

西安邮电大学

毕业设计（论文）

题目：____基于 ssm 物流系统的设计与实现____

学院：____通信与信息工程学院（人工智能学院）____

专业：____通信工程____

班级：____通工 1611____

学生姓名：____李康____

学号：____03161371____

导师姓名：____莎柯雪____职称：____讲师____

起止时间：2020 年 3 月 5 日 至 2020 年 6 月 30 日

毕业设计（论文）承诺书

本人所提交的毕业论文《基于 SSM 物流系统的设计与实现》是本人在指导教师指导下独立研究、写作的成果，论文中所引用他人的文献、数据、图件、资料均已明确标注；对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式注明并表示感谢。

本人深知本承诺书的法律责任，违规后果由本人承担。

签名：李康

日期：2020 年 3 月 5 日

西安邮电大学本科毕业设计(论文)选题审批表

申报人	莎柯雪	职 称	讲师	学 院	通信与信息工程学院 (人工智能学院)			
题目名称	基于 ssm 物流系统的设计与实现							
题目来源	科研				教学		其它	是
题目类型	硬件设计		软件设计	是	论文		艺术作品	
题目性质	应用研究		是		理论研究			
题目简述	<p>本设计是基于 ssm(SpringBoot + Spring MVC + MyBatis)平台，设计出一套物流系统，服务于移动、派送员、移动门店三方的物流支撑系统；该系统包含需求上报、物质下发、订单跟踪、政策发布四大模块；</p> <p>本次项目分为几个环节训练：需求概要设计文档、详细需求说明书、数据库设计文档、编码、测试、出测试报告、上线试运行等，整个项目结束后对需熟知软件开发流程，强化编程能力，提高逻辑思维能力。</p>							
对学生知识与能力要求	要有较强的软件编程能力，对 ssm 框架和 web 开发有一定的了解，深刻理解 java 基础 html 等相关技术，熟练使用 web 容器，有较强的学习和领悟能力，对调用第三方插件时，有较强的解决问题的能力。							
具体任务以及预期目标	1, 通过仿真项目实践，熟练掌握编程技巧； 2, 熟悉软件开发生命周期各个环节，能对每个环节做简单的风险把控； 3, 熟练掌握基于微信开发相关知识； 4, 熟练掌握调用第三方插件，以及处理在调用插件过程中遇到的问题； 5, 熟悉简单算法，并能将一些简单算法运用到开发中； 6, 归纳总结知识框架，做到融会贯通； 7, 最后完善并修改毕业论文，完稿；							
时间进度	2020.3.6 - 2020.3.16 编写软件需求说明书，概要设计文档，详细设计文档，数据字典； 2020.3.17 - 2020.4.15 熟悉基于微信开发相关技术，了解扫码相关插件，搭建软件框架,进行项目编码； 2020.4.15 - 2020.6.3 熟悉算法，按照详细设计文档开发软件并做功能及性能测试； 2020.6.4-2020.6.11 总结毕设成果，撰写毕业论文，撰写答辩 PPT 及相关资料，准备答辩。							
系（教研室）主任 签字	2020 年 1 月 10 日			主管院长 签字		2020 年 1 月 15 日		

西安邮电大学本科毕业设计（论文）开题报告

学生姓名	李康	学号	03161371	专业班级	通工 1611
指导教师	莎柯雪	题目	基于 ssm 物流系统的设计与实现		
选题目的（为什么选该课题） 当今，由于社会物流行业的快速增长，社会过渡关注 To C 的物流业务，而导致类似”海底捞“、”中国移动“，这种集团门店之间 To B 的物流业务方向，发展较少，导致一大批私有的物流管理公司（小作坊式）产生，由于他们没有专业的技术、管理人才、办公系统，导致企业之间类似于门店这种配送物资方式运营成本、人力资源、开销费用较高，因此市场调研之后，决定开发一款类似于中国移动门店之间配送的，由”管理员“、”配送员“、”门店“，三方构成小型物流系统，进而促进企业的成本，提高成本、人力资源的利用率；					
前期基础（已学课程、掌握的工具，资料积累、软硬件条件等） 一、需要掌握以下知识： Mysql 技术内幕（InnoDB 存储引擎）、Java 技术基础、Spring、SpringMVC、MyBatis 框架、可能用到 Redis、ES 等中间件； 二、使用以下工具： IDEA、PostMan、DataGrip、SecureCRT； 三、调研资料 1. 第三方物流整合对物流服务质量、伙伴关系及企业运营绩效的影响研究； 2. 基于物联网的生鲜农产品冷链物流体系构建:框架、机理与路径； 四、后期评估 1. 项目开发完成后，希望放到供一些小型的企业试用，评估项目价值；					
要研究和解决的问题（做什么） 一、面向后台管理员的后台管理系统，此系统完成以下功能； 1. 快递员管理：对于快递员账号的增加(单条和批量)、删除(批量)、修改、列表展示等操作； 2. 物资管理：对于派送物资信息的维护,其中包含新增、删除(批量)、修改、列表展示等操作； 3. 门店管理：对于门店信息的维护,其中包含新增(含批量)、删除(批量)、修改、列表展示等操作； 4. 物流订单管理：该功能包含对订单信息的批量上传、检索(搜索订单号/发货人电话/收货人电话/快递员电话的搜索、订单状态的搜索)、编辑、删除(批量)、派送(批量)、详情、列表展示等操作； 5. 快递公司管理：对于快递公司信息的维护,其中包含新增、删除(批量)、修改、列表展示等操作； 6. 需求上报审核：负责处理门店需求上报信息的审核(含二级审核),当审核通过后系统会自动将通过审核后的信息发送到配送员； 7. 订单配送统计：按渠道统计订单配送情况，可按日期筛选。 二、面向前台的类似于手机端的系统 1. 快递员：负责按时配送物资到门店； 2. 门店：上报需求物资，记录历史物资； 3. 管理员：审核物资需求，确定配送员，配送物资；					

工作思路和方案（怎么做）

一、面向后台管理员的后台管理系统，此系统完成以下功能；

1. 快递员管理：对于快递员账号的增加(单条和批量)、删除(批量)、修改、列表展示等操作；

2. 物资管理：对于派送物资信息的维护,其中包含新增、删除(批量)、修改、列表展示等操作；

3. 门店管理：对于门店信息的维护,其中包含新增(含批量)、删除(批量)、修改、列表展示等操作；

4. 物流订单管理：该功能包含对订单信息的批量上传、检索(搜索订单号/发货人电话/收货人电话/快递员电话的搜索、订单状态的搜索)、编辑、删除(批量)、派送(批量)、详情、列表展示等操作；

5. 快递公司管理：对于快递公司信息的维护,其中包含新增、删除(批量)、修改、列表展示等操作；

6. 需求上报审核：负责处理门店需求上报信息的审核(含二级审核),当审核通过后系统会自动将通过审核后的信息发送到配送员；

7. 订单配送统计：按渠道统计订单配送情况，可按日期筛选。

二、面向前台的类似于手机端的系统

1. 快递员：负责按时配送物资到门店；

2. 门店：上报需求物资，记录历史物资；

3. 管理员：审核物资需求，确定配送员，配送物资；

指导教师意见

该生毕设前期准备工作做得充分，对毕设内容理解深入，收集资料较为详实，有较为具体的可行工作方案和计划，且计划和方案较为实际、可行。同意开题。

签字

2020年3月16日

西安邮电大学毕业设计(论文)成绩评定表

学生姓名	李康	性别	男	学号	03161371	专业班级	通工 1611
课题名称	基于 ssm 物流系统的设计与实现						
指导教师意见	支撑课程目标/赋分	CT1/20	CT2/20	CT3/20	CT5/20	CT6/20	合计
	得分						
	评分(百分制): _____ 指导教师(签字): _____ 年 月 日						
评阅(验收)意见	支撑课程目标/赋分	CT2/40	CT3/40	CT4/10	CT5/10	合计	
	得分						
	评分(百分制): _____ 评阅教师(签字): _____ 年 月 日						
答辩小组意见	支撑课程目标/赋分	CT4/40	CT5/25	CT6/35	合计		
	得分						
	评分(百分制): _____ 答辩小组组长(签字): _____ 年 月 日						
评分比例	指导教师评分 (20%) 评阅(验收)评分 (40%) 答辩小组评分 (40%)						
学生总评成绩	百分制成绩			等级制成绩			
答辩委员会意见	毕业论文(设计)最终成绩(等级): _____ 学院答辩委员会主任(签字、学院盖章): _____ 2020 年 6 月 30 日						

目 录

第一章 绪论.....	1
1.1 国内外研究现状.....	1
1.2 课题研究的意义与价值.....	1
1.3 课题的论文结构.....	1
第二章 物流系统开发相关技术.....	3
2.1 开发工具及相关技术.....	3
2.1.1 JAVA 程序开发语言.....	3
2.1.2 WEB 程序开发技术—JSP.....	4
2.1.3 Tomcat 应用服务器.....	4
2.1.4 MySQL 数据库.....	5
2.2 MVC 设计模式.....	6
2.2.1 MVC 设计模式.....	6
2.2.2 Spring MVC Request 流程.....	6
2.3 本章小结.....	7
第三章 物流系统的分析.....	8
3.1 系统可行性分析.....	8
1. 可行性分析的前提.....	8
2. 技术可行性分析.....	8
3. 经济利益可行性分析.....	8
4. 运营操作可行性分析.....	8
5. 结论.....	8
3.2 系统业务功能需求分析.....	8

3.2.1 集团中心.....	9
3.2.2 APP 端门店中心.....	10
3.2.3 APP 端快递员中心.....	11
3.2.4 集团中心、门店中心、快递员中心三者之间关系.....	12
3.3 系统架构性能需求分析.....	12
系统安全性.....	12
系统的合理性.....	12
系统操作的易用性.....	13
3.4 本章小结.....	13
第四章 物流系统的设计.....	14
4.1 系统架构设计.....	14
4.1.2 业务模块设计.....	14
4.1.3 核心模块详细设计书.....	16
4.2 系统数据库设计.....	19
1. 管理员、快递员、门店店主账户信息表 (sys_account)	19
2. 组织机构信息表 (sys_group)	20
3. 后台管理系统资源列表信息表 (sys_resources)	20
4. APP 端菜单详情表 (sys_wx_menu)	20
5. 门店信息表 (mkl_store_demand)	21
6. 公司旗下品牌信息表 (mkl_vendor)	21
7. 需求上报审核表 (mkl_store_demand)	22
8. 物流订单详情表 (mkl_mail_info)	22
9. 快递公司详情表 (mkl_express_store)	24

10. 快递员实时坐标信息表 (mkl_courier_store)	24
4.3 本章小结.....	24
第五章 物流系统的实现.....	25
5.1 搭建开发环境.....	25
5.2 系统的主要框架及其配置.....	25
5.3 系统核心模块的实现.....	28
5.3.1 后台管理系统、APP 端用户登录的实现.....	28
5.3.2 后台系统主界面的实现.....	30
5.3.3 需求上报的实现.....	31
5.3.4 需求上报审核.....	33
5.3.5 订单配送流程图.....	35
5.4 本章小结.....	41
结 论.....	42
致 谢.....	43
参考文献.....	44

摘 要

针对物流行业内的企业结构、人员管理、配送流程越来越复杂的情况下，对于高效、便捷的物流管理系统被更多的企业所重视，因此开发一款企业所急需的信息化物流管理系统，是非常重要的，它可以有效的降低企业经营成本、提高配送效率。

面向企业的信息化物流管理系统的主要目标是向集团旗下在不同区域中的门店提供一种物资配送服务，高效的分配集团物资，提高集团的营业额，并且可以降低由于管理人员、配送员、门店店主的错误操作或者疏忽导致的各种人为因素的错误，为科学、准确和高效的完成集团门店间物资配送服务工作提供的强而有力的工具与管理方式；

本文主要介绍了以 Java 语言为基础、SSM 为系统框架的现代化物流管理系统。在现代化物流管理系统的开发中，主要通过分析国内外集团间物资配送方式以及管理人员提出的需求进行调研，总结出一套高效、科学、可行的物流系统需求。该系统主要包括网页的后端管理系统、APP 的门店、快递员操作软件，功能设计上主要是根据集团中心、门店中心、快递员中心之间的关系从而构建该系统。该系统的开发严格按照软件工程发指导原则来设计和开发。采用 Tomcat 8、JDK 1.8、Spring、MyBatis 搭建物流管理系统，MySQL 作为数据库存储数据、PostMan 作为测试工具、ProcessOn 绘制 UML、流程图。

关键词：物流；SSM；MySQL（数据库）；

ABSTRACT

In view of the increasingly complex enterprise structure, personnel management and distribution process in the logistics industry, more and more enterprises attach importance to the efficient and convenient logistics management system. Therefore, it is very important to develop an information-based logistics management system urgently needed by enterprises, which can effectively reduce the operating cost of enterprises and improve the distribution efficiency.

The main goal of the enterprise oriented information logistics management system is to provide a kind of material distribution service to the stores of the group in different regions, efficiently distribute the group's materials, improve the turnover of the group, and reduce the errors of various human factors caused by the wrong operation or negligence of managers, distributors and store owners, which is scientific, accurate and efficient. A strong and powerful tool and management mode to complete the material distribution service between the group stores;

This paper mainly introduces the modern logistics management system based on Java and SSM. In the development process of modern logistics management system, mainly through the analysis of the material distribution mode between domestic and foreign groups and the demands put forward by the management personnel for research, summed up a set of efficient, scientific and feasible logistics system requirements. The system mainly includes the back-end management system of web page, the store of app, the operation software of express delivery. The functional design is mainly the relationship among group center, store center and express delivery center to build the system. The system is designed and developed in strict accordance with the guiding principle of software engineering development. Tomcat 8, JDK 1.8, spring and mybatis are used to build the logistics management system, MySQL is used as the database to store data, postman is used as the testing tool, processon is used to draw UML and flow chart.

Key words: logistics; SSM; MySQL(database);

引 言

在科技、技术的不断创新下，国内互联网公司的进步成绩是众所周知的，这迫使很多大型公司进行转型，拥抱互联网变化，享受互联网所带来的盈利模式。而像酒店、物流等人员结构、业务方向交叉复杂的公司，使用互联网技术、信息化管理公司的各种人员，业务是非常有必要的一项任务。

本课题所设计的物流系统，可以让面向企业的大型物流公司的物资管理、人员管理、配送管理、订单管理、门店管理、审核管理等方面，会有极大的提升，缩短上报物资、审核物资、配送物资整个流程的所需时间，降低时间成本，并且也会使得企业的灵活度、竞争力、创新性得到正反馈，同时也会统计的配送物资列表，快递员配送订单次数，时间等数据，可以利用大数据进行分析，找到企业的不足之处，后期改进，为企业管理层提供有效的支持和进步改革方向。

此系统的研发将会打通整个物流企业的供应链，建立起一套行之有效的管理系统，完美的帮助企业进行信息管理，使他们从以前的纸质化管理信息、存储信息成功的迈向互联网，迎合时代的发展趋势，享受社会进步所给我们带来的方便之处。因此该系统商用价值是不言而喻的，它可以使尚未转型、拥抱互联网的企业快速的进入互联网+传统企业的运营模式，所以非常有必要作为本次的研究课题，这既是自己对物流行业管理运营模式的一次研究探索，也是大型物流公司一次弯道超车的机会。

第一章 绪论

1.1 国内外研究现状

2020 年国内的互联网已经逐渐发展到成熟阶段，各行各业纷纷意识到“互联网+”是下一个风口，而且随着国内 5G 的布局，基础建设逐步完善，网络化、信息化的管理可以极大的降低企业的运营成本，提升相关人员的工作的效率，使得公司在信息化方面具强大的实力，引领行业的潮流，特别是像物流行业、酒店行业这种人员结构复杂的企业，尤为重要。

互联网下，物流行业的快速增长，社会过渡关注 To C（面向用户）的物流业务^[5]，诞生了一大批百世、顺丰、中通等大型快递公司，而导致像类似“海底捞”，“苏宁易购”这种集团门店之间 To B（面向企业）的物流业务方向，发展极为缓慢，导致一大批私有的物流管理公司（小作坊式）产生，由于他们没有相对专业的技术、办公系统、管理人才，导致企业之间类似于门店这种配送物资方式的运营成本、人力资源、开销费用较高。现代化的企业对供应链的体系有了更加全面、清晰的定义，像集团间的物流配送，我们应该打通整个供应链系统，建立起一套行之有效的管理系统，为集团人员、门店、配送员制定操作软件，帮助他们从传统的纸质化管理、转到线上利用信息网络进行管理、最终形成一个具有完整配送流程的系统模型。

1.2 课题研究的意义与价值

物流管理信息系统是一个架构复杂，功能交叉的系统，其所涉及的内容包括：物资需求上报，需求审核，物资下方，配送物资，配送统计等许多方面的功能，并且需要集团中心，门店，快递员中心进行紧密的联系、衔接，才可以让该系统发挥其作用。如果像苏宁易购这种分店过多的集团采用此系统的话，可以让他们的物资管理，人员管理，配送等方面，会有极大的提升，缩短整个流程的所需时间，降低时间成本，这反之也会使得企业的灵活度、竞争力、创新得到正反馈，同时统计的配送物资列表，快递员配送订单次数，时间等数据，可以利用大数据进行分析，找到企业的不足之处，后期改进，为企业管理层提供有效的支持和改进方向。

1.3 课题的论文结构

第一章 绪论：本章主要对课题的背景、国内外现状进而说明，从而提出开发物流管理信息系统的价值和意义。

第二章 物流管理系统开发相关技术：本章主要对物流管理系统开发过程中所使用到的主要技术进行分析和探讨，其内容主要包括，Java 相关技术的介绍其中包括 Java 特性、JSP 前端开发技术、Tomcat 的工作机制等相关工具；对 MySQL 数据库的介绍，以及特性；最后对 MVC 设计模式的介绍，以及 MVC 的 request 流程做了一定的阐述。

第三章 物流管理系统的分析：本章主要对物流管理系统进行全面的可行性分

析、业务功能需求分析、架构性能分析三部分。基于物流管理系统分析的结论，接下来将对该系统进行详细的设计，规划与实现。

第四章 物流管理系统的设计：本章的主要工作是对物流管理信息系统的整体系统架构、业务功能模块的划分、核心模块的设计、数据库中核心表的表结构进行详细的设计和说明，尤其是在数据库中由于设计的表多而复杂，因此，在设计初我们完全按照数据库的范式，进行合理的拆解和划分，进而设计出高效、稳定的数据库系统。

第五章 物流管理系统的实现：本章主要对物流管理信息系统实现过程中，开发环境的搭建、项目框架的配置以及核心模块的实现做了简单的介绍和说明，确保系统各个子模块可以正常、稳定、高效的运行。

第六章 总结：本章主要对整个课题系统分析、设计、研发过程进行评价和分析，同时说明该系统研发后的不足之处，后期改进。

第二章 物流系统开发相关技术

2.1 开发工具及相关技术

2.1.1 JAVA 程序开发语言

Java 是面向对象的计算机语言，它不仅拥有其他开发语言的特性，而且抛弃了其他语言的缺点，是一个非常又发现有效的开发平台，为开发者们提供了许多有用的工具，例如：编译器，解释器和打包工具等。Java 主要划分为三种版本，分别是 J2ME，J2SE，J2EE，其中 J2EE 是 Java2 Enterprise Edition 的简称，主要目标是为企业的计算提供一种应用服务器运行和开发的操作平台。J2EE 本身就是一个相对开放的标准，任何的大型互联网公司都可以根据自己的用户需求制定符合自己的 J2EE 产品。

Java 语言的主要特性

1. 简单性；Java 语言在设计之初就以为开发者们提供一们简单、便捷的开发语言为目标，在后期不断的更新，迭代中，如今的 Java 已经非常适合大型项目的开发，其编写简单、语法易于理解不需要复杂、繁琐的细节，就可以编写自己的业务功能。

2. 面向对象；Java 不是面向过程的语言，它是一种面向对象的语言。对开发者们来说，意味着他们需要将复杂的流程化模式对象化，抽象出类的概念，用类作为基本单位编写代码。它的方法即是系统的功能，属性是系统内的特征，并且通过派生类可以继承父类的特性，拥有父类的功能，这就是继承。

3. 分布性；如今的项目架构越来越复杂，以前单机版的项目已不能符合现在的开发需求，为了负载均衡，现在都流行分布式的开发框架。而 Java 语言刚好是一门分布式语言。它不仅支持模块间的分布式部署、还可以在网络连接中支持流模式，导致用户可以产生分布式的 client 和 server。

4. 编译性和解释性；Java 编写的程序需要用编译器编译，产生字节码文件，它是一种 JVM 虚拟机专门执行的编码，经过 JVM 虚拟机的翻译，生成的二进制文件，可以在各种系统中执行，因此导致用户可以产生分布式的 clie 它实现了跨平台性，这是其他语言所不具备的特性，为 Java 编写的系统提供了极大的便利，有效的拓展了 Java 的市场占有率。

5. 稳健性；Java 的系统框架设计的非常的稳定、其配置简单、容易使其为开发者们所喜爱，它不支持指针的功能，直接操作内存，通过 JVM 虚拟机操作内存，极大的保证内存的安全性，不会导致内存溢出，内存未及时回收等情况的发生。它是一种强类型语言，在编译时会检查属性，方法，返回值等是否符合类型的要求，如果类型过大的话会直接抛出异常，让开发者们介入处理，不会容许这种隐形的 bug 产生，危机系统的稳定性。

6. 高性能；Java 属于一种先编译后解释执行的语言，它没有类似于嵌入式语言那种直接编译的语言快。但是在很多场景下性能对于系统而言是非常重要的，为了更好的支持这些场景，Java 开源社区一直在努力的尝试、研发，在不懈的努力下，终于制作出“及时”编译程序，它可以在运行时 (run-time) 把 Java 产生的字节码文件及时的翻译成特定中央处理器 (CPU) 的机器代码，即实现全编译。

2.1.2 WEB 程序开发技术—JSP

JSP 语言，它是一种专门服务于 Java 全栈开发者的前端语言。它的高级特性标签可以在 HTML 网页中直接插入 Java 代码，编译时会翻译成才能 Java 文件。在经过编译后我们查看生成的文件，可以发现 JSP^[3]实际上就是一个 Servlet，用于完成用户要实现的功能，开发者们可以通过它结合 HTML，CSS，JS 等语言来编写 JSP，通过表单来获取用户的提交的数据，可以直接使用后台所返回的数据，进行显示，是一个功能非常丰富的语言。

JSP 技术的主要特性

1. 对 Servlet 的扩展：JSP^[10]可以说是 Servlet 的升级版，它不仅具有 Servlet 的所有特性，还包含 JSP 中的隐式对象、预定义标记、表达式语言和自定义标记，使 JSP 易于开发任何应用程序。

2. 功能强大：编译生成的 Servlet，由字节码组成，所有 java 特性都适用于 jsp，比如健壮、动态、安全、独立于平台。

3. 可移植性：JSP 标记将由服务器端 web 容器处理和执行，从而使它们独立于浏览器和 j2ee 服务器。

4. 灵活性：允许定义自定义标记，开发人员可以在 JSP 中填充可协商使用的任何类型的、基于框架的标记。

5. 简单性：JSP 易于学习、理解和开发。jsp 比 servlet 更加容易编写，因为它们允许您将 Java 代码直接嵌入 HTML 页面，以防 servlet 将 HTML 嵌入 Java 代码中。

2.1.3 Tomcat 应用服务器

Tomcat 是由 Apache 开发的一款 Servlet 容器，他可以作为服务器部署 Java 开发的项目，把编译好的文件直接复制到相应的目录下，配置好端口文件即可运行，非常的方便，是现今全栈开发者们一种所钟爱的开发模式。他可以部署多个项目，支持多个项目在同一个 Tomcat 下运行，只要配置好其配置文件就行，否则会产生请求冲突。Tomcat 工作机制示意图，如图 2-1 所示。

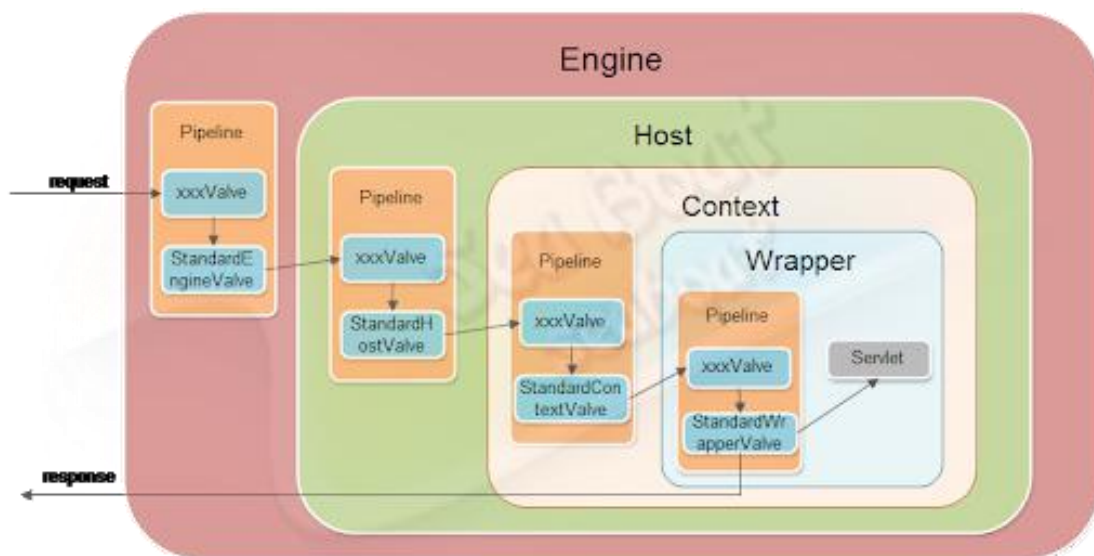


图 2-1 工作机制图

2.1.4 MySQL 数据库

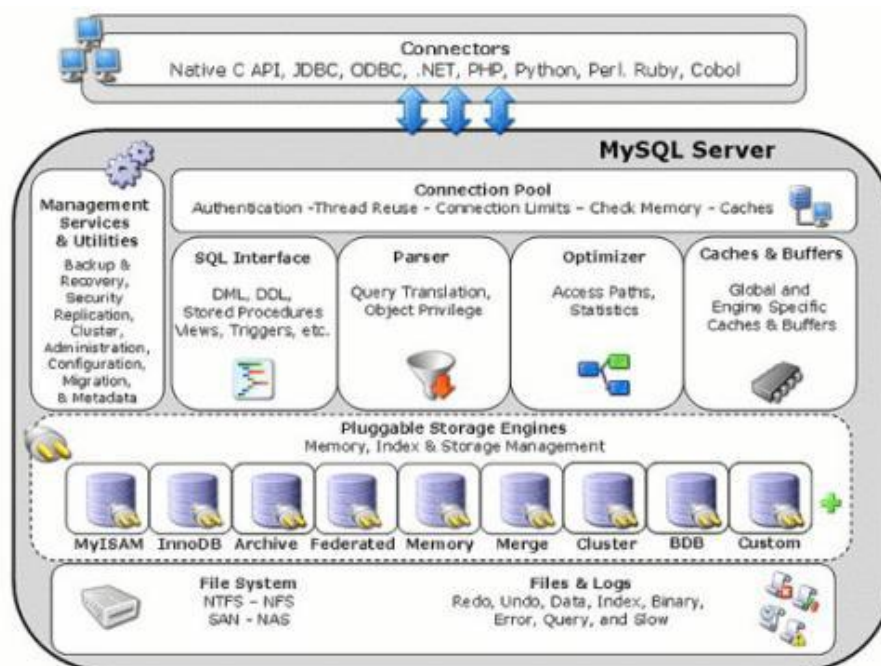


图 2-2 MySql 体系结构

MySQL^[2]是一种关系型数据库，它将所有的数据以行列的形式存储于表中，每一行数据作为两个单位进行存储，每个表在数据库中就是一个二叉树，使用索引可以快速检索出用户所需要的数据，但是表也是有大小的，不能无限的存储，所以当复杂的系统使用 MySQL 作为数据库时，开发者应该设计分库分表，分库就是用两个以上的数据库作为项目的数据库，分表是通过取模，时间等方式让类型相似的数据存储于不同的表中，以防止表的数据量过大，导致查询缓慢，影响系统性能。

构成 MySQL 的八大模块

1. 数据库驱动：用于连接数据库，JDBC。
2. 数据库连接池：管理数据库实例。
3. 管理服务 and 工具组件：系统管理和控制工具。
4. SQL 接口组件：接受用户发送的 SQL 语句，返回 SQL 语句所产生的结果。
5. 语法分析器 (Parse)：SQL 命令传递到解析器的时候，SQL 语句会被解析器所解析和验证，看其是否符合 SQL 语句语法的标准，如果错误会返回错误原因，以使用户定位错误位置，是个脚本。
6. 查询优化器：发送的 SQL 语句会经过查询优化器的优化，优化成执行效率更高，效果相同的语句。
7. 缓冲组件：查询缓存。
8. 插件式存储引擎：MySQL 的存储引擎是基于表的，和数据库没有相关关系。

2.2 MVC 设计模式

设计模式就是经过几代开发者们实操经验，总结出一套适合在某个场景下，最为合适的代码结构，它不仅可以有效提升系统的运行效率，减少代码量，还便于后期的重构优化。毫无疑问，设计模式对于刚入行的开发者们非常重要，可以快速帮助他们成长为企业的支柱，帮助企业更好的成长。在项目中如果合理的运用设计模式，可以非常方便的解决许多问题，他在项目中就像大厦的基石一样重要，可以有效避免很多问题，这也是其被广泛使用的原因。

2.2.1 MVC 设计模式

MVC 模式代表 Model-View-Controller（模型-视图-控制器）模式。

MVC 流程图如图 2-3 所示：

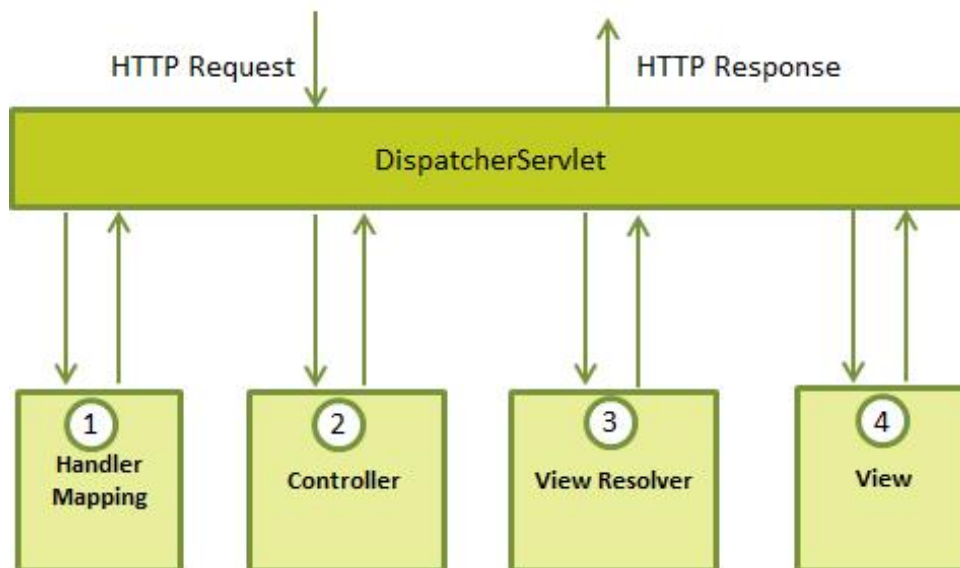


图 2-3 MVC 流程图

- Model（模型）：模型代表数据库对应的 Java 实体对象也称为 POJO 或者实体类。
- View（视图）：视图代表返回服务层处理的结果，以 Model 的形式返回。
- Controller（控制器）：控制器的功能主要是通过数据库和外界交互，向外界提供查询的数据。

2.2.2 Spring MVC Request 流程

1. 前端控制器：所有的前端请求都会先访问 Spring^[1]的 DispatcherServlet，它会根据请求所携带的信息，派送到指定的组件执行，返回结果。这是常用的 Web 应用程序模式。

2. 处理映射: DispatcherServlet 会将所有请求转发给 Spring MVC 控制器 (controller) , 控制器是处理前端请求的 Spring 组件, 因此 DispatcherServlet 需要知道应该将所有拦截的请求发送给哪个指定控制器, 所以 DispatcherServlet 会查询一个或多个处理器映射 (handler mapping) 其实就是请求路径, 来确定下一站的 Controller。
3. 控制器处理: DispatcherServlet 将会拦截所有的请求信息转发给该请求对应的控制器, 到了控制器, 控制器会处理请求携带的信息, 将处理结果封装成 Model 返回, Model 通常由 POJO 组成。
4. 控制器将模型和视图名返回给前端控制器: Controller 会将请求和请求产生的模型数据和用于渲染的视图名发送回 DispatcherServlet, DispatcherServlet 将根据视图名选择相应的视图解析器 (ViewResolver) 。
5. 视图的实现 (可能是 JSP) : 将它交付的模型数据, 使用视图解析器将 Model 渲染成美丽页面, 然后通过响应对象传递给客户端。

2.3 本章小结

本章主要对物流管理系统开发过程中所使用到的主要技术进行分析和探讨, 其内容主要包括, Java 相关技术的介绍其中包括 Java 特性、JSP 前端开发技术、Tomcat 的工作机制等及相关工具; 对 MySQL 数据库的介绍, 以及特性; 最后对 MVC 设计模式的介绍, 以及 MVC 的 request 流程做了一定的阐述。

第三章 物流系统的分析

3.1 系统可行性分析

系统的可行性分析用于判定一个系统是否可行的重要依据,其中主要包括技术可行性分析、经济可行性分析、安全可行性分析和运营可行性分析四部分。基于可行性分析得出的结论可以有效为管理者提供一种前进的方向,为项目的开发具有指导意义,以下是对该系统进行的可行性分析:

1. 可行性分析的前提

基于 SSM 物流管理系统由集团中心、快递中心、门店中心三者构成,其主要功能就是门店需求上报、集团中心审核上报需求、下方物资、快递员配送相应物资和其他一些管理人员账户、管理快递公司、物资统计等构成,实现对其增删改查等相关操作。

2. 技术可行性分析

基于 SSM 物流管理系统主要使用了 Java WEB 相关开发技术,采用 Spring + Spring MVC + MyBatis + Tomcat + Jsp + MySQL 技术组成来完成此项目。以上技术为 Java 全栈开发者们使用 Java 开发 WEB 项目的一揽子技术,整个开发流程已经经过多年的时间校验,其高效、准确的构建项目,并且其开源社区的活跃度等条件为我们开发此项目提供了巨大的保障。

3. 经济利益可行性分析

基于 SSM 物流管理系统在开发完成之后,类似于苏宁易购,居然之家等连锁店的企业可以直接使用配送门店之间的所需物资,大大减少公司使用其他同城快递所花费的金额,并且提供整个供应链的电子化管理,节省人力、企业投入金额,为企业的整体盈利做好开源。

4. 运营操作可行性分析

基于 SSM 物流管理系统主要用户群体是公司集团的管理审核、下方物资人员、门店店主、配送物资快递员等,其整理流程操作简单、快捷,具有较高的可行性。

5. 结论

根据以上分析,认为基于 SSM 物流管理系统在技术、经济利运营操作等方面具有较高的可行性。且项目整体开发时间充足,最本人所学知识又具有较高的契合度,是一个完美的把理论变为实践的过程,所以经过本人和老师的探讨之后,决定开发此系统。

3.2 系统业务功能需求分析

本系统主要是提供给类似于居然之家、苏宁易购具有连锁店的配送服务的企业使用,根据国内外这种公司的配送调研、分析结果及上一章对物流理论的研究结论、总结出一套适合于此类公司的配送服务流程。

物流配送基本流程如图 3-1 所示:

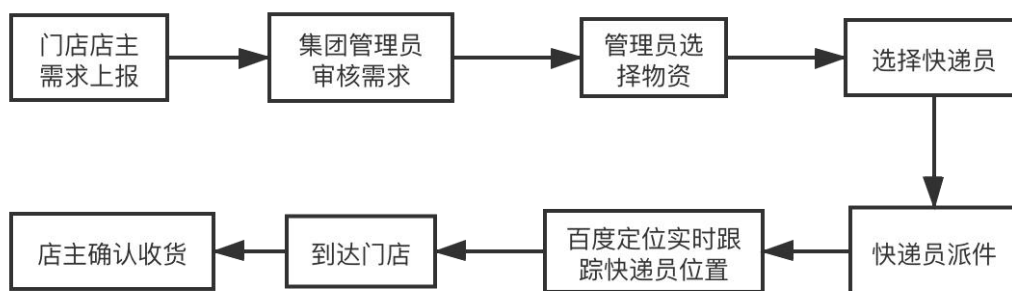


图 3-1 物流配送基本流程图

物流配送流程除了包括店主需求上报、生成物流订单、集团中心管理员审核需求、集团管理员选择物资、快递员配送物资、快递员开始派件、定位系统实施计算距离、时间、店主确认收货等基本流程外，还有快递员账号管理、合作快递公司管理、物资管理、物资配送统计、组织架构管理、门店管理公司政策的发布、阅读等功能，以保证物流系统的正常运行，其每个功能的具体分析在后续中详细说明。

3.2.1 集团中心

基础管理功能：

1. 用户登录注册：主要是在网页端对集团中心人员进行用户注册、登录、验证等功能。
2. 集团政策发布：集团中心会发布公司最新的一些政策、活动，此政策可以设置开始时间、结束时间，以及政策的查看对象等功能，并且当政策过多时可以置顶其最主要的、最及时的政策。其次还可以根据开始时间、结束时间搜索政策。
3. 集团政策阅读：后台管理系统会统计此政策的阅读人数，阅读次数，并且生成阅读报表工管理人工做汇总，且可以很具开始时间、结束时间搜索该政策阅读记录。
4. 组织机构：当公司的组织机构有所调整时，我们可以及时的调整机构，新增组织机构、删除组织机构、修改组织机构信息、为组织机构分配资源（后台管理系统权限）等功能。
5. 角色管理：每个组织机构可以对应一种或者多种角色，针对不同的角色分配不同的权限，含有新增、删除、分配、修改角色信息等功能。
6. 资源列表：由于该后台管理功能垂直交叉，较为复杂，因此为了后期重构方便，特对其主要接口调用进行表格化的显示。

物流管理功能

1. 快递公司管理：对合作的快递公司进行管理，新增、删除操作，显示快递公

- 司列表等功能。
2. 快递员管理: 对合作公司下的所属快递员进行显示信息列表, 新增快递员列表、删除快递员列表。
 3. 物资管理: 对集团所有类别物资进行分类管理, 含有显示物资信息列表、添加物资列表, 删除物资等功能。
 4. 门店列表管理: 对集团所属旗下的门店、直营门店进行管理, 含有显示门店信息列表、修改门店信息、新增门店、删除门店等功能。
 5. 需求上报审核: 对集团旗下门店进行的需求进行上报审核, 若审核通过则生成订单, 若驳回审核, 需提供驳回审核原因, 并且还可以根据审核状态 (待审核、审核通过、审核驳回)、开始时间, 结束时间等进行搜索。
 6. 物流订单列表: 提供对不同的订单状态进行显示管理, 包括在待完善信息列表中对审核通过的订单进行发货、删除处理、支持批量操作, 而在订单列表中对主界面进行订单显示, 根据配送中、派单中、配送完成三种状态以及订单号进行搜索等功能。
 7. 订单配送统计: 对集团中心下方的所有订单进行统计, 显示器门店名称、配货次数、签收次数、配送那些物资数量进行统计, 且可以根据起始时间、截止时间进行搜索等功能。

3.2.2 APP 端门店中心

1. 登录功能: 提供对于门店中心的店主进行登录、验证、忘记密码等功能
2. 订单列表: 对该门店下上报的需求生成的订单进行显示, 当订单列表过多时, 只显示固定值, 提供点击查看更多订单一栏继续查看剩余订单, 并且提供根据订单状态 (已发货, 派送中, 已收货)、订单编号查询以及根据日期查询订单功能, 最后若需要查看订单详情时, 点击此订单, 跳转至订单详情页。
3. 政策下达: 集团中心针对门店店主发布的政策、活动进行显示, 并且可以根据政策标题的关键字进行搜索该政策, 点击某个政策时, 跳转至政策详情页, 显示该政策的具体信息。
4. 报需功能:
 - a) 需求上报: 需求上报页面, 提供选择物资类型、填写物资名称等, 如果添加错误可以删除, 然后提供一栏填写扩展物资的面板, 提交之后会弹出会话框提示是否确认上报, 点击确认之后生成预订单带审核。
 - b) 上报列表: 显示该门店上报的所有需求列表信息, 并且可以根据审核状态 (审核通过、未审核、审核不通过)、日期进行查询列表信息。
 - c) 上报统计: 由于上报统计需求查询数据库所有信息, 查询时间较长, 因此我们会先加载一张处于加载中的图片做缓冲, 等待一定时间后再跳转至统计界面, 该界面用扇形图的方式统计出各种状态的需求占总上报需求的比例, 并且提供根据开始时间, 截止时间区间搜索统计出区间内的需求所占比例。
5. 我的: 我的菜单栏提供账号、类型、所属等账号基本信息, 另外显示个人设置、订单列表、退出系统功能列表。
 - a) 个人设置: 该页面显示此门店的信息页面, 另外提供修改密码和退出登录功能。
 - b) 修改密码: 该页面需要输入用户旧密码、再输入两次用户新密码经过系

统的校验后，方可更新密码成功。

- c) 订单列表：同上，只不过提供该接口的另一种进入方式。
- d) 退出系统：直接返回登录界面。

3.2.3 APP 端快递员中心

1. 登录功能：提供对于快递员中心的快递员进行登录、验证、忘记密码等功能。
2. 订单列表：对该快集团中心派给该快递员的订单进行显示，当订单列表过多时，只显示固定值，提供点击查看更多订单一栏继续查看剩余订单，并且提供根据订单状态（已发货，派送中，已收货）、订单编号查询以及根据日期查询订单功能，最后若需要查看订单详情时，点击此订单，跳转至订单详情页。
 - a) 开始派送：在订单详情页中，提供“开始派送”按钮，点击该按钮后，会自动获取该快递员当前百度坐标，绘制出该当前快递员坐标和配送门店坐标之间的配送路线图，并且此时后台自动修改该订单的状态值为派送中。
 - b) 查看详情：在配送路线图中，提供“查看详情”按钮，该按钮被点击之后会弹出会话框显示，距离终点的所需时间和距离终点路程信息。
 - c) 查看派送地图：当下次在进入订单详情时，在“开始派送”按钮处显示出新的“查看派送地图”按钮，点击该按钮会跳转至订单配送线路图，并且继续获取快递员实时坐标，直到某次快递员进来之后快递员坐标和门店坐标相距小于 100M 之后，会弹出已到达目的地窗口，然后快递员会向门店店主发送一条验证码，然后快递员获取到店主的验证码后，点击确认门店收货即可配送成功，修改订单状态。
 - d) 派送成功：再次进图订单详情页后，只会显示出订单的配送信息等，不会弹出按钮等功能。
3. 政策下达：集团中心针对门店店主发布的政策、活动进行显示，并且可以根据政策标题的关键字进行搜索该政策，点击某个政策时，跳转至政策详情页，显示该政策的具体信息。
4. 订单统计：由于统计需求查询数据库所有信息并且会执行一些复杂计算，时间较长，因此我们会先加载一张处于加载中的图片做缓冲，等待一定时间后再跳转至统计界面，该界面用扇形图的方式统计出各种状态的历史订单占该快递员总订单的比例，并且提供根据开始时间，截止时间区间搜索统计出区间内的订单所占比例。
5. 我的：我的菜单栏提供账号、类型、所属等账号基本信息，另外显示个人设置、订单列表、退出系统功能列表。
 - a) 个人设置：该页面显示该快递员的信息页面，另外提供修改密码和退出登录功能。
 - e) 修改密码：该页面需要输入用户旧密码、再输入两次用户新密码经过系统的校验后，方可更新密码成功。
 - b) 订单统计：同上，只不过是提供该接口的另一种进入方式。
 - c) 订单列表：同上，只不过是提供该接口的另一种进入方式。
 - d) 退出系统：直接返回登录界面。

3.2.4 集团中心、门店中心、快递员中心三者之间关系

集团中心需要管理门店、快递员，并且审核门店需求、联系快递员配送物资；门店中心需要上报需求，接受快递员配送的物资；快递员中心需要接受集团中心的派单任务，想门店店主配送物资；三者之间的关系如 3-2 所示：

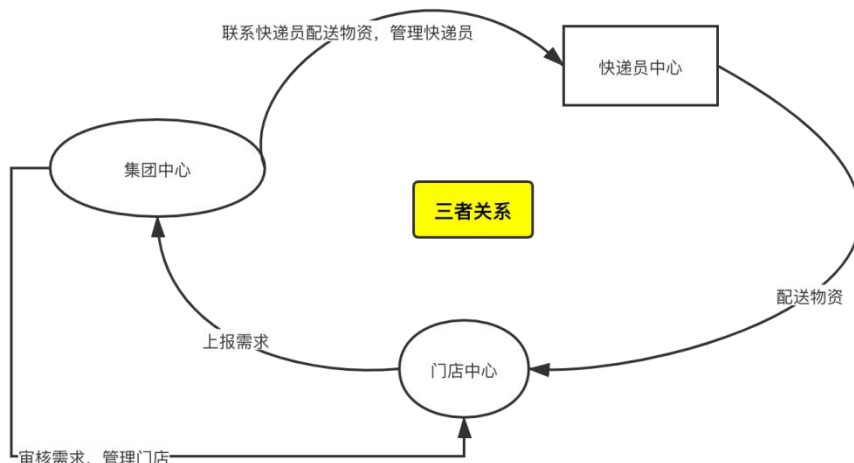


图 3-2 三者关系图

3.3 系统架构性能需求分析

系统安全性

众所周知，项目的安全性是决定软件质量的重要因素之一，虽然基于 SSM 的开发的系统本省框架具有较高的安全性，但是该系统中涉及多方不同的信息，其中有需求上报，快递员实时位置，物资下放等，如果有人蓄意攻击此系统，都可能会导致该系统某些功能的瘫痪，会给企业带来巨大的损失，因此必须采取一定的措施，防止用户越权，定义出每个用户群组的功能，使其相互独立，又相互联系，防止内部人员的非法操作等原因导致系统瘫痪。

系统的合理性

在开发项目的过程中，一定要保证系统开发的合理性，充分考虑系统的实际性能和硬件的具体要求，不能一味地只追求新的技术，而忽略系统本身的整体合理性。

系统操作的易用性

在设计系统的操作界面时,必须充分的考虑的不同层次的服务人员的计算机操作水平,总体来说现在全民的计算机水平偏高,简单的系统操作还是可以保证完成的,但是由于少数的个人可能计算机使用较为艰难,因此我们开发过程中一定要考虑到他们托需求,在较为复杂的地方,时刻弹出提醒窗口,保证使用者的正常使用,使之可以熟练的完成属于自己模块的操作流程。

3.4 本章小结

本章主要对物流管理系统进行全面的可行性分析、业务功能需求分析、架构性能分析三部分。基于物流管理系统分析的结论,接下来将对该系统进行详细设计,规划与实现。

第四章 物流系统的设计

4.1 系统架构设计

在完成系统需求分析之后，工作重心将从分析阶段转为设计阶段。在设计阶段本人将根据前期系统的分析结果，进一步确定系统的最终目标，完成系统功能的设计图，确定系统的架构方式，进而搭建开发环境。

4.1.1 系统总体架构设计

随着企业级的应用管理系统日益复杂，开发周期不断缩减，对系统的稳定性、可靠性等指标的不断提高，为了满足以上需求，提升开发效率，现流行的开发架构一般采用 MVC 模式进行架构搭建开发。其主要实现方式是：对业务功能进行拆分，使其页面的显示部分与逻辑处理部分分离、解耦，使其可以独立运行、部署，只是在接口处提供数据交互的部分，这种模式可以大大降低项目的耦合度，拆成多个子项目部署，提高项目的整体运行效率，支持高并发、高稳定。

一般分为四层：POJO 层、DAO 层、SERVICES 层、CONTROLLER 层；

- POJO 层：一般是用户，模块的实体属性对应数据库；
- DAO：用 Mybatis 框架操作 MySQL 数据库，将获取的数据封装成 POJO；
- SERVICES 层：用于处理复杂的业务逻辑，针对各种场景的业务功能开发；
- CONTROLLER 层：手机用户的请求进行响应，并调用 SERVICES 层的功能进行处理请求，最后返回响应结果；

系统的总体结构设计如图 4-1 所示：

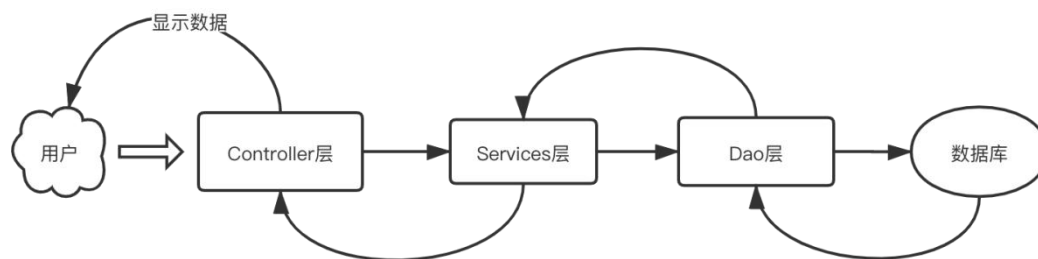


图 4-1 系统总体架构图

4.1.2 业务模块设计

增量模式开发模式使得功能模块在划分方面是从使用用户组的角度去思考，充分满足用户的需求，也就是根据面向用户的不同向不同用户对系统的使用情况可进行功能模块划分，这样划分后的模块就是一个单独的子系统，方便开发者们开发测试，耦合性较低，大大降低系统的复杂度。

根据分析得出，以 SSM 为基础的物流管理系统可分为 APP 端，网页版的后台

管理页面两大部分，其中 APP 端又分为快递员模块，门店模块。

其系统 APP 端模块如图 4-2 所示：

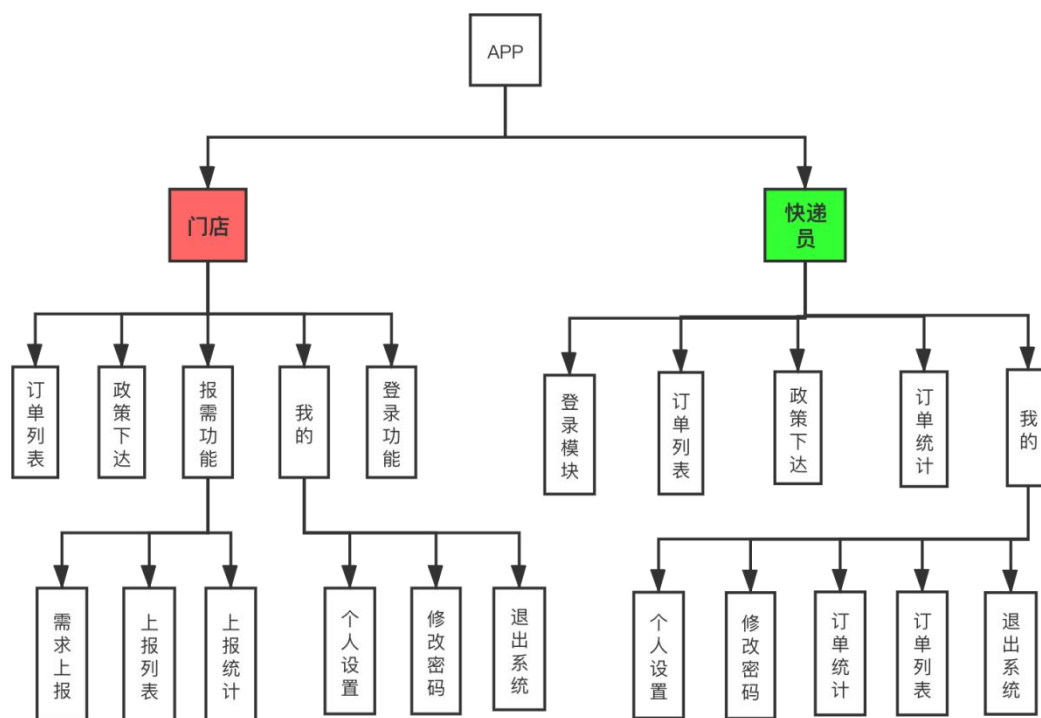


图 4-2 APP 功能模块图

其系统网页端模块如图 4-3 所示：

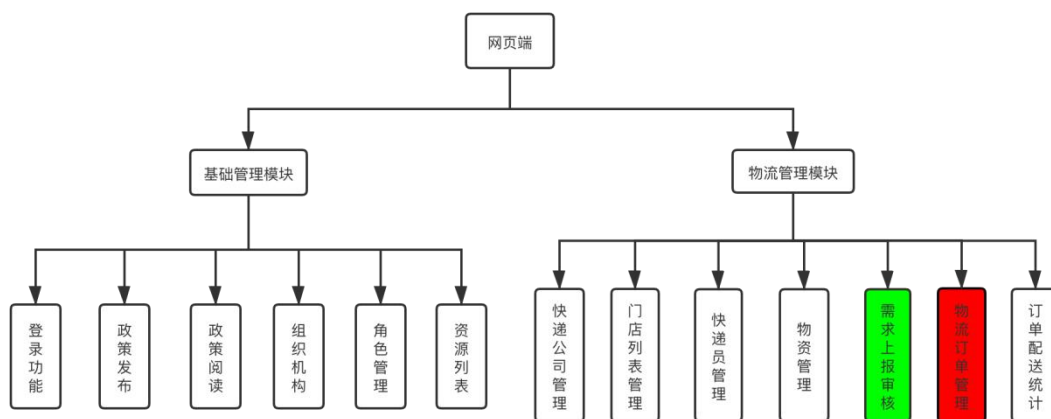


图 4-3 网页端功能模块图

4.1.3 核心模块详细设计书

登录模块

物流管理系统的集团人员、快递员、门店店主通过账号，密码登录该系统，如果密码输入错误，则不允许该员工登录，并且该账号登录五次以上错误时，会自动锁定该账号，使用时需要后台人员解封，登录成功后，该登录信息会自动保存在 SESSION 中以便跳转每个页面时，检查该用户是否登录，若登录，则跳转成功。具体模块流程如图 4-4 所示：

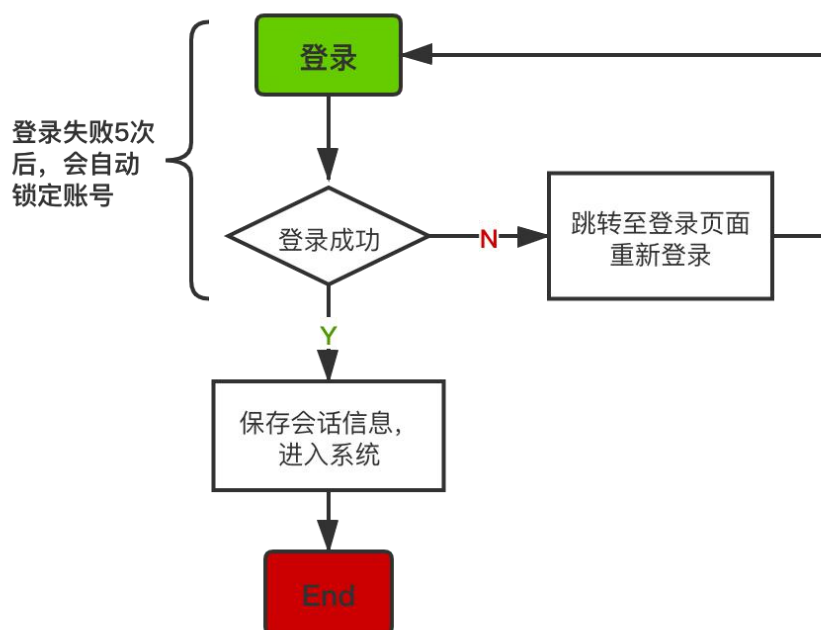


图 4-4 登录流程图

需求上报

需求上报需要店主登录 APP 端，根据现有物资，选择物资类型，物资数量，支持批量添加不同的物资，然后提供扩展物资选填，点击提交，生成预订单信息入库，若提交时，会检查是否含有添加物资，若无，则提示重新添加物资，提交成功后等待集团中心审核。需求上报流程图，如图 4-5 所示。

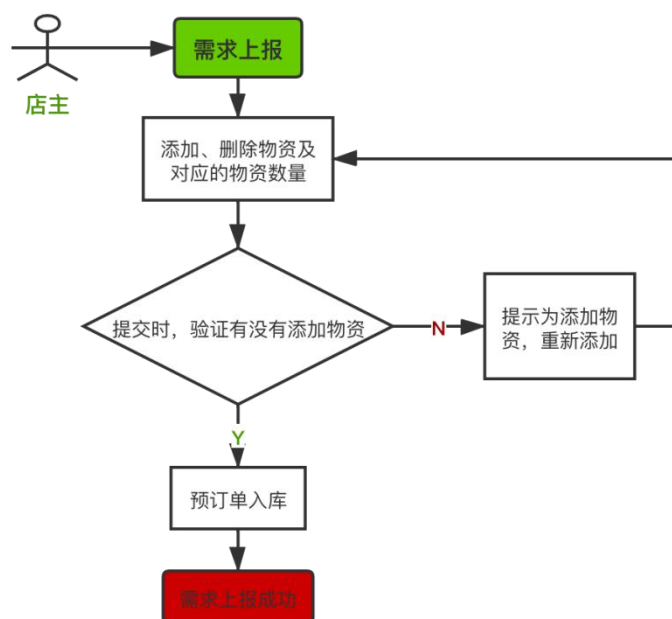


图 4-5 需求上报流程图

需求上报审核

门店上报的需求都会汇总到物流管理后台界面，其每一行会显示具体的需求详情，管理员审核需求时，会考虑实际情况，做出批复，若是审核通过，则生成订单，若是驳回，则说明驳回原因，以便店主知道需求驳回原因是哪方面，审核完成后，会提示是否审核成功。需求上报审核流程图，如图 4-5 所示。

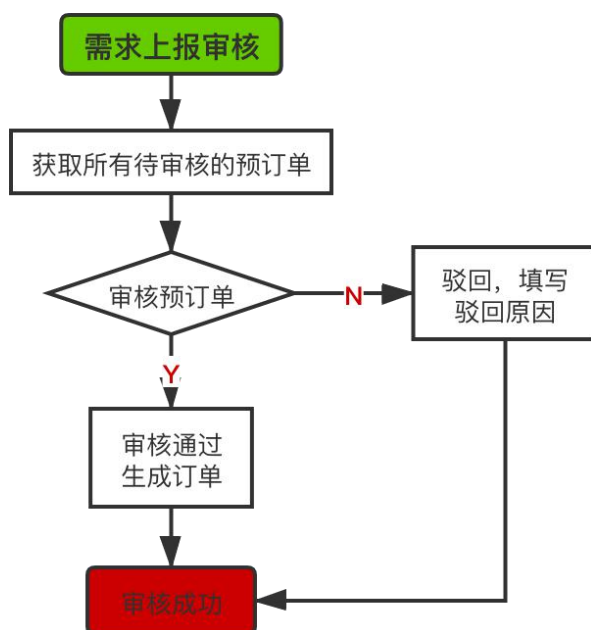


图 4-5 需求上报审核流程图

订单配送流程

审核后的订单，在物流订单列表中-待完善信息列表中显示，此时订单处于待发货状态，当管理员批量选择发货时，必须为此订单选择快递公司，快递员，才可发货，此时订单处于已发货状态，当快递员接受订单开始派送时，获取快递员坐标，计算派送订单距离，时间，此时订单处于派送中，当快递员到达门店时，可以点击按钮操作，向门店店主发送验证码，然后想店主索要店主已收到的验证码，输入订单校验，此时该订单已经派送成功。订单配送流程图，如图 4-6 所示。

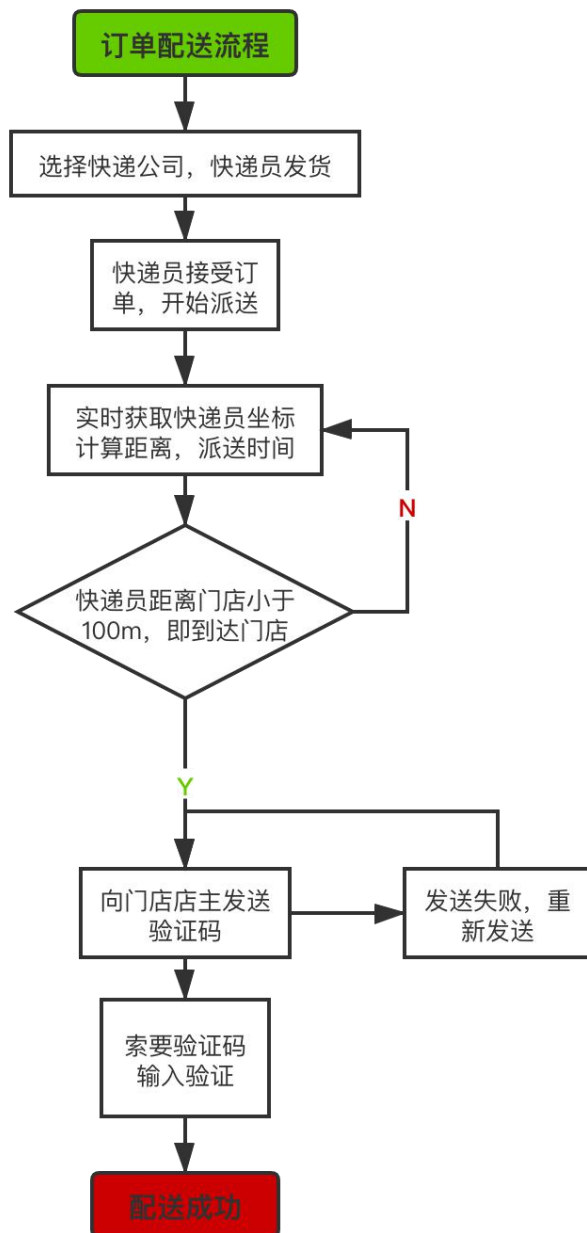


图 4-6 订单配送流程图

4.2 系统数据库设计

数据库是每个项目的基石，它不仅影响着整个系统的性能，而且对于系统的稳定安，安全性具有较高的依赖性，数据库的设计完全按照三范式的要求，合理、高效的设计出符合系统的数据库，同时设计时，也要考虑到数据库中每个表的后期的扩展性，方便系统的升级，下面将对核心数据库的库表结构用表格语句进行说明。

1. 管理员、快递员、门店店主账户信息表 (sys_account)

表 4-1 账户信息表

字段名称	字段类型	Comment
manager_id	bigint	'归属者 ID'
id	int(10)	auto_increment
accountName	varchar(20)	'用户账号 courier:phone
password	varchar(100)	'密码'
accountType	varchar(20)	'用户类型 (1 门店, 2 快递员, 3 集团中心)'
description	varchar(100)	'描述'
state	varchar(3)	'是否可用 1: 启用, 2: 禁用'
createTime	datetime	null
deletestatus	int(1)	'逻辑删除状态 0:存在 1:删除'
groupId	int	'组织架构 ID'
token	char(36)	null
area	varchar(100)	'片区'
real_name	varchar(50)	'姓名'
id_car	char(18)	'身份证'
code	varchar(50)	'渠道编码'
section_name	varchar(100)	'渠道名称'
employeesType	int(2)	'0 为内部店员, 1 为外部店员'
employeesClass	int(2)	'必填 (0: 营业人员; 1: 客户经理; 2: 代理商; 3: 社区直销; 4: 校园直销; 5: 其他; 6: 客服人员; 7: 地市客服人员)'
employeesNum	varchar(20)	'员工编码'
dianyuanNum	varchar(20)	'店员编码'
auth_flag	int	'审核标示(1 未审核 2 成功 3 失败)'
user_flag	int	'用户标示(1 普通用 2 代理商)注册用户默认为 0'
longitude	varchar(20)	'经度'
latitude	varchar(20)	'纬度'
auth_reason	varchar(100)	'审核通过提示'

city	varchar(255)	'地市 (冗余) '
company	char(32)	'归属 ID (当 accountType 为 2 时) '

2. 组织机构信息表 (sys_group)

表 4-2 组织结构信息表

字段名称	字段类型	Comment
id	int	auto_increment
parentId	int	'父级'
name	varchar(100)	'组织名称'
description	varchar(200)	'描述'
createTime	datetime	null
modifyTime	datetime	'修改时间'
remove	char	'删除标示'
removeTime	datetime	'删除时间'
level	int	'等级'

3. 后台管理系统资源列表信息表 (sys_resources)

表 4-3 资源列表信息表

字段名称	字段类型	Comment
id	int	auto_increment
name	varchar(50)	菜单名称
parentId	int	'用于区分一级、二级目录'
resKey	varchar(50)	null
type	varchar(40)	'0:目录, 1: 菜单, 2: 按钮'
resUrl	varchar(200)	Request URL
level	int	null
description	varchar(200)	描述

4. APP 端菜单详情表 (sys_wx_menu)

表 4-4 APP 端菜单详情表

字段名称	字段类型	Comment
id	int	auto_increment
name	varchar(255)	'菜单名称'
img	varchar(255)	'菜单图片地址'
address	varchar(255)	'菜单地址'
isDel	int	'是否删除 (0 正常, 1 删除) '
createTime	datetime	'添加时间'
delTime	datetime	'删除时间'
ioc	varchar(20)	'展示图标名字'
phrase	varchar(16)	'短语 (主要用于底部菜单展示) '

5. 门店信息表 (mkl_store_demand)

表 4-5 门店信息表

字段名称	字段类型	Comment
id	int	主键
store_name	varchar(255)	'门店名称'
store_address	varchar(255)	'门店地址 (收货地址) '
store_longitude	double(12,8)	'门店经度'
store_latitude	double(12,8)	'门店纬度'
store_img	varchar(255)	'门店照片 (门头) '
store_shopowner_phone	char(11)	'门店店长电话 (收货人电话) '
store_shopowner_name	varchar(255)	'门店店长姓名 (收货人姓名) '
channel_code	varchar(16)	'门店渠道编码'
vendor_id	int	'门店信息'
type	int(2)	'门店类型(0 表示直营,1 表示现金也就是社会门店)'
create_phone	varchar(255)	'创建人'
create_time	datetime	'创建时间'

6. 公司旗下品牌信息表 (mkl_vendor)

表 4-6 公司旗下平拍信息表

字段名称	字段类型	Comment
id	int	auto_increment
vendor_name	varchar(255)	'品牌名称'
create_time	datetime	'创建时间'
create_phone	varchar(255)	'创建人电话'
delete_flag	int(2)	'数据状态 (0 正常, 1 删除) '

7. 需求上报审核表 (mkl_store_demand)

表 4-7 需求上报审核表

字段名称	字段类型	Comment
id	int(100)	'主键'
store_name	varchar(255)	'门店名称'
channel_code	varchar(255)	'门店渠道编码'
store_shopowner_phone	varchar(255)	'上报人电话'
store_shopowner_name	varchar(255)	'上报人姓名'
materialContent	varchar(255)	'物资名称'
materialNumber	varchar(255)	'物资数量'
isdelete	int(1)	'是否删除',
report_time	datetime	'上报时间',
examine_time	datetime	'二级审核时间'
examine_two_name	varchar(255)	'二级审核人姓名'
examine_two_phone	varchar(255)	'二级审核人电话'
expanding_demand	varchar(255)	'扩展需求',
examine	int(1)	'审核标示, 0 未审核, 1 二级审核通过, 2 二级审核驳回, 3 一级审核通过, 4 一级审核驳回'
examine_reason	varchar(255)	'二级审核驳回原因'

8. 物流订单详情表 (mkl_mail_info)

表 4-8 物流订单详情表

字段名称	字段类型	Comment
id	varchar(20)	主键 (年月日时分秒毫秒+3 位随机数)
shipperName	varchar(255)	'发货人姓名'
shipper	varchar(11)	'发货人电话'
shipTime	datetime	'发货时间'
groupId	varchar(4)	'发货人的组织 id'
qdmc	varchar(255)	'收货人渠道'
store	varchar(255)	'收货门店名称'
address	varchar(255)	'收货具体地址'
consignee	varchar(50)	'收货人'
phone	varchar(11)	'收货人电话'
logistics	varchar(20)	'物流单号 (冗余) '
predictTime	varchar(10)	'邮寄预计时长'
factTime	varchar(225)	'实际邮寄时长'
state	int(1)	'发货状态 (默认 0 为待发货, 2 快递员接收订单已发货, 3: 派送中, 4: 已收货, 5 货物已送达, 6 已签收) '
leadTime	datetime	'收货时间'
signPeople	varchar(10)	'签收人'
signPhone	varchar(11)	'收货电话'
isDelete	varchar(1)	'是否删除 (必须是待邮寄状态) '
deleteReason	varchar(100)	'删除原因'
question	varchar(100)	'疑问'
rejectReason	varchar(100)	'拒绝收货原因'
isReplace	varchar(1)	'是否代收(0 代表否, 1 代表是)'
agentName	varchar(100)	'代收人姓名'
agentPhone	varchar(100)	'代收人电话'
brand	varchar(100)	'品牌'
createTime	datetime	'数据插入时间'
material	varchar(100)	'物资 id (以 “, ” 进行分割) '
materialContent	varchar(100)	'物资名称 (以 “, ” 分割) '
materialNumber	varchar(100)	'物质数量 (以 “, ” 分割) '
courierPhone	char(11)	'快递员电话'
courierName	varchar(16)	'快递员姓名'
recipientTime	datetime	'快递员收件时间'
serviceTime	datetime	'快递员送达时间'
store_longitude	double(12,8)	'门店经度'
store_latitude	double(12,8)	'门店纬度'
channel_code	varchar(255)	'收货地址渠道编码'
orderImg	varchar(255)	'订单照片图片信息'
storeImg	varchar(255)	'门店的图片的'

9. 快递公司详情表 (mkl_express_store)

表 4-9 快递公司详情表

字段名称	字段类型	Comment
id	char(32)	主键 ID
company_name	varchar(255)	'公司名称'
company_address	varchar(255)	'公司地址'
create_time	datetime	'创建时间'
create_phone	char(11)	'创建人电话, account_name'
delete_flag	int	'数据状态 (0 正常, 1 删除)'

10. 快递员实时坐标信息表 (mkl_courier_store)

表 4-10 快递员实时坐标表

字段名称	字段类型	Comment
id	int(100)	'主键'
logistics	varchar(100)	'订单号 (订单 id) '
courier_Phone	varchar(255)	'快递员电话'
courier_longitude	double(12,8)	'快递员当前经度'
courier_latitude	double(12,8)	'快递员当前纬度'
actualTime	datetime	'实时定位时间'

4.3 本章小结

本章的主要工作是对物流管理信息系统的整体系统的架构、业务功能模块的划分、核心模块的设计、数据库中核心表的表结构进行详细的设计和说明,尤其是在数据库中由于设计的表多而复杂因此,设计初完全按照数据库的范式,进行合理的拆解,划分,进而设计出高效、稳定的数据库系统。

第五章 物流系统的实现

5.1 搭建开发环境

基于 SSM 的物流管理系统开发过程主要用到的软件开发环境如下：

1. JDK: JDK 1.8 是目前使用 Java 语言开发时, 使用最多、功能最有效, 最全面的版本, 与 Java 8 的新特性可以完美的结合, 使开发中很多 bug 可以有有效的避免, 比如空指针异常、重复循环等。
2. Spring + MyBatis: 这种框架模式, 是现在使用 Java 语言开发的主流框架之一, 它的简单、便捷、高效, 不需要开发者们重复的配置文件, 直接套轮子进行开发, 大大的节省了项目的开发时间, 降低人力成本。
3. MySQL 数据库: MySQL 5.7 是业内公认的比较稳定的数据库版本, 并且该数据库免费开源是其他数据库不具有的, 另外它的特性非常多, 比如事务一致性, MVCC 等。
4. WEB 应用服务器: 本次项目所采用的 Tomcat 8 是目前最新的版本它的功能强大、稳定性高等以前的版本所不能及的, 非常适合项目管理系统这种中小量级的项目开发。

5.2 系统的主要框架及其配置

Spring 全家桶是一揽子轻量级的开源框架。它的出现不仅 Java 程序更加的方便、快捷、高效, 而且其主要目标是使搭建的系统变得越来越稳定、安全、简单、高效, 这反而促进了 Spring 框架越来越受开发者的欢迎, 形成了一个正向的循环, 现在 Spring 已经附带了很多的工具, 对于很多复杂的问题, 提供了最为直接的解决方案, 使使用者们不用再担心编写额外的代码, 从而节省开发的时间和尽力。

本系统主要采用 Spring 全家桶搭建框架, 因此需要对其进行一些配置, 具体配置如下:

1. 在项目的 WEB-INF 中新建 lib 文件夹, 向其中导入 Spring 全家桶 jia 包, 比如: spring-core-3.1.2.RELEASE.jar, spring-beans-3.1.2.RELEASE.jar, spring-context-support-3.1.3.RELEASE.jar 等。
2. 在项目的 web.xml 文件中, 添加如下配置:

```
<servlet>
    <servlet-name>springmvc</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet
</servlet-class>
    <init-param>
        <param-name>contextConfigLocation</param-name>
        <param-value>classpath:resources/spring-servlet.xml</param-value>
    </init-param>
    <load-on-startup>1</load-on-startup>
</servlet>
<servlet-mapping>
    <servlet-name>springmvc</servlet-name>
```

```

    <url-pattern>*.html</url-pattern>
</servlet-mapping>
<context-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>
        classpath:resources/spring-security.xml,classpath:resources/spring-application.xml
    </param-value>
</context-param>
<welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
</welcome-file-list>

```

3. 在项目的 src/resources/ 目录下新建 spring-application.xml 文件，其配置如下：

```

<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xmlns:context="http://www.springframework.org/schema/context"
    xmlns:tx="http://www.springframework.org/schema/tx"
    xmlns:aop="http://www.springframework.org/schema/aop"
    xsi:schemaLocation="http://www.springframework.org/schema/beans
        http://www.springframework.org/schema/beans/spring-beans-3.0.xsd
        http://www.springframework.org/schema/context
        http://www.springframework.org/schema/tx
        http://www.springframework.org/schema/tx/spring-tx-3.0.xsd
        http://www.springframework.org/schema/task
        http://www.springframework.org/schema/task/spring-task-3.0.xsd
        http://www.springframework.org/schema/aop
        http://www.springframework.org/schema/aop/spring-aop-3.0.xsd">
    <task:annotation-driven />
    <tx:annotation-driven />
    <bean
        class="org.springframework.beans.factory.config.PropertyPlaceholderC
onfigurer">
        <property name="location" value="classpath:resources/jdbc.properties" />
    </bean>
    <bean
        id="dataSource"
        class="org.springframework.jdbc.datasource.DriverManagerDataSource"
    >
        <property name="url" value="${url}" />
        <property name="username" value="${username}" />
        <property name="password" value="${password}" />
        <property name="driverClassName" value="${driverClassName}"

```

```

/>
    </bean>
    <bean id="pagePlugin"
class="com.ly.pulgin.mybatis.plugin.PagePlugin">
        <property name="properties">
            <props>
                <prop key="dialect">mysql</prop>
                <prop key="pageSqlId">.*query.*</prop>
            </props>
        </property>
    </bean>
    <bean
id="sqlSessionFactory"
class="org.mybatis.spring.SqlSessionFactoryBean">
        <property name="dataSource" ref="dataSource" />
        <property
name="mapperLocations"
value="classpath:resources/mappings/*-mapper.xml"/>
        <property name="typeAliasesPackage" value="com.ly.entity"/>
        <property name="plugins">
            <array>
                <ref bean="pagePlugin" />
            </array>
        </property>
    </bean>
    <bean class="org.mybatis.spring.mapper.MapperScannerConfigurer">
        <property name="basePackage" value="com.ly.mapper" />
    </bean>
    <bean
id="transactionManager"
class="org.springframework.jdbc.datasource.DataSourceTransactionMan
ager">
        <property name="dataSource" ref="dataSource" />
    </bean>
    <aop:config>
        <aop:pointcut
            expression="execution(public *
com.ly.service.impl.*Impl.*(..))"
            id="pointcut" />
        <aop:advisor advice-ref="txAdvice" pointcut-ref="pointcut" />
    </aop:config>
    <aop:aspectj-autoproxy/>
    <context:component-scan base-package="com.ly.service" />
    <context:component-scan base-package="com.ly.security" />

```

```
<context:component-scan base-package="com.ly.logAop" />
<context:annotation-config />
</beans>
```

5.3 系统核心模块的实现

5.3.1 后台管理系统、APP 端用户登录的实现

物流管理系统的集团人员、快递员、门店店主通过账号，密码登录该系统，如果密码输入错误，则不允许该员工登录，并且该账号登录五次以上错误时，会自动锁定该账号，使用时需要后台人员解封，登录成功后，该登录信息会自动保存在 SESSION 中以便跳转每个页面时，检查该用户是否登录时，可以跳转成功。后台管理系统登录页面如图 5-1 所示：

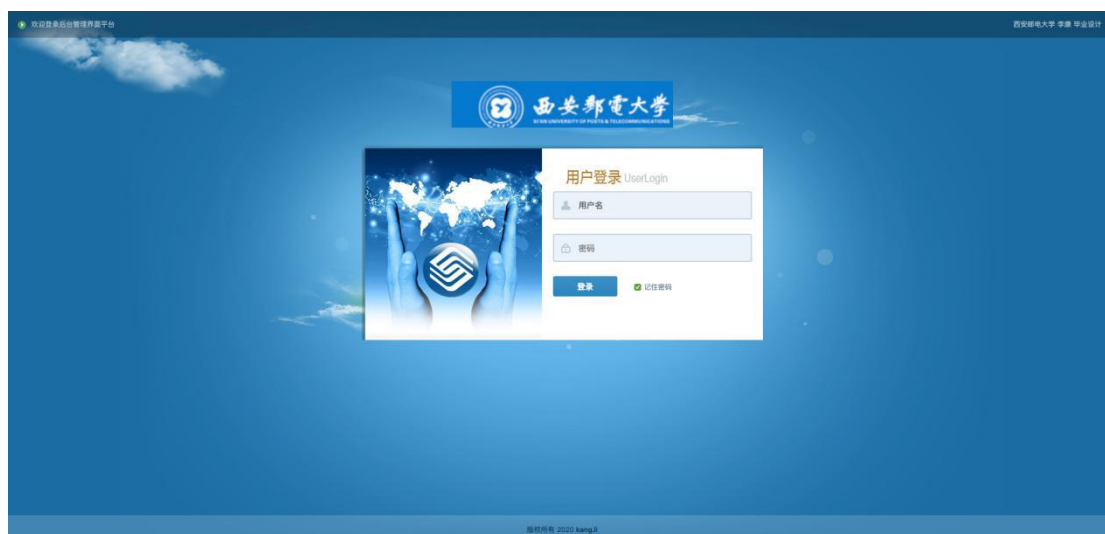


图 5-1 后台管理系统登录页面

核心代码如下：

```
@RequestMapping("loginCheck")
@ResponseBody
public Map<String, Object> loginCheck(String username, String password) {
    Map<String, Object> map = new HashMap<String, Object>();
    Account account = new Account();
    account.setAccountName(username);
    account.setPassword(Md5Tool.getMd5(password));
    Account users = this.accountService.countAccount(account);
    map.put("error", "0");
    if (users == null) {
        map.put("error", "用户或密码不正确! ");
    } else if (users != null && Common.isEmpty(users.getAccountName())) {
        map.put("error", "用户或密码不正确! ");
    }
    return map;
}
```


APP 端登录页面如 5-2 图所示:



图 5-2 APP 端登录界面

核心代码如下:

```
result = appLoginService.login(account);
if (result.get("type").equals(1)) {
    model.addAttribute("userInfo", JSONObject.fromObject(result.get("info")));
    model.addAttribute("managerPhone", result.get("managerPhone"));
    model.addAttribute("managerName", result.get("managerName"));
    Map<String, Object> parameter = new HashMap<>();
    parameter.put("groupId", ((Account) result.get("info")).getGroupId());
    parameter.put("vendorId", 0);
    parameter.put("brandId", 0);
    AppWxMenuShare wxMenuS = appMenuService.getWxMenuByAllInfo(parameter);
    AppWxMenu appWxMenu = null;
    if (wxMenuS != null) {
        List<AppWxMenu> LWxMenu = new ArrayList<AppWxMenu>();
        LWxMenu = new ArrayList<AppWxMenu>();
        parameter.put("ids", wxMenuS.getIndex());
        LWxMenu = appMenuService.getWxMenuByAll(parameter);
        appWxMenu = LWxMenu.size() > 0 ? LWxMenu.get(0) : null;
    }
    if (appWxMenu == null) {
        model.addAttribute("appIndexUrl", "/app/grabSingle.html");
    } else {
        model.addAttribute("appIndexUrl", appWxMenu.getAddress());
    }
    return Common.APP_PATH + "/setAccount";
}
```

5.3.2 后台系统主界面的实现

后台管理系统的主界面，是登录成功之后跳转的页面，此页面会从数据库中获取一级目录基础管理、物流管理，二级目录快递员管理、物资管理、快递公司管理、物流订单管理、门店管理、需求上报审核、订单配送统计等菜单项。后台管理系统的主界面如图 5-3 所示：

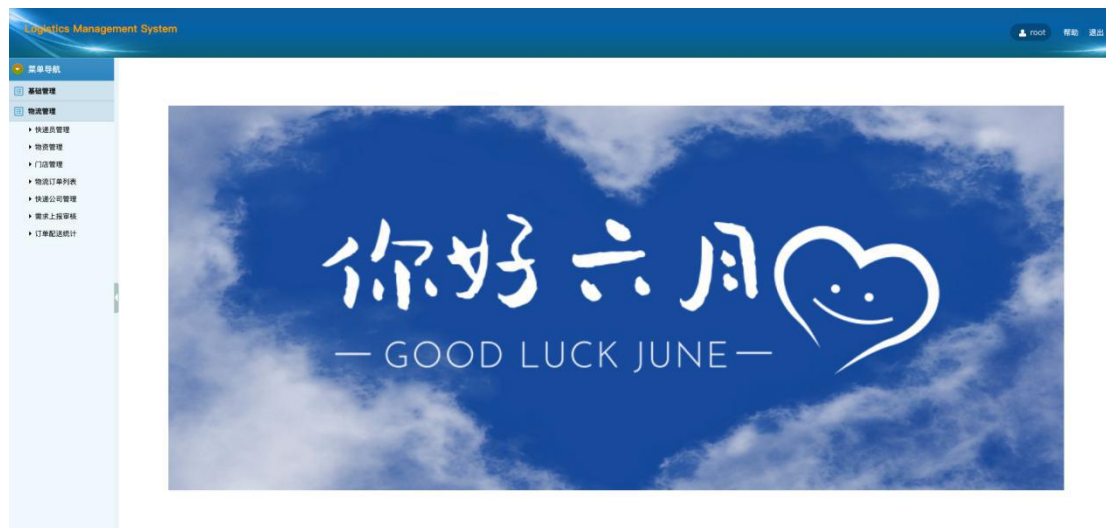


图 5-3 后台管理系统主界面

核心代码如下：

```
rs = resourcesService.queryAll(resources);
List<TreeObject> treeObjects = new ArrayList<TreeObject>();
for (Resources res : rs) {
    TreeObject t = new TreeObject();
    try {
        PropertyUtils.copyProperties(t, res);
    } catch (IllegalAccessException e) {
        e.printStackTrace();
    } catch (InvocationTargetException e) {
        e.printStackTrace();
    } catch (NoSuchMethodException e) {
        e.printStackTrace();
    }
    treeObjects.add(t);
}
List<TreeObject> ns = TreeUtil.getChildResourcess(treeObjects, 0);
model.addAttribute("resourceslists", ns);
return Common.BACKGROUND_PATH + "/homepage/left";
}
```

5.3.3 需求上报的实现

需求上报需要店主登录 APP 端，根据现有物资，选择物资类型，物资数量，支持批量添加不同的物资，然后提供扩展物资选填，点击提交，生成预订单信息入库，若提交时，会检查是否含有添加物资，若无，则提示重新添加物资，提交成功后等待集团中心审核。需求上报页面如图所示：

< 返回

需求上报



物资名称	数量	操作
显示屏	100	删除
手机	10	删除

现有物资

物资名称:

手机

物品数量:

10

↑

添加

扩展物资(选填)

测试页面...

提交

 订单

 政策

 报需

 我的

图 5-3 需求上报页面

核心代码如下图所示:

```
@RequestMapping("appendOrRemove")
public Object appendOrRemove(String materialId, String materialName,
                             String num, String operation, HttpSession
session) {
    List<Map<String, Object>> mList;
    mList = (List<Map<String, Object>>) session.getAttribute("storeMaterial");
    if ("append".equals(operation)) {
        int a = 0;
        if (mList != null && mList.size() > 0) {
            for (int i = 0; i < mList.size(); i++) {
                Map<String, Object> m = mList.get(i);
                if (materialId.equals(m.get("materialId").toString())) {
                    m.put("num", Integer.parseInt(m.get("num").toString())
                        + Integer.parseInt(num));
                    a++;
                }
            }
        }
        if (a == 0) {
            Map<String, Object> map = new HashMap<String, Object>();
            map.put("num", num);
            map.put("materialId", materialId);
            map.put("materialName", materialName);
            mList.add(map);
        }
    } else {
        mList = new ArrayList<Map<String, Object>>();
        Map<String, Object> map = new HashMap<String, Object>();
        map.put("num", num);
        map.put("materialId", materialId);
        map.put("materialName", materialName);
        mList.add(map);
    }
    session.setAttribute("storeMaterial", mList);
} else if ("remove".equals(operation)) { // 删除操作
    for (int i = 0; i < mList.size(); i++) {
        Map<String, Object> map = mList.get(i);
        if (materialId.equals(map.get("materialId").toString())) {
            mList.remove(i);
        }
    }
}
return mList; }
```

5.3.4 需求上报审核

门店上报的需求都会汇总到物流管理后台界面，其每一行会显示具体的需求详情，管理员审核需求时，会考虑实际情况，做出批复，若是审核通过，则生成订单，若是驳回，则说明驳回原因，以便店主知道需求驳回原因是哪方面，审核完成后，会提示是否、审核成功。需求上报审核过程如图 5-4 所示：

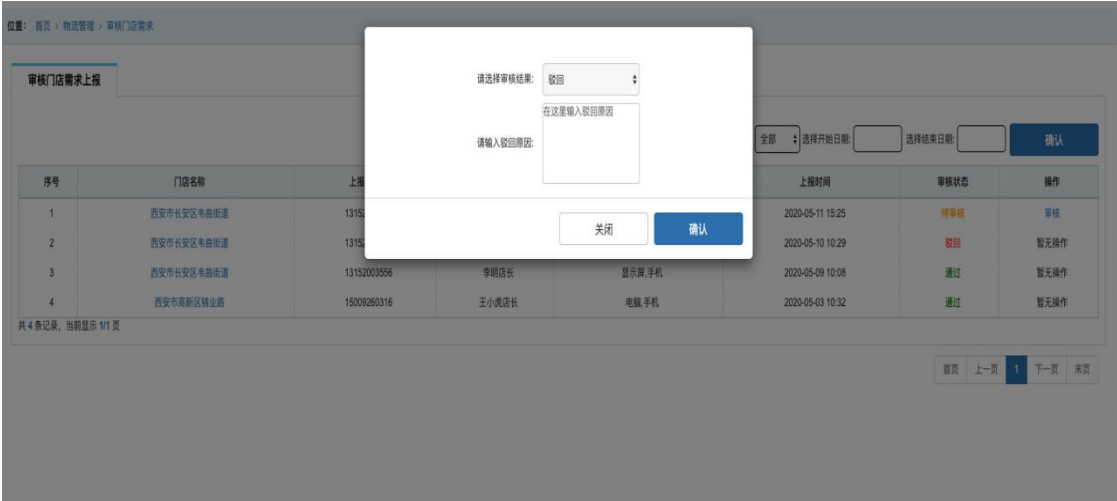


图 5-4 需求上报审核过程

需求上报审核结果列表如图 5-5 所示：

位置： 首页 > 物流管理 > 审核门店需求							
审核门店需求上报							
				选择审核状态：	全部	选择开始日期：	选择结束日期：
序号	门店名称	上报人电话	上报人姓名	物资名称	上报时间	审核状态	操作
1	西安市长安区韦曲街道	13152003556	李明店长	显示屏,手机,电脑	2020-05-11 15:25	通过	暂无操作
2	西安市长安区韦曲街道	13152003556	李明店长	打印机,电脑	2020-05-10 10:29	驳回	暂无操作
3	西安市长安区韦曲街道	13152003556	李明店长	显示屏,手机	2020-05-09 10:08	通过	暂无操作
4	西安市高新区锦业路	15009260316	王小虎店长	电脑,手机	2020-05-03 10:32	通过	暂无操作

共 4 条记录，当前显示 1/1 页

首页 上一页 1 下一页 末页

图 5-5 需求上报审核列表

核心代码如下:

```

        boolean flag = false;
        Account ac = (Account)request.getSession().getAttribute("userSession");
        String userName = ac.getAccountName();
        String realName = ac.getReal_name();
        int groupId = ac.getGroupId();
        String dateNow = new SimpleDateFormat("yyyy-MM-dd").format(new
Date());
        Map<String, Object> parameter = new HashMap<String, Object>();
        parameter.put("checkDatas", checkDatas);
        parameter.put("examine_time",
new SimpleDateFormat("yyyy-MM-dd HH:mm:ss").format(new Date()));
        parameter.put("examine", examine);
        parameter.put("userName", userName);
        parameter.put("realName", realName);
        parameter.put("examine_reason", examine_reason);
        Integer result = backDemandAuditMapper.updateAuditData(parameter);
        if(result > 0 && !"2".equals(examine)){
            List<Map<String, Object>> ls
            backDemandAuditMapper.getPassInfo(parameter);
            for(int i = 0; i < ls.size(); i++){
                if(ls.get(i).get("store_name") == null){
                    ls.remove(i);
                    continue;
                }
                ls.get(i).put("id", new
SimpleDateFormat("yyyyMMddHHmmss").format(new Date()) +
Common.getRandom(3, true, false, false, false, 1).get(0));
                ls.get(i).put("shipperName", realName);
                ls.get(i).put("shipper", userName);
                //ls.get(i).put("shipTime", dateNow);
                ls.get(i).put("groupId", groupId);
                ls.get(i).put("createTime", dateNow);
            }
            if(ls.size() > 0){
                if (backDemandAuditMapper.addOrderInfo(ls) > 0) {
                    flag = true;
                } else {
                    flag = false;
                }
            } else {
                flag = false;
            }
        }
    }
}

```

```
if("2".equals(examine)){
    List<Map<String, Object>> storeUserPhones =
        backDemandAuditMapper.getById(checkDatas);
    for (Map<String, Object> map : storeUserPhones) {
        SmsSend send = new SmsSend();
        String msg =
            "您需求上报的信息已被审核,请您尽快前往系统查看结果";
        send.sendSms(
            map.get("store_shopowner_phone").toString(), msg, null);
    }
    flag = true;
}
return flag;
```

5.3.5 订单配送流程图

审核后的订单，在物流订单列表中-待完善信息列表中显示，此时订单处于待发货状态，当管理员批量选择发货时，必须为此订单选择快递公司，快递员，才可发货，此时订单处于已发货状态，当快递员接受订单开始派送时，获取快递员坐标，计算派送订单距离，时间，此时订单处于派送中，当快递员到达门店时，可以点击按钮操作，向门店店主发送验证码，然后想店主索要店主已收到的验证码，输入订单校验，此时该订单已经派送成功。

订单配送待完善信息列表如图 5-6 所示：

位置: 首页 > 物流管理 > 物流订单列表									
订单列表		待完善信息列表						批量发货	批量删除
<input type="checkbox"/>	订单编号	发货人姓名	发货时间	快递员姓名	物资名称	收货人姓名	收货具体地址	收货时间	状态
<input checked="" type="checkbox"/>	20200511152818074	root	暂无记录		显示器,手机,电脑	李明店长	西安市长安区韦曲街道	暂无记录	待发货
共 1 条记录, 当前显示 1/1 页									
首页 上一页 1 下一页 末页									

图 5-6 订单配送待完善信息列表

订单配送待完善信息选择配送员如图 5-7 所示：

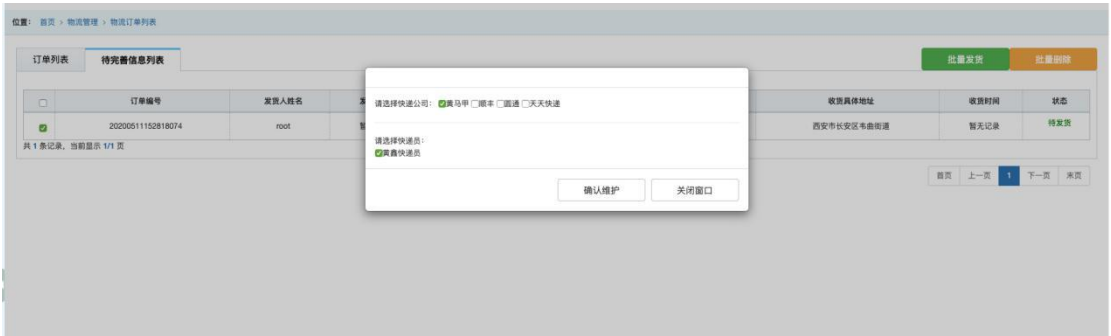


图 5-7 订单配送待完善信息选择配送员

快递员派单成功如图 5-8 所示：

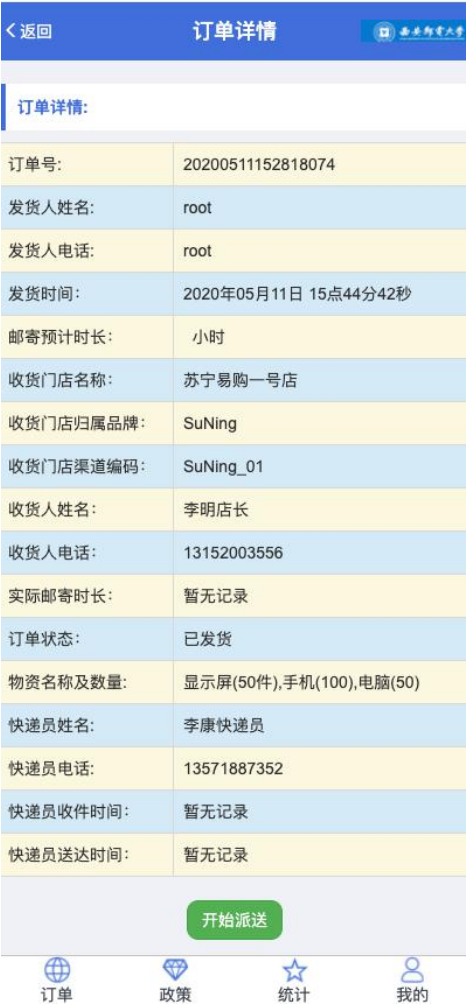


图 5-8 快递员配单成功

快递员开发派送如图 5-9 所示:

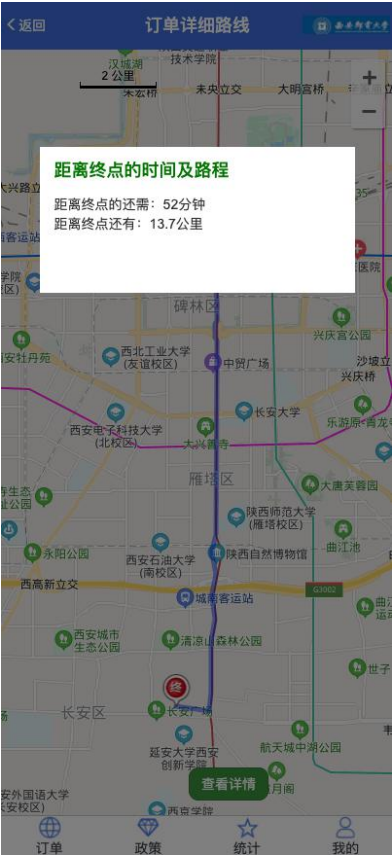


图 5-9 快递员开始派送

快递员确认收货如图 5-10 所示:



图 5-10 快递员确认收货图

快递员配送成功如图 5-11 所示:



图 5-11 快递员配送成功列表图

核心代码如下:

```
@RequestMapping(value = "displayMap")
public String displayMap(String id, String courier_Phone, Model model,
    HttpServletRequest request) {
    Map<String, Object> m = new HashMap<>();
    m.put("id", id);
    m.put("courier_Phone", courier_Phone);
    // 获取快递员当前地坐标
    Map<String, Object> maplist = logisticsOrderService.getDisplayMap(m);
    // 获取订单门店坐标
    Map<String, Object> maplist1 = logisticsOrderService.getDisplayMap1(m);

    model.addAttribute("maplist", maplist);
    model.addAttribute("maplist1", maplist1);
    return Common.BACKGROUND_PATH + "/order/displayMap";
}
```

```

@Override
public Map<String, Object> sendOrder(String orderName, String[] selectCourier) {
    Map<String, Object> m = new HashMap<>();
    m.put("orderInfo", "order Exception!");
    String[] orderId = orderName.split(",");
    if (orderId.length > 0) {
        //遍历订单
        for (int i = 0; i < orderId.length; i++) {
            String phone = selectCourier[0];
            //创建 map
            Map<String, Object> map = new HashMap<>();
            //设置订单 id
            map.put("id", orderId[i]);
            //为订单设置快递员电话
            map.put("courierPhone", phone);
            //添加发货时间
            map.put("shipTime",
                new SimpleDateFormat("yyyy-MM-dd HH:mm:ss").format(new Date()));
            //为订单设置快递员姓名
            map.put("courierName",
                logisticsOrderMapper.getCourierName(phone));
            //将匹配完成的订单进行配送操作
            logisticsOrderMapper.finishOrder(map);
            m.put("orderInfo", "delivery successful!");
        }
    } else {
        m.put("orderInfo", "Not find Order info! ");
    }
    return m;
}

@RequestMapping("addOrder")
public String addOrder(@RequestParam(value = "file", required = false) MultipartFile
    file, HttpServletRequest request, RedirectAttributes attributes) {

    // 订单信息维护
    attributes.addAttribute("info", logisticsOrderService.addOrder(
        file, request).get("addInfo"));

    return "redirect:list.html";
}

```

```

public Object appendOrRemove(String num, String material, String materialName,
String operation, HttpSession session) {
    List<Map<String, Object>> mList;
    mList = (List<Map<String, Object>>) session.getAttribute("myMaterial");
    if ("append".equals(operation)) {
        int a = 0;
        if (mList != null && mList.size() > 0)
            for (int i = 0; i < mList.size(); i++) {
                Map<String, Object> m = mList.get(i);
                if (material.equals(m.get("material").toString())) {
                    m.put("num",
                        Integer.parseInt(m.get("num").toString()) + Integer.parseInt(num));
                    a++;
                }
            }
        if (a == 0) {
            Map<String, Object> map =
                new HashMap<String, Object>();
            map.put("num", num);
            map.put("material", material);
            map.put("materialName", materialName);
            mList.add(map);
        }
    } else {
        mList = new ArrayList<Map<String, Object>>();
        Map<String, Object> map = new HashMap<String, Object>();
        map.put("num", num);
        map.put("material", material);
        map.put("materialName", materialName);
        mList.add(map);
    }
    session.setAttribute("myMaterial", mList);
} else if ("remove".equals(operation)) {
    for (int i = 0; i < mList.size(); i++) {
        Map<String, Object> map = mList.get(i);
        if (material.equals(map.get("material").toString())){
            mList.remove(i);
        }
    }
}
return mList;
}

```

5.4 本章小结

本章主要对物流管理信息系统实现过程中，开发环境的搭建、项目框架的配置以及核心模块的实现做了简单的介绍和说明，确保系统各个子模块可以正常、稳定、高效的运行。

结 论

本文对物流管理信息系统从深度、广度上进行了研究与探讨，并对该系统所具有的功能进行详细的分析、设计与实现，最终经过长达半年的时间彻底的完成物流信息管理系统，本文的主要工作如下：

1. 通过前期对国内外市场的调研、用户提出的需求进行了详细的分析，包括系统可行性分析、系统功能需求分析、系统架构性能分析等方面，我们通过流程图、类图等工具设计出一套符合市场、用户需求的物流管理系统，此系统无论从用户的角度，还是系统的性能都完全符合我们最初调研和需求。
2. 使用 Spring + SpringMVC + MyBatis 搭建项目开发架构、使用 MySQL 作为数据库，存储产生的数据、使用 Tomcat 做部署服务器实现了物流管理信息系统。
3. 最终通过论文的形式，输出物流管理系统一些模块的划分、分析、设计等文档，并对核心模块的是吸纳进行了详细而简单的说明。

在为期半年的毕设开发中，作为设计者和实现者，我发现目前系统可能存在的问题如下，后期我们将对系统进行优化：

1. 当快递员送达货物时，可能存在店主不在的情况，发送的验证码无法获取，因此需要提供 PLAN B，以防止此情况的发生。
2. 配送完成后，我们并没有做让门店对整个配送服务流程进行评价，不利于公司整体配送流程效率的提升，后期考虑增加此模块。

致 谢

光阴似箭，日月如梭，不知不觉中半年的课题研究、项目开发、论文撰写工作已经到了最后阶段。由于本人最初没有参与类似于管理系统的项目开发，因此在最初遇到了许多问题，是指导老师莎柯雪老师的细心指导与同学们的帮助下才使我解决了所有难题，在规定的时间内，顺利的完成毕设工作。

首先我要感谢我的指导老师莎柯雪老师。莎老师严谨的教学态度、精益求精的工作作风，渊博的教学知识让我产生了深刻的影响。如果没有莎老师一直在毕设期间对我的细心指导，我们论文工作难度是难以想象的。在此，我要向莎老师致以最诚挚的感谢。

其次，感谢我的母校——西安邮电大学，是您提供了我们四年来成长的机会，让我成长为一名有能力、有信心、有礼貌的工程师；还要感谢我的同学们，当我每次遇到苦难时，是你们一次次的讨论，拿出解决方案，让我的毕设工作顺利进行下去，谢谢你们。

最后感谢培养我成长至今的父母，你们的默默支持，得以让我专心致志的开展毕设工作，谢谢你们！

参考文献

- [1] Craig Walls.《Spring 实战》[M]. 北京: 人民邮电出版社, 2016: 35-61.
- [2] 姜承尧.《MySQL 技术内幕》[M].北京: 机械工业出版社, 2019: 187-196.
- [3] 孙鑫.Servlet/JSP 深入详解——基于 Tomcat 的 Web 开发[M].北京: 电子工业出版社, 2008: 200-252.
- [4] 戴君; 谢俐; 王强.第三方物流整合对物流服务质量、伙伴关系及企业运营绩效的影响研究[J].管理评论, 2014.05.
- [5] 王成金; 张梦天.中国物流企业的布局特征与形成机制[J].地理科学进展, 2014.01.
- [6] 李国旗; 陈娱.基于物流热度的中国物流业空间格局[J].地理科学进展, 2015.05.
- [7] 刘有升.基于复合系统协同度模型的跨境电商与现代物流协同评价分析[J].中国流通经济, 2016.05.
- [8] 刘培银.我国民营物流企业的扩张策略研究--以顺丰物流公司为例[P].苏州大学.
- [9] Jeff Langer.Agile Java [M].Beijing:Publish House Of Electronics Industey,2006:124-168.
- [10] JSP 语法.<https://www.runoob.com/jsp/jsp-syntax.html>.
- [11] Lifan Yang, Proceedings of 2012 International Conference in Humanities,Social Sciences and Global Business Management(ISSGBM 2012 V7) [J].2012-12-30.
- [12] Ling Zhang,Wei Yuan,Songyan Jiang,Huijun Wu,Tianming Chen,Xin Liu. Modelling the generation of household automobiles scrap in the context of urban-rural disparity: A case study of Nanjing, China[J]. Journal of Cleaner Production,2020,268.