

电 子 科 技 大 学

UNIVERSITY OF ELECTRONIC SCIENCE AND TECHNOLOGY OF CHINA

# 专业学位硕士学位论文

MASTER THESIS FOR PROFESSIONAL DEGREE

(电子科技大学图标)

论文题目	基于 Android 系统下 快递配送软件的设计与实现
专业学位类别	工程硕士
学 号	201191020403
作 者 姓 名	雷 鹏
指 导 教 师	曹宗杰 副教授

分类号

密级

UDC <sup>注1</sup>

# 学 位 论 文

基于 Android 系统下快递配送软件的设计与实现

(题名和副题名)

雷 鹏

(作者姓名)

指导教师

曹宗杰

副 教 授

电子科技大学

成 都

常 军

高级工程师

中国电子科技集团公司第十研究所 成 都

(姓名、职称、单位名称)

申请学位级别

硕士

专业学位类别

工程硕士

工程领域名称

软件工程

提交论文日期

2014 年 3 月

论文答辩日期

2014 年 5 月

学位授予单位和日期

电子科技大学

2014 年 6 月 28 日

答辩委员会主席

评阅人

注 1: 注明《国际十进分类法 UDC》的类号。

---

# **DESIGN AND IMPLEMENTATION OF EXPRESS DISTRIBUTION SOFTWARE BASED ON ANDROID SYSTEM**

A Master Thesis Submitted to  
University of Electronic Science and Technology of China

Major: **Master of Software**

Author: **Lei Peng**

Advisor: **Associate Professor Cao Zongjie**

School : **School of Electronic Engineering**



---

## 独创性声明

本人声明所呈交的学位论文是本人在导师指导下进行的研究工作及取得的研究成果。据我所知，除了文中特别加以标注和致谢的地方外，论文中不包含其他人已经发表或撰写过的研究成果，也不包含为获得电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同志对本研究所做的任何贡献均已在论文中作了明确的说明并表示谢意。

作者签名：\_\_\_\_\_

日期：    年    月    日

## 论文使用授权

本学位论文作者完全了解电子科技大学有关保留、使用学位论文的规定，有权保留并向国家有关部门或机构送交论文的复印件和磁盘，允许论文被查阅和借阅。本人授权电子科技大学可以将学位论文的全部或部分内容编入有关数据库进行检索，可以采用影印、缩印或扫描等复制手段保存、汇编学位论文。

（保密的学位论文在解密后应遵守此规定）

作者签名：\_\_\_\_\_

导师签名：\_\_\_\_\_

日期：    年    月    日



## 摘 要

面对日益火爆的网络交易，其背后的快递配送服务首当其冲，面临着较大的压力。快递配送过程中为了保证快件到达的及时性、安全性以及收件人与快递员沟通的方便性，快递配送手持终端也必不可少<sup>[1]</sup>。

而随着网络普及化，3G 和 WIFI 使无线技术接入到互联网。传统的快递行业手工操作为主，以纸质档案形式保存资料，这样的工作效率极低而且人工成本差错率很高，严重影响企业的发展。Android 系统平台的异军突起，对于快递行业专用设备价格昂贵方面，为小型企业提供了发展方向，因此 Android 系统拥有引入成本低廉，用户体验开放性较强的特点，可以让 Android 系统手机成为低成本的用户终端设备。

推广基于 Android 平台的手持终端有着非常重要的意义。从客户的角度来看，手持终端为客户能够提供全方位的 24 小时跟踪查询服务，能够提高客户对快件使用的信任度，增强客户对公司的信任，并建立长期的合作关系，方便客户更好的利用公司的服务，以促进良性循环。

本文对快递配送软件系统在设计过程中的关键技术进行了研究与分析，着重介绍了 JAVA 开发方面的相关技术，诸如：STRUTS 框架、SPRING 框架、iBATIS 框架、AJAX、JAVASCRIPT，以及 SQL Server 相关技术和 C/S、B/S 开发模式，接着从业务、功能性以及非功能性需求三方面对整个系统进行了需求分析，并针对需求设计整个快递配送系统的结构以及根据功能设计各模块，最后给出了软件的关键代码以及实现和测试的实景图片。

**关键词：**快递配送，手持终端，Android

## ABSTRACT

Faced with an increasingly popular online trading, express delivery services behind the brunt facing greater pressure. Express delivery process in order to ensure timely shipment arrived, security and communication between the recipient and the convenience of courier, express delivery handheld terminal is also essential.

With the popularity of the Android platform network also sprung up everywhere. Android system has introduced low-cost, user experience, openness strong features, it can be a low-cost Android phone user terminal equipment.

Promotion handheld terminal based on the Android platform has a very important significance. From the customer's point of view, handheld terminals to customers 24 hours to provide a full range of tracking and tracing services that improve customer trust express used to enhance customer confidence in the company and establish long-term cooperative relationship, to facilitate better customer the use of the company's services to promote a virtuous circle.

In this paper, express delivery software system in the design process of the key technology research and analysis focuses on the JAVA development of related technologies, such as: STRUTS framework, SPRING framework, iBATIS framework, AJAX, JAVASCRIPT, and SQL Server and related technologies C/S, B/S development model, and then from the business, functional and non-functional requirements tripartite face the whole system needs analysis and design of structures for the needs of the entire distribution system and delivery according to the functional design of each module, and finally to a key source software as well as implementation and testing of the real picture.

**Keywords:** Express Delivery, Handheld terminal, Android



## 目 录

<b>第一章 绪论 .....</b>	<b>1</b>
1.1 论文研究背景.....	1
1.2 研究目的和意义.....	2
1.3 国内外研究现状.....	3
1.4 章节安排.....	4
<b>第二章 关键技术介绍 .....</b>	<b>5</b>
2.1 快递配送软件系统概述.....	5
2.2 JAVA 开发相关技术介绍 .....	5
2.2.1 关于 JAVA .....	5
2.2.2 STRUTS 框架 .....	6
2.2.3 SPRING 框架 .....	7
2.2.4 iBATIS 框架 .....	9
2.2.5 AJAX .....	9
2.2.6 JAVASCRIPT .....	10
2.2.7 Android 系统优势 .....	10
2.3 SQL SERVER 2005 技术介绍 .....	11
2.4 C/S 和 B/S 开发模式 .....	12
2.4.1 C/S 和 B/S 开发模式简介 .....	12
2.4.2 C/S 和 B/S 开发模式对比 .....	13
2.5 本章小结.....	13
<b>第三章 系统需求分析 .....</b>	<b>14</b>
3.1 业务需求.....	14
3.1.1 用户角色分析 .....	14
3.1.2 核心业务流程 .....	16
3.2 功能需求.....	17
3.2.1 系统管理功能 .....	17
3.2.2 快递终端功能 .....	18
3.3 非功能性需求.....	18
3.3.1 系统性能需求 .....	18

3.3.2 系统运行环境需求 .....	18
3.4 本章小结 .....	19
<b>第四章 系统设计 .....</b>	<b>20</b>
4.1 系统设计总则 .....	20
4.2 系统总体设计 .....	20
4.2.1 系统结构 .....	20
4.2.2 系统服务器开发框架 .....	20
4.3 系统功能设计 .....	21
4.3.1 系统管理功能 .....	22
4.3.2 快递终端管理 .....	22
4.3.3 快递单任务管理 .....	23
4.3.4 统计分析管理 .....	23
4.4 数据库设计 .....	25
4.4.1 系统数据库关系图设计 .....	25
4.4.2 数据库表设计 .....	26
4.5 系统安全性设计 .....	28
4.6 本章小结 .....	28
<b>第五章 系统测试和实现 .....</b>	<b>29</b>
5.1 系统测试 .....	29
5.1.1 服务器性能测试 .....	29
5.1.2 终端性能测试 .....	29
5.1.3 系统功能测试 .....	30
5.1.4 业务流程测试 .....	31
5.2 软件实现 .....	34
5.2.1 添加快递单 .....	34
5.2.2 查询快递单 .....	35
5.2.3 编辑快递单 .....	36
5.2.4 删除快递单 .....	37
5.2.5 分发快递单 .....	37
5.2.6 添加手机终端 .....	38
5.2.7 分配组权限 .....	39
5.2.8 分配用户到组 .....	41
5.2.9 生成派件二维码 .....	43

5.2.10 手机端实现 .....	44
5.2.11 主菜单界面 .....	44
5.2.12 扫描取件 .....	45
5.2.13 发送签收短信 .....	47
5.2.14 派送签收及拍照上传 .....	49
5.2.15 查看任务 .....	51
5.3 本章小结 .....	51
<b>第六章 总结与展望 .....</b>	<b>52</b>
<b>致 谢 .....</b>	<b>53</b>
<b>参考文献 .....</b>	<b>54</b>



## 第一章 绪论

### 论文研究背景

自 2009 年 11 月 11 日起, 淘宝率先联合 27 家店铺以“五折促销”营造的了“双 11 抢购节”, 这场淘宝营销方案发展迅速, 京东、苏宁、腾讯等也随之加入。前不久的“双 11”, 截至 11 月 12 日零点, 阿里旗下的天猫加淘宝的单日销售总额达到 350.19 亿元; 易迅网公布, 促销 8 天下单金额突破 10 亿元; 京东方面表示, 截止到 11 日中午 12 点, 订单量达到平日全天水平, 网站流量是平日的 2.5 倍; 苏宁方面表示, 移动客户端销售同比增长了 10 倍。

在这巨大的蛋糕面前, 越来越多的人想占据一席之地、分一杯羹, 亚马逊、当当网、唯品会、国美在线、1 号店、凡客诚品、小米、新蛋等等也都步入了电商的行列, 在这欣欣向荣的表面背后, 快递物流行业也随之发生着翻天覆地的变化<sup>[2]</sup>。

据国家邮政局监测, 今年“双 11”期间(11 日—16 日), 国内全行业处理的邮件(快件)业务总量为 3.46 亿件, 比去年“双 11”同期增长 73%。快件单日最高业务量出现在 11 月 13 日, 为 6517 万件, 比去年增长 85%。各地快递纷纷出现了爆仓现象, 消费者也反映“双 11”的快递运送速度比平时慢了两三天。而最苦恼的应该算是快递员了, 要配送如此多的快件, 对他们来说, 是相当大的负担<sup>[3]</sup>。

阿里巴巴董事局主席马云说过, 我们把互联网当作是一场革命。今年“双 11”用令人震撼的销售业绩给传统零售业、物流业以及更多的服务型行业再次敲响了警钟。

数据显示, 2012 年在我国兴起的网络零售市场上, 共交易规模超过 1.3 万亿元, 同比增长 67.5%左右, 增长速相当于当年社会消费品零售总额增长率的 4.7 倍。跟据此现状预计今年全年销售额占我国整体社会消费品零售总额的比重将超过 10%<sup>[4]</sup>。

其实, 无论 10 年后零售市场份额究竟如何划分, 电商对于实体经济的影响早已跨越了多个行业。电商的本质不是将产品转移到网上去销售, 而是对供应商、物流渠道以及用户进行整合, 围绕用户需求, 将大流通与大制造进行对接。

由此看到如今的电商不再将注意力全部集中到“造节”和价格战本身, 而是将更多的功夫用于提升物流、服务等环节<sup>[5]</sup>。的确, 现在的“双 11”已经发出了经济转型信号, 也掀起了新经济、新的商业模式对传统商业模式的大战。

互联网毫不留情地把传统渠道不必要的环节省略了, 让服务商和消费者, 让生产制造商和消费者直接对接, 让厂商和服务商近距离接触消费者。消费者的喜

好、反馈可以很快速地通过网络来反映。从目前来看，各式各样的转型都在发生，对传统零售业带来巨大压力的同时，电子商务也在通过 O2O 实现更深层面的融合。与此同时，在制造业、物流业等各项领域，电子商务也因此显示出对传统行业颠覆与重构的力量<sup>[6-8]</sup>。

由此，互联网不仅颠覆了零售业，也在迅速改变着传媒业、娱乐业甚至已经在影响着曾经高不可攀的金融业。那些还没有顺应潮流的企业，很可能会骤然倒下，因为巨人稍微没有跟上形势就可能再也不是巨人了。

互联网日新月异，电商手段层出不穷，而想要在这个淘汰制的社会中经久不衰，就必须适应这种变革。快递物流所面临的挑战是无论在什么情况下，都必须把快件快速准确的投递到消费者手中。

“巴枪”的诞生正是帮助了快递员在一定程度上解决了这一烦恼。巴枪，这个普通人可能闻所未闻的专业设备，却是任何一个快递员都再熟知不过的必备“神器”。

它对于信息化技术在快递物流行业的转型升级、打造核心竞争力有着重要意义。

## 研究目的和意义

随着网络普及化 Android 系统平台也异军突起。Android 系统拥有引入成本低廉，用户体验开放性较强的特点，因此 Android 系统手机可以成为低成本的用户终端设备<sup>[9]</sup>。

推广手持终端的意义。从客户的角度来看，手持终端为客户能够提供全方位的 24 小时跟踪查询服务，能够提高客户对快件使用的信任度，增强客户对公司的信任，并建立长期的合作关系，方便客户更好的利用公司的服务，以促进良性循环。

对快递公司来讲，实现收快件环节的数据采集，为盈利管理提供基础数据信息，加强收快件的管理，对收快件时效全程监控，提高营运保护水平，为后续的业务流程优化提供信息基础。从而优化业务流程，便于作业指导，为快递公司的个性化的增值服务提供信息基础，进一步提升了服务质量，提升快递公司在同行中社会品牌知名度。

使用手持终端的好处，在整个快件的全生命周期，大部分关键环节都要用到手机终端，从处理场所到收快件环节都在进行广泛的运用。通过移动信息化水平的逐步的提高，一方面降低了管理成本，再一者提升了客户的满意度，第三就是安全货件管理，第四个就是安全及有效的现金管理，第五个是增加企业竞争力。

我国现有速递行业中，大都为单独购置手持式数据采集器，价格都在 3000 元左右，高端产品采用 Microsoft Windows CE 6.0 系统的手持机更是价格昂贵，因此

Android 系统手机在其应用上可扩展性强，成本较低，可作快递企业中的一个应用模块加以使用。

随着网络普及化 Android 系统平台也异军突起。Android 系统拥有引入成本低廉，用户体验开放性较强的特点，因此 Android 系统手机可以成为低成本的用户终端设备。主要特点为初期投入低，后期维护简单。速递人员只需带上安卓系统手机安装客户端软件就可以完成物品配送工作。

### 国内外研究现状

国内外对快递配送系统的管理主要集中在数据终端硬件厂商身上，手持终端硬件和软件高度耦合，不同的手持终端对应着不同的快递配送管理系统。而这些手持终端硬件厂商又有着自己的一套标准，不能完全统一。这就是快递员们熟知的“巴枪”，即快递手持式终端。

在国内，优博讯科技股份有限公司占据了手持终端市场 60% 的份额，公司的 i3000 系列、V5000 系列、i6000 系列、i6100 系列都广泛应用于物流快递、连锁零售、医药卫生、政府与公共事业、金融、烟草等行业，它们轻巧坚固，外观设计符合人体工程学，能带给用户安全舒适便利的操作体验。卓享科技思必拓品牌的 MT35A、MT02A，致远电子的 HPC3000、HPC6000，innotel 的 7530、7527 等或在耐久方面，或在外观上、或在软件使用舒适度上有着各自的优势。

国外鉴于人工成本高，故而很多知名大公司或者机构，比如美国德州仪器公司（TI）、日本的 intermec、datamaxcorp、casio 公司等，在手持终端设备方面较早就进行了投资与研究，并且也取得了很好的效果，不论在软件集成、耐用方面还是屏幕显示方面都领先于我国绝大多数同类产品。

消费者都盼望自己网购的产品能快速邮寄到自己家中，故而不少网购消费者有在网上查询快件“走到哪里”的经历，而要做到这一点，就需要快递公司在收寄、分拣、运输、投递等各个环节对快件信息做好记录。

以前，这项工作主要靠人工完成——从快递员收取快件到送至营业网点，再到网店派人将快件面单号一一敲进电脑、录入系统，仅仅一个环节就耗时两三个小时，不仅费时费力，而且差错率也较高。

随着业务量的暴增，手再快的人也忙不过来了，于是“自动录入”的新方式逐渐成为主流。现在，快递员或者网点只需使用扫码设备轻轻瞄准快件面单上的二维码一扫，“嘀”的一声，快件信息分分钟就上传到系统中了。

而巴枪，就是体积最灵巧、应用最广泛的一种手持扫码终端。多亏有了巴枪，缩短了接单、派单的时间，提高了投递准确率。

早在五年前，中通就开始使用手持终端，但受制于性能和价格等因素，应用效果并不明显。现如今快递一线人员也认识到手持终端的重要性，目前使用率已经达到 90% 左右。

不能否认，巴枪的产生确实令快递物流行业有了较快的提速，但国内外百花齐放的巴枪种类也给各快件在快递公司的衔接、巴枪更新换代的速度与需求、快递员对巴枪使用的个人喜好产生了一定的限制<sup>[10]</sup>。

## 章节安排

本论文是根据课题任务要求完成快速配送软件系统的系统软件和服务器，在 android 平台上，利用 JAVA 编程语言结合 spring+struts 框架，前台使用 JSP 和 javascript 等技术进行编写程序，完成数据输入、修改、存储、调用查询等功能；并使用 SQL Server 2005 数据库形成数据，通过网络进行转发到后台服务器进行数据存储。

本论文主要分为以下七个章节进行论述，具体如下：

第一章：绪论，主要分析国内外快递配送手持终端系统的国内外研究现状以及发展趋势，并简述研究背景与意义，对章节的安排做了论述。

第二章：关键技术介绍，简单介绍快递配送软件系统，并对 JAVA 开发中所应用到的 STRUTS 框架、SPRING 框架、iBATIS 框架、AJAX、JAVASCRIPT 以及 Android 系统等主要关键性技术进行了介绍，接下来介绍了系统中信息存储比较重要的 SQL SERVER 2005 技术，最后对比了 C/S 和 B/S 开发模式。

第三章：系统需求分析，首先进行业务需求分析，提取出用户角色、以及核心业务流程。接着从功能方面进行需求分析，比如系统管理功能、快递派送人员管理功能、线路导航、货物状态跟踪、客户签收定位、签收单影像资料保存。最后对系统性能、系统运行环境、系统安全性等非功能性需求进行分析。

第四章：系统设计，首先介绍该快递配送软件系统的设计总则，然后对系统结构及架构设计进行了阐述。接着从功能性出发，对各大模块进行了具体的设计，包括系统管理功能、快递终端管理、快递单任务管理、统计分析管理、数据库设计、安全性设计。

第五章：系统实现与测试，论文分为两部分，分别从客户端和服务端详细介绍了各功能模块的设计与实现。系统测试，从不同角度对快递配送软件系统进行测试，并总结测试结果。

第六章：总结与展望，对本文中快递配送软件系统实现时所采用的算法以及方法进行了总结。最后，针对快递配送软件系统中存在的问题以及待改进之处进行了展望。



## 第二章 关键技术介绍

### 快递配送软件系统概述

快递配送软件系统，是一个由用户、计算机及货物等组成，对业务流程进行相关信息的收集、存贮、统计的系统。它结合了管理科学、计算机科学、统计学的相关知识，用计算机应用实现现代化标志，结合数学及统计学的相关方法及观点。根据以上观点此软件开发将结合 MIS 相关理念进行结构化整合。

MIS 即 Management Information System 管理信息系统。

#### (1) 系统建设简介

每一个成功运行的系统都会有出现、发展、完善、改变的过程，这一过程即为系统的生命周期。在建立系统时，了解系统生命周期成为非常重要的工作。在进行系统管理，系统开发时有着举足轻重的地位。从系统开发到结束这一过程就是该系统的开发生命周期。

为了让系统更加科学化、规范化，保证在进行系统建设时按照规定要求进行管理，所以需要按照生命周期相关要求，使用结构化方法。采用结构化方法可以大幅提开系统开发的成功率，但这也需要以密集型的劳动成为主要工作；需要花费更长的时间，当客户需求变动时，很难完全满足客户的所有要求。

#### (2) 关于结构化生命周期的简介

在开发 MIS 时，主要流行的方法即结构化生命周期。此方法为了将整个系统开发分为不同阶段，在各个阶段实施不同的行为，以采用各种技术实现相关任务为目的，进而科学分类达到规范要求，从而提供出符合标准的相关产品。

MIS 开发需要按照以下规范：先逻辑后物理，用户参与等原则。其主要工作包括以下方面：第一，系统规划；第二，系统分析设计开发与实施；第三，系统运维；第四，用户评价。

结合上述开发方法特点和快递配送软件系统的实际情况，整个系统以结构化生命周期方法为主进行系统分析和设计。

### JAVA 开发相关技术介绍

#### 关于 JAVA

JAVA 在经过许多年的发展之后，现在已经成长为一门严格并且主流的开发语言。JAVA 中的诸多概念如面向对像、代码复用性强、垃圾回收、跨平台等等，都

使得利用 JAVA 开发程序的过程更为简洁。由于当今电子设备的多样化以及普及化, 导致跨平台与网络功能将是未来软件的发展趋势, 而 JAVA 在此方面具有得天独厚的优势。JAVA 三大特性封装、继承、多态性以及垃圾回收机制<sup>[11]</sup>。

### 1. 封装

即可以理解为抽象, 它是处理各种问题的第一步, 封装的好处就是它能把具体的问题与具体的解决问题的方法结合起来, 这就是抽象的过程。

### 2. 继承

允许存在父类与子类之间存在某种关系, 一个子类有且只有一个直接的父类 (Java 中的继承是类之间的单继承)。

对于类的内容, 不允许多继承, 一个子类只能有一个直接的父类, 信息绑定的复杂性加上 Java 本身运行原理的限制, 使它不能像 C 语言一样高效的运行, 若绑定的信息复杂度较高, 数据的访问过程更加复杂, 影响运行效率。抽象类就是类, 接口就是接口, 都处于较抽象的层级而使它们具有相近的特性。

### 3. 多态

简单的说就是一个接口、多种实现, 可以将之理解为是一种语法上的允许, 表现形式多样化。Java 中的多态大多数是通过接口实现的, 而抽象类与子类之间的多态只占一小部分。

### 4. Java 的垃圾回收机制

Java 区别于 C++ 最明显的就是垃圾回收机制 (GC)。使用 Java 时, 不需要考虑内存的回收, 虚拟机会以某种特定的方式 (依算法而定) 对无效或过期的内存进行回收, 以达到对内存空间的最大利用<sup>[12]</sup>。

因此, 此软件开发时主要语言选择 JAVA 作为优先考虑<sup>[13]</sup>。

## STRUTS 框架

Struts 框架是目前主流的程序开发框架<sup>[14]</sup>, 它是在 WebWork 的技术基础上开发出来的全新的 MVC 框架。其中 Action 类是 struts 框架的核心部分, Struts 2 中的 Action 类不仅可以实现仅有的 Action 接口, 还可实现其他接口, 使其他定制的各类服务成为可能。Struts2 同时还提供一个 ActionSupport 基础类去实现常用的数据接口。Action 接口并不是必须使用的, 任何带有 execute 标识的普通 JavaBeans (POJO) 对象均可成为 Struts2 的 Action 对象<sup>[15]</sup>。它主要具备以下几个功能和特点, 如下表 2-1 所示:

表 2-1 Struts 框架部分特点

特点	详细阐述
Action 类	Struts 2 中的 Action 类不仅可以实现仅有的 Action 接口, 还可实现其他接口, 使其他定制的各类服务成为可能。Struts2 同时还提供一个 ActionSupport 基础类去实现常用的数据接口。Action 接口并不是必须使用的, 任何带有 execute 标识的普通 JavaBeans (POJO) 对象均可成为 Struts2 的 Action 对象 <sup>[15]</sup> 。
线程模式	Struts2 的 Action 完全杜绝了线程安全的问题, 因为他对每一个服务请求都会建立一个新的实例。
Servlet 依赖	Struts 2 的 Action 并不受到容器的限止, 可心 Action 没有容器的情况下进行测试。根据程序实际情况, Struts2 的 Action 仍可调用原始的 response、request。完全降低甚至取消其它直接访问 HttpServletResponse、HttpServletRequest 的必要性。
可测性	Struts 2 的 Action 因为实现“依赖注入”概念使得测试变得更加容易, 并且还可以通过初始化、方法的调用, 属性的设置等来实现测试。
捕获输入	Struts 2 直接使用 Action 属性作为输入属性, 消除了对第二个输入对象的需求。输入属性可能是有自己(子)属性的 rich 对象类型。Action 属性能够通过 web 页面上的 taglibs 访问。Struts2 也支持 ActionForm 模式。rich 对象类型, 包括业务对象, 能够用作输入/输出对象。这种 ModelDriven 特性简化了 taglib 对 POJO 输入对象的引用。
类型转换	Struts 2 的 Action 因为支持 Object Graph Notation Language 语言表达试进行类型的转换, 从而充当类型转换器的角色。
Action 执行的 控制	拦截器堆栈技术即 IntereceptorStracks, 它为每一个 Struts2 的 Action 设置不等的生命周期与不同的 Action 一起发挥作用从而起到对 Struts2 的 Action 进行控制。 <sup>[16][17]</sup> 。

## SPRING 框架

RodJohnson 所创的 Spring 框架是一个开放源代码的架构。它的产生解决了企业轻量级应用开发的复杂性。从易用性、解耦性和可测性而言 EJB 被 Spring 框架下使用 JavaBean 取而代之<sup>[18][19]</sup>。Spring 开源框架的 7 个模块通过 7 个明确定义的模块组成了 Spring 开源框架, 如下图 2-1 所示:

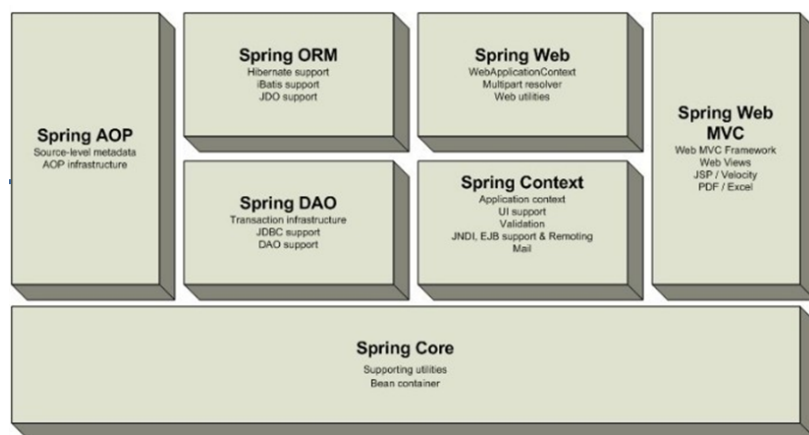


图 2-1 Spring 框架概览图

如果作为一个整体，Spring 框架提供了开发企业应用所需的一切。开发人员不仅可以直接使用整个 Spring 框架，也能从中挑选出几个模块进行组合使用。所有的 Spring 模块都是在核心容器之上搭建的。Spring 框架各模块介绍如下表 2-2 所示：

表 2-2 Spring 框架模块功能

模块名称	模块功能
核心容器	核心容器是 Spring 框架中最核心部分也是最基础的一部份，Bean 的管理交由容器来实现正是因为核心容器提供了依赖注入（DependencyInjection）。工厂模式是设计模式中的一个实现，要使 IOC 把应用程序的配置和说明从实际应用从代码中分离就需要用到 Spring 的核心容器 BeanFactory。
上下文应用文（Context）模块	Spring 框架要增加信息验证的支持、各类事件传播和国际化消息（I18N）等就需要通过上下文应用（Context）模块来扩展 BeanFactory 模块。使 Spring 成为一个容器的是 BeanFactory 核心模块，而上下文应用（Context）模块将使 Spring 成为一个框架。
Spring 的 AOP 模块	AOP 模块为 Spring 框架中解决了对面向切面编程的问题。成为了 Spring 框架中实现切面编程的基础。Spring 的 AOP 支持基于 AOP 联盟定义的 API，从而保证了其它 AOP 框架与 Spring 框架的通用性。
JDBC 抽象和 DAO 模块	JDBC 的使用通常会导致大量的重复代码，这对于一个应用来说是极其不好的，它的取得连接、创建语句、处理结果集，然后关闭连接等操作都是十分占据内存的，降低了程序运行的时间效率。而 Spring 的 JDBC 和 DAO 模块将这些重复代码提取并抽象出来，这样就可以保证在访问数据库的时候不会产生冗余代码。

## iBATIS 框架

iBATIS 是把 SQL 作为框架中心的持久化层设计。它能支持延迟加载、关联查询、继承等特性。iBATIS 有别于其他 OR 映射框架。OR 映射框架，将数据库表、字段等映射到类、属性，那是属于元数据（meta-data）映射。而 iBATIS 是把 SQL 查询的参数和结果集映射到类。所以，iBATIS 做的是 SQL 映射工作。它把 SQL 语句看成输入以及输出，结果集就是输出，而把 where 语句后面的条件作为参数输入。iBATIS 可以把输入的普通 POJO 对象、Map、XML 等映射到 SQL 的条件参数上，也可以把查询的结果映射到普通 POJO 对象、Map、XML 等上面。iBATIS 使用 xml 文件来映射这些输入以及输出<sup>[21]</sup>。

iBATIS 拥有的持久层框架主要由两个部分构成，其一是 SQL Maps，另外一部分则是 Data Access Objects (DAO)。与 Hibernate 和 Apache OJB 相比 iBATIS 属于“半自动化”的 ORM 实现。

1. 系统的部分或全部数据来自现有数据库，从安全角度考虑，只对开发团队提供几条 Select SQL 以获取所需数据，而具体的表结构则不予公开。

2. 在开发规范中要求，所有涉及到业务逻辑方面的数据处理，必须在数据库层交给存储过程来处理。

3. 系统数据处理量巨大，对其性能方面要求十分苛刻，这就意味着我们必须使用经过高度优化的 SQL 语句及存储过程才能达到系统性能设计的相关指标。

在这样的需求下 Hibernate 甚至无法使用。而“半自动化”的 ibatis，却正好解决了这个难题。ibatis 的“半自动化”，是相对 Hibernate 等提供了全面的数据库封装“全自动化”ORM 实现而言，“全自动化”ORM 确实实现了 POJO 和数据库表之间的映射，以及 SQL 的自动生成和运行。而 iBATIS 的特点，则在于 POJO 与 SQL 之间的映射关系。相当于 iBATIS 不会为程序员在运行期自动生成 SQL 执行。具体的 SQL 则需要程序员编写，然后通过配置映射文件，将 SQL 所需要使用的参数，以及正确运行处理后返回的结果字段映射到指定 POJO。

## AJAX

AJAX(Asynchronous JavaScript And XML, 异步 JavaScript 和 XML)它改进了传统的 WEB 应用，给浏览者一种更连续的体验。其最大优势在于异步交互，因为在软件服务器管理端为 B/S（浏览器/服务器）模式，在 B/S 模式下数据的安全性，数据的一致性，数据的实时性，系统的更新，网络应用的限制都比 C/S(客户机/服务器)模式下更具有优势。运用 AJAX 技术浏览者在浏览页面的同时可向服务器发送请求，甚至不用等待前一次请求得到完全响应，便可以再次发送请求，这种异

步的请求发送方式非常类似于传统的桌面应用。

## JAVASCRIPT

JavaScript 是一种基于对象和事件驱动并具有安全性能的解釋型脚本语言，在 Web 应用中得到了非常广泛的应用。JavaScript 是具有面向对象能力、解釋型的程序设计语言。在 Web 浏览器中应用最为广泛，在此环境中，通用用途的核心通过对象得到扩展，这些对象允许脚本和用户交互，控制 Web 浏览器，以及修改出现在浏览器窗口中的文档内容。JavaScript 的核心语言及其内建的数据类型都符合国际标准，它们具有跨平台实现的兼容性<sup>[22]</sup>。

JavaScript 是一种高级脚本语言，具有以下 3 个优点：

(1) JavaScript 采用在 HTML 文本中嵌入小程序段的方式，开发过程非常简单，并且提高了响应速度。

(2) JavaScript 直接面对用户或者客户的输入做出响应，而不需要经过 Web 服务器的程序的处理，大大减少了客户浏览器与服务器之间的通信量、提高了速度。

(3) JavaScript 是一种与平台无关的解釋性脚本语言，它完全依赖于浏览器，而与操作系统无关，只要计算机能运行浏览器，而且该浏览器支持 JavaScript，就可以执行 JavaScript 脚本程序。因此可应用在 JSP、ASP 和 PHP 等网站中

JavaScript 不但可以用于编写客户端的脚本程序，由 Web 浏览器解释执行，而且还可以编写在服务器端执行的脚本程序，在服务器端处理用户提交的信息并动态地向浏览器返回处理结果，通常在 JSP 中应用 JavaScript 编写客户端脚本程序。

## Android 系统优势

安卓从 2007 年至今，已有 7 个年头，在这七年里它拥有着迅猛的发展势态，并且一度霸占了世界上绝大部分电子设备的操作系统。当然，在安卓发展的同时，也必然会遇到了强硬的对手，以及现在正在崛起的手。2011 年以来 Android 实现了计算机诞生以来最快速度的增长，目前 Android 以每天 85 万新激活数的速度快速征服世界，成为世界范围内占有率最高的智能手机操作系统。根据 comScore 的统计，2012 年第一季度 Android 在北美的市场占有率又增长了 3.7% 达到 51%；而根据赛诺的调研报告，Android 系统因为开放性、挣脱束缚、丰富的硬件等方面的优势在中国的市场份额高达 74.7%。安卓之所以发展迅猛，主要是因为它具备以下优点：

(1) 开放性

开放性，即 Android 是一开源平台，修改源代码无需授权。从 Android 联盟建立以来，各 Android 终端厂家均可加入此联盟。由此越来越多的开发者因为用户对软件应用的需求积累了更多的经验，让 Android 系统发展的更加迅速，成熟。因而 Android 平台上的竞争越发激烈，终端用户在经济上就能得到更多的实惠。这对于开发者和消费者都是一个良性的循环。

### （2）挣脱束缚

过去手机应用均受到厂商和运营商制约，各项功能均为厂商或运营商所制定，例如网络接入模式等。自 iPhone 登录中国市场，用户便可以选择性的进行网络连接从而减少运营商的制约。随着 3G 逐步过渡 4G 网络的崛起，手机网络接入模式已经不在由运营商所限制。

### （3）硬件的多元化

因为 Android 系统的开放性，各厂商均推出具有自己特色功能的产品，但是这些产品确不会受到数据同步的影响，例如在 symbian、IOS、Android 各种系统上的联系人等信息均可进行转移。

### （4）开发商

Android 系统平台不会受到各种条款的制约和阴挠，让第三方开发商拥有更加灵活的开发环境，这样会诞生出更多别致的应用软件。

### （5）系统稳定实用

Android 系统是基于 Linux 的，而 linux 也经历了长久的发展，它的稳定性有目共睹，在经过一代代人的努力，它的系统代码年轻并且精简，手机运行比较快。系统升级后以前的软件都可用，目前支持的软件便多达 3 万种。

Android 是建立在 Linux 最具代表的开放性内核上的手机操作系统，Android 系统由 linux 操作系统、C 语言的函数库和虚拟机所构成的中间件、C++开发的用户界面及 JAVA 所组成的各类应用软件组成。因为此安卓系统的操作界面体现出更多的人性化操作。

## SQL SERVER 2005 技术介绍

SQL Server 2005 为实现 Internet 数据互联打造成为了一个具有灵活性和具扩展性的数据开发平台，拥有不断拓展应用空间；与此同时 SQL Server 2005 为增强企业中数据管理员的管理能力，减少系统运维成本降低系统运行风险 SQL Server 2005 拥有更高效的数据处理能力、更可靠安全的数据存储能力<sup>[23]</sup>。

SQL Server 在微软服务器商品中占有着举足轻重的地位，通过微软件服务器产品中 Engineering strategy 来达到更强的集成性和管理性，客户则可以更快捷简单

的进行系统的开发部署节省大量人力成本和部署时间。而且因为对系统的进行更加严格的安全控制，从而降低了应用程序的宕机时间，对各企业系统的要求层面上来更，登上了新的台阶。

SQL Server 2005 提供安全、可靠、高效的企业级数据管理平台。SQL Server 2005 在系统的扩展性、系统的安全性、数据的可靠性、用户的可管理性上有更大的进步，不仅对业务数据的稳定运行提供了安全保障还进一步提高了管理人员的管理效率，从而降低了操作难度和维护成本。硬件上，SQL Server 2005 支持 64 位运算和海量数据存储<sup>[24]</sup>。

SQL Server 2005 作为目前较为先进的商业智能平台，它不仅可以简单快捷的进行构建和部署各种商业存储解决方案，还能够为用户提供更深层次的业务数据分析统计和系统监控预测平台，成为推动企业的信息化管理和业务发展的新生力量。SQL Server 2005 拥有完善的智能商业套件，为海量的数据存储仓库、有力的数据分析、完整的数据报表、强大的数据挖掘能力提供了一系列程序设计、系统开发的管理工具。而随着市场的需求，它在构建具有实时性、扩展性的商业智能平台的领域也有着日新月异的发展趋势，主要分为两部分，第一部分是将全新的高效数据分析、强力的数据挖掘算法将更进一步的涉及到业务监控分析及决策支持；第二部分对支持各种异类数据和复杂数据业务通过企业级的数据提取、转换和加载工具进行整合<sup>[25]</sup>。

SQL Server 2005 所具备的扩展性和灵活性使得它为开发其他工具和各类新的开发特性提供了强而有力的后盾，它的运用不仅大幅度的提高了软件工具的开发效率，还拓展了应用的空间。<sup>[26]</sup>

## C/S 和 B/S 开发模式

### C/S 和 B/S 开发模式简介

C/S 模式即为 CLIENT/SERVER（客户端/服务器）模式，在这种结构下实现了软件、硬软件利用效率高、数据传输快、损耗小、服务器端压力较轻，充分保证了系统数据储存的透明性能。目前为止使用此架构的用户较多，因为在此架构下可以充分发挥资源共享。但在此架构下随着用户多样化服务器端必须提供不同版本的操作系统和软件以保证大量用户的同时使用，提高了对系统的维护难度和运营成本。

B/S 模式即为 BROWSER/SERVER（浏览器/服务器）模式。此模式随着 INTERNET 技术不断发展强大起来，是对 C/S 结构的一种革命。在这种模式下，



数据的处理，业务逻辑的流程都在服务器端完成。这样就减轻了客户端的负担，降低了系统运维的成本和工作量。

## C/S 和 B/S 开发模式对比

C/S 模式优点在于安全的存取控制和极强的交互性，因为 C/S 在点对点结构下运行，使数据安全能非常容易得到保证，而 B/S 则是多点对点的结构下运行，只有通过数据库加密等行为为数据安全提供保障。C/S 模式下网络通信流量较低，B/S 数据流量则远远高出 C/S。C/S 需要根据不同的操作系统开发各种不同版本软件，移植相对困难复杂，运维成本相对较高；B/S 则有统一用户视图，标准开放的平台，可以接受来自不同服务器和客户端的数据，集合动态的多媒体运用。

## 本章小结

将以上所描述的各类开发语言、数据库以及开发框架的优越性进行整合后，此款软件主要使用 MVC 设计模式，MVC 模式规定 B/S 结构应用分为三个部份 Model（模型 M）、View（视图 V）和 Controller（控制 C）。MVC 模式分离的数据访问和数据表现给系统提供了更好的解耦。通过使用 MVC 架构，更好提供应用的可扩展性。

在服务器端 MVC 架框让管理者有非常友好的用户介面。对快递包裹的集中分发，运送流程的监管，快递人员的业务量分析统计，实现可视化。同时终端选用 Android 系统实现 C/S 模式，Android 系统是手机、平板电脑等终端的操作系统，是当今最盛行的系统之一。自由的 Android 平台者拥有更多的自由，可以对地手机本身的功能进行修补安卓是多点触控的，用手指就可以操作了，非常方便，扩展性强。

### 第三章 系统需求分析

#### 业务需求

##### 用户角色分析

基于 Android 快递配送软件系统的搭建，其目的是帮助快递公司，建立一套可降低企业运营成本，实现无纸化办公，关联相关的审核、日志、快件管理、统计等为一体的产品综合平台。在此基础上，基于 Android 快递配送软件系统用户包括有：系统管理员，客服人员，快递派送人员三大类。

##### 1. 系统管理员

系统管理员的职责是负责系统的全局管理，具备所有系统功能。系统管理员能够对系统进行管理，包括新增用户、修改用户、删除用户，这里的用户包括快递管理员和快递员；其次是对权限的分配与管理，系统管理员无疑拥有最高权限，但用户就需要根据其职责进行权限的分级，此时系统管理员可以新建组，修改组，将用户添加到相应组中，并给不同组设置分配不同的权限；系统管理员还可以对手机终端进行管理，包括增加、修改和删除，此处的手机主要是快递配送人员配送时所使用的手机。具体的需求分析 UML 图如下图 3-1 所示：

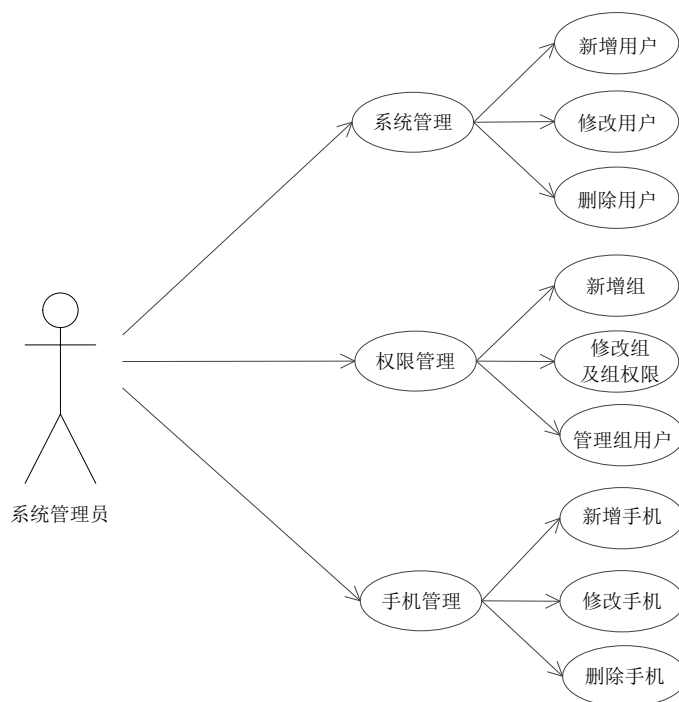


图 3-1 系统管理员需求分析 UML 图

## 2. 快件管理员

快件管理员主要的工作是对快递单进行管理与操作，快递单贯穿整个快递周期，从快递的开始一直到快递结束，乃至在未来很长一段时间内都将会存在于快递配送系统中。快递单从发货人将物品交由快递公司时产生，此时，快递管理员需要及时新增快递单；并录入快递单的某些具体信息，即是对快递单的编辑工作；接着就需要对快递单进行分配，根据寄往地点，分配相应的送货车、快递员进行派送；快件管理员还需要及时更新快递所在的位置，方便发件人和收件人查询物流状态；对于无效的快递单或者是早已完成配送并在数据库中保存了相当长一段时间的快递单可以进行删除操作。具体的需求分析 UML 图如下图 3-2 所示：

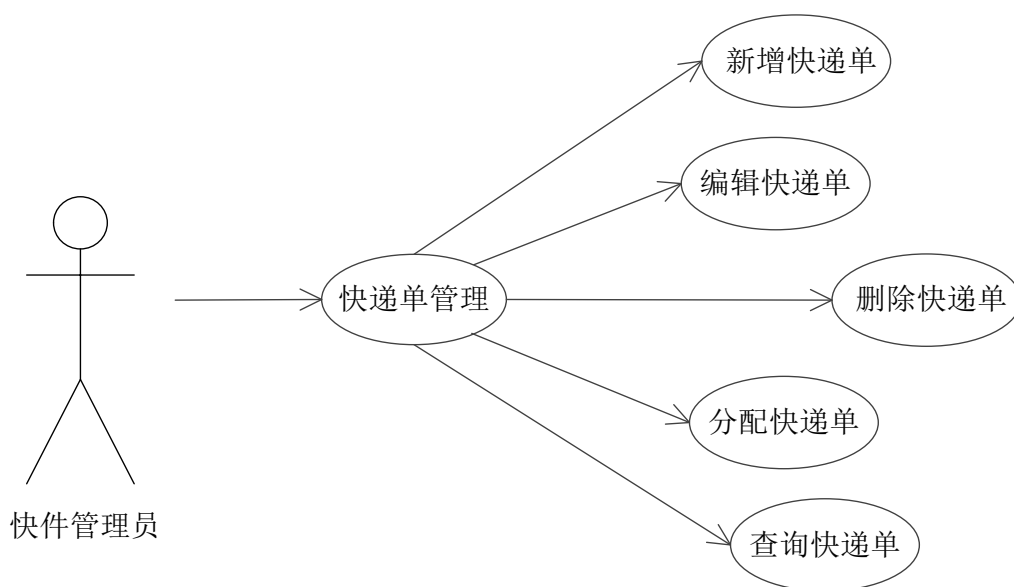


图 3-2 快件管理员需求分析 UML 图

## 3. 快递配送人员

快递配送人员是快件配送中唯一直接面向收件人的工作人员，他执行着快件配送最后也是最为关键的一步，在他的手机终端上，需要实现以下几个方面的功能：对于快件任务的划分，包括未派件任务、派送中任务、已完成任务，这样可以方便的告知快递配送人员需要做什么，其中未派件任务需要从本地手机终端读取服务器上的信息，包括快件的收件人、收件人联系方式、收件人地址等关键信息；其次还需要对地点进行定位及路线查询的操作，快递员可以查询到送货地点以及自身地点，然后选取到达方式进行最后的派送。具体的需求分析 UML 图如下图 3-3 所示：

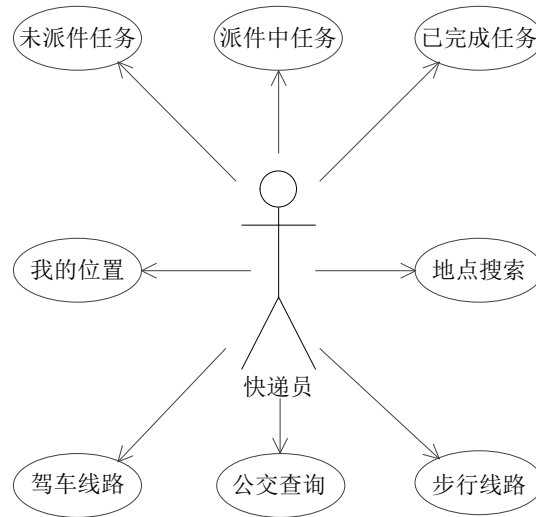


图 3-3 快递配送人员需求分析 UML 图

### 核心业务流程

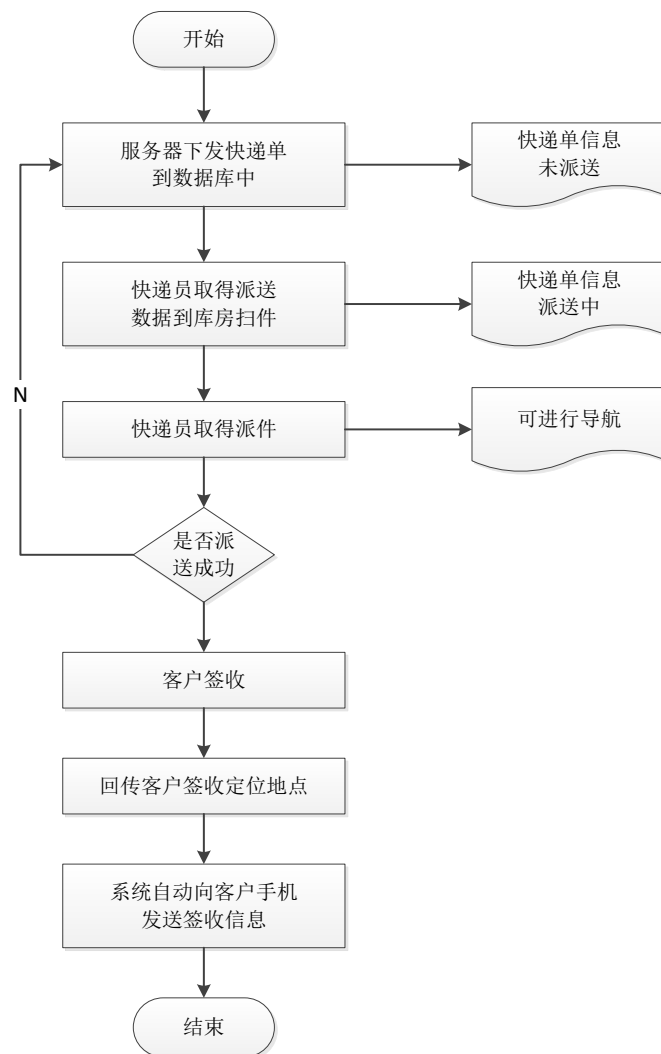


图 3-4 核心业务流程

该快递配送软件系统的核心业务流程主要是从服务器将快递单发送到数据库中，此时，快递员可以读取到未派送的快件信息，快递员取得派件数据后需要到库房扫件，将快件取走，此时，派件就从未派件状态更改为派送中状态，快递员在取得派件后，可以利用手机终端对派件地点进行定位并导航，到达地点后联系收件人，如果客户不在当地、收件地点出现错误或者其他导师派件无法顺利完成的因素发生时，将终止派件，此时派件状态将回归到未派件，并且入库，等待重新派送，如果派送地点正确、收件人正确，则客户需要签收，同时终端需要传送回客户签收的地点，服务器在收到客户签收成功的地点后向客户发送签收的短信，此时整个派件完成。

## 功能需求

### 系统管理功能

#### 1) 系统权限

系统用户操作服务器端：提供了对登录该系统用户进行添加、删除、停用、启用、修改操作。

系统权限操作服务器端：用户组的相关操作如添加、修改、删除及相应的系统权限赋予用户所在组，并将用户分发到各个系统组中，实现对系统的权限控制。

#### 2) 手机用户管理

手机用户管理（服务器上控制手机用户管理）：将用户手机相关信息进行录入、修改、启用、停用、删除、查询。主要包括手机终端的用户名、手机号、手机IMEI(MEID)号（\*#06#）和手机用户负责区域及所在组进行基本信息维护。

#### 3) 任务管理

##### （a）下发任务

快递人员用手机接入服务器，从服务器上获取分配任务到各自手机，通过手机上分发下来的序号条码，到分货场对货物进行条码扫描进行取货确认。

##### （b）检查派件完成任务

快递人员从服务器上获取派件是否完成派送。

#### 4) 统计管理

历史记录查询，送单统计：可分别对当日，当月进行速递人员送货量进行统计。

## 快递终端功能

- 未派送任务：由快递管理员分配的任务未到库房去扫描取件
- 派送中任务：在库房扫描取件后，进行派送的任务
- 已完成任务：其中包含客户签收定位，签收单影像资料保存
- 以下为辅助工具为快递派送人员线路导行：“我的位置”，“地点搜索”，“公交查询”，“驾车路线”，“步行搜索”。

## 非功能性需求

### 系统性能需求

此系统涉及到的业务处理量较大。一般情况下，系统业务处理量要求达到每分钟查询能力 400-500 次，每分钟更新能力要求在 300 次，系统同时在线人数至少 300 人左右。

对于快递业务处理执行的反应时间，快递公司要求：用户在系统登录，管理端快递数据处理、系统管理数据处理控制在 3 秒钟内完成响应，无操作响应时 10 分钟自动签退。快递员在送完件后上传的影像资料在 1 分钟内服务器能完全接收。该系统对服务器运行时的硬件资源分配要求其中 CPU 使用率在 50%-80%，内存使用率为 40%-60%，硬盘为 50%-80%（所有利用率不得超越上限）。

### 系统运行环境需求

（一）数据库服务器：需要运行 RDBMS 数据库管理系统，为管理端的 B/S 模式下提供数据库层的持久性服务。所有数据操作都在往到此服务器进行处理，通过 IBATIS 中的 XML 所写的 SQL 语句进行处理，对数据库进行持久化并且返回处理结果。

数据库应用使用 WINDOWS SERVER 2008 及以上版本（最好使用 DATACENTER）作为操作系统运行 SQL SERVER 2005 数据库。SQL Server 2005 是可以提供安全、可靠、高效的企业级数据管理平台。

SQL Server2005 提供安全、可靠、高效的企业级数据管理平台。SQL Server 2005 在系统的扩展性、系统的安全性、数据的可靠性、用户的可管理性上有更大的进步，不仅对业务数据的稳定运行提供了安全保障还进一步提高了管理人员的管理效率，从而降低了操作难度和维护成本。

（二）WEB 服务器：运行 WEB 应用软件服务采用 LINUX 系统，用于运行 TomcatServer 5 或以上版本。

(三) ANDROID 终端要求: android2.2 以上版本。

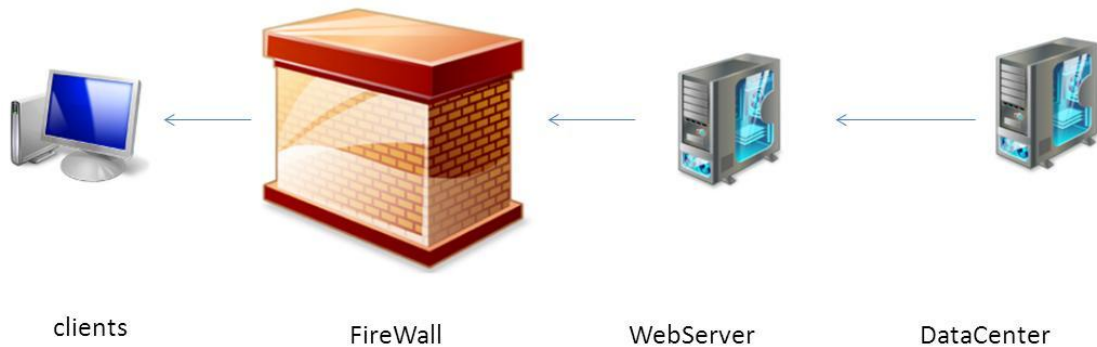


图 3-5 系统安全性需求

设备环境安全: 通风, 18-25℃。

## 本章小结

本章在业务需求方面根据业务人员所提到的功能绘制了 UML 图, 从中提炼出了核心业务流程; 在功能需求方面针对整个系统需要完成的任务进行了划分功能性的划分; 最后在非功能性需求方面详细分析了该软件系统所应具备的一些硬性指标。

## 第四章 系统设计

### 系统设计总则

快递配送软件系统旨在对系统中各类数据实现分权限共享，以便不同权限用户在任意系统接入点可以进行各类数据处理。

### 系统总体设计

快递系统结合了 B/S 和 C/S 的优点，管理层上充分利用 B/S 模式结构优点，派件员工终端上而充分利用 C/S 优点。在此混合模式中可以发挥两种模式各自优点来满足不同客户的需求。

### 系统结构



图 4-1 系统架构

### 系统服务器开发框架

系统服务器的开发框架主要采用 J2EE 轻量级架构，它是基于 Struts2+Spring+iBatis 框架实现的多层结构，其中 Struts2 用于表现层和控制层，Spring 位于业务逻辑层，主要处理业务逻辑，而 iBATIS 处于数据持久层<sup>[27]</sup>。J2EE 轻量级



架构图如下图 4-2 所示：

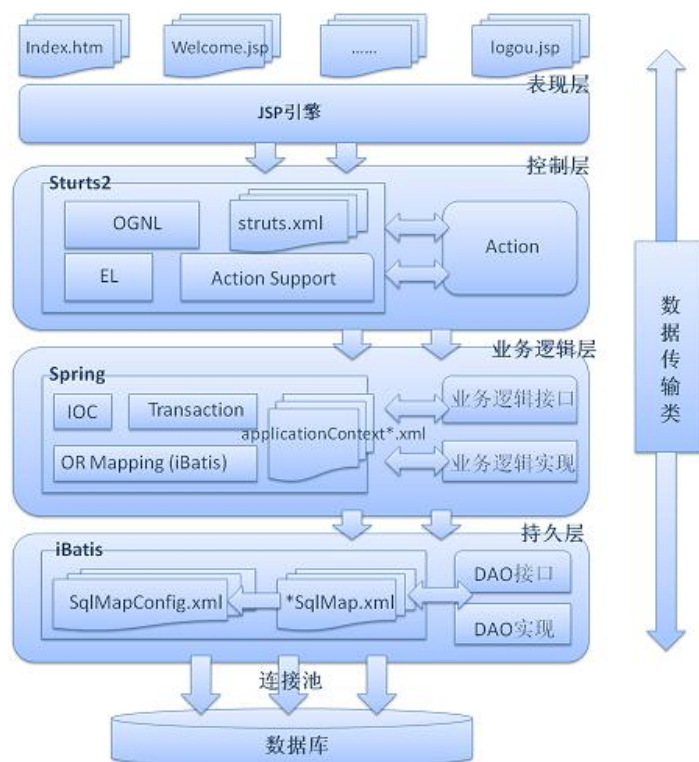


图 4-2 系统服务器开发框架

## 系统功能设计

从快件生命周期的角度来看，系统的功能可以划分为快件入库、快件派送、快件送达、快件记录四部分。其中快件入库后需要对快件进行管理，包括快件的配送管理、配送记录的录入、配送记录的统计等，直达最后快件送至收件人收入签收，其中每一个环节的实施都需要服务器的记录到日志中，以便期间或者后期查看。

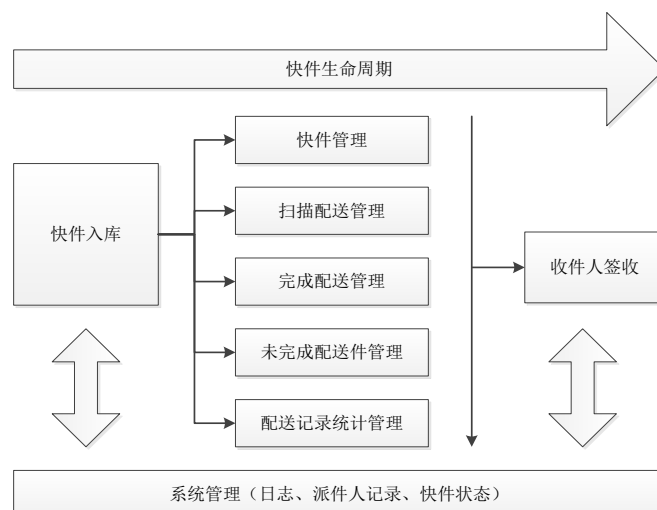


图 4-3 系统功能关系图

## 系统管理功能

系统管理功能主要实现对用户进行分组，并给予不同组不同的管理和使用权限。比如快件管理员组，他们的职责仅仅是面向快递单的，他们有权限对快递单号进行增删改查的操作，但是他们不能新增或者删除某个快件管理员帐号。而快递员他们更加无权去更改快件管理员所提供的快递单号，他们只能根据快递单上的信息对快件进行派送，保证快件从出库到送达这一过程的完整性。

## 快递终端管理

快递终端管理主要是实现对手机终端信息的更新，保证系统存储的是每个用户的最新手机号码及手机身份识别码(imei)。当有快递员的手机号码发送了变更，需要及时在此页面上进行手机号码的更改，以便快件管理员将快递单准确的投递给某位快递员。快递终端管理的结构图如下图 4-4 所示：

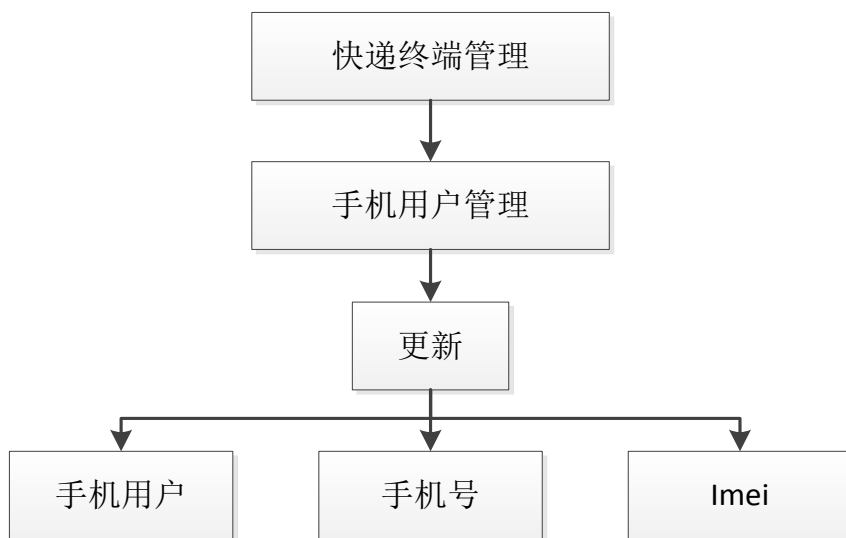


图 4-4 快递终端管理结构图

### 快递单任务管理

快递单任务管理主要是对快递单进行操作，包括添加、编辑、删除、查询、分发。

- 增加是指快件抵达快递公司后，需要入库并生成快递单号，此时快递单被添加进服务器；
- 编辑是对快递单的信息进行编辑，包括快件收件人、收件人联系方式、收件人地址、发件人相关信息等等；
- 删除是指如果该快递单因为某些原因无法送达而失效，需要删除该快递单；
- 查询是指客户可以通过快递单号来查看快件情况，历经哪些地市、分布到达时间，是否出现突发情况等；
- 分发是指快件管理员可以将快递单投递给最适合派件的快递员。

快递单任务管理结构图如下图 4-5 所示：

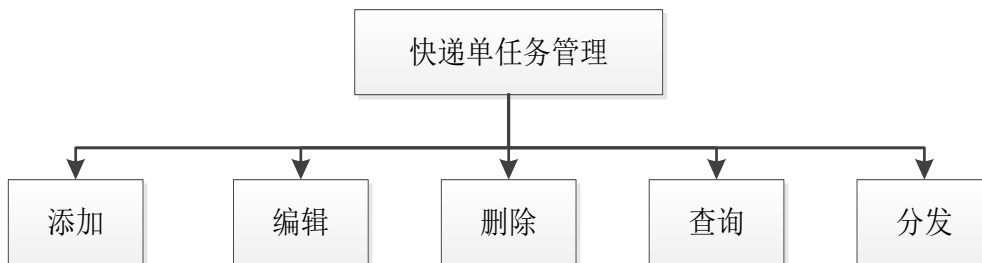


图 4-5 快递单任务管理结构图

### 统计分析管理

统计分析管理主要功能分为两部分：

- 实现历史记录的查询以及送单统计，它可以查询到任何有效的快递单，以及快递单的详细信息、配送状态、何时抵达何处、何人签收等等；
- 分别对当日，当月进行配送人员送货量的统计，并绘制出曲线图，直观的描述在一定时间内该快递公司所产生的送货量。

统计分析管理结构图如下图 4-6 所示：

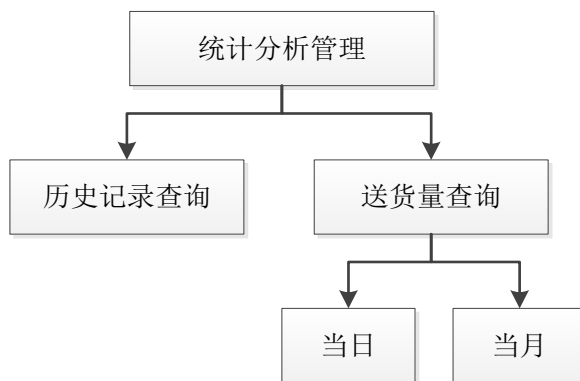


图 4-6 统计分析管理结构图

## 数据库设计

## 系统数据库关系图设计

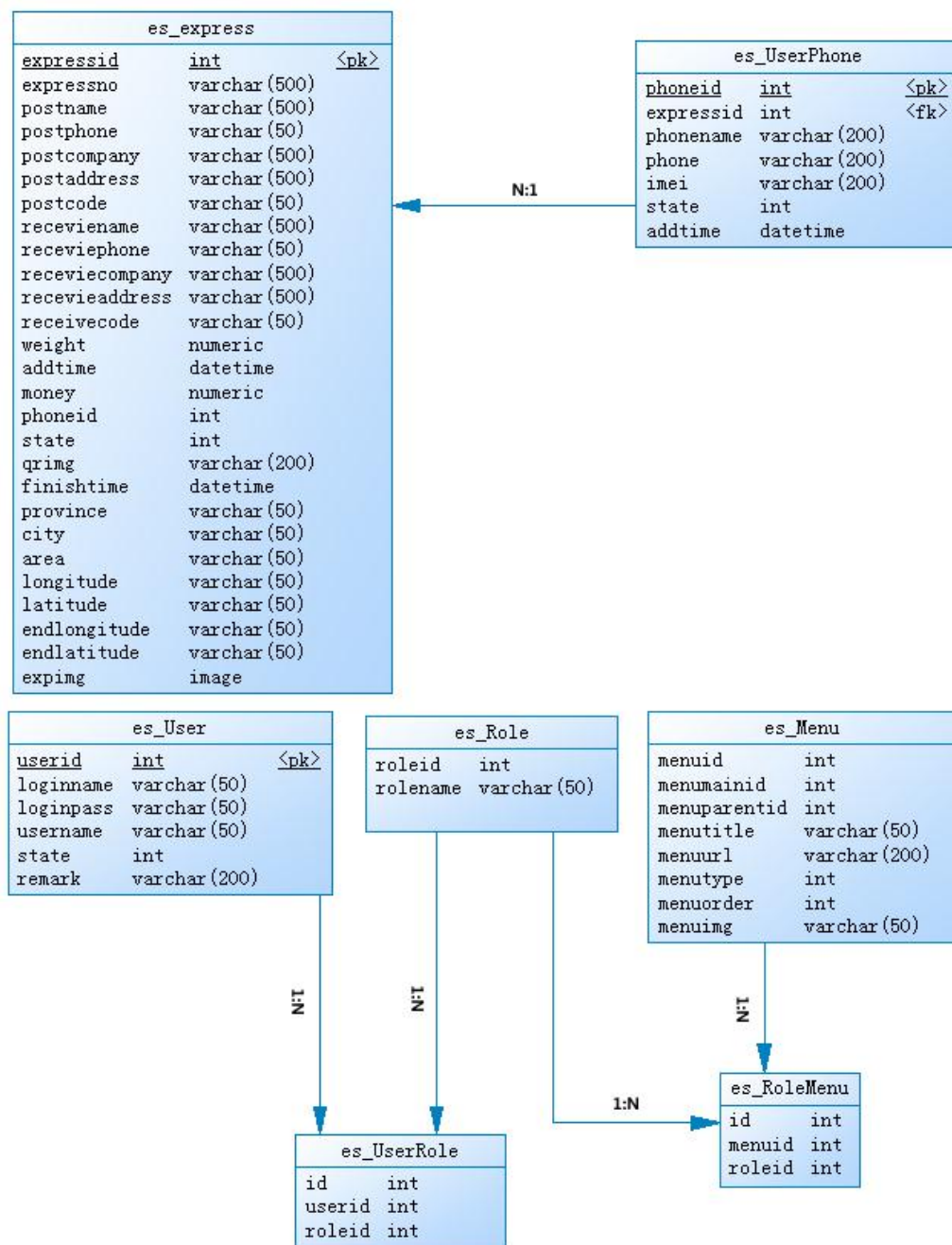


图 4-7 系统数据库关系图设计

## 数据库表设计

表 4-1 es\_User 登录用户表

字段名	字段类型	是否允许空值	字段说明
userid	int	否	主键自增
loginname	varchar(50)	否	用户登录名
loginpass	varchar(50)	否	登录密码
username	varchar(50)	否	用户姓名
state	int	否	用户状态
remark	varchar(200)	是	备注

表 4-2 es\_UserPhone 快递人员手机信息表

字段名	字段类型	是否允许空值	字段说明
phoneid	int	否	主键自增
phonename	varchar(200)	否	电话机主姓名
phone	varchar(200)	否	电话号码
imei	varchar(200)	否	电话 IMEI 号
state	int	否	电话状态
addtime	datetime	否	增加电话时间

表 4-3 es\_Role 权限表

字段名	字段类型	是否允许空值	字段说明
roleid	int	否	主键自增
rolename	varchar(50)	否	权限名称

表 4-4 es\_Menu 系统菜单表

字段名	字段类型	是否允许空值	字段说明
menuid	int	否	主键自增
menumainid	int	否	菜单主 ID
menuparentid	int	否	上级菜单 ID
menutitle	varchar(50)	否	菜单名
menuurl	varchar(200)	否	菜单 URL
menutype	int	否	菜单类型

表 4-5 es\_RoleMenu 权限菜单关联表

字段名	字段类型	是否允许空值	字段说明
id	int	否	主键自增
menuid	int	否	关联菜单 ID
roleid	int	否	关联权限 ID

表 4-6 es\_UserRole 用户权限关联表

字段名	字段类型	是否允许空值	字段说明
id	int	否	主键自增
userid	int	否	关联用户 ID
roleid	int	否	关联权限 ID

表 4-7 es\_express 快递信息存储表

字段名	字段类型	是否允许空值	字段说明
expressid	int	否	主键自增
expressno	varchar(500)	否	快递单号
postname	varchar(500)	否	寄件人名
postphone	varchar(50)	否	寄件人电话
postcompany	varchar(500)	否	寄件人公司
postaddress	varchar(500)	否	寄件人地址
postcode	varchar(50)	否	寄件人邮编
receviename	varchar(500)	否	收件人姓名
receviephone	varchar(50)	否	收件人电话
receviecompany	varchar(500)	否	收件人公司
recevieaddress	varchar(500)	否	收件人地址
receivecode	varchar(50)	否	收件人邮编
weight	numeric	否	快件重量
addtime	datetime	否	开始派件时间
money	numeric	否	费用
phoneid	int	否	派件人电话 ID
state	int	否	派件状态
qrimg	varchar(200)	否	二维码图

## 系统安全性设计

系统采用了用户密码加密，在系统用户信息表中设有密码字段，用户可以设置自己的密码来登录服务器获取相关快件信息。

## 本章小结

本章从整体到局部对快递配送软件系统的系统结构、框架、功能、数据库设计等方面做了阐述。具体实现的功能是系统管理功能、快递终端管理、快递单任务管理、统计分析管理，数据库表的详细字段等。



## 第五章 系统测试和实现

### 5.1 系统测试

整个系统的测试包括了系统服务器端性能测试、手机终端性能测试、系统功能测试和业务流程的测试。

#### 5.1.1 服务器性能测试

使用 Junit 测试工具结合多线程, 编写 TestDao 类, TestAction 类, 由 TestAction 去调用 TestDao 类下面的 save()、update()、insert()、del()四个方法。每个方法并发 10 次新操作  $50 \times 4 \times 10 = 2000$  操作总量。取开始时间和结束时间以秒为单位计算平均时间。

表 5-1 服务器性能测试结果

服务器端	JAVA 版本	容器	带宽	并发数量	操作平均响应时间
	1.6.15	tomcat-5.5.26	10M/4M	50 用户	<1s/<4s

#### 5.1.2 终端性能测试

使用 JAVA 多线程处理机制并发总量 3000 条业务信息, 以线程组方式分别为各类手机并发 600 条, 在线程组开时运行时设定变量, 结束时取到时间变量。通过这种方式每次以总量 3000 条信息, 组量 600 条信息的标准进行更新, 运行 500 次以秒为单位取平均值。

表 5-2 终端性能测试结果

	品牌	型号	Android 版本	并发数据总量	处理分量	接收平均响应时间	压缩图片平均时间	发送平均响应时间
手机端	三星	SCH-I579	2.3	3000 条	600 条	3s	8s	6s
	HTC	Supersonic	3.0		600 条	3s	5s	3s
	SONY	LMT25i	4.0		600 条	2s	5s	4s
	三星	GT-N7000	4.1		600 条	2s	5s	3s
	华为	荣耀 3C	4.2		600 条	2s	4s	2s

### 5.1.3 系统功能测试

使用 Junit 测试即所谓白盒测试，对每个 DAO 类和 ACTION 类的所有方法进行单元测试。

```
//Spring 框加中加载 Junit4
@RunWith(SpringJUnit4ClassRunner.class)
//初始化 Spring Bean
@ContextConfiguration(locations="/applicationContext.xml")
//测试 RoleDao 类
public class TestTabSysRoleDao
{
//装配 Dao 层
@Autowired
private ITab_Sys_RoleDao tabSysRoleDao;
//创建 Dao 实例
Tab_Sys_Role tabSysRole =new Tab_Sys_Role();
//测试 Dao 类 Del()方法
@Test
public void testDelObject()
{
System.out.println("testDeleteObject");
tabSysRole.setRolecode("c");
System.out.println(tabSysRoleDao.deleteObject(tabSysRole));
}
```

通过以上测试代码模版对每个系统功能进行相关测试，下面对系统进行分模块测试，系统测试结果如下表 5-3 所示：

表 5-3 系统功能测试结果

系统名称	运行模块	操作	结果
端 WEB 平 台 系 统 管 理 配 送 软 件 快 递	任务管理	添加快递单	正常
		查询快递单	正常
		编辑快递单	正常

		删除快递单	正常
		分发快递单	正常
	终端管理	添加手机终端	正常
		查询终端	正常
		编辑终端	正常
		删除终端	正常
	系统管理	个人信息修改	正常
		新增用户	正常
		修改用户	正常
		删除用户	正常
		新增用户组	正常
		分配组权限	正常
		分配用户到组	正常
	统计分析	查询发件量	正常
快递配送软件系统 手机端	未派送任务	查看任务	正常
		扫描取件	正常
	派送中任务	派送签收	正常
		拍照上传	正常
		发送签收短信	正常
	已派送任务	查看已完成任务	正常
	我的位置	是否定位	正常
	地点搜索	查询地址	正常
	公交查询	查询线路	正常
	驾车路线	导行	正常
	步行路线	导行	正常

#### 5.1.4 业务流程测试

本节将对系统软件进行流程上的实例测试，实测实例及结果如下所示：

寄件人姓名：赵盛华

寄件人手机：15308005569

寄件人单位：中国电信成都分公司

寄件人地址：四川省成都市贝森路 26 号

寄件人邮编：610000

收件人姓名：杨怡

收件人手机：13881905744

收件人单位：中国银行高新支行

收件人地址：四川省成都市高新区天府大道 1366 号软件园 E 区三期 E8 栋

快递重量：1kg

快递费用：6RMB

送货人：雷鹏

第一步：添加快递单号，完成对快递单的新增和编辑；



The screenshot shows the 'Express System' web interface. The top navigation bar includes '统计分析' (Statistics Analysis), '任务管理' (Task Management), '终端管理' (Terminal Management), '系统管理' (System Management), and '退出系统' (Exit System). The left sidebar has '任务管理' (Task Management) and '快递单管理' (Express Order Management). The main content area is titled '任务管理' (Task Management) and contains a form for adding a new express order. The form fields are as follows:

Field	Value
寄件人姓名	赵盛华
寄件人手机	15308005569
寄件人单位	中国电信成都分公司
寄件人地址	四川省成都市贝森路26号
寄件人邮编	610000
收件人姓名	雷鹏
收件人手机	13881905744
收件人单位	中国银行高新支行
省份/地区	四川省
城市	成都市
地区	武侯区
收件人详细地址	天府大道1366号软件园E区三期E8
经度	104.074638
纬度	30.541992
收件人邮编	610000
快递重量	1.0
快递费用	6.0
送货人	雷鹏

A QR code is displayed on the right side of the form. The bottom of the page shows the copyright information: 'Copyright ©2013 LeiPeng Project.All Rights Reserved.' and the user information: '工程硕士毕业设计:雷鹏(201191020403)'.

图 5-4 增加送货单

第二步：对快递单进行确认；

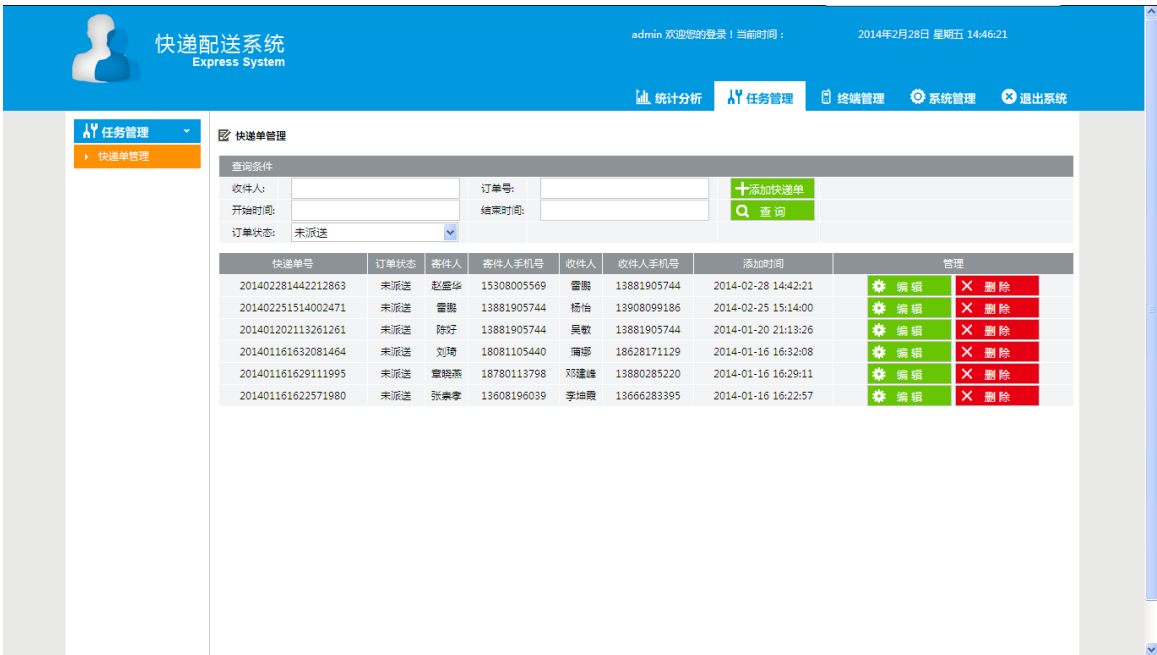


图 5-5 确认送货单状态

第三步：快递员打开手机终端的快递配送系统软件，并登录；



图 5-6 手机终端系统界面

第四步：快递员进入库房，扫描快件二维码，激活任务；



图 5-7 扫描二维码

第五步：快递员送货成功，由收件人签收，由快递员拍照确认并将照片及 GPS 地点发送回服务器，服务器将信息以短信的形式发送给收件人；



图 5-8 签收拍照并短信确认

关键性代码测试通过无报错。该系统经过测试，在运行过程中已经基本达到预期值。

## 5.2 软件实现

### 5.2.1 添加快递单

添加快递单是指快件抵达快递公司后，需要入库并生成快递单号，此时快递单调用 ExpressDao 类 save()方法存放快递记录被添加进服务器<sup>[28]</sup>。实现关键代码以及网页截图如下所示：

```
public String save(){
    form.setReturnurl("express!findByList.shtml");           //设置返回 url
    form.setAddtime(Tools.getCurrentTime());                 //设置时间标识
    form.setExpressno(Tools.getNowRandomThree());
```

```

form.setState(0);          //快件状态修改
form.setQrimg(Tools.getTwoCode(Tools.getTwoString(form)));
Integer result = expressDao.save(form);    //数据存入
if(result != 0){
    form.setReturncode(1);
    form.setReturninfo("添加快递单成功");
}else{
    form.setReturncode(0);
    form.setReturninfo("添加快递单失败");
}
return "finish";
}

```



图 5-9 添加快递单

### 5.2.2 查询快递单

查询快递单是指客户可以通过快递单号来查看快件情况，历经哪些地市、分布到达时间，是否出现突发情况等，调用 ExpressDao 类 findByList()方法查询可以根据收件人姓名、快递单号、快件开始派送时间、结束时间、订单状态等关键词进行搜索，ExpressDao 类 findByCount()方法可取得符合条件的快递单总数，实现关键代码以及网页截图如下所示：

```

public String findByList(){
    form.setPager_openset(10);        //设分页量
    form.setPager_count(expressDao.findByCount(form));//分页总量
    form.setList(expressDao.findByList(form));
    return "findByList";
}

```



图 5-10 查询快递单

## 5.2.3 编辑快递单

编辑快递单是调用 ExpressDao 类 update()方法对快递单的信息进行编辑,包括快件收件人、收件人联系方式、收件人地址、发件人相关信息等等,并且自动生成二维码,实现关键代码以及网页截图如下所示:

```

public String update(){
    form.setReturnurl ("express ! findByList . shtml ");
    Integer result = expressDao.update(form);
    if(result != 0){
        form.setReturninfo("更新快递单成功");
        form.setReturncode(1);
    }else{
        form.setReturninfo("更新快递单失败");
    }
}

```



```

        form.setReturncode(0);
    }
    return "finish";
}

```



图 5-11 编辑快递单

### 5.2.4 删除快递单

删除快递单是指如果该快递单因为某些原因无法送达而失效，需要删除该快递单，调用 ExpressDao 类 delExpress()方法来实现关键代码如下所示：

```

public String delete(){           //删除方法
    expressDao.delExpress(form);
    findByList();
    return "findByList";
}

```

### 5.2.5 分发快递单

分发快递单是指快件管理员可以将快递单投递给最适合派件的快递员，UserPhoneDao 类 findByList()方法列出快递人员供管理员选择由谁来分发该单快件<sup>[29]</sup>，实现关键代码以及网页截图如下所示：

```

public String userPhoneChoice(){
    form.setPager_openset(10);
}

```

```

form.setPager_count(userPhoneDao.findByCount(form));
form.setList(userPhoneDao.findByList(form));
return "userPhoneChoice";
}

```

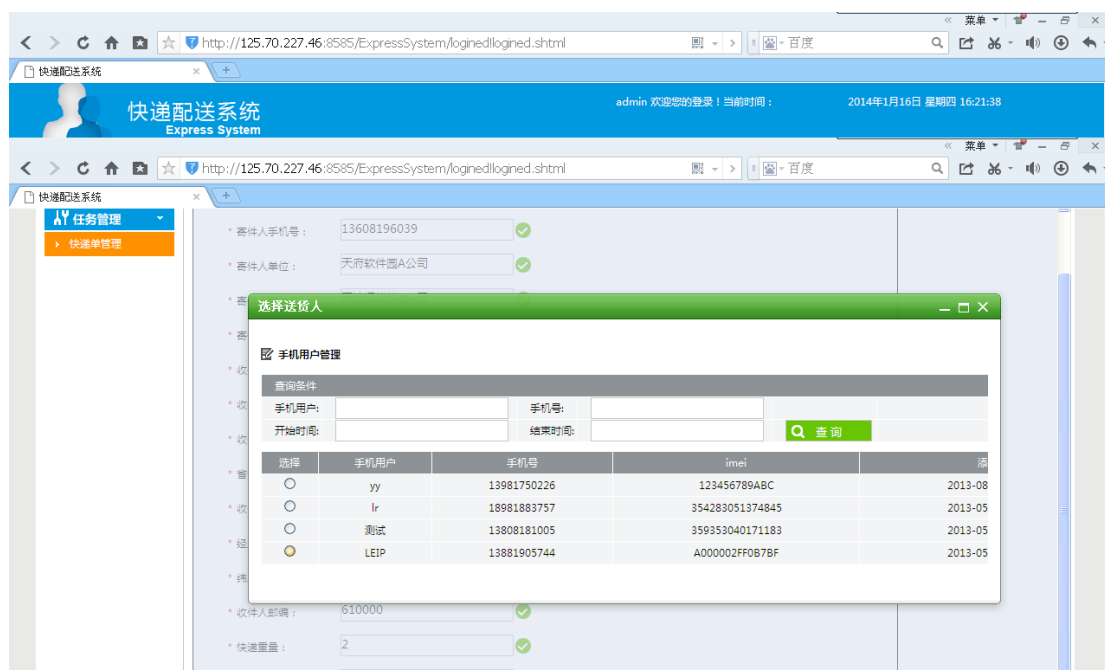


图 5-12 分发快递单

### 5.2.6 添加手机终端

添加手机终端主要是指通过调用 UserPhoneDao 类 save() 方法将快递员的姓名、手机号以及手机识别码 imei 录入进系统，方便快件管理员对派件进行分发与通知。实现关键代码以及网页截图如下所示：

```

public String save(){
    form.setReturnurl("userPhone!findByList.shtml");
    List<UserPhone> list = userPhoneDao.findByExist(form);
    //通过返回list大小判断用户是否存在
    if(list != null && list.size() > 0){
        form.setReturninfo("添加手机用户组失败,该手机用户已存在");
        return "save";
    }
    //默认状态是正常状态
    form.setState(0);
}

```

```

form.setAddtime(Tools.getCurrentTime());
Integer result = userPhoneDao.save(form);
if(result != 0){
    form.setReturncode(1);
    form.setReturninfo("添加手机用户成功");
}else{
    form.setReturncode(0);
    form.setReturninfo("添加手机用户失败");
}
return "finish";
}

```



图 5-13 添加手机终端

### 5.2.7 分配组权限

分配组权限主要是调整各个组对系统和对快件的操作权限,通过HashMap构建一个字符串和整型的键值对,调用RoleDao类saveRoleMenu()类来存储各个组所对应的权限,并进行分配。实现关键代码以及网页截图如下所示:

```

public String updateRoleMenu(){
    System.out.println(form.getMenustr());
    if(form.getMenustr()==null||"".equals(form.getMenustr())){
        form.setReturninfo("权限ID为空, 请重试");
    }
}

```

```
        findByRoleMenu();
        return "updateRoleMenu";
    }
    roleDao.deleteRoleMenu(form);
    String[] menustr = form.getMenustr().split(",");
    List<String> menuidlist = new ArrayList<String>();
    for(int i = 0 ;i<menustr.length;i++){
        String[] menuid =menustr[i].split("_");
        //HashMap构建一个字符串和整型的键值对，调用RoleDao类
        //saveRoleMenu()类来存储各个组所对应的权限
        Map<String,Integer> map = new HashMap<String,Integer>();
        map.put("roleid", form.getRoleid());
        map.put("menuid", Integer.valueOf(menuid[0]));
        roleDao.saveRoleMenu(map);
        if(menuidlist!=null&&menuidlist.size()>0){
            int flag = 1;
            for(int j=0;j<menuidlist.size();j++){
                if(menuidlist.get(j).equals(menuid[1])){
                    flag =2;
                }
            }
            if(flag==1){
                menuidlist.add(menuid[1]);
            }
        }else{
            menuidlist.add(menuid[1]);
        }
    }

    for(String menuidstr :menuidlist){
        Map map = new HashMap();
        map.put("roleid", form.getRoleid());
        map.put("menuid",menuidstr);
    }
}
```

```

        roleDao.saveRoleMenu(map);
    }
    form.setReturnurl("role!findByList.shtml");
    form.setReturncode(1);
    form.setReturninfo("更新组权限成功");
    return "finish";
}

```



图 5-14 分配组权限

### 5.2.8 分配用户到组

分配用户到组主要是系统管理员将用户分配到不同的组中，调用RoleDao类saveRoleUser()方法帮助用户快速的设置权限，而不用一条条的将权限附加在用户上，实现关键代码以及网页截图如下所示：

```

public String updateRoleUser(){
//删除用户权限
    if(form.getUseridtype().equals("deluser")){
        if(form.getUseriddel()!=null&&!form.getUseriddel().equals("")){
            String[] useriddel = form.getUseriddel().split(",");
            for(int i = 0 ;i<useriddel.length;i++){
                if(useriddel[i].equals("")){
                    continue;
                }
            }
        }
    }
}

```

```
        }
        Map<String,Integer> useridmap = new
HashMap<String,Integer>();
        useridmap.put("userid", Integer.valueOf(useriddel[i]));
        useridmap.put("roleid", form.getRoleid());
        roleDao.deleteRoleUser(useridmap);
    }
}
} else if(form.getUseridtype().equals("adduser")){
//增加用户权限
    if(form.getUseridadd()!=null&&!form.getUseridadd().equals("")){
        String[] useridadd = form.getUseridadd().split(", ");
        for(int i = 0 ;i<useridadd.length;i++){
            if(useridadd[i].equals("")){
                continue;
            }
            Map<String,Integer> useridmap = new
HashMap<String,Integer>();
            useridmap.put("userid", Integer.valueOf(useridadd[i]));
            useridmap.put("roleid", form.getRoleid());
            roleDao.saveRoleUser(useridmap);
        }
    }
}
findByRoleUser();
return "updateRoleUser";
}
```

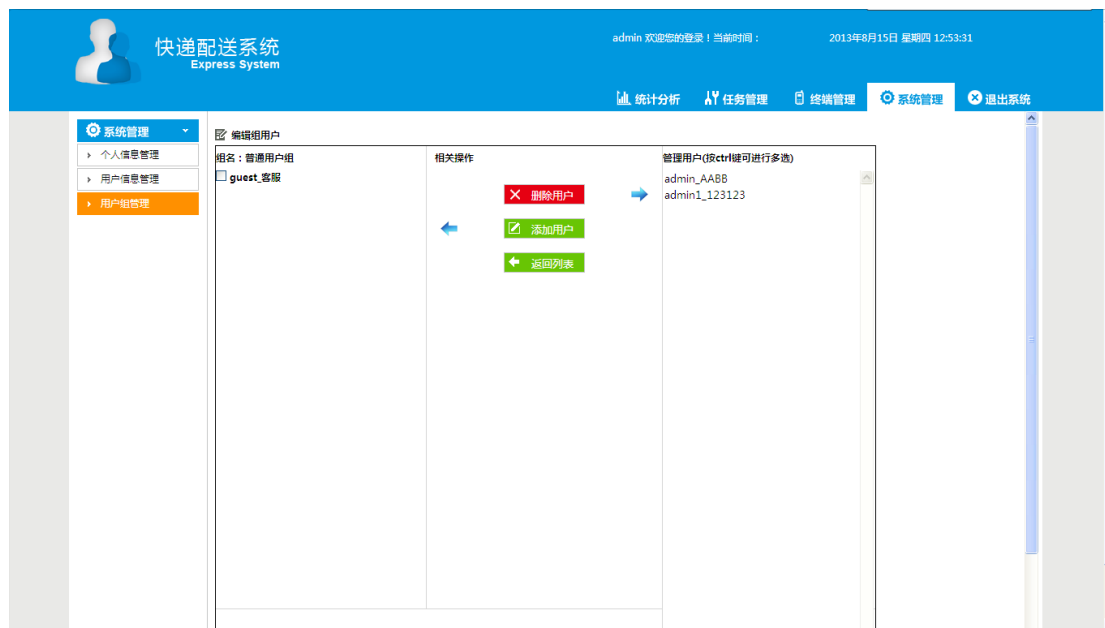


图 5-15 分配用户到组

### 5.2.9 生成派件二维码

生成派件二维码主要是为了方便用户和快递员利用手机马上能查询到该快件，调用 `getExpressImageFile()` 方法返回图片流，其实现关键代码以及网页截图如下所示：

```
public void getExpressImageFile() {
    try {
        HttpServletRequest request=this.getRequest();
        HttpServletResponse response=this.getResponse();
        Receive recevie = recevieDao.findExpImgById(expressId);
        if(recevie != null){
            response.reset();
            response.setContentType("application/octet-stream");
            response.addHeader("Content-Length",
                ""+recevie.getExpimg().length); //设置返回的文件类型
            InputStream fis = new
                ByteArrayInputStream(recevie.getExpimg());
            OutputStream toClient = new
                BufferedOutputStream(response.getOutputStream());
            byte[] buffer = new byte[1024*1024];
            while(fis.read(buffer)>0){
```

```

        toClient.write(recevie.getExpimg()); // 输出数据
    }
    fis.close();
    toClient.flush();
    toClient.close();
}
} catch (Exception ex) {
    ex.printStackTrace();
}
}

```

### 5.2.10 手机端实现



图 5-16 手机端登录界面

### 5.2.11 主菜单界面

主菜单界面主要是提供给快递员使用，它整合了快件状态、路线查询、定位、导航等多个功能，调用 `init()` 方法初始介面，实现关键代码以及手机截图如下所示：

```

public void init(){
    gridview = (GridView) findViewById(R.id.gridview);
    list = new ArrayList<GridInfo>();
}

```



```
list.add(new GridInfo("未派送任务"));
list.add(new GridInfo("派送中任务"));
list.add(new GridInfo("已完成任务"));
list.add(new GridInfo("我的位置"));
list.add(new GridInfo("地点搜索"));
list.add(new GridInfo("公交查询"));
list.add(new GridInfo("驾车路线"));
list.add(new GridInfo("步行搜索"));
list.add(new GridInfo("关于我们"));
adapter = new GridAdapter(this);
adapter.setList(list);
gridview.setAdapter(adapter);
gridview.setOnItemClickListener(new gridViewListener());
}
```



图 5-17 主菜单界面

### 5.2.12 扫描取件

扫描取件是客户签收快件以后，快递员可以通过手机终端扫描快件的二维码或者是快递单的条形码，将签收信息发送给服务器。调用 `initCamera()` 初始化摄像头，调用 `handleDecode()` 方法返回二维代码<sup>[30]</sup>，其实现的关键代码和手机运行图如

下所示:

//初始化手机摄像头

```
private void initCamera(SurfaceHolder surfaceHolder) {  
    try {  
        CameraManager.get().openDriver(surfaceHolder);  
    } catch (IOException ioe) {  
        return;  
    } catch (RuntimeException e) {  
        return;  
    }  
    if (handler == null) {  
        handler = new CaptureActivityHandler(this, decodeFormats,  
            characterSet);  
    }  
}  
  
public void handleDecode(Result obj, Bitmap barcode) {  
    inactivityTimer.onActivity();  
    playBeepSoundAndVibrate();  
    String str = obj.getText();  
    Express express = (Express) getIntent().getSerializableExtra("express");  
    txtResult.setText(obj.getBarcodeFormat().toString() + ":" +  
        str + "===" + express.getExpressno());  
    Intent intentresult = new Intent();  
    Bundle bundle = new Bundle();  
    bundle.putString("format", obj.getBarcodeFormat().toString());  
    bundle.putString("contents", str);  
    bundle.putSerializable("express", express);  
    intentresult.putExtras(bundle);  
    intentresult.setClass(ScanActivity.this, TaskScanResultActivity.class);  
    startActivity(intentresult);  
    this.finish();  
}
```



图 5-18 扫描取件

### 5.2.13 发送签收短信

发送签收短信发生在快递员将签收信息发送给服务器后，服务器接收到签收信息后会发送短信到客户手机上，以便客户确认是自己签收。其实现的关键代码和手机运行图如下所示：

//创建监听对象

```
Button.OnClickListener start_listener = new Button.OnClickListener(){
    public void onClick(View v){
        if("").equals(tv_phone.getText().toString())||tv_phone.getText().toString().length() != 11){
            Toast.makeText(PhoneNumActivity.this,
                "请输入用户手机号码!", Toast.LENGTH_LONG).show();
            return;
        }else{
            startTask();
        }
    }
};

public void startTask(){
    progress();
    handler = new Handler(){
        //打开监听器
        public void handleMessage(Message msg) {
            if(dialog != null){
```

```
        dialog.dismiss();}

        //Toast.makeText(PhoneNumActivity.this,
msg.getData().getString("returnData").toString(),Toast.LENGTH_LONG).show();
        btn_phone.setVisibility(View.GONE);
        btn_return.setVisibility(View.VISIBLE);
        Intent intent = new Intent();
        Bundle bundle = new Bundle();
                bundle.putSerializable("express",express);
                intent.putExtras(bundle);
        intent.setClass(PhoneNumActivity.this, CameraActivity.class);
        startActivity(intent);
        super.handleMessage(msg);
    }
};
new Thread(startTaskRunnable).start();
}
private Runnable startTaskRunnable = new Runnable() {
//打开短信接口发送信息
public void run() {
TaskStarttaskPacket packet = new TaskStarttaskPacket(
PhoneNumActivity.this,taskid,2,tv_phone.getText().toString());
packet.wrap();
HttpUtil httpUtil = new HttpUtil(packet.getRequestObj());
packet.unwrap(httpUtil.postRequest());
Message msg = handler.obtainMessage();
Bundle bundle = new Bundle();
bundle.putCharSequence("returnData", packet.getReturnData());
msg.setData(bundle);
handler.sendMessage(msg);
}
};
```



图 5-19 发送签收短信

#### 5.2.14 派送签收及拍照上传

```

public void onClick(View v) {
    // 从相册获取图片
    if (v == button2) {
        Intent intent = new Intent(Intent.ACTION_PICK, null);
        // 获取所有图片信息
        intent.setDataAndType(
            MediaStore.Images.Media.EXTERNAL_CONTENT_URI,
            IMAGE_UNSPECIFIED);
        startActivityForResult(intent, PHOTO_ZOOM);
    } else if (v == button1) {
        // 从拍照获取图片
        Intent intent = new Intent
            ("android.media.action.IMAGE_CAPTURE");
        long time = Calendar.getInstance().getTimeInMillis();
        intent.putExtra(MediaStore.EXTRA_OUTPUT,
            Uri.fromFile(new File(Environment.getExternalStorageDirectory(), "temp.jpg")));
        startActivityForResult(intent, PHOTO_GRAPH);
    }
}

public static String uploadSubmit(String url, Map<String, String> param,
    File file) throws Exception {

```

```

HttpPost post = new HttpPost(url);
MultipartEntity entity = new MultipartEntity();
if (param != null && !param.isEmpty()) {
    for (Map.Entry<String, String> entry : param.entrySet()) {
        if (entry.getValue() != null
            && entry.getValue().trim().length() > 0) {
            entity.addPart(entry.getKey(),
                new StringBody(entry.getValue()));
        }
    }
}
// 添加文件参数
if (file != null && file.exists()) {
    entity.addPart("expfile", new FileBody(file));
}
post.setEntity(entity);
HttpResponse response = httpClient.execute(post);
int stateCode = response.getStatusLine().getStatusCode();
StringBuffer sb = new StringBuffer();
if (stateCode == HttpStatus.SC_OK) {
    HttpEntity result = response.getEntity();
    if (result != null) {
        InputStream is = result.getContent();
        BufferedReader br = new BufferedReader(new
InputStreamReader(is));
        String tempLine;
        while ((tempLine = br.readLine()) != null) {
            sb.append(tempLine);
            System.out.println(tempLine+"=====");
        }
    }
}
post.abort();
return sb.toString();

```

}

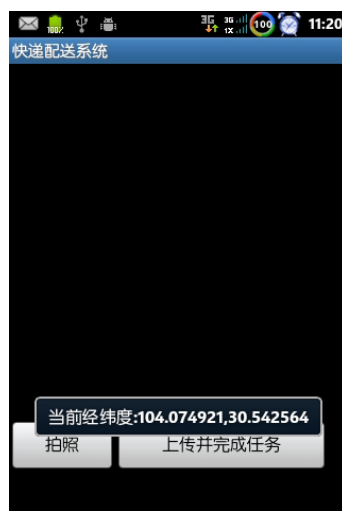


图 5-20 派送签收及拍照上传

### 5.2.15 查看任务



图 5-21 查看任务

## 5.3 本章小结

本章主要对整个快递配送系统的实现给出了各个模块的软件及网站运行图和关键代码，详细阐述了每个模块具体的实现方法，并对软件进行了分类测试，达到了前文所提出的各项需求。

## 第六章 总结与展望

本文主要实现的是基于 Android 系统下的快递配送软件,针对当前网络交易日趋火爆,结合 Android 系统的大众化、成本低及优点,设计并实现了 Android 系统下的快递配送软件。主要利用 JAVA 编程语言结合 spring+struts 框架,前台使用 JSP 和 javascript 等技术进行编写程序和数据处理。作者完成了此系统的需求分析到最后的实现并测试,从中汲取到了很多经验与教训,并认真思考了该系统今后的研究方向。

1、针对用户密集度,可以在快件配送中加入定点配送<sup>[31]</sup>和移动配送。在定点配送时,圈定一定范围内的用户,群发短信通知其领取;在移动配送时能按照用户所在地进行合理的路线规划,帮助快递人员方便及时的抵达送货地点。

2、催取功能,可以用于发送取件短信或电话告知取件后若干小时内仍无人前来取件,快件配送系统将自动发送催取短信,催促收件人取件。

3、除了配送端的开发,还需要进行用户端的开发,成立数据中心,分散下单,统一管理,这样更能体现出事半功倍的工作效果。并且增加自助服务系统例如:快件单查询、客户通过 APP 下单、相关业务查询功能。



## 致 谢

在经过几年的工作实践后，能够重新踏入电子科技大学的校门，较为系统的学习各方面知识，对我来说是难得的机会。在这几年的学习期间，我得到了来自老师和师兄的帮助，使我受益良多。

在本论文与软件的设计与实现的过程中，我的导师曹宗杰副教授给予了我很大的帮助在整个系统从提出到设计并完成测试，都离不开他的悉心指导和严格要求曹宗杰副教授学识渊博，为人热情，他精益求精的工作作风，朴实无华的生活态度，给我树立了很好的榜样，不仅使我明白了自己在学习和工作中存在的问题，也明白了待人接物的一些道理。在此，我向导师曹宗杰表示崇高的敬意和衷心的感谢！

除此之外，我还要感谢我的师兄弟们，感谢他们平日里对我的照顾和关心，并对我的毕设提出了很多宝贵的意见和建议。

同时，感谢在背后一直默默支持着我的父母、妻子，感谢他们的体谅、理解与付出，支持着我完成了学业，我将一生陪伴着你们，谢谢！

最后，向所有帮助过我的老师和同学表达我最诚挚的谢意！

## 参考文献

- [1] 晏敬东, 石银萍, 李谦. 我国快递业发展的现状、问题与对策[J]. 中国科技信息.2008
- [2] 赵煜. 网购“狂奔”, 快递为何“踉跄”? [N]. 天津日报. 2014-01-04
- [3] 蒋永霞. 服务时代 物流业将迎来二次变革[N]. 中国商报. 2014-01-10
- [4] Chia-Chi Lee, Tyrone T. Lin, Pei-Chu Cheng. The determinants of the transportation outsourcing strategy for the express delivery company[J]. Service Business. 2013. 7(2): 207-225
- [5] Xingjian Zhou. Research on Logistics Value Chain Analysis and Competitiveness Construction for Express Enterprises[J]. American Journal of Industrial and Business Management. 2013. 03 (02): 131-135
- [6] 邱官升. 快递业服务质量分析与改善方法研究[D]. 西安: 长安大学. 2008
- [7] 倪明, 孙潜. 快递业运营模式研究[J]. 北京交通大学学报(社会科学版). 2013. 12(1)
- [8] Tuckwell R. Express Deliveries[J]. Logistics & Transport Focus, 2008, (10):14
- [9] 赵强, 赵永峰, 杨发智, 等. 基于 Android 平台的物流信息查询软件的设计与实现[J]. 电脑知识与技术. 2012, 8(22):5298-5302
- [10] 吴蔚. 安卓平台快递派送辅助软件开发[D]. 北京 北京邮电大学. 2013 . 15-20
- [11] 昊斯特曼(Horstmann Gay S.), Gary Cornell. JAVA 核心技术(卷 1):基础知识
- [12] 昊斯特曼(Horstmann Gay S.), Gary Cornell. JAVA 核心技术(卷 2):高级特征
- [13] 埃史尔, 陈昊鹏 (译者). Java 编程思想(第 4 版)
- [14] 常建功,等. Java Web 典型模块与项目实战大全
- [15] 李刚. Struts 2.x 权威指南(第三版)
- [16] 陆舟. Struts2 技术内幕: 深入解析 Struts2 架构设计与实现原理
- [17] 陈云芳. 精通 Struts2:基于 MVC 的 Java Web 应用开发实战
- [18] 罗时飞, 飞思科技产品研发中心 (合著者). 精通 Spring:深入 JavaEE 开发核心技术
- [19] 陈雄华, 林开雄. Spring 3.X 企业应用开发实战
- [20] 哈罗普, 马可赛克. Spring 专业开发指南
- [21] Clive Begin , Brandon Goodin, Larry Meadors. iBATIS in Action
- [22] 弗拉纳根. Javascript 权威指南
- [23] 曼蒂(Joy Mundy), 桑斯维特, 金伯尔. 数据仓库工具箱:面向 SQL Server2005 和 Microsoft 商业智能工具集
- [24] Edward Whalen, Marilina Garcia, 黄湘情, 谢琳. SQL Server 2005 管理员手册. 人民邮电

出版社

- [25] 徐人凤,曾建华. SQLServer2005 数据库及应用
- [26] 朱如龙. SQLserver2005 数据库应用系统开发技术
- [27] 李刚. 基于 J2EE 的 AJAX 宝典
- [28] 哈希米(Sayed Hashimi), 克曼特内尼(Satya Komatineni), 麦克莱恩(Dave MacLean) . 精通 Android 2
- [29] 林城. Android 2.3 应用开发实战
- [30] 张元亮. Android 开发应用实战详解
- [31] Yingfeng Ji, Hualong Yang, Yan Zhang, et al.Location Optimization Model of Regional Express Distribution Center[J]. Procedia - Social and Behavioral Sciences. 2013, 96