摘要

在“互联网+”时代大潮的推动下，电商平台得到高速发展，网上购物已经融入到学生的日常生活。从全国各地来大学的学生和家人的物品往来靠的就是方便快捷的快递，特别在节日、假期前、毕业生离校前的几个时间段，快递数量出现激增的现象，快递的囤积也越来愈多。

本系统主要功能有学生快递信息添加，快递员配送快递，学生通过第三方支付宝接口平台支付快递费，管理员对系统信息进行维护等，以发送订单的方式，快速响应学生的需求，提供送货上门的服务，解决部分学生因为惰性而导致的快递积压的问题。系统是基于“Spring boot+Spring+Springmvc+Mybatis”框架开发，采用前后端分离的技术实现。网页前端基于JSP技术开发，并通过Ajax技术实现前后端异步调用，后端功能用Spring boot框架和java语言进行实现，通过Mysql数据库对快递信息、用户信息、系统消息进行管理数据存储。

关键词：Spring Boot;JSP;支付宝支付平台;Ajax;异步调用;MySql;

**ABSTRACT**

Driven by the tide of the "Internet +" era, e.commerce platforms have developed rapidly, and online shopping has been integrated into students' daily life. Students and their families who come to the university from all over the country rely on convenient and fast express delivery. Especially in the period before festivals, holidays, and before graduates leave school, the number of express delivery surges, and the hoarding of express delivery also increases. more and more.

The main functions of this system include adding student express information, couriers delivering express delivery, students paying express fees through a third.party Alipay interface platform, and administrators maintaining system information, etc., the method of sending orders, quickly responding to students' needs, and providing delivery Door.to.door service solves the problem of express backlog caused by inertia for some students. The system is developed based on the "Spring boot+Spring+Springmvc+Mybatis" framework, and is implemented by the separation of front and back ends. The front.end of the web page is developed based on JSP technology, and the front.end and back.end asynchronous calls are realized through Ajax technology. The back.end function is realized by the Spring boot framework and java language, and the express information, user information and system information are managed and stored through the Mysql database.

**Keywords**: Spring Boot;JSP;Alipay payment platform;Ajax;Asynchronous call;MySql;

目录

[1 绪论 5](#_Toc97647781)

[1.1 课题背景和意义 5](#_Toc97647782)

[1.2国内外发展现状 5](#_Toc97647783)

[1.3 论文的主要工作 7](#_Toc97647784)

[2 相关理论和技术 8](#_Toc97647785)

[2.1 SSM框架 8](#_Toc97647786)

[2.2 SpringBoot 8](#_Toc97647787)

[2.3 Ajax 9](#_Toc97647788)

[3 系统需求分析 10](#_Toc97647789)

[3.1 业务需求分析 10](#_Toc97647790)

[3.1.1 快递业务分析 10](#_Toc97647791)

[3.1.2 快递业务流程 10](#_Toc97647792)

[3.2 角色需求分析 13](#_Toc97647793)

[3.2.1 学生角色 13](#_Toc97647794)

[3.2.2 配送员角色 14](#_Toc97647795)

[4 系统设计 16](#_Toc97647796)

[4.1 系统总体设计 16](#_Toc97647797)

[4.2 功能详细设计 16](#_Toc97647798)

[4.2.1 快递模块 16](#_Toc97647799)

[4.2.2 留言模块 17](#_Toc97647800)

[4.2.3 快递公司模块 18](#_Toc97647801)

[4.2.4 系统模块 19](#_Toc97647802)

[4.3 系统数据库设计 20](#_Toc97647803)

[4.3.1 实体属性图 20](#_Toc97647804)

[4.3.2 系统完整E-R图 23](#_Toc97647805)

[4.3.3 数据库表设计 24](#_Toc97647806)

[5 系统功能实现 28](#_Toc97647807)

[5.1登录注册模块 28](#_Toc97647808)

[5.2 快递信息模块 29](#_Toc97647809)

[5.2.1 学生快递信息模块 29](#_Toc97647810)

[5.2.2 配送员快递信息模块 32](#_Toc97647811)

[5.2.3 管理员快递信息模块 34](#_Toc97647812)

[5.3用户信息模块 34](#_Toc97647813)

[5.4 系统设置模块 35](#_Toc97647814)

[5.5 其他功能模块 36](#_Toc97647815)

[5.5.1 留言功能模块 36](#_Toc97647816)

[5.5.2 首页功能模块 37](#_Toc97647817)

[6 系统测试 38](#_Toc97647818)

[6.1 系统测试环境 38](#_Toc97647819)

[6.2 系统功能测试 38](#_Toc97647820)

[6.2.1 登录注册模块功能 38](#_Toc97647821)

[6.2.2 快递信息查询功能 39](#_Toc97647822)

[6.2.3 配送快递功能 39](#_Toc97647823)

[6.2.4 确认收货和支付快递费功能 40](#_Toc97647824)

[7 总结 41](#_Toc97647825)

[8 参考文献 42](#_Toc97647826)

[致谢 43](#_Toc97647827)

# 1 绪论

## 课题背景和意义

校园实施代取快递，可以有效解决师生不能及时取件的问题，可以帮助 收发点解决当日快递积压、囤货、重复作业等问题，能够有效提升快递服务的质量，提升师生的满意度，且校园快递代取通常是大批量取件，集中派送，大大降低了快递派送终端优化服务的成本，满足快递派送终端优化的需求。据统计，在校大学生“预计近80%~95%的大学生有惰性思想” “其中约75%认为自己是惰性患者，50%的大学生反映他一贯如此,即惰性已成为贯穿他生活的问题行为。”大学生惰性的增长，使其很多事情都不愿意亲力亲为，甚至不愿意花几分钟时间排队吃饭，没课情况下，更不愿走出宿舍，因此，闪送﹑跑腿﹑外卖应运而生，而快递代取就可借鉴其业务模式,抓住大学生群体需求,顺势发展。对于日常教学而言，快递代取发展将会缓解校园快递散﹑杂、乱，缺少系统管理，及较多数里师生参与快递收发，引发人群拥堵，人身财产安全无法受到保障等一系列校园安全问题，满足学校后勤的安全、日常教学秩序的保障需求。

校园网点的存在正好中和了公司和学生客户群体时间上的矛盾，节约了公司的时间成本和人力成本;学生只需要利用课余时间随时收件寄件，又可以解决学生在时间上的问题。本文基于校园内快递派送﹑师生取件现状进行分析，提出大学校园快递代取新模式。

在校园代取快递业务中，因为需要接单员进行快递配送，所以这也为大学生提供了勤工俭学的途径。同学们可以自由地分配时间，在不耽误学习和生活的同时获得不错的收益，学习赚钱两不误！

学生创业最大的问题就是缺乏经验，而在校园跑腿代取快递业务中，不仅仅是投入低，风险小，而且还能在创业中积累经验，提升自己的管理能力、决策能力、协调能力、沟通能力。校园代取系统可以给学生更好的创业体验。

## 1.2国内外发展现状

据调查，高校校园内容有快递业务的公司分为有固定代理点和无固定代理点两个。以安徽国防科技职业学院为例，校园快递固定代理点主要有：圆通、申通、中通、邮政。无固定代理点的快递公司为韵达、顺丰、国通、天天等各快递公司。圆通为该校物流专业的校企合作企业之一，该校与2011年与上海圆通快递六安分公司签订合作协议并在校园内成立了圆通快递实训室，在学院大学生创业基地开展快递业务，并由物流管理专业教师进行指导，具体工作由物流专业学生完成。对于韵达这种没有固定代理点的快递公司，发件主要是在学院食堂外面，每天下午5点发件，只有半个小时取件时间。

校园快递现状，代理点纷繁复杂，目前大多数高校校园快递均存在代理点数量多、代理人员咱这样的现状。快递服务质量差，由于成本等问题，一些快递公司往往会选择学生公寓，学生食堂，学校大门口这样一些露天地面经营业务，货物摆放凌乱无序，造成容易丢失，安全系数不高的影响，导致师生难以对其产生依赖感，同时由于取件时间受限，给师生寄、取邮件带来诸多不便。

所以高校进行统一快递代取机制，已成为国内目前主流的趋势，以安徽国防科技职业学院为例，圆通物流专业实训室因通知准时、货物破损率低、送件上门、取件上门的到师生的广泛好评，服务满意度高，均优于其他快递公司。所以针对此情况，学院正在着手将所有快递整合纳入物流专业实训室，统筹规划、合理安排快递业务流程及相关资源。从而提升许愿快递服务质量及优化校园快递业务。

相比我国快递行业发展较晚，国外的快递行业发展已比较成熟。发达国家在快递行业信息使用及隐私保护大致有以下做法:

一、对快递公司的行业规范十分严格。英国邮政管理委员会把所有关于邮件安全的条款都写入颁发给运营商的许可证当中，以法律形式强制执行。因此，英国在管理邮政公司、快递人员等方面，都是严格按照行业规范准则执行的。

二、对快递员要求严格，不可有信用瑕疵。英国要求运营商以书面形式来担保所招聘人员无犯罪记录或无因邮政方面的事故受到过警告、解雇处分或在诚信方面有缺点，同时监督运营商检查应聘者所说内容是否真实。邮政行业准则还要求运营商确保每名员工都了解自己在保护邮件安全方面所承担的责任，加强上岗前培训。，在美国想当快递员必须提供社会安全号，有点类似我们的身份证号码。应聘者如有拖欠银行贷款或者犯罪记录，通过这个号码都能查到。

三、“高薪养廉”，给予优厚的福利。美国邮递员的薪资非常高，医疗和社会保险都由公司承担;而且在美国，快递员被认为是辛苦服务大众的职业，很受尊敬，所以快递员也会尽力保住自己的名声，不会为了蝇头小利，葬送自己的前程。德国也是如此，快递员的收入是很可观的。

## 1.3 论文的主要工作

正文主要讲述了如何使用SSM框架，利用Spring boot技术简化配置和JSP前端技术实现校园代取快递系统，做到校园快递的统一管理，以下为本文每章内容。

第一章以课题背景和国内外对于快递物流行业的发展现状作出了评估和分析。

第二章对本系统主要使用的技术原理和概念做一说明。

第三章从业务需求和角色需求两方面给出了系统的需求分析，说明系统开发的可行性。

第四章按照系统的需求分析给出系统总体功能架构图，明确系统的功能和基本的业务逻辑。

第五章按照系统总体框架，给出系统的详细设计，详细阐述。系统各功能的具体实现逻辑和实现效果。

第六章采用黑盒测试技术对系统功能进行测试，发现系统漏洞，提高系统可操作性行。

# 相关理论和技术

## SSM框架

SSM框架包括Spring、SpringMVC、MyBatis技术，常作为数据源较简单的web项目的框架。

Spring相当于一个大工厂，将项目用到的所有bean装配到一起，通过配置文件指定特定的参数取调用试题列的构造方法，实例化构造方法的对象，所以被称为项目的你合计。其核心思想是控制翻转（IOC），不用程序员去显式的new一个对象，而是通过spring 框架来完成一切。

SpringMVC就是在项目中拦截用户的请求，其核心Servlet即DispathcerServlet担任中介或者是前台的纸职责，对用户请求通过的HandlerMapping去找到相关的Controller，对对应的请求作出具体的操作。SpringMVC等同于SSH框架中Struts。

Mybatis就是对jdbc技术进行封装，它让底层的数据库操作变得透明。Mybatis的操作都是环绕这一个sqlSessionFactory实例展开的内容。Mybatis通过配置文件绑定到各实体类的Mapper文件，Mapper文件中配置了每个类对数据库进行的sql语句进行映射。在每次的数据库交互时，通过SqlSessionFactory实例化一个SqlSession，再执行sql命令，本系统中所有sql语句是由Mybatis统一管理。

总体来说，SSM框架就是把页面发送的请求交个控制器，然后再调用业务层逻辑，通过逻辑层向持久层发送请求进行数据库的交互，然后将记过返回给业务层，最终通过控制器调用视图展示数据。

## SpringBoot

SpringBoot是由Pivotal团队提供的全新框架，设计它的目的是为了简化Spring环境搭建的配置以及开发过程，该框架使用了特殊的方式来进行相关环境配置，从而使开发人员省去繁琐的环境配置，通过注解形势配置环境。通过这种方式，Spring boot在快速应用开发领域，成为了先驱者和领导者。

SpringBoot基于Spriing4.0设置，不仅继承了Spring框架原有的特性，并且还通过简化配置来进一步节省了Spring应用的整个搭建和开发过程，而且SpringBoot通过大量集成的框架的解决了各依赖包之间的版本冲突，使Spring不稳定性得到了很好的解决。

## Ajax

Ajax实现客户端与服务器可以在不刷新整个浏览器的情况下，与服务器进行异步通讯的技术，本系统中对于数据的删除和修改操作都是通过Ajax的异步通讯机制实现的。

传统的开发，每当用户进行请求时，都要刷新整个页面，哪怕只有一点内容，而Ajax就可以做到局部刷新的效果，提过了页面的性能，并且不会中断用户的操作。

# 系统需求分析

校园快递代取系统基于基本的快递业务流程，对于快递代取系统的核心功能进行了业务逻辑的分析。该部分是从学生和快递员两个角色进行了分析，了解快递业务的详细流程，并且针对两张角色进行的需求分析，确定了各自功能需求。

## 业务需求分析

### 快递业务分析

我们对于快递员送件业务进行了调研，充分了解快递业务各方面的详情。快递作为现在生活无可或缺的部分，一般是由学生发起快递代收申请。快递员查看快递列表后，通过查看快递的配送地址进行选择，来进行快递配送业务，快递配送完后，货物会进入指定的配送地点，快递员将快递配送地点告诉学生后，学生可以进入指定位置进行收货操作，收货完后给予快递员相应的报酬。如果配送地址变化，学生可以修改配送地址，若快递员没有配送到相应的地方，也要留言告诉学生，方便学生收货。

学生对于快递状态要有实时的获取，了解现在快递是处于何种状态，并且快递配送完成后，要及时通知到学生，方便学生获取。

对于快递员，要明确知道自己手中有多少快递还在配送当中，有哪些快递没有进行配送。

### 快递业务流程

通过对快递业务的分析，得到校园代取快递系统的核心业务流程：首先，学生输入账号密码，账号密码匹配成功后，登录进入代取业务系统首页，若不匹配，则提示账号密码是否正确。

学生进入后可以看到快递模块，该模块包括查询自己的快递和添加新的快递两部分。学生选择添加快递菜单，进入页面，输入快递信息。学生发起快递业务流程图如图3.1所示：

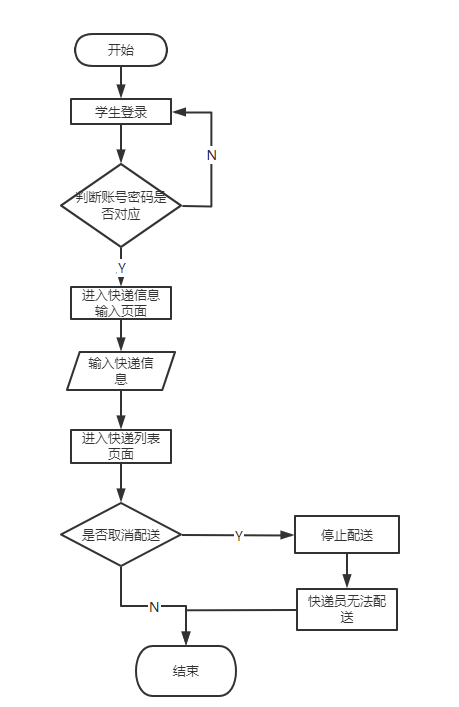


图3.1 学生发起配送业务流程图

在学生发起配送业务后，快递员可以对相应的快递单进行配送，如果学生取消配送后，快递员无法查看该快递信息。快递员将快递配送到指定位置后，输入取货地址后，配送完成，给学生进行留言，告诉学生具体的取货地点。快递员配送快递业务流程图如图3.2所示：

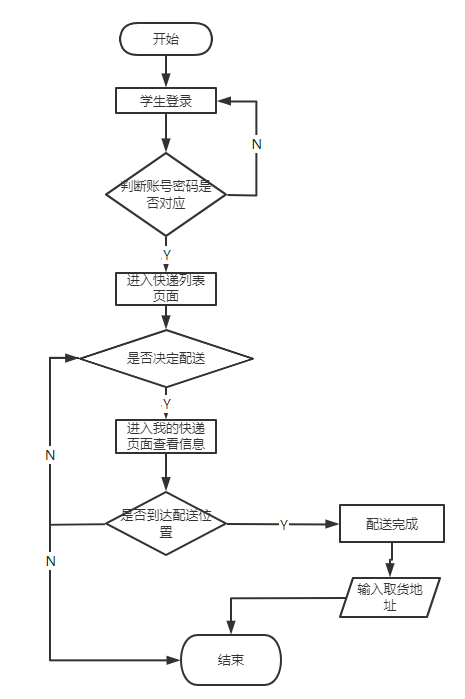


图3.2 快递员配送业务流程图

快递员配送完成后，学生进入我的快递页面查看此快递的信息和配送节点，并在留言页面查看配送员的留言信息和取货地址，确定收货，收货完毕后进行支付，成功支付后此快递业务结束。学生确认收货业务流程如图3.3所示：

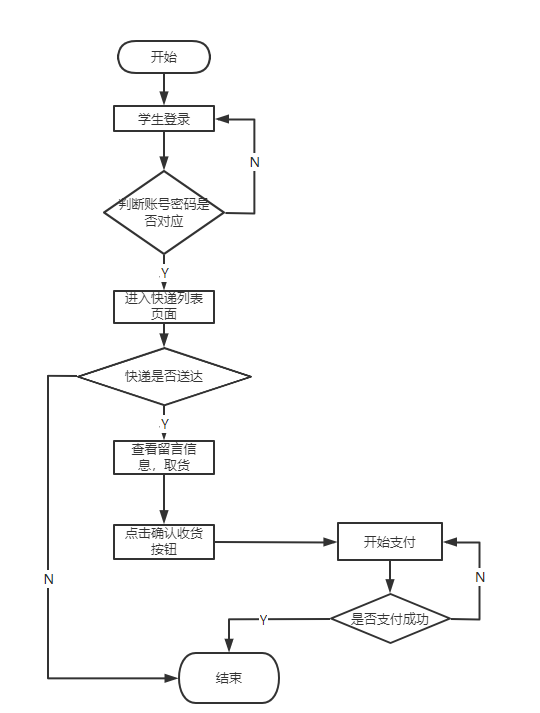


图3.3 学生确认收货和支付业务流程图

## 角色需求分析

通过上述对业务需求的分析，可以确定在快递配送业务中需要至少学生和快递员两个角色，所以该系统的角色可以分为快递员和学生。对于这两种角色的需求在以下小节进行具体的分析。

### 学生角色

学生作为该系统的主要发起人，负责整个快递业务的发起及快递信息的查看。学生可以进行快递信息的填写，修改快递信息，取消快递信息，查看快递信息，对于已到达的快递可以进行收货操作和支付操作，对于快递员的留言，可以进行查看和删除。

其角色功能图如3.4所示：

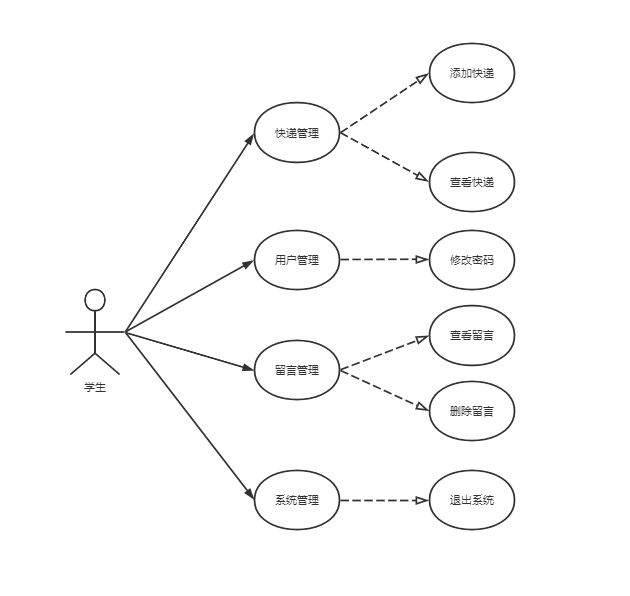


图3.4 学生角色功能图

### 配送员角色

配送员是该系统的另一个使用对象，主要用于快递的接单和配送功能。可以查看自己配送的单子，但是不能修改快递信息，还有给学生留言的功能，确认送达后，提醒学生取货地址。

其角色功能图如3.5所示：

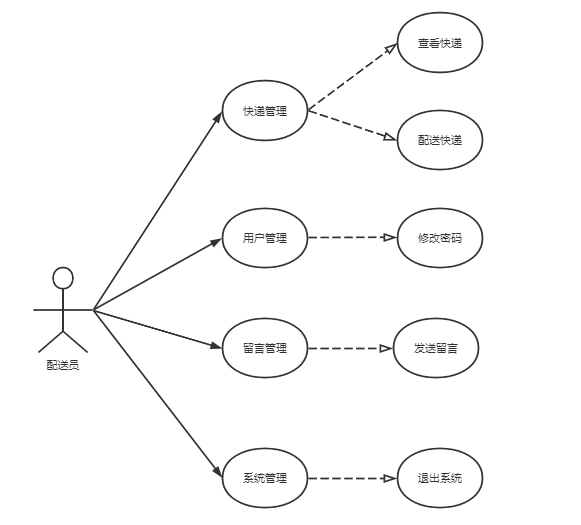


图3.5 配送员角色功能图

# 系统设计

## 系统总体设计

校园代取快递系统从角色划分，可以分为学生，配送员，管理员三种角色。因此，在系统权限方面，也可以分为学生，配送员，管理员三种角色。

学生权限，主要是针对快递信息的添加，查看，修改，删除的功能。快递员的权限主要用于对待配送的快递单进行查看和接单，对于配送中的快递单进行配送完成的操作。管理员的权限主要是对整个快递单的查看，订单的查看，用户信息管理和系统新闻进行管理。

校园代取快递系统的整体功能架构图如图4.1所示。

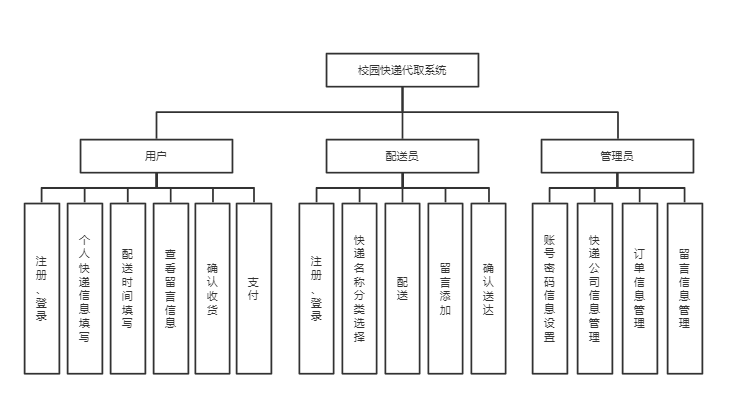


图4.1 系统整体功能架构图

## 功能详细设计

### 快递模块

快递模块为主要模块，该模块大致功能包括：创建快递单，修改快递信息，配送快递，送达快递，快递收货，支付快递等功能。学生创建完快递单后，配送员可以查看快递单，但是不能进行修改，学生可以修改快递单中的信息或者取消快递，配送员可以对快递进行配送选择，并且可以根据快递信息查询快递，对快递进行公司归类，通过快递公司查询到相应公司的快递信息，具体活动图如图4.2所示：

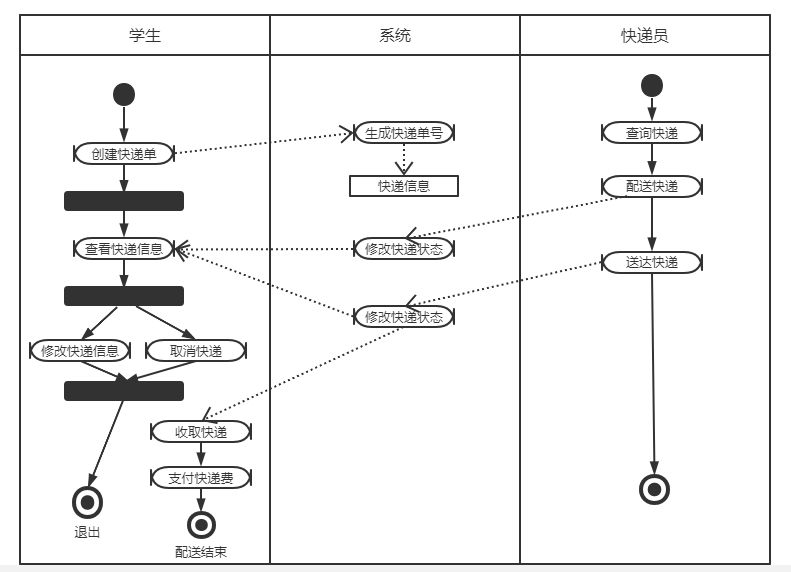


图4.2 快递模块活动图

### 留言模块

留言模块主要功能是为了给学生查看快递的收货地址，配送员送达快递后输入取货地址，系统自动补充留言信息发送给学生，学生登录到系统后在留言模块下可以看到新信息提示，进入后进行信息浏览，可以将信息标记为已读状态，并且可以查看历史所有的留言信息，对这些留言信息进行删除操作，具体活动图如图4.3所示：

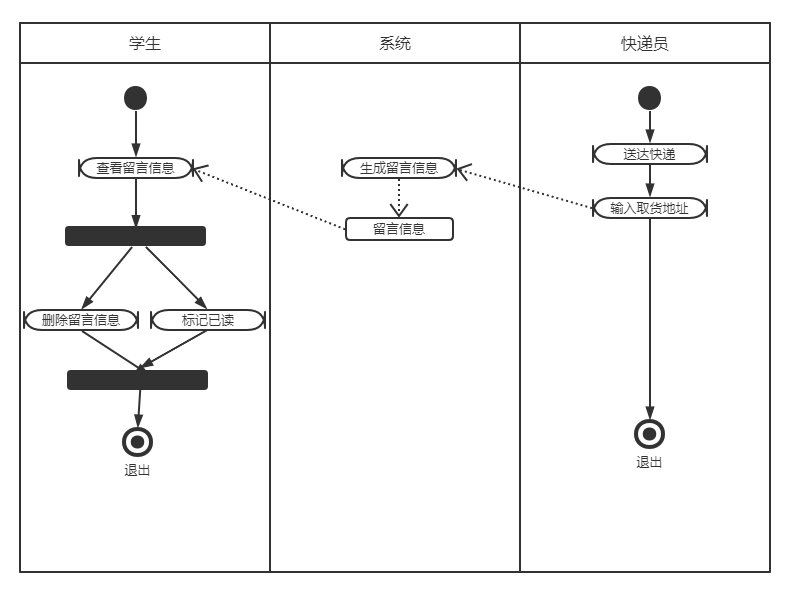


图4.3 留言模块活动图

### 快递公司模块

快递公司模块主要使用者是管理员，管理员可以对签订合同的物流公司纳入到系统中，学生填写快递信息时可以选择该物流公司，对于配送员来说，可以通过物流公司名称来筛选快递，管理员删除物流公司后，学生无法添加该物流公司的快递，配送员不能进行配送，具体活动图如图4.4所示：

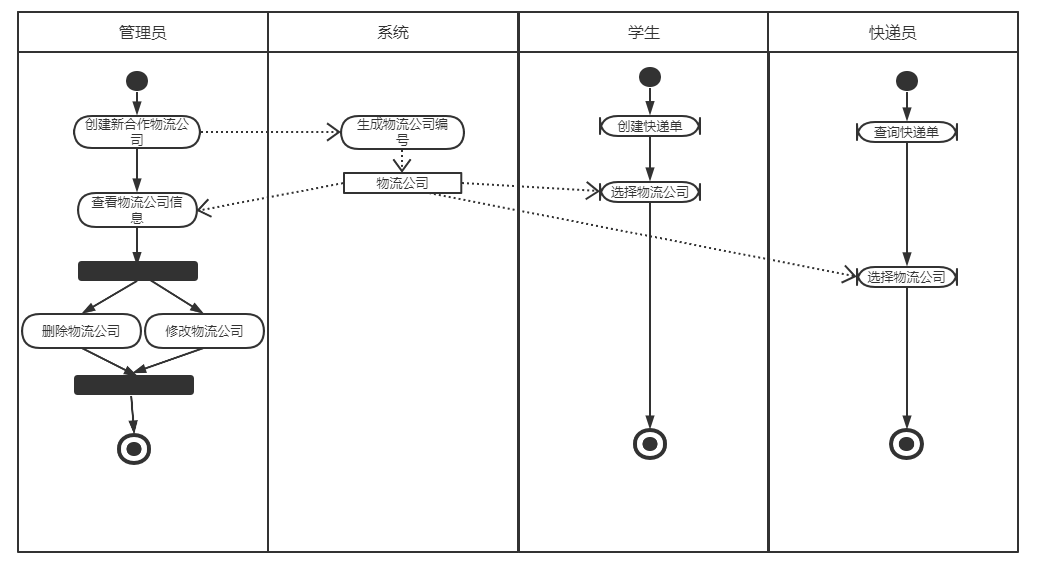


图4.4 留言模块活动图

### 系统模块

系统模块主要用于用户的注册和信息修改，对于管理员来说可以进行配送活动折扣的发布，并且管理员可以查看本系统当前月份的订单数，配送订单数，完成订单数和总收益额，且学生和快递员两种角色没有该功能缺陷，具体活动图如图4.5所示：

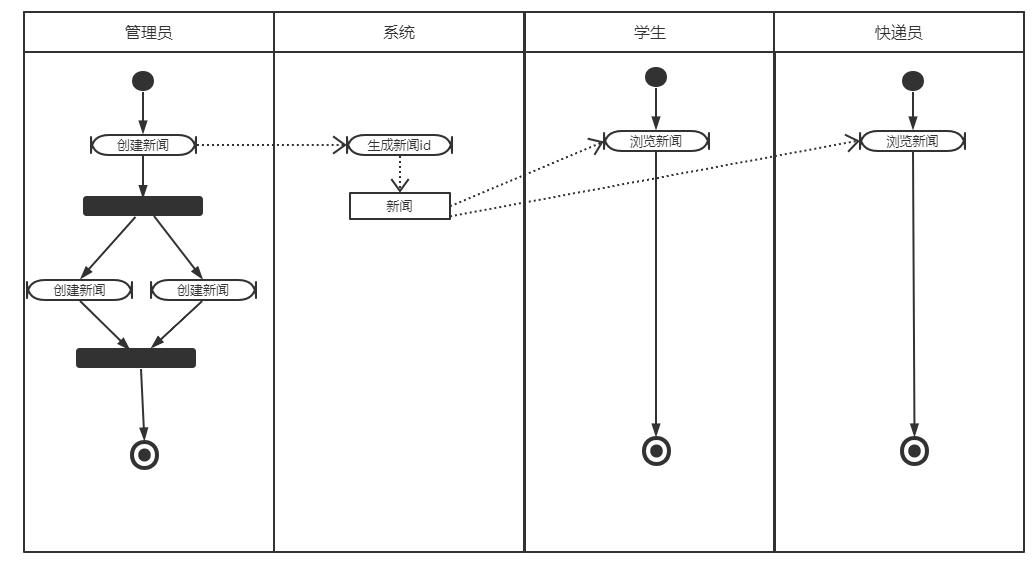


图4.5 系统模块活动图

## 系统数据库设计

### 实体属性图

根据上述分析的业务流程、功能需求等，可以得到该系统内涉及到的所有实体，包括：用户，快递，快递状态，快递公司，系统新闻，留言信息，订单信息。如下是我对每个实体所包含的属性进行的介绍。

用户类主要包含的属性有：用户id、用户账号、用户密码、用户角色。用户实体属性图如图4.6所示：

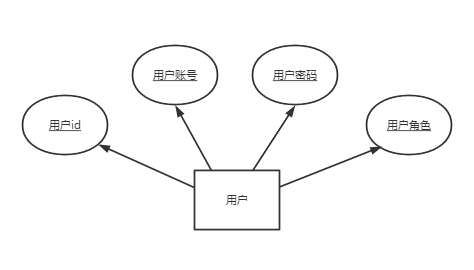


图4.6 用户实体ER图

快递类主要属性有：快递ID，取件码，收件人，收件人电话，物流公司，配送时间，宿舍号，留言信息，配送状态，收货状态，支付状态，配送员ID，用户ID，创建时间，更新时间。具体快递试题属性图如图4.7所示：

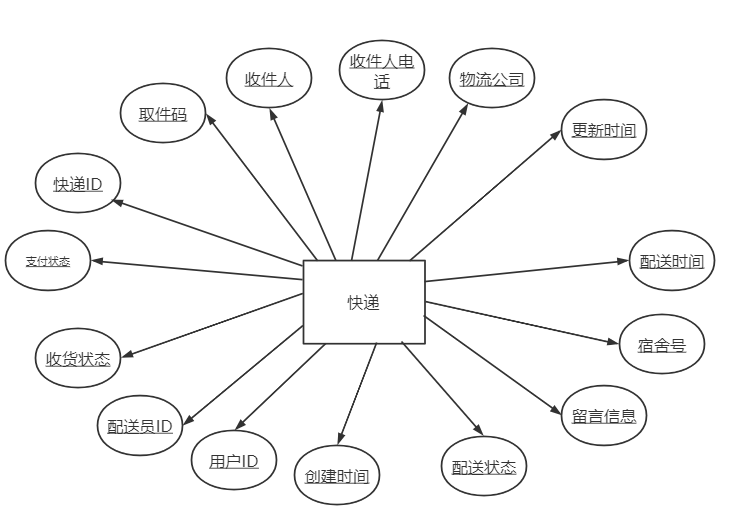


图4.7 快递实体属性图

快递状态类主要包含的属性有：快递状态ID，快递ID，创建时间，开始配送时间，完成配送时间，确认收货时间，支付完成时间。快递状态类实体属性图如图4.8所示：

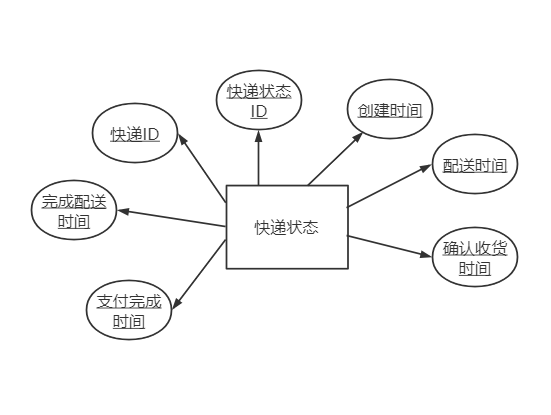


图4.8 快递状态实体属性图

快递公司类主要包含的属性有：快递公司id，快递公司名称。快递公司实体属性图如图4.9所示：

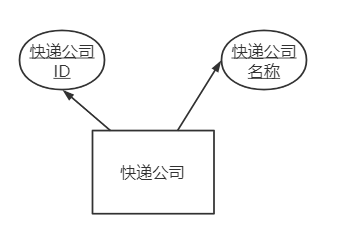


图4.9 快递公司实体属性图

系统新闻类主要包含的属性有：新闻ID，新闻标题，新闻内容，创建时间，更新时间。系统新闻实体属性图如图4.10所示：

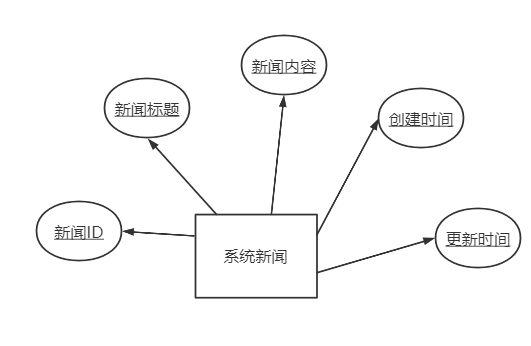


图4.10 系统新闻实体属性图

留言信息实体类主要包含的属性有：留言ID，留言内容，用户id，读取状态，创建时间，更新时间。具体留言信息实体属性如图4.11所示：

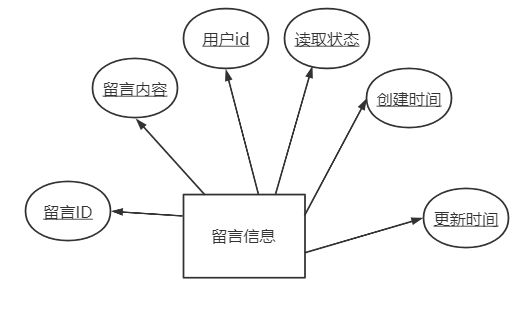


图4.11 留言信息实体属性

订单信息实体类主要包含的属性有：订单ID，订单号，订单金额，订单信息，快递id，创建时间，更新时间，支付状态。具体订单信息实体属性如图4.12所示：

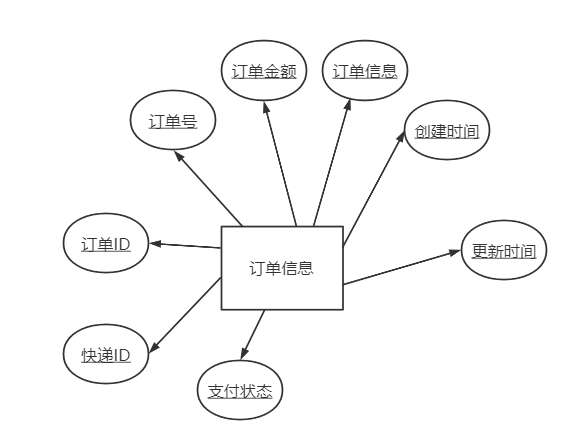


图4.12 订单信息实体属性图

### 系统完整E-R图

校园快递代取系统中的实体之间的关系有：

（1）1位学生能够管理n个快递单，接收n条留言信息。

（2）1位快递员可以配送n个快递单，发送n条留言信息。

（3）1位学生可以拥有n笔订单。

（4）1个快递单只能有1条留言，产生1笔订单。

校园快递代取系统完整E-R图如图4.13所示：

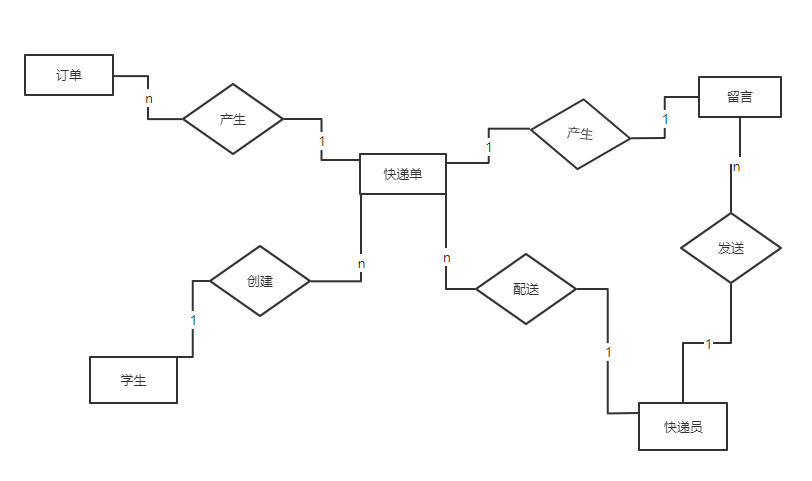


图4.13 校园快递代取系统整体E-R图

### 数据库表设计

根据实体属性图，设计数据库表。本系统设计到的数据库表如下：用户表，快递表，物流公司表，新闻表，快递状态表，菜单表，留言信息表，订单表。

用户表包含的字段有：用户id、用户账号、用户密码、用户角色。此处角色0代表管理员，1代表快递员，2代表学生。用户表详细设计如表4.1所示：

表4.1 用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **数据类型** | **长度** | **主键/外键** |
| uid | 用户id | int | 11 | 主键 |
| username | 用户账号 | varchar | 255 | 否 |
| password | 用户密码 | varchar | 255 | 否 |
| role | 用户角色（0：管理员、1：快递员、2：学生） | varchar | 255 | 否 |

快递表主要包含的字段有：快递ID，取件码，收件人，收件人电话，物流公司，配送时间，宿舍号，留言信息，配送状态，收货状态，支付状态，配送员ID，用户ID，创建时间，更新时间。快递表详细设计如表4.2所示：

表4.2 快递表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **数据类型** | **长度** | **主键/外键** |
| expressage\_id | 快递id | int | 11 | 主键 |
| expressage\_code | 取件码 | varchar | 255 | 否 |
| expressage\_recipient\_name | 收件人 | varchar | 255 | 否 |
| expressage\_recipient\_phone | 收件人电话 | varchar | 255 | 否 |
| expressage\_logistics\_company | 物流公司 | varchar | 255 | 否 |
| expressage\_delivery\_time | 配送时间 | varchar | 255 | 否 |
| expressage\_dormitory\_number | 宿舍号 | varchar | 255 | 否 |
| expressage\_message | 留言信息 | varchar | 255 | 否 |
| expressage\_delivery\_status | 配送状态 | varchar | 255 | 否 |
| expressage\_recipient\_status | 接收状态 | tinyint | 255 | 否 |
| expressage\_pay\_status | 支付状态 | varchar | 255 | 否 |
| delivery\_id | 配送员id | int | 11 | 否 |
| uid | 用户id | int | 11 | 否 |
| create\_time | 创建时间 | datetime | 0 | 否 |
| update\_time | 更新时间 | datetime | 0 | 否 |

快递状态表包含的字段有：快递状态ID，快递ID，创建时间，开始配送时间，完成配送时间，确认收货时间，支付完成时间。快递状态表详细设计如表4.3所示：

表4.3 快递状态表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **数据类型** | **长度** | **主键/外键** |
| status\_id | 快递状态id | int | 11 | 主键 |
| expressage\_id | 快递ID | int | 11 | 否 |
| start\_status\_time | 创建时间 | varchar | 255 | 否 |
| delivery\_status\_tiem | 配送时间 | varchar | 255 | 否 |
| complete\_delevery | 配送到达时间 | varchar | 255 | 否 |
| recipient\_status\_time | 确认收货时间 | varchar | 255 | 否 |
| complete\_pay\_time | 完成支付时间 | varchar | 255 | 否 |

快递公司表主要包含的字段有：快递公司id，快递公司名称。快递公司表具体详细设计如表4.4所示：

表4.4 物流公司表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **数据类型** | **长度** | **主键/外键** |
| expressagecompany\_id | 物流公司id | int | 11 | 主键 |
| expressagecompany\_name | 物流公司名称 | varchar | 255 | 否 |

系统新闻表主要包含的字段有：新闻ID，新闻标题，新闻内容，创建时间，更新时间。系统新闻表具体设计如表4.5所示：

表4.5 系统新闻表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **数据类型** | **长度** | **主键/外键** |
| expressage\_news\_id | 新闻ID | int | 11 | 主键 |
| expressage\_news\_title | 新闻标题 | varchar | 255 | 否 |
| expressage\_news\_info | 新闻内容 | text | 0 | 否 |
| create\_time | 创建时间 | datetime | 0 | 否 |
| update\_time | 更新时间 | timestamp | 0 | 否 |

留言信息表主要包含的字段有：留言ID，留言内容，用户id，读取状态，创建时间，更新时间。留言信息表具体表设计如表4.6所示：

表4.6 留言信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **数据类型** | **长度** | **主键/外键** |
| message\_id | 留言ID | int | 11 | 主键 |
| message\_info | 留言内容 | varchar | 255 | 否 |
| uid | 用户id | int | 11 | 否 |
| read\_status | 读取状态（0：未读，1：已读） | tinyint | 255 | 否 |
| create\_time | 创建时间 | datetime | 0 | 否 |
| update\_time | 更新时间 | timestamp | 0 | 否 |

订单信息表主要包含的字段有：订单ID，订单号，订单金额，订单信息，快递id，创建时间，更新时间，支付状态。订单信息表具体表设计如表4.7所示：

表4.7 订单信息表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **字段名称** | **字段含义** | **数据类型** | **长度** | **主键/外键** |
| order\_id | 订单ID | int | 11 | 主键 |
| order\_no | 订单号 | varchar | 255 | 否 |
| order\_amount | 订单金额 | double | 255 | 否 |
| order\_title | 订单标题 | varchar | 255 | 否 |
| expressage\_id | 快递ID | int | 11 | 否 |
| create\_time | 创建时间 | datetime | 0 | 否 |
| update\_time | 更新时间 | timestamp | 0 | 否 |
| order\_status | 支付状态（0：未支付，1：已支付”） | tinyint | 255 | 否 |

# 系统功能实现

## 5.1登录注册模块

登录注册模块实现了用户注册和用户登录功能，登录时账号密码校验并传达用户信息到整个系统的session中，方面后续功能操作获取用户属性。

用户注册页面实现效果如图5.1所示，用户在本页面输入用户名和密码，点击注册按钮，将页面信息传入到后台UserController中，并且系统自动分配角色为普通用户，通过Register方法将用户数据写入到数据库中。



图5.1 注册页面

用户登录页面实现效果如图5.2所示，用户在本界面输入用户名和密码后，将数据传入到UserController中，通过ulogin()函数将页面数据和数据库中的数据进行比对，数据比对成功后，进入系统首页。



图5.2 登录页面

用户登录到系统后，后台通过对用户角色信息读取数据库中的menu表，查询出该用户角色具备的菜单，将菜单数据读取后进行整合，输出到页面中。具体实现效果如图5.3所示：



图5.3 用户菜单信息

## 5.2 快递信息模块

## 5.2.1 学生快递信息模块

学生进入系统后，点击我的快递菜单，页面将学生信息传入到后台，后台根据用户id，获取到改用户的所有快递信息列表，展示到页面中，学生对快递具有的权限有修改、删除、确认收货、支付功能。具体实现效果如图5.4所示：

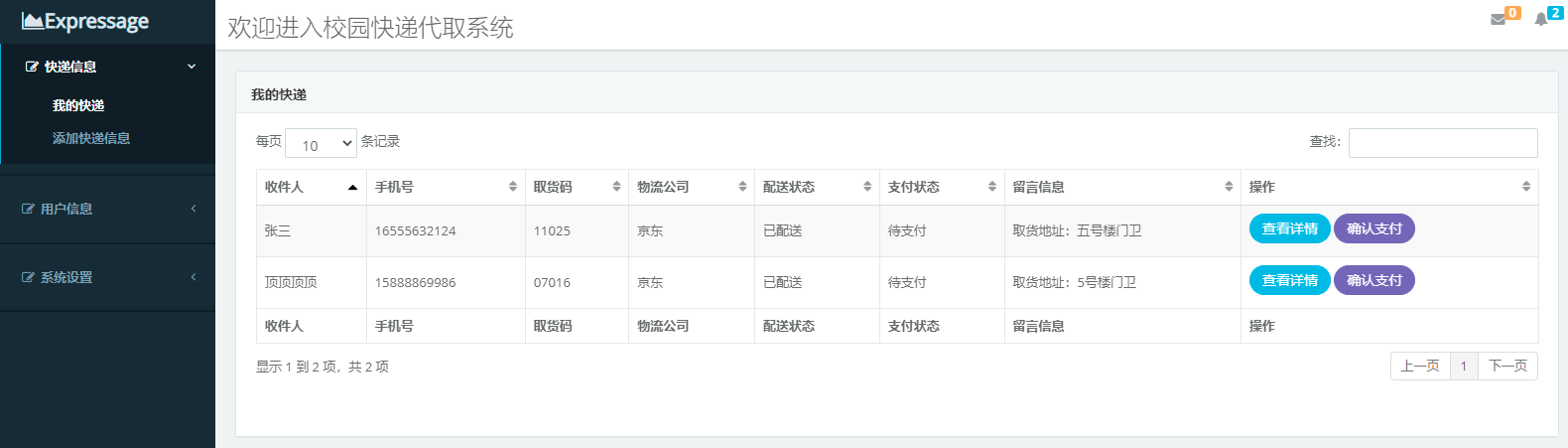


图5.4 我的快递页面

学生可以进行添加快递操作，点击添加快递信息菜单，页面跳转到添加快递页面，学生输入完快递信息后，点击添加按钮，页面将快递信息传入到ExpressageController中，ExpressageController通过session获取用户ID，并设置快递单的状态为待配送和待支付，通过addExpressageInfo()方法和mysql数据库交互，将快递信息保存到expressage表中，保存成功过后，前端返回到我的快递页面，显示出刚刚保存的快递信息，此时用户可以对快递进行查看和取消操作，具体实现效果如图5.5所示：



图5.5 添加快递信息页面

用户点击查看信息按钮后，页面将快递ID传给后台，后台通过getOneExpressageInfo()方法从数据库中查询到该快递ID 的所有值，并传给前台jsp页面，前台页面将后台的值显示到页面中。具体实现效果如图5.6所示：

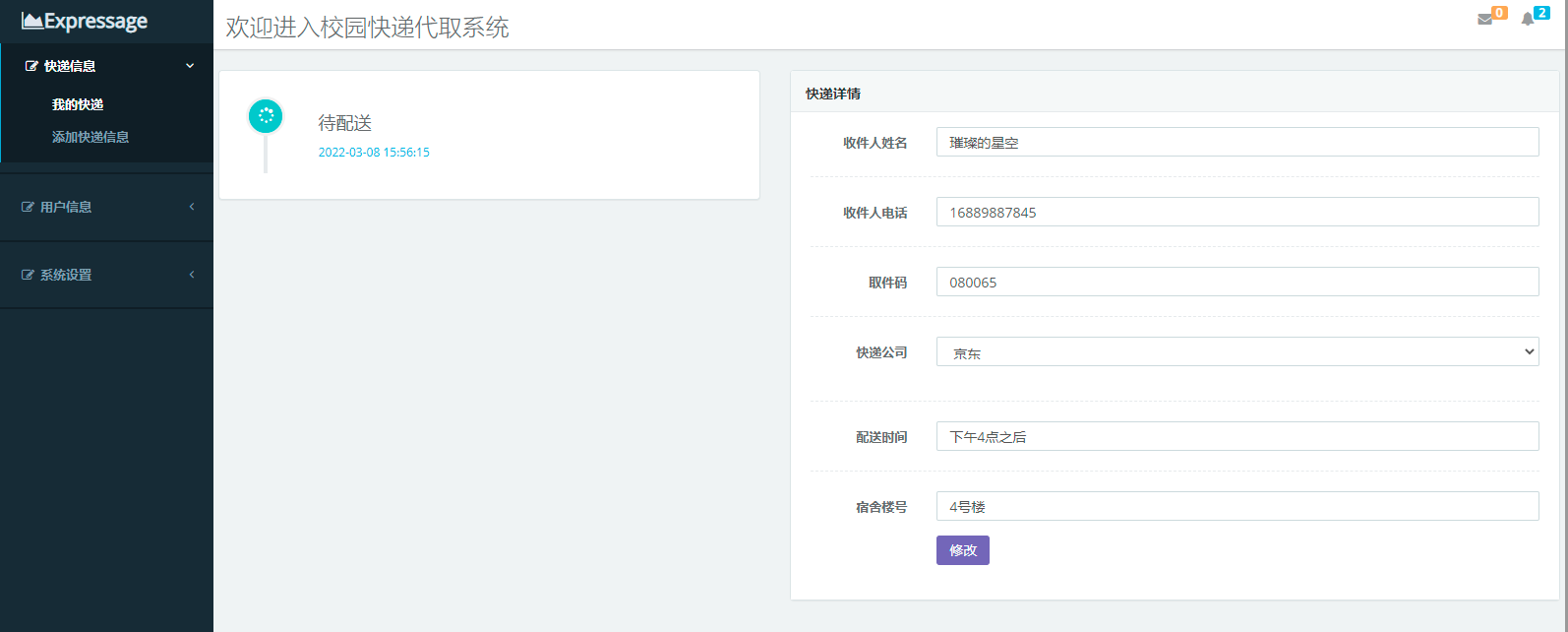


图5.6 查看详情页面

该页面展示快递的状态和快递信息，如果快递没有被接单，则页面显示修改按钮，用户可以修改快递信息，如果快递被接单，页面不显示修改按钮，用户无法修改快递信息。

用户若修改快递信息，通过页面输入要修改的值，点击修改按钮，将页面信息传给后台，后台通过updateExpressage()方法修改数据库中该快递数据的信息。

用户若想取消快递，则点击取消快递按钮，页面会提示用户“确定要取消吗”的提示框，用户点击确定后，通过ajax异步通讯技术将expressage\_id传给后台，后台通过delExpressage()方法将数据库中该快递信息删除，ajax得到后台调用成功的返回数据后刷新页面，则刚刚删除的快递信息就不显示到页面中。具体实现效果如图5.7所示：

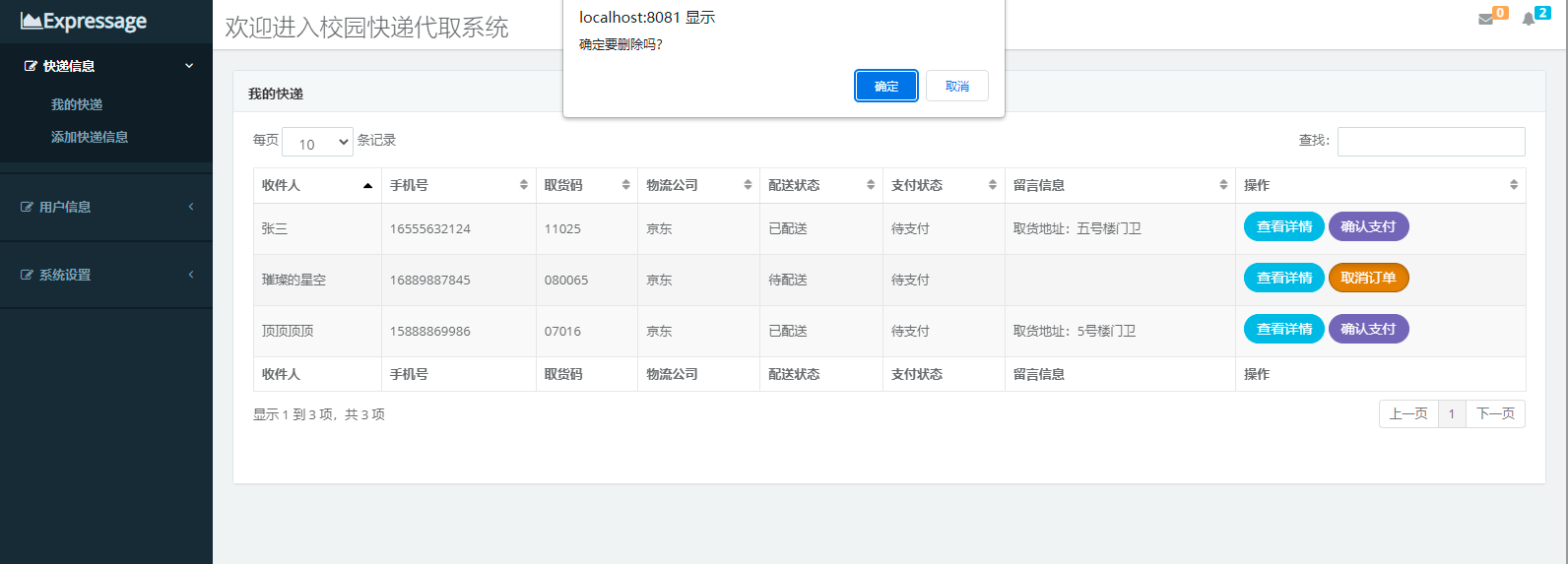


图5.7 取消快递功能

学生对快递还有确认收货的权限，等待快递状态变成“已配送”后，学生的我的快递页面对于该快递就会显示“确认收货”的按钮，点击按钮后页面提示“确定快递已送到手里了吗”的提示语，用户点击确定后，通过ajax技术将快递id传给后台，后台通过updateExpressage()方法修改本快递的接收状态为1，表示已接收，并且跳转到支付宝支付页面，进行该笔订单的支付。具体实现效果如图5.8、5.9所示：

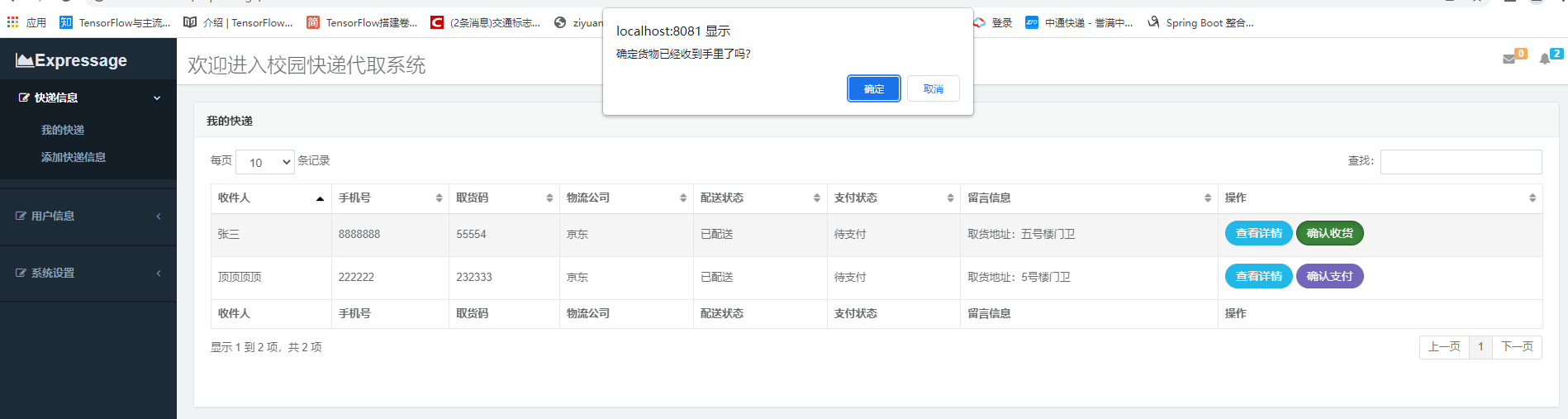


图5.8 确认收货功能



图5.9 支付宝支付页面

## 5.2.2 配送员快递信息模块

配送员进入系统后，点击我的菜单，页面将uid传给后台，后台通过查询类型将delivery\_id等于uid的快递信息传给前台，前台显示当前快递员正在配送和已经配送完成的所有快递。具体实现效果如图5.10所示：

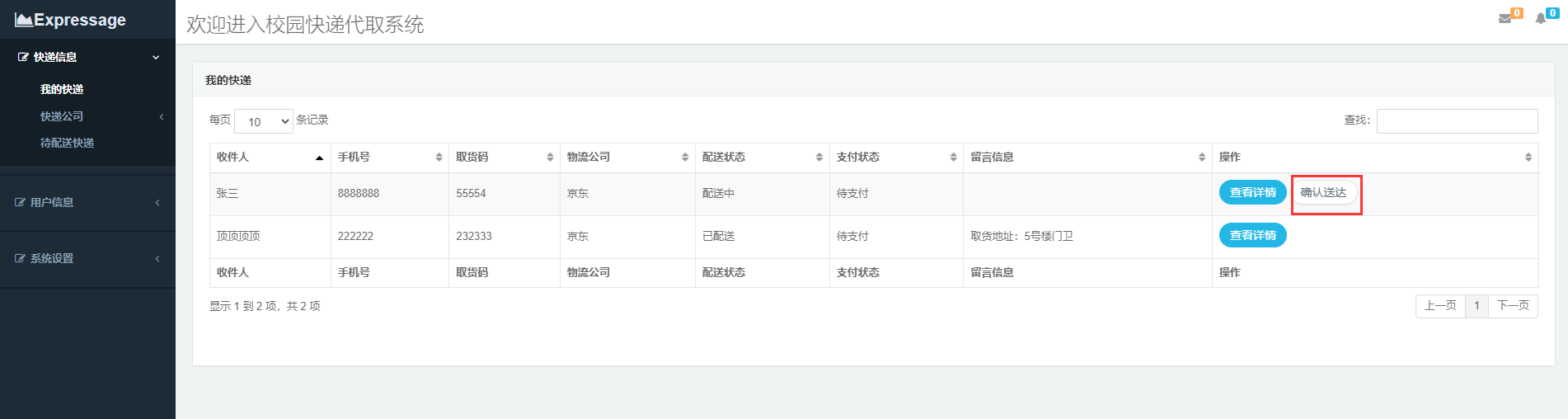


图5.10 配送员我的快递页面

配送员对于还没接单的快递可以进行接单操作，在待配送页面展示所有未接收的快递，点击我要配送按钮，页面提示“确定要配送吗”的提示信息，配送员点击确定后，页面将expressage\_id传给后台，后台修改数据库中该快递单的配送状态为配送中。具体实现效果如图5.11、5.12所示：

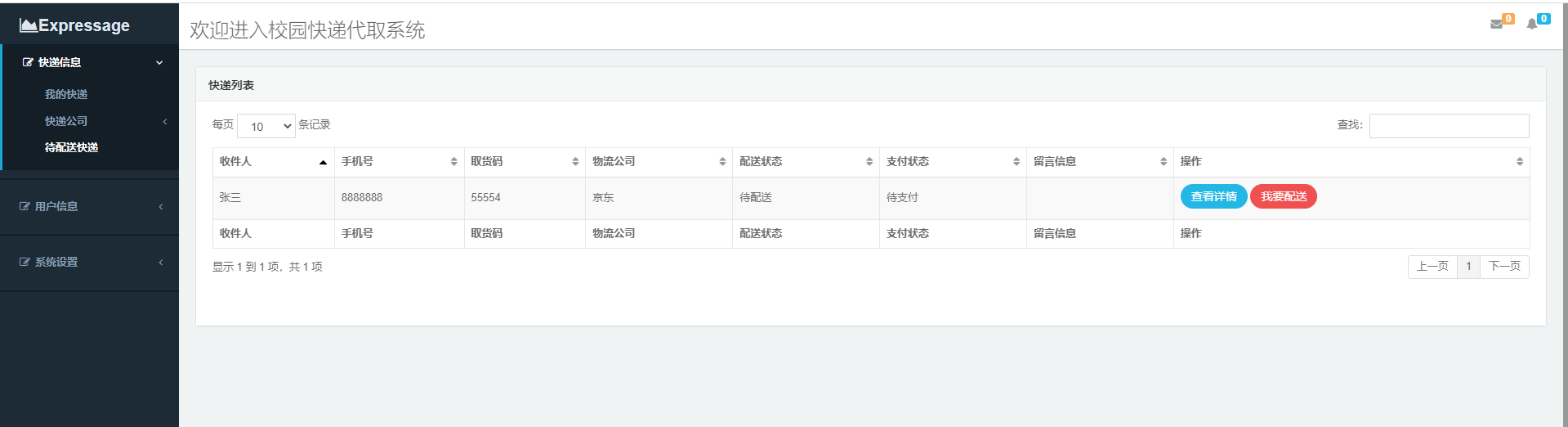


图5.11 待配送快递列表页面



图5.12 配送功能

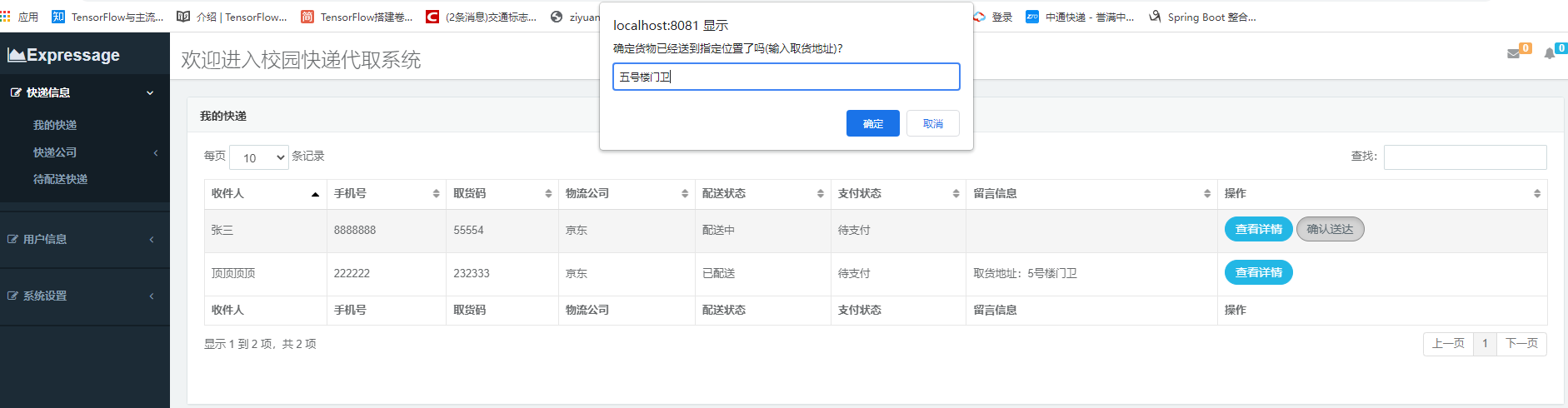
配送员将快递送达到位置后，可以对快递进行“确认送达”操作，点击确认送达按钮后，页面会提示输入取货地址，输入完成后，后台修改快递的状态为“已配送”。这是用户就可以对该快递进行收货操作。具体实现效果如图5.13所示：

图5.13 确认送达功能

配送员可以通过快递公司进行快递的数据筛选，快递员点击快递公司，点击相应的物流公司名，页面显示该物流公司的快递信息。具体实现效果如图5.14所示：



图5.14 快递公司数据筛选功能

## 5.2.3 管理员快递信息模块

管理员进入快递信息模块后，可以查看所有的快递信息，可以查看所有的快递详情，也可以根据快递公司进行数据的筛选。对于快递公司，管理员有特有的权限，可以添加快递公司，删除快递公司。

管理员添加快递公司后，系统自动增加快递公司的菜单和选项，方便快递员的查询和确保用户输入快递信息时可以选择新增的快递公司。具体实现效果如图5.15、5.16所示：

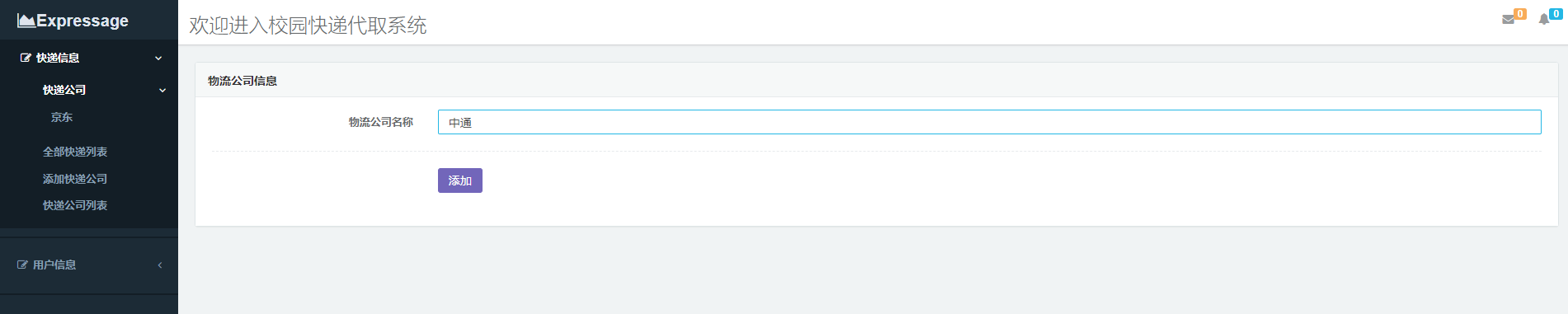


图5.15 添加快递公司功能

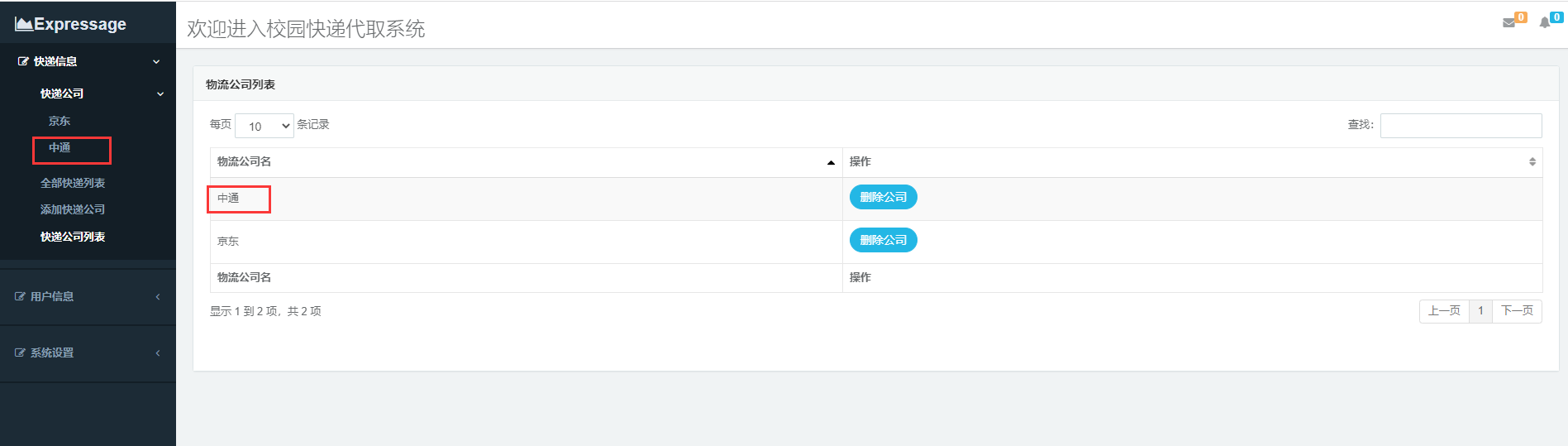


图5.16 添加快递公司后的效果

管理员对公司还是删除功能，点击快递公司列表的删除功能，后台删除数据库中该快递公司的数据，前台不再展示该快递公司数据。

## 5.3用户信息模块

用户信息模块主要功能是用户信息的修改，对于学生和配送员来说，该功能主要是修改用户名和密码，对于管理员来说，可以新增用户，删除用户，修改用户，删除用户后，该用户无法再使用该账号登录平台，修改用户，管理员可以修改用户的角色，从而是他具备不同的权限。具体实现效果如图5.17、5.18、5.19所示。

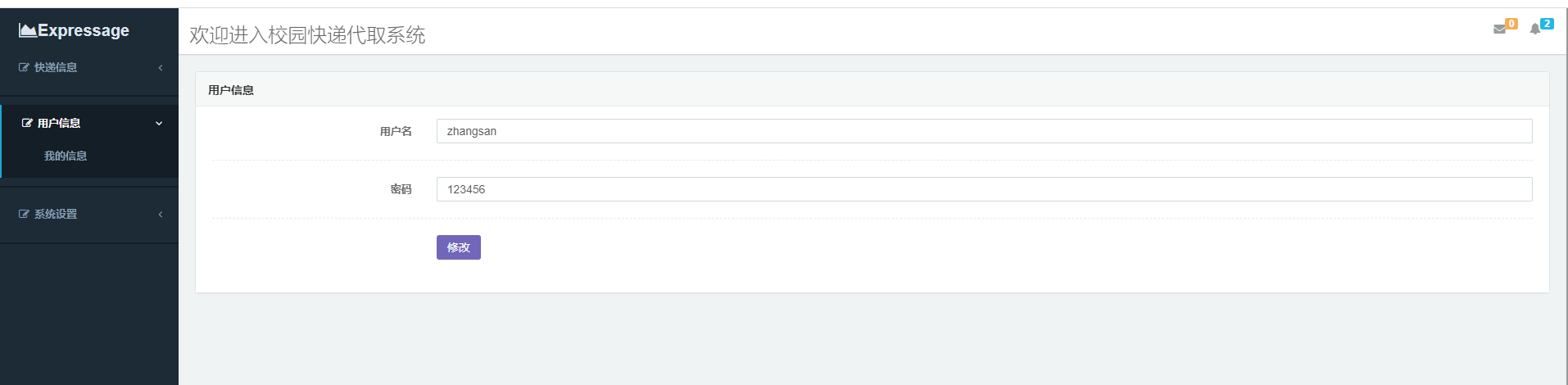


图5.17 学生和快递员用户信息模块

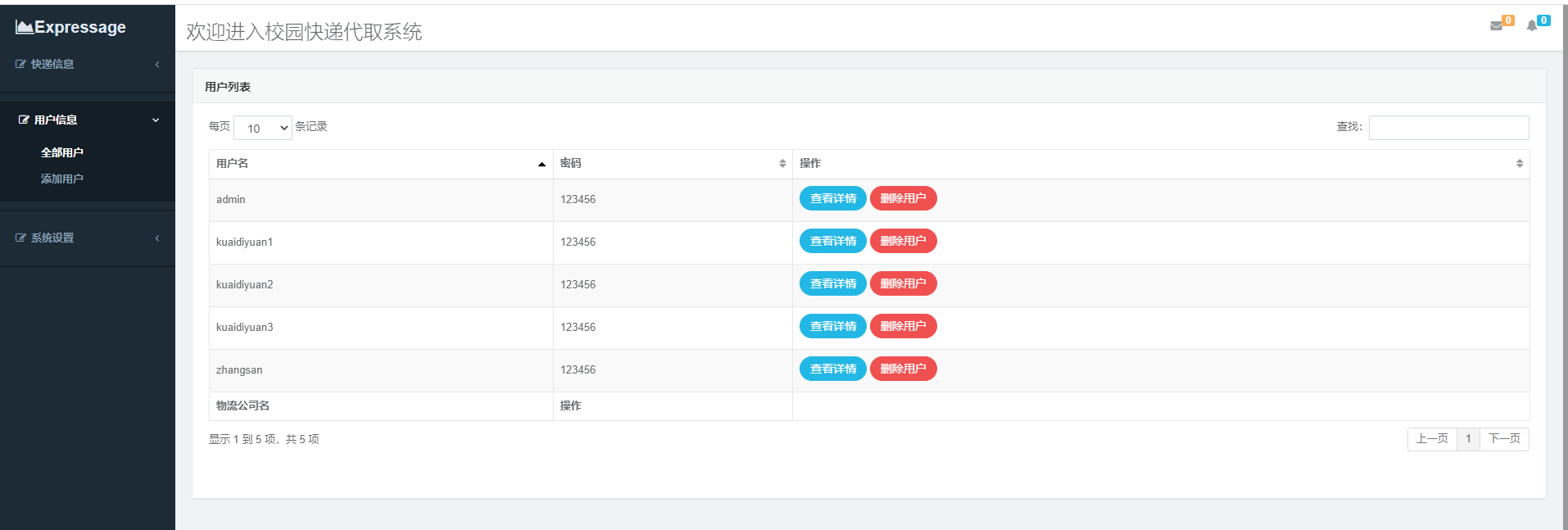


图5.18 管理员查看所有用户

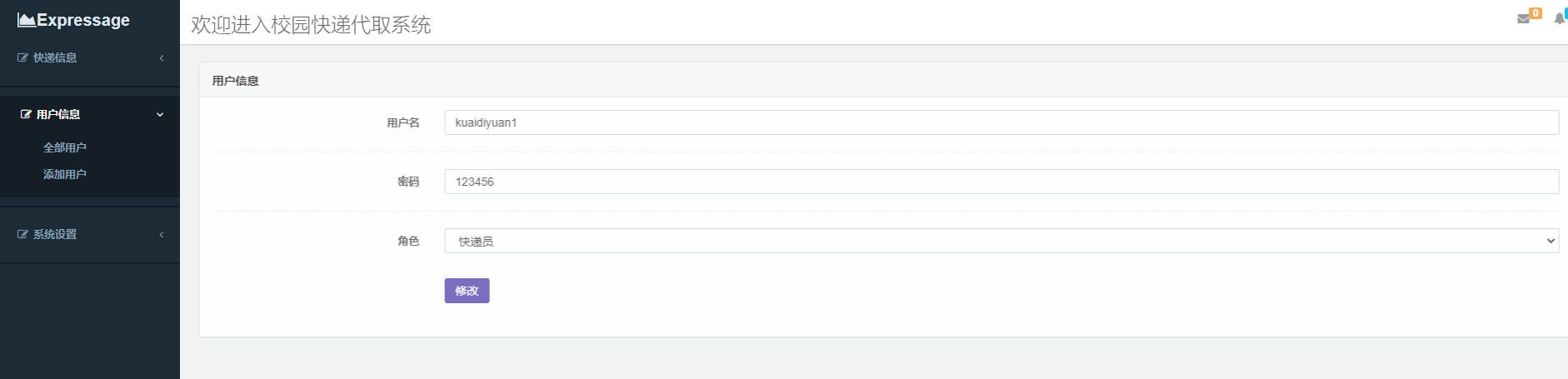


图5.19 管理员修改用户信息

## 5.4 系统设置模块

该模块主要功能是用于用户退出系统，而对于管理员来说，他具有系统新闻增删改查的权限，可以发布新闻，修改新闻，删除新闻。发布新闻后，用户可以在首页查看到系统新闻来了解最近有什么折扣活动。

管理员进入添加新闻页面，输入新闻的标题和内容，点击添加按钮，后台将前端的数据保存到数据库中，当用户登录系统时，读取数据库中新闻的数据展示到首页面中。具体实现效果如图5.20、5.21所示：

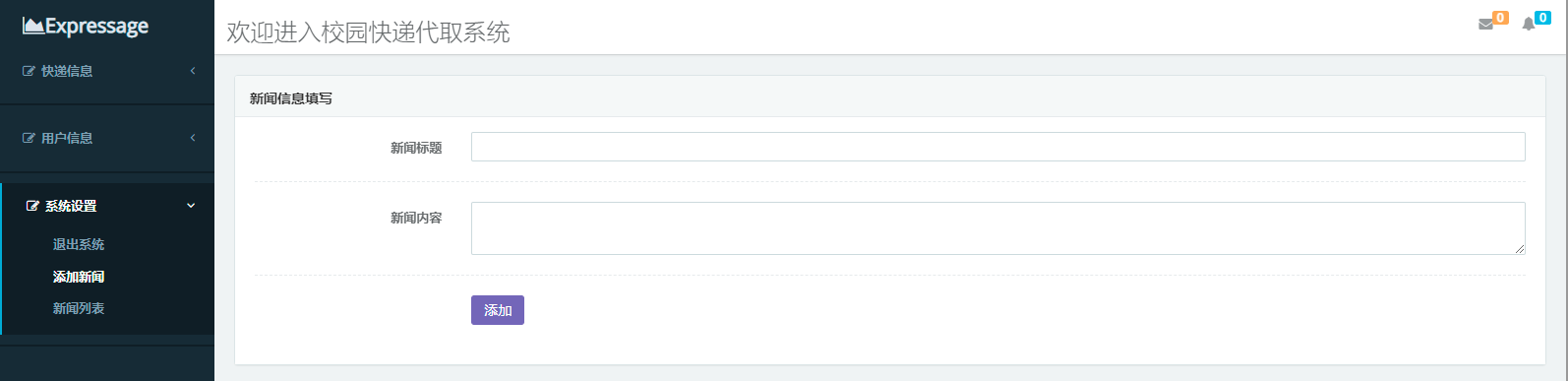


图5.20 添加新闻页面

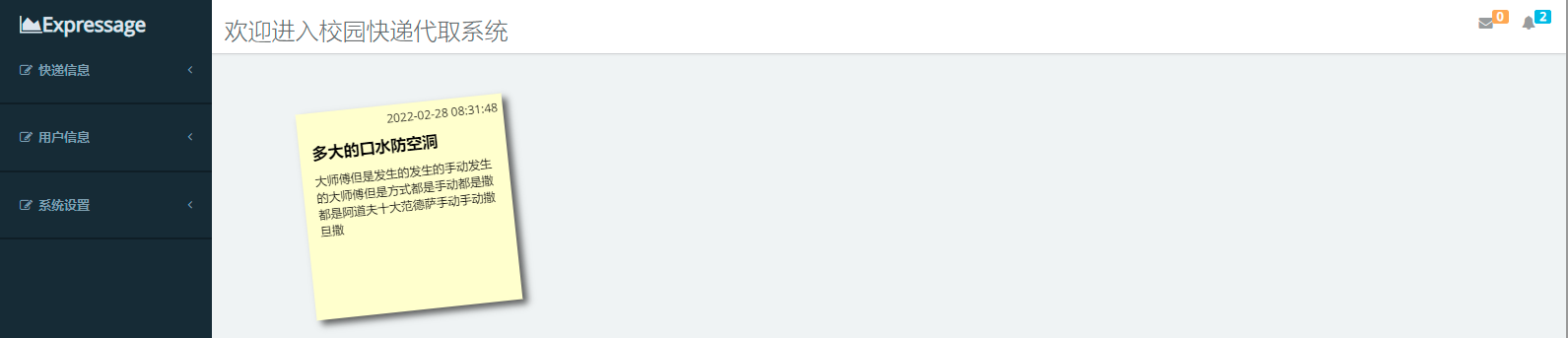


图5.21 首页新闻展示功能

## 5.5 其他功能模块

## 5.5.1 留言功能模块

配送员确认送达快递后，输入取货地址，后台系统自动添加留言信息，通过快递ID获取到本快递所属的用户ID，然后和数据库进行交互，将留言存入到数据库中，等用户登录到系统后，在留言板块可以查看留言信息。具体实现效果如图5.22、5.23所示：

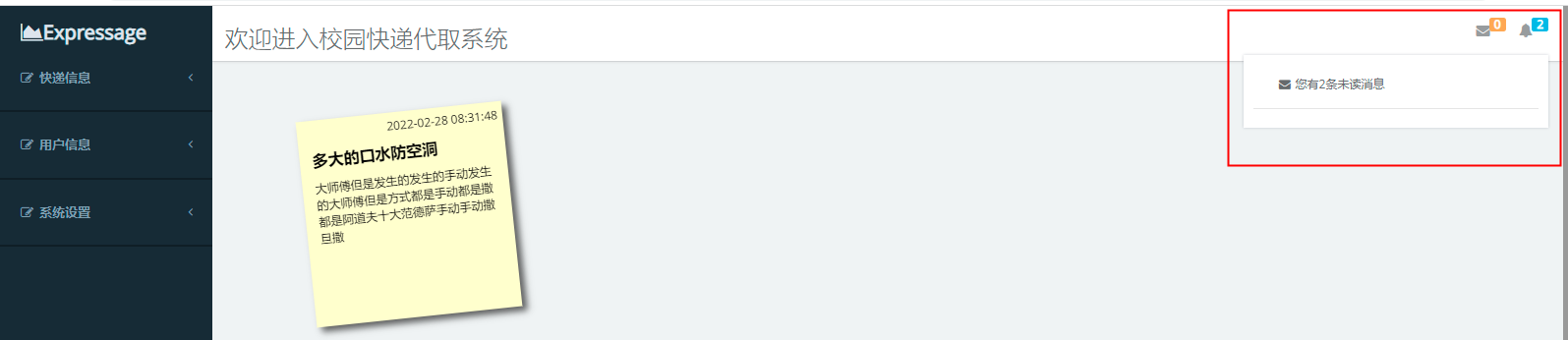


图5.22 留言板块

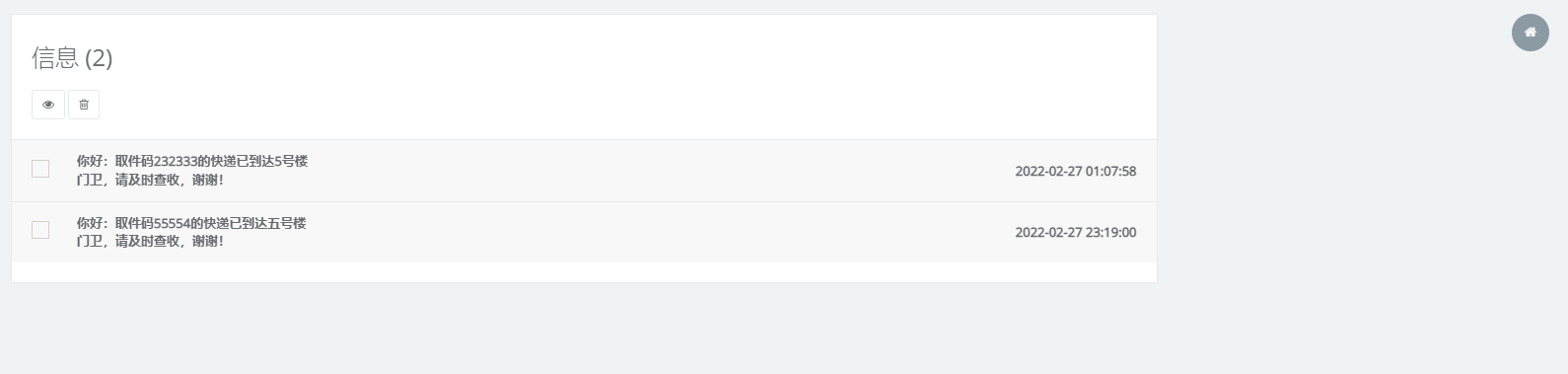


图5.23 留言页面

用户可以对信息标记已读状态，选择留言大，点击标记已读按钮，后台根据留言ID修改本留言的读取状态为“已读”。用户选择留言信息，点击删除按钮，后台根据留言ID删除数据库中该留言数据。

## 5.5.2 首页功能模块

首页功能模块对于学生和快递员的使用场景主要是查看系统的新闻，对于管理员来说，可以查看系统本月所有快递单数，已完成的快递数和待配送的快递数，并且系统会核算本月的交易额显示到页面中。具体实现效果如图5.24所示：



图5.24 管理员首页信息展示

# 系统测试

## 系统测试环境

软件环境：Windows10操作系统，Mysql5.7数据库，JDK1.8.0\_241，谷歌浏览器。

硬件环境：计算机一台。

## 系统功能测试

校园快递待取系统主要使用黑盒测试技术进行测试，验证系统中各个模块功能的可用性，本章测试内容主要对系统基础功能进行测试，包括登录注册、快递信息查询、配送快递、确认收货和支付快递费功能。

### 登录注册模块功能

本部分的测试内容主要为用户登录功能和用户注册功能，具体测试用例如表6.1所示：

表6.1登录注册功能测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 测试名称 | 前置条件 | 测试步骤 | 预期结果 |
| 1 | 登录功能测试 | 系统正常 | 1. 进入系登录页面 2. 用户名输入为空，点击登录按钮。 3. 密码输入为空点击登录按钮。 4. 用户名和密码都为空，点击登录按钮 5. 用户名和密码框输入正确的用户名密码 | 1. 成功进入登录也买你 2. 页面提示“用户名密码错误” 3. 页面提示用户名密码错误 4. 页面提示用户名密码错误 5. 成功进入系统首页 |
| 2 | 用户注册功能测试 | 系统正常 | 1. 进入用户注册页面。 2. 输入用户名和密码。 3. 进入登录界面输入用户名和密码。 | 1. 成功进入注册页面。 2. 用户名密码输入成功。 3. 登录成功，成功进入系统首页。 |

### 快递信息查询功能

本部分内容主要是对快递信息查询功能进行测试，具体测试用例如表6.2所示：

表6.2 快递信息查询测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 测试名称 | 前置条件 | 测试步骤 | 预期结果 |
| 1 | 学生登录查看快递信息 | 系统正常 | 1. 学生使用账号登录系统。 2. 点击我的快递菜单 | 1. 成功登录系统 2. 查询的数据都是本账号的快递单。 |
| 2 | 快递员登录查看快递信息 | 系统正常 | 1. 快递员使用账号登录系统。 2. 点击我都快递菜单。 | 1. 成功登录系统 2. 查询的数据都是本账号配送的快递单数据。 |
| 3 | 管理员登录查看快递信息 | 系统正常 | 1. 管理员使用账号登录系统。 2. 点击全部快递菜单。 | 1. 成功登录系统 2. 查询的数据是全部快递单数据 |

### 配送快递功能

本部分内容主要是对配送快递的权限，配送员配送的功能进行测试，具体测试用例如表6.3所示：

表6.3 配送快递功能测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 测试名称 | 前置条件 | 测试步骤 | 预期结果 |
| 1 | 配送快递权限功能 | 系统正常 | 1. 学生账号登录系统点击我的快递菜单 2. 快递员账号点击快递列表菜单。 3. 点击我要配送按钮。 | 1. 学生成功进入我都快递菜单且没有我要配送的操作按钮。 2. 快递员成功进入快递列表菜单且有我要配送操作按钮。 3. 点击我要配送后，快递的状态变成配送中。 |
| 2 | 快递员确认送到功能 | 系统正常 | 1. 快递员进入我的快递页面，点击配送中快递单的确认送达按钮。 2. 输入收货地址，点击确定。 | 1. 成功进入我的快递页面，点击确认送达按钮后，页面弹出请输入取货地址弹窗。 2. 输入收货地址点击确认后，快递单状态更改成已配送。 |

### 确认收货和支付快递费功能

本部分内容主要是对学生收取快递，支付快递费的功能进行测试，具体测试用例如表6.4所示：

表6.4 确认收货功能测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 序号 | 测试名称 | 前置条件 | 测试步骤 | 预期结果 |
| 1 | 学生收取快递功能测试 | 系统正常 | 1. 学生进入我的快递页面。 2. 点击已完成快递的确认收货按钮。 | 1. 成功进入快递页面。 2. 页面提示“确定快递送到手里了吗”的提示语 |
| 2 | 学生支付快递费功能测试 | 系统正常 | 1. 点击已完成快递的确认支付按钮。 2. 在支付宝支付页面输入账号和支付密码支付。 | 1. 页面跳转到支付宝支付页面。 2. 输入成功，扣款成功，快递单状态变成已支付。 |

# 总结

校园快递待取系统通过对快递市场的调研，对学生收取快递的需求做了实践的分析，最终确定了本系统的需求，对学生而言需要了解到快递的实时状态，快递送达后要有明确的感知；对于配送员来说，需要知道自己有多少快递已完成，有多少快递在配送中，并且了解有多少快递还没有进行配送；对于管理员来说，要对整个系统有明确的管理，对本系统的订单有统计的能力，最终将本系统分成了快递信息，用户信息，系统设置三个大模块。

该系统融合了各大物流公司，实现了校园快递物流的统一管理，缓解了目前校园快递代收点乱，取货时间短，快递积压的情况，大大提高了校园统一管理作用，使校园的氛围更加的偏向于学术。

# 参考文献

[1]沙之洲.Java编程语言在计算机软件开发中的应用[J].电子世界,2021,(24):125.127.

[2]李雅琴.(2021).Java编程语言的优势及其应用实践研究. *互联网周刊*(24),60.62. doi:  
[3]朱姝.(2021).Java程序设计语言在软件开发中的运用初探. *电子测试*(21),72.74. doi:10.16520/j.cnki.1000.8519.2021.21.025.  
[4]徐飞龙.(2021).JFINAL框架在Java web开发中的应用. *时代汽车*(19),27.28. doi:

[5]张梦元.(2021).基于技术演进的Java web程序设计教学实践. *信息与电脑(理论版)*(13),230.232. doi:  
[6]巢晟盛.基于SpringBoot微服务架构下前后端分离的MVVM模型浅析. *电脑知识与技术*(23),128.129+141. doi:10.14004/j.cnki.ckt.2021.2412.

[7]乔镔 & 隋首钢.(2021).Design of Background Management System of Physical Examination Reservation in School Hospital Based on SpringBoot. *软件工程与应用*(06), doi:10.12677/SEA.2021.106073.

[8]Jian Chen,Chen Jian & Pan Hailan.(2020).Design of Man Hour Management Information System on SpringBoot Framework. *Journal of Physics: Conference Series*(1), doi:10.1088/1742.6596/1646/1/012136.

# 致谢

感谢导师在毕业设计阶段对我的指导，在论文选题，设计指导，具体功能实现等方面给了我很多的建议，帮助我分析系统整体设计模式。在论文的书写方面，结构方面所注意的格式，做出了明确的建议和改正。最终感谢大家的关心和帮助。