

浙江农林大学

本科生毕业设计（论文）

（2019 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 5 月 8 日

浙江农林大学

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5月8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 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。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、公共服务等做出智能的应答。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，具体的内容将在本文章节进行阐述。

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 life, food, clothing, housing and transportation are inextricably linked with the Internet industry. More and more Internet technologies are used in people's daily life, such as artificial intelligence, cloud computing, big data, machine learning and so on. Through the Internet of Things and the new generation of Internet technology, combined with the new generation of innovative ideas, to respond intelligently to social life, public security, medical, public services and so on. Based on the relatively mature projects of intelligent government, intelligent transportation, intelligent medical treatment and intelligent tourism, this paper mainly studies the product circulation, visualization and data mining with agricultural products as the main research object, so as to build an application platform to facilitate people's livelihood.

Through market research, we understand the current demand of agricultural products and the vacancies of standardization of agricultural products in various vegetable markets, and formulate the appropriate demand model and design scheme. The specific content will be elaborated in this chapter.

Key Words: Big Data, Data mining, Predictive analysis, Smart City, Visualization platform

目 录

摘要	I
Abstract	II
1 绪论	1
1.1 课题的背景	1
1.2 现状的分析	1
1.3 产品流通的可视化表达与数据挖掘应用平台的优势	1
1.4 本论文主要结构	2
2 开发环境与相关技术	3
2.1 系统开发环境	3
2.2 React	3
2.3 ECharts	3
2.4 npm	4
2.5 axios	4
2.6 SSM 框架	4
2.6.1 Spring	4
2.6.2 SpringMVC	5
2.6.1 Mybatis	5
2.7 Java 8	5
2.8 Druid	6
2.9 Lombok	6
2.10 Logback	7
2.11 Maven	7
2.12 Websocket	7
3 系统设计分析	8
3.1 系统设计思想	8
3.2 可行性分析	8
3.2.1 技术可行性	8
3.2.2 经济可行性	8
3.2.3 操作可行性	9
3.2.4 组织可行性	9
3.3 业务流程	9
3.3.1 系统流程图	9
3.3.2 数据流程图	10
3.4 用例分析	10
4 系统设计	12
4.1 非功能性需求	12
4.2 功能性需求	12
4.2.1 数据爬取	12
4.2.2 产品未来价格预测分析	13
4.2.3 大数据汇总	13
4.2.4 商品查询	14
4.2.5 登录和注册	14

4.3	数据库概念结构设计	14
4.4	数据库的逻辑结构设计	15
4.5	业务代码执行流程	17
4.6	项目文件结构	18
4.7	项目命名规范	21
5	系统实现	22
5.1	用户模块	22
5.1.1	登录界面	22
5.1.2	注册界面	22
5.1.3	导航栏	23
5.2	首页模块	23
5.2.1	上月统计	24
5.2.2	农产品分类统计	24
5.2.3	农产品明细统计	25
5.3	产品价格详情模块	25
5.3.1	图片按钮分类	26
5.3.2	价格比较图	26
5.3.3	分页表格	27
5.4	未来数据挖掘模块	27
5.4.1	树状索引搜索查询	28
5.4.2	历史价格走势与未来数据挖掘	29
5.5	爬取数据更新模块	29
5.5.1	爬取进度交互	30
5.5.2	爬取信息交互	30
6	总结	32
	参考文献	33
	致谢	34

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。但物价上涨和通货膨胀的关键性因素当然不只是媒体的广泛报道，蔬菜卖起来困难、蔬菜买起来成本高的问题大部分原因在于购买蔬菜的价格上涨。政府控制房价不高的调整，但对一般农副产品蔬菜、肉类和水产等的监管力度却相当强大，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，农民们种出的菜滞销，卖不掉导致了“卖菜难”；另一方面，城市普通居民因为买菜市场稀少，菜贩子们“坐地起价”而感到“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]。

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，这些都与我们的日常生活有着千丝万缕的联系，我们每个人都每天都一日三餐。而这些商品在菜市场、超市等生活用品采购地都可以十分方便地购买到。商场采购地作为一个城市的公共配套设施联系着一座城市里千万家的衣食住行，是民生所系、政府之责、公共服务必需。买方的不便和不安全的，价格便宜的菜的安全问题密切相关的日常生活中的所有人，这将直接影响到人们的幸福指数的影响。所以，经过本人的前期市场调研，一个集智慧、

便利和互联网于一体的大数据应用平台在本人的脑海中逐渐清晰起来。本文所描述的大数据应用平台的主要优势体现在以下 5 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）本文所描述的农副产品大数据应用平台还能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、全面化的发展；

（5）该农产品大数据分析应用平台还能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

本论文将在项目课题的开发环境与相关技术、系统设计分析、系统设计交互以及实现等 7 个方面进行阐述，并展示本项目课题的研究成果。

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^[2]。它并不直接工作，而是直接工作在虚拟的文档对象模型。这一个框架不仅向我们介绍了“虚拟”的概念，并将它添加到文档对象模型在 JavaScript 和文档对象模型之间的逻辑实现，十分有效地提高了网络的性能。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^[3]。其次，这一虚拟文档对象模型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^[4]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，同时也提供了一个便捷的方式来管理不同版本的代码^[5]。

2.5 axios

axios 是一个 HTTP 客户端，它主要用于 NodeJS 和浏览器。之所以我选择使用 axios 而放弃使用 ajax 请求，是因为本文所描述的大数据应用平台在系统设计上就采用前后端分离技术。仅仅为了使用 ajax 请求就要把整个 JQuery（一个快速、简洁的 JavaScript 框架）全部引入，很显然，这种操作对于一个规范的大型 Web 项目是十分地不合理。但是 axios 就比 ajax 小巧灵活得多，只要使用 npm install axios 就能安装并导入使用，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^[6]。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。但是，Spring 框架的使用不仅仅应用在 Java Web 应用项目的服务端开发。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^[7]。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^[8]。下面我就简单介绍几个 Spring 的特点：

（1）控制反转：

控制反转的英文简称叫做 IOC，又称作是依赖注入，当工厂模式下的实体和对象是用集装箱模式操作时，它只需要配置文件中的相应的 Bean 和相应的属性，就能成为容器和管理容器中的物品。在所有的配置文件完成之前，然后具体说明 Bean 是在 A 类中启动的，如果你需要通过容器对它们进行调用分配。

（2）面向切面：

面向切面的英文简称叫做 AOP, 是 Spring 框架中的另一个重要知识体系架构。与 OOP 一样, AOP 是一种编程思想, 而不是一种技术^[9]。在面向对象编程的思想中我们引入了三个基本特征: 封装性、继承性和多态性。那么, 面向切面编程的思想就是对面向对象编程思想的补充完善和改进优化。面向对象编程的思想也只是允许声明决定大多数具体的事务事件行为的从上往下的纵向层级事务联系, 但面向对象编程的思想却往往不适合声明决定大多数具体的事务事件行为从左到右的横向并列事务联系。例如, 代码登记处的日志登记功能: 登记所有倾斜的代码横向分布对象, 而不论其是否为分配功能。在面向对象的编程中, 代码的设计和编写通常会产生大量的重复冗余代码块, 这对于一个项目的维护和管理是非常不好的, 所以在面向对象编程思想不能很好地完成横向层级之间的事务联系这种类似的任务时, 面向切面编程的编程思想就能很好地完成这部分任务, 并能在实际开发的过程中很好的解决代码功能块冗余和重复的问题。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)^[10]。

2.6.2 SpringMVC

SpringMVC 是一个 MVC 框架, 通过应用视觉控制模型来区分数据、操作和陈述。SpringMVC 是围绕着分发器(DispatcherServlet)设计的, 这种分发器负责将前端的所有请求按照一定的规范规则分别分发给某些特定的处理程序。这些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^[11]。

在一个标准的 Java Web 项目中, 由 SpringMVC 中的各种各样的 Controller 进行所有前端请求的拦截。经过之前 Spring 框架的 Bean 装配后, 将网站所需要的拦截的请求进行分配映射, 并经过后端的逻辑层和数据层进行后端运算后, 将数据渲染在 View 视图层, 渲染视图的快慢主要受硬件配置和大数据开发技术的影响。在本文后半部分将介绍快速渲染视图的方法和控件。

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^[12]。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 Java 8

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^[13]。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，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 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^[14]。Java8 的函数式编程的几个优秀实战技术和专业技能方面的知识在本文中我均已阐述，当然如果大家在这方面感兴趣，可以阅读我在文末的参考文献《Java8 实战》^[15]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^[16]。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，在此不再赘述。

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^[17]。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致编译失败。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。为了节省带宽，Websocket 将通过 HTTP 协议对数据服务器的大量数据和低数据率进行深入调查。

（2）减少浪费。询问的方法只能有一个服务器，第一次更新数据，然后调查是白色的，因为它没有改变，没有服务器的数据。WebSocket 由服务器主动进行数据的发送，它只会在数据进行更新时才进行发送的操作。

（3）实时性高。考虑到实际发送的时间不固定、服务器的压力动态地发生变化、在服务器超载时的短时间轮调以及及时自我修复反馈的方法，选举之间的时间可能会较长，有时几秒钟，当然也有可能超过 10 秒。但是 WebSocket 由服务器动态地进行推送，实时性能达到了最高。

3 系统设计分析

3.1 系统设计思想

本系统的功能模块分为横向功能和纵向功能。横向功能包括：用户管理、大屏展示、产品搜索、产品分析和数据获取等等大类功能；纵向功能细分为依次细分为：用户注册、用户登录，上月产品汇总数据、上月全国产品流通、上月热门产品汇总，所有产品查询、按类产品查询，单一产品搜索查询、单一产品数据挖掘，网络爬虫等模块功能。具体如图 3-1 所示。

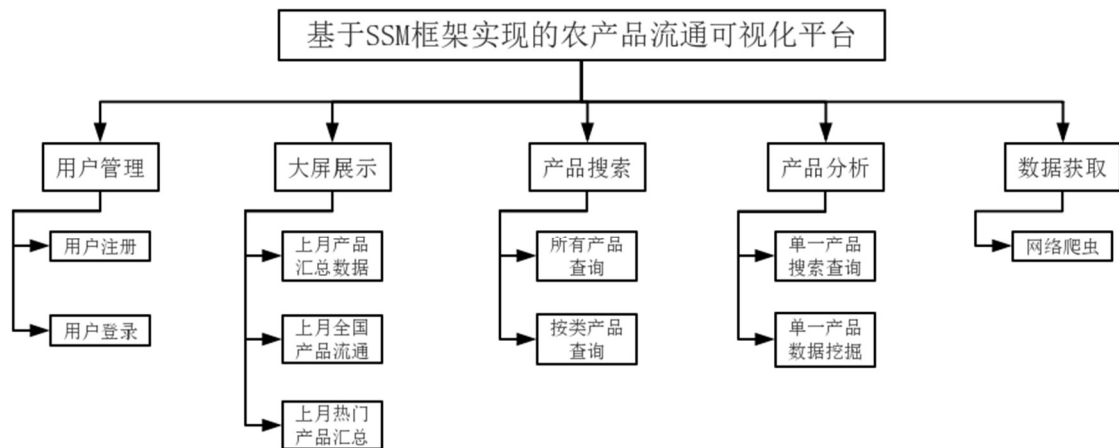


图 3-1 系统功能模块结构图

本大数据产品流通平台为基于 SSM 框架的 Java Web 项目，分成前端和后端两部分，并采用前后端分离编码、模块化编码。其中前端主要采用 React、Javascript、Bootstrap、Axios、Ant Design、Echarts、NPM 等编码技术，后端主要采用 SSM 框架进行模块开发，主要开发技术包括：SSM、Java8、Druid、Lombok、Logback、WebSocket、Maven 等等。本大数据应用平台开发严格按照 GIT 代码管理和阿里巴巴 Java 编码规范手册，具有非常强的可移植性、可扩展性和可维护性。

3.2 可行性分析

3.2.1 技术可行性

就目前现阶段来说，大数据开发技术已经不断成熟，很多公司都已经基于 Hadoop 和 Impala 成立专门部门进行大数据分析实践^[18]；至于 B/S 架构的 Java Web 网站也已在十几年前逐步发展成熟，所以，从技术层面上来说，这个大数据应用平台是完全可以实现的^[19]。

3.2.2 经济可行性

21 世纪互联网时代，信息技术逐渐平民化，开发成本逐渐在降低；本文所描述的大数据应用平台所使用的技术都是开源技术，不需要支付任何费用，开发成本比较低。最重要的一点是，本问所描述的应用平台研发成功后能给带来巨大回报，能很好地解决“卖菜难，买菜贵”等社会民生问题，极大便利了我国广大人民的生活方式，加快了我国智慧城市化建设。

3.2.3 操作可行性

用户动态页面交互只有若干个简明按钮和特色图片，同时考虑了不识字和识字人群，操作简单，功能强大，界面美观，符合用户与人机之间的交互体验性。

3.2.4 组织可行性

本项目是一个基于 SSM 框架的前后端分离的 Java Web 网站，前端使用 WebStorm 开发，后端使用 IDEA 开发，数据库使用 MySQL，服务器为 Tomcat。基本架构合理，网站部署简单高效，接口联调规范，各接口之间的方法互不干扰影响，组织架构明了，后期可扩展性和可维护性较高，模块简单清晰明了。

3.3 业务流程

3.3.1 系统流程图

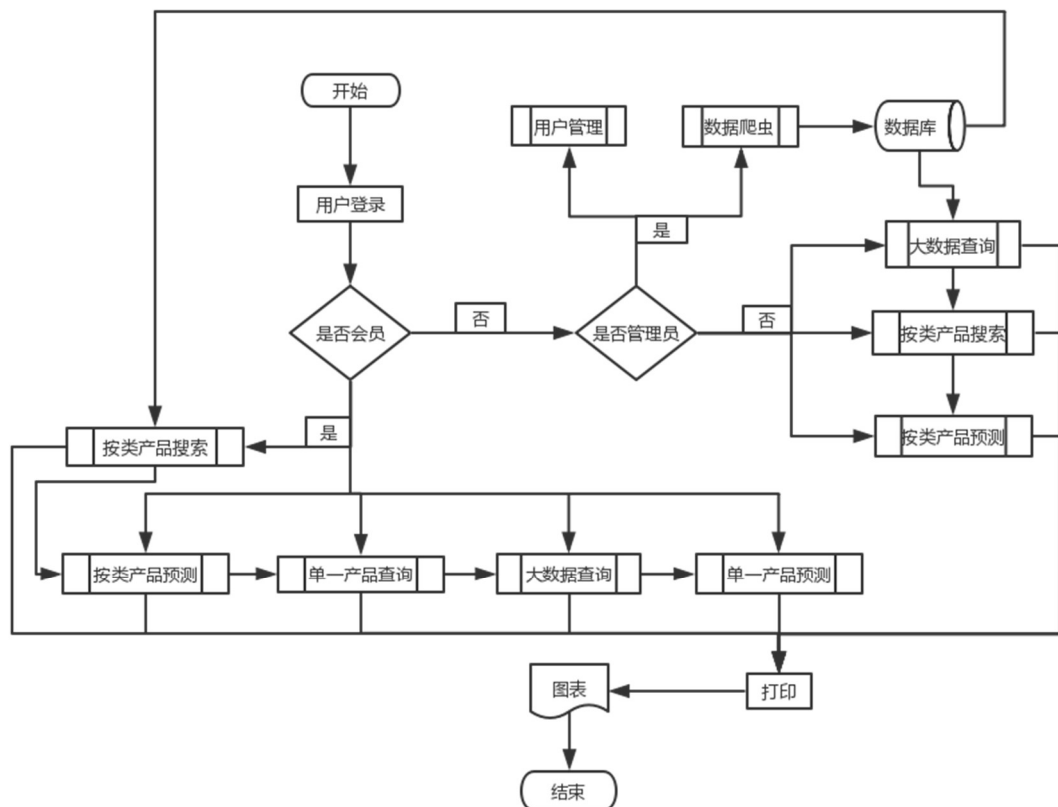


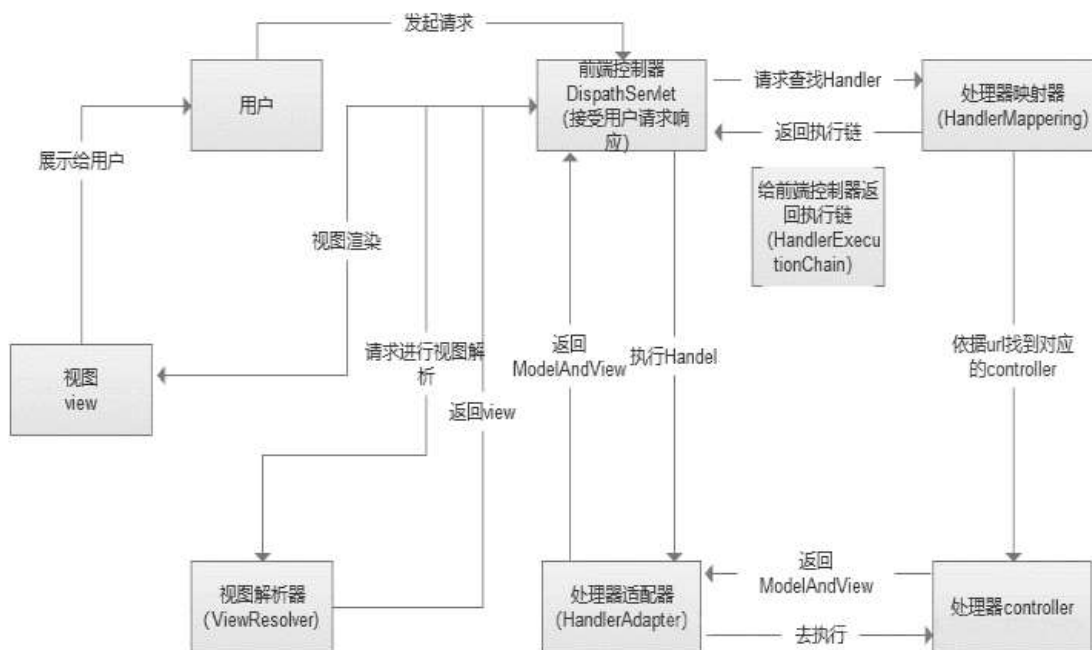
图 3-2 系统流程图

对于本大数据应用平台的完整操作流程大致如下：首先用户应该注册账号进行登录，然后本平台会自动识别用户权限，如果是普通用户将会看到上个月产品流通和各类热销产品的汇总数据，进入详情页可以查询各类产品的历史价格行情；如果平台识别出该用户为会员用户，除了普通用户的基本功能之外，还可以进行产品查询搜索进行单一产品的历史价格查询和对该产品的未来数

据分析预测；每种类型的用户都可以打印各类产品的价格图表方便今后直接查看。除此之外。管理员用户，负责每天的数据维护和更新还有用户权限的分配；但是因为目前该应用平台的推广和正处在成熟优化阶段，目前所有用户都免费享有会员用户的所有权限，所有历史数据也未在云端同步，各个用户也同时享有管理员权限，按天可以同步爬去所有数据。

3.3.2 数据流程图

前后端数据流转图：



业务流转图

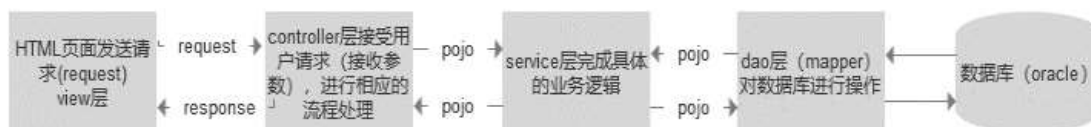


图 3-3 数据流程图

基于 SSM 框架搭建的 B/S 架构 Java Web 网站，其最大的优势就是用户所需的配置尽可能地简单，对用户的隐私有着合法的保护；用户无需进行复杂繁琐的配置，只需要安装一个浏览器就能安全使用这个大数据应用平台，本网站的工作原理和前后端程序流转及业务流转都在上图 3-3 中具体体现出。

3.4 用例分析

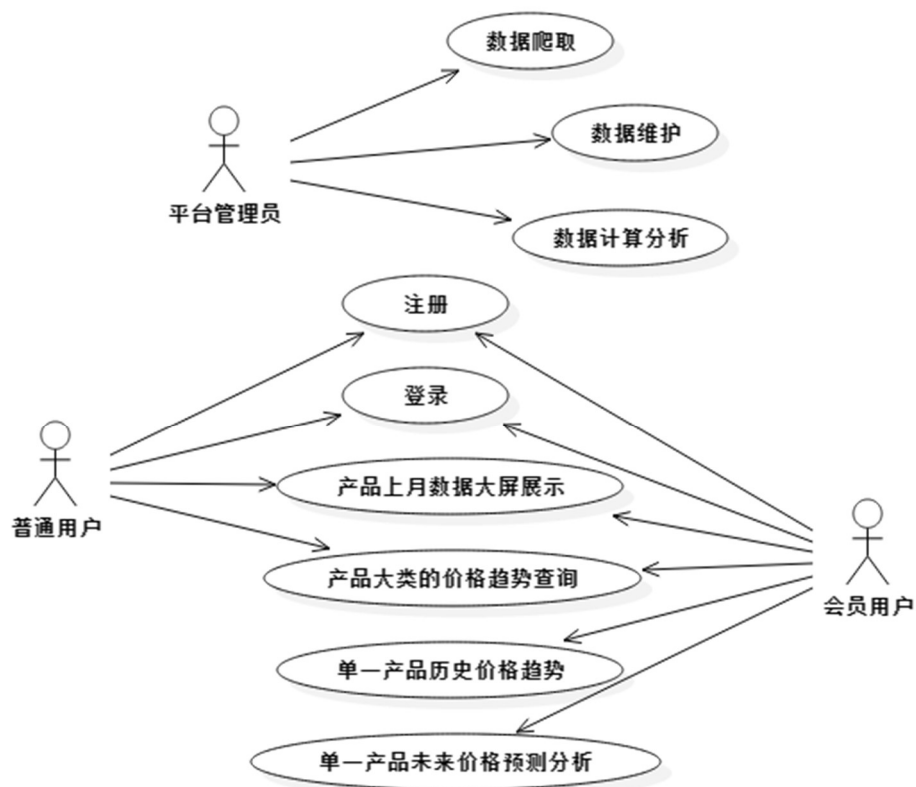


图 3-4 用例分析

本应用平台的用例图如图 3-4 所示。平台管理员主要负责系统的数据维护，数据爬取以及数据的计算分析和权限管理等工作。普通用户只有注册、登录、查看产品上月数据大屏展示和产品大类的价格趋势查询等；会员用户除了拥有普通用户的功能以外，还拥有单一产品历史价格趋势分析功能和单一产品未来价格预测分析等功能。本应用平台的主要核心功能为：产品未来价格预测和数据爬取等。

4 系统设计

4.1 非功能性需求

一个系统或者应用平台仅仅有着强大的功能和美观的界面并不足以称作是一个优秀的系统或应用平台。没有优良的用户人机交互体验、流畅的用户操作和系统或应用平台平稳的运行环境作为基础，系统和应用平台的优越性将会毫无体验，用户自然也不会对你的产品产生兴趣。目前软件免费化已是市场趋势，想要让自己开发的软件走入平常百姓家，系统和应用平台的优越性自然少不了，本应用平台就实现了以下非功能性需求：

- （1）所有功能模块响应时间极短。虽然系统本身的数据量比较大，但是经过大数据技术本身的优越性和对数据库进行了缓存读写，查数时间可以缩短到 3 秒以内。
- （2）视图层和逻辑层代码模块分离，互不干扰、互不影响，对后期的系统功能扩展和平台维护提供便利。
- （3）数据分页操作，极大方便了用户查询的定位。
- （4）Spring 注解自动注入，简化配置，高效开发。
- （5）数据单独存放，定时备份，避免了数据视图不存在。
- （6）系统界面科技感强，操作简便，适合各个年龄段人群使用。。

4.2 功能性需求

农产品流通和数据挖掘应用平台的核心模块为数据爬取模块和产品未来价格预测分析模块，除此之外还有登录模块，注册模块，大数据汇总模块，商品查询模块等。详细分析如下：

4.2.1 数据爬取

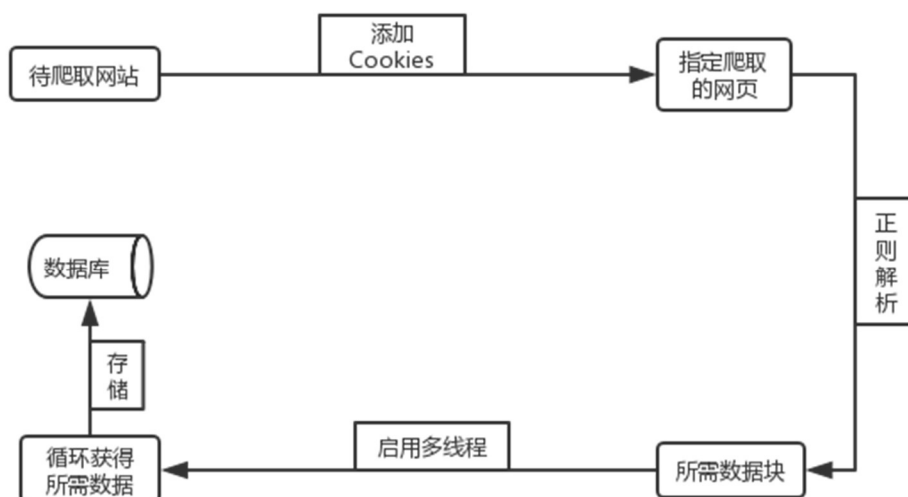


图 4-1 用例分析

本系统所拥有的所有数据都是通过互联网对特定网站进行爬取获得，所使用的技术是比较基础的 Java 常用爬虫技术，具体原理如上图 4-1 所示。首先需要前期调研进行爬取网站选取，本文所描述系统的网站是由我的导师所提供的。确定了要爬取的网站之后就要先注册该网站的账号，为了防止该网站的反爬虫机制，需要将登录账号后的 Cookie 信息添加到 HTTP 请求的头部。接着就要对该网站的 url 进行不断请求解析，然后进行对所爬取 html 页面的正则匹配。只有符合我们要求的有效规范数据才会被存储在数据库中。因为数据量比较大，所以在这里使用了多线程并发存储数据。这样不仅保证了爬虫获取数据效率的高效，同时还能有效防止所登录 Cookie 信息的过期而导致数据获取失败。

4.2.2 产品未来价格预测分析

本文所描述的大数据应用平台的产品未来价格预测分析模块所应用的算法是目前市场上比较成熟的二次指数平滑算法，下面就是该算法的基本公式。

$$S_t^{(1)} = \alpha Y_t + (1 - \alpha) S_{t-1}^{(1)}$$

$$S_t^{(2)} = \alpha S_t^{(1)} + (1 - \alpha) S_{t-1}^{(2)}$$

指数平滑法对实际序列具有平滑效果，加权系数（平滑系数） α 越小，平滑效果越强，但对实际数据变化的响应越迟缓^[20]。在线性变化的实时序列之中，随着顺序偏差指数（阿尔法系数为平滑）的柔性加权值的增加，但当时间序列的线性趋势与第一个平滑指数有很大差异时， N 表示 D 偏差线性趋势的方向。 e 延迟曲线根据的是第一和第二个指数的柔性指数，并预计模型预测将是准确和客观的。

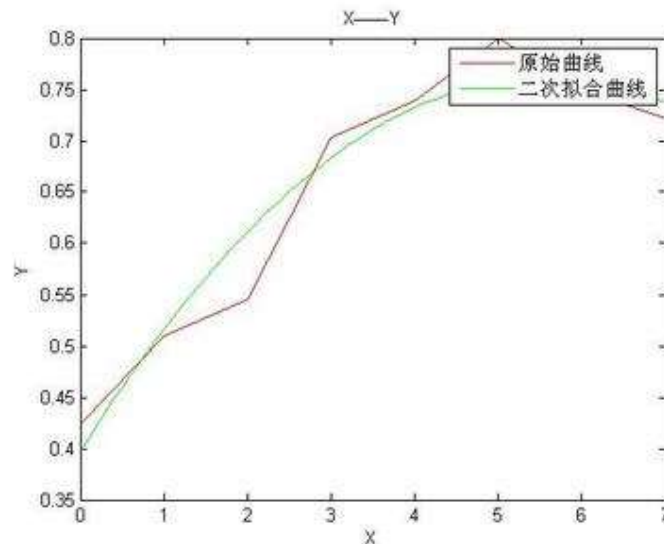


图 4-2 二次指数平滑算法原理图^[20]

4.2.3 大数据汇总

本文所描述的大数据应用平台的大部分功能都是对大数据的图表展示，这就离不开百度的开源项目技术——ECharts。百度开发的 ECharts（Enterprise Charts）拥有诸多特性：它提供了丰富

的可视化图标类型：柱状图、折线图、散点图、饼图等自定义个性化图表和混合图表；能满足用户动态数据实时传输；前端渲染的效果可以自定义展示；最主要的是兼容当前所有主流浏览器，这一特性将很好地满足本系统在前期运营的大力推广，用户不用额外安装任何软件，只需要一个域名就可以在任何地方访问，方便用户在买菜时实时比对商品价格，对本系统无疑是锦上添花。

4.2.4 商品查询

对于本系统的查询功能，在刚开始的需求设计仅仅考虑了每个查询接口的具体设计，但在实际开发中发现这样操作的话最后的接口数目会超过 100 个。为了避免代码的冗余和功能的重复，本人在系统完成后后期进行了代码重构，将所有查询接口整合到一个查询接口，所有入参被写成一个类，这样就能很好地对接口进行复用，优化代码效率。并且，当查询单一产品的时候用 Mybatis 和 Java 进行重构产品索引方法，使得产品搜索索引能在极短时间内进行响应。最终，商品查询的接口方法可以完美兼容任何个数和类型输入参数的查询请求。

Mybatis 实现树状索引方案：通过 ResultMap 可以实现将数据的查询结果映射为复杂类型的 pojo，在这里我将这个结果转换成了树的数据类型。

Java 实现树状索引方案：Java 实现主要通过 Java8 的 Stream 流式编程中的 map 和 collect 方法，具体原理不再赘述，实例如下：

```
nameValue.stream()
    .map(Product::getSizeType)
    .map(sizeType -> new MergeEnumProduct(ProductConstants.SIZE_TYPE, sizeType))
    .collect(Collectors.toCollection(LinkedList::new));
```

4.2.5 登录和注册

像普通系统的设计一样，本系统也拥有注册和登录的界面，注册和登录的合法性判定也和其他系统相类似。但是本系统追求以动态界面吸引刺激用户注册登录使用，所以主要从这两个界面的简约、动态特效来设计。另外，为了大数据展示效果达到最佳，本系统的导航栏也进行了可伸缩滑动设计。

4.3 数据库概念结构设计

本系统的数据库部分采用 PowerDesigner 和 Navicat Premium 进行建模，在建立数据模型之前绘制了数据 ER 图。

数据 ER 图：基于本人的思维方式，我在数据库建表前进行了思维可视化，其数据关系具体如图 4-3 所示。

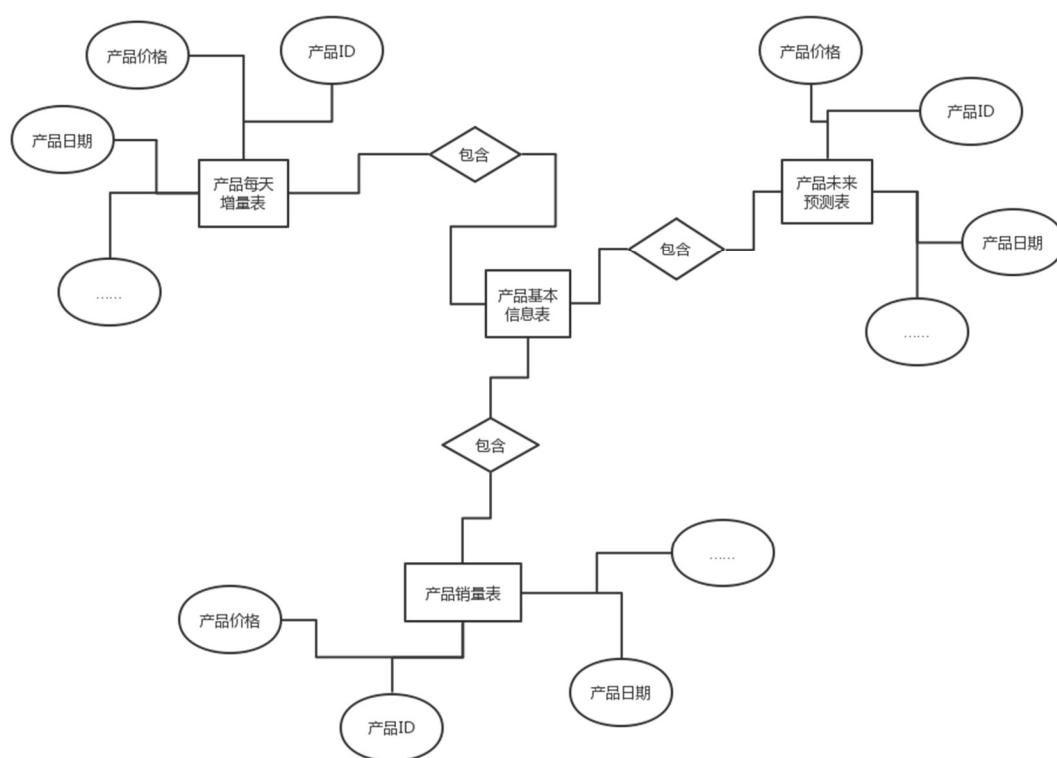


图 4-3 数据 ER 图

4.4 数据库的逻辑结构设计

由于本系统的业务性并不十分复杂，主要建立了 5 张数据表，表之间相互独立，表内有唯一的主键，具体表内字段属性值和字段类型如图 4-4 所示。

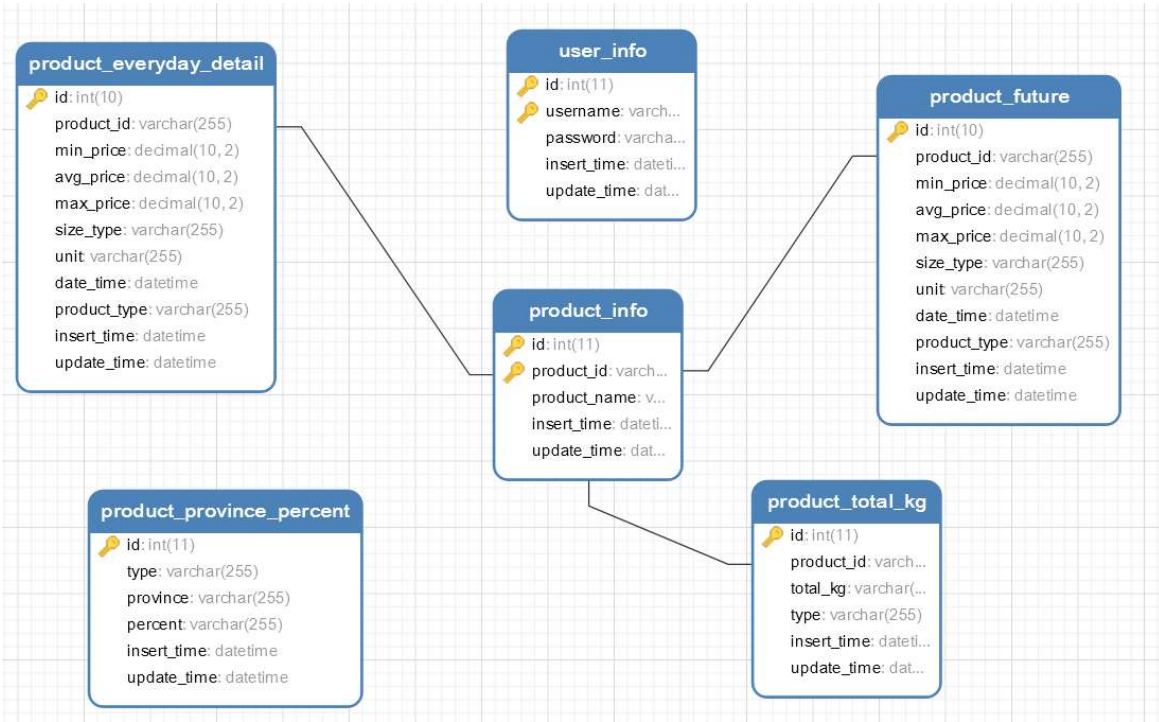


图 4-4 数据库逻辑结构图

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
PRODUCT Everyday DETAIL	主要存储产品每天价格行情明细
PRODUCT FUTURE	主要存储产品未来七天的价格行情明细
PRODUCT INFO	主要存储产品的基本信息
USER INFO	主要存储用户的信息明细和权限配置
PRODUCT PROVINCE PERCENT	主要存储各类产品在各个省市的流通明细
PRODUCT TOTAL KG	主要存储各类热门产品信息明细

4.5 业务代码执行流程

当用户在前端界面点击本应用平台的某个功能按钮，如用户点击了“产品价格预测”模块，具体如图 4-5 所示。首先 SSM 框架进行前端的所有请求拦截，这一步主要在 Controller 中实现，然后业务流程跳转到业务逻辑层，进行各种逻辑判断和工具类方法的调用，最后通过 Mybatis 进行与数据库的交互，进行数据分析预测。业务部分流程具体详见本文 3.3.2 的数据流程图部分，具体的核心项目方法类详见图 4-6。

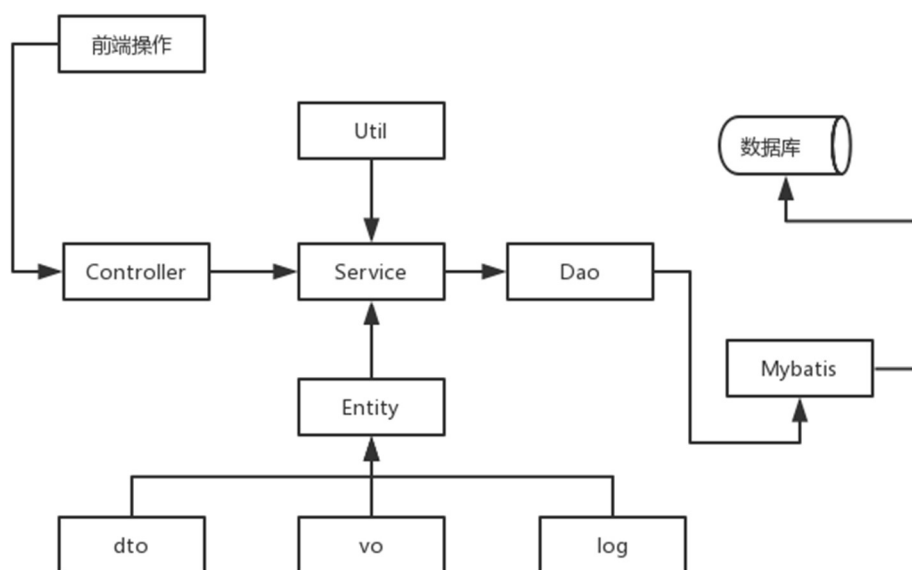


图 4-5 代码执行流程图

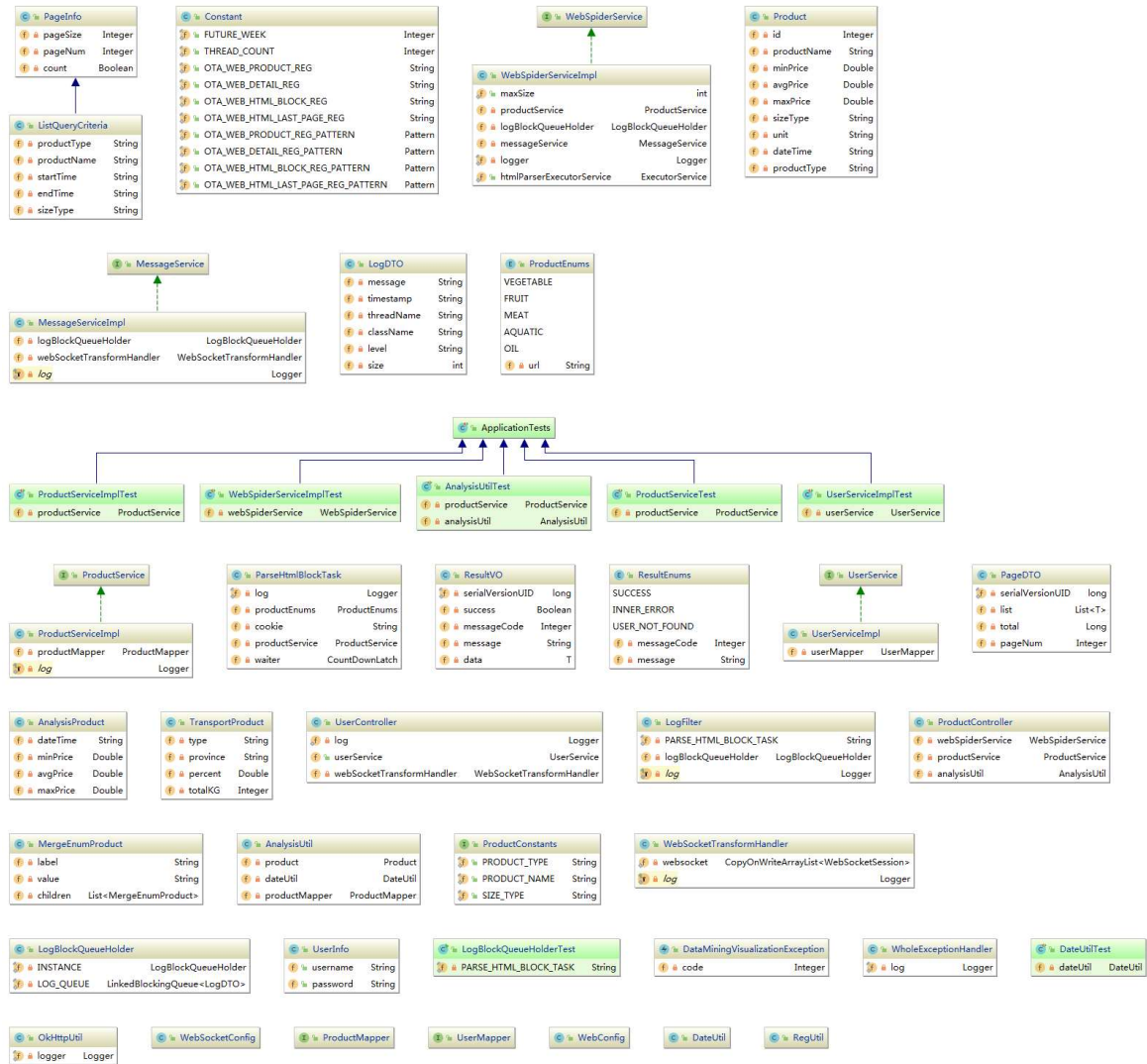


图 4-6 项目核心方法类图

4.6 项目文件结构

本项目采用前后端分离的技术，后端基于 Maven 进行自动化搭建，其具体的项目文件结构如图 4-7 所示。前端基于 NPM 进行 React 框架的搭建，其具体的项目文件结构如图 4-8 所示。

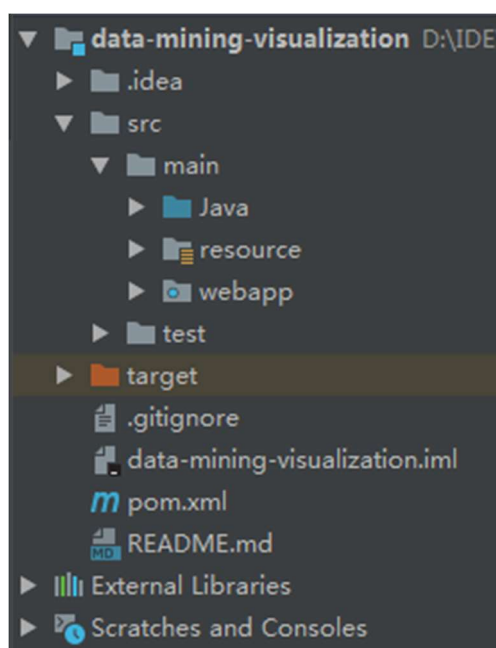


图 4-7 项目文件结构图（后端）

本系统后端的所有主要核心代码都在 Java 文件夹中，资源配置在 resource 文件夹中，所有的测试用例和方法在 test 文件夹中。使用 Maven^[21]最大的好处就是不需要进行 jar 包的单独下载和引入，本系统所有的 jar 包都在 pom.xml 文件中，管理规范，简便开发步骤和提高开发时效。

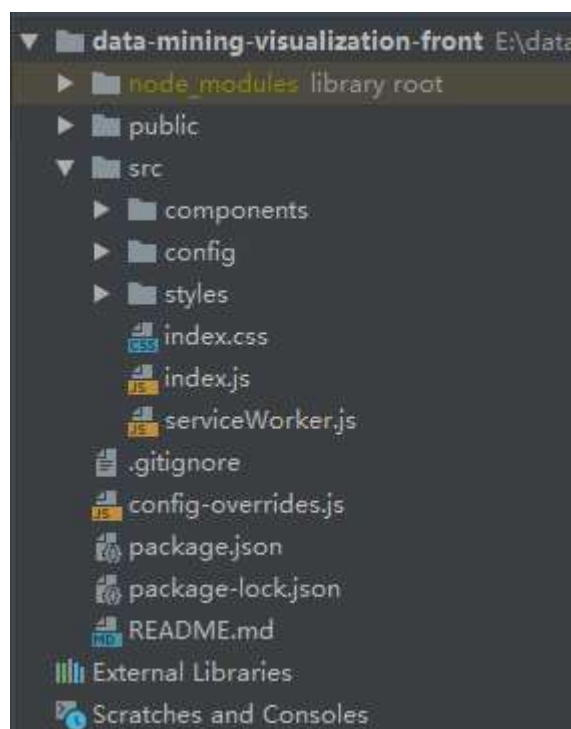


图 4-8 项目文件结构图（前端）

本系统前端的主要核心代码在 components 文件夹中，所有的依赖包都通过 NPM 进行安装引入在 node_modules 文件夹中，package.json 文件中存储了各种依赖包的版本，统一管理方便后期版本升级等。具体项目文件结构如图 4-9 所示。

Java 文件夹主要存放了所编写的后端核心代码，com.zafu.nichang 为本人的包名，config 中编写了主要的全局配置文件，比如跨域请求和 Websocket；constants 文件夹主要存放本项目的一些常量的申明；controller 中编写了本项目的所有接口，并进行不同 URL 请求的拦截；entity 中是一些实体对象的一些扩展，在这里统一规范了返回前端请求数据的主要结构；enums 进行全局的一些枚举值的申明；exception 对本项目的所有异常进行了统一规范的处理；filter 主要进行前后端日志格式的交互规范和本地后端控制台日志的保存存储；handler 中的主要模块是使用 Spring 中的 Websocket 功能，搭配后端日志进行实时前端推送消息；mapper 主要编写 Mybatis 和数据库交互的接口方法；model 申明了本项目中的所有实体类、抽象类；service 是本项目的主要逻辑层，包含各种接口方法和其实例函数；util 主要进行数据爬取模块的编写，解析网页、多线程高并发进行爬取等操作。本项目后端核心代码的文件结构如图 4-9 所示。

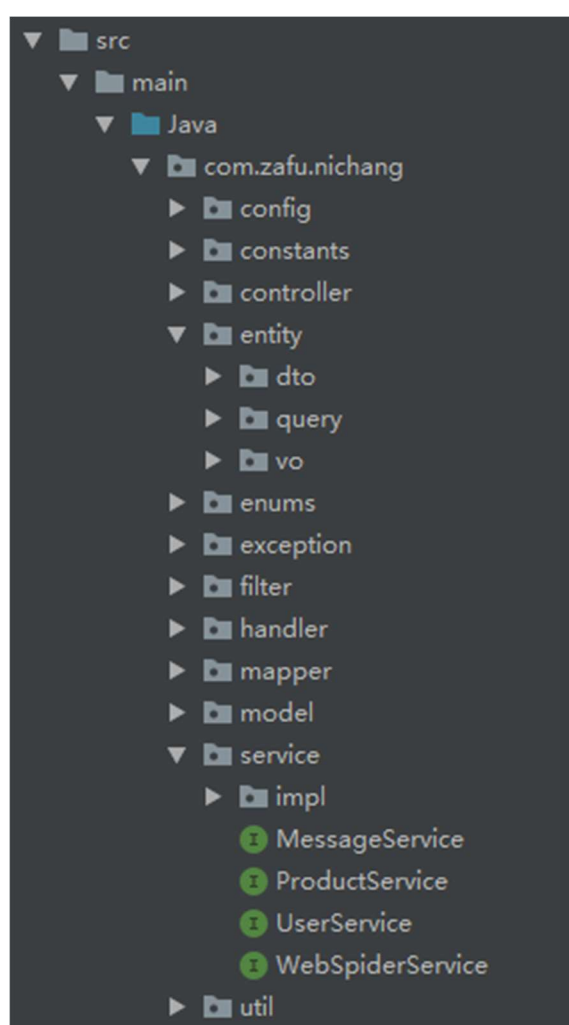


图 4-9 核心代码目录结构图（后端）

本项目的前端核心代码目录结构主要如图 4-10 所示，主要在 components 文件夹中进行编写各个界面；在 config 文件夹中主要进行全局的 IP 和 URL 配置和后端返回 data 数据格式的处理。

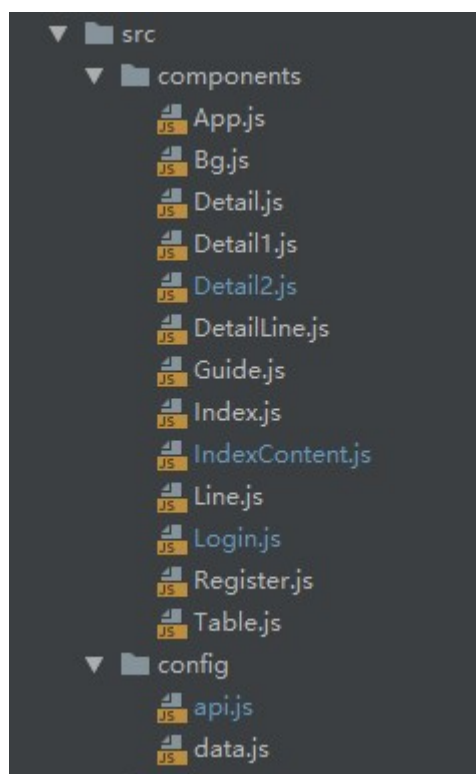


图 4-10 核心代码目录结构图（前端）

4.7 项目命名规范

本项目后端代码参考《阿里巴巴 Java 开发手册》进行标准化格式规范编码^[22]。前端采用标准化前端编码规范：HTML 规范、CSS 规范、JavaScript 规范以及各类文件命名规范。

5 系统实现

5.1 用户模块

本应用平台的主界面主要由三部分组成，标题部分、动态背景以及主要主体。其中动态背景是通过构图函数和时间延时函数进行组合所构建出来的。最终形成一个从左下方往右上方在一定范围内随机速度运行的动态背景壁纸。

5.1.1 登录界面

系统的登录界面的主体是两个输入框、保持登录按钮、忘记密码链接和登录注册按钮。输入正确的用户名和密码，点击“登录”按钮后，后台根据登录用户的用户名密码进行登录权限认证，认证成功后跳转到该用户所具有的功能模块，否则登录失败，显示登录失败具体信息。目前，为了应用平台的推广和使用，后台数据库默认添加所有新注册用户均为管理员用户。具体如图 5-1 所示。

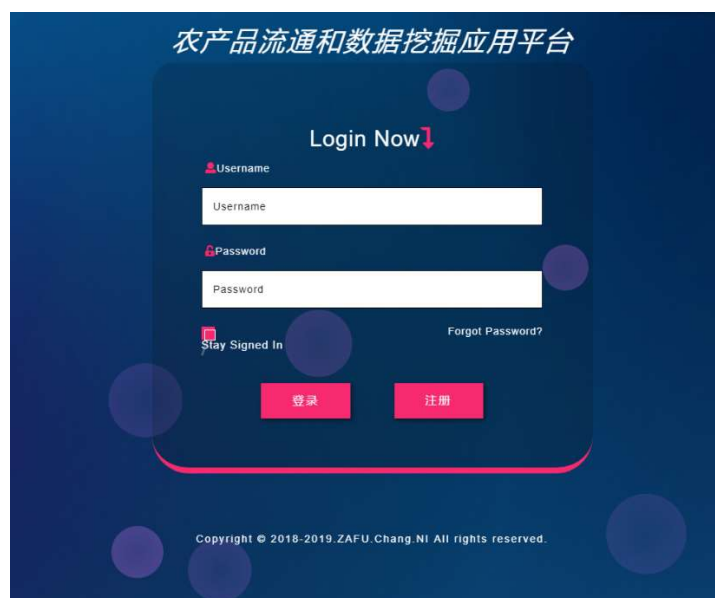


图 5-1 用户登录界面图

5.1.2 注册界面

在登录界面点击“注册”按钮即可跳转到新用户注册界面。用户输入自己设定的用户名密码后进行前端的非法字符判定，判定为合法后请求到后端进行用户名的合法性判定，这里主要进行用户名是否被占用等。验证通过后跳转到登录界面，否则让该用户重新注册。具体界面详情见图 5-2 所示。

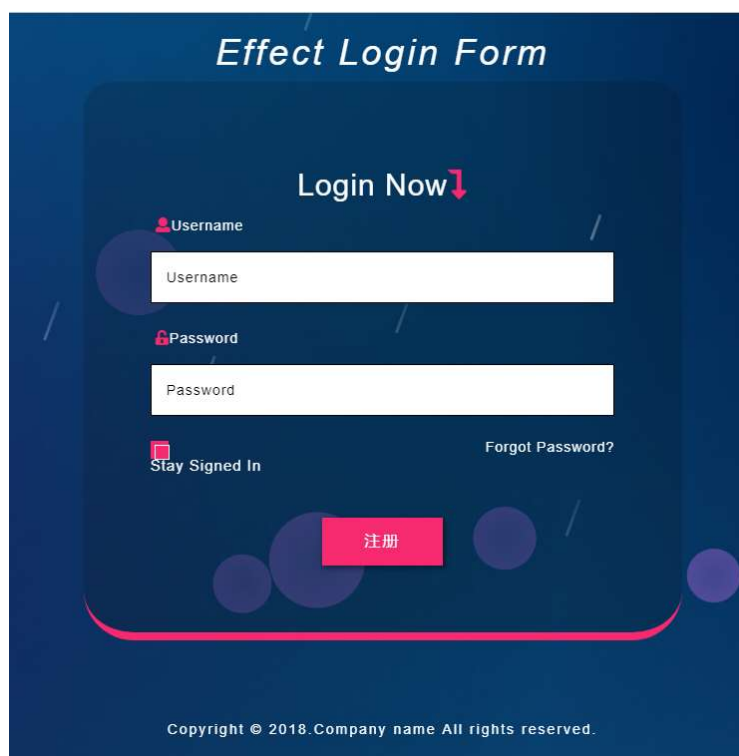


图 5-2 用户注册界面图

5.1.3 导航栏

在用户成功登录系统后，在本系统的左侧有一个可以伸缩的导航栏。这个可伸缩导航栏既能满足全屏首页界面的美观，又能满足用户正常随意跳转本系统的任一功能模块。具体界面详情见图 5-3 所示。



图 5-3 应用平台导航栏界面图

5.2 首页模块

本模块页面如图 5-4 所示。屏幕正上方可清晰看到本项目的名字《农产品流通可视化和数据挖掘平台》，首页主要由 8 个小模块组成。屏幕正中间是一张可以缩放的中国地图，其主要数据展示的是上个月以北京为例的农产品分销全国在各地的明细占比，弹框每隔一定时间进行跳转，能让用户更为直观地看出各地的供给需求占比。其余七个模块将在下方详细介绍。



图 5-4 农产品大数据应用平台首页界面图

5.2.1 上月统计

左上方导航栏右侧是上月销量总计图表(注:本项目中的数据都是以北京为产地的试点数据,并不是全国总数据)。主要展示蔬菜类、水果类、肉禽蛋类、淡水鱼类和粮油类的上月销量。详情模块如图 5-5 所示。



图 5-5 各类产品上月销量图

5.2.2 农产品分类统计

在“上月统计”模块的下方是“农产品分类统计”模块: 各类农产品的销量比重扇形图。鼠标经过扇形图的具体数据区域会变大突出, 并弹出提示框进行数值可视化展示。具体模块效果图参见图 5-6。

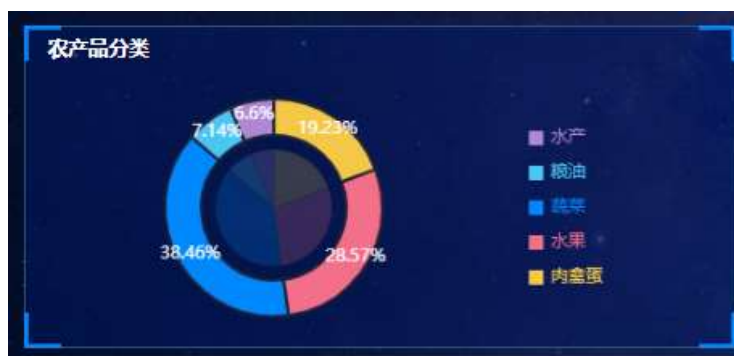


图 5-6 农产品分类扇形图

5.2.3 农产品明细统计

“农产品明细统计”模块由 5 个小模块组成，分别为 5 类产品的具体明细数据可视化表达。每隔明细数据模块的提示框和“流通模块”的提示框一样，每隔一定时间自动切换，方便用户查看。具体模块如图 5-7 所示（以肉禽蛋类数据为例）。

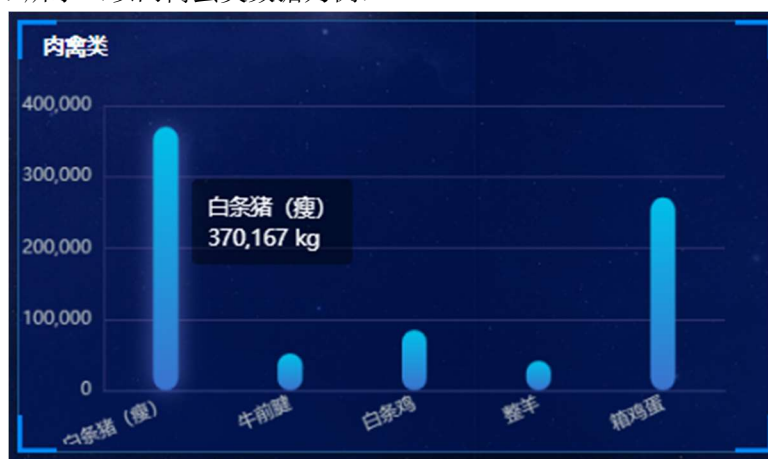


图 5-7 热门产品分类显示模块图（肉禽蛋）

5.3 产品价格详情模块

进一步通过点击伸缩式导航栏进行交互，选择“产品价格详情”即可进入该模块。该模块主要进行蔬菜、水果、肉禽蛋、水产和粮油等 5 大产品的每天价格历史明细。界面整洁简单，生动形象，适合绝大多数用户便捷操作。具体界面如图 5-8 显示。



图 5-8 产品价格详情页

5.3.1 图片按钮分类

这一模块用到了阿里巴巴开源的 Ant Design，图片按钮背景透明，简单形象，方便不识字的小孩子群体和文化水平程度低的老年人使用。点击对应图即可切换下方图表中的数据可视化展示。具体界面如图 5-9 所示。



图 5-9 分类按钮模块

5.3.2 价格比较图

本文所有的可视化图表大部分都采用百度开源的 ECharts 控件^[23]，能比较直观的显示每种产品的最高价，最低价和平均价，方便用户挑选性价比最高的产品。具体图表情况见图 5-10。



图 5-10 价格比较图

5.3.3 分页表格

“分页表格”模块详情记录了每种产品的历史价格，按照日期降序分布，下方展示每种数据的总数据量，并能支持用户根据自己喜好每页展示多少条数据，下方表格中的数据发生变化后，为了方便用户直观体现表格中数据的变化，上方的价格比较图也将实时进行数据展示。具体的模块展示图如图 5-11 所示。

品名	最低价	平均价	最高价	规格	单位	发布日期
箱装菠萝（上六下六）	1	1.15	1.3	其他类	斤	2019-01-31
圣女果（千禧）	4.1	5.35	6.6	其他类	斤	2019-01-31
黑枣	3	5.5	8	其他类	斤	2019-01-31
冬枣	15	22.5	30	其他类	斤	2019-01-31
红颜草莓	10	12.5	15	其他类	斤	2019-01-31
丹东九九草莓	14	16	18	其他类	斤	2019-01-31
甜宝(奶油)草莓	13	15.5	18	其他类	斤	2019-01-31
白雪公主草莓	30	32.5	35	其他类	斤	2019-01-31
巧克力草莓	18.5	19.75	21	其他类	斤	2019-01-31
甜查理草莓	11	11.5	12	其他类	斤	2019-01-31

当前第 1 到10 条，总共42246 条记录

< 1 2 3 4 5 ... 4225 > 10 跳至 页

图 5-11 产品分页展示图

5.4 未来数据挖掘模块

继续点击导航栏，就能进入本项目最为核心的模块——“未来数据预测”模块。具体的模块页面如图 5-12 所示。其核心的算法在本文的 4.2.2 中已经进行了详细阐述，在此不再赘述。下文会介绍具体的功能。

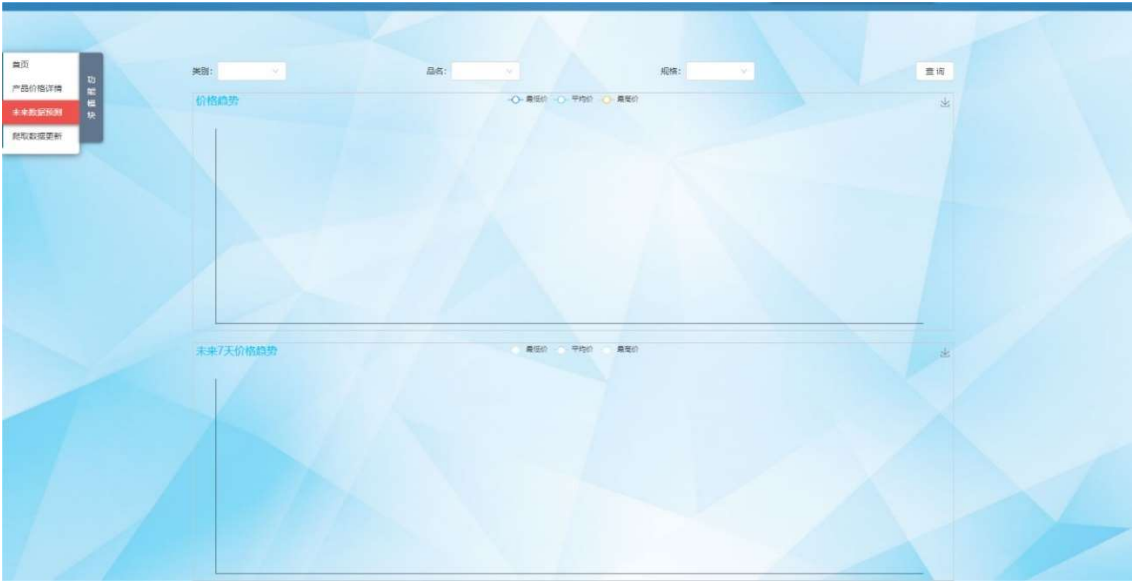


图 5-12 未来数据预测页面总览图

5.4.1 树状索引搜索查询

本页面上方的三个产品选择框之间有着密不可分的联动关系。三个选择框有着依次递进的层级关系，当用户首先选择了产品的类别（如图 5-13 所示）才能进行选择产品的具体名称。



图 5-13 产品类别选择列表

当用户选择了产品的类别，产品名称的列表将会显示具体的名称。以“肉类”为例，用户选择了肉类后，产品名称列表将显示具体的肉类名称，而不会显示其他类别产品下的具体产品名称。详情见图 5-14 所示。



图 5-14 产品名称选择列表

同理，产品规格类别列表也是如此，详情见图 5-15 所示。这样做的好处可以精确筛选出具体的某一种产品。避免出现因产品名称相同但规格不同而导致产品的未来价格预测分析出现极大误差。比如（品名：鲫鱼，规格：<500g；品名：鲫鱼，规格：500-750g），去过菜场或者农贸市场的用户都知道，显然，两者的价格肯定不一致。通过联动树状索引的查询，可以完美解决这一问题。



图 5-15 产品规格选择列表

5.4.2 历史价格走势与未来数据挖掘

本模块经过网站后端的实时运算，在第一张图表中展示该产品在过去 90 天的历史价格数据的可视化展示。之所以选择过去 90 天，是因为农产品不像是一般商品，它有着很强的季节性变化，所以分析数据的原始数据样本不宜超过一个季度，但是数据量太少也会产生数据分析的偶然性，所以本人就选择一个季度作为原始的数据分析样本数据。在第二张图表中展示该产品的未来 7 天价格趋势变化。这两张图表都包含某一产品每天的最高价格、最低价格和平均价格，方便用户了解市场行情。具体详情见图 5-16 所示。



图 5-16 未来数据分析预测图

5.5 爬取数据更新模块

此模块是为了管理员进行数据的定期维护和数据更新。具体的详情页面如图 5-17 所示。

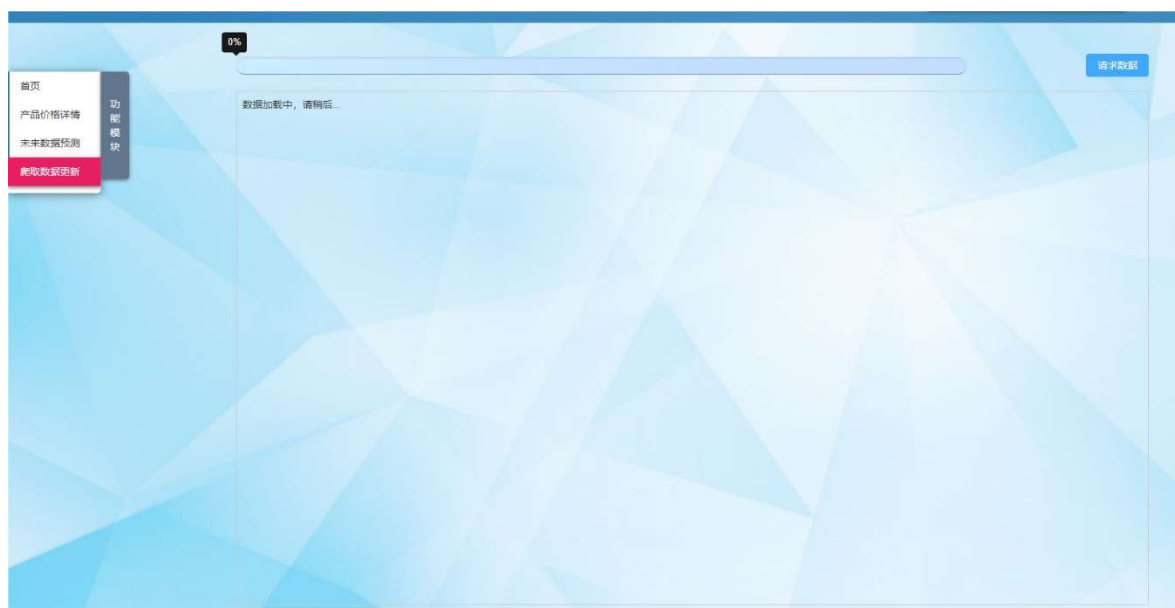


图 5-17 数据爬取模块总览图

5.5.1 爬取进度交互

此模块为了让用户（主要指管理员）进行数据更新的可视化展示。按下进度条后的“请求数据”按钮即可开始爬取数据，当数据爬取更新完成后进度条会增长至“100%”。具体如图 5-18 所示。



图 5-18 爬取进度交互

5.5.2 爬取信息交互

本模块主要通过 Websocket 进行实时交互，当数据爬取开始时，显示“数据加载中，请稍后...”开始提示语。此时页面会静止一段时间进行后台数据爬取的准备。每条语句会显示具体爬取的页面信息，比如：页面总页数、当前爬取页数、是否需要过滤爬取当前页等等。爬去完毕后会显示“数据请求完毕！”结束提示语。具体的页面如图 5-19 所示。

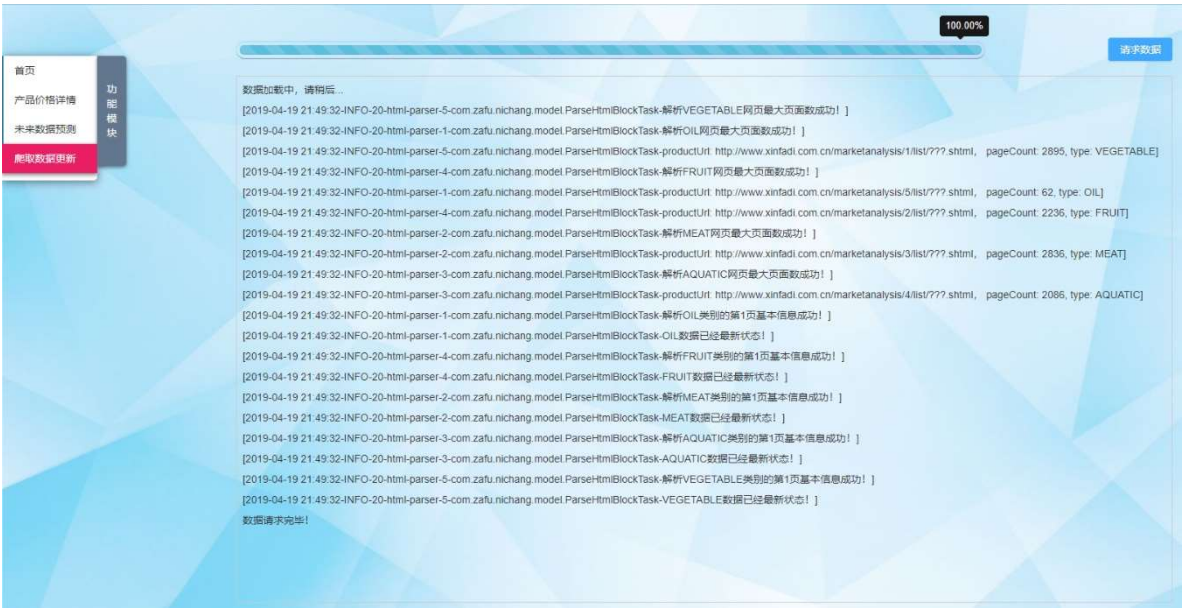


图 5-19 爬取信息交互

6 总结

这次的毕业课题研究《产品流通的可视化表达与数据挖掘》总历时五个月有余，通过这次的毕业设计制作和毕业论文撰写我对一个项目的从无到有、从零开始增长了许多经验。在这五个月内，我不仅只是在研究课题项目，还在某互联网软件公司进行实习锻炼。通过实习，我将公司的项目管理经验和自身相结合，将毕业课题研究当作一次实践。事实证明，这次实践的效果还是不错的。对于自身，在研究项目课题中学习新的技术知识技能；对于项目课题，我从头到尾充当了项目需求者、产品人员、项目架构者、项目开发者、项目测试者和项目管理者等多个角色，对于每种岗位的只能有了基本了解，对于毕业以后进入公司发展做了良好的铺垫。同时，这次项目课题的研究也让我明确了未来发展的方向——Java 开发工程师。当然，这次项目研究课题也让我了解了当前中国农民生活不易，“卖菜难，买菜贵”的社会民生问题依然没有得到良好的解决。但是，我相信通过我国对智慧城市的大力发展建设，这个问题一定会得到良好的解决。

然后就是对本项目的一些经验总结。通过本次的项目课题研究，我充分认识到一个项目的规范真的特别重要。其中就包括了项目代码编码的规范和项目的管理规范。在这一点方面，目前身为实习生的我体会很深。一个编码规范良好的项目（主要是在变量、类名的命名规则和具体函数方法中的注释规范）可以让像我这种新人开发者能在短时间内快速地熟悉这个项目，并进行功能的扩展开发和已开发功能的基本维护。其次，在互联网行业，互联网技术总是在不断飞速更新的，我们不能满足于目前已经比较成熟的专业技术。或许在目前，已经存在某种新的技术（既能满足眼前成熟技术的所有优势，又能将这一成熟技术的所有劣势消灭），但只是因为公司升级开发环境技术版本的成本过高而没有使用。如果我们不主动的去学习，那么我们终究会被这个社会所淘汰。“人无常师,学无止境”，这句话是我父母从小教育我对学习知识的态度。技术总在更新换代，我们只有活到老、学到老，才能适应当今互联网时代的新潮流。

最后，通过这次项目课题的研究，我也对自己未来发展的方向有了更好的认知。既然选择走 Java 大数据开发这一条路线，那么就不要再退缩。各行各业都有自己的不容易，也许在刚开始的阶段会遇到这样那样的困难和瓶颈，但是没关系，咬咬牙就会过去。毕竟在互联网行业，项目经验真的很重要，在工作的时候要好好总结项目经验并注重新知识技能的积累。未来，加油！

参考文献

- [1] 张引, 陈敏, 廖小飞. 大数据应用的现状与展望[J]. 计算机研究与发展, 2013, 50(S2):216-233.
- [2] 范洪春 译. Web Component 实战:探索 PolymerJS、Mozilla Brick、Bosonic 与 ReactJS 框架[M]. 电子工业出版社, 2015:32-72.
- [3] Zakas N, 泽卡斯, Zakas, et al. JavaScript 高级程序设计[M]. 人民邮电出版社, 2010:78-337.
- [4] 王子毅, 张春海. 基于 ECharts 的数据可视化分析组件设计实现[J]. 微型机与应用, 2016, 35(14):46-48.
- [5] Lapsley I. THE NPM AGENDA: BACK TO THE FUTURE[J]. Financial Accountability & Management, 2010, 24(1):77-96.
- [6] 李洋. SSM 框架在 Web 应用开发中的设计与实现[J]. 计算机技术与发展, 2016, 26(12):190-194.
- [7] Mitchell J G, Gibbons J, Hamilton G, et al. An Overview of the Spring System.[C]// Compcon Spring 94, Digest of Papers. 2002:1-3.
- [8] 沃尔斯. Spring 实战: 第 4 版[M]. 人民邮电出版社, 2016:457-479.
- [9] 朱俭. 面向方面编程(AOP)介绍[J]. 计算机工程, 2004, 30(b12):170-172.
- [10] Ragonis N, Benari M. A long-term investigation of the comprehension of OOP concepts by novices[J]. Computer Science Education, 2005, 15(3):203-221.
- [11] 黎永良, 崔杜武. MVC 设计模式的改进与应用[J]. 计算机工程, 2005, 31(9):96-97.
- [12] 荣艳冬. 关于 Mybatis 持久层框架的应用研究[J]. 信息安全与技术, 2015, 6(12):86-88.
- [13] Eckel B . Thinking in Java, 4th Edition[M]// Thinking in Java (4th Edition). Prentice Hall PTR, 机械工业出版社, 2005:123-256.
- [14] 连洁. 论函数式编程在 JAVA 程序中的应用[J]. 电脑知识与技术, 2015, 11(2X):99-100.
- [15] Raoul-Gabriel Urma , Mario Fusco , Alan Mycroft. Java8 实战[M]. 北京: 人民邮电出版社, 2016:34-110.
- [16] Bloch J . EFFECTIVE JAVA (2ND EDITION)[J]. 2008:72-99.
- [17] Di Glacomo M . MySQL: Lessons Learned on a Digital Library[J]. IEEE Software, 2005, 22(3):10-13.
- [18] Gorodov, Evgeniy Yur'evich, Gubarev, Vasiliy Vasil'evich. Analytical Review of Data Visualization Methods in Application to Big Data[J]. Journal of Electrical and Computer Engineering, 2013:1-7.
- [19] 杜小刚, 李舟军. J2EE Web 开发框架体系结构[J]. 计算机科学, 2006, 33(8):236-239.
- [20] 王慈光. 二次指数平滑法中确定初始值的简便方法[J]. 西南交通大学学报, 2004, 39(3):269-271.
- [21] Goldman M, Katz S. MAVEN: Modular Aspect Verification[J]. 2007:1-5.
- [22] Alibaba Group. 阿里巴巴 Java 开发手册(正式版)[M]. 阿里巴巴集团出版, 2018:1-34.
- [23] Gupta P , Kahng A B , Kasibhatla A , et al. Eyecharts: constructive benchmarking of gate sizing heuristics[C]// Design Automation Conference. IEEE, 2010:2-12.

致谢

首先，我要感谢我的导师，是陈文辉导师在我最迷茫的时候给我有效的建议。在我不知道自己该研究什么课题的时候，是陈文辉老师启发我结合自身的工作和专业技能进行产品数据挖掘课题的研究。其次，我要感谢学校课题的开题评审小组能给我研究这个课题的机会。我也衷心地希望，在未来的某一天，我的这个研究成果能为我国各行业的大数据分析带来新的思路和想法。当然，我还要感谢在背后为我提供帮助的兄弟朋友们。如果没有他们给我提供一些技能的快速学习方法，能让我在短时间内熟悉新的技术技能，我的项目课题研究成果就不会这么迅速地展现给大家。我也要感谢我的父母，是他们在课题陷入困境的时候给我新的研究灵感。最后，我也要感谢那个为了未来，一直在奋斗的自己！