

**本 科 生 毕 业 设 计（论文）**

**（2018届）**

**信息工程学院**

**题 目： 产品流通的可视化表达与数据挖掘**

**学 号： 201505010315**

**姓 名： 倪畅**

**专业班级： 计算机科学与技术153班**

**指导教师： 陈文辉 职称： 讲师**

**2019年03月20日**



**本科生毕业设计（论文）诚信承诺书**

我谨在此承诺：本人所写的毕业设计（论文）《产品流通的可视化表达与数据挖掘》均系本人独立完成，没有抄袭行为，凡涉及其他作者的观点和材料，均作了引用注释，如出现抄袭及侵犯他人知识产权的情况，后果由本人承担。

**承诺人（签名）：**

**2019年03月20日**

产品流通的可视化表达与数据挖掘

信息工程学院 计算机科学与技术153班 倪畅 指导教师：陈文辉

**摘要：**二十一世纪后，互联网行业的高速发展，正在慢慢改变人们的日常生活，衣食住行无不和互联网行业有着千丝万缕的联系。越来越多的互联网技术被运用在人们的日常生活中，诸如：人工智能、云计算、大数据、机器学习等等。为了加快提升人们的生活质量，我国正在大力推动智慧城市建设。旨在通过物联网和新一代互联网技术，结合新一代的创新思想，对社会的民生、治安、医疗、公共服务等作出智能的应答。基于目前已经相对成熟的智慧政务、智慧交通、智慧医疗和智慧旅游等项目的建设，本文主要研究以农产品为主要研究对象的产品流通和可视化及数据挖掘，从而搭建一个应用平台来便利民生。

通过市场的调研，了解了目前农产品的需求度和各个菜市场对农产品规范化的空缺，制定了合适的需求模型和设计方案：采用B/S架构；基于Spring、SpringMVC、Mybatis框架并利用Java8、Druid、Lombok、Logback、WebSocket、Maven等技术搭建服务端；基于React框架并利用Javascript、bootstrap、axios、Ant Design、 Echarts、npm等技术搭建用户端；基于ECharts进行大数据的可视化模块设计；基于二次指数平滑算法进行数据预测的实现；利用Tomcat和MySQL进行用户端和服务端的整合。

**关键词：**大数据；数据挖掘；预测分析；智慧城市；可视化平台

Visual representation and data mining of product circulation

**Abstract：**After the 21st century, the rapid development of the Internet industry is slowly changing people's daily lives, and the Internet, the industry, and the Internet industry are inextricably linked. More and more Internet technologies are being used in people's daily lives, such as artificial intelligence, cloud computing, big data, machine learning, and so on. In order to speed up the improvement of people's quality of life, China is vigorously promoting the construction of smart cities. It aims to respond intelligently to the people's livelihood, public security, medical care, and public services through the Internet of Things and next-generation Internet technologies, combined with a new generation of innovative ideas. Based on the current mature projects of smart government, smart transportation, smart medical care and smart tourism, this paper mainly studies product circulation and visualization and data mining with agricultural products as the main research object, thus building an application platform to facilitate people's livelihood.

Through market research, we have learned about the current demand for agricultural products and the vacancies in the standardization of agricultural products in various vegetable markets. We have developed a suitable demand model and design scheme: using B/S architecture; based on Spring, SpringMVC, Mybatis framework and using Java8, Druid , Lombok, Logback, WebSocket, Maven and other technologies to build the server; based on the React framework and use Javascript, bootstrap, axios, Ant Design, Echarts, npm and other technologies to build the user terminal; based on ECharts for large data visualization module design; based on the second Exponential smoothing algorithm for data prediction; using Tomcat and MySQL for client-side and server-side integration.

**KeyWords：**Big Data; Data mining; Predictive analysis; Smart City; Visualization platform

**目 录**

[1 绪论 3](#_Toc4089994)

[1.1 课题的背景 3](#_Toc4089995)

[1.2 现状的分析 3](#_Toc4089996)

[1.3 产品流通的可视化表达与数据挖掘应用平台的优势 3](#_Toc4089997)

[1.4 本论文主要结构 4](#_Toc4089998)

[2 开发环境与相关技术 4](#_Toc4089999)

[2.1 系统开发环境 4](#_Toc4090000)

[2.2 React 4](#_Toc4090001)

[2.3 ECharts 5](#_Toc4090002)

[2.4 npm 5](#_Toc4090003)

[2.5 axios 5](#_Toc4090004)

[2.6 SSM框架 5](#_Toc4090005)

[2.6.1 Spring 6](#_Toc4090006)

[2.6.2 SpringMVC 6](#_Toc4090007)

[2.6.1 Mybatis 8](#_Toc4090008)

[2.7 Java8 8](#_Toc4090009)

[2.8 Druid 9](#_Toc4090010)

[2.9 Lombok 9](#_Toc4090011)

[2.10 Logback 9](#_Toc4090012)

[2.11 Maven 9](#_Toc4090013)

[2.12 Websocket 10](#_Toc4090014)

[3 系统设计分析 10](#_Toc4090015)

[3.1 系统设计思想 10](#_Toc4090016)

[3.2 可行性分析 11](#_Toc4090017)

[3.2.1 技术可行性 11](#_Toc4090018)

[3.2.2 经济可行性 11](#_Toc4090019)

[3.2.3 操作可行性 11](#_Toc4090020)

[3.2.4 组织可行性 11](#_Toc4090021)

[3.3 业务流程 11](#_Toc4090022)

[3.3.1 系统流程图 11](#_Toc4090023)

[3.3.2 数据流程图 12](#_Toc4090024)

[3.4 用例分析 13](#_Toc4090025)

[4 系统设计 14](#_Toc4090026)

[4.1 非功能性需求 14](#_Toc4090027)

[4.2 功能性需求 14](#_Toc4090028)

[4.2.1 货运管理 14](#_Toc4090029)

[4.2.2 统计分析 15](#_Toc4090030)

[4.2.3 基础信息管理 15](#_Toc4090031)

[4.2.4 系统管理 15](#_Toc4090032)

[4.2.5 系统首页 16](#_Toc4090033)

[4.3 数据库概念结构设计 16](#_Toc4090034)

[4.4 数据库的逻辑结构设计 18](#_Toc4090035)

[5 系统实现 20](#_Toc4090036)

[5.1 系统流程图 20](#_Toc4090037)

[5.2 业务代码执行流程 20](#_Toc4090038)

[5.3 项目文件结构 21](#_Toc4090039)

[5.4 编号命名规范 22](#_Toc4090040)

[5.5 用户登录 22](#_Toc4090041)

[5.6 用户主界面 23](#_Toc4090042)

[5.6.1 留言板 23](#_Toc4090043)

[5.6.2 个人信息修改 24](#_Toc4090044)

[5.6.3 标题部分 24](#_Toc4090045)

[5.7 货运管理 24](#_Toc4090046)

[5.7.1 购销合同 25](#_Toc4090047)

[5.7.2 出货表 27](#_Toc4090048)

[5.7.3 出口报运 27](#_Toc4090049)

[5.7.4 装箱管理 28](#_Toc4090050)

[5.7.5 委托书 30](#_Toc4090051)

[5.8 统计分析 31](#_Toc4090052)

[5.8.1 生产厂家销售情况 32](#_Toc4090053)

[5.8.2 产品销售排行 32](#_Toc4090054)

[5.8.3 系统访问压力图 32](#_Toc4090055)

[5.8.4 财务统计 33](#_Toc4090056)

[5.9 基础信息 33](#_Toc4090057)

[5.9.1 数据字典 33](#_Toc4090058)

[5.9.2 厂家信息 33](#_Toc4090059)

[6 系统管理 35](#_Toc4090060)

[6.1 角色管理 35](#_Toc4090061)

[6.2 部门管理 37](#_Toc4090062)

[6.3 用户管理 37](#_Toc4090063)

[7 总结 39](#_Toc4090064)

[参考文献 41](#_Toc4090065)

[致谢 42](#_Toc4090066)

**1** **绪论**

**1.1 课题的背景**

“买菜”这一动作行为可以说是生活中我们每个人都会遇见的，每一个人都将会长大，承担家庭的责任。那么，为自己的家人做一顿可口的饭菜就免不了买菜。但是，随着生活水平的提高，反而越来越多的人对于买菜没有任何经验。他们吃惯了美味的外卖，就连水果，也是帮你切好，外卖到家，完全不知道现在蔬菜水果的价格行情。这就让有些卖菜商人们有机可乘：一味地抬高菜价，坐地起价，以次充好。这极大影响了广大市民们的生活水平和质量。

其次，一方面，生活在一线城市的人们的时间都是很宝贵的，他们工作压力大，为节省买菜时间，他们从不议价；另一方面，很多刚刚进入社会的小年轻们，从小享受着父母的宠爱，他们也不会议价。这就造成了目前市场鱼龙混杂，农产品价格不公开化透明化的问题。

于是，本文所描述的农产品的流通可视化应用平台也就应运而生。它不仅可以提供给市民们公正透明公开的菜场价格，还能让国家清楚的知道我国人民对各类农产品的需求，便于国家大局的规划；同时，他也能让菜农们了解现在市场最需要什么，让他们耕耘有方，不再滞销浪费。

**1.2 现状的分析**

“物价高”已经是我国的十大社会问题之一。价格上涨与食品安全和房价密切相关。令人惊讶的是物价高当然不仅仅是价格的上涨。当下媒体广泛报道的“卖菜难、买菜贵”也是价格上涨的问题。政府对房价调整的控制力不高，但对一般农副产品的监管力度相当强大，但由于它太强大，到最后，生活在底层的老百姓们就成了最大的受害者。一方面，农民烦恼“卖菜难”，另一方面，城市普通居民感到“买菜贵”。在价格上涨之后，买卖的不和谐声音必然是不可避免的问题。那么，我们为什么不去打造一个平台，让市场中的菜价透明化公正化规范化，每一种农副产品的价格都能追溯它的源头，真正的解决好买卖农副产品双方的主要矛盾，有利于提高市场自我管理的能力。同时，该平台既解决了菜农们农副产品的滞销，也有效解决了买菜市民对菜品价格和质量的抱怨。

其次，国家目前对智慧城市的建设如火如荼，其中在公共交通、公共资源、医疗、政务、旅游等行业已经逐渐成熟，但是在农产品这方面还未见其雏形，所以，这一行业的市场空缺很大，值得去研究开发一个智慧农产品平台——产品流通可视化和数据挖掘应用平台。

**1.3 产品流通的可视化表达与数据挖掘应用平台的优势**

柴米油盐酱醋茶，这些都与菜市场有着密不可分的关系。作为城市公共配套设施，菜市场关系着千家万户的菜篮子，是民生所系、政府之责、公共服务必需。买菜方不方便，菜价便不便宜，菜品安不安全，与每个人的日常生活息息相关，直接影响着人们的幸福指数。对于本平台的优势主要体现在以下五个方面：

（1）实时价格显示：减少了菜农和买菜市民之间还价时间，便于规范市场价格规范，避免了市民买菜贵的现象；

（2）单一农副产品价格查询：用户买菜前可以先在平台上了解菜品的历史价格，防止出现农副产品价格溢价的情况；

（3）所需农副产品未来七天价格预测：帮助买菜市民们了解未来的价格趋势，同时也能让菜农们知道未来该种植什么作物能带来更多的收益；

（4）推动我国的智慧城市建设发展，为我国智慧城市建设带来多元化发展；

（5）适当解决我国的某些社会民生问题，促进我国的社会经济的发展。

**1.4 本论文主要结构**

**2 开发环境与相关技术**

**2.1 系统开发环境**

系统开发环境详情见表2-1所示：

表2-1 系统开发环境表

|  |  |
| --- | --- |
| 类别 | 内容 |
| 系统 | Windows10  Windows7 |
| 编译器 | IntelliJ IDEA 2018.3.4  JetBrains WebStorm 2018.2.5 |
| 服务器 | Tomcat 7.0.92 |
| 数据库 | MySQL5.7 |
| 项目构建 | Maven  Npm |
| 框架 | Spring+SpringMVC+ MyBatis  React |
| 测试工具 | Postman |
| 浏览器 | Chrome |
| 设计语言 | Java8  JavaScript |
| 数据库建模 | PowerDesigner15.1  Navicat Premium 12 |

**2.2 React**

React的速度很快，与其他框架相比，React框架采取了一种特立独行的操作DOM的方式。它并不直接对DOM进行操作，它引入了一个“虚拟DOM”的概念，并把它安插在JavaScript逻辑和实现DOM之间；这一概念有效地提高了Web的性能。在渲染的过程中，React通过在虚拟DOM中的微操作来实现对实际DOM的局部更新。其次，这一虚拟DOM的概念还帮我们解决了跨浏览器的兼容问题，他为我们提供了标准的API接口。

React能进行模块化管理，使用React可以对要实现的每一个功能进行模块化隔离，模块之间互不影响，在开发过程中进行Bug修复、单元测试都十分简单高效，有效地提高了代码的可维护性。

**2.3 ECharts**

ECharts这项技术对于目前的大数据可视化可以说是十分地重要，它支持多种图表的格式显示，具有丰富的可视化图标类型，包括：折线图、柱状图、饼图、地图、热点图、散点图等等。用户都能通过配置图表的属性来定制自己所需要的报表。而且ECharts的多种数据格式无需转换可以直接使用，通过后台的JSON数据返回，既能做到定时刷新，也能做到实时更新数据，这一特性对于本文所描述的“产品流通可视化和数据挖掘应用平台”具有很好的可利用性：进行千万数据的前端渲染展现，对开发这个大数据应用平台有着很大的帮助，极大地简化了大数据可视化的开发步骤，缩短了开发所需的时间。

**2.4 npm**

npm的全称叫做Node Package Manager，它是一款Node的包管理工具，目前Node.js已经集成了npm。npm对我们开发者来说最大的优点就是方便我们引入各种所需要的包（npm install XXX），npm还提供了一个重用其它开发者代码的方式，npm也能够分享开发者自己的代码，同事也提供了一个便捷的方式来来管理不同版本的代码。

**2.5 axios**

axios就是基于promise用于浏览器和node.js的http客户端。它的主要特点如下：

（1）从浏览器生成xmlhttpRequests

（2）从node.js发出HTTP请求

（3）支持Promise API

（4）拦截请求和响应

（5）转换请求和响应数据

（6）取消请求

（7）JSON数据的自动转换

（8）客户端支持防止XSRF

之所以我选择使用axios而不使用ajax请求，是因为本文所描述的应用平台在系统设计上就采用前后端分离技术，没有使用传统的MVC编程，ajax不符合现在前端MVVM的潮流，而且JQuery的整个项目太大，单纯使用ajax就要把整个JQuery全部引入，这显得非常不合理。而axios就比较小巧灵活，使用npm install axios就能安装使用，十分方便。

**2.6 SSM框架**

SSM框架的全称是Spring、SpringMVC和Mybatis。SSM框架是继SSH（Struts、Spring和Hibernate）之后目前市面上比较主流的Java EE企业级开发框架，适用于搭建各种中大型的企业级应用系统。当然，目前市面上还有一种框架——Spring Boot，但是Spring Boot作为一个微服务框架，距离微服务的实现还是存在一些距离的。这个框架只是为了提高开发效率，为了提高开发人员的生产力，对于Restful格式的URL请求还需要开发人员自行结合实际进行规范化工作。就框架的配套插件来说，SSM框架还是远远强于Spring Boot框架的，所以，本课题所研究开发的大数据应用平台还是决定使用SSM框架。

2.6.1 Spring

Spring的优点有很多，它可以很好地解决企业级大型项目的复杂性。然而，Spring的用途不仅限于服务器端的开发。从简单性、可测试性和松耦合的角度而言，任何Java应用都可以从Spring中受益。 简单来说，Spring是一个轻量级的控制反转（IoC）和面向切面（AOP）的容器框架。下面我就简单介绍几个Spring的特点：

（1）控制反转：

控制反转简称IOC，也叫依赖注入。在工厂模式下，对象由容器管理。您只需要在Spring配置文件中配置相应的bean，并设置相关的属性，这样Spring容器就可以生成类的实例对象和管理对象。当Spring容器启动时，Spring初始化配置文件中配置的所有bean，然后将那些已经初始化的bean分配给需要调用的类（假定类名为a），当需要调用它们时。分配方法是调用的setter方法来注入它们，而不是需要您在中更新这些bean。

（2）面向切面：

面向切面简称AOP，是Spring中另一个特点。首先，需要说明的是，AOP只是Spring的特征。与OOP一样，AOP是一种编程思想，而不是一种技术。AOP可以说是对OOP的补充和改进。OOP引入了封装、继承和多态的概念，建立了一个对象层次结构来模拟一组公共行为。当我们需要将公共行为引入分散的对象时，OOP似乎是无能为力的。也就是说，OOP允许您定义自上而下的关系，但它不适合定义从左到右的关系。例如，日志函数。日志代码倾向于水平地分布在所有对象级别，而不管其分布到的对象的核心功能是什么。在OOP设计中，会导致大量的代码重复，这不利于每个模块的重用。程序中的跨业务逻辑（如安全性、日志、事务等）被封装到一个方面，然后注入目标对象（特定的业务逻辑）。

2.6.2 SpringMVC

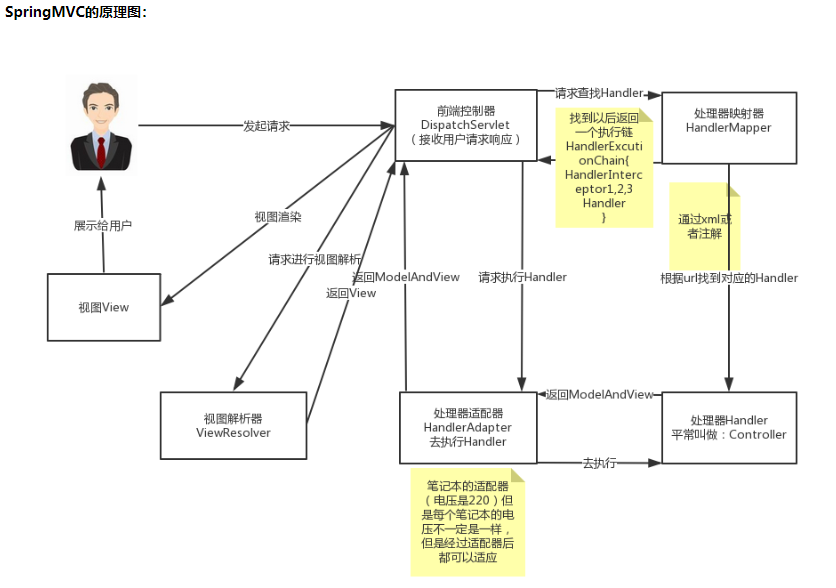
SpringMVC框架是一个MVC框架，它通过实现模型-视图-控制器模式来分离数据、业务和表示。从这个角度来看，SpringMVC与Struts和Struts2非常相似。SpringMVC是围绕DispatcherServlet设计的，它负责将请求分派给特定的处理程序。请求通过可配置的处理程序映射（handler mappings）、视图解析（view resolution）、区域设置（locale）和主题解析（theme resolution）进行处理，并传输到相应的视图。

前端控制器（DisatcherServlet）:接收请求，响应结果，返回可以是json,String等数据类型，也可以是页面（Model）。

处理器映射器（HandlerMapping）:根据URL去查找处理器，一般通过xml配置或者注解进行查找。

处理器（Handler）：就是我们常说的controller控制器啦，由程序员编写。

处理器适配器（HandlerAdapter）:可以将处理器包装成适配器，这样就可以支持多种类型的处理器。

视图解析器（ViewResovler）:进行视图解析，返回view对象（常见的有JSP,FreeMark等）。

下面我就简单的描述一下SpringMVC的主要流程：

（1）用户发送请求到前端控制器（DispatcherServlet）。

（2）前端控制器请求处理器映射器（HandlerMapping）去查找处理器（Handler）。

（3）找到以后处理器映射器（HandlerMappering）向前端控制器返回执行链（HandlerExecutionChain）。

（4）前端控制器（DispatcherServlet）调用处理器适配器（HandlerAdapter）去执行处理器（Handler）。

（5）处理器适配器去执行Handler。

（6）处理器执行完给处理器适配器返回ModelAndView。

（7）处理器适配器向前端控制器返

回ModelAndView。

（8）前端控制器请求视图解析器（ViewResolver）去进行视图解析。

（9）视图解析器向前端控制器返回View。

（10）前端控制器对视图进行渲染。

（11）前端控制器向用户响应结果。

2.6.1 Mybatis

Mybatis相对于传统的数据库连接方式JDBC来说，具有很大的优越性和先进性。首先，Mybatis可以做到对SQL语句的统一管理，有专门的xml文件来配置数据库和SQL语句，根据查询所需要的参数进行动态的拼接SQL语句；而我们使用JDBC对数据库进行操作时，SQL查询语句分布在若干个Java类中，这样可读性和可维护性都比较差，代码不够规范，缺少模块化。其次，Mybatis可以对结果进行映射，你可以返回任意类型的数据，这个类型包括但不限于java类、list、map、tree等等。但是JDBC就表现的很不起眼，它仅仅返回一个ResultSet结果集，然后开发人员还要从这个ResultSet结果集中取出我们想要的结果并封装成所需要的类型。

**2.7 Java8**

对于本研究的课题，本人考虑了后期数据量会数百上千万的情况，为了让用户能有更好的软件使用体验和更优的软件流畅及稳定性。再加上目前市场Java8的环境逐渐成熟，出于综合考虑，本人决定使用Java8作为开发语言；下面，本人将阐述Java8的新特性在系统中的使用。

（1）Stream流

Java8中的流式处理在本应用系统平台中已经无处不在了，Java8的流式处理代替了此前版本的for循环，用简洁的语法，就能对集合进行操作。身为一个程序开发者，我们都知道编码中对集合进行操作处理是非常麻烦的：传统操作不仅需要大量的编码，操作效率还十分低下。但在Java8中我们可以通过流，它就像是一个SQL语句，你只要告诉它你需要什么，他就会返回给你什么结果，这极大提高了开发效率和开发质量。

（2）Lambda表达式

身为一个开发者，对于匿名内部类大家可能都不陌生，编码的简洁让越来越多的人喜欢上了它。千万不要以为这已经是最简单的，Java8就对匿名内部类进行了优化升级，甚至可以说是一次翻天地府的改版。无中生有，从无到有可以说是对Java8中的Lambda表达式的最简概括。

Lambda表达式格式：

(parameters) -> expression或(parameters) ->{statements;}

不管你写的方法需不需要参数或者这个方法需不需要返回值，用Lambda表达式编写你都能很好的解决，并且带来很好的可读性和可维护性，维护人员也不用再抓着头皮反复阅读他人编写的冗长代码。

（3）函数式编程

函数编程的最直接的表现，莫过于将函数作为数据自由传递，结合泛型推导能力，使代码表达能力获得飞一般的提升。Java8的函数式编程主要体现在以下三个方面：函数接口(Function)、流(Stream)和聚合器(Collector)。这几个方面在本文中我均已阐述，当然如果大家对这方面感兴趣，可以阅读我在文末的参考文献Java8实战一书，相信你会因此爱上Java8。

（4）Java8中的线程池

从Java线程创建、销毁和减去线程对于计算机资源来说是一项相对昂贵的任务。如果我们需要使用多线程来处理任务，频繁地创建和销毁线程，就会造成计算机资源的浪费，从而出现线程池技术。Java8中的ThreadPoolExecutor方法完美地提供了解决Java中多线程中的方法。本系统就是使用了这种方法，在该部分功能的具体实现中我会详细介绍，在此不再敖述。

**2.8 Druid**

数据库的连接建立和关闭是十分消耗服务器集群的内存和资源的，尤其是在大型项目中，数据库带来的资源浪费和消耗尤为明显。合理的控制数据库请求次数和数据库连接池的最大连接数对于开发者来说简直苦不堪言。这时候，使用阿里巴巴开源的Druid连接池就不存在这种问题，所有的设置只需要建立一个javax.sql.DataSource的实例就完全搞定。相比原来的JDBC，使用Druid不仅可以减少大量开发时间来配置连接池的相关设置，还能保证数据库的安全性，阿里巴巴Druid连接池仅支持通过特殊算法加密后的加密密码进行连接，对于软件安全的提升有很大帮助。

**2.9 Lombok**

Lombok可以说是另一个给程序员带来的福音，只要在IDE和项目中配置好Lombok，所有Javabean不用再编写构造函数方法、GetterAndSetter方法，也不用重写toString和hashcode的方法，节省开发时间用在业务逻辑层上，具体用法就是在Javabean中加入注解：@Setter、@Getter、@Log4j、@NoArgsConstructor和@AllArgsConstructor等，在项目编译的时候，自动在.class文件进行自动生成上述方法，极大简化了Javabean的代码量。但是要注意：使用此方法一定要保证Javabean中的所有变量全都符合CamelCase命名法，否则在编译时会发生错乱。

**2.10 Logback**

每一个开发工程师都希望自己设计的系统在无论多么恶劣的环境下，它的日志记录系统都能正常运行，以便于后期维护升级或者遇到Bug的时候能够有迹可循。Logback就可以很好的胜任这个工作，它相比较log4j有着更快的执行速度，还能对想要的日志进行过滤操作，节省了很大一部分的物理内存。本文所描述的应用平台详细地记录了每一次工作时，该平台出现的的所有异常和所有数据插入记录；根据日志，可以很快的知道哪一天因为应用平台的某种异常情况我们没有爬取新的数据，以便后期进行数据补充处理。

**2.11 Maven**

身为一个后端开发者，我已经不止一次地遇到这种情况：因为某些引入包的版本冲突所导致项目不能按照我们预期的结果进行运行。在没有使用Maven之前，我经常犯这样的错，包与包之间版本不一致导致报错或者同一类型的包被重复引用不同的版本导师编译失败。使用了Maven之后，这种问题完全不存在，完美适配目前这个提倡自动化配置、自动化发布的互联网时代。

**2.12 Websocket**

想要实现一个系统的前后端实时通信，离不开Websocket。在本文所描述的应用平台，本人使用Spring自带的Websocket模块，SockJS。前端可以直接通过“HTTP://”为头的URL进行监听后端服务端口，使用Logback的过滤机制，把所需要的信息进行过滤操作，实时和前端进行交互。使用Websocket的优点主要有下：

（1）节省带宽。持续轮询服务器端数据的方法是使用HTTP协议。头部信息量大，有效数据比例低。然而，使用websocket，头部信息非常小，有效数据的比例很高。

（2）没有浪费。轮询方法可以轮询10次，只会遇到服务器数据更新，那么前9次都是白色轮询，因为没有更改任何数据。WebSocket由服务器主动发回，它所来自的只是新数据。

（3）实时性，考虑到服务器的压力，不可能用轮询方法进行短时间的间隔，否则服务器的压力太大，所以轮询时间间隔比较长，几秒钟，设置超过10秒。WebSocket由服务器推送，实时性能最高。

**3 系统设计分析**

**3.1 系统设计思想**

系统菜单分为横向菜单和纵向菜单，其中横向菜单包括：首页管理，货运管理，统计分析，基础信息管理，系统管理。纵向菜单包括：留言板模块，合同模块，报运模块，装箱模块，委托书模块，出货表模块，历史合同模块，厂家销售统计，产品销售排行，系统压力分布，财务统计，数据字典管理，厂家信息模块，以及用户，角色等管理。如图3-1所示。



图3-1 系统功能结构图

本管理平台为JavaWeb项目，分成前端和后端两部分。其中前端主要采用HTML、CSS、JavaScript、Ajax、jQuery、dTree的编写方式，后台主要采用SSM框架进行开发，整个项目由Maven提供自动化构建。

**3.2 可行性分析**

3.2.1 技术可行性

该系统是一个B/S架构的Web项目来对数据库进行管理查询的系统，且当前的技术能够具备开发这样系统的能力，所以是可行的。

3.2.2 经济可行性

对于现在计算机的普遍性，且又由于JSP的IDE都是开源的，对应的Tomcat，MySQL都是开源免费的软件，在框架上Spring，MyBatis等也都是开源的，所以经济上是可行的。

3.2.3 操作可行性

工作人员可以根据自己的职责要求被分配不同任务，通过简单的培训就能使用它，并且附带了操作说明，所以操作性是可行的。

3.2.4 组织可行性

本项目是基于MVC的模式开发，分成前端和后端两部分。在业务上，开发前台的人员和开发后台的人员可以独立开发，在整合部分也无需花费很多功夫。在功能上，本项目有Dao层，Service层，开发不同功能的人员可以独立开发，在项目进度上会比较快。此外，前台的图形化界面，以及简洁的按钮，管理人员无需了解其内部业务也可很快上手管理，所以这也是可行的。

**3.3 业务流程**

3.3.1 系统流程图



图3-2 系统流程图

一次交易的完成，应当进行以下的操作：首先是本公司代表与客户代表洽谈合作并签订合同，然后由本公司的相应人员进入合同录入。审核后可上报到报运模块，再由报运人员填写相关信息后进行存档，报运通过后再由装箱人员进行装箱工作，并且制定对应的海运委托书，还可以制定相应的发票，最后合同中的服装将在码头上送往世界各地，当此单完成后，将合同归档，便于后续查看，也提供系统查询速度。完成交易后，还可以进行一些简单的统计分析，观察哪些厂家的销售情况最好以及哪些产品销量最高，可以调整后期的产品生产以及厂家合作。过程如图3-2所示。

3.3.2 数据流程图



图3-3 数据流程图

浏览器通过访问作为处理器的B/S架构的好处之一就是用户无需配置复杂的环境，只需要一个浏览器便可以使用本平台，其工作原理如图3-3所示。

**3.4 用例分析**



图3-4 用例分析

用例图如图3-4所示。管理员主要负责系统的数据字典维护，以及功能的增删，权限管理等工作。领导在系统中只能查看所有业务，但是不能参与修改。一般用户指的是负责某部分功能的用户，他们只能操作自己当前的模块。高级用户一般指的是位于权限在一般用户之上的人（如主管），他们能够对当前模块进行删除操作，发布留言给下级职员等。

**4 系统设计**

**4.1 非功能性需求**

一个较为完善的系统，除了应该完成客户所要求的基本功能外，还应当做好一些细节的地方，为更好的体现本系统的友好性，本系统还完成了以下需求。

（1）软件的界面应简洁美观。

（2）软件响应时间不应过长。

（3）在用户完成相关操作后，应当及时给出反馈结果。

（4）分页操作，便于用户快速定位信息。

（5）Dao层，Service层的设立，便于后期维护以及功能扩展。

（6）启用数据库缓存，查询速度更快。

（7）启用框架的自动注入，实现零配置。

（8）数据库信息单独存放，移植更方便。

**4.2 功能性需求**

？？？综合管理平台的核心模块为货运管理模块，此外还有统计分析模块，基础信息模块，留言板模块，系统管理模块。详细分析如下：

4.2.1 货运管理

货运管理模块下还有二级菜单，包括：合同模块，出货表模块，报运模块，装箱模块，委托模块，历史合同模块，发票模块。

（1）合同模块

在该模块下，用户可以将已签订的合同录入本平台，录入成功后显示为“草稿”状态，随后用户可以查看该合同，也可以修改合同。在填写好基本信息后，可以在此页面总为合同添加货物(服装)和附件(包装盒)，添加完成后，在初始页面就可以查看到本次的成本总额。此后便可以将此合同打印输出Excel报表，也可提交至报运模块，此时合同会更新状态为“已上报”。

（2）出货表

此模块下可以得到对应月份的货物信息，用户选中相关日期便可以得到对应的出货表(Excel)。

（3）报运模块

在此模块下，由专门的报运人员负责，在此模块下列举出所有报运单，其状态均为“草稿”，此时可以查看或修改报运单，在确认无误之后，点击装箱按钮，填写相关的装箱信息便可提交到装箱模块，此时报运单状态更新为“已装箱”。

（4）装箱模块

在此模块下，由专门负责装箱的工作人员可以查看编辑装箱单，在此页面下，可以打印装箱单纸质稿以便于送交上级部门，也可以定制发票并打印输出。工作人员还能根据装箱单制定对应的海运委托书给负责船运的部门。

（5）历史合同模块

此模块的设立是为了提升查询的性能，在累积多次交易后，数据库中会有相当多的合同数据，而这些数据由于业务限制不能被删除，但是存放在数据库中影响查询效率，因此此模块是存放无效的合同或者是已经走完业务流程的合同的地方，只有查询和删除两个功能按钮。

4.2.2 统计分析

此模块下提供了四个子模块。包括了：生产厂家销售情况，产品销售排行，系统访问压力图，财务统计。

（1）生产厂家销售情况

此模块下提供了一个饼形图，图的数据来源于数据库，给用户直观了展示了每个厂家的销售情况的比率。

（2）产品销售排行

此模块下提供了一个柱形图，数据同样来源于数据库，横向对比了不同产品的销售量。

（3）系统访问压力

此模块下提供了折线图，用户每登录一次系统都会被记录在数据库中，此图展现了不同时间的访问量。

（4）财务统计

此模块下提供了简单的财务统计功能，目前仅记录了产品的利润，税率按5%计算。

4.2.3 基础信息管理

本模块下由两个模块组成，包括：系统代码，厂家信息。

（1）系统代码

本模块使用dTree实现，列举了本平台中的一些限定字符，如管理级别的名称选择被限定成“超级管理员”，“普通管理员”，“领导”等。采用两级树形结构展示，清晰简单。

（2）厂家信息

本模块录入了与本公司有合作关系的生产厂家的一些基本信息。用户在此模块下能够查看编辑这些信息，也能录入新厂家。对于不再合作的厂家可以改变其状态，由“启用”变成“停用”，为方便统计，本模块还提供了打印功能，可以快速提取厂家的联系电话及联系人等一些基本信息。

4.2.4 系统管理

本模块提供了三个子模块，包括：角色管理，部门管理，用户管理。

（1）角色管理

此模块为权限管理模块，在初始页面显示平台中已存在的角色以及对应的描述，此模块应当由管理员来操作。在本模块，可以查看当前角色有哪些用户，也可以新建新角色，新建角色的时候，可以选择系统内置的缺省角色然后进行自定义配置权限。

（2）部门管理

此模块提供了关于部门的基本操作，包括查看编辑，同时还能更改部门的启用状态。

（3）用户管理

此模块下展现了用户的基本信息，目前仅限普通管理员及以上用户才能编辑此页面。

4.2.5 系统首页

本模块提供了一些日常操作导航。

（1）左侧

个人工作台

该导航可以直接跳转到留言板编辑模块，用户可以给本公司的其他用户发布留言，目前暂不支持手机短信留言。

我的常用功能

该导航提供了创建留言，退出登录，查看帮助三个功能按钮。

用户设定

该导航目前只提供了用户个人信息修改。

（2）中间

该导航提供了五个菜单，包括：系统首页，货运管理，统计分析，基础信息，系统管理。点击不同菜单会跳转到不同模块。然后主体部分显示的是留言信息，这里显示了其他用户给当前用户发布的留言信息，用户可以查看信息，删除信息，更改信息的摆放位置。

（3）上方

该导航现在了当前在线用户的用户名(而非登录名)，所在部门。在靠近下方的三个按钮依次为刷新首页，查看个人信息，退出登录。

## 4.3 数据库概念结构设计

数据库部分采用PowerDesigner进行建模，在建模之前绘制了ER图。

用户ER图：用户隶属于某个部门，而用户将会拥有某种级别的角色，该角色在系统中会拥有不同的权限，如图4-1所示。



图4-1 用户ER图

合同ER图:一个合同下拥有不同的合同货物，一个合同货物对应了一个合同附件，其关系如图4-2所示。



图4-2 合同ER图

报运单ER图：系统中的报运单下拥有多个报运货物，是一对多的关系，而一个货物对应一个附件，是一对一的关系，具体如图4-3所示。



图4-3 报运单ER图

数据库总体ER图：本系统中涉及了25张数据库，其具体关系如图4-4所示。



图4-4 数据库总体ER图

## 4.4 数据库的逻辑结构设计

由于业务的复杂性，本系统涉及了25张数据表，表与表之间由外键联系，大体如图4-5所示。

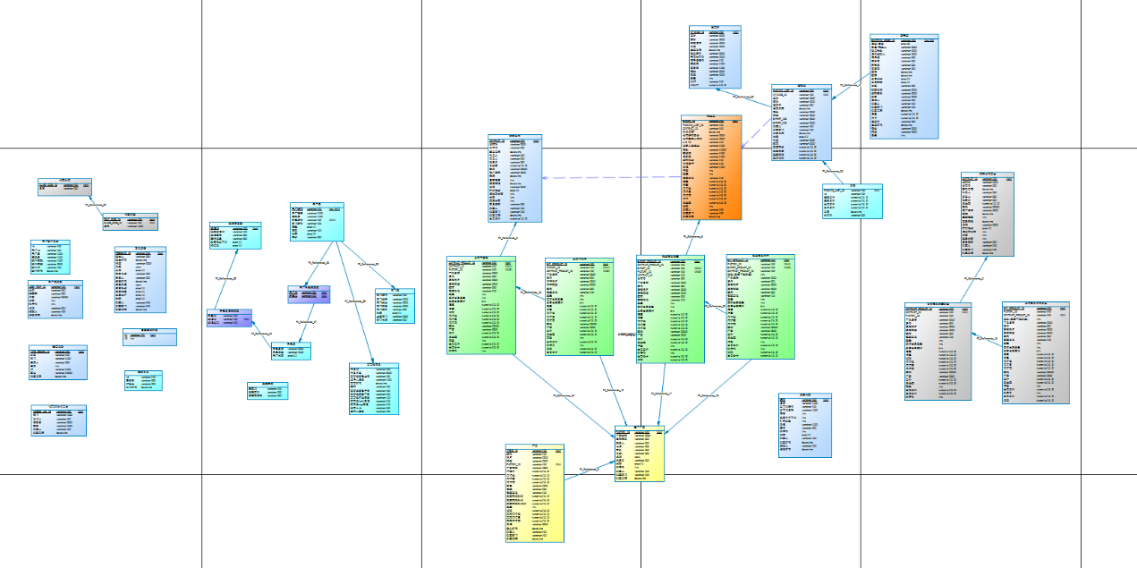


图4-5 数据库逻辑结构图

逻辑结构图中的数据库表如表4-1所示。

表4-1 数据库表

|  |  |
| --- | --- |
| 表名 | 备注 |
| 用户表 | 存放了用户的基本信息 |
| 部门表 | 存放了部门的基本信息 |
| 角色表 | 存放了角色的基本信息 |
| 权限表 | 存放了权限的ID，父ID等 |
| 用户角色关联表 | 关联了用户与角色，一个用户有一个角色，一个角色对应多个用户 |
| 角色权限关联表 | 关联了一个角色对应的权限信息 |
| 留言板信息表 | 存放了用户的留言信息 |
| 数据字典表 | 存放了一些名称，用户平台中的某些字段的限定 |
| 生产厂家表 | 存放了厂家的基本信息 |
| 合同表 | 存放了合同的基本信息 |
| 合同货物表 | 存放了某个合同中的货物的基本信息 |
| 合同附件表 | 存放了某个合同中的附件的基本信息 |
| 报运表 | 存放了报运单的基本信息 |
| 报运货物表 | 存放了报运货物的信息 |
| 报运附件表 | 存放报运附件的信息 |
| 装箱表 | 存放了装箱单的基本信息 |
| 发票表 | 存放了发票的基本信息，根据装箱单定制 |
| 委托单表 | 存放海运委托书的基本信息，根据装箱单定制 |
| 财务表 | 存放财务的基本信息，用于统计利润 |
| 历史合同表 | 存放已经归档的合同 |
| 历史合同货物表 | 存放已经归档的合同货物 |
| 历史合同附件表 | 存放已经归档的合同附件 |
| 登录日志表 | 存放了登录的一些基本信息，包括IP，登录名，时间等 |

**5 系统实现**

**5.1 系统流程图**

实际所做系统流程图如图5-1所示，为保证安全性，账号统一由公司管理员分配，用户可以自定义一些基本信息，但不涉及登录名，管理员可以随时启用或者弃用某个账号。用户成功登录后会自动匹配权限，系统会扫描数据库，查询当前登录用户的权限，系统会根据权限的有无而决定用户的显示页面，提供对应的二级菜单和功能按钮。



图5-1 系统流程图

**5.2 业务代码执行流程**

当用户在前台点击某个功能按钮，如用户点击了“查看”，首先由SpringMVC接管了该请求，随后映射到具体的Controller，其后执行Service层的业务方法，最后由Dao层来对数据库进行CRUD操作，具体如图5-2所示。



图5-2 代码执行流程图

**5.3 项目文件结构**

本项目由Maven提供自动化构建，其具体文件结构如图5-3所示。

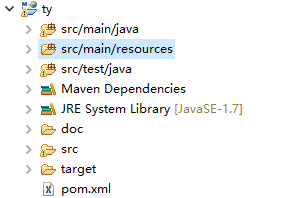


图5-3 代码结构图

Src/main/java中存放了核心代码部分，src/main/resources存放了相关配置文件包括数据库信息，mybatis映射文件，spring和SpringMVC配置文件及log4j配置文件，src/main/java存放了一些测试代码，doc存放了API文档，src存放了JSP页面，target存放了由Maven编译后的文件，pom.xml为Maven配置文件。

核心代码的文件结构如图5-4所示。

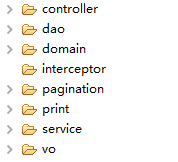


图5-4 src目录结构

Controller部分实现各类业务的逻辑功能，Dao部分实现的是对不同数据库的操作，domain部分是各种实体(PO对象)，VO部分是对PO的一些扩展，Service部分是对业务的操作。

**5.4 编号命名规范**

在系统的编号设计中，采用了统一的规范命名，图5-5列举了部分命名规范，包括了“人员级别”、“附件类型”等。

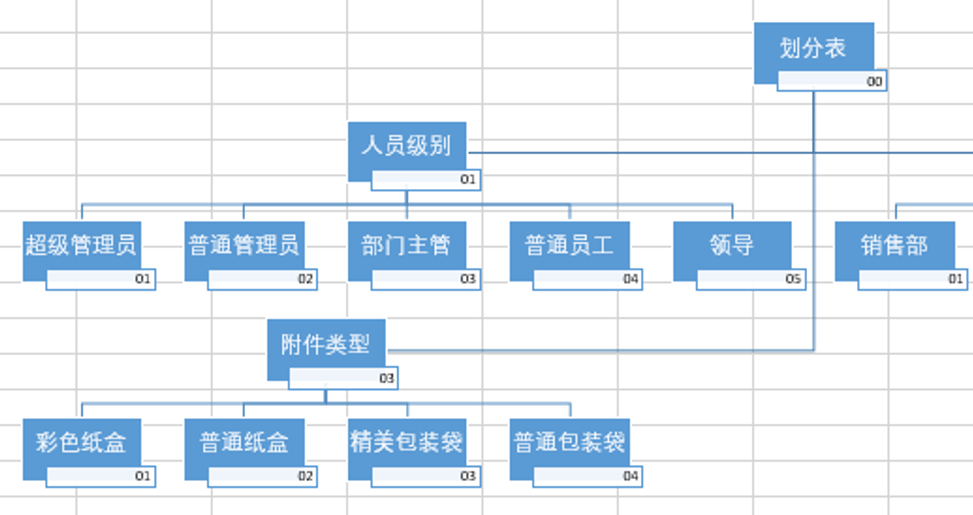


图5-5 数据字典命名规范

图14是存储在数据字典中的编号规范，当此数据项被应用于不同表时，需加上不同的前缀。部门表加前缀“D”，角色表加前缀“R”，附件类型表不加。

部分权限命名：命名每一位为一个权限，权限命名均为小写。子权限命名为父权限加子权限共同组合。如a代表系统首页，其子权限则为aa-az，以此类推，如图5-6所示。

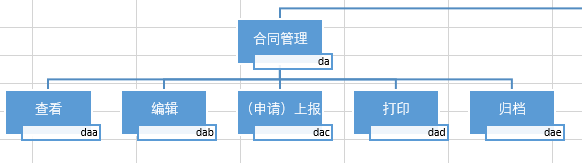


图5-6 权限命名规范

**5.5 用户登录**

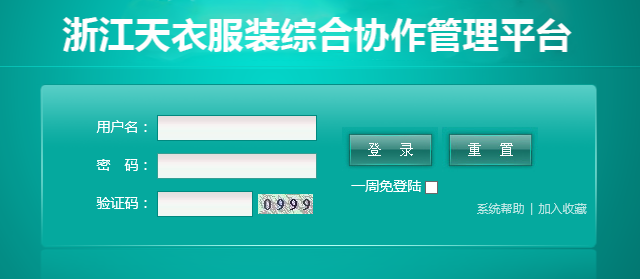


图5-7 用户登录界面图

用户从管理员处得到账号密码后可以登录此平台，所有文本框属于必填项，不填写会弹窗通知，其中验证码随机产生，并且限定为4位，此外用户名文本框被限定只能输入除中文外的其他字符。通过勾选“一周免登陆”，后台会设置Cookie以保存用户的账号及密码，时间为一周。界面如图5-7所示。点击登录后，后台会首先查看验证码是否通过，然后再去匹配数据库中的账号密码。验证成功后，后台会自动记录登录信息到日志表去，如果登录失败则重定向到登录页面。

**5.6 用户主界面**

用户主界面由三部分组成，标题部分、左侧工具栏、主体部分。



图5-8 系统首页图

主页面上有五个横向菜单：系统首页、货运管理、统计分析、基础信息、系统管理，如图5-8所示。

5.6.1 留言板

系统首页的主体是一面留言板，当用户登录系统后，后台会从数据库中查询收信人是当前用户的所有留言板信息，并显示在留言板上，用户可以阅读、删除某个留言。在IE浏览器中还可以随意拖动留言的位置，点击“位置”功能按钮即可保存新的留言位置。

（1）查看留言

由业务层根据当前用户的ID得到其留言信息，并传递到页面上。

（2）创建新留言

点击“新建”或者是“我的留言板”，后台会返回新建留言板的物理地址给前台。如图5-9所示。



图5-9 留言板创建图

选择好标签颜色及装饰后，编辑留言信息，点击确定后台会保存到数据库并回到系统首页。

5.6.2 个人信息修改

点击信息修改，后台会根据当前ID查询数据库，返回相关信息到页面以修改，其中，状态和登录名是用户不能修改的。如图5-10所示。



图5-10 用户信息修改

5.6.3 标题部分

本部分显示了本公司的 LOGO以及公司全称，在右侧上方，是用户名和用户所处部门，点击即可查看个人信息和部门信息，下方三个按钮分别是跳转到系统首页、查看个人信息，退出登录。页面如图5-11所示。



图5-11 标题栏

**5.7 货运管理**

本模块提供了对合同、报运、装箱、委托书、发票的操作，并且提供了Excel表格的导出。其中左侧这些菜单均由权限控制，只有当前用户拥有此权限才会显示。页面如图5-12所示。

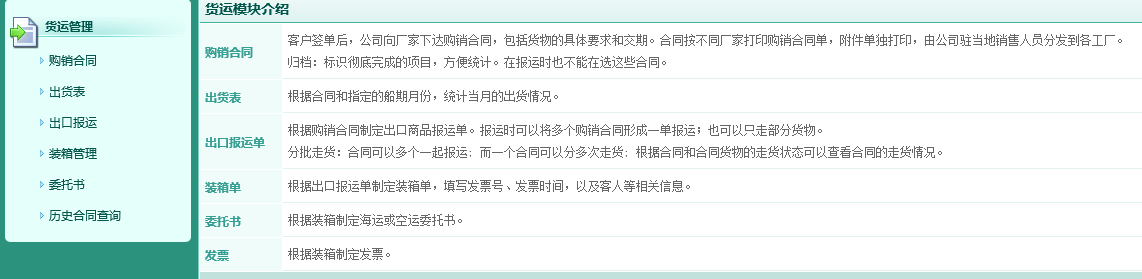


图5-12 货运管理首页

5.7.1 购销合同

点击“购销合同”功能按钮，出现下图，此页面提供对合同的查看编辑功能，上报至报运部门，打印输出合同表，合同归档等功能。这些操作按钮由权限控制显示，只有拥有对应权限的才会显示。页面如图5-13所示。



图5-13 购销合同首页

（1）新建

点击“新建”，前台会传递ID到后台，后台经过RequestMapping映射后将返回新建页面的物理地址。页面如图5-14所示。

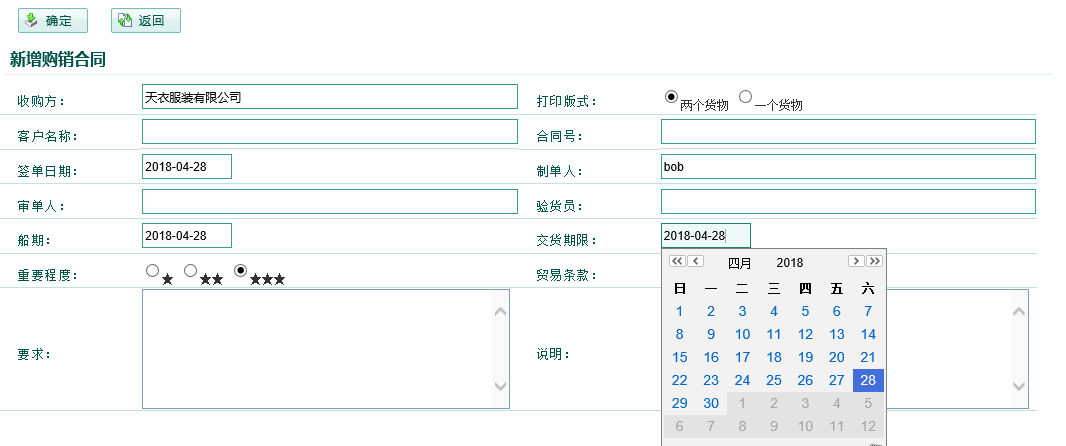


图5-14 新增购销合同

填写好信息后，点击“确定”，后台会根据反射，将值赋给domain，其后调用业务层方法来存储到数据库。

（2）查看

勾选某个合同，点击查看按钮。前台会以POST的方式提交ID到后台，后台接收到ID后，调用业务层的VIEW方法来查询数据库，并回显到前台页面中。如图5-15所示。



图5-15 查看购销合同

（3）修改

勾选需要修改的条目，点击“修改”。前台会携带ID给后台，后台先查询数据库，然后将数据回显到前台以便于修改。页面如图5-16所示。

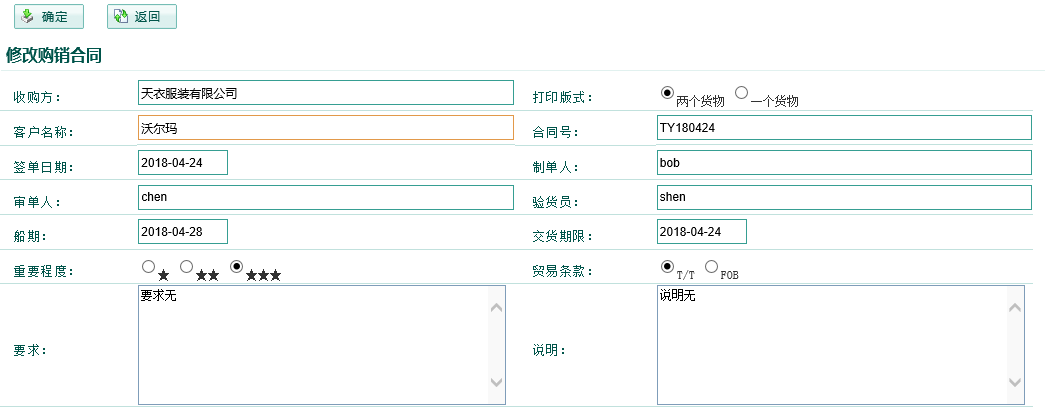


图5-16 修改购销合同

（4）删除

点击“删除”按钮，后台会根据ID来查询数据库，然后删除此条目，此后重新返回到首页。

（5）打印

勾选某个合同，点击“打印”按钮，后台会查询相关信息，然后提供下载。报表如图5-17所示。



图5-17 合同报表示例图

打印功能使用Apache的POI实现。

（6）归档

勾选某个合同，点击“归档”后，平台会根据ID，查询数据库，将此合同相关的数据均转移到历史合同表。

5.7.2 出货表

点击左侧“出货表”功能按钮，前台会展示一个新页面，如图5-18所示。点击“打印”，传递给后台一个船期，后台根据此日期查询数据库，然后打印输出表格。



图5-18 出货表首页图

5.7.3 出口报运

该模块提供了关于报运单的查看功能，修改功能，删除功能以及上报至装箱部门。

点击“出口报运”功能按钮，前台页面会改变。如图5-19所示。



图5-19 出口报运首页

（1）查看

勾选某个报运单，点击“查看”，返回页面如图5-20所示。



图5-20 查看报运信息

（2）装箱

点击“装箱”，后台根据前台传递过来的ID，返回如下页面，如图5-21所示。

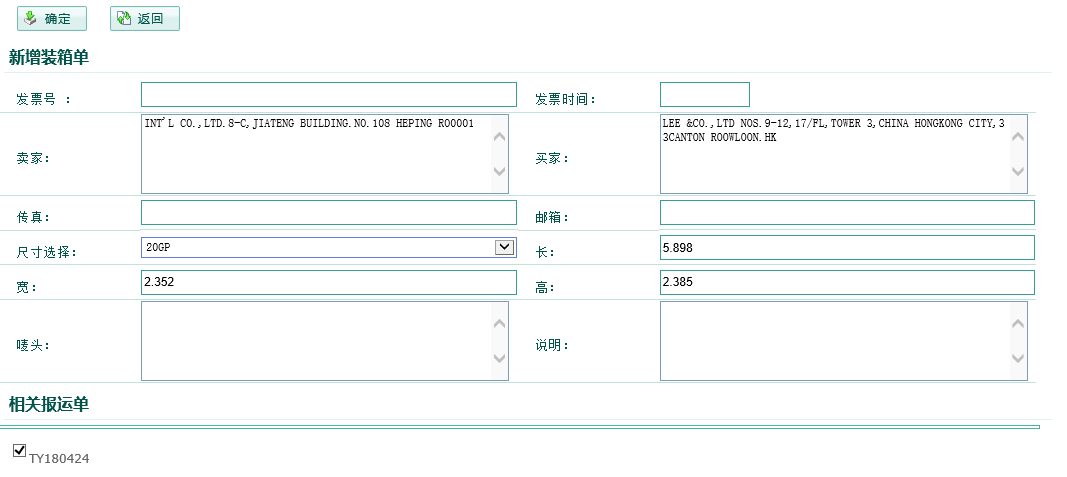


图5-21 新增装箱单

首先根据ID，更新报运单的状态为“已装箱”，然后将相关的合同ID置于页面上，用户填写相关信息后即可完成装箱。

5.7.4 装箱管理

该部分提供了关于装箱单的查看功能，修改功能，删除功能，定制发票以及打印输出报表功能。

（1）查看

根据ID查询数据库，页面如图5-22所示。



图5-22 查看装箱单

（2）委托

勾选某个装箱单，点击“委托”，可定制对应的海运委托书，填写好相关内容后，点击“确定”提交。如图5-23所示。



图5-23 创建委托书

（3）打印

勾选某个装箱单，点击“打印”即可输出Excel报表。样例如图5-24所示。

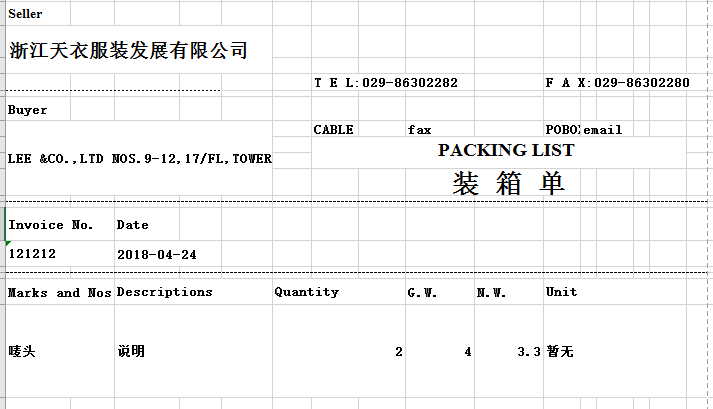


图5-24 装箱单报表示例图

（4）发票

点击“发票”，后台根据当前ID查询数据库，然后提供报表下载。如图5-25所示。

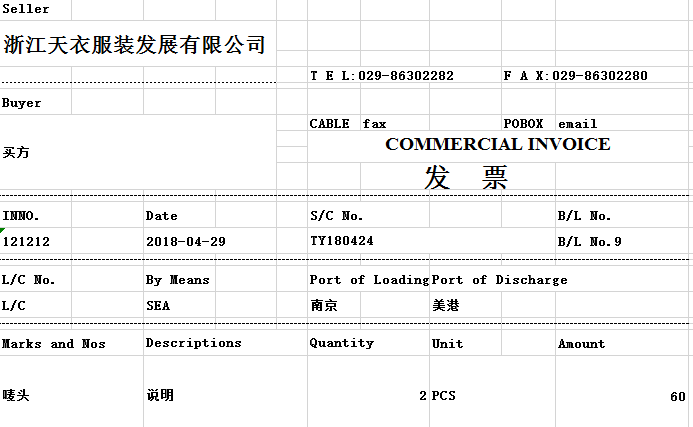


图5-25 发票示例图

5.7.5 委托书

委托书模块，本模块提供了委托书的打印、删除功能。页面如图5-26所示

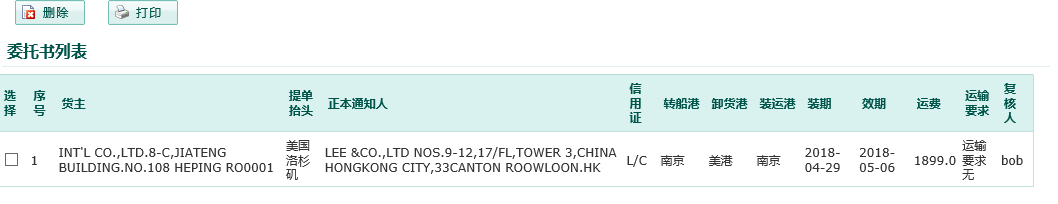


图5-26 委托书首页

（1）打印

点击“打印”，后台生成委托书以供下载。样例如图5-27所示。

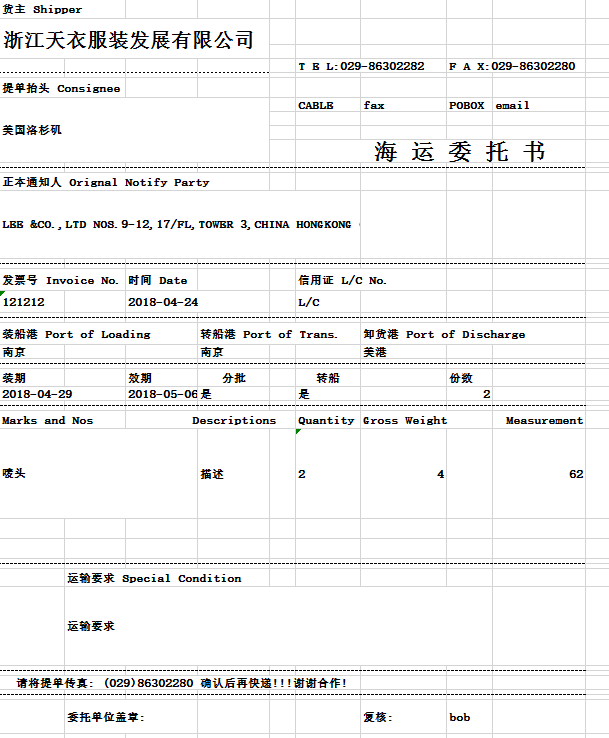


图5-27 海运委托书示例图

**5.8 统计分析**

本模块主要是一些统计信息，包括生产厂家的销售情况、产品的销售排行记录、系统访问压力图、以及简单的财务统计。页面如图5-28所示。

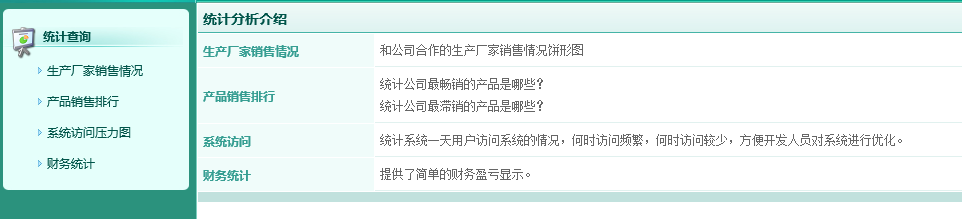


图5-28 统计分析首页

5.8.1 生产厂家销售情况

此页面采用了amCharts的饼形图来显示销售情况，其数据来源于数据库的查询，然后通过预置flash动画读取xml配置文件，然后在支持Adobe Flash Player的浏览器中可以显示出来，如图5-29所示。

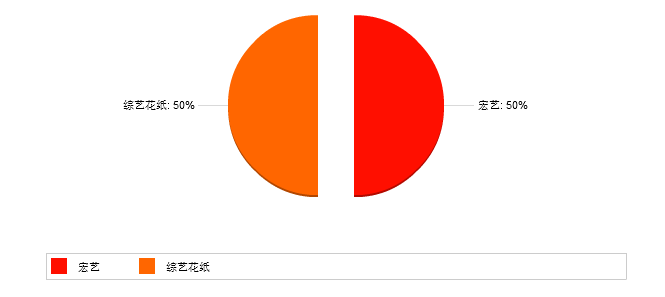


图5-29 厂家销售情况图

5.8.2 产品销售排行

此页面采用amCharts的柱形图来直观的展示不同产品的销售情况，该柱形图也属于flash动图数据来源于contract\_product\_c数据表，图5-30展示了两种货物的销售排行情况。

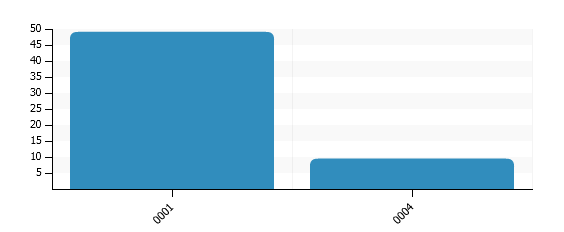


图5-30 产品销售排行图

5.8.3 系统访问压力图

此页面的折线图的数据来源于login\_log\_p数据表，用户每次登陆的时候，都会往此表写入数据，用户记录不同时刻的访问量。如图5-31所示。

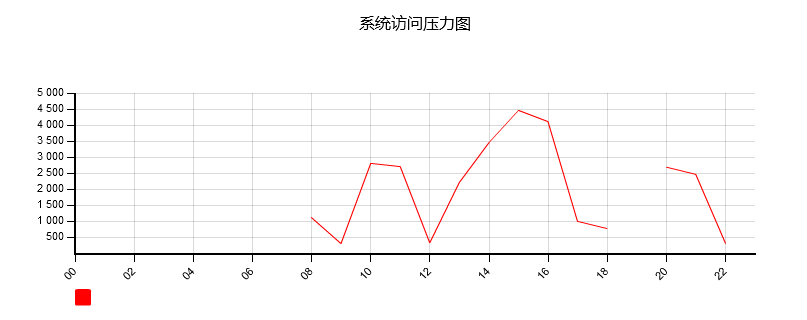


图5-31 系统访问压力图

5.8.4 财务统计

本页面是简单的财务统计情况，记录了成本和销售额，以及利润情况，如图5-32是2018年4月份的财务情况。

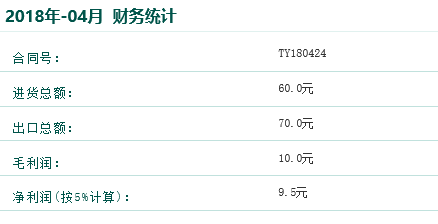


图5-32 财务统计样例

**5.9 基础信息**

此模块是记录了本系统的一些基础信息，包括数据字典以及厂家信息，其中厂家信息支持报表导出。页面如图5-33所示。



图5-33 基础信息首页

5.9.1 数据字典

此页面展示了系统中的数据字典，数据字典的作用是限定了系统中某些字段的选择，如人员级别、附件类型等，具体如图5-34所示。

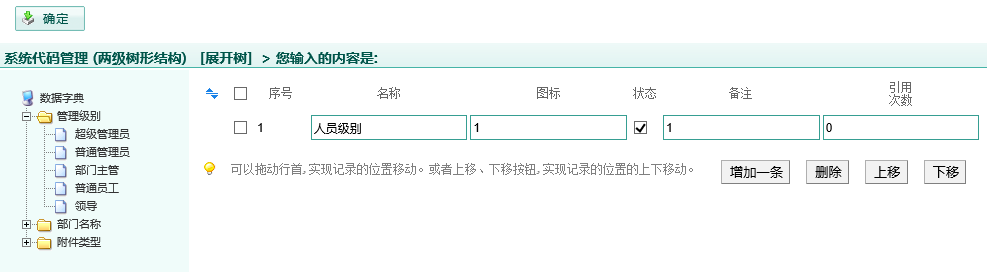


图5-34 数据字典树

5.9.2 厂家信息

本页面记录了所有与本公司合作过的厂家的基本信息，支持报表导出通讯录。页面如图5-35所示。



图5-35 生产厂家首页

本处使用了开源项目pageHelper进行分页。需在pom文件中添加如下依赖。

<dependency>

<groupId>com.github.pagehelper</groupId>

<artifactId>pagehelper</artifactId>

<version>5.1.2</version>

</dependency>

后台默认分页以5条记录为一页，在select框提供了“5”，“10”，“20”三种类型。



图5-36 分页选择

页面中图5-36部分做了验证处理，当没有前一页或者是已经是第一页的时候，图片变成灰色，并且点击无跳转效果。

**6 系统管理**

本系统并未设立独立的维护系统，管理员和用户使用的是同一个系统，有所不同的是，管理员由于拥有最完善的权限以及数据库账号密码，因此，管理员能对系统进行高级别的操作和对数据库的直接修改。本模块提供了角色管理、部门管理、用户管理三个功能。如图6-1所示。

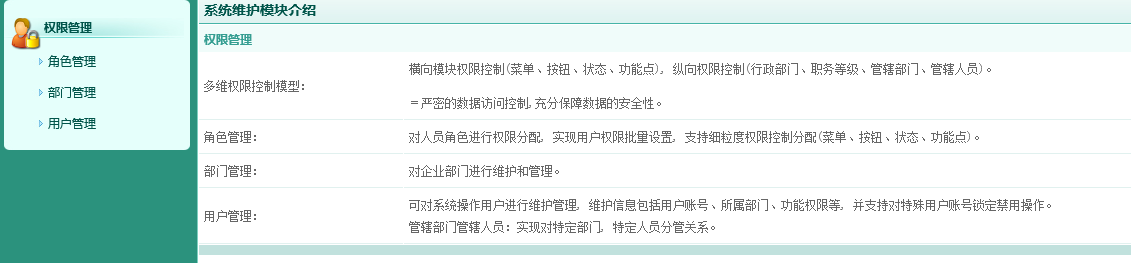


图6-1 系统管理首页

**6.1 角色管理**

本模块提供了对系统内的角色的查看编辑，以及新建等功能。如图6-2所示。



图6-2 角色管理首页

（1）查看

勾选某个角色，点击“查看”，后台会根据已选ID查询数据库中为此角色的所有用户，别显示到前台。页面如图6-3所示。



图6-3 查看角色

（2）新建角色

点击“新建”，后台查询数据字典表，将其中的关于角色的数据项显示到前台，并将所有权限以表格的方式展现。系统中已为各种角色预设一部分权限，管理员还可以为新角色自定义新权限。页面如图6-4所示。



图6-4 新增角色

（3）修改角色

点击“修改”，后台将相关数据显示到前台，如图6-5所示。



图6-5 修改角色

**6.2 部门管理**

部门管理模块提供了“查看”、“新建”、“修改”、“删除”、“启用”、“停用”等功能，如图6-6所示。页面列表同样提供了分页功能，同时也包含了部门信息的Excel导出，表格如图6-7所示。



图6-6 部门管理首页



图6-7 部门信息表示例图

**6.3 用户管理**

用户管理提供了“查看”、“新建”、“修改”、“删除”、“启用”、“停用”等功能，如图6-8所示。分页部分采用PageHelper进行分页，提供用户信息导出功能。信息表格如图6-9所示。



图6-8 用户管理首页

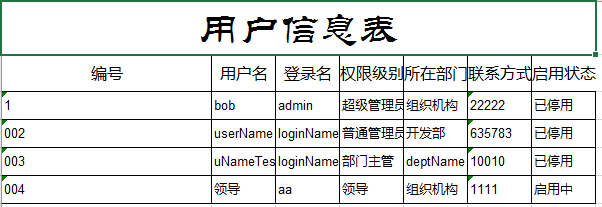


图6-9 用户信息表示例图

**7 总结**

。

**参考文献**

[1] Craig Walls.Spring实战[M]. 北京：人民邮电出版社，2015：123-245.

[2] Simplifying Enterprise Java Applications with the Spring Framework[M].Apress: 2008-06:10-45.

[3] Leslie M. Fine. Selling and sales management[J]. Business Horizons, 2007,50(3):2-8.

[4] H.Sebastian Heese,Jayashankar M. Swaminathan. Inventory and sales effort management under unobservable lost sales[J]. European Journal of Operational Research, 2010,207(3):12-20.

[5] 申和平. 设计模式在企业管理系统中的研究与应用[D]. 大连理工大学，2007.

[6] 冯金辉. J2EE设计模式比较及应用实现研究[D]. 浙江大学，2004.

[7] 郑希彬. Web应用系统设计模式的研究与应用[D]. 山东科技大学，2005.

[8] 石明辉. 基于JAVA的多数据库中间件的设计与实现[D]. 上海交通大学，2011.

[9] 余冬梅, 张秋余, 张涛, 王海军.JAVA在WEB数据库中间件中的应用[J]. 计算机工程与应

用，2002(17):208-210.

[10] 徐雯, 高建华. 基于Spring MVC及MyBatis的Web应用框架研究[J]. 微型电脑应用， 2012，28(07):1-4-30.

[11] 李霞. MVC设计模式的原理与实现[D]. 吉林大学，2004.

[12] 刘增杰. MySQL5.5从零开始学[M].北京：清华大学出版社，2012-03：23-45.

[13] 许晓斌. Maven实战[M].北京：机械工业出版社，2011-04：123-205.

[14] 李刚. Spring+Mybatis企业应用实战[M].北京：电子工业出版社，2017-07：78-103.

[15] 许令波. 深入分析Java Web技术内幕[M].北京：电子工业出版社，2014-09：203-245.

[16] 陈静红. 服装企业销售管理信息系统的分析与设计[D].云南大学，2013.

[17] 杨丽辉. 基于J2EE架构MVC模式的房地产销售管理信息系统的设计与实现[D].吉林学，2004.

**致谢**

。