | 查看订单 | 可以查询已处理订单和未发货订单 |
| 2 | 发货 | 对于未处理订单单击发货按钮即可进行发货处理记录 |

4 数据库设计

整个系统通用一个数据库（“my”）。在该数据库中，建立了4张表。包括admin，user，book，shop表。用于存放整个系统用到的数据。如图4-1。

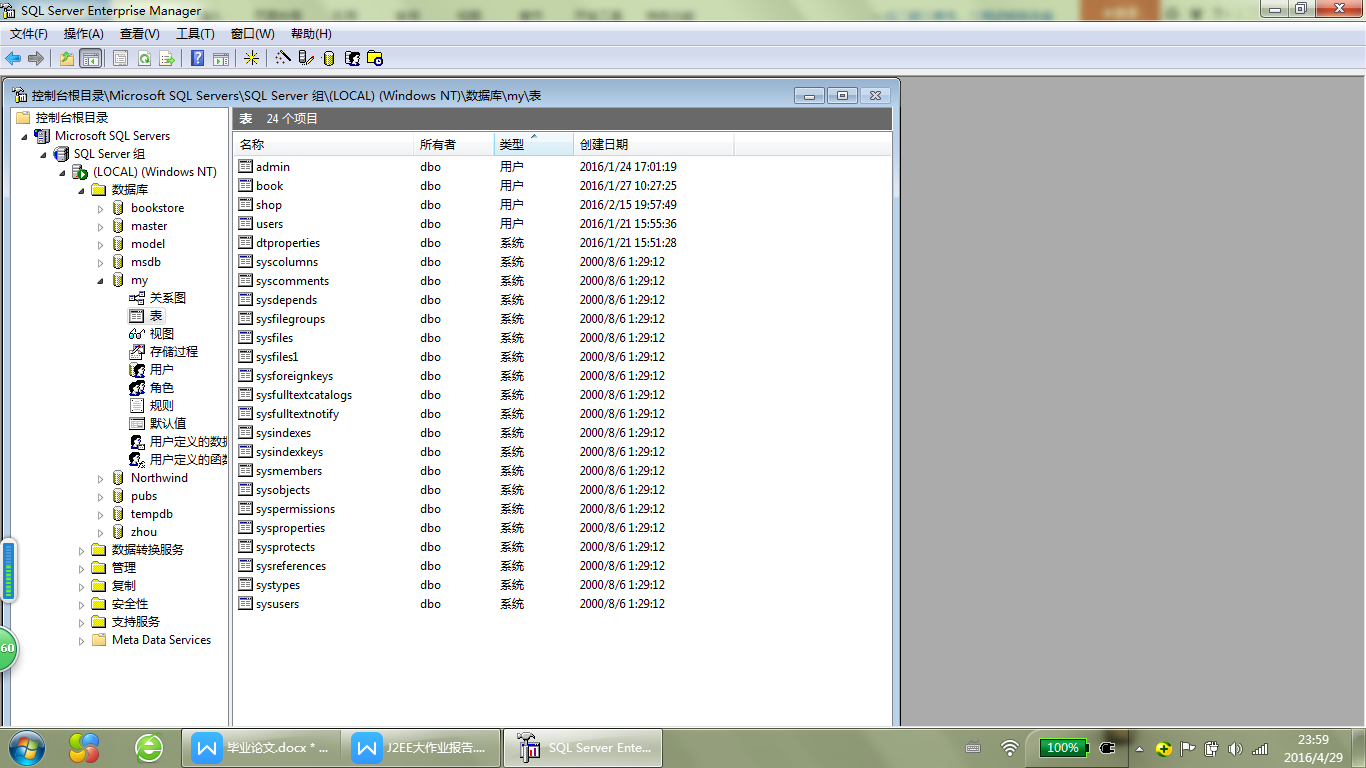


图4-1 数据库总表浏览

admin表，是存放管理员信息的一张表，包括了管理员的ID（userId）主键[8]、管理员名（username）、管理员密码（passwd）。如图4-2。

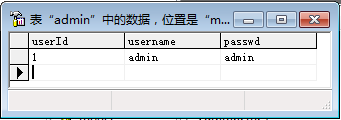


图4-2 admin表

users表，是存放用户信息的一张表，包括了用户的用户ID（userId）主键、用户名（username）、密码（passwd）、邮箱（email）、电话号码（tel）、地址（address）。如下图4-3。

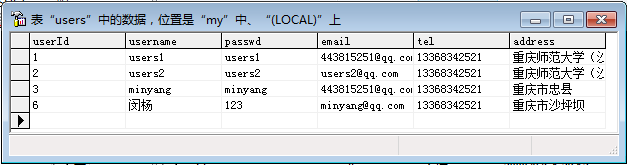


图4-3 users表

book表，是存放书籍信息的一张表，包括书籍的书籍ID（bid）主键、书名（bookname）、价格（price）、作者（writer）、库存量（counts）、简介（intro）、照片（picture）、分类（type）。如下图4-4。



图4-4 book表

shop表，是存放定单信息的一张表，包括订单的点单ID（oid）主键、书籍ID（bid）、书名（bookname）、库存量（counts）、用户购买数量（buynum）、用户名（username）、用户电话号码（tel）、用户地址（address）、发货（power）发货为1未发货为0。如下图4-5。

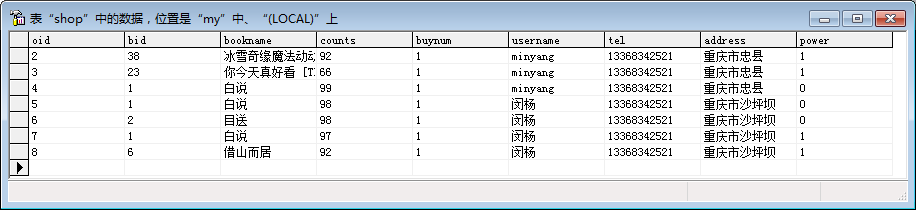


图4-5 shop表设计

5 系统架构分析

Web服务器：Tomcat 6.0

数据库服务器：SQL Server 2000

平台：MyEclipse Profressional 2014

操作系统：win 7

设计模式：MVC

MVC是Model（模型）、View（视图）和Controler（控制）这三个单词的首字母的缩写。MVC是一种设计模式，这个模式认为，一个程序不论是简单还是复杂，从结构上看，都可以分成M、V、C三层[9]，以下便对这三层做一个简单的介绍。

Model数据层即模型，与数据库创建连接，是最核心的一层。它持有所有的数据、状态和程序逻辑[10]，是被控制器和视图所调用的对象，独立于控制器和视图。

View视图层即视图，是用户能够直接看到的用户界面，便于用户的操作。

Controller控制层即控制，位于视图层和数据层中间，是模型与视图之间的枢纽，用户在视图层发出指令，通过控制层所对应的一系列操作，选择需要调用的数据层的程序及数据，然后又通过控制层的处理，输出到视图层，得到用户想要的结果[9]。

以下分别展示的是系统M、V、C层的所有源文件，如图5-1。

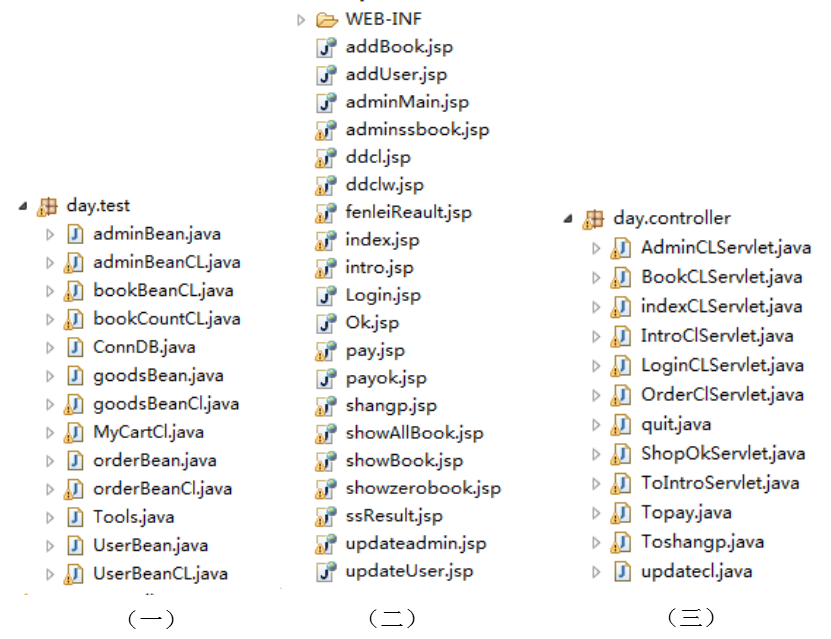


图5-1 M、V、C层源文件

.jsp

.java

.java

xx表

在本项目中的M, V, V规划：其中 表示V层视图代码 ， 表示C层控制代码， 表示M层模型代码， 表示在数据库中所对应的表。图5-2是登录注册的关系图，图5-3是用户信息修改的关系图，图5-4是搜索主流程图，图5-5是购买主流程图，图5-6 是订单查询流程图，图5-7是后台管理图关系图，图5-8是安全退出关系图。

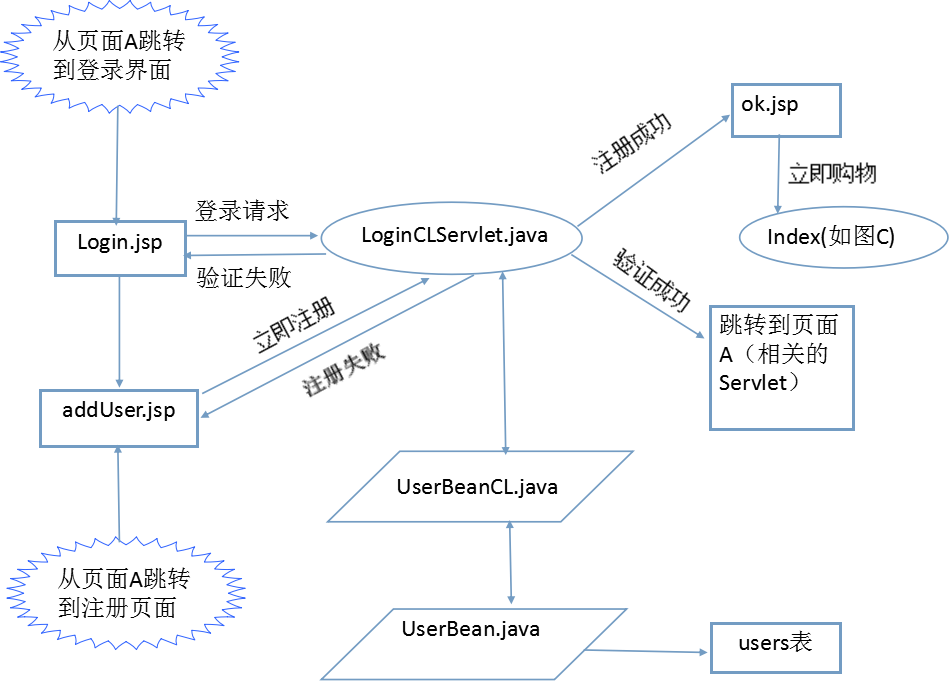


图5-2 登录注册

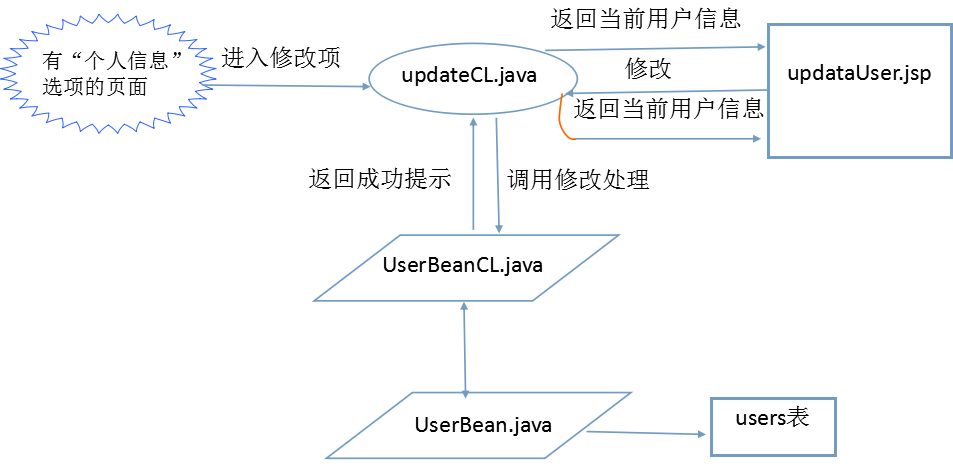


图5-3 用户信息修改

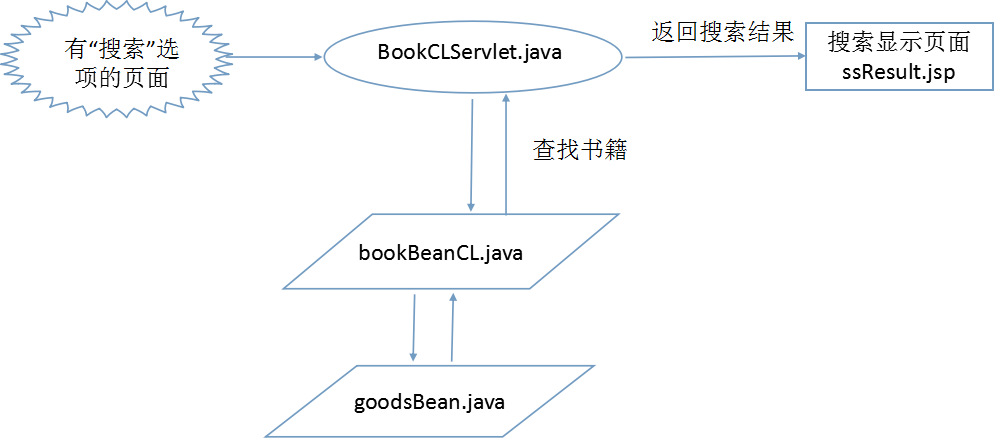


图5-4 搜索主流程

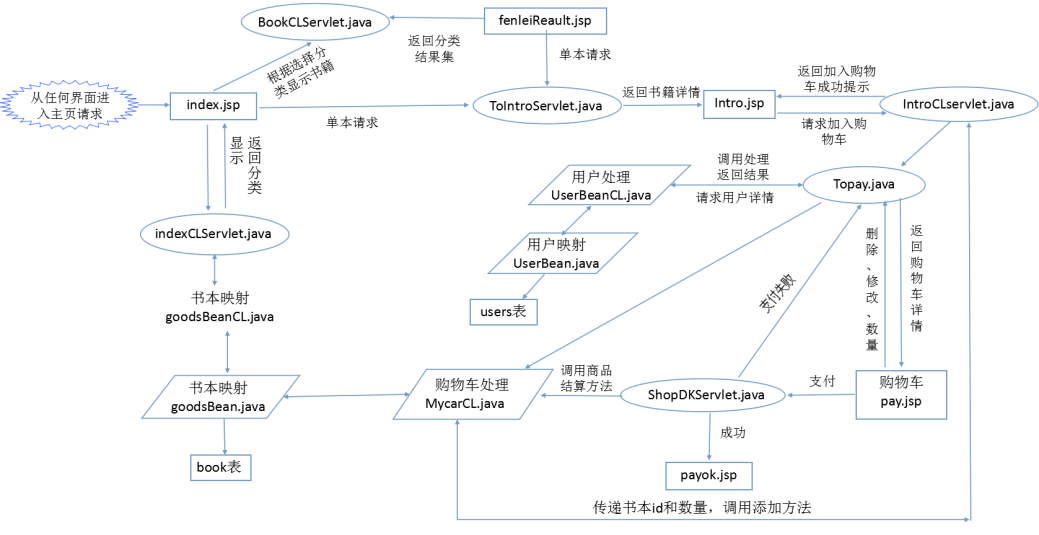


图5-5 购买主流程

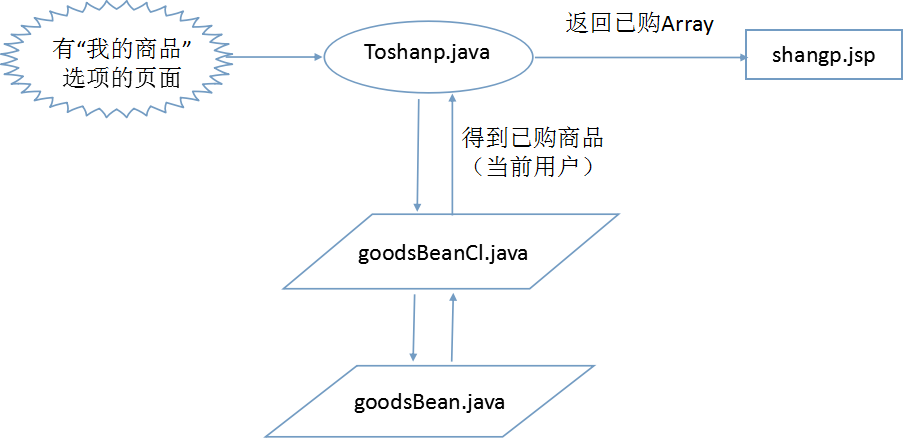


图5-6 订单查询（已登录）

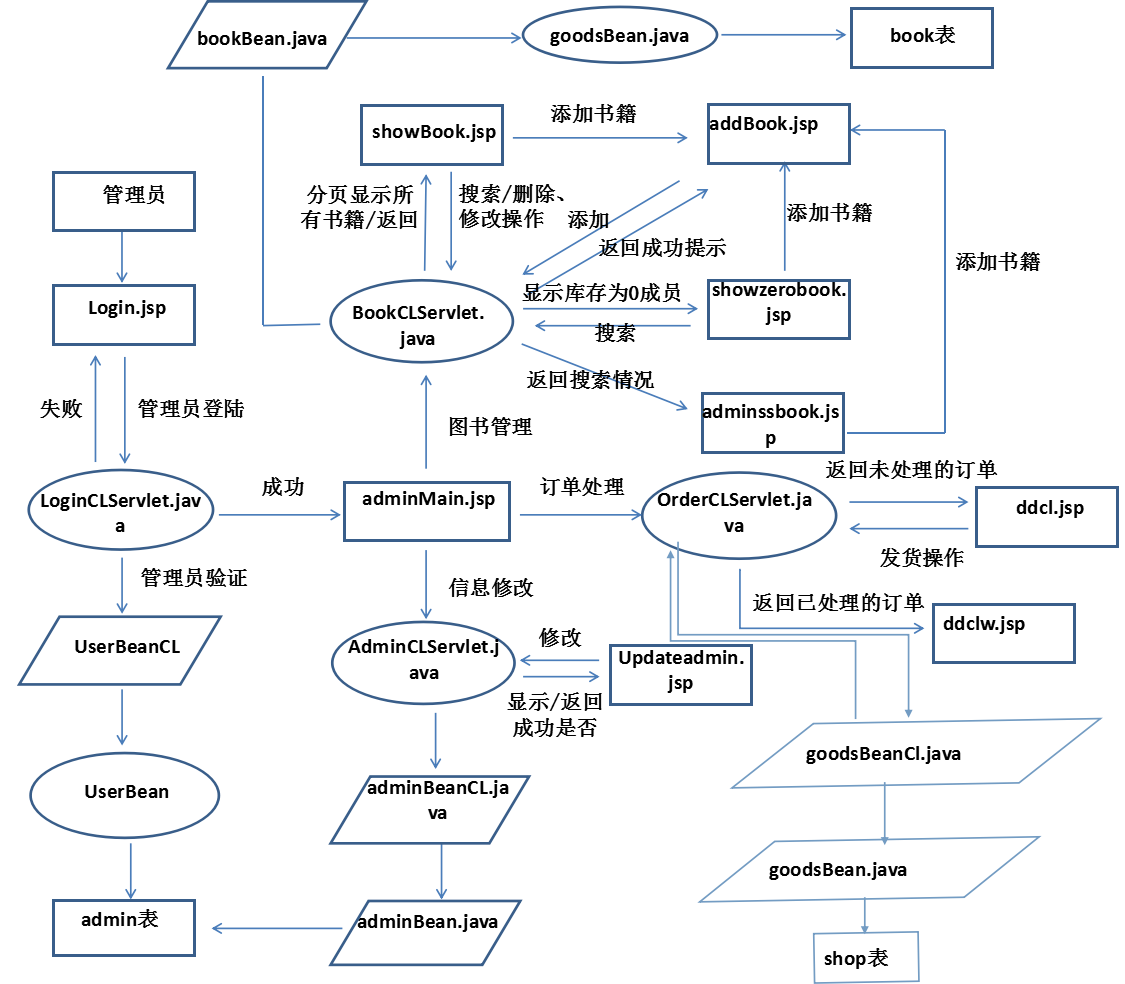


图5-7 后台管理

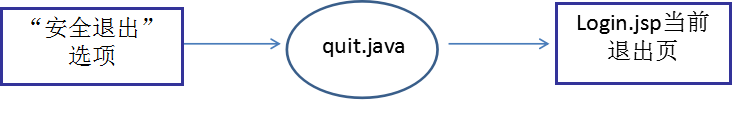


图5-8 安全退出

6 系统界面的实现

6.1 系统前端界面的实现

6.1.1 主界面

在系统的主界面里，如图6-1。最上方是登录和注册的按钮，下方有书籍搜索栏、我的购物车、书籍的分类、活动的介绍、新书上架（取的是book表中的最后5条记录）等板块。

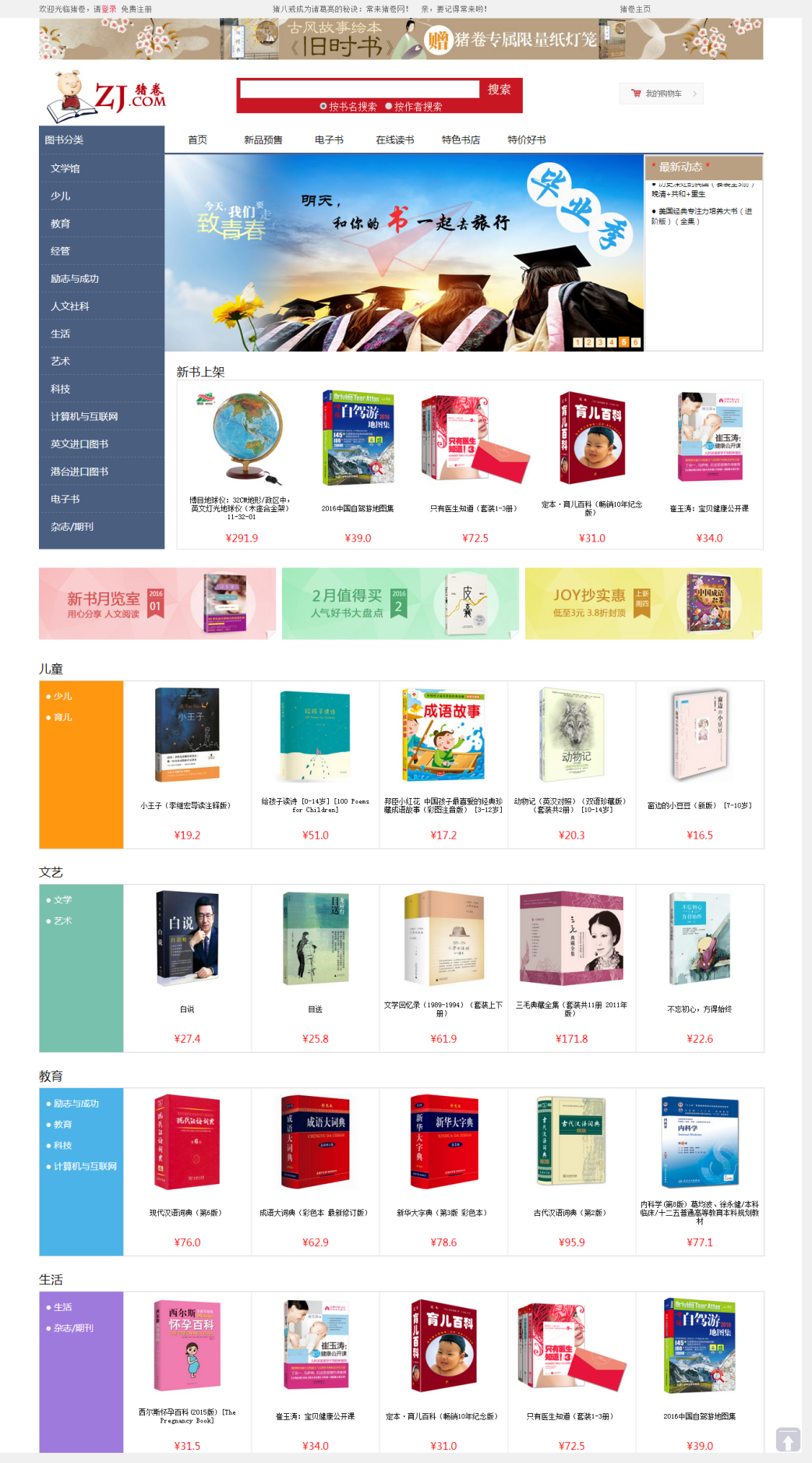


图6-1 主界面

6.1.2 登录界面

在各个界面的页头上，点击登录按钮进入登录界面，如图6-2。在登录界面分为普通用户登录和管理员登录，输入好用户名和密码，点击“登录”按钮进行登录。如果没有登录账号，可以点击“登录”下方的“用户注册”，进行注册。



图6-2 登录界面

6.1.3 注册界面

进入注册界面，输入用户的各种信息，点击“立即注册”按钮进行注册，如图6-3。若已有账号，可点击“立即注册”按钮下的“登录按钮”进行登录。

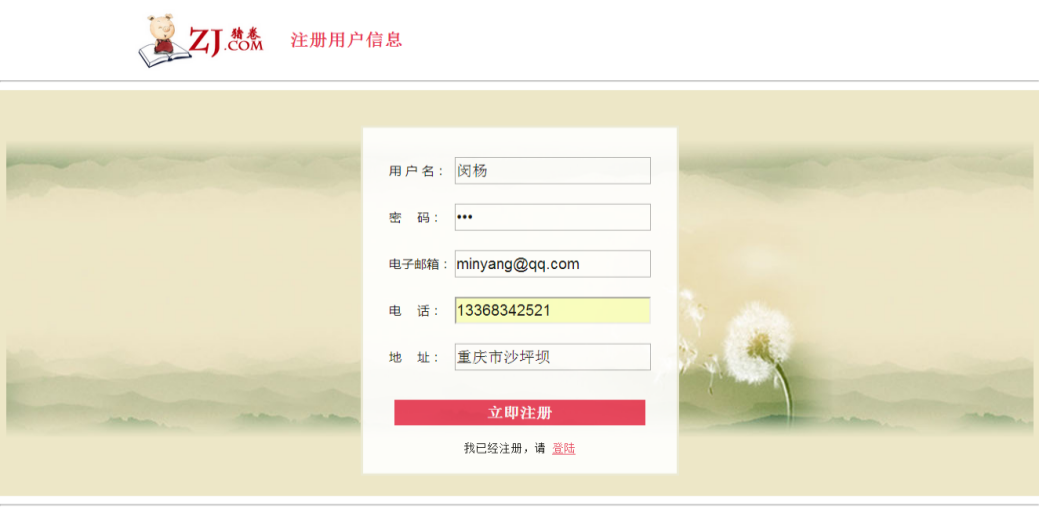


图6-3 注册界面

若注册成功，会弹出注册成功界面，如图6-4。



图6-4 注册成功界面

登录成功后，界面的页头发生了变化，图6-5为登录前的页头，图6-6为登录后的页头。



图6-5 登录前的页头



图6-6 登录后的页头

6.1.4 个人信息界面

登录成功后，可以点击页头右方的“个人信息”按钮，进入个人信息界面，如图6-7，可以查看和修改个人的信息。



图6-7 个人信息界面

6.1.5 搜索结果界面

在任何界面的搜索栏里可以进行按书名或按作者搜索书籍，点击“搜索”按钮，跳转到搜索结果界面。如图6-7。



图6-7 搜索结果界面

6.1.6 书籍分类结果界面

在主页面左方是书籍大的一个分类，当鼠标放到其中一个大分类时会显示出该分类的下拉子分类。例：单击“文学馆”下的“文学”，结果如图6-8。



图6-8 书籍分类结果图

6.1.7 图书详情界面

单击书本图标，查看书籍的详细信息，里面显示了书籍的书名、作者、价格、简介等信息。如图6-9。



图6-9 图书详情界面

6.1.8 购物车界面

在图书详情界面，将书籍加入购物车，若用户没有登录，系统会跳转到登录界面，进行登录操作，然后才能加入购物车。点击“搜索”栏旁的“我的购物车”按钮查看购物车详情，在该界面，可以对书籍订单进行修改、删除等操作。如图6-10。

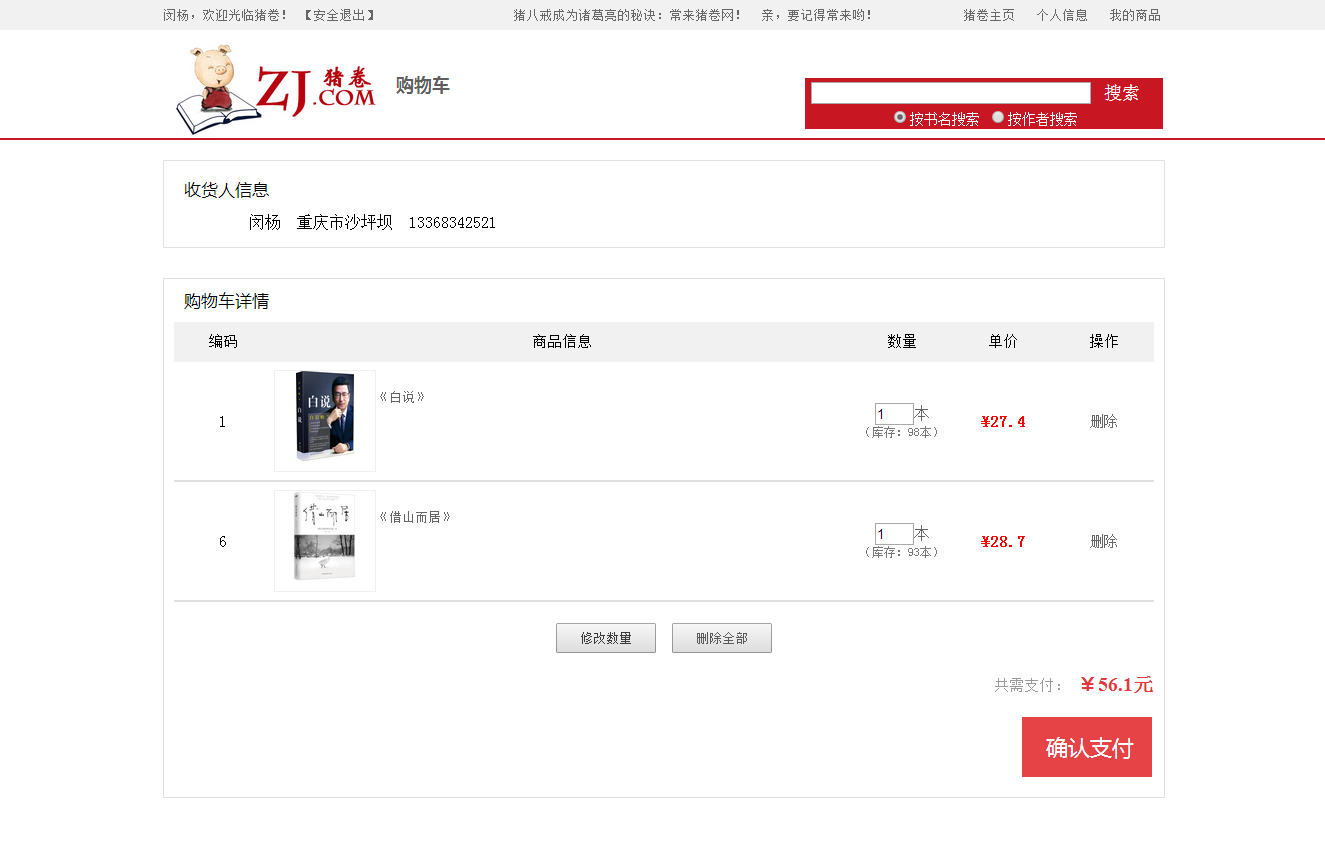


图6-10 购物车界面

6.1.9 订单支付成功界面

单击购物车右下方的“确认支付”按钮，若支付成功，便跳转到订单支付成功界面。如图6-11。



图6-11 订单支付成功界面

6.1.10 我的商品界面

单击页头右方的“我的商品”按钮，查看购买书籍的详细情况。如图6-12。



图6-12 我的商品界面

6.2 系统后台界面的实现

6.2.1 管理员主界面

在登录在界面，以管理员登录，进入管理员主界面，里面有图书管理、订单处理、管理员信息修改和安全退出四个菜单，如图6-13。

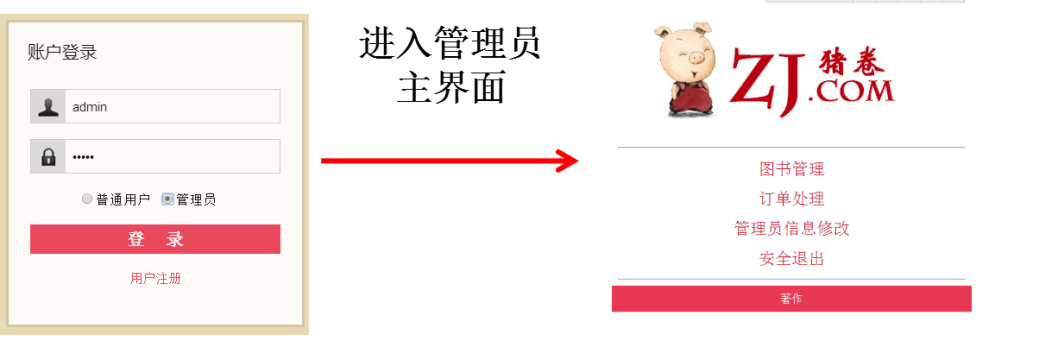


图6-13 管理员主界面

6.2.2 图书管理界面

该界面显示了所有图书的信息，可以对图书的各项信息进行修改、删除，可以搜索图书，添加图书，显示库存不足的图书，如图6-14。



图6-14 图书管理界面

6.2.3 添加图书界面

单击图书管理界面里的“添加图书”按钮，跳转到图书添加界面，进行书籍各项信息的录入，添加书籍。如图6-15。



图6-15 添加图书界面

6.2.4 订单处理界面

单击主界面里的订单处理，跳转到订单处理界面，包括了未处理订单和已处理订单，对未处理订单里的订单进行发货处理，进行完发货处理的订单会出现在已处理订单里。如图6-16。



图6-16 订单处理界面

6.2.5 管理员信息修改界面

单击主界面里的管理员信息修改，跳转到管理员信息修改界面，对管理员的信息进行修改。如图6-17。



图6-17 订单处理界面

7 总结

网上书城最重要的是客户界面的友好度、购买书籍的方便性，我浏览许多的购物网站，学习如何制作出更友好的用户界面，翻阅了大量的书籍和整合大学以来学过的所有知识来解决在设计的过程中遇到的问题。当然，这个系统还有很多的不足，比如用户界面的美观度还可以提高、用户个人信息的管理还可以更完善、后台图书管理的方便性还需提高、没有添加后台管理员的选项等。但我这次设计的网上书城系统做到了最主要和最基本的三项功能（登录、购物、后台），通过本次设计，我意识到了自身的不足，某些知识的缺乏，我希望在我未来的努力中做出更加完善的系统。

参考文献：

[1] 张丹.浅析快递行业发展前景及趋势[M].社科纵横,2012

[2] 史轶男.新媒介环境下网上书店对学术类图书营销的影响.浙江传媒学院学报.2013

[3] 李智勇.网上图书销售系统的设计与实现.电脑学习.2011

[4] [http://blog.sina.com.cn/s/blog\_483a5989010084q2.ht](http://blog.sina.com.cn/s/blog_483a5989010084q2.html" \t "C:/Users/Thinkpad/Desktop/%E7%BD%91%E4%B8%8A%E4%B9%A6%E5%9F%8E%E7%B3%BB%E7%BB%9F%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0/%E3%80%8A%E7%BD%91%E4%B8%8A%E4%B9%A6%E5%9F%8E%E7%B3%BB%E7%BB%9F%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0%E3%80%8B%E6%96%87%E6).网上购书的好处

[5] 邓子云. JSP网络编程从基础到实践（第3版）[M]. 北京:电子工业出版社,2010.4

[6] <http://zhidao.baidu.com/question/389167144.html>.什么是JSP

[7] http://www.docin.com/p-571079733.html.JSP外文文献

[8] 王珊，萨师煊. 数据库系统概论（第四版）[M]. 北京市：高等教育出版社,2006.5

[9] http://blog.csdn.net/xuxiaoping1989/article/details/51200334.谈谈MVC模式及 MVP和MVVM的图示

[10] [http://blog.sina.com.cn/s/blog\_563cc1be0100emym.ht](http://blog.sina.com.cn/s/blog_563cc1be0100emym.html" \t "C:/Users/Thinkpad/Desktop/%E7%BD%91%E4%B8%8A%E4%B9%A6%E5%9F%8E%E7%B3%BB%E7%BB%9F%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0/%E3%80%8A%E7%BD%91%E4%B8%8A%E4%B9%A6%E5%9F%8E%E7%B3%BB%E7%BB%9F%E8%AE%BE%E8%AE%A1%E4%B8%8E%E5%AE%9E%E7%8E%B0%E3%80%8B%E6%96%87%E6).MVC模式浅谈