

**毕业设计（论文）**

题 目 **基于Android的滑雪用具商城的设计与实现**

专 业 软件工程

学　　 号 141110422

学 生 宣国鑫

指 导 教 师 权光日

答 辩 日 期 2018.06.16

摘 要

近年来，移动互联网正迅猛发展。与此同时，人们的生活节奏在不断加快。因此，电商如何能够满足消费者正在改变的生活方式与消费观念以适应趋势已经成为行业内亟待解决的问题。移动端APP，使用户利用碎片时间享受移动购物成为可能，这种方式不仅让用户免除的购物的时间与空间限制，更加方便地享受服务；同时也给了商家全时段的推广的平台，并且可以帮助节约成本。

本文围绕手机购物的核心业务流程，设计并实现了一个C/S架构的滑雪用具商城APP。后台管理系统使用Java EE技术，采用Struts框架进行了页面布局，基于Spring MVC框架实现了业务逻辑控制，使用Hibernate框架完成了数据持久化；前台使用Android移动计算技术，模拟器模拟真实手机。系统后台分为信息管理模块、商品管理模块、订单管理模块、注册用户管理模块；APP端包括注册登录模块，浏览信息模块、购物车模块、订单管理管理模块、信息管理模块；拥有系统管理员、访客用户和注册用户三个角色。

经过测试，该系统在三种角色的互相配合之下，成功地把业务和互联网有机地结合在了一起，让移动购物成为可能，并走近了人们的生活。

关键词：滑雪用具商城；Java EE；Spring MVC；Hibernate；Android

Abstract

In recent years, the mobile internet is rapidly developing. At the same time, the rhythm of people's life is accelerating. Therefore, how e-commerce can satisfy consumers' changing lifestyles and consumer attitudes to adapt to trends has become a pressing problem in the industry. The mobile terminal APP makes it possible for users to enjoy mobile shopping through the use of fragmented time. This approach not only allows users to dispense with the time and space constraints of shopping, but also allows them to enjoy services more conveniently. At the same time, it also gives businesses a platform for full-time promotion, and Help save costs.

This article focuses on the core business process of mobile shopping, designing and implementing a C/S framework ski mall APP. The background management system uses Java EE technology, uses the Struts framework for page layout, implements the business logic control based on the Spring MVC framework, and uses the Hibernate framework to complete the data persistence; the foreground uses Android mobile computing technology and the simulator simulates the real mobile phone. System background is divided into information management module, product management module, order management module, registered user management module; APP includes registration module, browsing information module, shopping cart module, order management module, information management module; has system administrator, Guest users and registered users have three roles. The system, with the cooperation of the three roles, successfully integrated the business and the Internet organically, making mobile shopping possible and approaching people's lives.

**Keywords:** ski mall APP, Java EE, Spring MVC, Hibernate, Android

**目 录**

[摘 要 I](#_Toc516939253)

[Abstract III](#_Toc516939254)

[**第1章 绪 论** 1](#_Toc516939255)

[**1.1** 课题背景和意义 1](#_Toc516939256)

[**1.2** 国内外相关领域开发及应用现状分析 2](#_Toc516939257)

[**1.2.1** 国内现状分析 2](#_Toc516939258)

[**1.2.2** 国内现状分析 2](#_Toc516939259)

[**1.3** 系统开发相关技术 3](#_Toc516939260)

[**1.3.1** Struts2 3](#_Toc516939261)

[**1.3.2** Spring MVC 3](#_Toc516939262)

[**1.3.3** Hibernate 3](#_Toc516939263)

[**1.3.4** Android 4](#_Toc516939264)

[**1.4** 论文的主要研究内容及结构安排 4](#_Toc516939265)

[**第2章 需求分析与总体设计** 6](#_Toc516939266)

[**2.1** 系统需求分析 6](#_Toc516939267)

[**2.1.1** 功能性需求分析 6](#_Toc516939268)

[**2.1.2** 非功能性需求 12](#_Toc516939269)

[**2.2** 系统总体设计 14](#_Toc516939270)

[**2.2.1** 系统总体架构 14](#_Toc516939271)

[**2.2.2** 系统核心业务类设计 15](#_Toc516939272)

[**2.2.3** 系统时序图 16](#_Toc516939273)

[**2.3** 开发环境及开发工具 17](#_Toc516939274)

[**2.4** 本章小结 18](#_Toc516939275)

[**第3章 系统详细设计** 19](#_Toc516939276)

[**3.1** 系统部署设计 19](#_Toc516939277)

[**3.2** 数据库设计 20](#_Toc516939278)

[**3.3** 类的详细设计 22](#_Toc516939279)

[**3.3.1** 类功能描述 22](#_Toc516939280)

[**3.3.2** 部分模块的详细设计 23](#_Toc516939281)

[**3.4** 本章小结 34](#_Toc516939282)

[**第4章 系统实现** 36](#_Toc516939283)

[**4.1** 商品管理模块流程图及界面实现 36](#_Toc516939284)

[**4.1.1** 核心模块流程图 36](#_Toc516939285)

[**4.1.2** 界面实现 37](#_Toc516939286)

[**4.2** 订单管理模块流程图及界面实现 38](#_Toc516939287)

[**4.2.1** 核心模块流程图 38](#_Toc516939288)

[**4.2.2** 界面实现 39](#_Toc516939289)

[**4.3** 浏览商品信息模块流程图及界面实现 40](#_Toc516939290)

[**4.3.1** 核心模块流程图 40](#_Toc516939291)

[**4.3.2** 界面实现 40](#_Toc516939292)

[**4.4** 购物车模块流程图及界面实现 41](#_Toc516939293)

[**4.4.1** 核心模块流程图 41](#_Toc516939294)

[**4.4.2** 界面实现 42](#_Toc516939295)

[**4.5** 用户订单管理模块流程图及界面实现 42](#_Toc516939296)

[**4.5.1** 核心模块流程图 42](#_Toc516939297)

[**4.5.2** 界面实现 43](#_Toc516939298)

[**4.6** 本章小结 43](#_Toc516939299)

[**第5章 系统测试** 44](#_Toc516939300)

[**5.1** 测试概要 44](#_Toc516939301)

[**5.1.1** 测试目标 44](#_Toc516939302)

[**5.1.2** 测试方案 44](#_Toc516939303)

[**5.2** 单元测试 44](#_Toc516939304)

[**5.2.1** 商品管理模块测试 44](#_Toc516939305)

[**5.2.2** 订单管理模块测试 45](#_Toc516939306)

[**5.2.3** 注册登录模块测试 46](#_Toc516939307)

[**5.2.4** 注册登录模块测试 47](#_Toc516939308)

[**5.3** 系统测试 47](#_Toc516939309)

[**5.3.1** 功能测试 48](#_Toc516939310)

[**5.3.2** 性能测试 49](#_Toc516939311)

[**5.4** 本章小结 49](#_Toc516939312)

[**结 论** 50](#_Toc516939313)

[**参考文献** 51](#_Toc516939314)

[**致 谢** 52](#_Toc516939315)

# **绪 论**

本章从项目来源、项目开发的目的和意义、国内外相关领域及应用现状、主要研究内容和组织结构这几个方面对本文的课题和论文的书写做了一个简要的介绍和说明。

## 课题背景和意义

移动互联网的迅速发展，正在逐渐改变人们的生活方式和消费观念。移动设备让人们在生活、工作的间隙，用碎片时间来享受移动购物成为可能。相比PC和传统购物形式而言，移动购物更方便、更快速，可以为用户随时随地提供所需的服务、应用、信息和娱乐，同时满足用户及商家的需求，还能节省大块时间，易被广大网民、尤其是年轻一代所接受。

移动客户端现如今受到了越来越人的喜爱，根据这样的趋势，移动电商APP开发市场存在着巨大的用户跟商业价值，再者移动电商APP开发能让消费者享受更简单的选购体验，受到越来越多的用户的青睐。在未来电子商务发展的应用会成为用户选购产品的重要工具，不仅只是对用户来说比较方便，满足了用户的体验，保证了用户的消费权益，在企业方面，电商APP开发能帮助企业更好的管理企业，企业能通过电商APP与用户进行更深入的互动，提升用户粘性与用户长期的保持联系，同时也推广了品牌，电商APP能为用户创造多种形式的互动，在将来能更好的打造成O2O的商业模式。

在移动客户端迅猛发展的背景下，许多电子商务企业都推出了自己的移动端APP。但是这类APP大多数都是销售大众商品，在许多小众商品的销售方面显得并不是很专业，例如一些小众体育用品如专业滑雪器材及高尔夫器材的销售亦或是烟草酒类等用户群体针对性比较强的商品，而缺陷体现在销售的此类商品的种类质量的欠缺以及与特定用户交流不紧密上。这些小众领域很多时候没有一款针对自己领域的较为专业的APP。于是本次研究希望帮助滑雪器材领域的销售平台开发一款专门销售滑雪用具的APP，可以销售较为专业的滑雪器材，不仅如此，我们希望在这款APP上，商家和滑雪爱好者以及滑雪爱好者之间能够更加紧密的交流，使更多的人热爱这项运动并获得更好更专业的器材与指导，以促进行业的发展。

## 国内外相关领域开发及应用现状分析

### 国内现状分析

据一组市场调研组结合第三方下载平台及电商App的数据显示，手机网购用户已从去年第四季度的12.58%增至今年第一季度的20.42%，环比增长62.32%，其中有90%都安装或使用过APP。以iPhone为代表的手机APP Store(应用程序商店)模式，则引发了app制作软件消费革命，一个平日看来，仅用于通话、发短信的手机，如今已经和PC相提并论。这种和超市差不多的电商app，让更新app软件变得非常简单。自从其诞生的那一刻起，一些相继的电商app应用软件，开始了一场拉锯战。从苹果的iOS到谷歌的Android，针锋相对的app制作软件数量，成为相互竞争的资本。在众多新增加的app之中，电商类app的开发程序，成为使用率高的APP之一。而据统计，在美国，绝大多数B2C平台，甚至个人的网络商铺，都几乎一对一，甚至一对多的推出各自版本的购物电商APP。同时国外的一些国家在电商APP领域还有很大的发展空间，有利条件主要体现在1.经济实力较为强劲2.基础设施水平较高3.城市人口较为密集4.网络普及率较高5.移动设备使用率高6.移动设备在消费者购物过程中拥有较为重要的地位7.消费者社交行为较为活跃。

除此之外，国外在专业行业如体育运动等领域的针对性比较强的小众化电子商务APP出现较多，许多售卖专业产品的商家拥有自己的APP，这也促进了相应领域的发展，同时满足了这类受众的需求。

### 国内现状分析

如今，中国电子商务平台的商城APP，还尚处部署阶段，只有局部有实力的电子商务平台，完成了APP制作开发。中国电商APP制作开发的出现，大大提升了人们在网上消费的频率，在电商app上完成一笔消费，仅需几个简单的步骤，就可以等着电子商务公司为你送货上门，没有了传统PC网上购物空间上的限制，同时也大大节省了时间。而与其它APP应用制作不同的是，通过电商app特色功能节约你的时间。即使你不是购物狂，相信也会被提升自己原本购物频率。  
 电商APP开发虽然容易，但要想获得用户的持续使用，还得使APP制作开发与众不同。据网络调研称，85%的用户会在1个月内将app应用从其手机中删除，而5个月后，应用留存率仅有5%，这对很多电商APP来说无疑是一个噩耗;淘宝无线的数据则显示，移动购物APP软件平均逗留时间不超过10分钟，远低于PC购物时间。这意味着发轫初期的电商APP制作显然挑战更多。

虽然电子商务APP行业面临诸多难题与挑战，国内移动电商应用市场规模仍增长迅速，各大电商企业加紧抢滩移动领域。各大电商企业先后推出Android、iPhone等手机客户端，高调进入移动电商领域，移动电商成为各大电商纷纷瞄准的新战场，像广州、深圳、珠海等珠三角发达地区肯定是首当其冲”。

于此同时，与国外的电子商务APP行业相比，国内小众领域的电子商务APP开发略有欠缺。一些小众领域的受众很难找到一个较为专业的平台获取需要的商品和互相交流。

## 系统开发相关技术

### Struts2

Struts2是一个基于MVC设计模式的Web应用框架，它本质上相当于一个servlet，在MVC设计模式中，Struts2作为控制器(Controller)来建立模型与视图的数据交互。Struts 2是Struts的下一代产品，是在 struts 1和WebWork的技术基础上进行了合并的全新的Struts 2框架。其全新的Struts 2的[体系结构](https://baike.baidu.com/item/%E4%BD%93%E7%B3%BB%E7%BB%93%E6%9E%84)与Struts 1的体系结构差别巨大。Struts 2以WebWork为核心，采用拦截器的机制来处理用户的请求，这样的设计也使得业务逻辑控制器能够与[Servlet](https://baike.baidu.com/item/Servlet)API完全脱离开。

### Spring MVC

Spring MVC属于Spring Frame Work的后续产品已经融合在Spring Web Flow里面。Spring 框架提供了构建 Web 应用程序的全功能 MVC 模块。使用 Spring 可插入的 MVC 架构，从而在使用Spring进行WEB开发时，可以选择使用Spring的Spring MVC框架或集成其他MVC开发框架，如Struts1，Struts2等。系统后台实现采用Spring MVC架构，Spring MVC是一种基于Java的实现了Web MVC设计模式的请求驱动类型的轻量级Web框架，即使用了MVC架构模式的思想，将Web层进行职责解耦，基于请求驱动指的就是使用请求-响应模型，框架的目的就是帮助我们简化开发，Spring MVC简化了Web开发。

### Hibernate

Hibernate是一个开放源代码的对象关系映射框架，它对JDBC进行了非常轻量级的对象封装，它将POJO与数据库表建立映射关系，是一个全自动的orm框架，Hibernate可以自动生成SQL语句，自动执行，使得Java程序员可以随心所欲的使用对象编程思维来操纵数据库。Hibernate可以应用在任何使用JDBC的场合，既可以在Java的客户端程序使用，也可以在Servlet/JSP的Web应用中使用，最具革命意义的是，Hibernate可以在应用EJB的J2EE架构中取代CMP，完成数据持久化的重任。

### Android

Android是基于Linux内核的软件平台和操作系统，是Google 在2007 年11 月5日公布的手机系统平台，早期由Google 开发，后由开放手机联盟（Open Handset Alliance）开发。它采用软件堆层（software stack，又名以软件叠层）的架构，主要分为3 部分。低层以Linux内核工作为基础，只提供基本功能；其他的应用软件则由各公司自行开发，以Java 作为编写程序的一部分。其主要优势有：

1) 开放性：在优势方面，Android平台首先就是其开放性，开发的平台允许任何移 动终端厂商加入到Android联盟中来。

2) 丰富的硬件选择：这一点还是与Android平台的开放性相关，由于Android的开放性，使其硬件的支持较多，故拥有了更多的硬件选择。

3) 不受任何限制的开发商：Android平台提供较广的平台给第三方开发商，因此不受框架束缚。

4) 无缝结合的Google应用：Google公司本身拥有的应用能够完美无缝的与Android 平台结合。

#### 专业可穿戴传感设备

近年来，随着制造执行系统的重要性的逐渐提高，美日欧等工业发达国家越来越重视对制造执行系统技术的研究开发与系统应用。制造执行系统国际协会（1997）在White Paper Number 6 中提出制造执行系统的概念和基本功能。[9]制造执行系统能通过信息的传递，对订单下达到产品完成的整个过程进行优化管理。针对制造执行系统的体系结构，Bruce M R （1995）提出了企业三层体系结构，并指出制造执行系统是位于计划层与控制层之间，任务是将业务系统生产的生产计划传递给生产现场，并将生产现场的信息及时收集、上传、处理。黄永东（2006）指出制造执行系统被广泛应用于航天、汽车、光电、制药、石化等行业。生产管理信息化系统与自动化设备可以连接到相关信息的自动记录和在制品（WIP）的质量控制，统计分析，生产调度，设备和仪器的维修等。程纪元（2007）指出，国外对制造执行系统开发的思路主要有两类：一种是专用解决方案，另一种为集成解决 方案。

## 论文的主要研究内容及结构安排

本次研究拟设计一款基于Android的销售滑雪器材的手机APP，希望这款APP不仅能够提供用户一个购买专业高质量滑雪器材的机会，还给予广大爱好者一个学习技术交流经验的平台，满足专业受众团体提升水平的需求，促进行业的发展。系统包括：一个基于C/S架构的管理商城商品、订单与注册用户的管理端后台，供商城管理员进行一系列管理操作；一个基于Android技术的移动购物APP，供用户进行查看商品以及购物活动。

论文在结构上共分为5章，这5章的具体内容如下所示。

第一章：绪论。主要介绍课题背景和意义、国内外研究现状、主要研究内容和系统开发相关技术等。

第二章：系统分析与总体设计。主要针对系统的功能进行分析和描述，在明确了系统功能的基础上，对系统进行了整体的设计包括功能模块划分、系统架构设计、领域建模、时序图等等。

第三章：系统详细设计。主要依据系统分析和总体设计对系统数据库和核心类图进行了设计。

第四章：系统实现。主要是依据前期的设计对系统各个功能模块流程进行了详细的说明并展示了系统主要的界面。

第五章：系统测试。主要介绍单元测试和集成测试，其中单元测试运用黑盒和白盒测试两种方法进行测试，集成测试运用黑盒测试方法进行测试，通过测试，保证了系统的健壮性同时也说明了系统很好地完成了需求分析中的用户需求。

# **需求分析与总体设计**

本章主要对系统进行分析，包括业务需求、用户需求、功能需求，系统需求及其他需求，并在系统分析的基础上进行系统总体设计。

## 系统需求分析

### 功能性需求分析

#### 功能性需求描述

滑雪用具商城所服务的主要客户群体包括：①滑雪爱好者②滑雪初学者③线下商店经营者。

系统的管理者是商城的管理人员。

针对客户群体，由于其对商品的专业性与细分要求较高，系统推出了以下服务：通过滑雪用具的用途不同将商品进行细致的分类，可以按照用途进行检索；商品简介后有完整的参数展示与大量图片以提供对比的凭据。以此来使用户获得足够的信息以选择最符合自身需求的商品。除此之外，需要满足用户的注册、登陆、查看商品、添加商品到购物车、购物车管理、订单下达、订单管理、用户信息管理等需求。

针对系统管理者，需要满足其登录、查看商品、对商品进行分类、添加商品、查看订单、处理订单、注册用户管理、发布相关信息等需求。

#### 系统用例

系统管理员用例。

系统管理员通过系统管理端管理商城的各类信息，管理商品信息、商品分类信息；管理订单信息，包括接受订单和修改订单状态等；管理用户信息，包括查看信息和删除信息。

图2-1是系统管理员用例图。

该角色具有四个主要功能，发布资讯信息、处理订单、管理商品以及管理用户。



图 2-1 系统管理员用例图

表2-1是系统管理员用例图中的“修改订单状态”用例描述：

表 2-1 修改订单状态用例描述

|  |
| --- |
| 用例编号： User\_1 用例名称：修改订单状态 |
| 简要说明 |
| 后台管理员处理订单 |
| 执行者 |
| 后台管理员 |
| 前置条件 |
| 网络畅通，后台管理员进入管理界面 |
| 后置条件 |
| 提示操作成功，订单状态修改成功 |
| 事件流 |
| 1. 选择相应订单号 2. 查看订单信息 3. 检查订单商品与收货地址 4. 确认无误后修改订单信息 5. 用例结束 |

访客用户用例。

访客用户只拥有在商城中查看信息的功能，包括查看相关资讯和商品信息，但不具备购买权限。

图2-2是访客用户用例图。



图 2-2 访客用户用例图

表2-2是访客用户用例图中“查看商品”用例描述。

表 2-2 查看商品用例描述

| 用例编号： User\_2 用例名称：查看商品 |
| --- |
| 简要说明 |
| 访客用户进行查看商品操作 |
| 执行者 |
| 访客用户 |
| 前置条件 |
| 网络畅通，系统正常开启 |
| 后置条件 |
| 商品商品正常显示 |
| 事件流 |
| 1. 打开手机APP 2. 选择访客用户查看商品 3. 正常浏览商品 |

注册用户用例。

注册用户可以通过手机登录商城，在首次进入商城时，可以以访客身份查看商品信息，但购买商品等操作需要注册登录。注册用户拥有查看商品，购买商品的权限。

图2-3是注册用户用例图。



图 2-3 APP注册用户用例图

表2-3、表2-4以及表2-5是注册用户用例图中“注册登录”、“下单结算”以及“查看订单信息”的用例描述。

表 2-3 注册登录用例描述

|  |
| --- |
| 用例编号： User\_3 用例名称：注册登录 |
| 简要说明 |
| 用户进行注册和登录操作 |
| 执行者 |
| 商城用户 |
| 前置条件 |
| 网络畅通，APP正常开启 |
| 后置条件 |
| 注册成功并登录成功 |
| 事件流 |
| 1. 进入手机APP，点击进入注册界面 2. 输入用户名、真实姓名、联系方式、密码等信息 3. 完成注册 4. 返回登陆界面 5. 输入正确用户名密码 6. 成功登录商城 |

表 2-4 下单结算用例描述

|  |
| --- |
| 用例编号： User\_4 用例名称：下单结算 |
| 简要说明 |
| 商城用户确定订单信息并提交 |
| 执行者 |
| 商城注册用户 |
| 前置条件 |
| 网络畅通 |
| 后置条件 |
| 订单下达成功 |
| 事件流 |
| 1. 登录APP 2. 添加商品进入购物车 3. 将购物车商品提交生成订单 4. 查看订单确认信息 5. 填写收货地址 6. 点击确认按钮，提交订单 7. 用例结束 |

表 2-5 查看订单信息用例描述

|  |
| --- |
| 用例编号： User\_5 用例名称：查看订单信息 |
| 简要说明 |
| 商城用户查看下单的处理结果 |
| 执行者 |
| 商城注册用户 |
| 前置条件 |
| 网络畅通，成功登入 |
| 后置条件 |
| 无 |
| 事件流 |
| 1. 进入APP 2. 登录指定用户 3. 查看已下达订单的处理结果 4. 如商家已发货可以点击收货确认收货 5. 用例结束 |

#### 滑雪用具商城系统领域模型

经过详细的用例分析可知，滑雪用具商城系统涉及到订单，商品分类，购物车信息，管理员信息，商品信息，商品图片信息，用户信息共7个概念实体，实体的主要属性和相互之间的关系如图 2-5所示。

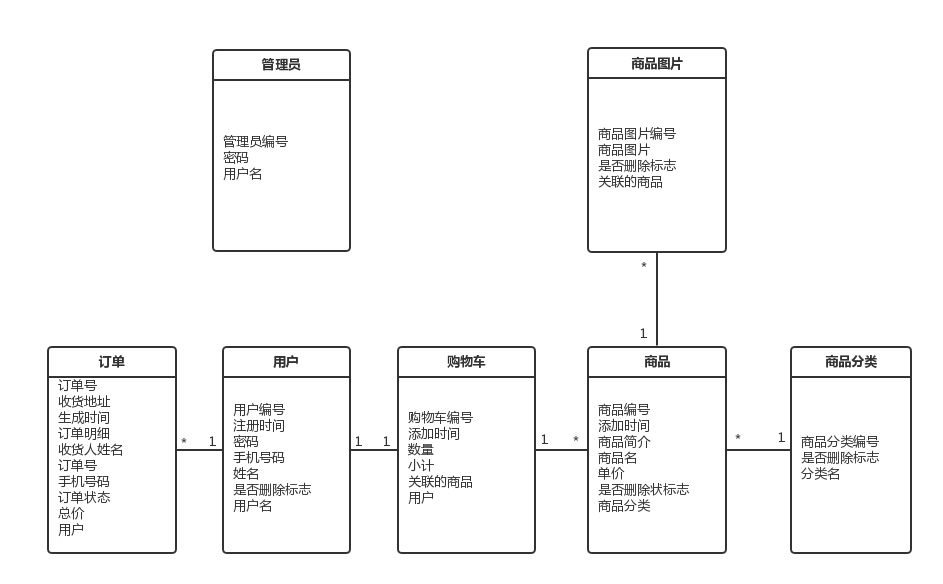


图 2-5 滑雪用具商城系统领域模型图

#### 功能模块划分

根据相关用例，如2-6所示，系统可以分为服务端管理员模块和APP端用户模块；

后台管理模块：该模块主要是针对整个系统进行管理，包含相关信息资讯发布子模块，商品管理子模块，订单管理子模块，注册用户管理子模块。

APP用户模块：分为登录注册子模块，浏览信息子模块，购物车管理子模块，订单管理子模块，个人订单管理子模块。

图 2-6 功能模块图



服务端相关信息资讯发布与管理子模块可以方便商城管理者发布和滑雪相关的信息，如雪场信息和天气信息；商品管理子模块可以供管理者对商品分类以及商品进行管理；订单管理子模块供管理员进行订单的查看与状态的更改；注册用户管理子模块可以供管理员查看用户信息并删除用户。APP端有注册登录子模块供访客用户注册以及注册用户登录；浏览信息或商品子模块可以供用户查看商品信息或其他资讯；购物车模块与订单管理模块供用户管理选购的商品；信息管理模块供用户管理个人信息如姓名及收货地址等。

### 非功能性需求

#### 测试环境

硬件环境如下：

(1)CPU：2.9 GHz Intel Core i5

(2)内存：8 GB 2133 MHz LPDDR3

(3)硬盘：SSD 128G，硬盘 128G

(4)网络：校园局域网 100Mbps

软件环境如下：

1. 操作系统：macOS sierra 10.12.5
2. JDK：JDK 1.8 64位
3. 服务器：Tomcat 7.0
4. 数据库：MySQL 5.7

#### 性能需求

滑雪用具商城系统在上述运行环境下需要满足以下性能要求：

1. 支持50个用户同时登录系统并进行操作。
2. 所有的用户请求应在3秒内做出响应。

#### 可靠性需求

滑雪用具商城系统数据库操作应开启事务处理，以保证在故障出现可以进行事务回滚；浮点数操作精确到小数点后两位四舍五入，统一标准；404和500异常需要进行捕获，并重定向到异常页面。

#### 易用性需求

滑雪用具商城系统应提供友好美观的用户图形界面，简单易上手的交互性，尽可能的方便用户进行操作，并且需要帮助中心模块，对手机购物的流程和注意事项进行解释说明，以满足系统管理员，访客用户，注册用户各个使用层面的需求。

#### 安全性需求

滑雪用具系统的用户密码需要进行MD5加密处理，数据库中存储密码的密文形式；同时商城使用者中存在不同权限的用户，不同的权限能够使用不同的功能。

#### 兼容性需求

滑雪用具商城系统是C/S架构的网站，需要考虑到兼容常用的浏览器，具体的说明如下：

1. 兼容IE 8 及其以上的IE系列浏览器和其他IE内核的浏览器
2. 兼容Firefox浏览器
3. 兼容Chrome内核的浏览器
4. 兼容Safari浏览器
5. 兼容Opero浏览器

#### 可移植性需求

内测试阶段项目的部署测试在本机进行，公测和商用阶段项目都要部署到服务器上，并且Web服务和数据库服务器要分开，运行环境可以方便地从window系统迁移到linux系统上。

## 系统总体设计

### 系统总体架构

本文提出的系统架构如下图2-7所示，阐明了滑雪用具商城系统的整体工作架构。本系统总体架构设计主要采用了客户端和服务器（C/S）模型以及客户层-服务层-数据层模式的三层架构。

图 2-7 系统架构图



### 系统核心业务类设计

根据业务需求，我们得到了如图2-8所示的核心业务类图。

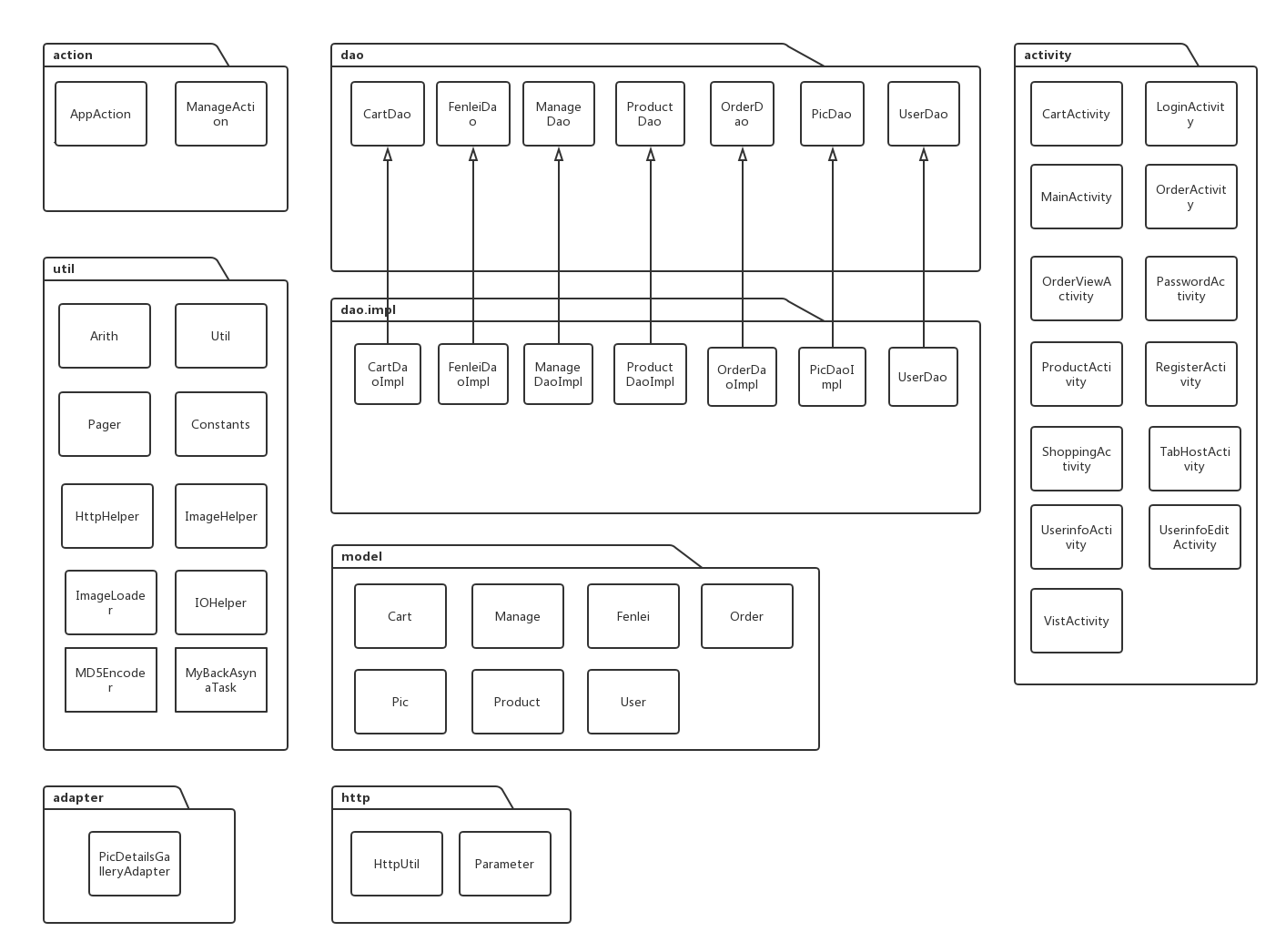


图 2-8 系统核心业务类图

### 系统时序图

根据需求以及类的核心业务类的分析，我们得到如图2-9 至 2-12 的系统时序图。



图2-9 注册功能时序图



图2-10 登录功能时序图



图2-11 购物功能时序图

## 开发环境及开发工具

系统开发环境如表2-8所示。

表2-8系统开发环境表

|  |  |
| --- | --- |
| **设备名称** | **规格** |
| 型号 | MacBook Pro (13-inch, 2016, Four Thunderbolt 3 Ports) |
| 处理器 | 2.9 GHz Intel Core i5 |
| 内存容量 | 8 GB |
| 外存容量 | 256GB |
| 操作系统 | macOS Sierra |

系统开发语言及开发工具如表2-9所示。

表2-9系统开发语言及工具表

|  |  |
| --- | --- |
| **名称** | **版本号** |
| 系统开发语言 | Java，Html，JavaScript |
| 系统开发框架 | SSH |
| 系统开发工具 | MyEclipse 2017CI |
| 数据库操作工具 | MySQL workbench |
| 数据库 | MySQL |

## 本章小结

本章对滑雪用具商城系统进行了全面的分析，从明确的业务需求到功能性需求、非功能性需求，采用了用例图的方式加以描述，同时对系统进行了模块化的分析。在需求分析的基础上对系统进行了总体设计，完成了系统架构，系统时序图。最后对系统开发环境及开发工具做了说明。

# **系统详细设计**

本章的主要工作是在系统设计的基础上进行详细设计，首先对数据库进行详细设计，然后针对系统分析的内容进行类的详细设计。

## 系统部署设计

根据系统设计分析，我们得到了如图3-1所示的系统部署设计。其中，服务器架设本机，系统为mac OS，采用的数据库为My SQL 5.7。系统包括商城管理员网页后台和Android APP端前台以及数据库。

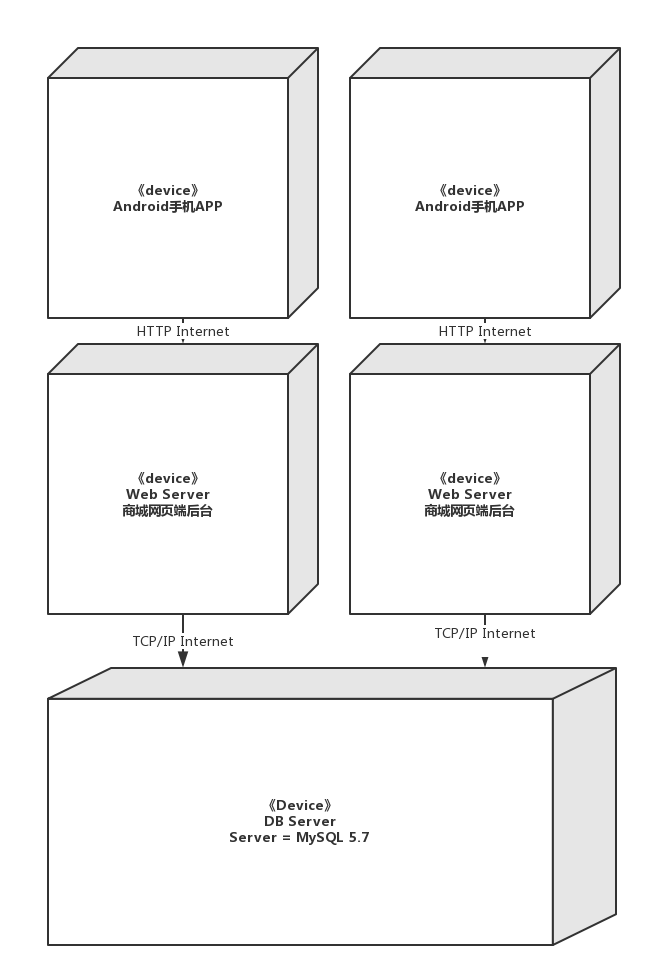


图 3‑1 系统部署图

## 数据库设计

滑雪用具商城系统的数据库共包含7张表，表3-1为数据库表设计目录。

表 3-1 数据库表目录

|  |  |  |
| --- | --- | --- |
| 表名 | CODE | 说明 |
| 订单表 | T\_Cart | 存储每一个用户的订单 |
| 分类表 | T\_Fenlei | 存储商品的分类信息 |
| 购物车表 | T\_Manage | 存储购物车中的商品信息 |
| 管理员表 | T\_Order | 存储管理员信息 |
| 商品表 | T\_Pic | 存储各类商品信息 |
| 商品图片表 | T\_Product | 存储商品图片信息 |
| 用户表 | T\_User | 存储商城用户信息 |

表3-2至表3-8为数据库表的详细设计。

表 3-2 订单表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 类型 | 长度 | 是否为空 | 备注 |
| id | 主键 | int | 11 | 否 | 主键、自增 |
| address | 收货地址 | varchar | 256 | 否 |  |
| createtime | 生成时间 | varchar | 256 | 否 |  |
| details | 订单明细 | text | 256 | 否 |  |
| name | 收货人姓名 | varchar | 256 | 否 |  |
| orderid | 订单号 | varchar | 256 | 否 |  |
| phone | 手机号码 | varchar | 256 | 否 |  |
| status | 订单状态 | varchar | 256 | 否 |  |
| totalprice | 总价 | double | 256 | 否 |  |
| userid | 用户 | int | 11 | 否 | 外键 |

表 3-3 分类表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 类型 | 长度 | 是否为空 | 备注 |
| id | 主键 | int | 11 | 否 | 主键、自增 |
| fenleilock | 是否删除标志 | int | 11 | 否 | 0正常，1删除 |
| name | 分类名 | varchar | 256 | 否 |  |

表 3-4 购物车表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 类型 | 长度 | 是否为空 | 备注 |
| id | 主键 | int | 11 | 否 | 主键、自增 |
| createtime | 添加时间 | varchar | 256 | 否 |  |
| sl | 数量 | int | 11 | 否 |  |
| total | 小计 | double | 256 | 否 |  |
| productid | 关联的商品 | int | 11 | 否 | 外键 |
| userid | 用户 | int | 11 | 否 | 外键 |

表 3-5 管理员表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 类型 | 长度 | 是否为空 | 备注 |
| id | 主键 | int | 11 | 否 | 主键、自增 |
| password | 密码 | varchar | 256 | 否 |  |
| username | 用户名 | varchar | 256 | 否 |  |

表 3-6 商品图片表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 类型 | 长度 | 是否为空 | 备注 |
| id | 主键 | int | 11 | 否 | 主键、自增 |
| imgpath | 商品图片 | varchar | 256 | 否 |  |
| piclock | 是否删除标志 | int | 11 | 否 | 0正常，1删除 |
| productid | 关联的商品 | int | 11 | 否 | 外键 |

表 3-7 商品表字典

| 字段名 | 含义 | 类型 | 长度 | 是否为空 | 备注 |
| --- | --- | --- | --- | --- | --- |
| id | 主键 | int | 11 | 否 | 主键、自增 |
| createtime | 添加时间 | varchar | 256 | 否 |  |
| content | 商品简介 | text | 256 | 否 |  |
| name | 商品名 | varchar | 256 | 否 |  |
| price | 单价 | double | 256 | 否 |  |
| productlock | 是否删除状态 | int | 11 | 否 | 0正常，1删除 |
| fenleiid | 商品分类 | int | 11 | 否 |  |

表 3-8 用户表字典

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 字段名 | 含义 | 类型 | 长度 | 是否为空 | 备注 |
| id | 主键 | int | 11 | 否 | 主键、自增 |
| createtime | 注册时间 | varchar | 256 | 否 |  |
| password | 密码 | varchar | 256 | 否 |  |
| phone | 手机号码 | varchar | 256 | 否 |  |
| truename | 姓名 | varchar | 256 | 否 |  |
| userlock | 是否删除标志 | int | 11 | 否 | 0正常，1删除 |
| username | 用户名 |  |  |  |  |

图3-2为数据库表的关系模型图。

该图描述了数据库表之间的关系，订单表与用户表相关联，一个用户对应多个订单；用户表与购物车表关联，一个用户对应一个购物车；购物车表与商品表相关联，购物车对应多种商品；商品表与分类表关联，一个商品对应一个分类；除此之外，商品表和商品图片表相关联。

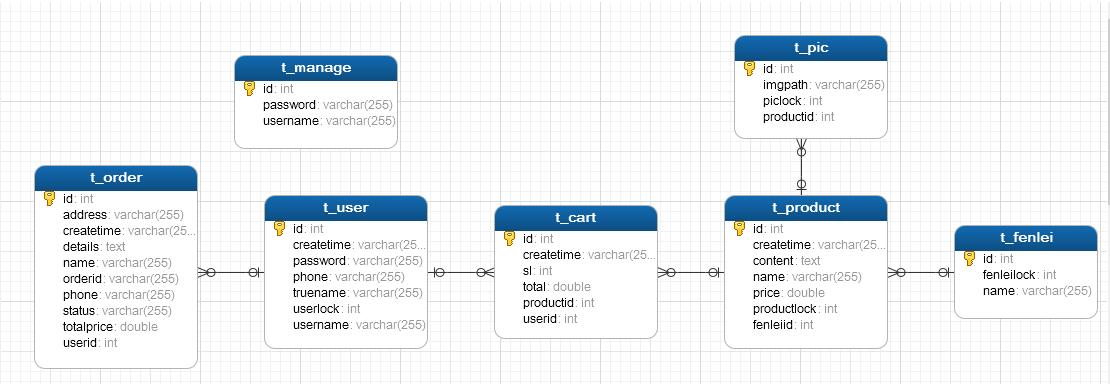


图 3-2 数据库关系模型图

## 类的详细设计

### 类功能描述

根据系统设计，我们得到了核心业务类的设计，表3-9为系统核心业务类的功能描述。设计包括action，dao，dao.impl，model，util五个包，action中包含系统实现类，dao包含系统数据库操作类，dao.impl包括接口的实现类，model为封装数据类，util为工具类。

表 3-9系统核心业务类功能描述

|  |  |  |
| --- | --- | --- |
| 包 | 类 | 类功能 |
| action | AppAction | App功能实现类 |
| ManageAction | 后台管理端功能实现类 |
| dao | CartDAO | 购物车表数据持久类 |
| FenleiDAO | 分类表数据持久类 |
| ManageDAO | 管理员表数据持久类 |
| OrderDAO | 订单表数据持久类 |
| PicDAO | 图片表数据持久类 |
| ProductDAO | 商品表数据持久类 |
| UserDAO | 用户表数据持久类 |
| dao.impl | CartDAOImpl | 购物车表实体类 |
| FenleiDAOImpl | 分类表实体类 |
| ManageDAOImpl | 管理员表实体类 |
| OrderDAOImpl | 订单表实体类 |
| PicDAOImpl | 图片表实体类 |
| ProductDAOImpl | 商品表实体类 |
| UserDAOImpl | 用户表实体类 |
| model | Cart | 购物车数据封装类 |
| Fenlei | 分类数据封装类 |
| Manage | 管理员数据封装类 |
| Order | 订单数据封装类 |
| Pic | 图片数据封装类 |
| Product | 商品数据封装类 |
| UserDAO | 用户数据封装类 |
| util | Arith | 算数运算工具类 |
| Pager | 分页工具类 |
| Util | 文件工具类 |

### 部分模块的详细设计

#### 商品管理模块的详细设计

商品管理与分类模块功能概述：进入购物商城管理系统，点击商品管理，进入商品信息管理界面，可以添加新商品，查询商品，修改、删除、查看商品信息。

商品管理设计表如图3-10所示；

表 3-10 商品管理设计表

|  |  |
| --- | --- |
| 功能名称：商品管理 | |
| 模块名：商品管理 | |
| 输入数据：无 | 输出数据：商品信息 |
| 处理过程：  1.通过String name = request.getParameter("name");  String fenlei = request.getParameter("fenlei");  从页面获取查询的选项   1. 通过判断语句if(name!=null &&!"".equals(name))、if(fenlei!=null &&!"".equals(fenlei))判断值是否为空。 2. 当不为空时则通过StingBuffer 来叠加组装查询的sql语句。 3. 通过productDao.selectBeanList方法来调用数据库的查询方法来查到符合条件的记录 4. 通过request.setAttribute方法把查询到的记录传到显示的页面。 5. 同时通过this.setUrl方法指定传回到具体指定的是哪个JSP页面。 6. 通过这样实现对数据查询功能的实现，同时在该商品列表页面定义了对商品信息的增加，修改，删除等操作。 | |

商品管理接口设计表如图3-11所示；

表 3-11 商品管理接口设计表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | | | |
| 方法名称 | productlist() | | | | | |
| 参数列表 | 参数名称 | 别名 | | 类型 | | 备注 |
| name | 商品名 | | String | | 商品名 |
| fenlei | 商品分类名 | | String | | 商品分类名 |
| 返回值 | 字符串，以判断商品显示结果 | | | | | |
| 相关对象描述 | Product：封装商品信息的类  属性：商品分类（int fenleiid）、商品名（String name）、单价（double price）、添加时间（String createtime） | | | | | |
| 错误列表 | 错误代码 | | 错误概述 | | 详细说明 | |
| 查找商品返回为空 | | 商品不存在 | | 商品不存在 | |
| 概要逻辑说明 | 商品查找成功后显示在页面上，失败返回错误信息 | | | | | |
| 返回数据示例 | SUCCESS:”success” | | | | | |

删除商品设计表如图3-12所示；

表 3-12 删除商品设计表

|  |  |
| --- | --- |
| 程序名称：删除商品 | |
| 模块名：商品信息管理 | |
| 输入数据：无 | 输出数据：商品删除结果 |
| 处理过程：  进入商品信息管理界面，通过String id = request.getParameter("id")获取查询权限。  在商品信息列表的操作栏中点击删除按钮，调用productDao.updateBean方法更新商品信息  点击确定，系统提示“提交成功”  writer.print("<script language='javascript'>alert('提交成功'); | |

删除商品接口设计表如图3-13所示；

表 3-13 删除商品接口设计表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | | | |
| 方法名称 | Productdelete | | | | | |
| 参数列表 | 参数名称 | 别名 | | 类型 | | 备注 |
| id | 商品编号 | | String | | 商品编号 |
| 返回值 | 字符串，以判断删除是否成功 | | | | | |
| 相关对象描述 | Product：封装商品信息的类 | | | | | |
| 错误列表 | 错误代码 | | 错误概述 | | 详细说明 | |
| 查找商品返回为空 | | 商品不存在 | | 商品不存在 | |
| 概要逻辑说明 | 点击删除按钮，提示用户是否删除，确定则返回删除成功提示，取消则删除失败 | | | | | |
| 返回数据示例 | SUCCESS:”success” | | | | | |

查询商品设计表如图3-14所示；

表 3-14 查询商品设计表

|  |  |
| --- | --- |
| 程序名称：查询商品 | |
| 模块名：商品信息管理 | |
| 输入数据：商品名、商品分类 | 输出数据：商品信息 |
| 处理过程：  进入商品信息管理页面，输入商品名和商品分类  点击查询按钮，系统自动检索是否存在符合查询条件的商品  如果存在则显示商品信息，如果不存在则返回为空 | |

查询商品接口设计表如图3-15所示；

表 3-15 查询商品接口设计表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口地址 | POST：ManageAction | | | | | |
| 方法名称 | 查询商品操作 | | | | | |
| 参数列表 | 参数名称 | 别名 | | 类型 | | 备注 |
| id | 商品编号 | | String | | 商品编号 |
| 返回值 | 字符串，以判断是否查询到符合条件的商品信息 | | | | | |
| 相关对象描述 | Product：封装商品信息的类 | | | | | |
| 错误列表 | 错误代码 | | 错误概述 | | 详细说明 | |
| 查询为空 | | 没有符合查询条件的商品 | | 没有符合查询条件的商品 | |
| 概要逻辑说明 | 通过输入商品名称和商品分类，点击查询按钮，若存在符合条件的商品，则返回该商品的具体信息，若不存在符合条件的商品，则返回为空。 | | | | | |
| 返回数据示例 | 符合查询条件的商品信息 | | | | | |

添加商品分类设计表如图3-16所示；

表 3-16 添加商品分类设计表

|  |  |
| --- | --- |
| 程序名称：添加商品分类 | |
| 模块名：商品分类信息管理 | |
| 输入数据：商品分类名 | 输出数据：商品分类名 |
| 处理过程：  进入商品分类管理页面，通过String name = request.getParameter("name")获取查询选项，点击添加新分类按钮。  输入要添加的商品分类名，并通过新增javaBean后设置Bean的属性：Fenlei bean = new Fenlei();bean.setName(name)。  若输入不为空，点击提交按钮，则成功添加新的商品分类。提示提交成功。  writer.print("<script language='javascript'>alert('提交成功') | |

添加商品分类接口设计表如图3-17所示；

表 3-17 添加商品分类接口设计表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 接口地址 | POST：ManageAction | | | | | |
| 方法名称 | FenleiAdd | | | | | |
| 参数列表 | 参数名称 | 别名 | | 类型 | | 备注 |
| name | 商品分类名 | | String | | 商品分类名 |
| 返回值 | 字符串，以判断是否查询到符合条件的商品信息和添加成功 | | | | | |
| 相关对象描述 | Fenlei：封装分类信息的类  属性：分类id（int id）、是否删除标志（int fenleilock）、分类名（String name） | | | | | |
| 错误列表 | 错误代码 | | 错误概述 | | 详细说明 | |
| 输入为空 | | 输入为空 | | 输入为空 | |
| 概要逻辑说明 | 点击添加商品新分类按钮，输入需要添加的商品分类名，点击提交按钮，提示添加成功 | | | | | |
| 返回数据示例 | SUCCESS:”success” | | | | | |

添加商品设计表如图3-18所示；

表 3-18 添加商品设计表

|  |  |
| --- | --- |
| 程序名称：添加商品 | |
| 模块名：商品信息管理 | |
| 输入数据：商品分类名、商品名、销售单价、商品简介信息 | 输出数据：新商品的信息 |
| 处理过程：   1. 管理员进入到商品分类管理界面，然后点击左侧的添加新商品按钮，跳转到添加商品信息的界面。 2. 在添加商品信息的界面，管理员在商品分类名，商品名，销售单价，商品简介信息填写此商品的有关信息。 3. 点击提交按钮，即可将此商品的信息添加到商品列表中。 | |

添加商品接口设计表如图3-19所示；

表 3-19 添加商品接口设计表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | |
| 方法名称 | ProductAdd | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | 备注 |
| name | 商品名 | String | 商品名 |
| fenlei | 分类名 | String | 分类名 |
| info | 商品信息 | String | 商品简介信息 |
| price | 商品价格 | String | 商品价格 |
| 返回值 | 字符串，以返回操作结果 | | | |
| 相关对象描述 | Product：封装商品信息的类  属性：商品名（String name）、分类名（String fenlei）、商品信息（String info）、商品价格（String price）。 | | | |
| 概要逻辑说明 | 商品信息管理中的添加新商品操作，点击左侧的添加新商品按钮，即可出现添加商品信息的界面。 | | | |
| 返回数据示例 | SUCCESS:”success” | | | |

修改商品设计表如图3-20所示；

表 3-20 修改商品设计表

|  |  |
| --- | --- |
| 程序名称：修改商品信息 | |
| 模块名：商品信息管理 | |
| 输入数据：商品分类名、商品名、销售单价、商品简介信息 | 输出数据：修改后商品的信息 |
| 处理过程：  1.管理员进入到商品分类管理界面，然后点击修改商品按钮，跳转到修改商品信息的界面。  2.在修改商品信息的界面，管理员在商品分类名，商品名，销售单价，商品简介信息填写此商品的有关信息。   1. 点击提交按钮，即可将修改后商品的信息添加到商品列表中。 | |

修改商品接口设计表如图3-21所示；

表 3-21 修改商品接口设计表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | |
| 方法名称 | ProductUpdate | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | 备注 |
| name | 商品名 | String | 商品名 |
| fenlei | 分类名 | String | 分类名 |
| info | 商品信息 | String | 商品简介信息 |
| price | 商品价格 | String | 商品价格 |
| 返回值 | 字符串，以返回操作结果 | | | |
| 相关对象描述 | Product：封装商品信息的类  属性：商品名（String name）、分类名（String fenlei）、商品信息（String info）、商品价格（String price）。 | | | |
| 概要逻辑说明 | 商品信息管理中的修改商品信息操作，点击修改按钮，即可出现修改商品信息的界面。 | | | |
| 返回数据示例 | SUCCESS:”success” | | | |

查看商品设计表如图3-22所示；

表 3-22 查看商品设计表

|  |  |
| --- | --- |
| 程序名称：查看商品信息 | |
| 模块名：商品信息管理 | |
| 输入数据：商品分类名、商品名、销售单价、商品简介信息 | 输出数据：修改后商品的信息 |
| 处理过程：  1.管理员进入到商品分类管理界面，然后点击查看商品按钮，跳转到修改商品信息的界面。  2.在查看商品信息的界面，管理员查看商品分类名，商品名，销售单价，商品简介信息。 | |

查看商品接口设计表如图3-23所示；

表 3-23 查看商品接口设计表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | |
| 方法名称 | ProductUpdate3 | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | 备注 |
| name | 商品名 | String | 商品名 |
| fenlei | 分类名 | String | 分类名 |
| info | 商品信息 | String | 商品简介信息 |
| price | 商品价格 | String | 商品价格 |
| 返回值 | 字符串，以返回到商品信息管理界面 | | | |
| 相关对象描述 | Product：封装商品信息的类  属性：商品名（String name）、分类名（String fenlei）、商品信息（String info）、商品价格（String price）。 | | | |
| 概要逻辑说明 | 商品信息管理中的查看商品信息操作，点击查看按钮，即可出现查看商品信息的界面。 | | | |
| 返回数据示例 | SUCCESS:”success” | | | |

商品图片管理设计表如图3-24所示；

表 3-24 商品图片管理设计表

|  |  |
| --- | --- |
| 程序名称：删除商品图片管理和添加图片管理 | |
| 模块名：商品信息管理 | |
| 输入数据：商品图片信息 | 输出数据：修改后商品图片的信息 |
| 处理过程：  1.管理员进入到商品分类管理界面，然后点击商品图片管理按钮，跳转到添加和删除商品图信息的界面。  2.在删除商品图片的界面，点击删除操作，即可删除图片信息。  3.在添加商品图片的界面，点击浏览按钮，即可查看图片列表里的图片信息，任何点击上传，任何点击提交按钮，即可将新的图片信息上传到商品信息列表。 | |

商品图片管理接口设计表如图3-25所示；

表 3-25 商品图片管理接口设计表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 接口地址 | POST：ManageAction | | | |
| 方法名称 | PicList | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | 备注 |
| pageinfo | 商品图片信息 | String | 商品图片信息 |
| 返回值 | 字符串，以返回到商品信息管理界面 | | | |
| 相关对象描述 | Piclist：图片信息的类  属性：图片信息（pageinfo） | | | |
| 概要逻辑说明 | 商品信息管理中的商品图片管理操作，点击商品图片管理按钮，即可出现删除商品图片和添加商品图片的界面。 | | | |
| 返回数据示例 | SUCCESS:”success” | | | |

#### 用户订单管理模块的详细设计

功能概述：管理员进入订单管理页面后可以看到前台客户端传送过来的订单信息，包括订单下达的用户账户、生成订单号、生成时间、收货人姓名、地址、订单状态、总价。除此之外管理员可以进行发货操作以及查看订单详细信息操作。

显示订单信息设计表如图3-26所示；

表 3-26 显示订单信息设计表

|  |  |
| --- | --- |
| 功能名称：显示订单信息 | |
| 模块名：用户订单管理 | |
| 输入数据：无 | 输出数据：商城用户订单信息 |
| 处理过程：   1. 管理员进入系统点击左侧订单管理按钮进入订单管理页面即order.jsp。 2. orderlist.jsp调用manageAction类中的orderlist方法进行处理。 3. 通过request.getParameter方法从页面获取查询的选项。 4. 通过条件语句分别判断下这个值是否为空，当不为空时则通过StingBuffer 来叠加组装查询的sql语句。最后，在通过request.setAttribute方法把查询到的记录传到显示的页面。同时通过this.setUrl("product/productlist.jsp"); 5. 当sql语句组装好了之后，，通过orderDao.selectBeanList方法来调用数据库的查询方法来查到符合条件的记录。 6. 通过request.setAttribute方法把查询到的记录传到页面并显示。 7. 通过this.setUrl方法传回到指定的jsp页面。 | |

显示订单信息接口设计表如图3-27所示；

表 3-27 显示订单信息接口设计表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | | |
| 方法名称 | Orderlist | | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | | 备注 |
| username | 用户名 | String | | 用户名 |
| orderid | 订单号 | Int | | 订单号 |
| status | 状态 | String | | 订单状态 |
| 返回值 | 字符串，以判断订单显示结果 | | | | |
| 相关对象描述 | Order：封装订单信息的类  属性：用户账户（String name）、生成订单号（int id）、生成时间（String createtime） 收货人姓名（String name）、地址（String address）、订单状态（String status）、总价（double totalPrice） | | | | |
| 错误列表 | 错误代码 | 错误概述 | | 详细说明 | |
| 查找订单返回为空 | 订单不存在 | | 订单不存在 | |
| 概要逻辑说明 | 订单查找成功后显示页面，失败返回错误信息 | | | | |
| 返回数据示例 | SUCCESS:”success” | | | | |

处理订单信息设计表如图3-28所示；

表 3-28 处理订单信息设计表

|  |  |
| --- | --- |
| 功能名称：处理订单信息 | |
| 模块名：用户订单管理 | |
| 输入数据：无 | 输出数据：处理后订单状态 |
| 处理过程：   1. 管理员进入系统orderlist.jsp页面点击发货按钮。 2. orderlist.jsp调用manageAction类中的orderdelete方法进行发货处理。 3. 通过orderDao.selectBean方法及选中订单的id使用查询语句获取相应的javaBean。 4. 通过bean.setStatus方法修改订单状态。 5. 通过orderDao.updateBean来更新javaBean并更新数据库中的信息。 6. 返回页面一个操作结果的提示信息。 | |

处理订单信息接口设计表如图3-129所示；

表 3-29 处理订单信息接口设计表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | | |
| 方法名称 | Orderdelete | | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | | 备注 |
| orderid | 订单号 | Int | | 订单号 |
| 返回值 | 字符串，以判断订单处理结果 | | | | |
| 相关对象描述 | Order：封装订单信息的类  属性：生成订单号（int id）、订单状态（String status） | | | | |
| 错误列表 | 错误代码 | 错误概述 | | 详细说明 | |
| 发货失败 | 订单不存在 | | 抛出异常 | |
| 概要逻辑说明 | 发货成功后显示处理结果，失败返回错误信息 | | | | |
| 返回数据示例 | 发货成功后向页面返回“提交成功” | | | | |

查看订单详情设计表如图3-30所示；

表 3-30 查看订单详情设计表

|  |  |
| --- | --- |
| 功能名称：查看订单详情 | |
| 模块名：用户订单管理 | |
| 输入数据：无 | 输出数据：处理后订单状态 |
| 处理过程：   1. 管理员进入系统orderlist.jsp页面点击查看订单详情按钮。 2. orderlist.jsp调用manageAction类中的orderupdate3方法进行处理。 3. 通过orderDao.selectBean方法及选中订单的id使用查询语句获取相应的javaBean。 4. 将Bean中保存的数据通过request.setAttribute方法进行在页面上的设置。 5. 通过this.setUrl方法跳转至orderupdate3.jsp页面显示详单结果。 | |

查看订单详情接口设计表如图3-31所示；

表 3-31 查看订单详情接口设计表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 接口地址 | POST：manageAction | | | | |
| 方法名称 | orderupdate3 | | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | | 备注 |
| orderid | 订单号 | Int | | 订单号 |
| 返回值 | 字符串，以详单显示结果 | | | | |
| 相关对象描述 | Order：封装订单信息的类  属性： 用户账户（String name）、生成订单号（int id）、生成时间（String createtime） 收货人姓名（String name）、地址（String address）、订单状态（String status）、总价（double totalPrice） | | | | |
| 错误列表 | 错误代码 | 错误概述 | | 详细说明 | |
| 查找订单失败 | 订单不存在 | | 提示错误信息 | |
| 概要逻辑说明 | 操作成功后显示订单详细结果，失败返回错误信息 | | | | |
| 返回数据示例 | SUCCESS:”success” | | | | |

#### 注册登录模块的详细设计

用户在登录的界面输入自己的用户名和密码，并选择自己的账号类型，点击登录按钮。若用户名和密码匹配，则进入app主界面；若密码不正确，则显示提示信息。

用户登录设计表如图3-32所示；

表 3-32 用户登录设计表

|  |  |
| --- | --- |
| 程序名称：用户登录 | |
| 模块名：用户管理 | |
| 输入数据：用户名 密码 | 输出数据：登录信息 |
| 处理过程：   1. 用户进入首页后输入用户名密码，点击提交按钮 2. 将用户名和密码存入表单中并提交给服务器端进行处理。 3. 服务器端查询数据库，并返回结果给app端 4. App根据返回结果显示主界面或者提示错误信息 | |

用户登录实现表如图3-33所示；

表 3-33 用户登录实现表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实现类 | LoginActivity | | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | | 备注 |
| username | 用户名 | String | | 用户名 |
| password | 密码 | String | | 用户密码 |
| 数据存储 | SharedPreferences，用于数据交换 | | | | |
| 相关对象描述 | Consumer：封装用户信息的类  属性：用户ID（int id）、用户名（varchar username）、姓名（varchar truename） 手机号码（varchar phone）、密码（varchar password）等属性。 | | | | |
| 错误列表 | 错误代码 | 错误概述 | | 详细说明 | |
| 查找用户返回为空 | 用户不存在 | | 用户不存在 | |
| 概要逻辑说明 | 登录成功后进入app主界面，错误返回错误信息 | | | | |
| 返回数据示例 | SUCCESS:”success”,INPUT:”input” | | | | |

#### 购物车模块的详细设计

功能概述：用户登陆之后可以查询商品的详细信息，进而可以选择是否进行购买。

表 3-34 购买商品设计表

|  |  |
| --- | --- |
| 程序名称：购买商品 | |
| 模块名：购物车模块 | |
| 输入数据：用户名、手机号、收货地址 | 输出数据：具体订单 |
| 处理过程：   1. 用户登陆后跳转只tabHostActivity，用户可以查看具体商品信息 2. 用户选中一件商品进行查看，app向服务器端发送请求，获取商品的信息，并且app端跳转至orderaddActivity 3. 用户填写相关信息后，确认购买，app向服务器端发送请求，并根据返回结果判断是否购买成功 | |

表 3-35 购买商品实现表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 实现类 | shoppingActicity | | | | |
| 参数列表 | 参数名称 | 别名 | 类型 | | 备注 |
| showList | 商品 | List | | 商品 |
| selType | 类别 | Bool | | 商品类别 |
| 数据存储 | SharedPreferences，用于数据交换 | | | | |
| 相关对象描述 | 添加时间（createtime）、商品简介（content）、商品名称（name）、  单价（price） | | | | |
| 错误列表 | 错误代码 | 错误概述 | | 详细说明 | |
| 无 | 无 | | 无 | |
| 概要逻辑说明 | 购买成功后加入购物车 | | | | |
| 返回数据示例 | SUCCESS:”success”,DEFAULT:”default” | | | | |

## 本章小结

根据上一章对系统需求和总体设计的描述，本章主要对数据库和核心功能类进行了设计，详细说明了完成系统核心功能需要的数据库支持以及定义了业务逻辑层所需要的核心功能类及完成各项操作的方法，为系统实现提供了更加详细的技术支持。

# **系统实现**

本章主要依据前期的需求和设计对系统进行实现，系统实现主要介绍房屋短租系统的核心功能流程图以及系统的主要界面。

## 商品管理模块流程图及界面实现

商品管理模块包括商品分类管理子模块、商品细节管理子模块和图片管理子模块。该模块在系统后台由管理员使用，管理员登录后可进入系统管理商品信息。

### 核心模块流程图

图4-1为商品管理模块的操作流程图。

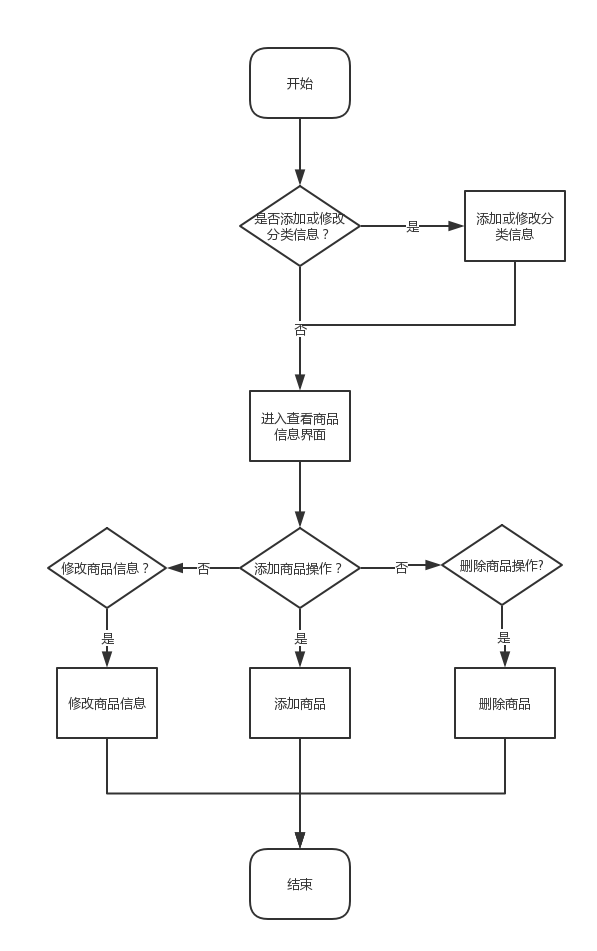


图 4-1 商品管理模块操作流程图

### 界面实现

图4-2至图4-4为商品管理模块的界面实现的部分截图。



图 4-2商品管理模块界面



图 4-3商品管理模块界面

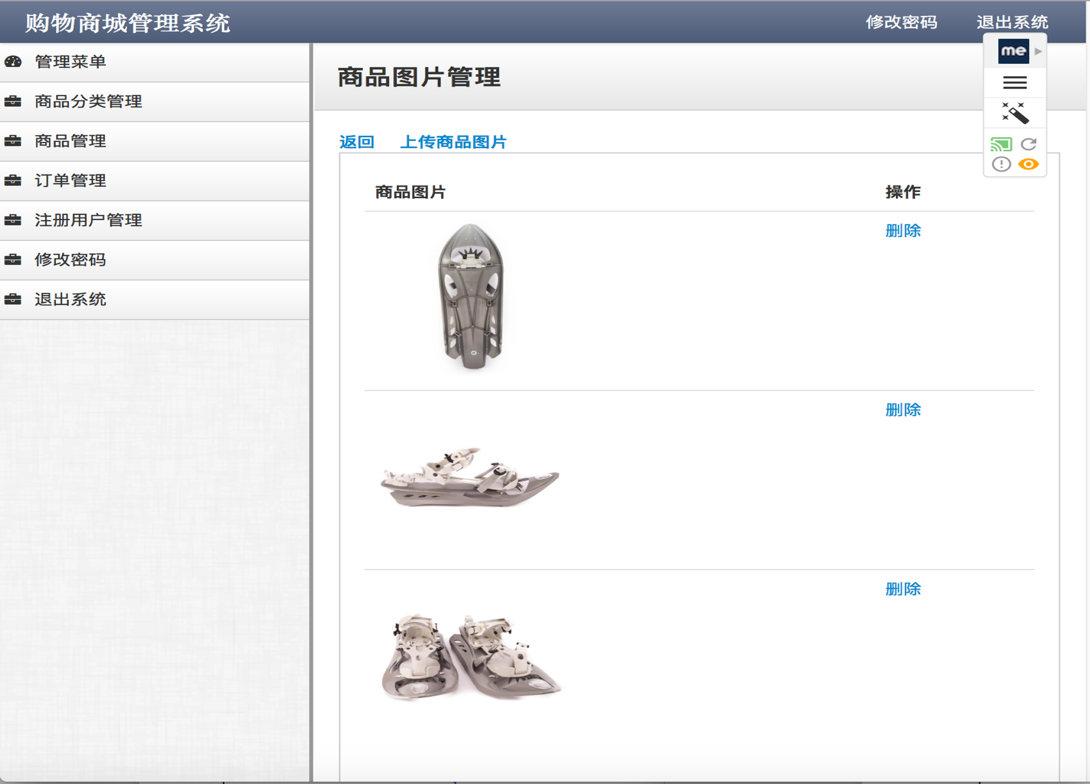


图 4-4商品管理模块界面

## 订单管理模块流程图及界面实现

### 核心模块流程图

图4-5订单管理模块操作流程图。

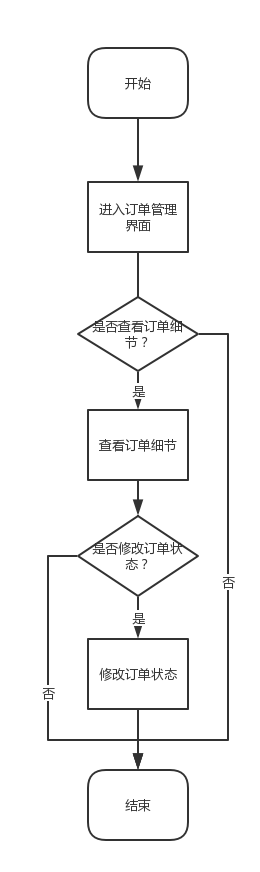


图 4-5 订单管理模块操作流程图

### 界面实现

图4-6至图4-7为订单管理模块的界面实现的部分截图。

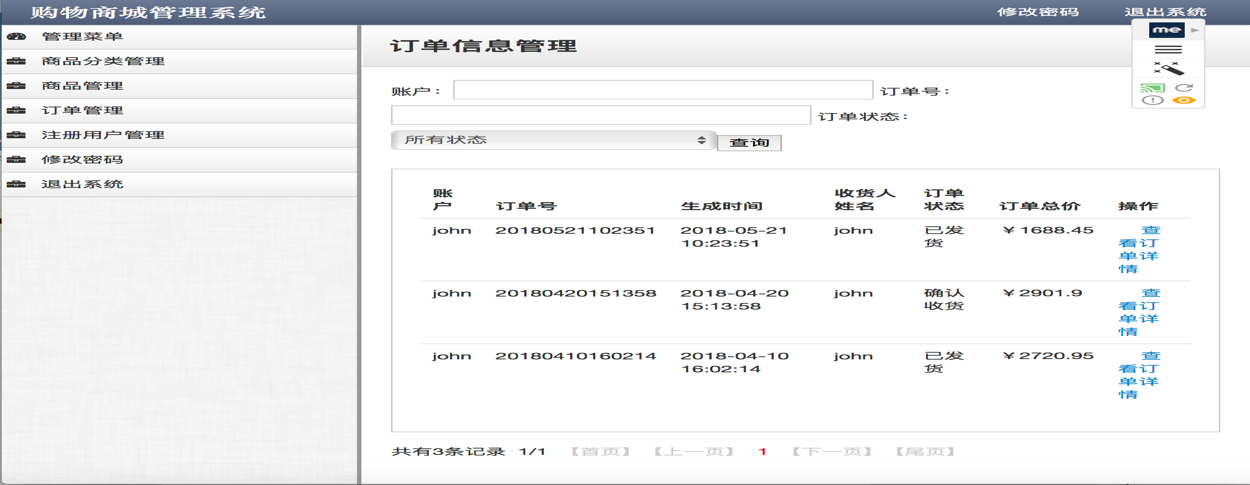


图 4-6 订单管理模块界面

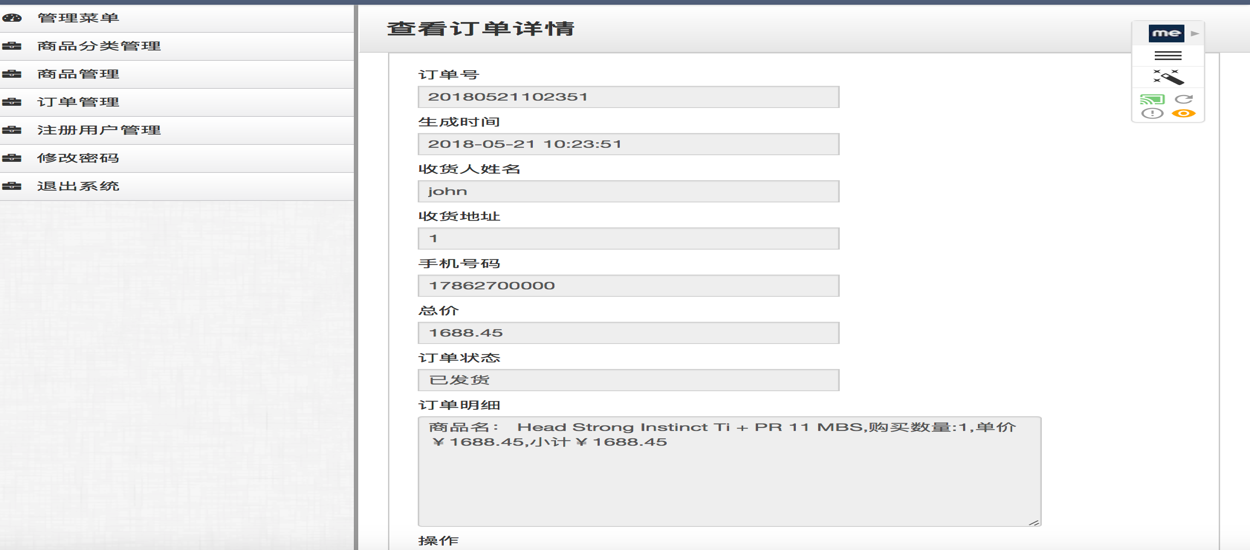


图 4-7 订单管理模块界面

## 浏览商品信息模块流程图及界面实现

浏览商品信息模块包括浏览分类商品信息子模块和浏览全部商品信息子模块。该模块在APP端中使用，访客用户和注册用户都可使用此功能。

### 核心模块流程图

图4-8为浏览商品信息模块的操作流程图。

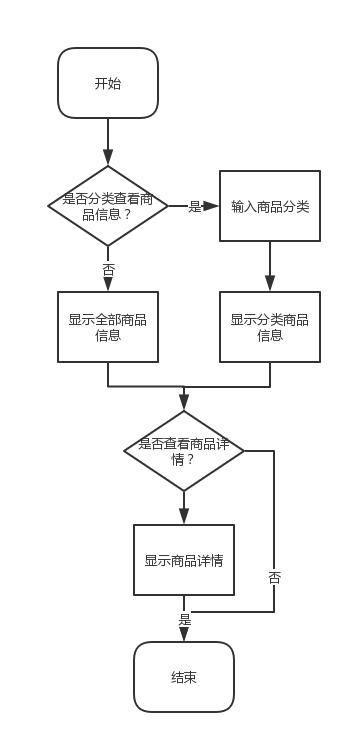


图 4-8 浏览商品信息模块流程图

### 界面实现

图4-9为浏览商品信息模块的部分截图。

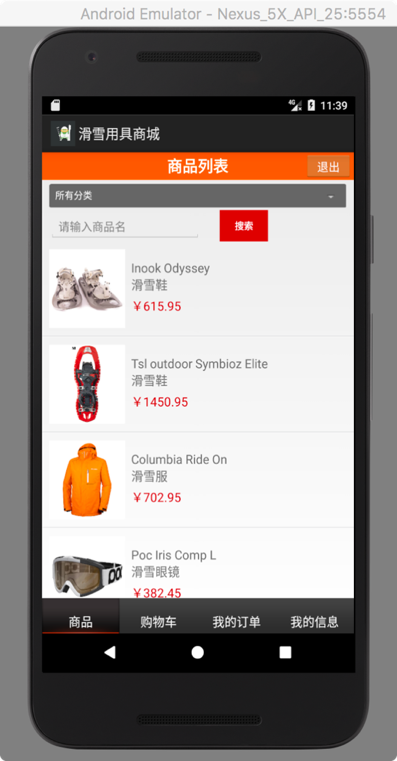


图 4-9浏览商品信息模块界面

## 购物车模块流程图及界面实现

### 核心模块流程图

图4-10为购物车模块的操作流程图。

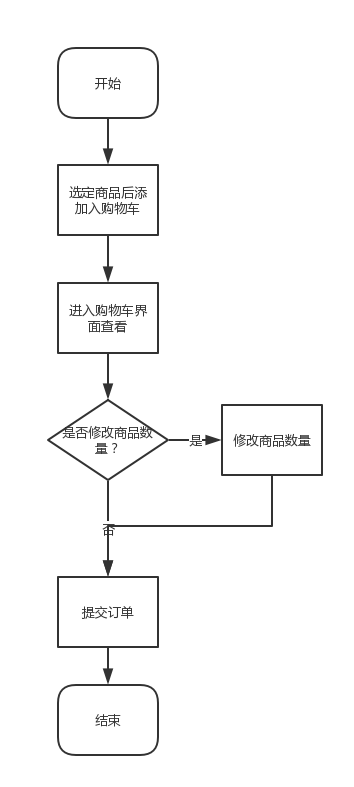


图4-10购物车模块操作

### 界面实现

图4-11为购物车模块的界面实现的部分截图。

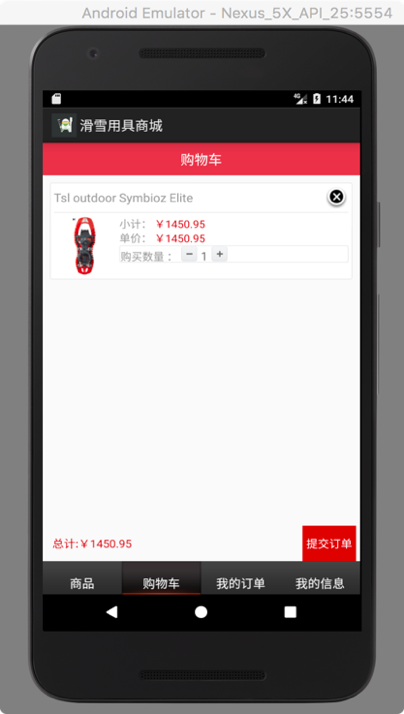


图 4-11 购物车模块界面截图

## 用户订单管理模块流程图及界面实现

### 核心模块流程图

图4-12为用户订单管理模块的操作流程图。

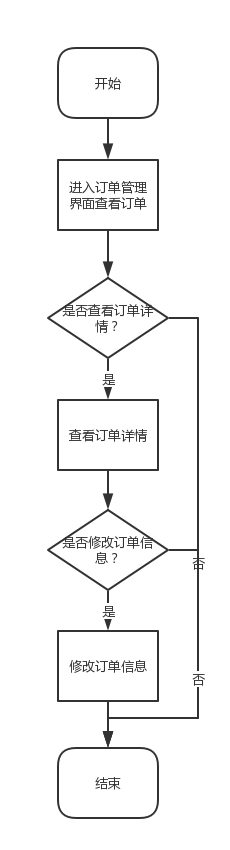


图4-11用户订单管理模块操作

### 界面实现

图4-13为用户订单管理模块的界面实现的部分截图。



图 4-13 用户订单管理模块界面截图

## 本章小结

本章主要对系统关键模块的流程实现进行了说明，介绍了商品管理模块、订单管理模块、商品信息浏览模块、购物车模块和用户订单管理模块的核心流程，并且分别展示了这些模块的具体程序页面。

# **系统测试**

本章节重点介绍系统的测试工作，单元测试对各功能模块进行验收，集成测试对各功能模块的组合调用进行测试。通过系统测试和改进工作，以求系统达到最佳使用效果。

## 测试概要

### 测试目标

对本系统进行全面的测试，对各功能模块进行验证和错误排查，发现系统中存留的功能缺陷或代码错误，并进行相应的修改和调整，减少系统运行中的故障率，以保证系统满足用户的功能和性能需求。

### 测试方案

滑雪用具商城系统的测试依次包括单元测试、集成测试，其中集成测试包括系统功能测试以及性能测试。

## 单元测试

在单元测试中，需要对每个子模块进行黑盒测试和白盒测试，用于检查程序内部逻辑和功能的准确性，其中白盒测试采用基本路径覆盖的方法，主要针对两个方面进行测试，一方面是流程图的各个判定框，另一方面是程序执行路径。黑盒测试是根据被测试程序功能来进行测试，这里采用划分等价类和边界值分析的方式设计黑盒测试的测试用例，来保证平台功能的准确性。由于篇幅限制，在这里我们对主要模块的核心业务流程进行测试说明。

### 商品管理模块测试

商品管理功能包括商品分类管理，包括添加新分类、编辑分类和删除分类；以及商品详细信息管理，包括修改、查看、删除和图片管理。

如表5-1所示，该表为服务器端在黑盒测试中商品管理功能测试的等价类划分类结果。

表 5-1商品管理模块测试等价类划分结果

|  |  |  |
| --- | --- | --- |
| **输入数据** | **有效等价类** | **无效等价类** |
| 点击操作 | 点击添加新分类按钮 | 点击其他按钮 |
| 点击操作 | 点击编辑分类按钮 | 点击其他按钮 |
| 点击操作 | 点击删除分类按钮 | 点击其他按钮 |
| 点击操作 | 点击修改商品信息按钮 | 点击其他按钮 |
| 点击操作 | 点击删除商品按钮 | 点击其他按钮 |
| 点击操作 | 点击商品图片管理按钮 | 点击其他按钮 |

如表5-2所示，该表为服务器端在黑盒测试中管理商品测试用例表。

表 5-2商品管理模块测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **测试内容** | **测试数据** | **期望结果** | **实际结果** |
| 1 | 进入添加新分类界面 | 点击添加新分类按钮 | 进入添加新分类界面 | 进入添加新分类界面 |
| 2 | 进入添加新分类界面 | 点击其他按钮 | 无法进入添加新分类界面 | 无法进入添加新分类界面 |
| 3 | 进入编辑分类界面 | 点击编辑分类按钮 | 进入编辑分类界面 | 进入编辑分类界面 |
| 4 | 删除分类 | 点击删除按钮 | 分类被删除 | 分类被删除 |
| 5 | 进入修改商品信息界面 | 点击修改商品信息按钮 | 进入修改商品信息界面 | 进入修改商品信息界面 |
| 6 | 删除商品 | 点击删除商品按钮 | 商品被删除 | 商品被删除 |
| 7 | 删除商品 | 点击其他按钮 | 删除失败 | 删除失败 |
| 8 | 进入图片管理界面 | 点击图片管理按钮 | 进入图片管理界面 | 进入图片管理界面 |

### 订单管理模块测试

如表5-3所示，该表为订单管理模块功能等价划分结果。

表 5-3订单管理模块测试等价类划分结果

|  |  |  |
| --- | --- | --- |
| **输入数据** | **有效等价类** | **无效等价类** |
| 账户 | 有效的账户，包括前几位字母的匹配 | 1. 不存在的账户 2. 不匹配的字母 3. 超过长度的账户 |
| 订单号 | 有效的订单号，包括前几位数字的匹配 | 1. 不存在的订单号 2. 不匹配的前位数字 3. 超过长度的订单号 |
| 点击操做 | 点击查看订单详情按钮 | 点击其他按钮 |
| 点击操作 | 点击发货按钮 | 点击其他按钮 |

如表5-4所示，该表为服务器端在黑盒测试中订单管理测试用例表。

表 5-4 订单管理模块测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **测试内容** | **测试数据** | **期望结果** | **实际结果** |
| 1 | 输入账户查询 | 完整账户 | 查询有效 | 查询有效 |
| 2 | 输入账户查询 | 不存在的账户 | 查询无效 | 查询无效 |
| 3 | 输入账户查询 | 账户前3位字母 | 查询有效 | 查询有效 |
| 4 | 输入订单号查询 | 完整订单号 | 查询有效 | 查询有效 |
| 5 | 输入订单号查询 | 不存在的订单信息 | 查询无效 | 查询无效 |
| 6 | 输入订单号查询 | 订单前5位 | 查询有效 | 查询有效 |
| 7 | 点击查看订单详情按钮 | 点击查看订单详情按钮 | 进入查看订单详情界面 | 进入查看订单详情界面 |
| 8 | 点击查看订单详情按钮 | 点击其他按钮 | 进入查看订单详情界面失败 | 进入查看订单详情界面失败 |
| 9 | 点击发货按钮 | 点击发货按钮 | 发货成功 | 发货成功 |
| 10 | 点击发货按钮 | 点击其他按钮 | 发货失败 | 发货失败 |

### 注册登录模块测试

如表5-5所示，该表为注册登录模块测试等价划分结果。

表 5-5注册登录模块功能测试等价类划分结果

|  |  |  |
| --- | --- | --- |
| **输入数据** | **有效等价类** | **无效等价类** |
| 用户名 | 10位字母和数字的组合 | 1. 超过长度的用户名 2. 用户名为空 3. 带有特殊符号的用户名 |
| 输入密码 | 10位字母和数字的组合 | 1. 超过长度的密码 2. 密码为空 3. 带有特殊符号的密码 |
| 确认密码 | 与输入密码相同 | 1. 与输入密码不同 |
| 手机号 | 11位数字 | 1. 超过长度或短于长度的手机号 2. 手机号为空 3. 带有特殊符号或字母的手机号 |

如表5-5所示，该表为注册登录模块录入功能测试用例表。

表 5-5注册登录模块功能测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **测试内容** | **测试数据** | **期望结果** | **实际结果** |
| 4 | 用户名 | John | 输入有效 | 输入有效 |
| 5 | 用户名 | 空 | 输入无效 | 输入无效 |
| 6 | 用户名 | John123 | 输入有效 | 输入有效 |
| 7 | 用户名 | 1234567890ll | 输入无效 | 输入无效 |
| 8 | 密码 | 空 | 输入无效 | 输入无效 |
| 9 | 密码 | 12¥¥¥ | 输入无效 | 输入无效 |
| 10 | 密码 | 123456789009 | 输入无效 | 输入无效 |
| 11 | 密码 | 123 | 输入有效 | 输入有效 |
| 12 | 密码 | 123acvv | 输入有效 | 输入有效 |
| 13 | 确认密码 | 与输入密码不同的确认密码 | 输入无效 | 输入无效 |
| 14 | 确认密码 | 与输入密码相同的确认密码 | 输入有效 | 输入有效 |
| 15 | 电话号码 | 1786270 | 输入无效 | 输入无效 |
| 16 | 电话号码 | 178627032721 | 输入无效 | 输入无效 |
| 17 | 电话号码 | 17862703272 | 输入有效 | 输入有效 |
| 18 | 电话号码 | 178\*\*& | 输入无效 | 输入无效 |

### 注册登录模块测试

如表5-6所示，该表为购物车模块测试用例等价划分结果。

表 5-6购物车模块功能测试等价类划分结果

|  |  |  |
| --- | --- | --- |
| **输入数据** | **有效等价类** | **无效等价类** |
| 点击数量增加按钮 | 点击数量增加按钮 | 点击其他按钮 |
| 点击数量减少按钮 | 点击数量减少按钮 | 点击其他按钮 |
| 点击结算按钮 | 点击结算按钮 | 点击其他按钮 |

如表5-7所示，该表为购物车模块功能测试用例表。

表 5-7购物车模块功能测试用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **序号** | **测试内容** | **测试数据** | **期望结果** | **实际结果** |
| 1 | 点击数量增加按钮 | 点击数量增加按钮 | 数量增加 | 数量增加 |
| 2 | 点击数量增加按钮 | 点击其他按钮 | 数量没有变化 | 数量没有变化 |
| 3 | 点击数量减少按钮 | 点击数量减少按钮 | 数量减少 | 数量减少 |
| 4 | 点击数量减少按钮 | 点击其他按钮 | 数量没有变化 | 数量没有变化 |
| 5 | 点击结算按钮 | 点击结算按钮 | 显示结算成功 | 显示结算成功 |
| 6 | 点击结算按钮 | 点击其他按钮 | 没有结算反应 | 没有结算反应 |

## 系统测试

对系统的各个功能模块进行功能性的测试，确保每一个模块的功能满足系统需求，并对系统整体进行性能测试。

### 功能测试

#### 商品管理模块功能测试

商品模块包括商品分类管理模块和商品详细信息管理等子模块。以上几个模块在系统中表现为不同的界面，因此该部分采用黑盒测试对其进行功能性的测试。表5-8是商品管理模块功能测试用例。

表 5-8 商品管理模块功能测试

|  |  |  |  |
| --- | --- | --- | --- |
| **用例编号** | **测试内容** | **预期结果** | **实际结果结果** |
| 1 | 商品分类管理模块在系统中集成测试 | 商品分类信息与实际商品分类信息一致，且可以进行添加、查看、编辑和删除操作 | 与预期结果相同 |
| 2 | 商品详细信息管理模块在系统中集成测试 | 商品信息与实际商品情况一致，且可以进行添加新商品、修改、查看、删除、管理商品图片等操作 | 与预期结果相同 |

#### 订单管理模块功能测试

订单管理模块包括查询订单模块、查看订单模块和处理订单模块。采用黑盒测试的方式对其进行功能性的测试。表5-9是订单管理模块功能测试用例。

表 5-9 订单管理模块功能测试

|  |  |  |  |
| --- | --- | --- | --- |
| **用例编号** | **测试内容** | **预期结果** | **实际结果结果** |
| 1 | 查询订单模块在系统中集成测试 | 能够进行通过账户、订单号或订单状态的模糊查询 | 与预期结果相同 |
| 2 | 查看订单模块在系统中集成测试 | 能够查看订单信息和每条订单的详细信息 | 与预期结果相同 |
| 3 | 处理订单模块在系统中集成测试 | 能够对未发货的订单进行发货处理 | 与预期结果相同 |

#### 浏览商品模块功能测试

后台管理员模块包括按商品分类浏览和浏览全部商品信息。滑雪用具商城系统的功能性测试采用黑盒测试的方式对其进行测试。

表5-10是浏览商品模块功能用例。

表 5-10 浏览商品模块功能测试

|  |  |  |  |
| --- | --- | --- | --- |
| **用例编号** | **测试内容** | **预期结果** | **实际结果结果** |
| 1 | 分类浏览商品 | 能够通过分类名分类浏览商品 | 与预期结果相同 |
| 2 | 浏览全部商品 | 商品信息与录入的信息一致 | 与预期结果相同 |

#### 购物车模块功能测试

购物车模块包括添加和减少商品数量和结算功能。采用黑盒测试的方式对其进行功能性的测试。表5-11是购物车模块功能测试用例。

表 5-11 购物车模块功能测试

|  |  |  |  |
| --- | --- | --- | --- |
| **用例编号** | **测试内容** | **预期结果** | **实际结果结果** |
| 1 | 添加商品数量 | 能够成功添加商品数量 | 与预期结果相同 |
| 2 | 减少商品数量 | 能够成功减少商品数量 | 与预期结果相同 |
| 3 | 结算 | 能够结算购物车中商品 | 与预期结果相同 |

### 性能测试

性能测试的目的是测试系统是否满足项目性能指标要求。该系统的性能测试使用了工具Loadrunner 。

## 本章小结

本章介绍了本系统的测试部分：测试概要、单元测试。测试概要描述了本系统测试方法。在测试中，我们对系统四大模块分别进行了黑盒测试和白盒测试。通过测试，发现并优化了系统存留的一些问题和缺陷，更好的保证了系统的健壮性、可用性和高效性。实践证明，本系统达到预期指标要求

# **结 论**

本论文描述了滑雪用具商城系统分析、设计、实现和测试的各个过程。根据用户需求构建了系统功能模块图，把系统划分成两个大的模块：服务端管理员模块和APP用户模块，每一个模块又被分为多个小模块。服务端模块主要由商城管理员使用，该模块位于Web端，可通过管理员账号进行登录，然后管理员可以发布与滑雪用具有关的相关资讯，还可以进行商品管理、订单管理、注册用户管理等操作。APP模块主要由商城用户使用，该模块位于手机移动端，商城注册用户通过账号密码登录系统，在该模块中进行浏览商品信息、购物车管理、订单管理和个人信息管理等操作。由用户需求和系统功能模块划分得出系统架构，整个后台系统被分为三层，显示层放置Html，Css，JS等静态文件显示系统前台界面，整个前台的布局使用Struts框架完成。中间层被细分为业务逻辑层和基本功能层，业务逻辑层用来将用户请求转发给基本功能层处理然后将处理结果返回给用户，基本功能层用来处理用户的请求。数据库日志文件访问层对数据库表和日志文件进行增、删、改、查等操作；前台系统通过Android技术设计，java实现各种方法，安卓模拟器设计界面。通过各种技术的整合，最后实现了该滑雪用具商城。

# **参考文献**

[1] 刘玉军,杨晔.我国移动电子商务运营模式分析与发展对策研究[J].情报科学,2014,32(04):122-125.

[2] 雷灵光,荆继武,王跃武,张中文.一种基于行为的Android系统资源访问控制方案[J].计算机研究与发展,2014,51(05):1028-1038.

[3] 倪红军.基于Android平台的消息推送研究与实现[J].实验室研究与探索,2014,33(05):96-100.

[4] 廖卫红,周少华.移动电子商务互动营销及应用模式[J].企业经济,2012,31(03):67-71.

[5] 彭博. APP视角下的移动电子商务发展潜力研究[D].重庆工商大学,2015.

[6] 龚秀芳.移动电子商务的现状和发展前景分析[J].电子商务,2013(11):6-7+9.

[7] 刘翔. 面向移动计算的WEB中间件关键技术研究[D].电子科技大学,2013.

[8] 鹿浩.移动计算技术及应用[J].湖北邮电技术,2001(02):11-15.

[9] Zhang Li. The Research and Implementation of JavaEE in Telecom’s CRM System[M].Springer New York:2013-06-15.

[10] Zhenwu Wang. A Rapid Development Framework of Enterprise Information System Based on Components and JavaEE[M].Springer Berlin Heidelberg:2015-06-15.

[11] Yusuke Kosuda,Ryoichi Sasaki. Proposal and Implementation of SSH Client System Using Ajax[M].Springer Berlin Heidelberg:2009-06-15.

[12] Benjamin Jenghorng Wu,Guang-Han Chung,Yen-Ning Su. Developing the Upgrade Detection and Defense System of SSH Dictionary-Attack for Multi-Platform Environment[J]. iBusiness,2011,03(01).

# 

# **致 谢**

在论文即将完成之时，我在这里由衷的感谢本科四年的时间给予我无私帮助的人。

感谢我的导师权光日老师，在我论文的指导方面向我提供了很多帮助，并为我撰写论文提出了许多宝贵的意见和建议，另外及时的督促和认真的提示，让我在把握论文进度方面游刃有余。

感谢大学四年所有给我授课的老师们，感谢他们为我传道授业解惑，让我在专业知识方面有所增长、思维能力得到锻炼，感谢他们教会我很多做人做事的道理。

感谢和我朝夕相处了四年的室友，感谢他们一如既往，并肩作战的陪伴。

最后我要感谢我的父母和家人，是他们关心与支持，让我能够顺利完成学业，找到奋斗的方向和前进的动力。

我谨向所有关心、帮助我的人们表示最诚挚的感谢和最美好的祝愿。