

**毕业设计（论文）**

题 目 基于MVC的超市管理

系统的设计与实现

专 业 软件工程

学　　 号 150140132

学 生 尹文琪

指 导 教 师 韩希先

答 辩 日 期 2019年6月12日

**摘 要**

在当今科技迅猛发展的环境下，计算机信息化管理技术已经进入到各个领域，成为各种行业发展必备的工具；生产经营活动中的各个环节在计算机技术的帮助下，必然会朝着高效精确的方向发展。在超市初期发展中，大多都是通过人工来完成各种信息管理，效率低下，随着超市的规模增大，商品数量和品类越来越多，传统的人工管理方式的弊端就愈发明显，不仅增加了人力成本，而且容易造成工作效率降低、错误率增加等问题。为解决超市在目前发展中存在的问题，需要集合国内中小型超市实际发展情况，开发一个适用的超市管理系统。

系统开发采用基于MREN的MongoDB、React、Koa、NodeJS框架技术，完成了MVC三层架构。以IntelliJ IDEA为开发工具，用React+AntD完成前端，采用MongoDB作为数据库管理系统管理后台数据库，采用NodeJS作为基本编程语言开发。超市管理系统功能主要包括：基本信息管理、顾客管理、产品管理、订单管理和数据分析等功能。本系统的核心功能为数据分析模块的设计与实现，该模块通过对超市经营数据进行多方面、多层次的数据分析，可以有效的提高超市经营效率，节约超市人工管理成本，极大降低超市日常管理中的错误率，并能够为超市管理层决策通过辅助参考。

关键词：超市管理；数据分析；MERN

**Abstract**

With the rapid development of modern science and technology,computer information management technology has entered into various fields, become a necessary tool for the development of various industries; With the help of computer technology, each link in the production and operation activities will inevitably develop towards the direction of high efficiency and precision. At the beginning of the supermarket development, mostly by artificial to complete all kinds of information management, low efficiency, increase with the size of the supermarket, more and more quantity and category, the disadvantages of traditional artificial management way is increasingly obvious, not only increased the manpower cost, and easy to cause such problems as low efficiency, error rate increase. In order to solve the problems existing in the current development of supermarkets, it is necessary to set up the actual development situation of domestic small and medium-sized supermarkets and develop a suitable supermarket management system.

System development adopts mren-based MongoDB, React, Koa and NodeJS framework technologies, and completes MVC three-tier architecture. IntelliJ IDEA is used as the development tool, React+AntD is used to complete the front-end, MongoDB is used as the database management system to manage the back-end database, and NodeJS is used as the basic programming language to develop. The functions of the supermarket management system mainly include: basic information management, customer management, product management, order management and data analysis. The core functions of this system for the design and implementation of data analysis module, the module of supermarket management data for various, multi-level analysis of data, can effectively improve the management efficiency of supermarket, supermarket artificial management cost saving, greatly reduce the error rate in the supermarket daily management, and be able to make a decision on the supermarket management by auxiliary reference.

**Keywords:** Supermarket Management, Data Analysis, MERN

**目 录**

**摘要** Ⅰ

**Abstract** Ⅱ

**第1章 绪论** 1

1.1 课题背景及研究的目的和意义 1

1.1.1 课题背景 1

1.1.2 研究的目的和意义 1

1.2 国内外发展现状 2

1.2.1国外现状及分析 2

1.2.2国内现状及分析 2

1.3 主要研究内容 3

1.4 论文组织架构 3

**第2章 系统的相关技术** 4

2.1 系统技术框架 4

2.1.1 MVC架构 4

2.1.2 MERN 4

2.2 系统相关技术介绍 5

2.2.1 React 5

2.2.2 NodeJS 5

2.2.3 MongoDB 6

2.2.4 Express 6

2.3 本章小结 8

**第3章 需求分析** 9

3.1 系统概述 9

3.1.1 业务描述 9

3.1.2主要业务流程 10

3.2 功能需求分析 13

3.2.1 角色分析 14

3.2.2 业务功能 15

3.2.3 顾客管理 17

3.2.3 商品管理 18

3.2.3订单管理 19

3.2.3 数据分析 21

3.3 系统非功能性需求分析 22

3.3.1性能需求 22

3.3.2开发及运行环境 22

3.4 本章小结 22

**第4章 系统的设计** 23

4.1 系统总体设计 23

4.2 系统模块设计 24

4.2.1 数据分析模块设计 24

4.2.2 基本信息管理模块设计 25

4.2.3 顾客管理模块设计 26

4.2.4 商品管理模块设计 28

4.2.5 订单管理模块设计 29

4.3 数据库设计 30

4.3.1 数据库建模 30

4.3.2基于MongoDB数据库系统建模 36

4.4 本章小结 37

**第5章 系统的实现** 38

5.1 系统部署 38

5.2.1 部署环境准备 38

5.2.2 数据准备 38

5.2 系统功能实现 39

5.2.1 用户登录 40

5.2.2 基本信息管理 41

5.2.3 顾客管理 43

5.2.4 商品管理 44

5.2.5 订单管理 46

5.2.6 数据分析 47

5.3 本章小结 49

**第6章 系统的测试** 50

6.1 系统测试目标 50

6.2 系统测试内容 50

6.3 系统功能测试 50

6.3.1 基本信息管理功能测试 51

6.3.2 商品管理功能测试 51

6.3.3 顾客管理功能测试 51

6.3.4 订单管理功能测试 52

6.3.5 数据分析功能测试 52

6.4 系统非功能性测试 53

6.5 本章小结 53

**结论** 54

**参考文献** 55

**致谢** 57

**第1章 绪 论**

## 1.1 课题背景及研究的目的和意义

### 1.1.1 课题背景

随着大型超市体系的出现，所需要处理的超市数据信息日益增多，这对超市信息化管理系统的性能要求也愈发严格，系统不但需要实现对各种商品信息的实时更新，还需要保证各种超市经营数据的准确性。面对大量的各种各样类型的商品，一个结构良好的数据库将成为系统必备的支撑[1]。

先进的信息化管理作为一种高效管理手段，在一些发达国家和地区已经有着多年的实战经验，为超市信息化管理提供了成熟，稳定可靠的解决方案，这些解决方案成就了现在庞大的大型超市零售业体系。另外，随着在实践过程中逐步确定的物流管理方面的系列优化方法和可行性方案，系统科学合理的超市信息管理安排，不仅能够为超市提供更加高效的工作模式，还能够降低超市的经营管理成本，提高超市利润[2]。

### 1.1.2 研究的目的和意义

就零售体系而言，对于大型的零售企业来说，其悠久的发展历史使它在技术体系方面相当已经成熟。而对于国内的众多中小型超市而言，其信息化发展还处于落后水平。在和大中型超市的商业竞争中，对于高效的信息化管理系统的需求更加急切[3]。

对于国内一些中小型超市而言，其信息化发展水平远落后于大型超市，采用现代高效的信息化操作手段和管理方法，是中小型超市在激烈的商业竞争中生存和发展的必要条件。因此，信息化发展对于中小型超市而言非常重要。虽然中小型超市的管理方式简单，相对大型超市而言经营风险也更低，商品品类数量相对较少，但在日常经营管理中还是存在许多问题。例如人工管理成本相对较高、超市经营数据分析困难等，这些缺点在很大程度上阻碍了超市的进一步发展。一旦扩大规模，商品各种信息就会急速增加，员工、商品、订单、客户等方面的信息越来越复杂，超市管理也会更加复杂。如今引入新的信息化技术来解决中小型超市在目前发展中存在的问题是必要的​，信息化管理手段的引入能够实现科学规范化管理，帮助超市适应激烈的市场竞争[4]。

## 1.2 国内外发展现状

### 1.2.1国外现状及分析

在大多数发达国家和地区，超市管理系统已经相当成熟，如美国、日本、德国等。完善的进销存管理系统在这些国家和地区已经拥有较高影响力，能够快速准确地帮助超市经营者了解市场动态，供应商信息和超市库存信息等。多方面的超市数据分析也极大促进了超市行业的发展[5]。

超市是一个零售行业，客户可以前往零售店进行选购，也可以在网上选购物流送货上门。20世纪30年代，美国产生了第一家超市，开启了零售业的革命[5]。随着现代计算机信息化技术的发展，西方发达国家和地区的超市信息化程度一直处于领先地位，相对完善的管理系统取得了长足进步。国际化超市沃尔玛超市是超市行业中的领头羊，成为其他超市模仿的目标，他们在超市日常经营管理中所用到的的先进技术值得学习。

良好的超市管理系统有助于改善超市的管理，提高增加超市和客户之间的沟通，提高员工的工作效率，并降低数据信息管理中的错误率。这些超市行业领导者对先进技术的关注和应用在超市行业信息化管理的发展中起着不可或缺的作用。在欧洲和美国的超市发展历史中，当计算机信息化管理技术首次应用于线下实体超市管理时，其管理的项目从最初的不到300个[6]，到如今超过九成的管理都由计算机完成。根据调查，沃尔玛在超市发展的早期阶段拥有独立的网络系统，并实现了各个业务部门的信息管理。相比于同一时期同规模的其他超市，沃尔玛超市这种超前思维为它未来的成功打下了良好的基础。

### 1.2.2国内现状及分析

和欧美发达国家和地区的快速发展相比，中国虽然起步较晚，但发展势头迅猛，并已经取得极大成效，随着如ERP / MRPII等国外先进的超市信息化管理系统的引进，中国超市也正在逐步实现信息化技术管理的现代化，使超市管理变得越来越科学合理[7]。中国的超市数量在世界遥遥领先，各类型超市之间的跨度也非常大，而国外推出的大型超市管理系统只适合部分大型超市，国内数量众多的中小型超市无法使用。在我国，中小型超市大多出现在居住集中地区，如居民街道附近。但随着日益激烈的竞争，众多超市不断扩大销售模式，如在线预定送货上门等。因此，即使中小型超市的经营成本相对较低，但也需要改善经营管理模式，寻求更好的发展，这就会导致一个重要问题的出现，中小型超市并没有适合的超市管理系统，若强行使用不符合中小型超市实际经营情况的大型超市所用的软件将不可避免的造成资源浪费。

调查研究发现，目前我国的大多数中小型超市仍然缺乏科学高效的管理手段，手动管理仍占大多数。而且随着超市不断扩张，所面临的数据量也会越来越多，初期的管理模式已经不适用于新超市发展。因此，结合国外先进超市管理系统和国内中小型超市的实际经营情况，非常有必要开发适用于国内中小超市的信息化管理系统。

## 1.3 主要研究内容

本文通过对中小型超市在日常运营中的需求进行分析，结合真实超市运营订单数据，对超市管理系统进行研究与开发。在开发过程中采用新一代Web开发框架，实现MVC三层架构的超市管理系统，主要解决超市商品管理、用户管理、订单管理以及数据分析功能。保证了数据准确性和操作便捷性，在数据分析中采用先进商业数据分析手段，能够真实反应超市经营销售情况，从而为超市管理层提供决策分析的参考依据，使超市管理层进行科学合理化的管理。

## 1.4 论文组织架构

全文共分七个部分：

第一章是绪论部分，本章主要对论文课题背景和研究意义进行研究，并结合国内外超市管理系统发展现状对主要研究内容进行说明。

第二章是系统的相关技术的探讨，主要对系统实现中的MVC框架和MERN技术栈的特点进行了介绍。

第三章是需求设计部分，完成了系统总体设计以及子模块的用例图和用例说明，并对系统的非功能性需求进行了简单分析。

第四章时系统的设计，为本文的重点章节，专注于数据分析、订单管理、用户管理、商品管理模块的详细设计，包括功能模块图、流程图等。并对数据库进行建模设计，完成E-R图和数据表详细的设计。

第五章为系统实现，根据需求分析和总体设计，对系统子模块的功能进行具体实现，包含主要功能流程图，并对系统界面进行简单阐述。

第六章为系统测试，从功能测试和非功能性测试两方面对系统进行测试，并更加测试结果对系统进行改善，以实现预期目标。

**第2章 系统的相关技术**

## 2.1 系统技术框架

### 2.1.1 MVC架构

MVC的全称是 Model View Controller，是模型( model)－视图( view)－控制器( controller)的缩写，是一种将应用程序划分为三个主要逻辑组件的架构模式：模型、视图和控制器。构建这些组件中的每一个都是为了处理应用程序的特定开发方面。MVC是用于创建可伸缩和可伸缩项目的最常用的行业标准Web开发框架之一[8]。

Model组件对应于用户使用的所有与数据相关的逻辑。这可以表示在View和Controller组件之间传输的数据或任何其他与业务逻辑相关的数据。例如，Customer对象将从数据库中检索客户信息，对其进行操作并将数据更新回数据库或使用它来呈现数据。

View组件用于应用程序的所有UI逻辑。例如，Customer视图将包含最终用户与之交互的所有UI组件，如文本框，下拉列表等。

Controller充当Model和View组件之间的接口，用于处理所有业务逻辑和传入请求，使用Model组件操作数据，并与Views交互以呈现最终输出。例如，Customer控制器将处理来自Customer View的所有交互和输入，并使用Customer Model更新数据库。将使用相同的控制器来查看客户数据。

### 2.1.2 MERN

本文采用MERN技术栈实现MVC三层架构。MEAN是Javascript平台的现代Web开发框架[9]。MEAN是一组开源组件，它们共同构成了构建动态Web应用程序的端到端框架[10]，从顶部（在浏览器中运行的代码）开始到底部（数据库）。堆栈由以下部分组成：

MongoDB：文档数据库，由后端应用程序用于将其数据存储为JSON（JavaScript Object Notation）文档。

React：前端Web应用程序框架，直接在用户浏览器中运行的JavaScript代码，允许应用程序UI动态化。

Express（有时称为Express.js）：在NodeJS之上运行的后端Web应用程序框架。

NodeJS：JavaScript运行时环境，允许在JavaScript中实现应用程序后端。

MERN的架构原理如图2-1所示。

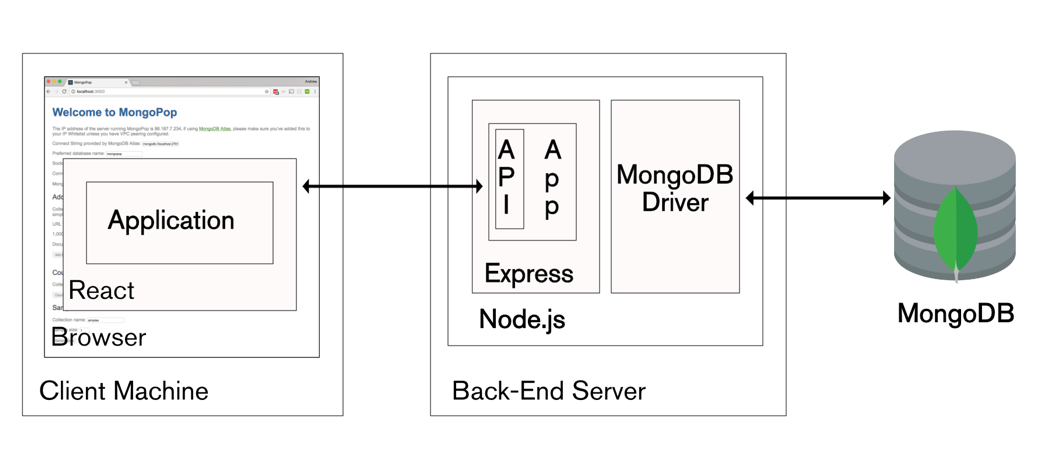


图2-1 MERN的架构原理

## 2.2 系统相关技术介绍

### 2.2.1 React

React 起源于 Facebook 的内部项目，用来架设 Instagram 的网站。于2013年5月开源。由于 React 的设计思想极其独特， 与Angular一样，React将前端应用程序分解为组件。 每个组件都可以保持自己的状态，父组件可以将其状态向下传递给子组件，这些组件可以通过使用回调函数将更改传递回父组件。React组件通常使用JSX实现，JSX是JavaScript的扩展，允许将HTML语法嵌入代码中。

React最常在浏览器中执行，但也可以在NodeJS中的后端服务器上运行，或者作为使用React Native的移动应用程序运行[11]。

### 2.2.2 NodeJS

NodeJS是一个运行后端应用程序的JavaScript运行时环境（通过Express）。基于谷歌的V8 JavaScript引擎，它被重新打包，重写了后端API，并构建了一个高效的JavaScript运行环境[12]。使得 JavaScript的执行速度远超 Ruby、 Python等脚本语言。.另外，NodeJS具有开发实时监控系统的能力，它采用事件驱动、异步编程、非阻塞模式的 IO处理工作机制，它以相对较低的系统资源消耗提供高性能和卓越的负载容量，使其成为依赖各种 IO资源的中间层服务的理想选择[13]。适用于数据密集型分布式部署环境中的实时应用程序的完美解决方案。

NodeJS通过对Web功能的部分扩展可以得到快速有效的Express框架。它还包括许多模块，这些模块提供了实现Web应用程序所必需的功能，包括HTTP等网络协议。可以使用npm工具安装第三方模块，包括MongoDB驱动程序。它是一个异步的，事件驱动的引擎，其中应用程序发出请求，然后继续处理其他有用的任务，而不是在等待响应时停止。完成请求的任务后，应用程序将通过回调通知结果。这使得能够并行执行大量操作，这在扩展应用程序时是必不可少的。MongoDB也被设计为异步使用，因此它适用于NodeJS应用程序。

### 2.2.3 MongoDB

MongoDB是一个用C ++编写的非关系数据库。具有高性能，易于部署，易于使用和易于存储数据的特点[14]。在设计时考虑了可伸缩性和开发人员敏捷性，弥补了快速和可扩展的键值存储与具有丰富功能的关系数据库之间的差距。MongoDB不是像关系数据库那样将数据存储在行和列中，而是将JSON文档存储在具有动态模式的集合中，这是一个开源文档数据库。主要特性有：

（1）面向集合存储，易于存储对象类型的数据

（2）模式自由

（3）支持动态查询

（4）支持完全索引，包含内部对象

（5）支持复制和故障恢复

（6）使用高效的二进制数据存储，包括大型对象

（7）文件存储格式为BSON(一种JSON的扩展)

关系数据库中的表存储一些结构化数据，每个记录的字段具有相同的组成。即使并非每条记录都需要所有字段，数据库也会为每条数据分配所有字段。非关系数据库存储为键值，其结构不固定。每条记录可以有不同的密钥，每条记录可以根据需要添加一些键值对。这不限于固定结构，其可以减少一些时间和空间开销。

MongoDB的文档数据模型使使用者可以轻松地存储和组合任何结构的数据，而无需放弃复杂的验证规则，灵活的数据访问和丰富的索引功能[15]。可以在不停机的情况下动态修改架构，这对于快速发展的应用程序至关重要。

### 2.2.4 Express

Express是运行后端应用程序（JavaScript）代码的Web应用程序框架，在NodeJS环境中作为模块运行[16]。可以处理将请求路由到应用程序的正确部分（或在同一环境中运行的不同应用程序）。一方面，可以在Express中运行应用程序的完整业务逻辑，甚至可以生成由用户的浏览器呈现的最终HTML。另一方面，Express可用于简单地提供RESTFULAPI，使前端应用程序能够访问所需的资源，例如数据库。

在本次系统设计中，通过 Express完成了两个主要功能，其一为当用户浏览应用程序时，将前端应用程序代码发送到远程浏览器，其二则提供前端可以使用 HTTP网络调用访问的 RESTFULAPI，以便访问数据库。

Koa 是一个新的 web 框架，由 Express 幕后的原班人马打造，通过利用 async函数，丢弃回调函数，有力地增强错误处理[17]。主有特点如下：

（1）封装创建应用程序功能。组织中间件（在侦听请求之前），生成上下文对象，执行中间件，执行默认响应方法或处理异常。

（2）扩展res和req，Koa通过自定义请求和响应对象，然后保持对res和req的引用，最后通过getter和setter方法实现扩展。

（3）中间件实现原则，中间件不是NodeJS中的概念，它只是从connect，express和Koa框架派生的概念。

（4）异常处理，对于同步代码，连接中间件的异常捕获是通过try / catch完成的。对于异步代码，无法捕获try / catch。Koa执行此操作，并在catch方法中发送错误事件，以便开发人员可以自定义异常处理逻辑[18]。

Koa框架图如图2-2所示。

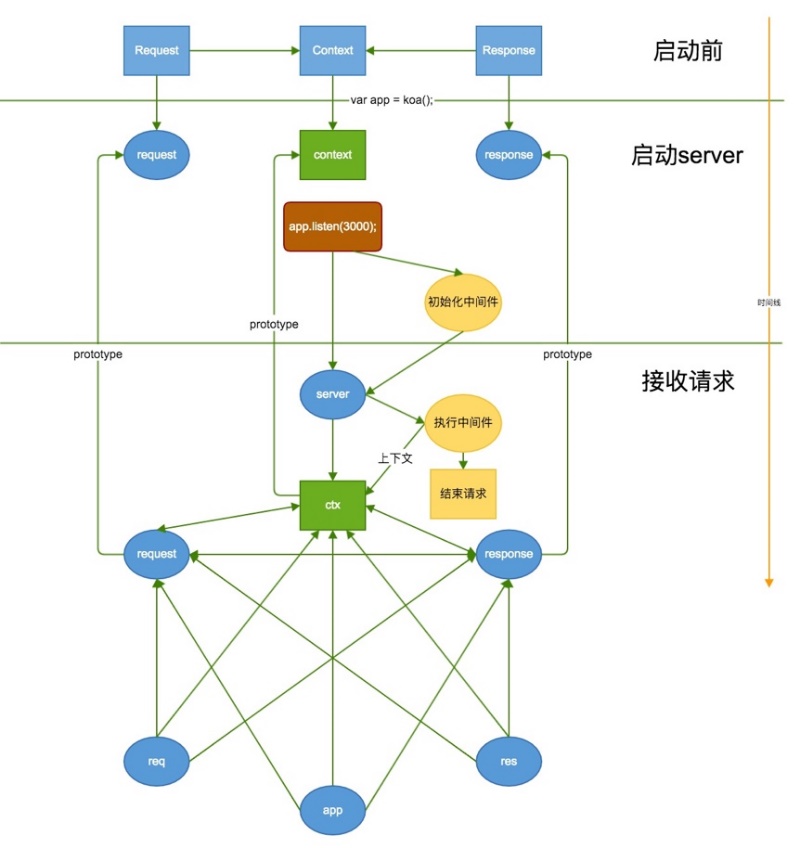


图2-2 Koa框架图

## 2.3 本章小结

本章对超市管理系统设计与实现中的技术框架进行了简单介绍，对MVC模式和MERN技术栈进行了简要说明，对MongoDB数据库、Express框架、NodeJS、React进行了详细介绍，为后续论文的设计与实现打下了坚实的基础。

**第3章 需求分析**

本系统主要实现了对超市信息管理以及数据分析的功能，不仅要对超市员工、顾客、商品、订单的基本信息进行管理，还需要对超市管理数据进行多方位分析，实现辅助超市管理决策的作用[19]。因此本系统主要从两个方面进行完成。一方面，它是超市基本信息的管理，包括添加、修改、查询和删除数据的基本操作。另一方面是从商业数据分析的角度对数据进行多层次系统全面的分析。

本章将从三个方面分析超市管理系统的需求，分别是业务需求，系统功能需求和系统非功能需求。并设计相应角色的用例图和用例说明，基本功能的流程图。

## 3.1 系统概述

本文分析了超市管理系统的具体业务，在大量调研的基础上，理解了用户需求和业务流程后，分析研究了MERN框架的工作原理以及流程，并采用 NodeJS实现该系统。

本文主要分析了系统的具体功能，围绕中小型超市的业务流程和日常经营情况将其划分为超市数据分析等五个模块。数据分析模块包括客户数据分析、商品数据分析和订单数据分析等。通过对超市的具体调研，确定了每个模块实现的功能，简化人工操作。通过分析每个模块实现的具体功能，借鉴 MERN架构体系设计系统的架构，确定界面风格，完成前台与后台交互的接口设计，后台类方法的设计以及数据库表的设计，并规定具体代码中函数的命名方式和数据库字段的命名方式。

本系统的主要任务不仅是要实现相应的功能模块，处理好数据的来源与格式也很重要。故而，本系统研究的另一个主要内容是对一个合适的数据源进行分析处理，不仅要对其格式进行处理，满足本系统格式的要求，还需要对数据情况进行拓展完善，最终达到本系统的要求。

最后，由于 Node具有单线程的特点，故而本系统的部署也是决定系统性能的关键因素，所以在开发过程中寻找高效率管理 Node进程的工具和完成 MERN框架搭建也是系统研究的重要内容之一。

### 3.1.1 业务描述

（1）基本信息管理：能够提供产品和客户基础数据或系统接口参数，完成权限管理和系统维护，以及保存和处理相应的数据。

（2）顾客管理：能够对超市的目标客户的基本信息进行管理，并在顾客购买超市商品产生订单的基础上对顾客进行分析，得出市场细分、用户购买习惯等结论。

（3）商品管理：能够对商品的基础信息进行管理，包括商品品类、进货价格、销售价格等，并针对实际销售情况对商品销售进行分析，如商品销售额、销售数量以及利润等。

（4）订单管理：主要对该超市的销售订单等相关信息进行管理，比如订单商品、订单折扣、订单利润等。

（5）数据分析：该模块能够智能地分析订单、客户和产品信息，并生成相应的可视化图表，然后为超市管理层提供参考，以作出相关决定。

（6）响应速度合理，安全性高。

（7）系统运行稳定，并且应易于维护。

### 3.1.2主要业务流程

3.1.2.1基本信息管理

能够提供产品和客户基础数据或系统接口参数，完成权限管理和系统维护，以及保存和处理相应的数据。基本信息业务管理的系统登陆流程图的设计如图3-1所示，密码设置如图3-2所示。



图3-1 系统登陆流程图



图3-2 密码设置流程图

3.1.2.2用户管理

用户管理可以管理用户的基本信息，包括查询、修改、添加、删除等功能，可以对用户名、密码和权限等信息进行操作。

新增用户需要超级管理员权限，进行操作之前，首先需要对管理员权限进行检验，若检验通过，才能进行操作。若所登陆的操作员无相应权限，则不能进入系统进行相关操作，此时，操作员可注销本次登陆的账号，换用其他具有该操作权限的操作员权限账号登陆系统进行相关操作。添加用户需要输入用户信息，在用户信息输入完成之后，需要点击保存按钮提交用户信息，此时系统将调用后台接口完成用户的添加，完成用户添加之后，可以在用户信息查询界面看到新添加的用户信息，并能够对新添加的用户信息进行修改等操作。如果在输入用户信息后单击取消按钮，则取消新增用户操作，并进入结束状态。

添加用户流程图如图3-3所示。



图3-3 添加用户

修改用户需要超级管理员权限，在进行操作之前需要先进行权限验证，在用户选择所需要修改的用户信息之后保存即可。修改用户信息流程图如图3-4所示。



图3-4 修改用户

删除用户需要超级管理员权限，在进行操作之前需要先进行权限验证，在用户选择所需要删除的用户之后，还需要第二次确认，第二次确认后将成功删除用户。删除用户信息流程图如图3-5所示。



图3-5 删除用户

3.1.2.2数据分析

通过该模块对该超市的相关商品的销售信息进行管理，借助相关的算法利用该模块对订单、顾客、商品信息进行智能分析，并生成相应的图表，为超市的管理层进行参考，从而做出相关决策。

通过该模块，能够管理超市相关产品的销售信息，该模块通过相关算法智能地分析订单、客户和产品信息。并生成相应的可视化图表，以便为超市管理层的决策提供参考。

## 3.2 功能需求分析

功能要求定义了超市管理系统所需的软件功能，使超市可以通过这些功能完成相关的超市经营信息管理业务，从而满足超市的管理需求。

超市管理系统主要解决影响商品日常经营管理过程中的信息数据问题，并结合高端管理技术。有针对性地进行开发，改进软件内部商品管理方法，为超市管理层提供参考，协助决策分析，专注于解决影响采购和商品仓储过程中的信息收集。 超市管理系统的各个子模块可以独立使用，也可以与数据分析模块结合使用，两部分共同构成了商品信息管理和超市数据分析的应用解决方案。

根据超市管理系统的基本业务场景，该系统主要为超市管理者提供数据分析与辅助决策功能，因此可以将超市管理系统的用户角色分为高层管理和普通操作员，不同的角色有不同的功能需求，另外，为了配合种种角色完成整个超市管理系统的业务流程，超市管理系统自身也有自动触发的定时任务。

根据超市管理系统的基本业务场景，该系统主要为超市管理者提供数据分析与辅助决策功能，因此，超市管理系统的用户角色可分为高级管理员和一般操作员。 不同的角色有不同的功能需求，另外，为了配合多种角色完成整个超市管理系统的业务流程，超市管理系统自身也有自动触发的定时任务。系统总体人员组织机构图如图3-6所示。



图3-6 系统总体人员组织机构图

### 3.2.1 角色分析

以下从超市管理系统的实际需求分析，系统涉及到角色如表3-1所示。

表3-1 角色分析

|  |  |
| --- | --- |
| 角色 | 职责或功能 |
| 高层管理人员 | 查询客户信息，产品信息和订单信息，并根据数据分析系统的分析结果，获得对超市运营和产品信息的全面控制。从而制定相应的销售策略。 |
| 业务员 | 主要对顾客信息进行管理，联系顾客，包括顾客购物反馈评价等。 |
| 库管员 | 主要对商品信息、商品品类信息进行管理，定期查看商品状态，保证商品质量，辅助销售员进行发货。 |
| 销售员 | 主要对销售订单进行管理，在顾客购买并发起订单之后完成发货。 |
| 系统管理员 | 对系统进行维护，用户权限进行管理 |

### 3.2.2 业务功能

从业务角度来看，超市管理系统主要包括基础信息管理、客户管理、商品管理、订单管理和数据分析。具体的各个功能的需求情况如下所示。

3.1.2.1基本信息管理

系统的主要管理功能离不开系统安全性、可靠性和机密性。因此要设置密码与检验手段是非常必要的。在基本信息管理部分中，需要详细分析和设计密码设置和用户登录部分。基本信息管理用例图如图3-7所示。



图3-7 基本信息管理用例图

系统管理员能够查询基本用户信息，并进行相应的修改操作，出于对系统安全性的考虑，在进入系统之前需要对用户权限进行检验，所以需要进行登陆操作。系统管理员登陆用例描述如表3-2所示。

表3-2 系统管理员登陆用例描述

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 登陆 |
| 标识符 | XY00 |
| 用例描述 | 描述系统管理员在进入系统之前的登陆检验功能 |
| 参与者表 | 管理员 |
| 优先级别 | 1 |
| 前置条件 | 无 |
| 后置条件 | 用户登陆成功 |
| 基本操作流 | 1.打开系统，若检测处于未登陆状态，自动跳转至登陆页面  2.在登陆窗口中输入用户名和密码，点击登陆按钮  3.提交登陆请求，若检验通过则跳转至系统，若检验失败则输出错误信息 |
| 可选操作流 | 若用户忘记密码，可联系超级管理员进行密码重置 |

系统管理员不仅能够对基本用户信息进行查询和相应的修改操作，还可以对系统用户密码进行修改，不仅能够防止用户忘记密码而无法登陆的情况，还能够通过定期修改密码进一步增加系统安全。密码设置用例描述如表3-3所示。

表3-3 密码设置用例描述

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 密码设置 |
| 标识符 | XY01 |
| 用例描述 | 描述了系统管理员使用本系统基本信息管理子系统进行密码设置的整个过程 |
| 参与者表 | 系统管理员 |
| 前置条件 | 系统管理员已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.管理员在用户管理子系统中选择相应用户，进入用户列表查看界面或在搜索框中输入关键用户信息进行检索，提取符合条件的用户列表。  2.找到相应用户后点击“编辑”按钮，进入编辑用户详细信息页面  3.在用户编辑页面中，根据用户的身份信息，设置默认密码，点击保存。 |
| 可选操作流 | 管理员对所有用户的密码进行设定后，点击确认按钮，设置成功。 |

系统管理员能够对基本用户信息进行查询和相应的修改操作，可以在登陆用户信息管理系统之后对用户信息进行操作，如编辑用户名称、密码、联系方式、地址等信息，用户信息管理用例描述如表3-3所示。

表3-4 用户信息管理用例描述

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 用户信息管理 |
| 标识符 | XY02 |
| 用例描述 | 描述了系统管理员使用本系统基本信息管理子系统对用户信息进行管理整个过程 |
| 参与者表 | 系统管理员 |
| 前置条件 | 系统管理员已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.管理员在用户管理子系统中选择相应用户，进入用户列表查看界面或在搜索框中输入关键用户信息进行检索，提取符合条件的用户列表。  2.找到相应用户后点击“编辑”按钮，进入编辑用户详细信息页面  3.在用户编辑页面中，输入需要修改的用户信息，点击保存。 |
| 可选操作流 | 管理员对用户信息进行修改后，点击保存按钮，修改成功。 |

### 3.2.3 顾客管理

顾客管理是超市销售环节中的重要部分，业务员可以对顾客基本信息进行管理，进行查询修改等操作。顾客基本数据结合订单数据成为了用户数据分析中的重要部分，具体顾客管理用例图的设计如图3-8所示。



图3-8 顾客管理用例图

用户角色的主要功能：新增顾客、修改顾客信息、查看顾客信息、删除顾客、顾客数据分析，该部分所有操作都需要操作员进行登陆。顾客数据分析主要包括市场细分数据、用户购买习惯、用户消费等级等数据。

顾客信息管理主要是业务员对顾客信息进行查询和修改，增加顾客、删除顾客等操作。顾客信息管理用例描述如表3-5所示。

表3-5 顾客信息管理用例说明

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 顾客信息管理 |
| 标识符 | GK01 |
| 用例描述 | 描述了业务员使用顾客管理子系统对顾客信息进行管理整个过程 |
| 参与者表 | 业务员 |
| 优先级别 | 1 |
| 状态 | 进行中 |
| 前置条件 | 业务员已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.业务员在系统登陆界面中输入自己的账号和密码，进入系统后，点击顾客管理模块。  2.在顾客管理模块中，查看用户列表，可对顾客相关信息进行查询。  3.当需要新增顾客时，点击添加顾客按钮，输入顾客基本信息，保存，即可添加顾客。  4.当顾客信息发生变化需要修改的时候，由业务员进行修改、删除等操作，并点击保存。 |

### 3.2.3 商品管理

商品管理是超市库存、销售管理环节中的重要部分，库管员可以对商品基本信息进行管理，进行查询、修改等操作。商品基本数据结合订单数据成为了商品数据分析中的重要部分，具体商品管理用例图的设计如图3-9所示。



图3-9 商品管理用例图

用户角色的主要功能：新增商品、商品品类管理、修改商品信息、查看商品信息、删除商品、商品数据分析，该部分所有操作都需要操作员进行登陆。商品数据分析主要包括商品销量排行、商品利润排行等数据。

商品信息管理主要是库管员对商品信息进行查询和修改，增加商品、删除商品等操作。商品信息管理用例描述如表3-6所示。

表3-6 商品信息管理用例说明

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 商品信息管理 |
| 标识符 | SP01 |
| 用例描述 | 描述了库管员使用商品管理子系统对商品信息进行管理整个过程 |
| 参与者表 | 库管员 |
| 优先级别 | 1 |
| 前置条件 | 库管员已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.库管员在系统登陆界面中输入自己的账号和密码，进入系统后，点击商品管理模块。  2.在商品管理模块中，查看商品列表，可对商品相关信息进行查询。  3.当需要新增商品时，点击添加商品按钮，输入商品基本信息，保存，即可添加商品。  4.当商品信息发生变化需要修改的时候，由库管员进行修改、删除等操作，并点击保存。 |

商品品类管理主要对商品所属品类信息进行操作，商品品类主要分为两级，一级品类和二级品类，其中，二级品类从属于一级品类。商品信息管理用例描述如表3-7所示。

表3-7 商品品类管理用例描述

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 商品品类管理 |
| 标识符 | SP02 |
| 用例描述 | 描述了库管员使用商品管理子系统对商品品类进行管理整个过程 |
| 参与者表 | 库管员 |
| 状态 | 进行中 |
| 前置条件 | 库管员已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.库管员在系统登陆界面中输入自己的账号和密码，进入系统后，点击商品管理模块。  2.在商品管理模块中，进入商品品类管理模块，查看商品品类列表，可对商品品类相关信息进行查询。  3.当需要新增商品品类时，点击添加商品品类按钮，输入商品品类信息，保存，即可添加商品品类。  4.当商品品类信息发生变化需要修改的时候，由库管员进行修改、删除等操作，并点击保存。 |

### 3.2.3订单管理

订单管理是超市销售环节中的重要部分，销售员可以对订单基本信息进行管理，进行查询修改等操作。订单数据是超市销售数据分析中最重要的部分，具体订单管理用例图的设计如图所示。



图3-10 订单客管理用例图

用户角色的主要功能：订单基本信息管理、发货管理、订单数据分析。

订单基本信息管理包括新增订单、修改订单信息、查看订单信息、删除订单。订单数据分析主要包括年度、月度订单销售情况等数据。订单管理描述见表3-8。

表3-8 订单管理用例描述

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 订单信息管理 |
| 标识符 | DD01 |
| 用例描述 | 描述了销售员使用订单管理子系统对订单信息进行管理整个过程 |
| 参与者表 | 销售员 |
| 优先级别 | 1 |
| 状态 | 进行中 |
| 前置条件 | 销售员已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.销售员在系统登陆界面中输入自己的账号和密码，进入系统后，点击订单管理模块。  2.在订单管理模块中，查看订单列表，可对订单相关信息进行查询。  3.当需要新增订单时，点击添加订单按钮，输入订单基本信息，保存，即可添加订单。  4.当订单信息发生变化需要修改的时候，由销售员进行修改、删除等操作，并点击保存。 |

订单发货是指当客户完成订单，仓库准备好订单商品之后，销售员根据客户所提供的地址信息进行发货，并将发货时间、发货方式、运费等发货信息添加到订单中。订单发货用例描述如表3-8所示。

表3-9 订单发货用例说明

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 订单发货 |
| 标识符 | DD02 |
| 用例描述 | 描述了销售员使用订单管理子系统对订单进行发货的整个过程 |
| 参与者表 | 销售员 |
| 状态 | 进行中 |
| 前置条件 | 销售员已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.销售员在系统登陆界面中输入自己的账号和密码，进入系统后，点击订单管理模块。  2.在订单管理模块中，查看订单列表，可对订单相关信息进行查询。  3.点击订单发货按钮，输入发货基本信息，保存，即可发货成功。  4.当订单发货信息发生变化需要修改的时候，由销售员进行修改、删除等操作，并点击保存。 |

### 3.2.3 数据分析

在超市的管理中，数据分析功能可用于分析影响超市商品价格的相关信息，例如：利润、价格、运输成本等，还可以根据具体的销售情况和库存情况，对相关商品的囤积和降价做出相应的决策。通过查询商品的采购记录和销售情况查询，利用核算储备功能，帮助超市解决商品保有量的问题，真正起到辅助的作用，同时还可以根据具体的销售情况来做相应的活动和措施，从而使其超市的销售处在一个较好的状态。

同时，超市管理层还需要分析每日销售情况统计、商品销售排行、顾客消费和退货状态。根据时间段和产品编号，查询超市库存的盈亏，并根据超市订单表进行统计分析。通过对超市经营情况的可视化数据分析，制定相应的处理措施和销售策略，尽量使超市的利益最大化。

具体的数据智能分析用例图的设计如图3-11所示。



图3-11 数据分析用例图

用户角色的主要功能：销售情况分析、商品情况分析、顾客情况分析等。

表3-10 数据分析用例说明

|  |  |
| --- | --- |
| 描述项 | 说明 |
| 用例名称 | 顾客信息管理 |
| 标识符 | SJ01 |
| 用例描述 | 描述了超市高层管理人员使用数据分析模块对超市销售情况进行分析的整个过程 |
| 参与者表 | 高层管理人员 |
| 状态 | 进行中 |
| 前置条件 | 用户已经登陆 |
| 后置条件 | 系统给出操作成功的对话框 |
| 基本操作流 | 1.超市高层管理人员登录系统后查询相应的销售情况以及商品相关情况(如:销售情况、价格等)  2.查询和分析完毕后，通过该部分制定相应的采购订单和进销存信息表，并进行审核。 |

## 3.3 系统非功能性需求分析

### 3.3.1性能需求

通过对超市实际运营状况的调查，系统需要很好地完成一般超市业务管理工作，并可以根据具体需要，保证多用户同时登陆。超市管理系统平均每天需要完成和处理数千次操作。为了保证系统的运行效果和可靠性，系统的响应时间最长为5秒。

（1）系统的可靠性：它可以稳定运行，并且指定时间内的故障率必须低于某个阈值。此外，整个系统还应对系统运行期间发生的各种错误提供相应的提示。也就是说，系统应该具有相对完整的错误处理机制，以方便用户的下一步操作。 同时，整个系统还需要具有自动备份和转储处理机制，当系统突然发生时，导致程序错误或崩溃等。它可以通过以前的自动备份及时回复相关数据，从而确保系统的可靠和高效运行。

（2）系统的效率：当整个系统处于其运营的高峰时，相关的简单搜索需要在1秒内回复。其他复杂操作也需要在2秒内响应。

（3）系统安全性：整个系统的安全性能不仅需要从网络安全的角度进行研究， 还需要从系统的软件，数据库安全性中考虑，在系统访问时，可以设置严格的访问机制，用户不允许超出系统的权限和级别登录系统。

（4）系统的专用性：需要具有友好的界面，易于用户理解和学习。

### 3.3.2开发及运行环境

操作系统：macOS Mojave

数据库：MongoDB

开发工具：IEAD

应用平台:：Nginx

## 3.4 本章小结

本章从超市的日常业务流程开始，考虑采购、库存和销售环节中的各种因素。结合超市管理系统的需求，在功能和非功能需求方面进行了详细的需求分析。它为系统的设计和实现提供了坚实的基础。也为本文下一步工作提供理论依据。

**第4章 系统的设计**

## 4.1 系统总体设计

通过初步调查研究，根据超市管理系统的设计目标和需要实现的功能，超市管理系统的功能模块主要包括：数据分析功能模块，基本信息管理功能模块，顾客管理功能模块，商品管理功能模块，订单管理功能模块等五大功能模块。

其中，数据分析模块包含商品数据分析、销售数据分析和顾客数据分析功能。基本信息管理模块包括用户管理、修改密码、权限管理等功能。顾客管理模块包括新增顾客、修改顾客信息、删除顾客等功能。商品管理模块包括新增商品、修改商品信息、删除商品和商品品类管理等功能，在商品品类管理中又包括对商品一级品类和二级品类的增、删、改、查等操作。订单管理模块包括新增订单、修改订单信息、删除订单以及订单发货等操作。顾客、商品和订单信息数据为数据分析模块提供了数据基础，四大信息管理模块和数据分析模块共同构成了超市管理系统。系统功能模块结构图如图4-1所示。



图4-1 超市管理系统功能模块图

用户通过登录界面输入帐号和密码进入系统，系统将用户输入的信息与后台数据库中的信息进行比较和分析，确定用户的身份是否合法。只有当用户具有相应的操作权限，才能实施基本信息管理，客户管理，商品管理，订单管理和数据分析等相关操作[20]。超市管理系统活动图如图4-2所示。



图4-2 超市管理系统活动图

## 4.2 系统模块设计

### 4.2.1 数据分析模块设计

超市管理系统的数据分析模块实现了对超市日常经营中的各种销售相关数据进行分析的功能，从而为超市管理层提供参考决策的依据。超市数据分析主要包括商品信息数据分析、销售分析和客户数据分析。商品信息数据分析主要包括商品销售情况分析和商品基本信息分析。销售情况分析主要包括超市订单的分析，比如年度销售额变化情况、订单量、利润变化情况等。客户数据分析主要包括不同客户市场细分、顾客黏度和客户购买水平等。

超市数据分析功能模块如图4-3所示。



图4-3 数据分析模块图

### 4.2.2 基本信息管理模块设计

超市管理系统的基本信息管理模块实现了对超市各种经营数据的管理。超市基本信息管理主要是对系统使用者信息、密码和权限的管理，包括密码修改和权限设置功能。基本信息管理中的用户管理主要是对用户，即超市管理系统使用者的基本信息进行管理、用户的创建和删除等。密码更改主要是为超市员工忘记密码后提供重置密码的操作。权限管理是超市管理系统针对不同类型的用户提供不同功能的操作，如销售员只能进行订单相关的操作，而不能修改顾客信息。超市管理基本信息管理模块如图4-4所示。



图4-4 基本信息管理模块图

超市管理系统基本信息管理的主要流程是，当用户想要执行基本信息管理操作时，通过登录界面输入用户名和密码，并调用后台接口，对输入信息进行检验，然后根据不同的用户角色，进入相应的信息管理界面，进行用户信息管理、密码修改、权限管理等相应的操作。任何想要执行操作的用户都必须首先登录系统。系统接受登录信息后，根据后台数据库中的相应信息对用户信息进行比较，根据不同的用户权限进行不同操作。这可以确保明确的分工，并防止由于非授权用户的误操作引起的一系列问题。同时，它还提高了系统的安全性和可靠性。用户应在完成相应操作后退出系统，以防账号非法使用。超市管理系统超市信息管理活动图如图4-5所示。



图4-5 基本信息管理功能活动图

### 4.2.3 顾客管理模块设计

超市客户管理主要包括超市销售中各种客户的信息管理，如添加客户，修改客户信息和删除客户等。新增顾客是当业务员发展新客户时，将顾客信息添加至超市管理系统中，便于加强客户与超市之间的沟通，当顾客信息发生变化时，可对相关数据进行修改。当顾客要求删除信息时，可对顾客进行删除操作。超市顾客管理功能模块如图4-6所示。



图4-6 顾客管理模块图

超市管理系统基本信息管理的主要流程是，当用户想要执行操作时，通过登录界面输入用户名密码，在验证用户权限之后，进行顾客管理页面。超市管理系统顾客管理活动图如图4-7所示。



图4-7 顾客管理功能活动图

### 4.2.4 商品管理模块设计

商品管理模块实现与超市商品有关的信息的管理。商品管理主要包括商品信息管理，如添加新商品，修改商品信息，删除商品等。商品管理功能模块见图4-8。



图4-8 商品管理模块图

超市管理系统商品管理的主要流程是，当用户想要执行操作时，通过登录界面输入用户名密码，在验证用户权限之后，进行商品管理页面。超市管理系统商品管理活动图如图4-9所示。



图4-9 商品管理功能活动图

### 4.2.5 订单管理模块设计

订单管理模块实现与超市订单相关的信息的管理。订单管理主要包括对超市中各种销售订单的信息管理，如添加订单、修改订单信息和删除订单。超市订单管理功能模块如图4-10所示。



图4-10 订单管理模块图

超市管理系统商品管理的主要流程是，当用户想要执行操作时，通过登录界面输入用户名密码，在验证用户权限之后，进行商品管理页面。超市管理系统订单管理活动图如图4-11所示。



图4-11 订单管理功能活动图

## 4.3 数据库设计

为了更好地处理超市管理系统的后台数据，将超市日常经营中产生的大量的数据通过相应的管理软件进行整合，使之构成一个便于操作和管理的整体，这个整体被称为数据库系统。数据库系统可以实现对数据库中信息的存储、更新、删除以及各种类型的查询操作。数据库是信息化超市管理系统的后端数据操作的基础，在信息管理和数据分析部分起着非常重要的作用。根据系统功能设计要求和模块划分，对超市管理系统中包含的数据信息进行了建模与分析，得出超市管理系统的数据表主要为用户表、商品表、商品品类表、订单表、订单商品表、数据分析表等。超市管理系统采用E-R图进行数据库概念结构的设计，系统E-R图如图4-12所示。



**图4-12 超市管理系统E-R图**

## 4.3.1 数据库建模

在对数据库建模时，我们首先采用MySQL数据库作为建模对象，以作为后续采用MongoDB数据库系统进行建模的参考。通过对关系型数据库MySQL的建模与分析，能够得到不同数据表之间的关联，其建模也更加具体形象，对后续采用非关系型数据库建模起到了很大帮助。

4.3.1.1用户信息设计

超市管理系统的用户信息包括用户ID、用户名、用户密码、联系方式、地址、用户权限等信息，用户E-R图如图4-13所示。



图4-13 用户E-R图

在用户表中，用户ID作为主键，能够唯一标识用户。用户联系方式和地址信息可为空，用户表图表4-1所示。

表4-1 用户表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 字段名称 | 字段类型 | 主键/索引 | 非空 | 字段说明 |
| 1 | userid | varchar | key | Yes | 用户ID |
| 2 | username | varchar |  | Yes | 用户名 |
| 3 | password | varchar |  | Yes | 用户密码 |
| 4 | tele | varcahr |  |  | 联系方式 |
| 5 | address | varchar |  |  | 地址 |
| 6 | right | varchar |  | Yes | 用户权限 |

4.3.1.1商品信息设计

超市管理系统的商品信息包括商品ID、商品类别、商品子类别、商品名称、商品价格、进货价格、商品规格、商品产地、商品备注等信息，商品E-R图如图所示，商品表如表4-2所示。



图4-14 商品E-R图

在商品表中，商品ID作为主键，能够唯一标识商品。商品类别与商品品类表进行关联，只可选取商品品类表中品类。商品价格和进货价格为双精度浮点型。商品规格、产地和备注可为空。商品表如表4-2所示。

表4-2 商品表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 字段名称 | 字段类型 | 主键/索引 | 非空 | 字段说明 |
| 1 | goodid | varchar | key | Yes | 商品ID |
| 2 | category | varchar |  | Yes | 商品类别 |
| 3 | subCategory | varchar |  | Yes | 商品子类别 |
| 4 | goodName | varcahr |  | Yes | 商品名称 |
| 5 | price | double |  | Yes | 商品价格 |
| 6 | buyprice | double |  | Yes | 进货价格 |
| 7 | Type | Varchar |  |  | 商品规格 |
| 8 | Address | Varchar |  |  | 商品产地 |
| 9 | Note | Varchar |  |  | 商品备注 |

超市管理系统的商品品类信息包括品类ID、品类名称、品类描述等信息，品类E-R图如图4-15所示。

。

图4-15 商品品类E-R图

在商品品类表中，品类ID作为主键，能够唯一标识品类。品类ID作为外键与子品类表中一级品类ID关联，形成了一级品类、二级品类和商品之间的的层次关系。商品品类表如表4-3所示。

表4-3 商品品类表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 字段名称 | 字段类型 | 主键/索引 | 非空 | 字段说明 |
| 1 | categoryid | varchar | key | Yes | 品类ID |
| 2 | Categoryname | varchar |  | Yes | 品类名称 |
| 3 | Detail | varcahr |  |  | 品类描述 |

超市管理系统的商品子品类信息包括品类ID、子品类ID、子品类名称等信息，品类E-R图如图4-16所示。

。

图4-16 商品子品类E-R图

在商品子品类表中，子品类ID作为主键，能够唯一标识子品类。品类ID与品类表中一级品类ID关联，商品子品类表如表4-4所示。

表4-4 商品子品类表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 字段名称 | 字段类型 | 主键/索引 | 非空 | 字段说明 |
| 1 | categoryid | varchar |  | Yes | 品类ID |
| 2 | subcategoryid | varchar | key | Yes | 子品类ID |
| 3 | Subcategoryname | varcahr |  | Yes | 子品类名称 |

4.3.1.1用户信息设计

超市管理系统的顾客信息包括顾客ID、顾客名称、顾客类型、超市、省份、国际、邮政编码等信息，顾客E-R图如图4-17所示。



图4-17 顾客E-R图

在顾客表中，顾客ID作为主键，能够唯一标识顾客。顾客类型只能在Customer、Corporate和Home Office三种类型之间选择，将作为数据分析模块中市场细分的重要依据。顾客城市、省份和国家作为顾客地址的组成部分，邮政编码可空。顾客表如表4-4所示。

表4-5 顾客表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 字段名称 | 字段类型 | 主键/索引 | 非空 | 字段说明 |
| 1 | customerid | varchar | key | Yes | 顾客ID |
| 2 | customerName | varchar |  | Yes | 顾客名称 |
| 3 | segment | varchar |  | Yes | 顾客类型 |
| 4 | city | varcahr |  | Yes | 城市 |
| 5 | state | varcahr |  | Yes | 省份 |
| 6 | country | varcahr |  | Yes | 国家 |
| 7 | postalCode | Varchar |  |  | 邮政编码 |

4.3.1.1用户信息设计

超市管理系统的订单信息包括订单编号、订单日期、发货日期、邮寄方式、邮寄费用、顾客ID、市场、订单优先级等信息，订单E-R图如图4-18所示。



图4-18 订单E-R图

在订单表中，订单ID作为主键，能够唯一标识订单。订单日期和发货日期均为日期格式。顾客ID和顾客表联系，作为数据分析中顾客分析的基础。订单表如表4-6所示。

表4-6 订单表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 字段名称 | 字段类型 | 主键/索引 | 非空 | 字段说明 |
| 1 | orderid | varchar | key | Yes | 订单编号 |
| 2 | orderDate | date |  | Yes | 订单日期 |
| 3 | shipDate | date |  | Yes | 发货日期 |
| 4 | shipMode | varcahr |  | Yes | 邮寄方式 |
| 5 | customerID | varcahr |  | Yes | 顾客ID |
| 6 | market | varcahr |  | Yes | 市场 |
| 7 | shipCost | double |  |  | 邮寄费用 |
| 9 | orderPriority | Varchar |  |  | 订单优先级 |

超市管理系统的订单商品信息包括订单编号、商品ID、商品数量、折扣、金额、利润等信息，订单商品E-R图如图4-19所示。订单商品表如表4-7所示。



图4-19 订单商品E-R图

在订单商品表中，订单ID作为外键与订单表关联，一个订单中可以含有多个订单商品。商品ID和商品表关联，作为数据分析中商品销售分析的基础。由于商品定价一般不变，采用商品折扣进行促销处理，在一个订单中，不同订单大盘可以有不同的折扣，金额为实际销售金额，利润则需要结合商品进货价格和折扣金额进行分析。订单商品表如表4-7所示。

表4-7 订单商品表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 字段名称 | 字段类型 | 主键/索引 | 非空 | 字段说明 |
| 1 | orderid | varchar | key | Yes | 订单编号 |
| 2 | Goodid | date |  | Yes | 商品ID |
| 3 | quantity | date |  | Yes | 商品数量 |
| 4 | discount | varcahr |  | Yes | 折扣 |
| 5 | Sales | varcahr |  | Yes | 金额 |
| 6 | Profit | varcahr |  | Yes | 利润 |

从上述的 MySQL数据库系统建模的举例中可以看出， MySQL数据库符合 ACID特性，数据库的表要遵循严格的规范，对于一个非常简单的业务还需要多个表共同实现完整的数据存储要求。基于 MySQL数据库系统建立数据模型时，要存储文档的全部信息数据，需要建立多张与文档实体类相关联的数据表，当进行数据分析时，需要在用户、顾客、商品、订单信息表中使用数据，业务逻辑需要对多张表进行多次查询才能返回用户所需的数据，多个表的关联查询会极大降低系统的性能。对于超市管理系统，基于超市各类信息的数据分析模块是核心功能，也是用户使用频度最高的功能，这部分功能的性能将直接影响系统的用户体验。

使用 MongoDB数据库系统时，由于 MongoDB数据库系统支持无模式数据结构，并且支持使用 JSON格式来存储数据，因此 MongoDB数据库系统更适合适合存储嵌套结构的数据[21]。

当数据建模基于MongoDB数据库系统时，订单的所有数据只需要一组即可实现完整存储。执行数据读写和查询操作时避免了大量的冗余和复杂度，显著提高了查询效率。

以下是基于MongoDB数据库进行数据建模时的数据样例，MongoDB数据样例如图4-20所示。

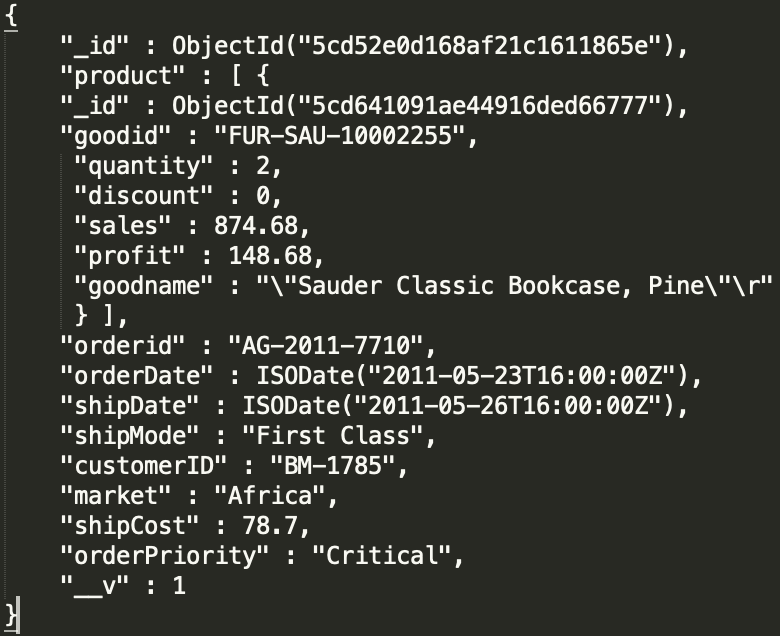


图4-20 MongoDB数据样例

从以上的基于MongoDB数据库系统进行数据建模的样例来看，数据建模更加清晰易懂，以JSON格式存储的完整文档数据更加紧凑。 MongoDB数据库系统实现的内嵌子文档的机制，可以实现深度惯性JOIN操作，只需要查询一次，就能通过深度遍历读取文档的所有信息[22]。和SQL关系型数据库系统对比，这种处理方式更加简单，也具备更高的读写性能。

## 4.3.2基于MongoDB数据库系统建模

在设计关键数据模型时，设计了多个实体类。基于MongoDB数据库系统支持嵌套数据模型的特性，可以把订单商品嵌套进订单类中，存储于一个嵌套的数据集合中。MongoDB数据库系统只需要这个集就可以完成系统核心订单模块的数据存储模型。

对于超市管理系统，订单数据是系统的核心数据，也是数据分析处理的重点。 订单数据的存储和管理效率直接关乎到超市管理系统的总体性能体验。订单数据是本文所开发的超市管理系统需要重点关注和处理的数据，因为订单数据在平台中的访问频次最高，使用最为频繁，订单数据的存储方式和读写性能会直接影响到系统的总体性能表现。

MongoDB数据库系统会为每一个文档生成一个全局唯一的\_id健值数据[23]。\_id健值的数据类型可以是任意类型[24]。在不做设置的情况下ObjectID对象将会是\_id健值的默认数据类型。ObjectID对象是一种轻量级的，满足在不同设备中全局唯一的方法来生成的。ObjectID生成后是一个24位的字符串数据，需要使用12个字节的存储空间来存储，算法会使用机器码、时间戳、PID和特定规则的计数器综合计算后得出新创建的ObjectID数据。计数器保证了同一时间内生成不同的数值，而PID则是用来表示产生这个ObjectID的进程的标识符号。MongoDB数据库系统允许每个进程在同一秒钟内最多生成16777216个全局唯一的ObjectID键值数据[25]，时间戳和随后的5个字节数据组合起来之后，就可以提供秒级别的唯一性支持。

[通过比较MySQL数据库建模和基于MongoDB数据库的数据建模，可以清楚地看到MongoDB数据库系统更适合于超市管理系统。](file:///C:\Users\Yuwen\Desktop\%E7%AC%AC%E5%9B%9B%E7%AB%A0\PaperPass-%E6%97%97%E8%88%B0%E7%89%88-%E6%A3%80%E6%B5%8B%E6%8A%A5%E5%91%8A\htmls\sentence_detail\75.html)

## 4.4 本章小结

本章主要介绍超市管理系统的总体设计，包括功能模块设计和相关数据库设计。超市管理系统的模块设计主要包括数据分析模块的设计和基本信息管理模块等多个模块的设计，例如顾客管理模块设计，商品管理模块设计和订单管理模块设计。超市管理系统的数据库系统设计，通过E- R关系图的形式给出了超市用户、顾客、商品、订单等几类实体及其之间相互关系，在充分考虑数据库完整性和一致性的基础上，给出了相关数据表以及相互之间的依赖关系。

**第5章 系统的实现**

## 5.1 系统部署

### 5.2.1 部署环境准备

由于超市管理系统的实施方式为B / S结构模式，因此部署时环境的配置分为服务器端和客户端环境配置。

（1）服务器配置

超市管理系统将采用三层B / S模式进行实现。因此在超市管理系统实际运行的过程中，会对服务器 WEB应用端和数据库应用端进行分离设置[26]，在配置过程中分开以确保安全。由于对于超市管理系统来说， WEB的服务器应用主要体现在对管理人员相关业务工作逻辑方面的处理，通过对管理员权限的限制，处理数据库的操作，并在服务器端调用后台接口进行来实现对数据库中数据的相应操作，同时要兼顾超市管理系统性能的运行需求。

（2）客户端配置

由于系统是采用三层B/S模式进行开发设计，因此用户只要运行WEB 浏览器就可以正常使用系统。但网页浏览器的运行情况也会受到计算机的性能的影响，为确保超市管理系统的正常运行，并能快速的回应客户的业务操作，所以对用户的计算机硬件配置也有一定要求。

### 5.2.2 数据准备

待整个系统环境完全搭建完毕之后，整个系统还要考虑和第三方数据接入口的标准规范性的相关调整，根据相关数据类型要求和设计标准，积极开展第三方软件数据导入工作，确保整个系统的数据完整性。同时还需要对系统所涉及到的业务系统数据，如订单数据、商品数据、客户数据等的正常修改操作提供完整的平台。

超市管理系统的基本数据来源为Tableau自带的超市订单数据，含有约5万条订单数据。Tableau是用于可视分析数据的商业智能工具[27]。用户可以创建和分发交互式和可共享的仪表板，以图形和图表的形式描绘数据的趋势，变化和密度。Tableau可以连接到文件，关系数据源和大数据源来获取和处理数据。该软件允许数据混合和实时协作，这使它非常独特。它被企业，学术研究人员和许多政府用来进行视觉数据分析。它还被定位为Gartner魔力象限中的领导者商业智能和分析平台[28]。

由于Tableau中自带数据与本次超市管理系统开发过程中所需要的数据存在一些差异，所以在使用该数据之前需要对数据进行一些处理。处理之后才能用于超市管理系统的实现中。超市订单数据源如表5-1所示。

**表5-1 数据源示例**

|  |  |  |  |
| --- | --- | --- | --- |
| Row ID | 32298 | 26341 | 25330 |
| Order ID | CA-2012-124891 | IN-2013-77878 | IN-2013-71249 |
| Order Date | 2012/7/31 | 2013/2/5 | 2013/10/17 |
| Ship Date | 2012/7/31 | 2013/2/7 | 2013/10/18 |
| Ship Mode | Same Day | Second Class | First Class |
| Customer ID | RH-19495 | JR-16210 | CR-12730 |
| Customer Name | Rick Hansen | Justin Ritter | Craig Reiter |
| Segment | Consumer | Corporate | Consumer |
| City | New York City | Wollongong | Brisbane |
| State | New York | New South Wales | Queensland |
| Country | United States | Australia | Australia |
| Postal Code | 10024 |  |  |
| Market | US | APAC | APAC |
| Region | East | Oceania | Oceania |
| Product ID | TEC-AC-10003033 | FUR-CH-10003950 | TEC-PH-10004664 |
| Category | Technology | Furniture | Technology |
| Sub-Category | Accessories | Chairs | Phones |
| Product Name | Plantronics CS510 - Over-the-Head monaural Wireless Headset System | Novimex Executive Leather Armchair, Black | Nokia Smart Phone, with Caller ID |
| Sales | 2309.65 | 3709.395 | 5175.171 |
| Quantity | 7 | 9 | 9 |
| Discount | 0 | 0.1 | 0.1 |
| Profit | 762.1845 | -288.765 | 919.971 |
| Shipping Cost | 933.57 | 923.63 | 915.49 |
| Order Priority | Critical | Critical | Medium |

## 5.2 系统功能实现

超市管理系统核心功能是对超市日常经营中所需的数据集中高效管理，通过管理员来操作该系统，实现对超市日常业务流程的处理。在用户登陆系统并具有相关权限时可对信息进行修改和查询。对于超市高层管理人员而言，可以在其他人员所管理的各类型信息的基础上，借助数据分析模块，对超市经营情况进行分析。用户信息管理、顾客信息管理、商品信息管理、订单信息管理和数据分析等具体的各个模块的实现情况如下所述。

### 5.2.1 用户登录

超市管理系统用户登录的主要操作步骤是用户需要通过登录界面输入用户名和密码。如果输入的用户和密码不匹配，将给出相应的错误信息，禁止非法用户登录系统。当输入信息被证实是合法的并且彼此匹配时，用户进入管理系统并根据用户权限选择不同的操作界面。这样，用户可以确保分工明确，任务明确，并且防止非授权用户由于误操作而在超市管理系统中非法添加，删除，修改或更新信息。同时，它还提高了超市管理系统中各种信息的安全性和稳定性，以及整个超市管理系统的安全性和可靠性。用户登录界面如图5-1所示。

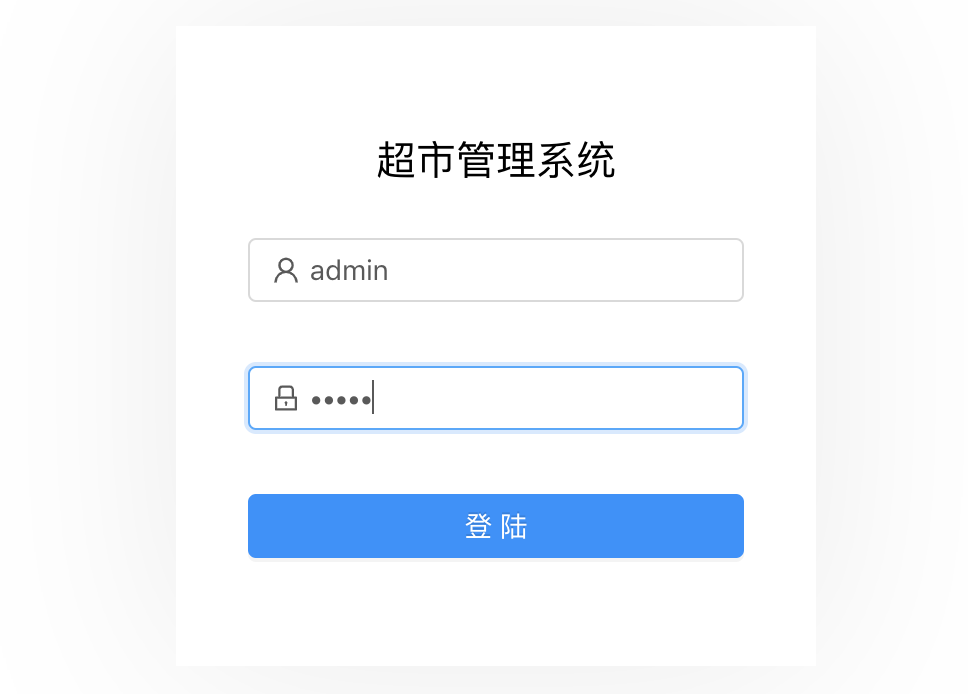


图5-1 用户登陆界面

用户登陆在系统使用者输入登陆信息之后，首先对输入信息合法性进行判断，若输入信息未达到要求，则输出相应的错误提示。在第一步输入信息符合规定的情况下，系统将调用后台接口，将输入信息传递到处理模块，与后台数据库中存储的信息进行比对，判断用户名和密码是否相匹配。如果用户名和密码匹配，则还需要进行用户权限检验，根据不同用户权限，进行不同的操作。若检验用户名和密码不匹配，则需要输出错误提示。用户登陆流程图如图5-2所示。



图5-2 用户登陆流程图

### 5.2.2 基本信息管理

超市管理系统的基本信息管理功能主要包括三个子部分：用户信息管理主要是指超市超级管理员对用户信息的管理，包括用户帐号、密码、联系方式、地址等信息。权限管理主要是指超市超级管理员对不同员工权利的管理。更改密码是当用户忘记系统登录密码时，管理员可以执行后台操作来重置新密码。

在基本信息管理中，可以点击添加用户按钮添加新用户，点击编辑按钮修改用户信息，点击删除按钮对用户进行删除操作，界面简洁友好，可直观显示用户各种信息数据，基本信息管理界面如图5-3所示。

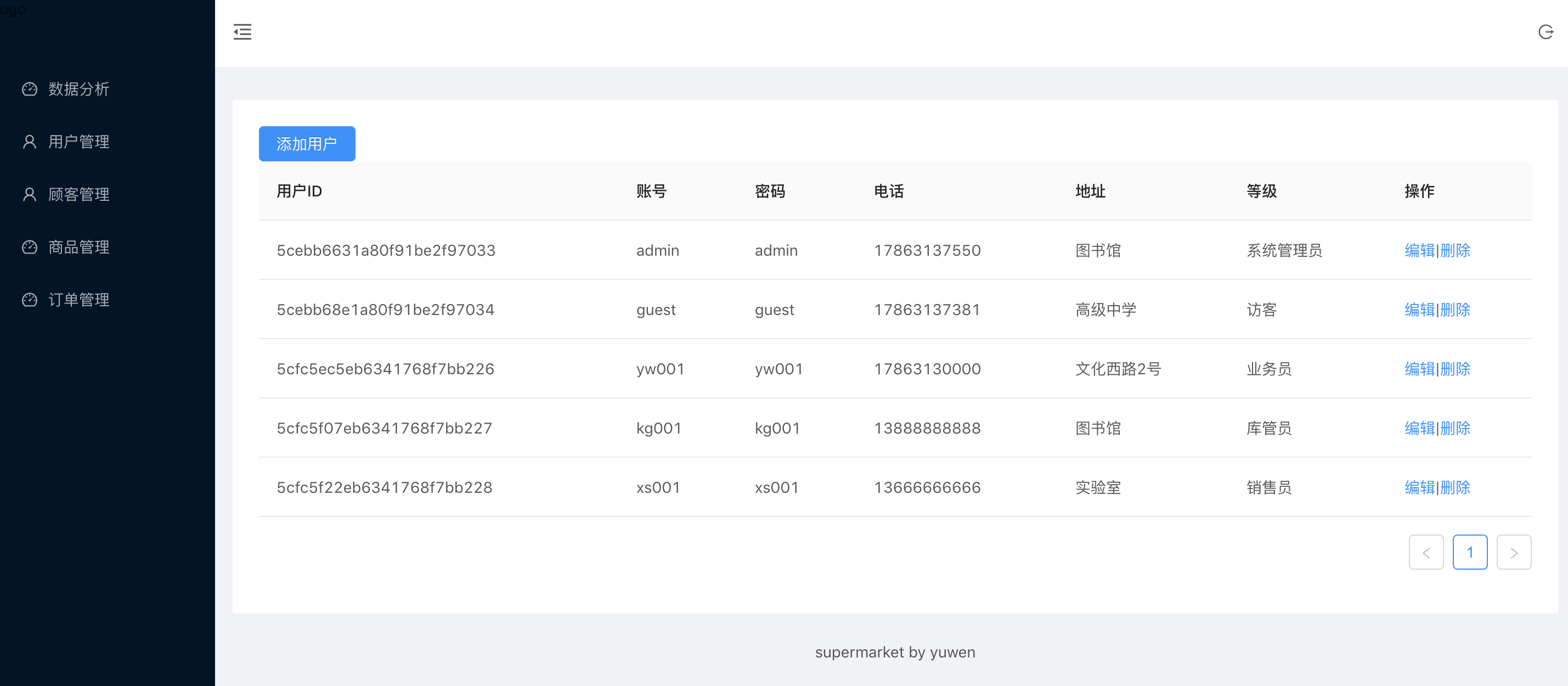


图5-3 基本信息管理界面

超市管理系统的基本信息管理功能的主要过程是用户通过登录界面输入帐号和密码。系统接受登录信息后，根据后台数据库中的用户账号信息对用户信息进行比较，确定操作者的身份权限后进行相关操作。基本信息管理流程如图5-4所示。



图5-4 基本信息管理流程图

### 5.2.3 顾客管理

超市管理系统顾客管理功能主要包括三个子部分：添加顾客主要是指超市营业员的操作添加目标客户信息。顾客信息管理，主要是指超市业务员对顾客信息进行相关修改，如顾客城市、省份、国家、邮政编码等信息。删除顾客是指当顾客要求删除信息时，可由超市业务员进行相关操作。顾客管理界面如图5-5所示。

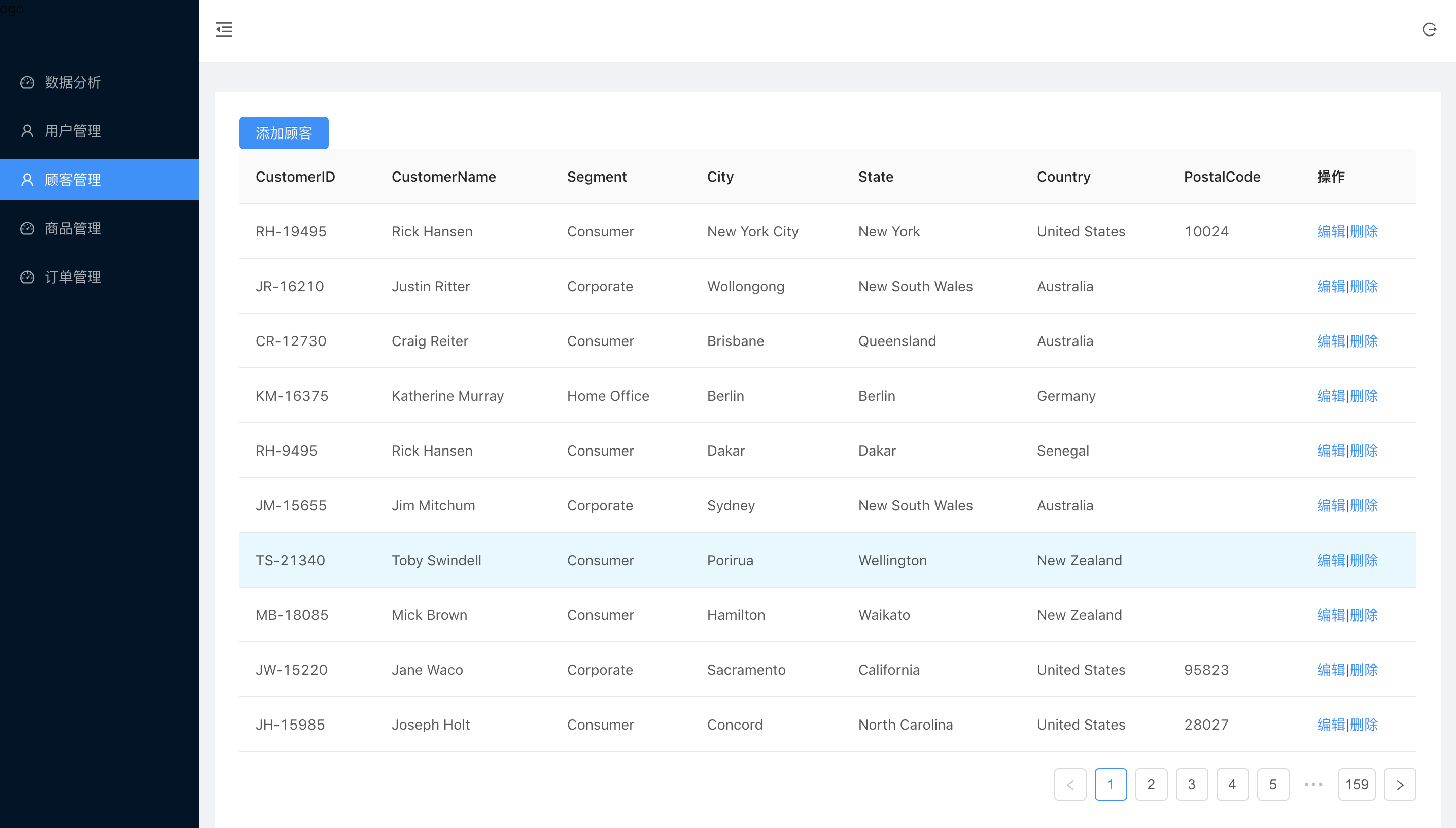


图5-5 顾客管理界面

顾客管理中点击添加顾客按钮，进入添加顾客界面。在表格中输入顾客相关信息，点击保存按钮，当输入信息检验进行校验通过之后可以成功添加新顾客，并提示顾客添加成功。顾客添加界面如图5-6所示。

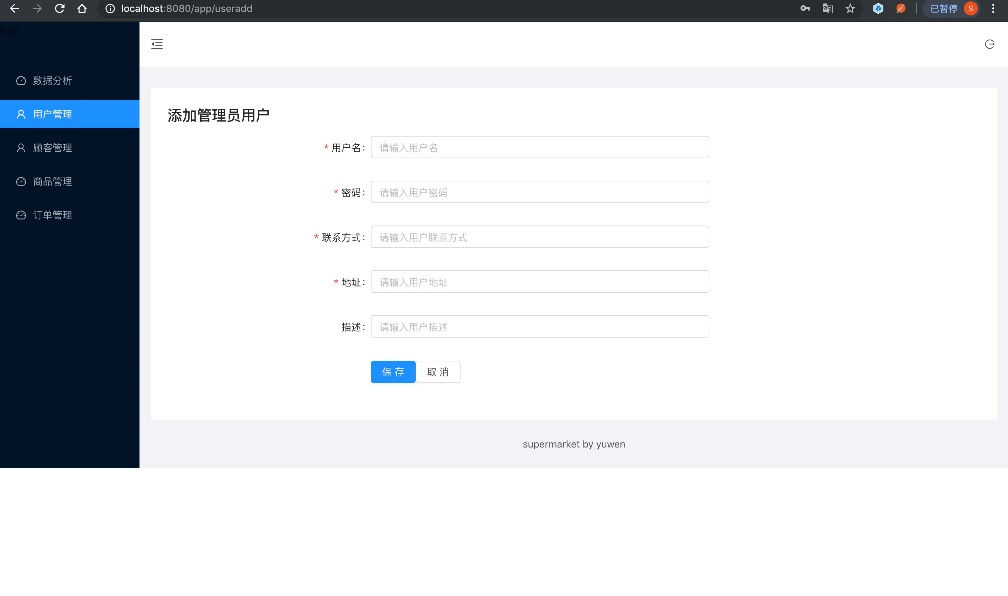


图5-6 顾客添加界面

超市管理系统顾客管理功能的主要流程是，用户通过登录界面输入账号、密码，系统接受登录信息后，根据后台数据库中用户信息其相应权限进行比照，根据不同的用户权限，登录操作不同的工作任务，验证为合法用户后方可进入顾客管理系统。顾客管理流程如图5-7所示。



图5-7 顾客管理流程图

### 5.2.4 商品管理

超市管理系统商品管理功能主要包括三个子部分：新增商品，主要是当超市采购员对新商品完成采购时，库管员对采购员所采购商品进行登记入库操作，需要录入商品基本信息，如商品ID、商品名称、商品类别、商品子类别、商品价格、商品进货价格等。商品信息管理，主要是指超市库管员根据实际情况对商品信息进行相关修改，如商品ID、商品名称、商品类别、商品子类别、商品价格、商品进货价格等。删除商品是指当商品要求被删除时，可由超市库管员进行相关删除操作。商品管理界面如图5-8所示。

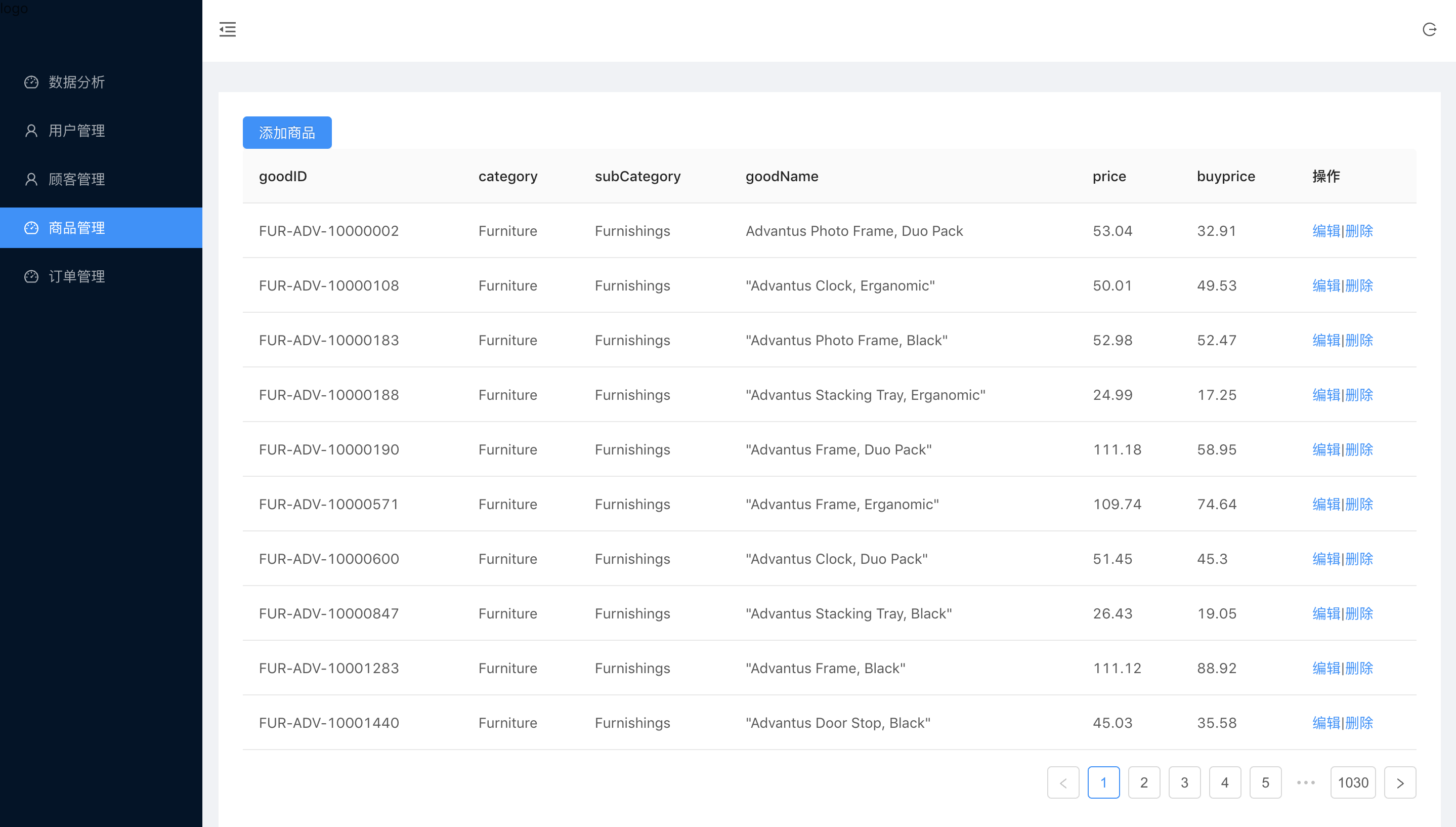


图5-8 商品管理界面

超市管理系统的商品管理功能的主要过程是用户通过登录界面输入帐号和密码。系统接受登录信息后，根据后台数据库中的用户账号信息对用户信息进行比较，确定操作者的身份权限后进行相关操作。商品管理流程如图5-9所示。



图5-9 顾客管理流程图

### 5.2.5 订单管理

超市管理系统订单管理功能主要包括三个子部分：新增订单，主要是当超市销售员完成订单销售时，对订单信息进行信息化管理，销售员对订单信息进行录入系统操作，需要输入有关订单的基本信息，例如订单 ID、客户 ID、订单日期、订单商品、订单优先级等。订单发货，主要是指超市销售员根据实际发货情况对订单信息进行修改，如发货时间、发货方式、邮寄费用等。删除订单是指当订单要求被删除时，可由超市销售员进行相关删除操作。订单管理界面如图5-10所示。

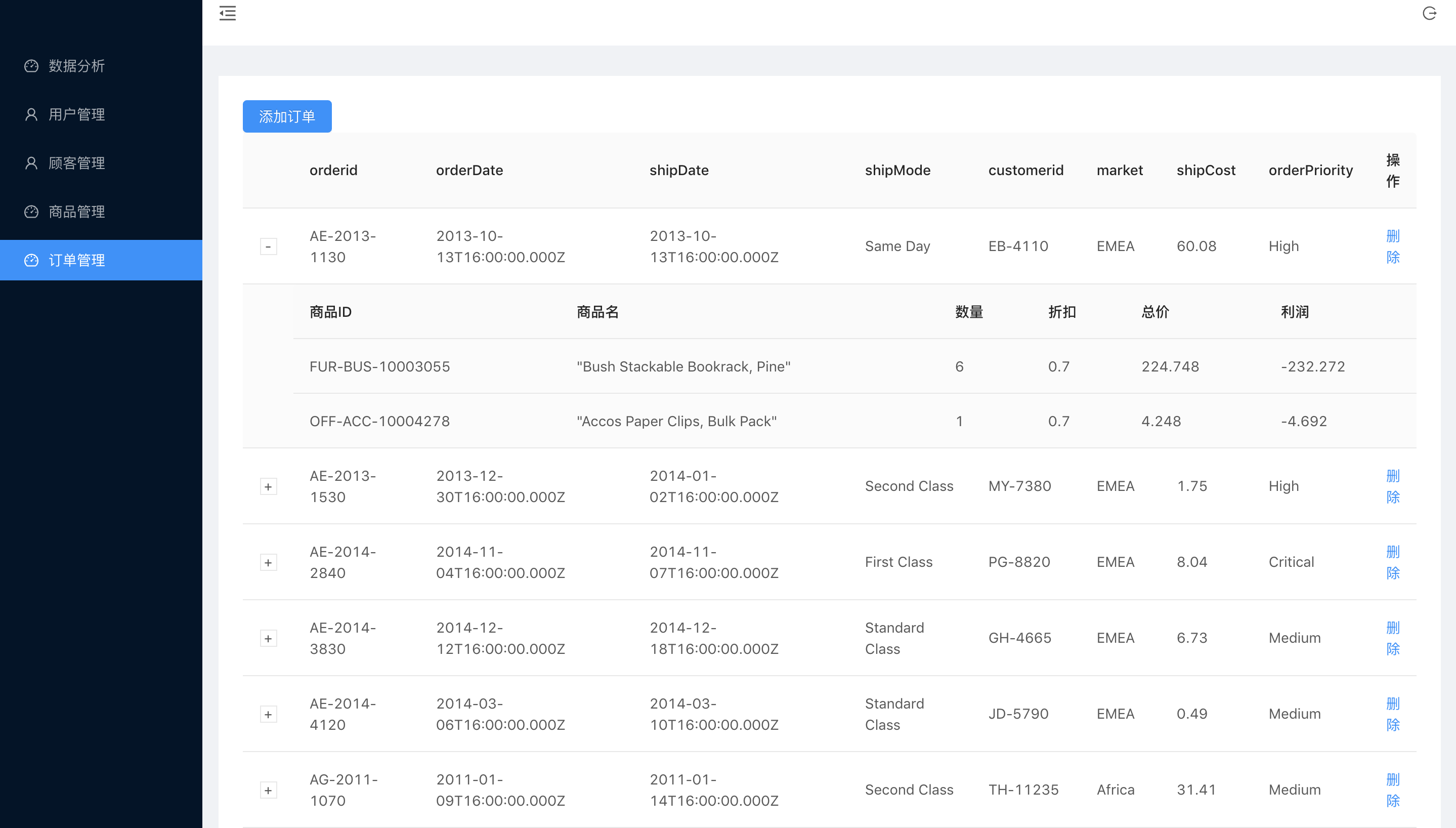


图5-10 订单管理界面

在订单管理界面中，点击左上角添加订单按钮即可进入添加订单界面，在订单信息表格中输入订单信息点击保存即可添加订单。订单添加界面如图5-11所示。

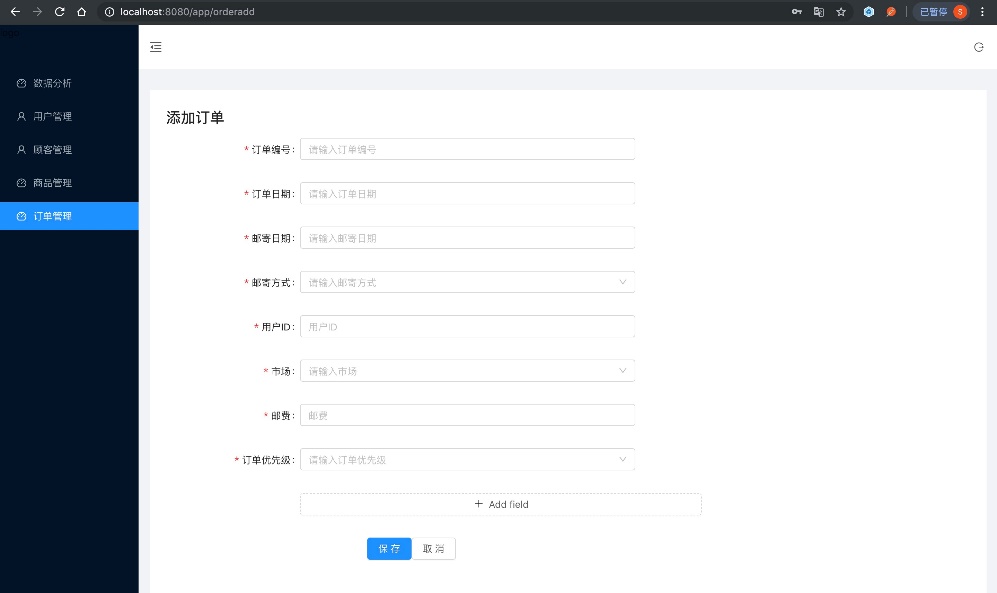


图5-11 订单添加界面

超市管理系统的订单管理功能的主要过程是用户通过登录界面输入帐号和密码。系统接受登录信息后，根据后台数据库中的用户账号信息对用户信息进行比较，确定操作者的身份权限后进行相关操作。订单管理流程如图5-12所示。



图5-12 订单管理流程图

### 5.2.6 数据分析

超市管理系统的数据分析功能主要包括四个子部分：基本数据分析，主要是指对超市日常经营中的基本信息进行分析展示，如总销售额、总订单量、商品数、用户数等信息。客户数据分析主要基于超市客户信息，分析超市销售中的客户相关数据，如客户市场细分。可通过对不同客户群体分析为不同类型的客户提供不同的销售策略。商品数据分析，主要是在超市商品信息的基础上，对超市销售情况中商品相关的数据进行分析，如超市商品分类比例，不同大类商品所含子类产品比重，该数据有助于超市对库存中商品比例进行分析。商品销售额分析，主要为超市销售过程中，根据订单销售数据，对商品销售额进行排列，可以直观显示商品销售额排名前十商品。商品销售利润分析，主要为超市销售过程中，根据订单销售数据，对商品销售利润进行排列，可以直观显示商品销售利润前十商品。商品销量分析，主要为超市销售过程中，根据订单销售数据，对商品销量进行排列，可以直观显示商品销售量前十商品。根据不同类型的商品分析，可以获得不同的分析结果，并协助超市管理决策。订单数据分析，主要是指对超市销售订单数据进行分析，如不同时间段超市订单量、订单销售额、订单利润的变化，能够为超市季节性销售战略提供参考意见，为超市经营变化情况提供分析。数据分析界面如图5-13所示。

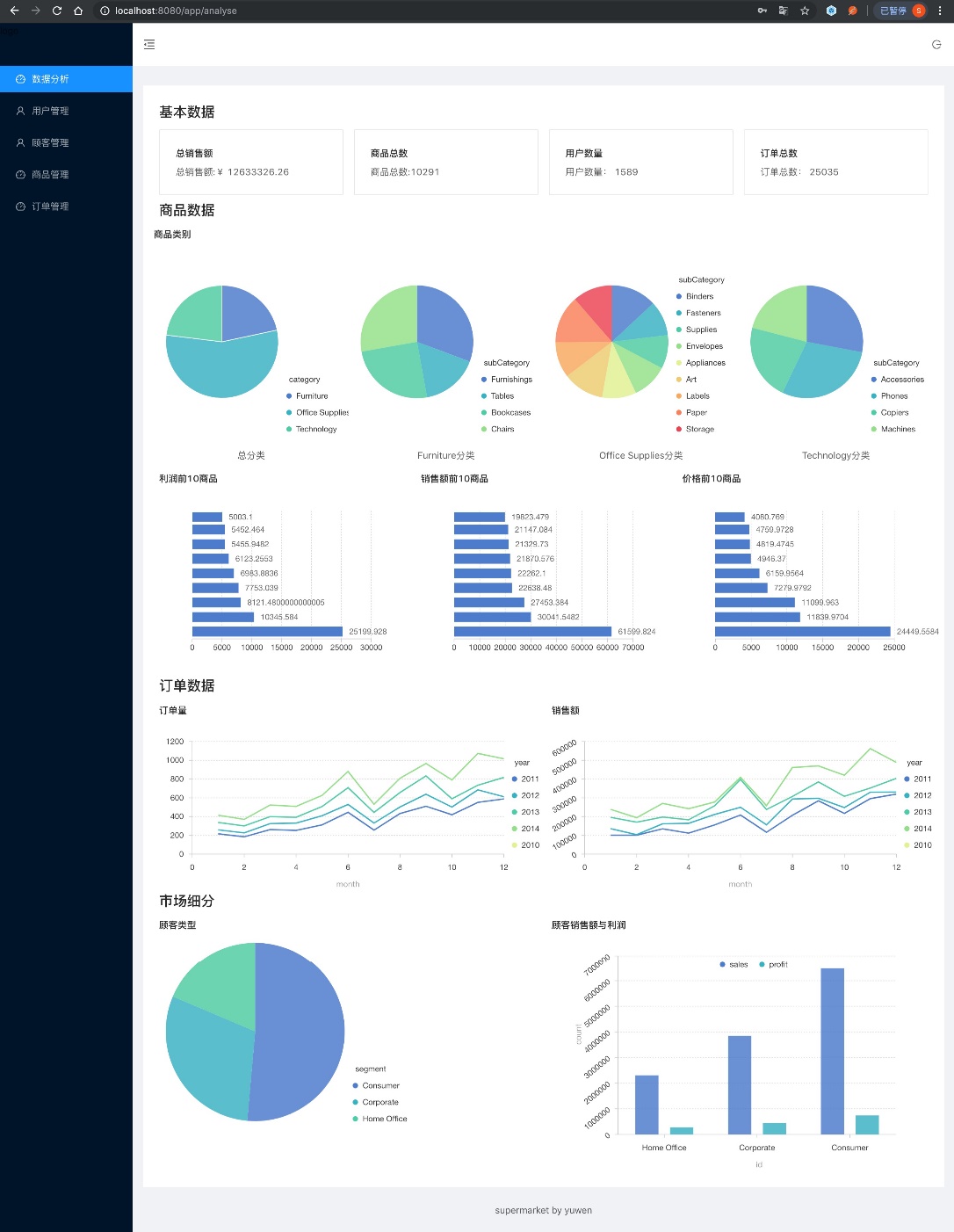


图5-13 数据分析界面

超市管理系统的数据分析功能的主要过程是用户通过登录界面输入帐号和密码。系统接受登录信息后，根据后台数据库中的用户账号信息对用户信息进行比较，确定操作者的身份权限后进行相关操作。数据分析流程如图5-14所示。



图5-14 数据分析流程图

## 5.3 本章小结

本章详细介绍了超市管理系统的平台实现和详细设计，分别介绍了各个功能模块的实现过程，并给出了操作界面设计和基本操作流程图，展示了超市管理系统的初步成果。

**第6章 系统的测试**

## 6.1 系统测试目标

软件测试是在执行系统程序期间发现问题。为了验证软件是否符合设计要求，或找到预期结果与实际结果之间的差异[29]。软件开发系统在投入实际使用之前应进行测试。在系统开发周期的每个阶段都使用技术检查，但无法避免偏差和错误。 在系统正式使用之前，尽可能多的找出问题，以便及时发现和纠正。通过软件测试，可以在开发之前判断开发的软件系统是否已达到功能和性能相关的要求。软件测试是软件开发系统确保完成质量的一个重要组成部分。软件测试代表了需求分析、系统设计、系统实现的最终复审阶段。

有许多类型的软件测试及其技术实现方法。根据需要可分为静态测试方法和动态测试方法；根据功能，可分为白盒测试和黑盒测试方法。为了充分满足系统功能的实际需要，超市管理系统采用白盒测试方法和黑盒测试方法，针对典型的功能测试和性能测试测试用例，根据测试结果验证系统功能和逻辑的正确性[30]。

## 6.2 系统测试内容

执行用户登录测试，主要是验证在登录界面中输入不同种类的用户名和密码，验证系统是否针对不同情况提供不同的错误消息，例如用户名不能为空。密码不能为空，用户名和密码不匹配，非法用户无法进入系统。

基本信息管理测试主要是验证用户信息管理，密码重置和权限管理功能是否可以正常运行。顾客管理模块测试主要是检查添加新顾客，修改顾客信息和删除顾客的功能是否可以正常运行。商品管理模块测试主要是验证添加新商品，商品信息管理和商品类别管理的功能是否能够正常运行。数据分析模块测试，主要是检验系统对超市订单数据、商品数据、顾客数据的分析是否高效准确。

## 6.3 系统功能测试

功能测试主要负责对系统的各项需求、功能以及系统配置的完整性和准确性等进行验证和测试。本节主要对整个系统中的基本信息管理模块、顾客管理模块、商品管理模块、订单管理模块以及数据分析模块这样五大功能的测试情况进行了详细的分析，具体测试情况如下所述。

### 6.3.1 基本信息管理功能测试

基本信息管理模块共设计了3个功能，分别是用户信息管理、密码设置、权限管理。因此设计如下用例来验证模块是否满足所要求的功能，需要前提条件为系统管理员登陆，测试用例如表6-1所示。

表6-1 基本信息管理功能测试用例

|  |  |  |
| --- | --- | --- |
| 输入/动作 | 预期的输出/响应 | 实际情况 |
| 点击添加用户，输入用户信息，不输入用户权限，点击保存。 | 提示请输入用户权限 | 符合 |
| 点击添加用户，输入完整用户信息，点击保存。 | 提示用户添加成功 | 符合 |
| 点击添加用户，点击取消按钮。 | 回到用户列表界面 | 符合 |
| 选择用户，点击编辑修改密码，输入用户新密码，点击保存。 | 提示密码修改成功 | 符合 |
| 选择用户，点击删除。 | 提示删除确认框 | 符合 |

### 6.3.2 商品管理功能测试

商品管理模块共设计了4个功能，分别是商品信息管理、新增商品、商品品类管理、删除商品。因此设计如下用例来验证模块是否满足所要求的功能，需要前提条件为库管员登陆，测试用例如表6-2所示。

表6-2 商品管理功能测试用例

|  |  |  |
| --- | --- | --- |
| 输入/动作 | 预期的输出/响应 | 实际情况 |
| 点击添加商品，输入商品信息，点击保存。 | 提示保存成功 | 符合 |
| 选择商品，点击编辑按钮，输入需要修改信息，点击保存。 | 提示保存成功 | 符合 |
| 选择商品，点击编辑按钮，输入需要修改信息，点击取消。 | 提示保存取消，返回商品列表界面 | 符合 |
| 选择商品，点击删除按钮。 | 提示二次确认窗口 | 符合 |
| 选择商品，点击删除按钮，在确认框中点击取消删除。 | 提示删除商品成功 | 符合 |
| 点击添加商品，输入商品信息，点击保存。 | 提示保存成功 | 符合 |

### 6.3.3 顾客管理功能测试

顾客管理模块共设计了3个功能，分别是新增顾客、顾客信息管理、删除顾客。因此设计如下用例来验证模块是否满足所要求的功能，需要前提条件为业务员登陆，测试用例如表6-3所示。

表6-3 顾客管理功能测试用例

|  |  |  |
| --- | --- | --- |
| 输入/动作 | 预期的输出/响应 | 实际情况 |
| 点击添加顾客，输入顾客信息，点击保存。 | 提示保存成功 | 符合 |
| 选择顾客，点击编辑，输入需要修改信息，修改用户名为空，点击保存。 | 提示用户名为空 | 符合 |
| 选择顾客，点击编辑，输入需要修改信息，点击保存按钮。 | 提示保存成功 | 符合 |
| 选择顾客，点击删除，在确认框中点击确认删除。 | 提示删除顾客成功 | 符合 |
| 选择顾客，点击删除，在确认框中点击取消删除。 | 返回顾客列表 | 符合 |

### 6.3.4 订单管理功能测试

订单管理模块共设计了4个功能，分别是创建订单、订单信息管理、订单发货、删除订单。因此设计如下用例来验证模块是否满足所要求的功能，需要前提条件为销售员登陆，测试用例如表6-4所示。

表6-4 订单管理功能测试用例

|  |  |  |
| --- | --- | --- |
| 输入/动作 | 预期的输出/响应 | 实际情况 |
| 点击创建订单，输入订单信息，输入订单商品信息，点击保存。 | 提示保存成功 | 符合 |
| 选择订单，点击编辑发货，输入订单发货信息，点击保存。 | 提示保存成功 | 符合 |
| 选择订单，点击删除，在确认框中点击确认删除。 | 提示删除成功 | 符合 |

### 6.3.5 数据分析功能测试

数据分析模块共设计了4个功能，分别是基本信息分析、顾客数据分析、商品数据分析、订单数据分析。由于数据分析是以其他几个信息管理模块的数据作为基础，因此还需要测试数据关联的正确性，测试是否能够及时获取数据更新。因此设计如下用例来验证模块是否满足所要求的功能，需要前提条件为超市高层管理人员登陆，测试用例如表6-5所示。

表6-5 商品管理功能测试用例

|  |  |  |
| --- | --- | --- |
| 输入/动作 | 预期的输出/响应 | 实际情况 |
| 在商品中删除一个商品。 | 数据分析中商品总数减少 | 符合 |
| 创建一个订单，输入订单商品信息并保存。 | 数据分析中订单销售额发生变化 | 符合 |
| 在销售订单分析中选择某一年数据 | 显示该年销售情况 | 符合 |
| 在商品分析中选择某一类商品 | 显示该类商品占比 | 符合 |

## 6.4 系统非功能性测试

在系统非功能性测试中，主要对系统性能进行了测试，性能测试是指系统面向海量数据时对系统运行效率和运行速度的测试。

（1）性能测试目标为：确保系统在运行期间处理海量数据，满足以下要求：在正常的业务范围内，系统可以有效地接收、处理、保存和其他数据；在数据量很大的情况下，系统仍然可以完成接收、处理和保存数据的操作。

（2）测试方法为：通过测试用例和测试数据，用系统的正常数据量测试系统的运行状态；使用主机对海量数据的运行状态进行定期和重复的测试；使用多个主机执行上述类似的测试。

（3）测试完成标准为：在正常数据量下系统的高效可靠运行；在多业务操作时系统响应速度任处于可接受范围。

## 6.5 本章小结

本系统的测试主要采用用例测试方法来测试，在功能测试中，主要测试了基本信息管理、顾客管理、商品管理、订单管理和数据分析的功能。性能测试则主要以测试系统的稳定性和可靠性为主。经过本次测试，证明本系统基本满足了需求分析的要求。

**结 论**

超市管理系统对中小型超市的实际需求进行分析，对软件系统框架进行了设计进和研究。基于各功能模块的详细设计，最终实现了系统软件的开发。系统开发采用基于MREN的MongoDB、React、Koa、NodeJS框架技术，完成了MVC三层架构。 以 IntelliJ IDEA为开发工具，以React+AntD为前台，采用MongoDB作为数据库管理系统管理后台数据库，采用NodeJS作为基本编程语言开发。 超市管理系统功能主要包括：基本信息管理、顾客管理、产品管理、订单管理和数据分析等功能。本系统的核心功能为数据分析模块的设计与实现，该模块通过对超市经营数据进行多方面分析，能够为超市管理层决策通过辅助参考。

本文首先介绍课题背景及研究的目的和意义，对国内外发展现状进行分析，并对主要研究内容进行阐述。然后，介绍了在设计和实现超市管理系统中使用的一些关键技术栈和开发工具，并对基本数据源进行了简要介绍。在需求分析的基础上，进行了总体设计，详细阐述了相关功能模块的设计。并通过相关代码编程最终实现了超市管理系统的运行，最后对超市管理系统进行功能性测试，各模块的功能基本完成预期目标。

虽然超市管理系统基本实现了预期的模块和相应的功能，满足了一般中小型超市管理的要求。但是，由于本人的技术和时间的限制，系统仍然有一些令人不满意的地方。在未来的工作中，将进一步完善该系统中的一些功能，使系统的功能更加完善，丰富实用。

**参考文献**

1. 欧阳电平，陈潇怡. 论信息技术环境下会计信息系统的演进[J]. 武汉大学学报，2010 (3) :33-34.
2. 汤谷成. 企业财务管理学[M]. 北京：经济科学出版社，2008.7:75-79.
3. 李海明，朱保安，冯飞. 基于局域网的财务预结算管理系统设计与应用[J]. 电脑编程技巧与维护，2011(04) :50-52.
4. 李海明，朱保安，冯飞. 基于局域网的财务预算填报系统[J]. 电脑编程技巧与维护，2011(05) :101-103.
5. 廖芹等. 工业企业库存管理信息系统的设计和研究[J]. 华南理工大学学报，2010(5) :254-260.
6. 吴琼. 天扬超市进销存管理系统的设计与实现[D]. 西安电子科技大学,2016.
7. 郑少兰. 电子表格与SQLServer的结合应用[J]. 电脑知识与技术，2009 (35): 17-18
8. Gunawan, Armin Lawi, Adnan. Web Menggunakan Model View Controller (MVC) pada Framework Java Server Faces[J]. Scientific Journal of Informatics,2016,3(1).
9. Andrew Morgan. Introducing the MEAN and MERN stacks(20171.27)[2019.6.1]. https://www.mongodb.com/blog/post/the-modern-application-stack-part-1-introducing-the-mean-stack.
10. Subramanian V. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node[M]. Apress, 2017.
11. 冯自信. 基于react-native的移动零售进货系统的设计与实现[D]. 华中科技大学.2019.
12. Nimesh Chhetri. A Comparative Analysis of Node.js (Server-Side JavaScript)[D]. St.Cloud State University.2016.
13. 李汝佳,胡婧. 基于Nodejs的异步非阻塞服务器研究[J]. 计算机科学与应用, 2013,03(03).
14. Mongodb Inc. Mongodb, Inc. Files SEC Form 8-K, Current Report: (Dec. 21, 2018)[J]. Computers, Networks & Communications,2019.
15. Divya Mahajan, Cody Blakeney, Ziliang Zong. Improving the energy efficiency of relational and NoSQL databases via query optimizations[J]. Sustainable Computing: Informatics and Systems,2019.
16. 王伶俐,张传国. 基于NodeJS+Express框架的轻应用定制平台的设计与实现[J]. 计算机科学,2017,44(S2):596-599.
17. 毛英杰. 基于Koa框架的停车诱导系统[J]. 福建电脑,2019,35(01):143-144+125.
18. 杨明戊, 戴经国, 陈正铭. 基于Node.js+koa2的全景图上传与展示技术研究[J]. 电脑编程技巧与维护, 2018, 400(10):151-154.
19. 中国运筹学会企业运筹学分会. 中国企业运筹学2009(1)[M]. 四川：电子科技大学出版社.2009.
20. 高巍. 基于J2EE架构的佳家乐超市营销管理系统设计与实现[D]. 电子科技大学,2014.
21. 刘卓. 基于NoSQL的空间数据云存储的研究[D]. 河南大学,2014.
22. Botoeva E , Calvanese D , Cogrel B , et al. A Formal Presentation of MongoDB (Extended Version)[J]. 2016.
23. 任凯. 基于NoSQL海量数据分析引擎的研究与实现[D]. 西南石油大学,2016.
24. 吴秀君. 面向电子政务的MongoDB与MySQL混合存储策略[J]. 计算机与现代化,2014(08):62-66.
25. 祁兰. 基于MongoDB的数据存储与查询优化技术研究[D]. 南京邮电大学,2016.
26. Feilong Wang. Research and design of industrial information monitoring system based on B/S structure[P]. 2017 4th International Conference on Machinery, Materials and Computer (MACMC 2017),2018.
27. Adam Selipsky, Tableau Software[OL]. [2019.6.1]. https://en.wikipedia.org/wiki/Tableau\_Software
28. W3cschool. Tableau教程[OL]. (2016.12.26) [2019.6.1]. https://www.w3cschool.cn/tableau/
29. Ammann P , Offutt J . Introduction to Software Testing[M]. Cambridge University Press, 2008.
30. Basili V R , Selby R W . Comparing the Effectiveness of Software Testing Strategies[J]. IEEE Transactions on Software Engineering, 1988,13(12):1278-1296.

**致 谢**

光阴似箭，如月如梭。转眼间学生时代即将结束，到了参加工作，步入社会的时候，却又格外怀念大学时光。经过几个月的努力，毕业论文即将完成，回想论文全程，琐碎而平淡的工作，每一个细节，每一处字斟句酌，原本以为会轻松的我，却没想到过程是如此的艰辛，借此机会，我要对四年来帮助过我的人表示深深的感谢。

首先，衷心感谢导师韩希先教授对我的精心指导。从论文选题开始，他就给予我很大的支持和理解。在论文完成过程中，遇到了很多困难与疑惑，韩老师都耐心解答，为我指明前进的方向，他的言传身教将使我终生受益。

其次，感谢殷同学在论文完成过程中给予我的支持，每当我遇到技术难题的时候，他总是不厌其烦的为我讲解，攻克难题。

同时我还要感谢在我学习期间给我极大关心和支持的各位老师以及关心我的同学和朋友。正是由于他们，我才能在各方面取得显著的进步，在此向他们表示我由衷的谢意，并祝所有的老师培养出越来越多的优秀人才，桃李满天下。也祝愿我的同学的在以后的道路上越走越宽，前程似锦。

最后，诚挚的感谢论文评阅老师和各位专家教授对本文提出宝贵意见。