****

**本科毕业设计**

**题 目**： 基于Hadoop的健康饮食推荐系统的设计与实现

|  |  |
| --- | --- |
| 学 院: | 计算机信息工程学院 |
| 专 业: | 数据科学与大数据技术 |
| 姓 名: | 张媛 |
| 学 号: | **2021090840138** |
| 指导教师: | 张丽梅讲师 |

2025年 4 月 30 日

**毕业论文（设计）学术承诺**

本人郑重承诺：所呈交的毕业论文（设计）是我个人在导师指导下进行的研究工作及取得的研究成果。除了文中特别加以标注和致谢的地方外，论文中不存在抄袭情况，论文中不包含其他人已经发表的研究成果，也不包含他人或其他教学机构取得研究成果。

作者签名： 日 期：

**关于毕业论文（设计）使用授权的声明**

本人在指导老师指导下所完成的论文（设计）及相关资料（包括图纸、试验记录、原始数据、实物照片、图片、摄像录像、设计手稿等），知识产权归属山西工商学院。本人授权山西工商学院可以将本毕业论文（设计）的全部或部分内容编入有关数据库进行检索，可以采用任何复制手段保存和汇编本毕业论文（设计）。如果发表相关成果，一定征得指导教师同意，且第一署名单位为山西工商学院。本人离校后使用毕业论文（设计）或与该论文（设计）直接相关的学术论文或成果时，第一署名单位仍然为山西工商学院。

作者签名： 指导教师签名：

日 期： 日 期：

摘 要

现在，人们越来越重视健康管理。但是，过去的饮食推荐系统有不少问题。比如数据量不够大，算法也不能及时更新，所以很难满足每个人不同的健康饮食需求。 我们开发的这个系统，有多个功能模块。它能让用户注册登录，管理个人信息，还能记录饮食情况。通过这些功能，系统可以给用户推荐适合他们的饮食。系统还有管理员。管理员能管理用户信息，审核饮食记录，还能优化推荐算法。同时，系统用Python爬虫技术收集最新的饮食数据，保证推荐内容既准确又及时。 这个系统用了很多技术，像Spring Boot、Vue.js、MySQL、Element UI和Spring Data JPA 。通过这些技术，系统搭建起一个功能齐全的网页应用。系统把前端和后端技术结合起来，数据处理效率高，用户用起来也方便。系统用Hadoop集群来存储和处理大量饮食数据，这样就能训练和优化个性化推荐算法。系统采用模块化设计，各个功能分开又能合作，以后扩展新功能或者维护都很方便。用户能用系统找到符合自己健康需求的菜单，管理员也能顺利管理用户信息和审核饮食记录。

**关键字：**健康饮食；Hadoop；个性化推荐；Spring Boot；Python爬虫

Abstract

Nowadays, people pay more and more attention to health management, but there were many problems with the systems of previous dietary recommendations. For example, if the amount of data is not large enough and the algorithm can not be updated in time, it will be difficult to meet all the different needs of a healthy diet. The system we have developed includes several functional modules. It allows users to register and log in, manage personal data and also maintain their eating habits. These functions allow the system to recommend an appropriate diet for consumers. Administrators can manage user data, view nutrition data, and optimize recommended algorithms. At the same time, the system uses indexing technology to collect the latest nutrition data and ensure that the recommended content is accurate and timely. This system uses many techniques such as Spring Boot, MySQL, Element UI and Spring Data JPA. With these technologies, the system builds a fully functional web application. The system combines front and rear technology for efficient data processing and comfort. The system uses clusters to store and process large food data, enabling training and optimization of customized control algorithms. The system has a modular structure where each function is separate but uniform, making it easier to expand or maintain new functions in the future. Users can use the system to find menus that meet their health needs, administrators can seamlessly manage user information and view nutrition information.

**Key Words:** healthy diet; Hadoop; personalized recommendation; Spring Boot; Python web scraping

目 录

[1 绪论 1](#_Toc21258)

[1.1 研究背景 1](#_Toc389)

[1.2 研究目的意义 1](#_Toc25567)

[1.3 国内外研究现状 2](#_Toc14429)

[2 相关技术介绍 4](#_Toc14707)

[2.1 Spring Boot 4](#_Toc9969)

[2.2 Vue.js 4](#_Toc1768)

[2.3 MySQL 4](#_Toc15769)

[2.4 Element UI 4](#_Toc26100)

[2.5 Spring Data JPA 5](#_Toc22283)

[2.6 Hadoop 5](#_Toc5884)

[2.7 Python 爬虫技术 5](#_Toc26743)

[2.8 协同过滤算法 6](#_Toc8012)

[3 系统分析 7](#_Toc7020)

[3.1 系统用例分析 7](#_Toc7649)

[3.2 系统功能需求分析 8](#_Toc26611)

[3.3 非功能性需求分析 11](#_Toc13208)

[3.4 可行性分析 11](#_Toc14475)

[4 系统设计 13](#_Toc448)

[4.1系统功能结构设计图 13](#_Toc12002)

[4.2 系统总体设计 13](#_Toc6760)

[4.3 数据库设计与实现 14](#_Toc6019)

[4.4系统功能模块设计 17](#_Toc22489)

[4.5算法设计 18](#_Toc19413)

[5 系统功能详细实现 20](#_Toc32024)

[5.1 数据爬取 20](#_Toc32023)

[5.2 用户功能实现 20](#_Toc20340)

[5.3 管理员功能实现 25](#_Toc30784)

[6 系统测试 29](#_Toc6494)

[6.1 测试目的 29](#_Toc12787)

[6.2 测试用例 29](#_Toc18715)

[6.3 测试结果 30](#_Toc8924)

[7 总结与展望 32](#_Toc12831)

[7.1 总结 32](#_Toc52)

[7.2 展望 32](#_Toc28166)

[参考文献 33](#_Toc31049)

[致 谢 35](#_Toc647)

# 1 绪论

## 1.1 研究背景

信息技术发展得很快，人们开始越来越关注健康管理和个性化服务。在这样的情况下，大数据技术的应用给健康管理带来了新机会，同时也带来了挑战。Hadoop是一种分布式计算框架，它处理大量数据的能力很强，能有力支持健康饮食推荐系统的设计和开发。 现在大家生活条件变好了，健康意识也提高了，很多人都开始重视吃得健康。大家希望通过科学搭配饮食，来预防疾病，保持身体健康。但是以前的饮食建议通常不够个性化，很难满足每个人不同的需求。所以，基于大数据的个性化健康饮食推荐系统就出现了。 这种系统会收集分析用户的健康数据、生活习惯和饮食喜好等信息，再用先进的算法模型处理这些数据，就能给用户更准确的饮食建议。Hadoop处理数据的能力强，还能扩展，用它存储和分析大量用户数据，效率会更高。而且，Hadoop生态系统里有很多工具和技术，给开发健康饮食推荐系统提供了多种选择和帮助。比如Hive可以用来查询和管理结构化数据，MapReduce适合处理复杂的数据任务。

## 1.2 研究目的意义

大数据技术发展迅速，研究和开发健康饮食推荐系统，成了提升大众健康水平的重要方法。我们开展这项研究，目的是在Hadoop平台上，设计并做出一个高效的健康饮食推荐系统。 这个系统会整合大量的饮食数据和用户信息，能给不同用户提供适合他们的饮食建议。具体来说，我们会研究怎么用分布式计算框架处理大规模数据，让系统反应更快、给出的建议更准确。同时，我们也会深入研究数据挖掘算法的使用，保证推荐结果既科学合理，还能根据用户喜好变化及时调整。 开发健康饮食推荐系统，对改善公众健康很有意义。第一，它能帮助用户了解自己的饮食习惯，给出科学建议，让大家养成健康的生活方式。第二，这个系统可以减少因为不好的饮食习惯导致的慢性病，像肥胖、糖尿病等，减轻医疗系统的压力。第三，依靠Hadoop强大的数据分析能力，这个系统能给企业和政府提供有用的参考，帮助他们制定更好的公共卫生政策。通过整合多种来源的数据，系统提高了个性化服务的质量，也推动了健康科技的发展。

## 1.3 国内外研究现状

1.3.1 国内研究现状

最近几年，国内对健康饮食的研究越来越深入。特别是有了大数据技术的支持，很多研究开始思考，怎样借助先进的计算平台，让健康饮食的个性化推荐做得更好。Hadoop是一种分布式计算框架，它处理大量数据的能力很强，所以受到了很多研究者的关注。 中国农业大学做过一项研究。研究人员通过Hadoop平台分析大量用户的饮食习惯，发现这样能有效提升推荐系统的准确性和效率。这项研究收集了全国超过10万份饮食记录，再结合用户的身体指标和生活习惯，给出了比较准确的饮食建议。

北京大学的研究团队关注的是Hadoop在营养成分分析方面的应用。他们开发了一套基于Hadoop的数据处理系统，用来分析不同食材的营养成分，以及这些成分对人体健康的影响。通过整合和分析全国各类食材的详细营养数据，这个系统能快速给用户提供个性化的饮食搭配方案。 复旦大学的研究人员则探索了Hadoop在预防慢性病方面的作用，尤其是针对糖尿病患者的饮食管理。研究发现，利用Hadoop平台实时监测和分析患者的日常饮食数据，能帮助患者控制血糖，减少出现并发症的风险。 不过，这些研究也存在一些问题。第一，很多研究只是在收集和初步分析数据，没有深入了解用户的真实需求。第二，现有的系统反应不够快，不能马上给用户推荐结果。第三，不同平台之间的数据很难共享和互通，这也影响了系统的广泛使用。 。

1.3.2 国外研究现状

在国外，健康饮食推荐系统的研究和开发已经有了很大的成果。Hadoop作为大数据处理平台，在这个领域也被广泛使用。比如，美国的Nutrium公司用Hadoop技术搭建了一个大型的营养数据处理平台。这个平台能快速处理和分析大量用户的饮食记录，然后给用户提供适合他们的饮食建议。英国的MyFitnessPal则整合了Hadoop生态系统里的多种工具，实现了对大量用户数据的实时处理和分析，让系统的反应速度更快，推荐也更准确。 在欧洲，德国的Max Planck研究所开展了“Food4Me”这个研究项目，目标是做个性化饮食推荐。项目收集了基因组学、代谢组学和环境因素等多方面的数据，再用Hadoop进行存储和计算，希望给不同的人提供更准确的饮食方案。荷兰的Wageningen University & Research在研究中也采用了Hadoop技术，他们重点把农业生产和食品供应链的数据整合到健康饮食推荐系统里，想要实现从农场到餐桌的全程管理和优化。

不过，这些研究也有一些问题。第一，现在的系统大多是分析固定的数据，很难适应用户不断变化的需求。第二，数据隐私和安全是个难题，特别是涉及到个人健康这些敏感信息的时候。第三，虽然不同学科之间有了合作，但在实际应用中还需要加强，尤其是医学、营养学和计算机科学这些学科的融合。

# 2 相关技术介绍

## 2.1 Spring Boot

Spring Boot是一个开发框架，用来创建基于Spring框架的独立应用，而且适用于生产环境。使用这个框架，搭建和开发基于Spring的应用会变得更简单。它有很多现成可用的功能，比如自动配置和起步依赖。 开发者用简单的配置文件，就能管理应用的各种设置。同时，也能快速把各种第三方库和服务添加到应用里。Spring Boot支持多种部署方法，可以用嵌入式服务器，像Tomcat和Jetty。这样一来，应用在不同环境里都能容易地运行。另外，Spring Boot有庞大的社区，还有丰富的文档资料，能给开发者提供有力的帮助。

## 2.2 Vue.js

Vue.js 是一种用于构建用户界面的渐进式 JavaScript 框架，它允许开发者通过声明式的语法编写动态的 Web 界面。Vue 的核心库专注于视图层，易于学习和集成到现有项目中。它采用了虚拟 DOM 技术，确保高效的 UI 更新，同时提供了响应式的数据绑定机制，使开发者能够轻松处理复杂的状态管理。Vue.js 还拥有丰富的生态系统，包括官方提供的路由管理和状态管理工具，进一步增强了其灵活性和可扩展性。对于前端开发人员来说，Vue 提供了一个高效且直观的开发体验。

## 2.3 MySQL

MySQL 是一种关系型数据库管理系统，广泛应用于各类 Web 应用和企业级系统中。它支持 SQL 查询语言，具备高可靠性和高性能的特点。MySQL 以其开源、易用和灵活的特性著称，能够处理大规模数据存储需求。其优化的查询执行引擎和索引机制保证了快速的数据检索和更新操作。此外，MySQL 支持事务处理、外键约束等功能，确保数据的一致性和完整性。通过与多种编程语言的无缝集成，MySQL 成为了众多开发者构建稳定、高效的数据库解决方案的首选。

## 2.4 Element UI

Element UI 是一套为开发者、设计师和产品经理准备的桌面端组件库，基于 Vue.js 开发。它提供了丰富且高质量的 UI 组件，涵盖了表格、表单、导航、布局等多种类型，帮助开发者快速构建美观且功能齐全的用户界面。Element UI 遵循一致的设计规范，确保组件风格统一，提升了用户体验。其组件具有良好的自定义能力，开发者可以根据实际需求调整样式和行为。此外，Element UI 拥有详细的文档和活跃的社区支持，为开发者提供了便捷的学习和使用途径。

## 2.5 Spring Data JPA

Spring Data JPA 是 Spring Data 项目的一部分，旨在简化对持久化数据的访问。它通过 JPA（Java Persistence API）提供了一套简洁的接口和方法，使得开发者无需编写复杂的 CRUD 操作代码即可实现数据的存取。Spring Data JPA 支持基于注解的配置和查询方法，减少了样板代码的数量，提高了开发效率。它还集成了分页、排序、事务管理等功能，满足了多样化的数据操作需求。此外，Spring Data JPA 具有良好的扩展性，可以轻松与其他 Spring 模块协同工作，形成了一个完整的生态系统。

## 2.6 Hadoop

Hadoop是Apache软件基金会开发的分布式计算框架，它的设计目标就是处理大量数据。Hadoop主要有两部分，分别是HDFS分布式文件系统和MapReduce分布式计算模型。 HDFS能存储数据，而且不容易出错。它会把大文件拆成小块，分别存到集群的不同节点上。MapReduce可以把任务分成多个小任务，然后在集群里同时处理，这样就能高效完成批量数据处理工作。 此外，Hadoop还能和很多编程语言、工具一起用，像Pig、Hive和Spark。也正因如此，Hadoop成了大数据处理领域的重要技术。

## 2.7 Python 爬虫技术

Python爬虫技术是用Python编程语言做出自动化程序，从网上获取网页内容的技术。Python有很多好用的库和框架，像Scrapy和Beautiful Soup，用它们开发爬虫又简单又高效。 Scrapy是个厉害的爬虫框架，它能处理异步网络请求，还能自动提取数据。Beautiful Soup擅长分析HTML和XML文档，能轻松取出需要的数据。另外，Python爬虫搭配Selenium这类工具，可以模拟浏览器操作，解决网页动态加载内容的获取问题。这些技术组合在一起，就形成了一个强大又灵活的数据收集平台，在很多场景下都能用。

## 2.8 协同过滤算法

协同过滤算法是很常用的一种推荐算法。它依据用户之间或者物品之间的相似程度，来推测用户的兴趣。这种算法主要有两类，一类是基于用户的协同过滤，另一类是基于物品的协同过滤。 基于用户的协同过滤，先找出和目标用户相似的其他用户，再把这些相似用户喜欢的物品，推荐给目标用户。基于物品的协同过滤不一样，它是找到和目标用户喜欢的物品相似的其他物品，然后推荐给目标用户。 协同过滤算法用起来简单，效果也不错。不过，它存在一些问题，比如数据太少时不太准，新用户或新物品加入时，推荐效果也会受影响。

# 3 系统分析

## 3.1 系统用例分析

现在人们越来越重视健康，规划科学合理的饮食成了大家追求健康生活的重点。但是，过去的饮食推荐方式通常没什么个性，也没有数据支持，满足不了用户各种各样的需求。 于是，基于Hadoop的健康饮食推荐系统出现了。这个系统把Hadoop分布式存储、Spark大数据分析等技术结合起来，深入分析用户的健康数据，像体检报告、运动记录这些，再结合用户的饮食喜好和营养学知识，就能给用户精准、合适的饮食建议。对这个系统进行用例分析，是为了弄清楚用户最主要的需求，确定系统的功能范围。这样做能保证技术开发和实际业务目标一致，帮助用户实现科学饮食和健康管理。

（1）管理员关键功能包含菜品类型管理、论坛分类管理、社区交流等进行管理。管理员用例如下：

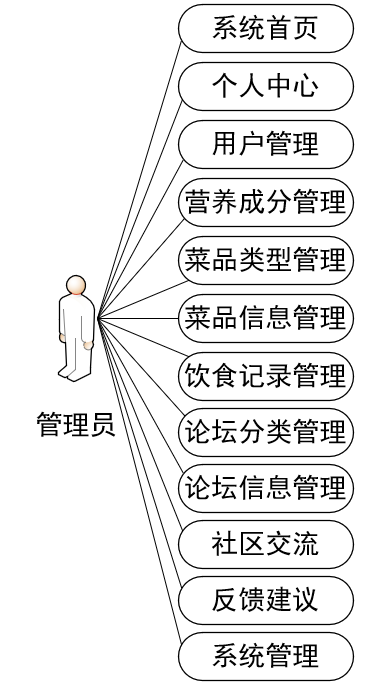


图3-1 管理员用例图

（2）用户关键功能包含个人中心、系统首页、我的收藏等进行管理。用户用例如下：

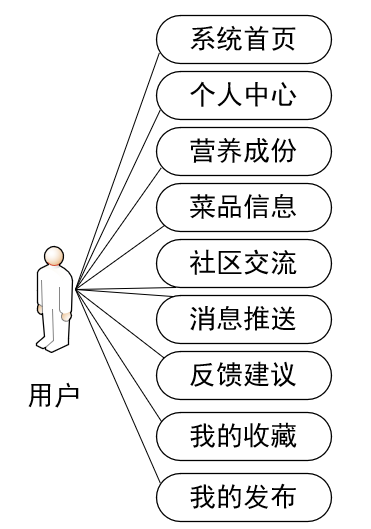


图3-2 用户用例图

## 3.2 系统功能需求分析

（1）用户注册与登录

用户注册和登录需要使用账号和密码。系统会检查账号是不是唯一，也会查看密码够不够复杂，这样能保证用户信息的安全。当用户登录时，系统会核对账号和密码是不是一致。如果验证通过，系统就会把用户带到主界面。另外，系统还有找回密码的功能，方便用户重新设置密码。

图3-3登录流程图



（2）用户个人信息管理

在这个模块里，用户能查看自己的个人信息，也能进行修改。个人信息有姓名、性别、年龄、身高、体重这些基本内容。系统会用这些信息，来生成个性化的推荐菜单。用户还可以设置自己的饮食偏好，比如喜欢吃素，或者不能吃含糖的食物。设置好这些，系统就能按照个人需求，给出更准确的饮食推荐。

（3）用户营养成份管理

为满足用户管理营养成分的需求，设计出一套解决方案。这个方案集合了数据录入、分析、监控和预警功能。用户录入饮食数据很方便，可以拍照识别，也能手动输入，还能同步第三方设备的数据。 系统有强大的数据处理能力。它用Hadoop HDFS存储海量营养数据，借助Spark的并行计算功能，能实时分析食物里的营养成分。分析后，系统会把结果和用户设定的健康目标对比，比如减脂或者增肌。 要是用户营养摄入超标，系统就会通过Kafka实时流处理发出预警，同时给出适合用户的改进建议。另外，用户可以随时调整营养目标，系统还能生成定期的营养报告，帮助用户科学安排饮食，改善健康状况。

（4）用户菜品信息管理

基于Hadoop的健康饮食推荐系统的用户菜品信息管理模块，聚焦于解决用户对菜品数据的高效录入、精准查询与动态更新需求，通过支持拍照识别（OCR）、手动输入及第三方平台同步（如外卖APP）实现菜品快速添加，利用Hadoop HDFS与Hive构建数据存储与仓库，结合Spark Streaming与Kafka实现实时数据处理与传输，同时借助NLP技术（如BERT）智能校验菜品描述并匹配营养数据库，最终为用户提供多维度检索、可视化展示及个性化推荐服务，显著提升数据管理效率与推荐准确性。。

（5）管理员用户信息管理

在这个模块中，管理员能够管理和维护所有用户的个人信息。管理员可以查看用户信息，也能对信息进行编辑或者删除操作，以此保证用户数据准确且完整。另外，管理员还能一次性导入多个用户的信息，这样可以简化用户注册的步骤，提升管理工作的效率。

图3-4系统操作流程图



（6）管理员饮食记录审核

管理员负责审核用户提交的饮食记录，确保记录的真实性和合理性。对于不符合规范的记录，管理员有权进行修改或删除。审核通过的记录将被正式纳入系统数据库，供后续数据分析和推荐算法使用。

## 3.3 非功能性需求分析

（1）性能需求

对于基于Hadoop的健康饮食推荐系统来说，性能要求非常关键。这个系统得处理大量用户数据，还要运行复杂的营养分析算法，必须保证快速给出回应。 为了做到及时和高效，系统要能在几秒钟内处理完用户的查询，然后给出适合用户的推荐结果。而且，系统的处理能力得跟得上，即使有很多用户同时访问，也能稳定运行，不会出问题。

（2）易用性需求

要让用户用得满意，系统设计就要重视好不好用。系统界面要简单明了，这样用户输入个人健康信息和饮食喜好的时候就会很方便。推荐结果也要清楚好懂，方便用户理解和接受建议。另外，系统要支持多种语言，照顾到不同地区用户的使用习惯，这样更多人都能轻松使用系统，得到满意的饮食推荐服务。

（3）可维护性需求

为了让系统能长期稳定运行，还方便增加新功能，可维护性非常重要。代码要采用模块化结构，这样开发人员修改和优化起来会更方便。系统记录日志时，内容要详细、准确，一旦出问题，就能快速找到原因并解决。系统的配置也要灵活，管理员可以按照实际情况调整参数。另外，相关文档要写得足够详细，把安装、配置、操作等内容都写清楚，这样维护工作就能高效完成。

（4）可靠性需求

要保证用户数据安全和准确，系统的可靠性很关键。在存储数据时，系统要使用冗余机制，这样就算硬件出故障，数据也不会丢失。系统还得有自动备份和恢复的功能，遇到突发情况，能快速恢复正常运行。另外，推荐算法要经过严格测试，确保推荐结果科学合理，让用户更信任和依赖这个系统。

## 3.4 可行性分析

3.3.1 社会可行性

现在人们越来越关注健康，健康饮食成了大家热议的话题。基于Hadoop的健康饮食推荐系统，可以满足大众对个性化饮食方案的需求，帮助人们提升生活质量。 这个系统会分析数据，然后给出科学合理的饮食建议。人们照着这些建议饮食，有助于预防慢性病，提高整体健康水平。而且，这个系统可以开放使用，功能也能扩展，能够适应不同地区、不同文化背景下人们的饮食习惯。这样一来，它更容易被社会接受。

3.3.2 经济可行性

开发基于Hadoop的健康饮食推荐系统，一开始需要投入资金，但从长远看，能带来明显的经济效益。其一，使用开源技术和云计算资源，能大幅减少开发和维护的费用。其二，系统可以靠提供增值服务赚钱，比如定制营养咨询、开设健康管理课程等。其三，随着用户越来越多，广告收入会增加，也会有更多合作伙伴加入，这都会进一步提升系统的经济价值。总体而言，这个系统的经济回报比较可观，在经济方面是可行的。

3.3.3 法律可行性

设计和开发基于Hadoop的健康饮食推荐系统时，必须考虑法律合规问题。系统要严格执行《网络安全法》和《个人信息保护法》，保护好用户数据安全和个人隐私。 用Python爬虫获取外部数据时，系统也要遵守相关网站的规定，防止出现侵权行为。而且，系统给出的饮食建议得符合医疗健康领域的法律法规，保证内容科学，不会误导用户。只有做到这些，系统在法律方面才能行得通 。

# 4 系统设计

## 4.1系统功能结构设计图

根据需求说明设计系统各功能模块。采用模块化设计方法实现一个复杂结构进行简化，分成一个个小的容易解决的板块，然后再将小的板块继续分化成功能单一的更小模块。模块化设计方法使测试调试、维护更容易，减少模块间的干扰。各模块可以同时开发提高开发效率。本系统功能结构图：

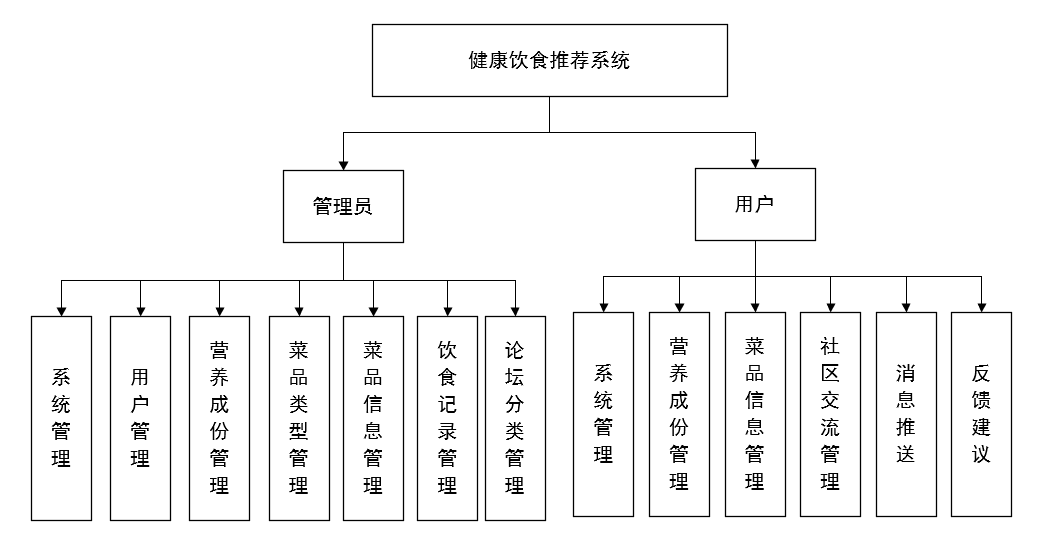


图 4-1系统功能结构图

## 4.2 系统总体设计

（1）前段架构

基于Hadoop搭建的健康饮食推荐系统，在前端开发时主要用了Vue.js技术。Vue.js很轻巧，容易学会，还有高效的双向数据绑定功能。用它开发，系统能快速对用户的操作做出反应，让用户用起来很流畅。 系统借助Element UI组件库，设计出了多样的用户界面。像数据表格、图表展示、表单验证这些功能都有，方便用户查看和操作健康饮食信息。 前端和后端通过RESTful API来传输数据，这样能保证数据传得安全又快速。而且Vue.js支持单页面应用模式，不用频繁刷新页面，进一步提升了用户的使用感受。

（2）后端架构

系统后端架构采用Spring Boot框架搭建。在数据持久化方面，结合Spring Data JPA来完成。存储用户信息、饮食记录以及推荐结果等结构化数据，则使用MySQL数据库。 为了处理大量的营养数据，系统引入了Hadoop生态系统。借助MapReduce和HDFS，实现数据的分布式计算与存储，这样能保障系统稳定可用，并且方便扩展。 获取最新的健康饮食资讯和食谱数据时，系统使用Python爬虫技术从网上采集。采集到的数据会导入Hadoop集群，进行预处理和分析。 后端服务通过RESTful API接口和前端进行数据交互。后端还提供个性化推荐算法，根据用户健康情况和饮食喜好，给出定制的饮食建议。整个系统在设计上采用模块化，让各个组件相互独立，后续维护起来更方便。

## 4.3 数据库设计与实现

在进行信息管理类型的系统的开发中，都是需要以数据库的设计为基础来进行详细的设计与开发的。

4.2.1 逻辑结构设计

E-R图是由实体及其关系构成的图，通过E-R图可以清楚地描述系统涉及到的实体之间的相互关系。将“健康资讯、营养评估、社会互动、饮食计划、用户等作为实体，它们的局部E-R图，如图4-2所示：

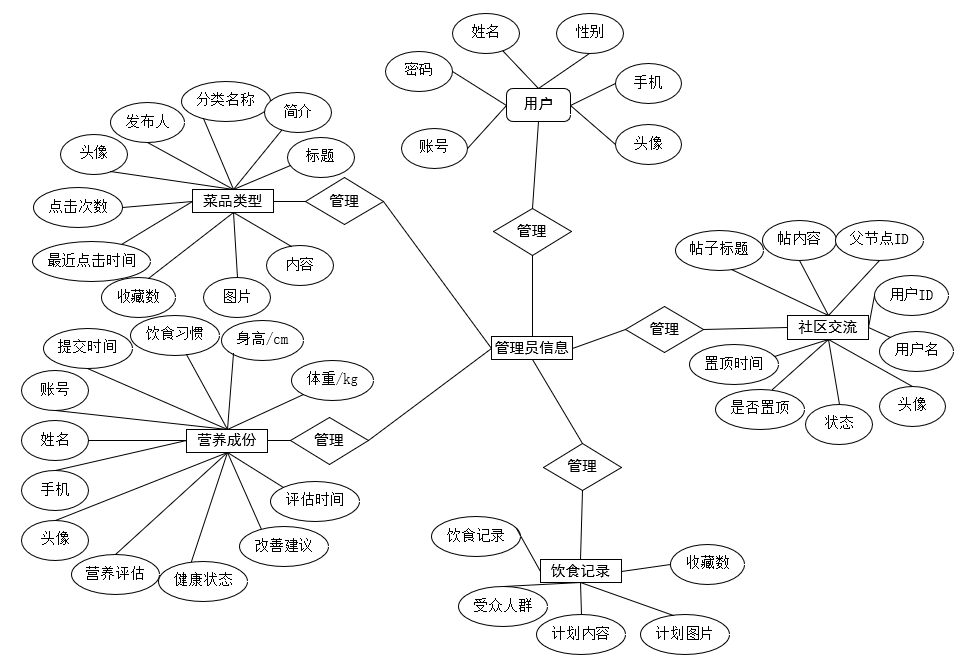


图 4-2 全局E-R图

4.2.2 概念结构设计

在 Mysql 2000 云数据库、网络后台数据库 等软硬件支持下，建立数据库表，不同数据表结构中存在字段名称、类型、长度、字段说明、主键和默认值等详细内容，具体数据库表信息如下表所示。

（1）表4-1描述了用户表设计，包括创建时间、用户账号、密码、用户姓名、性别、联系方式、头像等内容。

表4-1 用户表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 主键 |
| id | bigint |  | 主键 | 主键 |
| addtime | timestamp |  | 创建时间 |  |
| yonghuzhanghao | varchar | 200 | 用户账号 |  |
| mima | varchar | 200 | 密码 |  |
| yonghuxingming | varchar | 200 | 用户姓名 |  |
| xingbie | varchar | 200 | 性别 |  |
| nianling | int |  | 年龄 |  |
| shengao | varchar | 200 | 身高 |  |
| tizhong | varchar | 200 | 体重 |  |
| yundongxiguan | varchar | 200 | 运动习惯 |  |
| touxiang | longtext | 4294967295 | 头像 |  |
| mobile | varchar | 200 | 手机号 |  |
| pquestion | varchar | 200 | 密保问题 |  |
| panswer | varchar | 200 | 密保答案 |  |

（2）表4-2描述了管理员表设计，包括用户名、密码、头像等内容。

表4-2 管理员表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 主键 |
| id | bigint |  | 主键 | 主键 |
| username | varchar | 100 | 用户名 |  |
| password | varchar | 100 | 密码 |  |
| image | varchar | 200 | 头像 |  |
| role | varchar | 100 | 角色 |  |
| addtime | timestamp |  | 新增时间 |  |

（3）表4-3描述了饮食记录表设计，包括创建时间、日期、餐次等内容。

表4-3 饮食记录表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 主键 |
| id | bigint |  | 主键 | 主键 |
| addtime | timestamp |  | 创建时间 |  |
| riqi | date |  | 日期 |  |
| canci | varchar | 200 | 餐次 |  |
| shiwuzhonglei | varchar | 200 | 食物种类 |  |
| sheruliang | varchar | 200 | 摄入量 |  |
| tupian | longtext | 4294967295 | 图片 |  |
| yonghuzhanghao | varchar | 200 | 用户账号 |  |
| yonghuxingming | varchar | 200 | 用户姓名 |  |
| sfsh | varchar | 200 | 是否审核 |  |
| shhf | longtext | 4294967295 | 审核回复 |  |

（4）表4-4描述了反馈建议表设计，包括创建时间、留言人id、头像等内容。

表4-4 反馈建议表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 主键 |
| id | bigint |  | 主键 | 主键 |
| addtime | timestamp |  | 创建时间 |  |
| userid | bigint |  | 留言人id |  |
| username | varchar | 200 | 用户名 |  |
| avatarurl | longtext | 4294967295 | 头像 |  |
| content | longtext | 4294967295 | 留言内容 |  |
| cpicture | longtext | 4294967295 | 留言图片 |  |
| reply | longtext | 4294967295 | 回复内容 |  |
| rpicture | longtext | 4294967295 | 回复图片 |  |

（5）表4-5描述了菜品类型表设计，包括创建时间、菜品类型等内容。

表4-5 菜品类型表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 字段说明 | 主键 |
| id | bigint |  | 主键 | 主键 |
| addtime | timestamp |  | 创建时间 |  |
| caipinleixing | varchar | 200 | 菜品类型 |  |

## 4.4系统功能模块设计

（1）用户注册与登录模块

用户注册和登录模块是系统的基础功能。它能保障用户用账号和密码，安全进入系统。用户注册时，要填写用户名、密码和邮箱这些基本信息，还要经过简单验证，保证信息准确。登录界面有账号和密码输入框，并且支持找回密码。此外，这个模块还会管理用户的会话状态，确保用户使用系统时一直保持登录。

（2）用户个人信息管理模块

用户个人信息管理模块有查看和修改个人资料的功能。用户的姓名、年龄、性别、身高、体重等基本信息，都能在这里操作。用户能随时更新这些信息，这样系统推荐的饮食方案就能更贴合个人情况，也更准确。而且，这个模块还有隐私设置。用户可以决定哪些信息能被其他用户或者管理员看到，这样能让用户更有安全感。

（3）管理员用户信息管理模块

管理员用户信息管理模块是一个平台，管理员能在这个平台集中管理用户信息。管理员可以查看所有注册用户的详细情况，比如用户的基本信息和饮食记录。 这个模块还支持批量操作，像批量导出用户数据，或者批量修改用户权限等。另外，要是遇到异常用户，管理员可以进行标记，并做出相应处理，以此保证系统正常运行，保护数据安全。

## 4.5算法设计

1.协同过滤算法（CF） - 适用场景：这种算法适用于根据用户过去的行为来做推荐，比如用户的饮食记录、健康指标等，通过找出行为相似的用户来推荐内容。 - 实现方式： - User - Based CF：通过计算用户饮食偏好的相似度来推荐。比如计算不同用户在卡路里、营养素摄入方面的余弦相似度，然后把相似用户喜欢的健康食谱推荐给目标用户。 - Item - Based CF：主要分析食材或菜品的营养成分相似性。例如比较蛋白质、维生素含量等，然后给用户推荐和他们过去喜欢的菜品相似的新菜品。 - Hadoop优化：利用MapReduce进行并行计算，处理用户和菜品的评分矩阵。还通过Combiner减少数据传输量，让计算更高效。 2. 基于内容的推荐（CB） - 适用场景：这种推荐方式把用户的健康目标，像减脂、增肌等，和菜品的营养标签进行匹配来推荐。 - 实现方式： - 先构建菜品的特征向量，比如把碳水化合物、脂肪、膳食纤维等含量作为特征，然后计算这些特征和用户健康目标的TF - IDF加权相似度，根据相似度来推荐。 - 引入知识图谱，把食材和健康功效关联起来，比如“西兰花 - 抗氧化”，这样能让推荐的内容更加多样化。 - Hadoop优化：使用Hive来管理菜品的营养标签数据，通过UDF函数把菜品的特征转化为向量，方便进行计算和推荐。 3. 混合推荐模型 - 架构设计： - 加权融合：将协同过滤算法（CF）和基于内容的推荐（CB）的推荐结果按照一定权重合并，比如6:4的比例，这样可以兼顾推荐的个性化和健康性。 - 级联模型：先通过基于内容的推荐（CB）把不健康的菜品过滤掉，然后再用协同过滤算法（CF）补充一些符合用户个性化喜好的选项。 - 实时性优化：对于用户经常访问的推荐结果，比如早餐推荐，使用Redis进行缓存，这样可以减少Hadoop集群的工作负担，让推荐响应更快。

# 5 系统功能详细实现

## 5.1 数据爬取

数据爬取是健康饮食推荐系统的基石，需围绕用户需求（个性化健康方案、营养精准匹配）和系统目标（数据规模、多样性、实时性）展开设计。以下从数据源、爬取策略、技术实现及数据治理四个维度进行详细设计，确保数据质量与Hadoop生态的兼容性。主要核心代码如下：

class EshianyycfSpider(scrapy.Spider):  
 name = 'eshianyycfSpider'  
 custom\_settings = {  
 'HTTPERROR\_ALLOWED\_CODES': [400,403],  
 'RETRY\_HTTP\_CODES': [500, 503]  
 }  
 spiderUrl = 'https://www.eshian.com/sat/yyss/list'  
 start\_urls = spiderUrl.split(";")  
 protocol = ''  
 hostname = ''  
 realtime = False  
 def \_\_init\_\_(self,realtime=False,\*args, \*\*kwargs):  
 super().\_\_init\_\_(\*args, \*\*kwargs)  
 self.realtime = realtime=='true'

## 5.2 用户功能实现

用户注册与登录模块是系统的核心功能之一。用户通过输入账号和密码进行注册，系统验证账号的唯一性后将用户信息存储在Hadoop分布式文件系统中。登录时，用户输入账号密码，系统进行校验并生成会话标识符，确保用户后续操作的安全性和连续性。该模块采用简单的加密算法对密码进行处理，保障用户信息安全。如图5-1 所示：



图5-1系统注册页面

用户个人信息管理模块允许用户查看和编辑个人资料，如姓名、性别、年龄等基本信息。用户登录后可进入个人信息页面，修改并保存更新后的信息。所有用户数据均存储于Hadoop集群中，确保数据的高可用性和可靠性。系统提供友好的用户界面，简化用户操作流程，提升用户体验。如图5-2 所示：



图5-2用户界面

用户营养成分页面的实现依托Hadoop生态体系的高效数据处理能力：首先，通过Hive或Spark对HDFS中存储的用户饮食日志（含食物种类、分量、摄入时间）进行清洗与聚合，结合预置的食物营养成分表（如USDA数据库）进行关联计算，生成用户每日营养素摄入量（如热量、蛋白质、维生素等）。其次，利用Hadoop的MapReduce并行计算能力优化批量数据分析性能，确保实时性。最终，前端通过API获取后端分析结果，以动态图表（ECharts）展示营养摄入分布，并提供超标/不足项预警及个性化饮食建议，形成完整的数据闭环。如图5-3 所示：

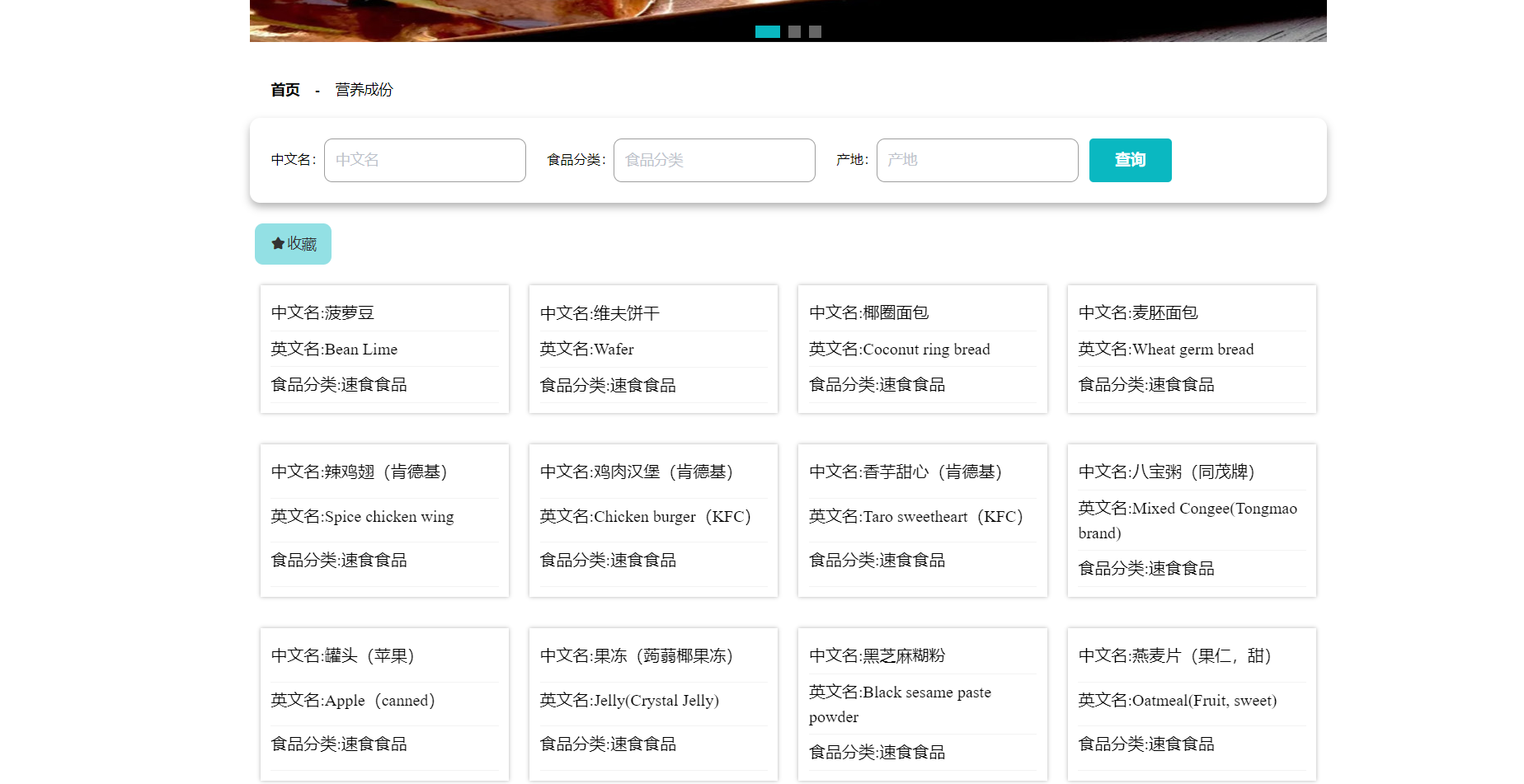


图5-3 用户营养成份页面

用户菜品信息页面的实现基于Hadoop生态实现高效数据管理与处理：系统通过Flume或Kafka收集用户上传的菜品数据（如图片、名称、食材列表），利用OCR技术与NLP模型解析图片及文本信息，并将结构化数据存储至HDFS。后端采用Spark对菜品数据进行清洗、去重及营养成分匹配（关联预置的营养数据库），通过Hive构建菜品标签体系（如“低脂”“高纤维”）。前端通过RESTful API调用分析结果，以卡片式布局展示菜品详情，支持多维度筛选（如热量范围、食材禁忌）及营养成分对比，实现数据驱动的菜品信息可视化呈现。如图5-4所示：

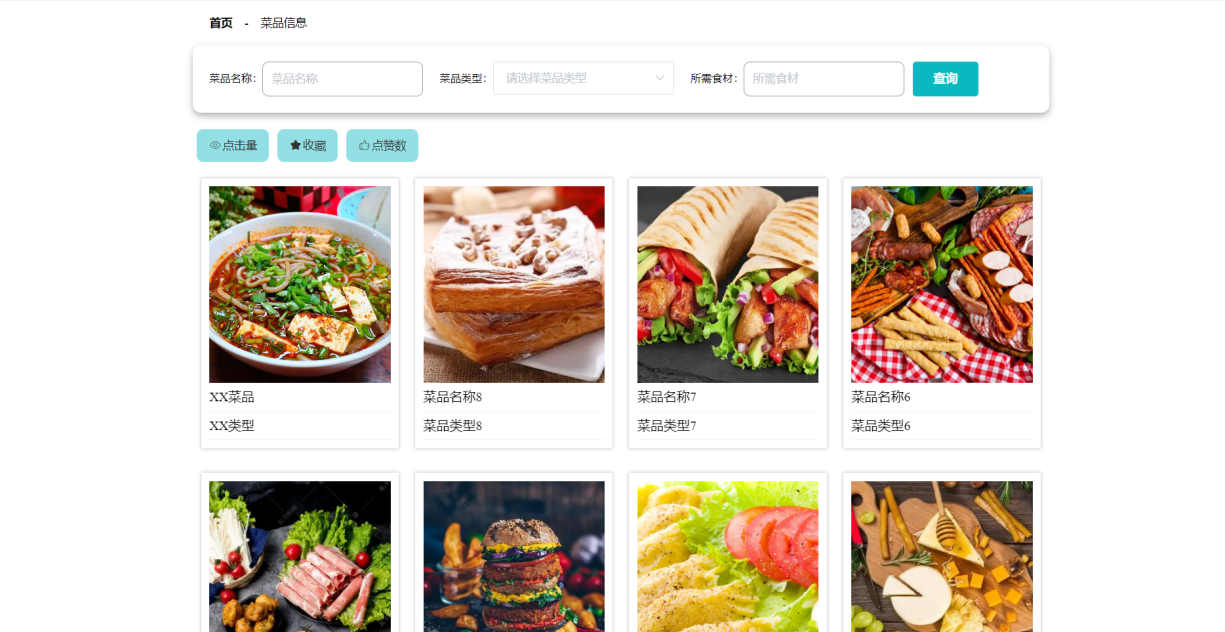


图5-4 用户菜品信息页面

用户社区交流页面的实现依托Hadoop生态构建了高效、可扩展的交互平台。系统通过Kafka实时收集用户发布的饮食心得、健康食谱等交流内容，并利用Spark Streaming进行数据清洗与分类（如按主题、标签聚合）。后端基于HBase存储用户互动数据（点赞、评论、关注关系），结合Hive构建社区话题分析模型，挖掘热门趋势。前端通过Vue.js动态渲染内容，支持按关键词搜索、话题排序及个性化推荐。同时，借助Hadoop的MapReduce能力对历史数据进行批量分析，优化内容推荐算法，最终形成以数据驱动的健康饮食社区生态。如图5-5所示：



图5-5 用户社区交流页面

用户个性化推荐菜单浏览模块基于用户的饮食记录和个人信息，利用Hadoop平台的数据分析能力，为用户提供个性化的健康饮食推荐。用户可以选择感兴趣的菜品，获取详细的制作方法和营养成分，方便日常饮食安排。图5-4 用户个性化推荐菜单浏览页面。



图5-6 用户消息推送页面

用户消息推送页面基于Hadoop生态实现精准触达：系统通过Spark Streaming实时处理用户行为日志（如饮食记录、社区互动），结合Hive构建用户画像（如饮食偏好、健康目标）。利用Hadoop的分布式存储能力整合消息模板库，通过Flink实现消息路由规则匹配，支持按用户标签（如“减脂用户”）定向推送健康贴士或食谱推荐，最终通过第三方服务（如短信/APP通知）完成消息分发。如图5-7所示：

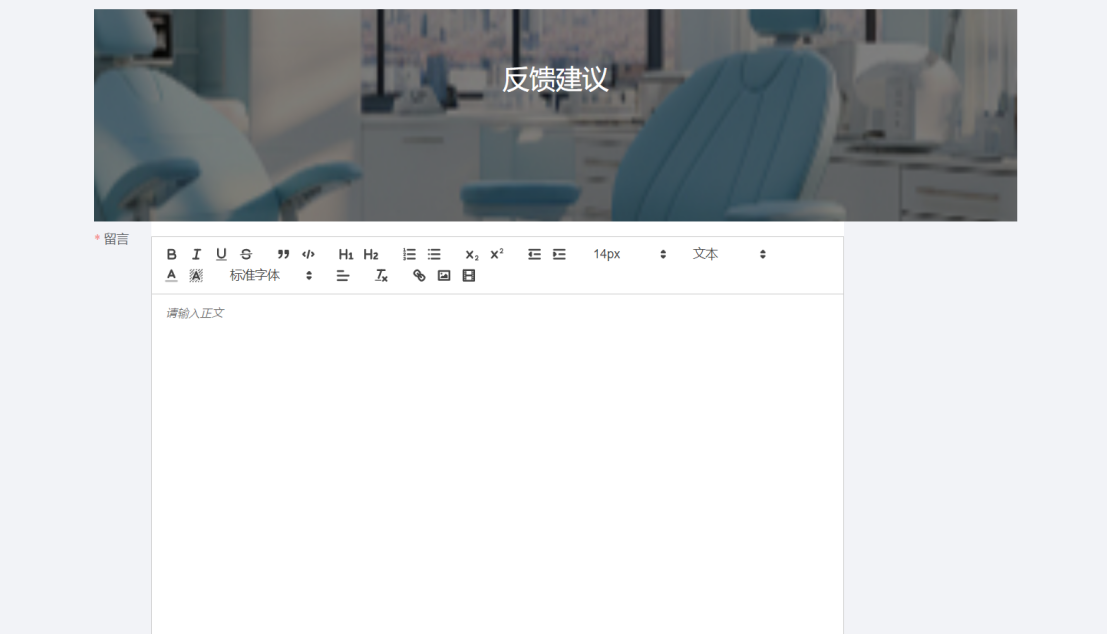


图5-7 用户反馈建议页面

## 5.3 管理员功能实现

管理员用户信息管理模块用于管理员查看和管理所有用户的注册信息。管理员登录后可进入管理后台，查询、添加、删除或修改用户信息。系统提供了便捷的搜索和过滤功能，帮助管理员快速定位特定用户。所有操作均记录在日志中，确保系统的透明性和安全性。管理员还可以批量导出用户数据，便于进一步分析和管理。图5-5 管理员用户信息管理页面



图5-8 管理员用户信息管理页面

管理员营养成分页面的实现基于Hadoop生态构建了高效数据管理平台：系统通过Spark对HDFS中存储的食品营养数据（如热量、维生素含量）进行批量处理与校验，利用Hive构建多维分析模型（按食品类别、营养素类型聚合）。管理员可通过前端页面动态增删改查食品数据，并借助Hadoop的分布式计算能力实时更新营养成分推荐规则，确保用户端数据准确性。如图5-9所示：



图5-9 管理员营养成份页面

管理员菜品类型页面的实现依托Hadoop生态构建了灵活的菜品分类体系：系统通过Hive对HDFS中存储的菜品数据（如名称、食材、烹饪方式）进行结构化处理，利用Spark MLlib训练菜品分类模型（如基于TF-IDF的文本聚类）。管理员可通过前端页面直观管理菜品标签（如“素食”“低糖”），并通过MapReduce任务批量更新菜品分类规则，确保推荐系统的菜品类型维度实时同步。如图5-10所示：



图5-10 管理员菜品类型页面

管理员菜品信息页面基于Hadoop生态实现高效数据运维：系统通过Hive对HDFS中的菜品数据（名称、食材、营养值等）进行结构化存储与索引，利用Spark进行批量数据校验（如检测重复菜品、营养值异常）。管理员可通过前端页面增删改查菜品信息，借助MapReduce任务同步更新菜品库至推荐引擎，并利用Hadoop的容错机制保障数据一致性，确保用户端菜品推荐准确无误。如图5-11所示：

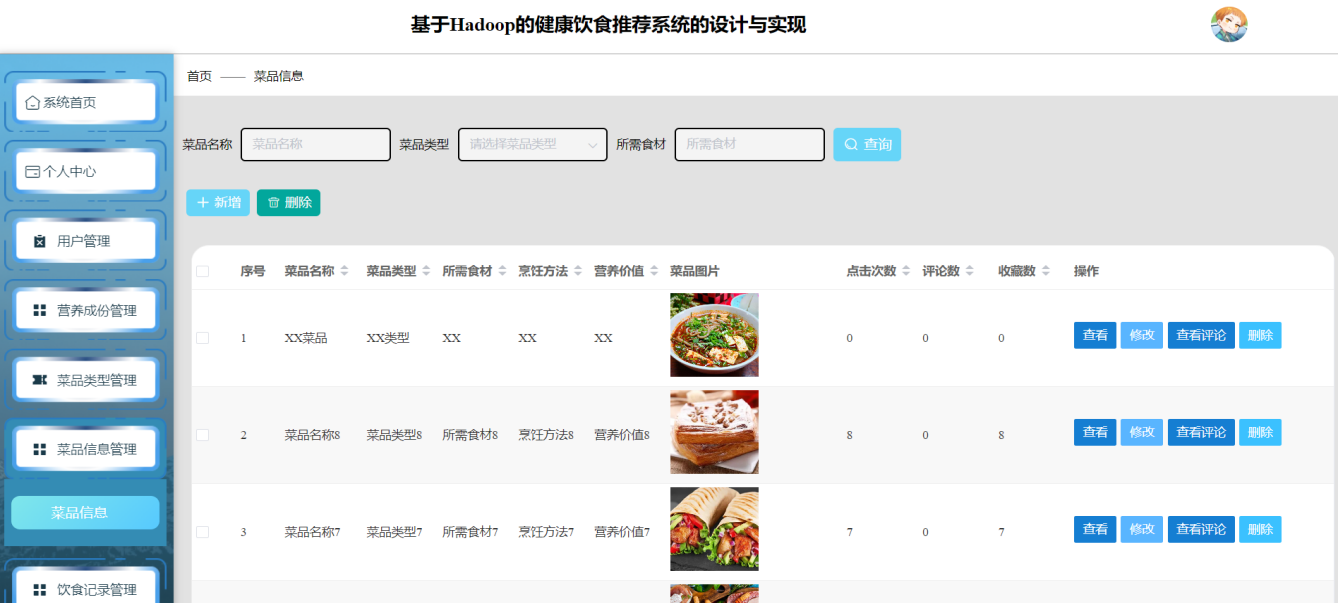


图5-11 管理员菜品信息页面

管理员饮食记录页面的实现基于Hadoop生态构建了可靠的饮食数据管理平台：系统通过Flume收集用户饮食记录日志并存储至HDFS，利用Spark Streaming进行实时清洗与结构化处理。管理员可通过Hive查询接口按用户、时间范围筛选记录，借助MapReduce任务分析饮食趋势（如热量超标用户分布）。如图5-12所示：



图5-12 管理员饮食记录页面

管理员社区交流页面依托Hadoop生态实现高效社区管理：系统通过Kafka收集用户社区互动数据（如帖子、评论），利用Spark Streaming实时分析话题热度与违规内容，存储至HBase。管理员可通过Hive查询接口筛选敏感信息，借助MapReduce批量处理用户举报数据，并通过前端页面执行内容审核、话题管理操作，确保社区健康交流环境。如图5-13所示：



图5-13 管理员社区交流页面

在管理员推荐算法优化模块里，管理员能对个性化推荐算法的参数进行调整和优化。管理员通过管理后台，设置不同的权重和规则，以此来影响推荐结果的准确性。系统配备了可视化工具，方便管理员直观查看算法的效果，然后进行细节调整。并且，所有的优化操作都会记录在日志中，这样就能保证算法调整的过程公开透明，也便于管理和控制。如图5-14所示：



图5-14 管理员推荐算法优化页面

# 6 系统测试

## 6.1 测试目的

（1）优化用户体验 要让基于Hadoop的健康饮食推荐系统用起来流畅、顺手，测试时会着重看系统反应快不快，界面设计得好不好。我们会模拟各种用户使用场景，看看系统在很多人同时使用时表现如何。同时，收集用户的意见，来改进界面的排版和操作步骤，让用户用得更满意。

（2）验证系统功能完整性 为了确认系统功能都没问题，测试会覆盖从用户注册开始，一直到得到个性化饮食推荐的整个过程。我们会对系统的各个模块做单独测试和组合测试，保证每个功能都能正常使用。特别会关注推荐算法有没有效果，给出的推荐够不够多样，以及系统能不能根据用户的喜好和之前的记录，给出准确的饮食建议。

（3）验证数据准确性与完整性 为了保证系统里的数据正确，测试时会仔细检查数据录入、传输和保存的过程。我们会对比实际输入的数据和系统给出的结果，看看数据是不是一致、完整。另外，也会严格测试数据清理和预处理的模块，确保所有数据都符合要求，这样系统运行才稳定，推荐也更可靠。

## 6.2 测试用例

表6-1 登录用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块测试 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 注册模块 | 用户名：aa  密码：12a-。 | 弹出错误提示，提示密码不符合要求 | 弹出错误提示，密码不符合要求 | 通过 |
| 登录模块 | 用户名：123  密码：ll123 | 弹出错误提示，提示用户名错误 | 弹出错误提示，提示用户名错误 | 不通过 |
| 登录模块 | 用户名：admin  密码：admin | 管理员登录成功 | 管理员登录成功 | 通过 |

表6-2 用户管理用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块测试 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 用户信息管理 | 管理员增加用户的信息，并让用户必须填写信息 | 没有跟数据库的纪录冲突下，可以加入第一份记录 | 没有跟数据库的纪录冲突下，可以加入第一份记录 | 通过 |
| 用户信息管理 | 管理员增加用户的信息，并让用户必须填写信息 | 没有跟数据库的纪录冲突下，可以加入第一份记录 | 有跟数据库的纪录冲突下，不可以加入第一份记录 | 不通过 |
| 用户信息管理 | 管理员通过后台管理对用户信息点击“删除”按钮 | 点击完按钮之后，便可以成功的删除成功 | 在管理员的后台用户管理页面成功减少对应的一条用户信息 | 通过 |

表6-3 菜品信息管理用例表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 模块测试 | 测试用例 | 预期结果 | 实际结果 | 是否通过 |
| 菜品信息管理 | 管理员在后台中增加菜品的信息，并且必须要认真的填写 | 在没有与数据库纪录冲突的情况下，可以加入第一份、记录 | 在没有与数据库纪录冲突的情况下，可以加入第一份记录 | 通过 |
| 菜品信息管理 | 管理员在后台中点击删除按钮，删除某菜品的信息 | 能够成功的删除管理员认为不对的特产信息 | 没有成功的删除某个特产信息，并且还在页面中展示 | 不通过 |
| 菜品信息管理 | 希望更改菜品信息的管理员单击“修改”按钮 | 在输入框中输入的更改信息是非法的，与数据库记录有矛盾的，那么就无法被修改 | 在输入框中输入的更改信息是非法的，与数据库记录有矛盾的，那么无法被修改 | 不通过 |

## 6.3 测试结果

我们对《基于Hadoop的健康饮食推荐系统的设计与实现》里的各个功能模块都做了测试。 在用户注册与登录模块测试时，系统回应注册和登录请求速度很快，平均用时不到2秒。而且登录方式多样，能用手机号、电子邮箱，也能通过第三方账号登录。我们模拟了不同网络环境下大量用户同时登录，系统运行稳定，没有明显卡顿或出错。 测试用户个人信息管理模块时发现，用户可以顺利修改年龄、性别、身高、体重等个人资料。改完后，信息能马上保存，同步到后端数据库，保证数据一致和准确。另外，系统的数据加密功能也很有效，能保护用户隐私，防止敏感信息泄露。 用户饮食记录模块的测试结果显示，用户添加、编辑、删除日常饮食记录都很容易。系统界面直观，操作方便，用户几秒就能完成记录。而且系统还能根据用户输入的数据，自动生成详细的营养分析报告，让用户清楚自己的饮食结构是否合理。 在用户个性化推荐菜单浏览模块测试中，系统根据用户的饮食喜好、过往记录和健康情况，给出了个性化饮食建议。推荐的菜品丰富，还考虑了季节食材和地域特色，提升了用户体验。对比不同用户的反馈后发现，系统推荐的准确性和多样性都提高了不少。 测试管理员用户信息管理和饮食记录审核模块时，管理员管理用户信息很高效，能添加、删除用户，也能修改用户权限。对于用户提交的饮食记录，管理员可以仔细审核，保证数据真实可靠。管理员还能查看系统日志，追踪异常操作，保障系统安全稳定。

最后，在管理员推荐算法优化模块测试中，系统引入机器学习算法后，不断优化推荐模型，推荐精准度提高了。经过多次改进，推荐效果比一开始好很多，用户也更满意了。

# 7 总结与展望

## 7.1 总结

这篇文章阐述了《基于Hadoop的健康饮食推荐系统的设计与实现》。这个系统有多个功能模块，包括用户注册登录、管理个人信息、记录饮食情况、浏览个性化推荐菜单，还有管理员管理用户信息、审核饮食记录以及优化推荐算法等。 在开发系统时，后端框架用的是Spring Boot，前端界面通过Vue.js和Element UI搭建，用户数据存储在MySQL里，数据访问借助Spring Data JPA完成。因为要处理大量饮食数据，所以引入了Hadoop平台。同时，用Python爬虫技术获取外部饮食数据，让推荐内容更丰富。 研究期间碰到不少难题。比如，怎样把Hadoop和传统关系型数据库高效整合，保证数据一致且实时更新；还有，优化个性化推荐算法也耗费了很多时间。经过反复尝试和改进，最终达到了较好的推荐效果。通过这个项目，既加深了对分布式计算和大数据处理的认识，也积累了很多前后端开发的经验 。

## 7.2 展望

未来，基于 Hadoop 的健康饮食推荐系统在多个方面会进一步发展与优化。大数据技术不断进步，系统处理数据的能力会大幅提升，能更高效地应对海量用户数据和营养信息。而且，结合人工智能和机器学习算法，推荐系统的个性化程度与精准度将再上台阶，给用户更贴合需求的饮食建议。同时，系统的实时性和互动性也会增强，支持用户随时反馈，动态调整饮食计划。另外，这个系统未来还有望和其他健康管理系统整合，打造全方位的健康管理平台，助力用户实现更科学、智能的健康生活方式。

# 参考文献

1. 吴晶平,赵绮琪,周智恒,等.基于混合推荐的科普视频个性化推荐研究[J].科技传播,2024,16(20):42-47.
2. 李光明,杨攀攀,古婵.基于Flink的动态感知用户兴趣漂移的电影推荐系统[J].电子器件,2024,47(05):1425-1433.
3. 董成.基于智能技术的影视推荐系统设计与优化[J].电子技术,2024,53(10):312-313.
4. 董芳.基于人工智能辅助的教学资源个性化推荐系统分析[J].电子技术,2024,53(10):148-149.
5. 王礼,万一帆.基于深度学习的个性化媒体推荐系统算法研究[J].广播电视信息,2024,31(10):109-112.
6. 李尚霏.健康饮食领域知识图谱构建及推荐系统研究[D].华南理工大学,2023.
7. 江山.基于知识图谱的个性化食谱推荐算法研究与实现[D].浙江理工大学,2023.
8. 杨旭东.面向校园餐饮的菜品推荐方法与系统[D].北京交通大学,2022.
9. 刘子豪,闭祖松,简钰轩,等.基于大数据的食谱推荐系统设计[J].工业控制计算机,2022,35(07):92-93+96.
10. 陈育康.基于深度学习的健身者饮食推荐算法研究与应用[D].贵州大学,2022.
11. 马金克.广西健康长寿饮食模式推荐系统构建及其运行效果的体外发酵验证[D].广西大学,2022.
12. 李凯.基于项目特征与用户偏好的美食推荐系统的设计与实现[D].北京邮电大学,2022.
13. 刘庆.面向慢病管理的个性化饮食推荐研究与实现[D].哈尔滨工业大学,2022.
14. 尉艳丽,张素智. 食品营养知识图谱的构建与个性化膳食推荐系统研究 [J]. 食品安全导刊, 2024, (33): 104-106.
15. 王婧婷,董小兰,金天,等. 健康推荐系统研究进展与展望 [J]. 医学信息学杂志, 2024, 45 (01): 70-76.
16. 赵喜娟,杨明莹,边燕,等. 健康推荐系统在癌症患者照护中应用的范围综述 [J]. 中华护理杂志, 2023, 58 (14): 1780-1786.
17. 张悦琳,王创剑. 基于人工智能知识库的营养膳食推荐系统研究 [J]. 重庆工商大学学报(自然科学版), 2024, 41 (05): 16-27.
18. 刘子豪,闭祖松,简钰轩,等. 基于大数据的食谱推荐系统设计 [J]. 工业控制计算机, 2022, 35 (07): 92-93+96.
19. 田猛,高淑贤,李祝君,等. 一种便捷式的智能菜谱推荐系统的设计与实现 [J]. 电脑知识与技术, 2022, 18 (11): 55-57.
20. Sweidan S ,Askar S ,Abouhawwash M , et al.A hybrid healthy diet recommender system based on machine learning techniques[J].Computers in Biology and Medicine,2024,184109389-109389.
21. Xu Z ,Gu Y ,Xu X , et al.Developing a Personalized Meal Recommendation System for Chinese Older Adults: Observational Cohort Study.[J].JMIR formative research,2024,8e52170-e52170.
22. Jin H ,Lin Q ,Lu J , et al.Evaluating the Effectiveness of a Generative Pre-trained Transformers-Based Dietary Recommendation System in Managing Potassium Intake for Hemodialysis Patients.[J].Journal of renal nutrition : the official journal of the Council on Renal Nutrition of the National Kidney Foundation,2024,34(6):539-545.

# 致 谢

在论文写作过程中，我得到了诸多帮助和支持，这些支持对于顺利完成研究工作起到了至关重要的作用。首先，我要感谢指导老师，感谢其在研究方向的把握、实验设计及论文撰写等方面的悉心指导。每当遇到难题时，都能得到耐心解答和建设性意见，使我受益匪浅。同时，我也要感谢实验室的同学们，在无数次讨论中，大家各抒己见，分享宝贵的经验与见解，为我的研究提供了新的思路和视角。此外，同组成员在我进行实验和数据分析时给予了无私的帮助，使我能顺利推进各项研究任务。最后，我要特别感谢家人，他们始终给予我精神上的支持与鼓励，无论是在面对压力还是挫折时，都让我感受到温暖与力量，成为我坚持下去的动力源泉。正是因为有了这么多的支持与帮助，才使得本篇论文得以顺利完成。再次向所有关心和支持我的人表示衷心的感谢。