# 摘 要

在当今中国社会经济发展中，农业、农村和农民问题一直备受瞩目。我国农村仍然面临信息不对称和商品供应链短缺等问题，制约了农业发展和农民收入的提升。结合当前三农政策背景，为农村居民提供综合服务，促进农村经济发展。通过微信小程序平台和Vue3框架构建，以提升农产品购物、社区交流和信息分享的便捷性。

本系统通过“三农E家”微信小程序，为农村居民提供了三农政策和农业科技的新闻、互助社区和农资商城于一体的服务平台，旨在解决这些问题。同时，系统还通过后台管理系统“三农E家”进行用户、商品和订单的全面管理，以确保系统高效运行。在实现方面，前台采用微信小程序和Vant-Weapp组件库，后台管理系统基于Vue3和TypeScript，采用Element-Plus组件库，均使用Node.js提供后端接口服务。系统强调社区交流模块，为农村居民提供更广泛的社交和信息分享机会，增进彼此了解。

本系统的设计与实现，不仅符合当下社会对农业发展的需求，也为农村居民提供更多社交和信息互动机会，具有显著的社会意义。

**关键词：**三农政策、微信小程序、Vue3、Node.js、农产品购物、社区交流

# ABSTRACT

In the current socio-economic development of China, issues related to agriculture, rural areas, and farmers have always been under close scrutiny. Rural areas in China still face challenges such as information asymmetry and shortages in the supply chain, which hinder the development of agriculture and the improvement of farmers' income. Aligning with the current background of the "Three Rural" policy, this system provides comprehensive services for rural residents to promote rural economic development. Built on the WeChat Mini Program platform and Vue3 framework, the system aims to enhance the convenience of agricultural product shopping, community interaction, and information sharing.

The "Three Rural E-house" WeChat Mini Program, part of this system, offers a platform that combines news, mutual assistance communities, and an agricultural supplies mall to address these issues and deliver the benefits of the "Three Rural" policy and agricultural technology to rural residents. Simultaneously, the system ensures comprehensive management of users, products, and orders through the "Three Rural E-house" backend management system, ensuring the efficient operation of the system. In terms of implementation, the front end adopts the WeChat Mini Program and the Vant-Weapp component library, while the backend management system is based on Vue3 and TypeScript, utilizing the Element-Plus component library and Node.js to provide backend interface services. The system emphasizes the community interaction module, providing broader opportunities for socializing and information sharing for rural residents, fostering mutual understanding.

The design and implementation of this system not only meet the current societal demands for agricultural development but also provide more social and interactive opportunities for rural residents, carrying significant social significance.

**KEY WORDS:** Three Rural Policy, WeChat Mini Program, Vue3, Node.js, agricultural product shopping, community interaction.

目 录

[摘 要 I](#_Toc5626)

[ABSTRACT II](#_Toc16840)

[第1章 前言 1](#_Toc19772)

[第2章 系统需求分析 2](#_Toc30779)

[2.1. 角色分析 2](#_Toc32757)

[2.2. 功能分析 2](#_Toc21267)

[2.3. 可行性分析 4](#_Toc10950)

[2.3.1. 应用可行性 4](#_Toc30066)

[2.3.2. 技术可行性 4](#_Toc21893)

[2.3.3. 经济可行性 4](#_Toc32440)

[第3章 开发工具及关键技术 6](#_Toc18892)

[3.1. 开发工具 6](#_Toc27580)

[3.1.1. H Builder X 6](#_Toc14039)

[3.1.2. Visual Studio Code 6](#_Toc21362)

[3.1.3. 微信小程序 6](#_Toc4561)

[3.2. 关键技术 6](#_Toc12372)

[3.2.1. Uni-app 6](#_Toc17754)

[3.2.2. Django 7](#_Toc31820)

[3.2.3. ORM框架 7](#_Toc24149)

[第4章 系统设计 9](#_Toc10444)

[4.1. 系统架构及功能设计 9](#_Toc10275)

[4.2. 系统流程图 10](#_Toc15513)

[4.2.1. 系统登录流程图 10](#_Toc31668)

[4.2.2. 系统注册流程图 11](#_Toc27937)

[4.2.3. 来访登记流程图 12](#_Toc24591)

[4.2.4. 处理预约记录流程图 13](#_Toc30547)

[4.3. 系统数据库设计 14](#_Toc20555)

[4.3.1. E-R图设计 14](#_Toc23627)

[4.3.2. 数据库逻辑结构设计 15](#_Toc23045)

[第5章 系统实现 21](#_Toc8412)

[5.1. 用户端功能设计 21](#_Toc24339)

[5.1.1. 注册功能实现 21](#_Toc30331)

[5.1.2. 登录功能实现 23](#_Toc15818)

[5.1.3. 修改密码功能实现 25](#_Toc29747)

[5.1.4. 首页功能实现 27](#_Toc10812)

[5.1.5. 导航栏功能实现 28](#_Toc1510)

[5.1.6. 数据功能实现 33](#_Toc23050)

[5.1.7. 图表功能实现 34](#_Toc10342)

[5.1.8. 个人中心功能实现 35](#_Toc7272)

[5.1.9. 个人资料功能实现 36](#_Toc9428)

[5.1.10. 访客记录功能实现 37](#_Toc6443)

[5.1.11. 绑定老人功能实现 37](#_Toc8839)

[5.2. 后台管理系统功能设计 38](#_Toc26044)

[5.2.1. 管理系统页面 38](#_Toc1775)

[5.2.2. 账号管理模块 39](#_Toc2260)

[5.2.3. 员工管理模块 39](#_Toc8411)

[5.2.4. 老人管理模块 40](#_Toc21779)

[5.2.5. 房间管理模块 41](#_Toc820)

[5.2.6. 监护人管理模块 41](#_Toc6495)

[5.2.7. 每日数据管理模块 42](#_Toc12901)

[5.2.8. 程序管理模块 42](#_Toc23853)

[第6章 系统测试 45](#_Toc8695)

[6.1. 系统配置 45](#_Toc20514)

[6.2. 功能测试 45](#_Toc11318)

[6.2.1. 小程序端注册功能测试 45](#_Toc4016)

[6.2.2. 小程序端登录功能测试 46](#_Toc15318)

[6.2.3. 小程序端数据展示功能测试 47](#_Toc14997)

[6.2.4. 小程序端提交预约来访功能测试 47](#_Toc24566)

[6.2.5. 后台系统老人模块搜索功能测试 48](#_Toc18796)

[6.2.6. 后台系统来访管理是否预约功能测试 48](#_Toc1260)

[6.3. 测试总结 49](#_Toc7346)

[第7章 总结与展望 50](#_Toc8384)

[7.1. 总结 50](#_Toc29667)

[7.2. 展望 50](#_Toc22599)

[参考文献 51](#_Toc24226)

# 前言

在当今中国社会，农业、农村和农民问题一直备受瞩目。随着国家三农政策的深入实施和农业科技的不断发展，我们深刻认识到农村信息不对称和商品供应链短缺等问题制约了农业的发展和农民收入的提升。为了促进农业经济的繁荣，提高农村居民的生活水平，我们设计并实现了三农E家系统。本系统以中国农村的实际情况为背景，以微信小程序和Vue3为技术支撑，致力于为用户提供便捷的购物体验、实时的社区交流和全面的信息分享。通过系统的运作，我们期望为农村居民搭建一个全方位服务平台，实现农业与信息技术的深度融合，助力农村的经济社会发展。

该系统以微信小程序作为前端应用，通过vant-weapp框架和node.js后端技术，实现了新闻浏览、互助社区交流、农资商城购物、购物车管理、用户中心等一系列核心功能。同时，后台管理系统采用vue3、ts、element-plus和node.js技术栈，为管理员提供了用户管理、菜单管理、权限管理、角色管理、商品属性管理、订单管理等全面的后台支持。本文主要介绍了系统需求分析，系统开发工具及关键技术，系统的设计、实现过程及测试结果。

# 系统需求分析

## 角色分析

农民（用户）：登录注册小程序，利用微信小程序进行农产品购物、社区交流，获取三农政策新闻和农业科技信息等操作。

管理员：系统的管理者，拥有三农E家后台系统的所有权限，进行用户管理、商品管理、订单管理，维护系统的正常运行，也负责上架商品和订单发货。

商品管理员：系统商品的管理者，拥有三农E家后台系统的商品管理和订单管理，进行商品管理、订单管理，可以进行平台属性管理，spu管理，sku管理，负责上架商品和订单发货。

## 功能分析

用户的小程序端主要功能是农民进行对信息的了解和交流以及对农资产品的购买。农民第一次进入小程序要进行注册，注册后自动登录，退出登录后要重新登录。登录小程序后，可以查看首页的三农政策和农业科技的相关新闻，可以在互助社区发表问题寻求解决，也可以点赞评论帮助别人。可以在农资商城便捷的购买所需的农资产品，如假如购物车的商品，可在购物车里查看。可以在我的，对个人资料进行填写，例如地址管理，手机号，头像，昵称等等。而没有注册登录的人只能在首页和互助社区且不能使用点赞评论功能。用户功能用例如图 2-1所示。

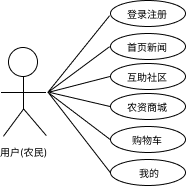


图 2-1 用户功能用例图

管理员端运行于后台管理系统，包含用户管理，角色管理，权限管理，菜单管理，平台属性管理，销售属性管理，spu管理，sku管理，订单管理，管理员功能用例图如图 2-2所示。

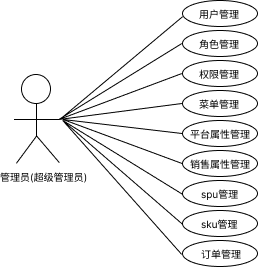


图 2-2 管理员功能用例图

商品管理员权限低于管理员，能访问除账号管理外的其他模块，商品管理员可以进行台属性管理，销售属性管理，spu管理，sku管理，订单管理，对应的信息将展示于用户所看到的小程序端，便于用户对商品的购买与发货更新。商品管理员功能用例图如图 2-3所示。

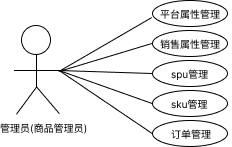


图 2-3 商品管理员功能用例图

## 

## 可行性分析

为确保三农E家系统的设计和实施能够成功，我们进行了全面深入的可行性分析，涵盖了应用可行性、技术可行性和经济可行性三个关键方面。这一综合性的分析不仅有助于发现系统可能面临的挑战，还能够明确其优势和潜在问题，为系统的进一步发展提供科学依据。

### 应用可行性

在应用可行性方面，三农E家系统旨在满足农村居民的购物需求、社区交流和信息分享。考虑到当前社会对农业发展和农民问题的高度关注，系统在应用方面呈现出了极大的市场需求。采用微信小程序平台，系统不仅提供了丰富多彩的功能，还通过互联网的便捷性为农村居民提供了全新的服务体验。在中国农村庞大人口基数的支撑下，系统在应用层面展现出了极大的潜力和可行性，为促进农业经济的发展创造了有利条件

### 技术可行性

技术可行性主要关注系统的架构和开发工具的选择。三农E家系统选用微信小程序+Vant-Weapp前端，Vue3+Element-Plus后台，Node.js提供后端接口服务。这一技术组合保证了系统的灵活性和高效性。微信小程序和Vue3等技术具有广泛的开发者基础和强大的社区支持，Vant-Weapp和Element-Plus等组件库提供了丰富的开发资源。采用Node.js作为后端服务，保障系统的稳定性。综合而言，系统在技术可行性方面拥有现代、可靠的技术基础，为顺利实施提供了坚实支持。

### 经济可行性

经济可行性主要考察项目的成本与效益关系。系统的制作过程分为前期调研、中期开发和后期测试。前期调研成本较低，中期开发和后期测试使用的工具，如Vant-Weapp、Element-Plus等，均为免费开源，降低了制作过程中的经济负担。系统的经济成本相对较低，与当前市场的投资回报比相匹配，为系统的可持续发展提供了经济基础。

# 开发工具及关键技术

## 开发工具

### 微信开发者工具

微信开发者工具是小程序开发的官方集成开发环境，具有强大的调试和实时预览功能。该工具提供了便捷的开发和调试环境，加速了小程序前端的开发过程。通过微信开发者工具，我们能够高效地构建、调试和发布微信小程序，确保用户能够顺畅地使用系统。

### Visual Studio Code

Visual Studio Code是一款轻量级、强大且免费的集成开发环境。作为一种通用编辑器，Visual Studio Code支持多种编程语言，拥有丰富的插件生态系统。在系统的开发中，Visual Studio Code为我们提供了高效的代码编辑、调试和版本控制的功能，提升了前后端开发的整体效率。

## 关键技术

### 微信小程序

微信小程序是一种轻量级的应用程序，无需下载安装即可使用。通过微信小程序，我们实现了系统的前端部分，提供了用户友好的界面和良好的响应速度。微信小程序的广泛普及也确保了系统的覆盖面和可用性。

### Vue3

Vue3是一款流行的JavaScript框架，具有响应性、灵活性和高效性。在系统的后台管理部分，我们使用Vue3框架，通过组件化的方式构建了清晰结构的用户界面，提高了系统的可维护性和开发效率。。

### Node.js + SQL

Node.js作为后端技术，结合SQL数据库，为系统提供了高性能、非阻塞I/O和强大的数据存储与管理能力。Node.js处理后端逻辑，而SQL数据库用于结构化的数据存储和高效的查询操作，共同构建了系统稳定、高效的后端基础。这种组合适用于对性能和数据操作有较高要求的应用场景，为用户提供了可靠的服务。

# 系统设计

## 系统架构及功能设计

本三农E家系统充分运用前后端分离的理念，借助微信小程序实现前端，旨在提供出色的用户体验和高效的开发效率。后端则构建于Node.js之上，采用Express框架，为管理人员提供了强大而灵活的后台管理系统。

系统核心体现为两大端面：管理员端和用户端。管理员端专为系统管理者量身定制，主要涵盖用户、商品和订单管理等关键业务。用户端则以微信小程序为媒介，提供直观而简便的操作体验，使用户能轻松浏览、查询以及使用系统中的各项功能和服务。每个模块的功能各不相同，系统架构图和功能图如图 4-1和图 4-2（下页）所示。

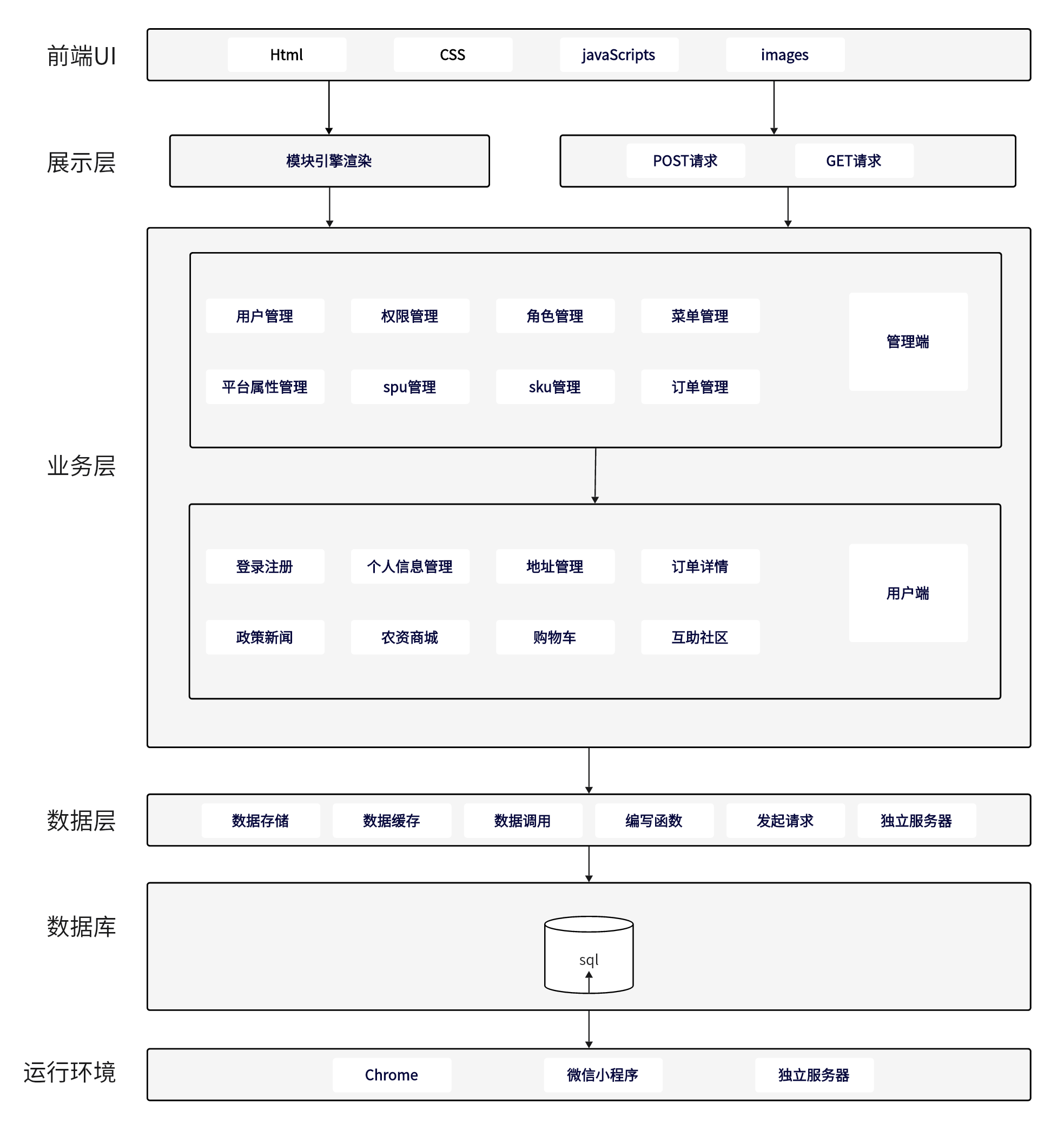


图 4-1 系统结构图

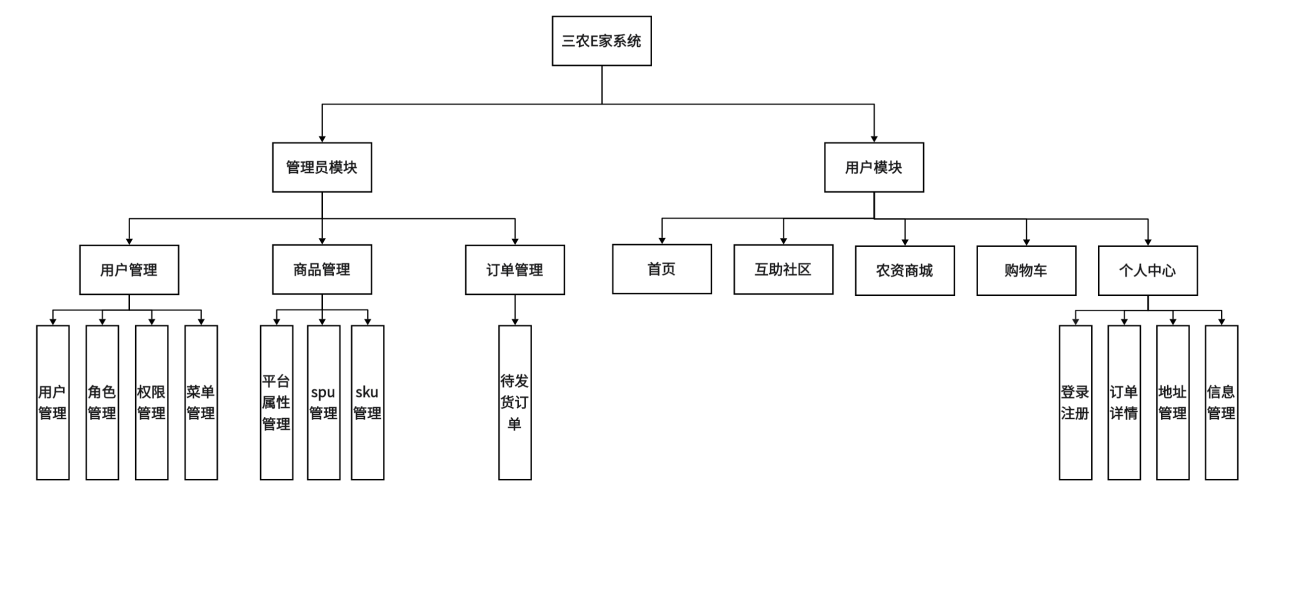


图 4-2 系统功能图

## 系统流程图

### 系统登录流程图

进入小程序后，游客可访问页面有首页以及首页中的各个跳转页面，互助社区的展示页，农资商城，购物车，订单详情，信息更改和互助社区的详情页，只有用户登录后才能访问，用户输入账号密码后会先判断是否为空，再判断用户名密码是否正确，具体登录流程图如图 4-3所示（下页）。

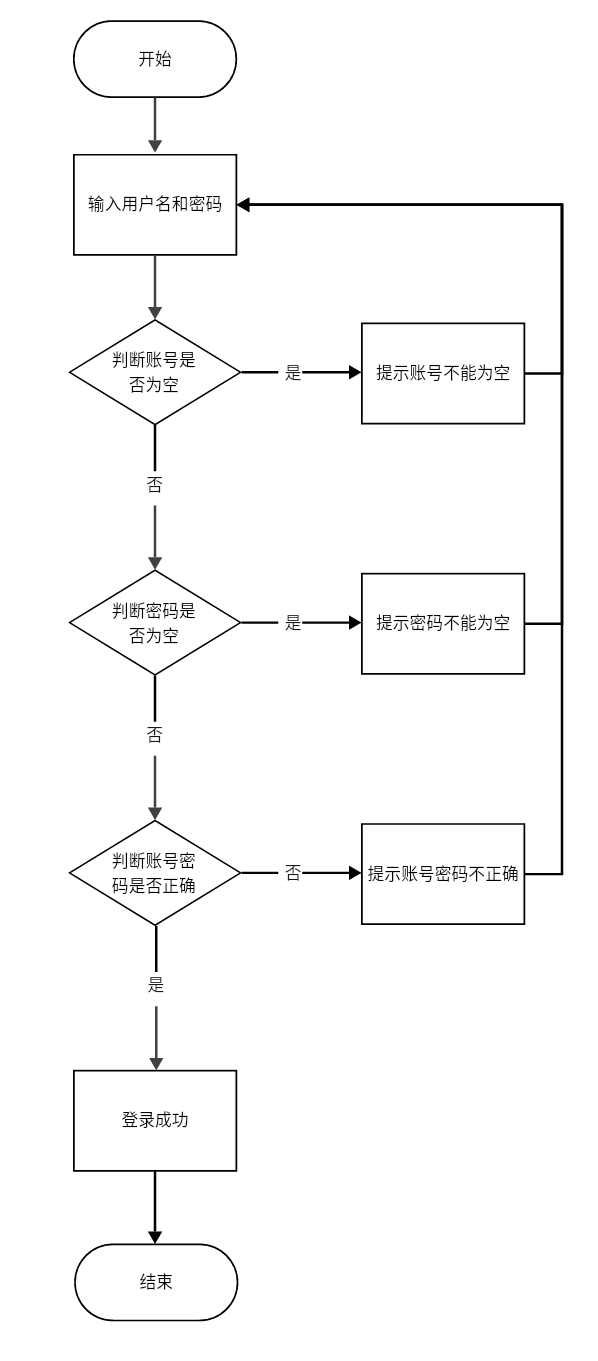


图 4-3登录流程图

### 系统注册流程图

用户要登录的话要先注册账号，通注册的账号密码才能进行登录，详细的注册过程如图 4-4所示（下页）。

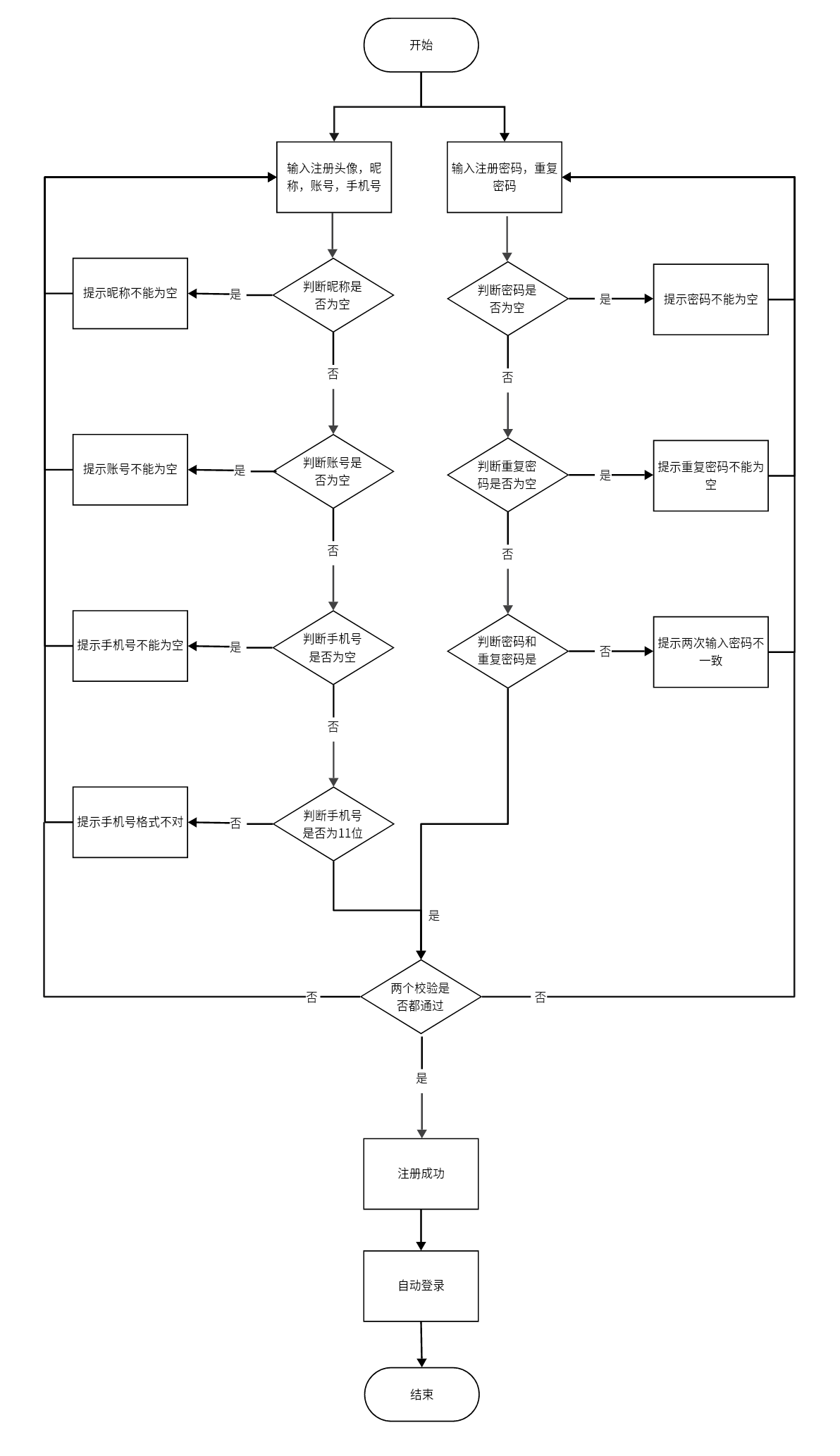


图 4-4 注册流程图

### 用户购买流程图

三农E家系统购买有两种方式：一种是直接购买，一种是购物车购买，购买后判断是否付款，付款后再到商家发货，用户收货后可以进行对商品的评论。具体流程图如图 4-5所示。

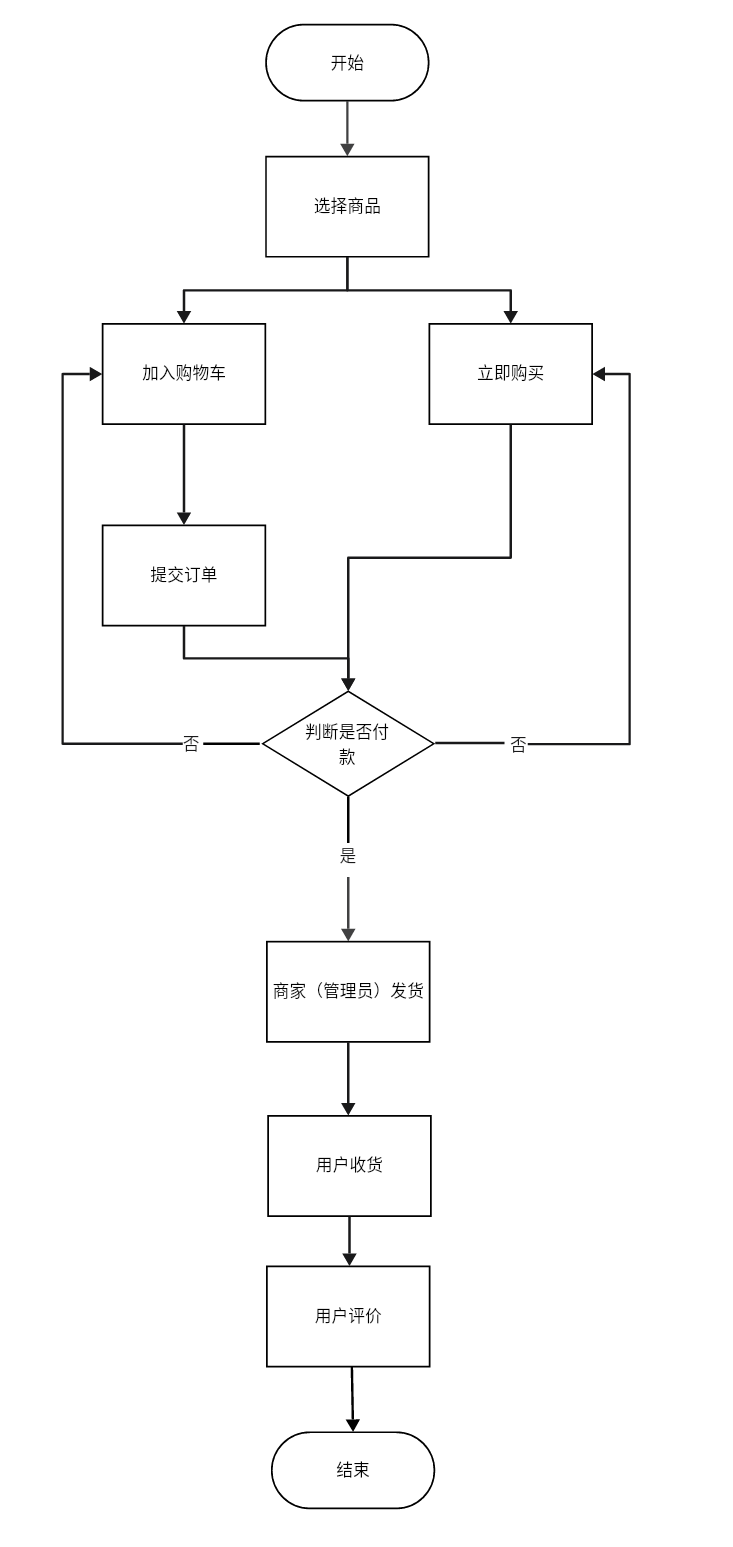


图 4-5 用户购买流程图

## 系统数据库设计

### E-R图设计

结合上述系统功能结构和功能流程图，对三农E家各角色功能及关系进行剖析，最终主要业务关系E-R图设计如图 4-6所示。

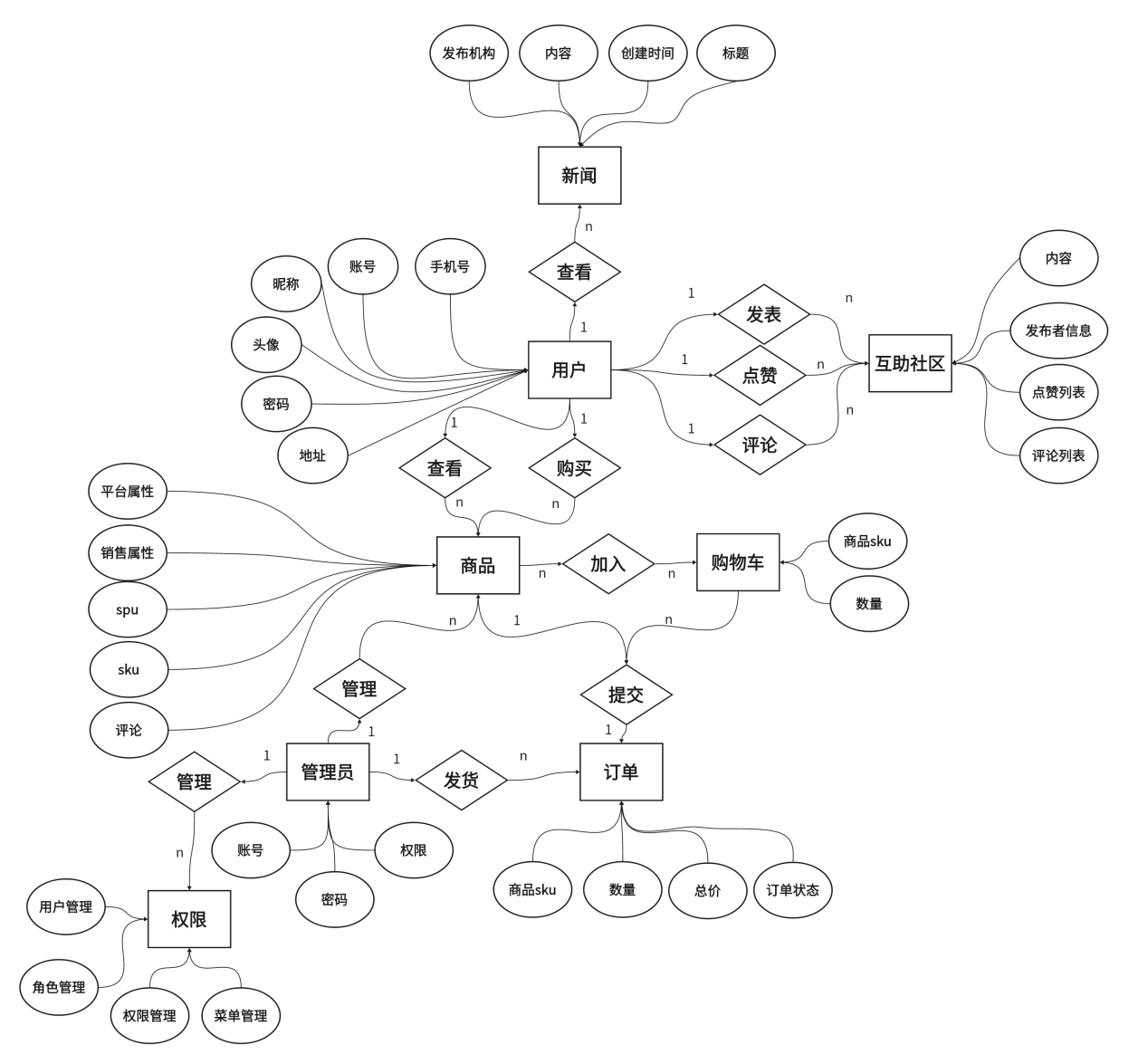


图 4-6 业务关系E-R图

### 数据库逻辑结构设计

根据上述所设计的E-R图以及系统的功能分析，设计出数据表如下所示，系统一共涉及到了17个数据表，具体情况如表 4-1所示（下页）。

表 4-1 数据表

|  |  |  |
| --- | --- | --- |
| 序号 | 集合名称 | 说明 |
| 1 | adminusers | 管理员表 |
| 2 | userinfo | 用户信息表 |
| 3 | adress | 用户地址表 |
| 4 | comment | 用户评论表 |
| 5 | dingdan | 订单表 |
| 6 | existingattribute | 销售属性表 |
| 7 | helpcommunity | 互助社区表 |
| 8 | menu | 菜单表 |
| 9 | news | 新闻表 |
| 10 | platformattribute | 产品属性表 |
| 11 | property | 产品分类表 |
| 12 | role | 角色表 |
| 13 | salesattribute | 销售属性分类表 |
| 14 | shopcart | 购物车表 |
| 15 | skudetail | sku表 |
| 16 | spudetail | Spu表 |
| 17 | spuimage | Spu图片表 |

以下为各表的具体情况：

管理员表（adminusers），如下方表 4-2所示。

表 4-2 adminusers管理员表

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| userid | int | 11 | NOT NULL | 是 | UNIQUE | 管理员id |
| username | varchar | 255 | NOT NULL | 是 | UNIQUE | 账号 |
| password | varchar | 255 | NOT NULL | 否 |  | 密码 |
| avatar | varchar | 255 | NOT NULL | 否 |  | 头像 |
| roles | json |  | NOT NULL | 否 |  | 角色数组 |
| token | varchar | 255 | NOT NULL | 否 |  | 管理员标识 |
| name | varchar | 255 | NOT NULL | 否 |  | 管理员昵称 |

用户信息表（userinfo），如下方表 4-3所示。

表 4-3 userinfo表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| \_id | varchar | 255 | NOT NULL | 是 | UNIQUE | 用户id |
| username | varchar | 255 | NOT NULL | 否 | UNIQUE | 账号 |
| nickname | varchar | 255 | NULL | 否 |  | 昵称 |
| headSculpture | varchar | 255 | NULL | 否 |  | 头像 |
| phone | varchar | 255 | NULL | 否 |  | 手机号 |
| sex | varchar | 255 | NULL | 否 |  | 性别 |
| birthday | date | 0 | NULL | 否 |  | 生日 |
| password | varchar | 255 | NOT NULL | 否 |  | 密码 |

用户地址表（adress），如下方表 4-4所示。

表 4-4 adress表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 255 | NOT NULL | 是 | UNIQUE | 地址id |
| \_id | varchar | 255 | NOT NULL | 否 | UNIQUE | 用户id |
| useraddress | varchar | 255 |  | 否 |  | 详细地址 |
| phone | varchar | 255 |  | 否 |  | 手机号 |
| sjr | varchar | 255 |  | 否 |  | 收件人 |
| isdefault | varchar | 255 |  | 否 | UNIQUE | 是否为默认地址 |
| addressspace | varchar | 255 |  | 否 |  | 地区 |

用户评论表（comment），如下方表 4-5所示。

表 4-5 comment表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 评论id |
| comment | varchar | 255 | NOT NULL | 否 |  | 评论 |
| userinfo | json | 0 | NOT NULL | 否 |  | 用户信息 |
| pid | int | 11 | NULL | 否 |  | 父亲评论 |
| createtime | datetime | 0 | NOT NULL | 否 |  | 评论时间 |
| likelist | json | 0 | NULL | 否 |  | 点赞列表 |

订单表（dingdan），如下方表 4-6所示。

表 4-6 dingdan表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| dingdanid | int | 11 | NOT NULL | 是 | UNIQUE | 订单id |
| \_id | varchar | 255 | NOT NULL | 否 |  | 用户id |
| status | int | 255 |  | 否 |  | 订单状态 |
| shopcart | json | 0 |  | 否 |  | 订单数据 |
| adress | Json | 0 |  | 否 |  | 地址信息 |
| cjtime | datetime | 0 |  | 否 |  | 创建订单时间 |
| fukuantime | datetime | 0 |  | 否 |  | 付款时间 |
| fahuotime | datetime | 0 |  | 否 |  | 发货时间 |

销售属性表（existingattribute），如下方表 4-7所示。

表 4-7 existingattribute表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 销售属性id |
| spuid | int | 11 |  | 否 |  | Spuid |
| basesaleattrid | int | 11 |  | 否 |  | 销售属性分类id |
| saleattrname | varchar | 255 |  | 否 |  | 销售属性名称 |
| saleattrvaluelist | json | 0 |  | 否 |  | 销售属性数组 |

互助社区表（helpcommunity），如下方表 4-8所示。

表 4-8 helpcommunity表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 互助社区id |
| \_id | varchar | 255 | NOT NULL | 否 |  | 用户id |
| nickname | varchar | 255 |  | 否 |  | 用户昵称 |
| headSculpture | varchar | 255 |  | 否 |  | 用户头像 |
| creattime | datetime | 0 |  | 否 |  | 创建时间 |
| likelist | json | 0 |  | 否 |  | 点赞列表 |
| fileurls | json | 0 |  | 否 |  | 发表图片列表 |
| adress | varchar | 255 |  | 否 |  | 位置信息 |
| title | varchar | 255 |  | 否 |  | 发表内容 |
| comdata | json | 0 |  | 否 |  | 评论id列表 |

菜单表（menu），如下方表 4-9。

表 4-9 menu表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 菜单id |
| pid | int | 11 |  | 否 |  | 父级id |
| name | varchar | 255 |  | 否 |  | 菜单名称 |
| code | varchar | 255 |  | 否 |  | 路由名称 |
| level | int | 255 |  | 否 |  | 菜单等级 |

新闻表（news），如下方表 4-10所示。

表 4-10 news表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 新闻id |
| soure | varchar | 255 | NOT NULL | 否 |  | 发表单位 |
| article\_type | int | 11 | NOT NULL | 否 |  | 新闻类型 |
| title | varchar | 255 | NOT NULL | 否 |  | 新闻标题 |
| img\_url | varchar | 255 | NOT NULL | 否 |  | 新闻图片 |
| content | json | 0 | NOT NULL | 否 |  | 新闻内容 |
| datetime | date | 0 | NOT NULL | 否 |  | 发表时间 |
| comment\_count | int | 11 | NOT NULL | 否 |  | 阅读量 |

产品属性表（platformattribute），如下方表 4-11所示。

表 4-11 platformattribute表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 平台属性id |
| propertyid | int | 11 | NOT NULL | 否 |  | 分类属性id |
| name | varchar | 255 |  | 否 |  | 平台属性名称 |
| value | json | 0 |  | 否 |  | 平台属性参数 |

商品分类表（property），如下方表表 4-12所示

表 4-12 property表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 分类属性id |
| name | varchar | 255 | NOT NULL | 否 |  | 分类属性名称 |

角色表（role），如下方表 4-13所示

表 4-13 role表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | id | 是 | UNIQUE | 角色id |
| rolename | varchar | 255 | rolename | 否 |  | 角色名称 |
| permission | json | 0 | permission | 否 |  | 菜单权限列表 |

销售属性分类表（salesattribute），如下方表 4-14所示

表 4-14 salesattribute表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| basesaleattrid | int | 11 | NOT NULL | 是 | UNIQUE | 销售属性分类id |
| saleattrname | varchar | 255 | NOT NULL | 否 |  | 销售属性名称 |

购物车表（shopcart），如下方表 4-15所示

表 4-15 shopcart表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | 购物车id |
| \_id | varchar | 255 | NOT NULL | 否 |  | 用户id |
| skuid | int | 11 |  | 否 |  | Skuid |
| count | int | 11 |  | 否 |  | 数量 |

sku表（skudetail），如下方表 4-16所示

表 4-16 skudetail表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | Skuid |
| spuid | int | 11 | NOT NULL | 否 |  | Spuid |
| price | float | 10 |  | 否 |  | 价格 |
| weight | varchar | 10 |  | 否 |  | 重量 |
| skucount | int | 11 |  | 否 |  | 库存 |
| skusaleattrvalueList | json | 0 |  | 否 |  | Sku销售属性 |
| skuplatformattributeList | json | 0 |  | 否 |  | Sku平台属性 |
| skuimage | varchar | 255 |  | 否 |  | Sku展示图 |
| skudescription | varchar | 255 |  | 否 |  | Sku详细介绍 |
| skuname | varchar | 255 |  | 否 |  | Sku名称 |
| putaway | int | 255 |  | 否 |  | 是否上架 |
| selledcount | int | 255 |  | 否 |  | 销售量 |

Spu表（spudetail），如下方表 4-17所示

表 4-17 spudetail表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | Spuid |
| spuname | varchar | 255 | NOT NULL | 否 |  | Spu名称 |
| description | varchar | 255 |  | 否 |  | Spu介绍 |
| propertyid | int | 11 |  | 否 |  | 商品分类id |
| comdata | json | 0 |  | 否 |  | Spu评价 |

Spu图片表（spuimage），如下方表 4-18所示

表 4-18 spuimage表结构

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 字段名称 | 类型 | 长度 | 允许空值 | 主键 | 唯一 | 字段说明 |
| id | int | 11 | NOT NULL | 是 | UNIQUE | Spu图片id |
| spuid | int | 11 | NOT NULL | 否 |  | Spuid |
| imagename | varchar | 255 |  | 否 |  | Spuimage名字 |
| imgurl | varchar | 255 |  | 否 |  | Spuimage地址 |

# 系统实现

## 用户端功能设计

### 注册功能实现

用户进入小程序后，点击我的，选择登录注册，注册页面如图 5-1所示，用户进行头像，昵称，账号，密码，重复密码，手机号的填写，如果填写有不符合规则的会在点击注册后弹出提示如图 5-2所示，如果账号已存在弹窗账号已存在，请重试。如果注册成功，自动登录。返回上一级。



图 5-1 注册页面



图 5-2 注册信息校验页面

注册账号时前端校验通过后把用户注册信息提交到后端。后端先用uuid生成特有的\_id,再查询改账号是否注册，如果已经被注册，弹出提示：账号已存在，如果用户名不存在，插入新用户信息，弹出提示：用户注册成功，并返回\_id,用于自动登录，注册验证后端逻辑代码如图 5-3所示。



图 5-3 注册验证身份代码

### 登录功能实现

用户进入小程序后，点击我的，选择登录注册，登录页面如图 5-4所示，用户输入账号，密码，账号密码在前端进行校验，账号不能为空，且账号长度必须在6到20之间，密码不能为空校验通过在向后端发送请求前端校验逻辑代码如图 5-5所示，后端接到账号密码后，进行账号密码的校验，校验不通过的返回账号或密码错误，如果正确则返回用户标识\_id，后端的账号密码校验逻辑代码如图 5-6所示。

图 5-4 用户登录图

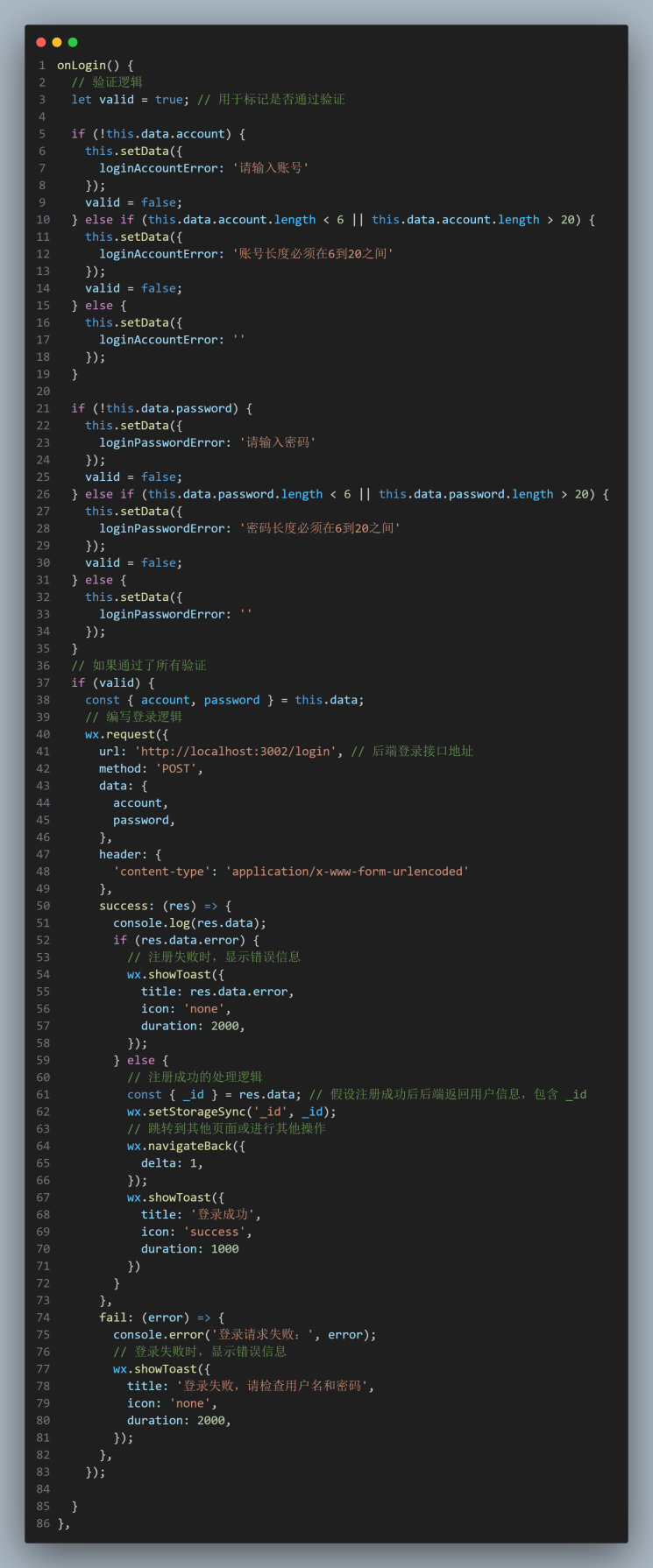


图 5-5前端校验逻辑代码

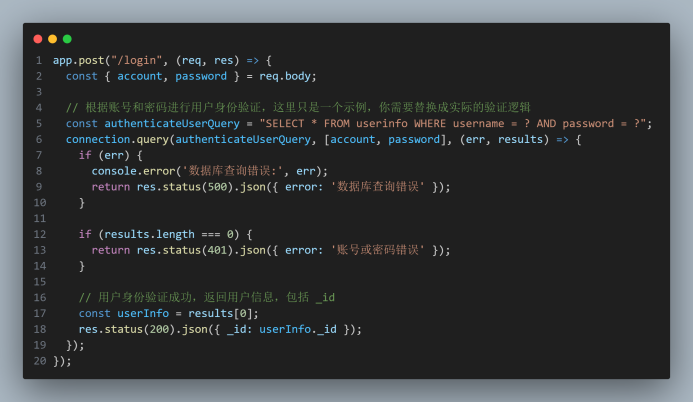


图 5-6 后端校验逻辑代码

### 首页功能实现

首页展示了轮播图，和新闻资讯，如图 5-7所示，点击相关资讯的栏目可以切换新闻类型如图 5-8所示

图 5-7首页功能图  图 5-8切换新闻类型图

想查看哪个新闻就点击新闻进入详情页，如图 5-9所示。进入新闻详情页首先通过新闻id获取新闻详情，获取新闻详情代码如图 5-10所示，通过获取的新闻详情进行页面渲染，进行页面渲染的代码如图 5-11所示。页面用了van-nav-bar组件是一个自定义的导航栏组件，主体内容通过scroll-view来渲染让主体内容可以滚动展示。Content的数据是根据段落进行分解成数组存进数据库里的，所以用wx：for渲染出来每一段。



图 5-9新闻详情页

图 5-10 新闻详情代码图  图 5-11渲染新闻代码图

### 互助社区功能实现

互助社区是一个用于农民交流问题的一个平台，用户可以发布问题，查看问题，评论问题。进入互助社区模块首先看到的是互助社区的展示如图 5-12所示，点击右下角的编辑按钮可以进入到发布问题页，如图 5-13所示。点击文章跳转到详情页如图 5-14 所示。

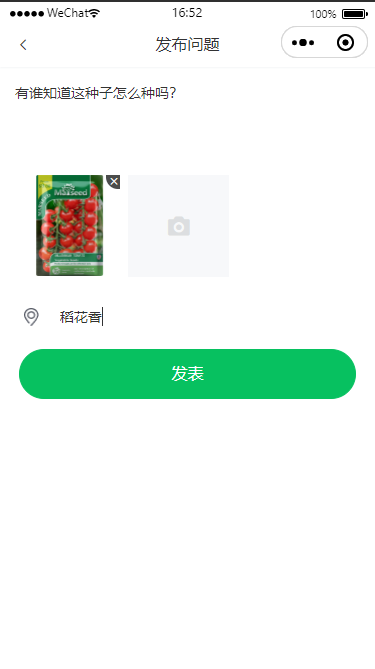


图 5-12互助社区图  图 5-13发表问题图  图 5-14社区详情图

在互动社区页我封装了一个自定义组件community-card，用来多次复用，在详情页里也使用了该组件。通过<community-card / >标签调用组件传出参数communityData：互助社区内容（用户信息，发表内容，点赞列表，评论），isdetail：是否为详情页，bind:comment方法：点击评论按钮触发，bind:upvote方法点赞该社区。组件代码如图 5-15所示，其中还有一个自定义组件grid-view，这个组件是展示图片的，一张图片为大图，四张图片为四宫格，其他的按九宫格排布，grid-view代码如图 5-16所示。

图 5-15 community-card组件



图 5-16 grid-view组件

### 农资商城功能实现

农资生成模块主要分为农资商城页和商品详情页。农资商城页如图 5-17所示，点击选择规格弹出购物弹窗，购物弹窗如图 5-18所示，点鸡商品进入详情页，详情页如图 5-19所示，详情页里包括商品轮播图，购物弹窗，商品信息，商品评论。

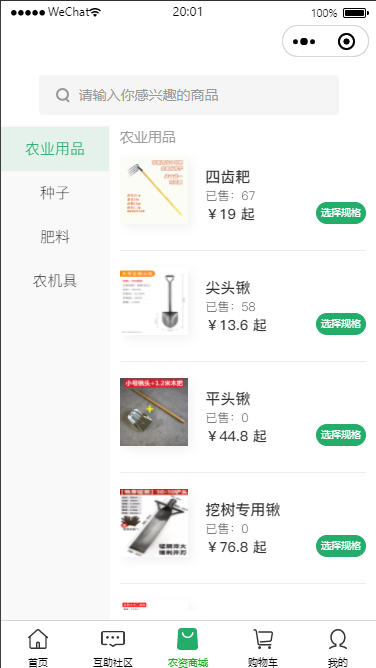


图 5-17 农资商城图 图 5-18购物弹窗图 图 5-19商品详情图

其中农资商城页点击左边的商品品类可以使右边的商品展示模块滚动到指定商品类型区域，反之滚动商品展示模块可以使左边商品品类选择指定品类，这部分的实现是通过scroll-view里的scroll-into-view属性，该属性可以该属性值变化时使scroll-view滚动到与该属性值一样id的锚点出，具体代码如图 5-20所示。反之右边scroll-view滚动是监听scroll事件，根据滚动的距离是否超过上个标签来确定是否到达某区域。代码如图 5-21所示

图 5-20 scroll-view代码图

图 5-21 scroll滚动监视代码图

### 购物车功能实现

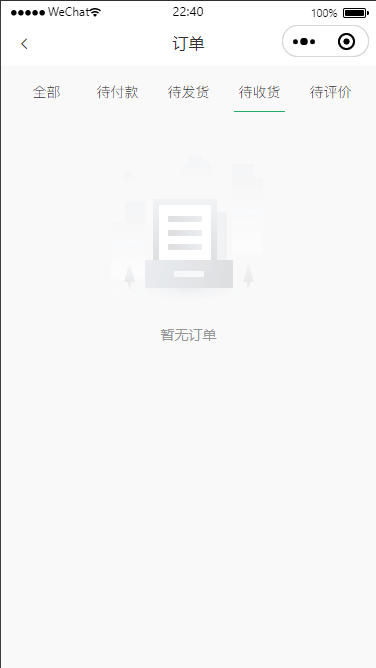
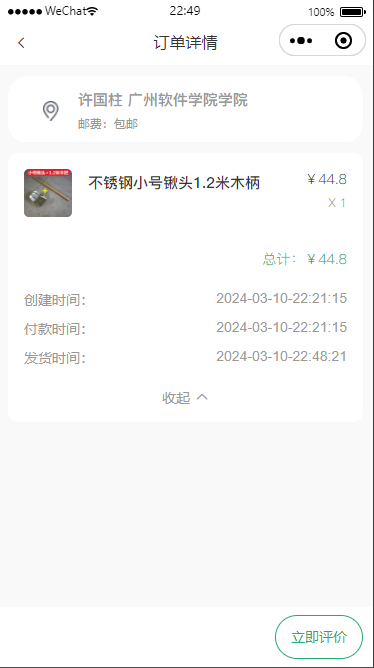
在农资商城模块点击了加入购物车，把具体的商品和数量加入了购物车。在购物车里可以查看到你所加入的商品，如果购物车里没有商品会显示空状态购物车空空如也，如图 5-22所示，如果购物车有商品，则会获取商品数量并且显示在购物车右上角，选择购物车里的商品会自动计算总价。如图 5-23所示

 图 5-22 空购物车图  图 5-23 购物车全选图

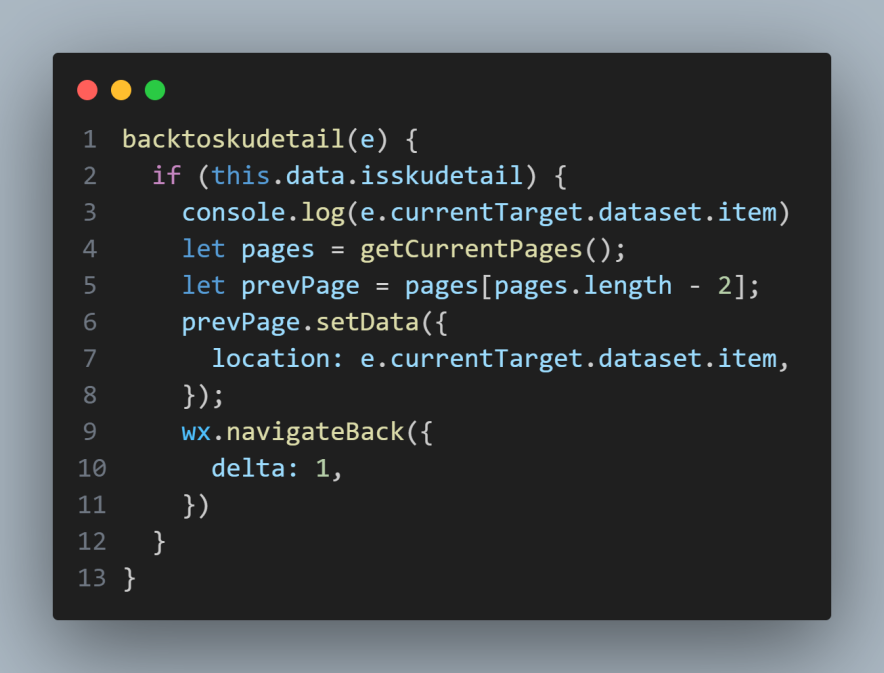
其中购物车右上角的小红点是调用了微信小程序的API，showTabBarRedDot：显示红点，setTabBarBadge：显示红点里的数字，hideTabBarRedDot：隐藏红点，removeTabBarBadge：移除红点数字，在有缓存里有用户唯一标识\_id的时候调用接口/getshopcartcount，获取购物车商品的数量，如果有商品数量不为0，则展示红点和数量，反之则隐藏红点和数量。此外为了在各个模块对购物车有修改能够检测到，我在农资商城和我的和购物车都在可能修改购物车操作的地方都调用了这个方法代码如图 5-24所示

图 5-24 红点显示代码

### 个人中心功能实现

我的页面有两种状态，一种是未登录状态，如图 5-25所示。一种是已登录状态，如图5-26所示。未登录状态只能点击登录/注册，点击其他功能会强制跳转到登录页。已登录的用户点击设置可以对个人信息进行设置，账号的退出登录，注销账号和进行地址管理。订单模块可以查看订单情况，如果没有所查询的订单状态的订单，会显示暂无订单的空状态，如图 5-27所示。如果有则展示订单，点击订单会跳转到订单详情页，可以对订单进行操作，如图 5-28所示。 图 5-25未登录图  图5-26已登录图  图 5-27空订单图  图 5-28订单详情图

其中用户设置的地址管理模块中有两种方式可以到达这个页面，一种是从设置模块进来对地址进行管理。可以对自己的地址进行管理，还有一种方式是在购物流程中，无论是购物车购买还是立即购买时会选择默认地址，如果没有默认地址就选择第一个地址。要想更换地址就点击地址信息便会跳转到地址编辑页如图所示。如果是从购物流程跳转过来的点击地址信息，则会跳转回去并填写说选地址信息。这里所用的是微信小程序提供的方法getCurrentPages（）来获取当前页面信息，同样可以获取路由历史的信息，获取到后让他的页面长度-2就是上个页面的index值在用这个对象使用setdata方法使购物流程的页面中的location的值为当前选中的值。然后再返回上一页。使用方法如图所示。

 图 5-29管理地址图 图 5-30选择地址返回实现代码

## 管理员端功能设计

### 登录功能的实现

管理员端登录，输入账号密码，进行账号密码的规则校验再向服务器发出请求校验账号密码的正确与否。此账号密码由管理员统一分配和管理。登录页面如图 5-31所示。

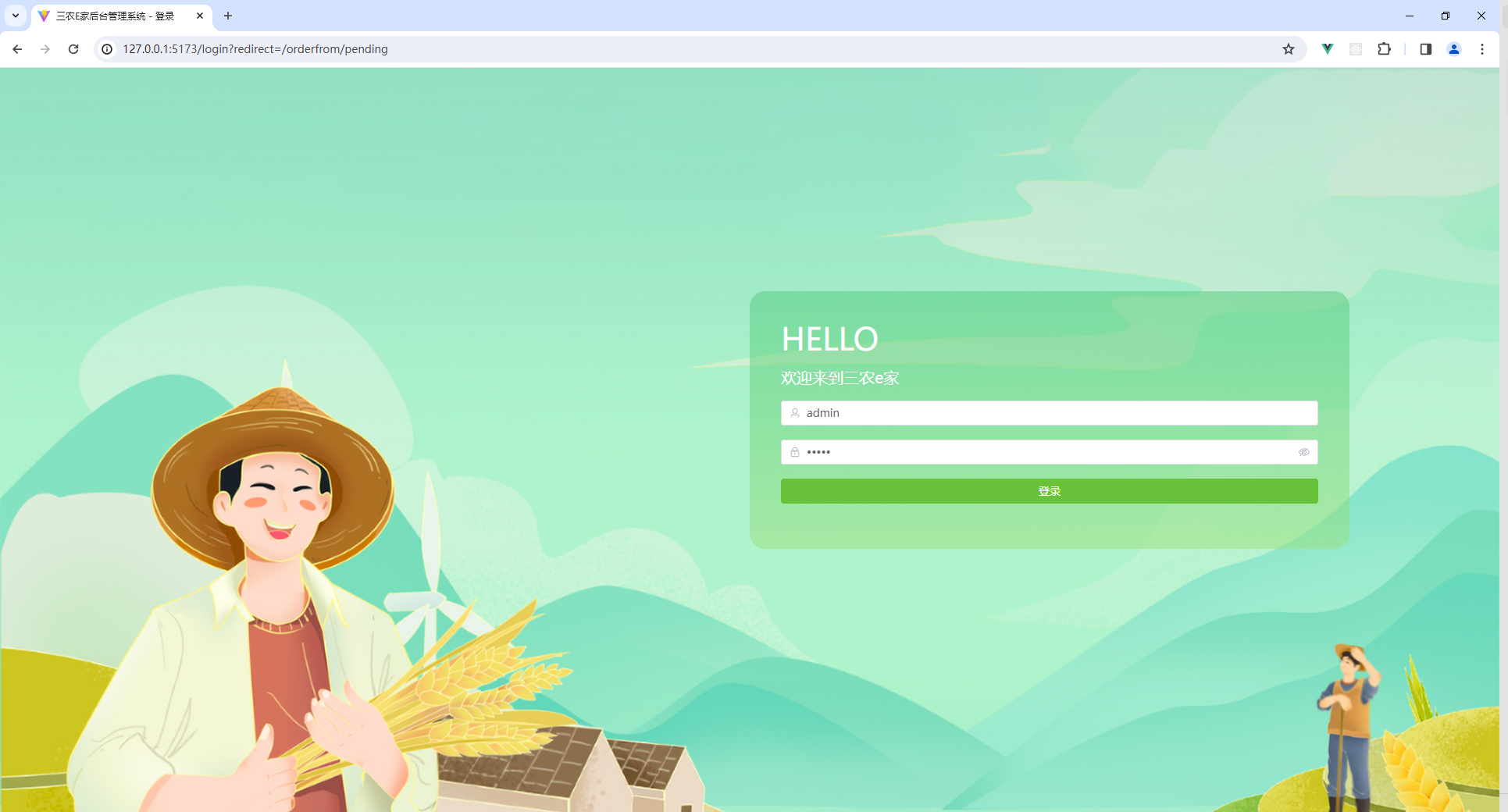


图 5-31 登录页图

在上图中可以看到url处除了路由信息还有redirect的信息，此处信息为重定向信息，用于登录成功后跳转到指定页面。如果没有权限的话则跳转404页面。此功能实现使用到的技术是vue中的路由守卫。路由守卫会在路由跳转时被访问，如果路由login页面query中有redirect的参数则重定向到指定页面。代码实现如图 5-32所示。

图 5-32 登录重定向代码实现图

### 首页功能实现

首页展示了管理员信息，商品的信息，订单的信息。其中右上角有刷新页面和切换暗黑模式的功能。首页的展示情况如图 5-33所示。

图 5-33 首页图

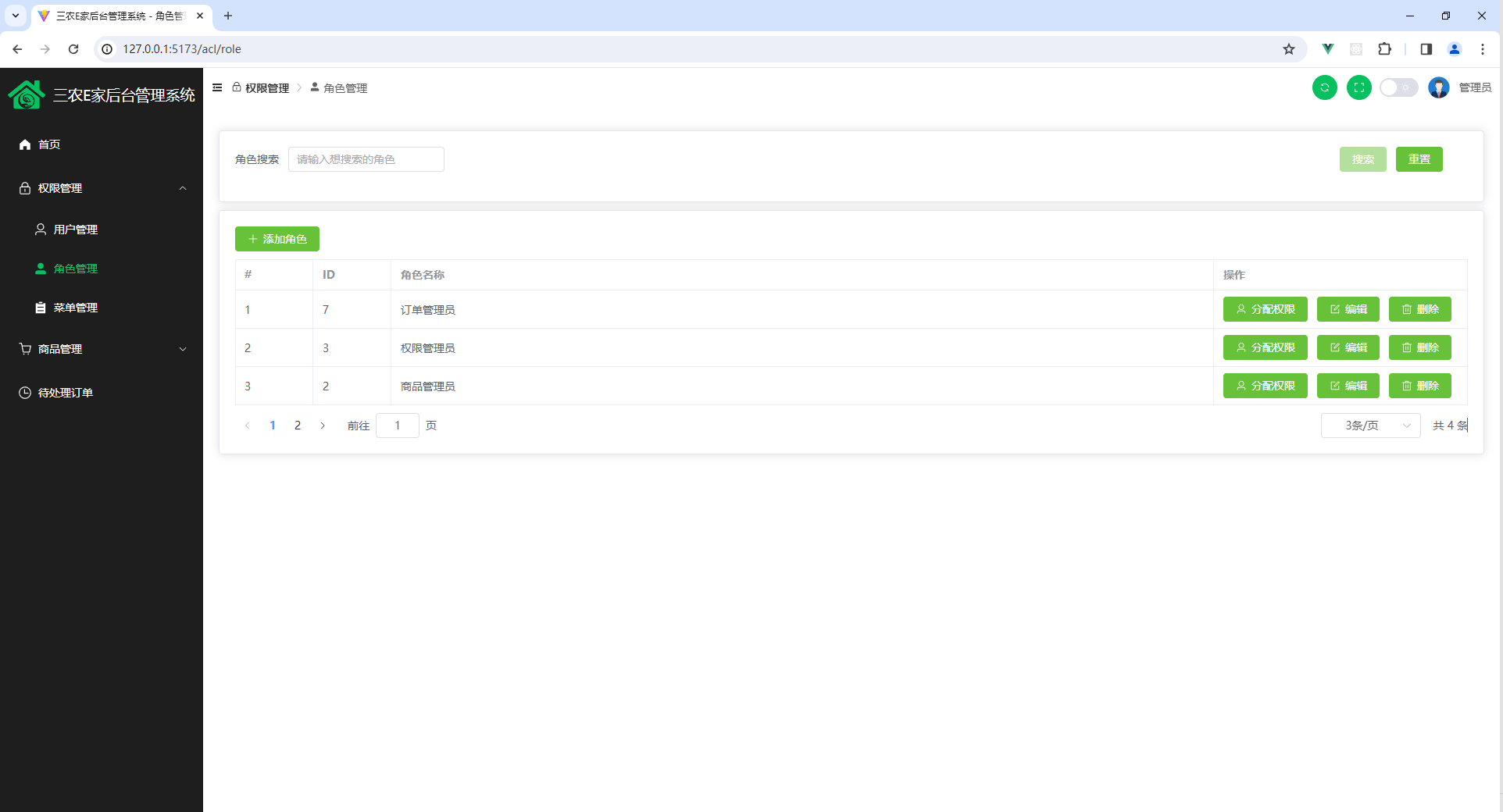
其中所展示的图表都是用e-chart框架所绘制的，如库存数量通过接口请求到库存中sku的名称和库存数。通过echart画出来。代码实现如图所示。其中xAxis为X轴的显示内容，yAxis为y轴显示内容。Series为填充数据的信息。

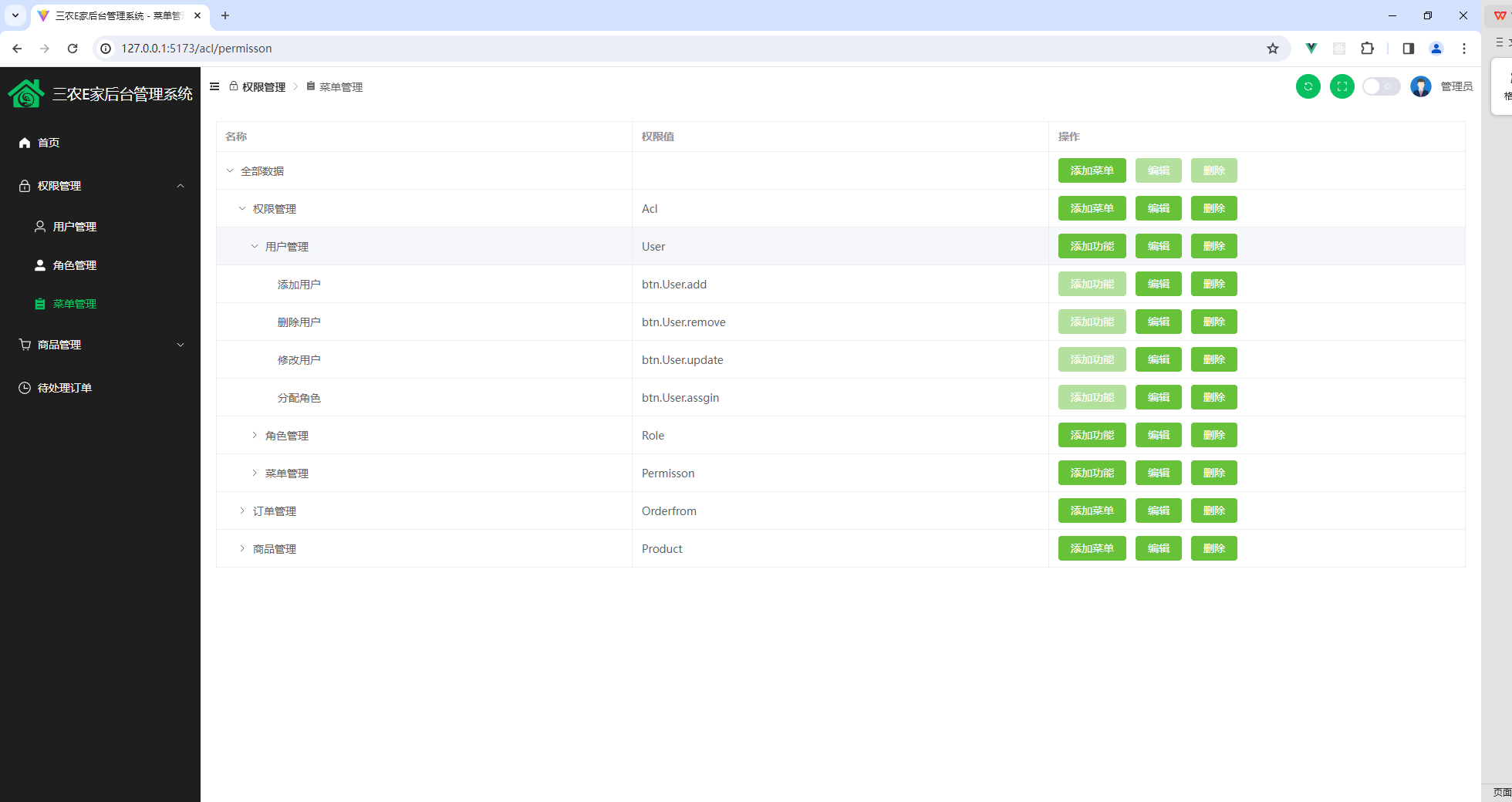
图 5-34 echart绘制图表代码

### 权限管理功能实现

权限管理分为三部分：用户管理，角色管理，菜单管理。用户管理可以进行新建用户，编辑用户，分配角色，删除用户，搜索用户。实现效果如图所示。角色管理可以进行新建角色，编辑角色，分配菜单权限，删除角色，搜索角色。实现效果如图所示。菜单管理可以进行添加菜单，编辑菜单，删除菜单的功能。菜单权限最小单位是按钮的权限。实现效果如图所示。

图 5-35 角色管理图

图 5-36角色管理图

图 5-37菜单管理图

其中菜单管理列表在后端获取数据是把有pid的数据放到了对应的id数据的children元素里。此处处理用了递归的方法。在pid等于父id时把改id的children属性进行递归。代码实现图 5-38所示。

图 5-38 递归方法取菜单图

### 商品管理功能实现

商品管理分为属性管理，spu管理，sku管理。属性管理可以某类商品的产品属性进行添加，修改，删除。实现效果如图 5-39所示。Spu管理，可以对某类商品进行添加spu，添加sku，查看全部sku和删除spu。实现效果如图所示。Sku管理，可以对sku进行查看，删除和上下架。实现效果如图所示。

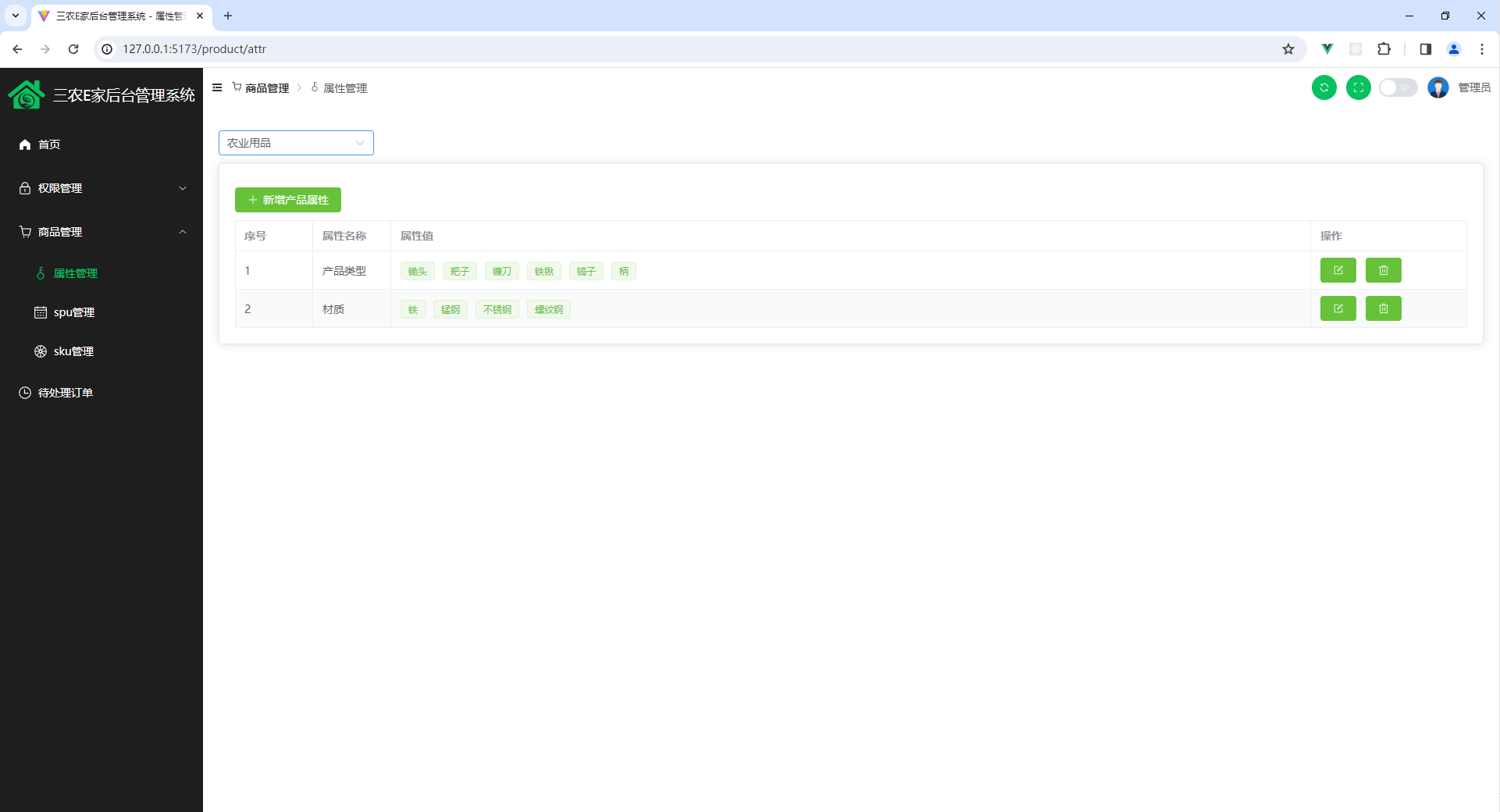


图 5-39 属性管理图

其中在spu管理中修改spu和添加spu功能中有个选择销售属性的列表，它会根据已有的销售属性和所有的销售属性做对比动态的选择可添加的销售属性。如图 5-40所示。此处用了数组的filter和find方法和vue的computed属性，让全部的销售属性过滤（filter）出不和spu详情中已有销售属性一样的id的数组。以此来动态的控制销售属性的列表。代码如图所示

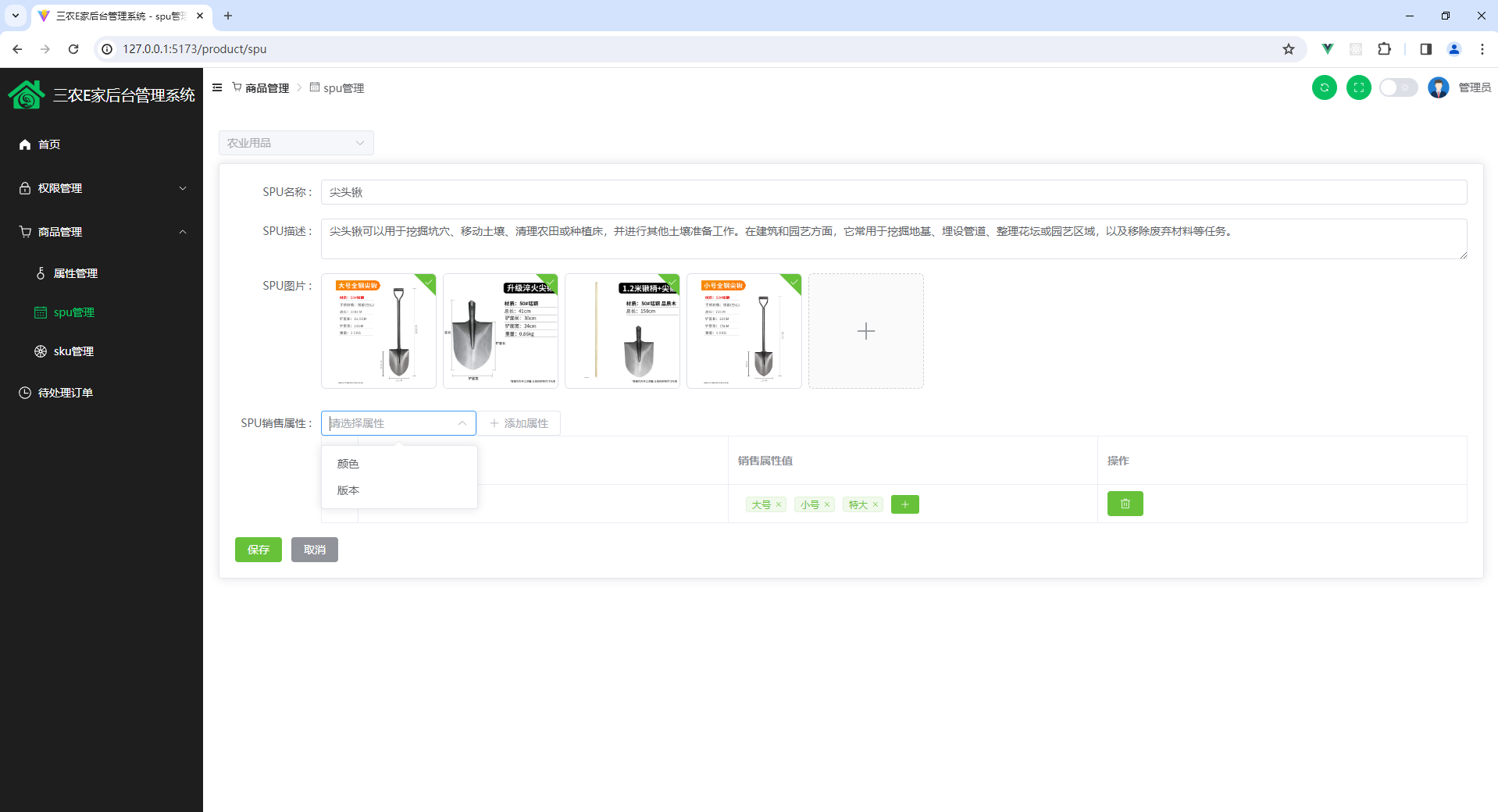
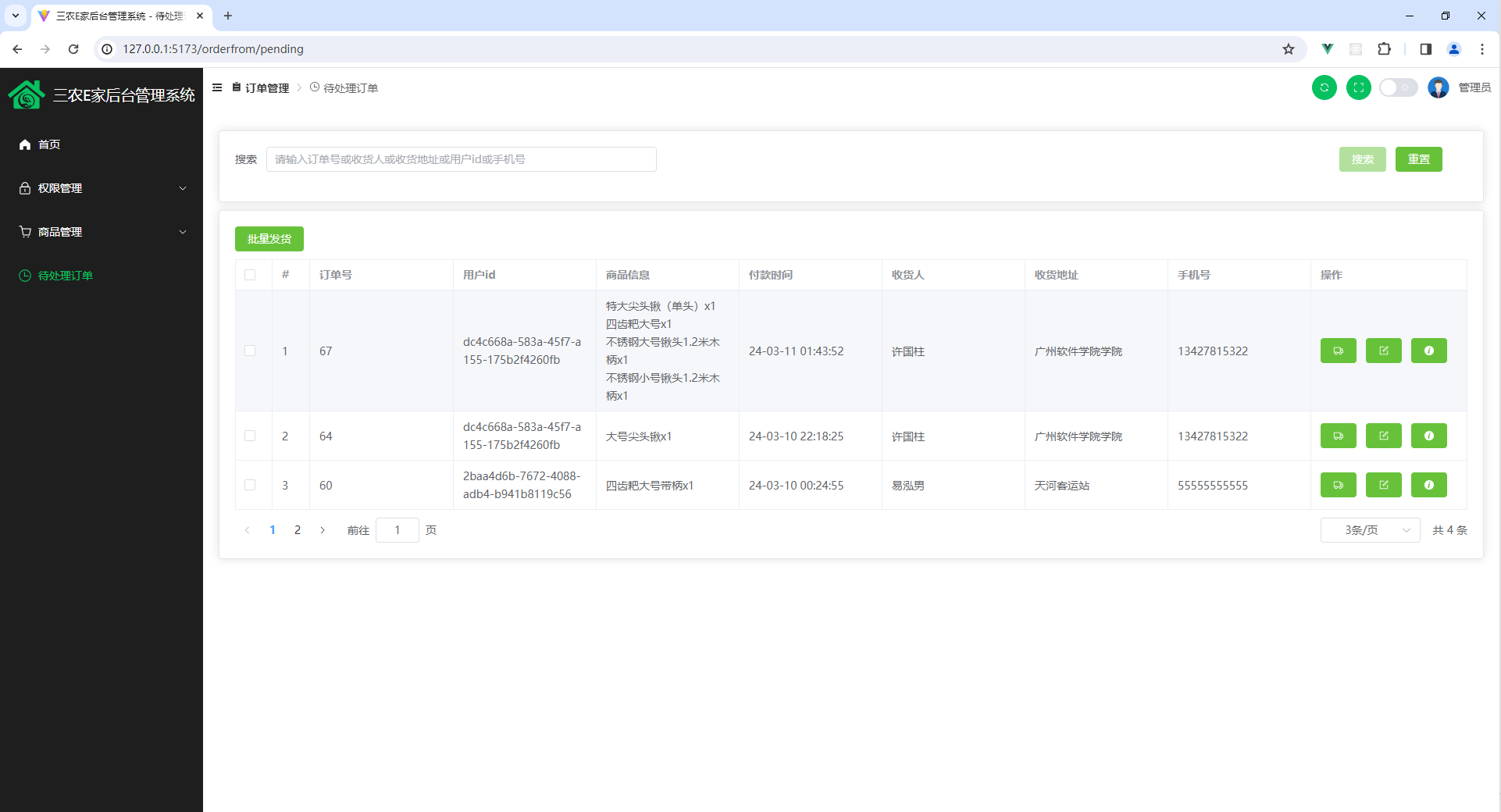
图 5-40 添加spu销售属性图

图 5-41 动态计算销售属性列表图

### 订单管理功能实现

订单管理在管理员端的功能有对已付款的订单进行管理，订单管理有待处理订单可以进行待处理订单的搜索，对订单的发货，修改订单和查看订单。功能实现如图 5-42所示

图 5-42待处理订单

其中对显示的待处理订单可以进行批量发货，点击订单前面的选择框，点击批量发货。弹出提示是否确定批量发货，这里用了element-plus的popconfirm的组件然后调用batchdelivery，通过调用el-table的getSelectionRows()。获取选中的表单。然后用数组map方法把选择的订单id整成数组。然后再进行批量发货。代码如图 5-43所示。

图 5-43批量发货功能实现图

# 系统测试

在本章节中，我们对系统的各个接口以及功能进行了测试，旨在确保系统的稳定性和可靠性，为后续系统的正常运行提供有力保障。

## 系统配置

本系统所进行测试的环境具体配置如表 6-1所示。

表 6-1 测试配置表

|  |  |
| --- | --- |
| 配置名称 | 详细配置 |
| 处理器（CPU） | AMD Ryzen 7 4800H |
| 显卡（GPU） | NVIDIA GeForce RTX 2060 |
| 内存 | 16GB |
| 硬盘空间 | 1TB |
| 操作系统 | win10X64 |
| 测试软件 | 微信开发者工具、Uni-app、Postman |
| 数据库 | DB Browser (SQLite)、Navicat Premium 15 |
| 网络环境 | 有线网络、无线网络 |
| 物理硬件 | PC 机、服务器 |
| 浏览器 | Google Chrome、FireFox、Microsoft Edge |
| 开发工具 | VS Code、Uni-app |

## 功能测试

### 小程序端注册功能测试

用户小程序端注册功能具体测试用例如表 6-2所示。

表 6-2 用户小程序端注册测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例ID | 1 | 用例名称 | 用户小程序注册 | |
| 用例描述 | 用户通过输入身份证号码和手机号码验证身份通过后输入账号和密码进行注册操作 | | | |
| 用例入口 | 小程序端登录页面注册组件点击注册按钮 | | | |
| 测试场景ID | 测试场景 | 测试步骤 | 预期成果 | 实际结果 |
| Test01 | 小程序端注册页面 | 不输入身份证号码或手机号码 | 提示“身份证号码不为空”或“手机号码不为空” | 符合预期 |
| Test02 | 小程序端注册页面 | 输入错误身份证号码或手机号码 | 提示“手机号码或身份证号码验证失败” | 符合预期 |
| Test03 | 小程序端注册页面 | 输入已存在用户的身份证号码和手机号码 | 提示“该用户已注册” | 符合预期 |
| Test04 | 小程序端注册页面 | 输入正确身份证号码和手机号码 | 提示“身份验证成功”，进入下一步设置账号密码 | 符合预期 |
| Test05 | 小程序端注册页面 | 不输入账号或密码 | 提示“账号不为空”或“密码不为空” | 符合预期 |
| Test06 | 小程序端注册页面 | 输入的两次密码不一致 | 提示“两次密码输入不一致” | 符合预期 |
| Test07 | 小程序端注册页面 | 输入已存在的账号 | 提示“账号已存在” | 符合预期 |
| Test08 | 小程序端注册页面 | 输入符合要求的账号密码 | 提示“注册成功，请返回登录” | 符合预期 |

### 小程序端登录功能测试

用户小程序端登录功能具体测试用例如表 6-3所示。

表 6-3 用户小程序端登录测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例ID | 2 | 用例名称 | 用户小程序登录 | |
| 用例描述 | 用户通过手机号或账号进行登录操作 | | | |
| 用例入口 | 小程序端个人中心点击登录按钮 | | | |
| 测试场景ID | 测试场景 | 测试步骤 | 预期成果 | 实际结果 |
| Test01 | 小程序登录页面 | 不输入账号或密码 | 提示“账号不为空”或“密码不为空” | 符合预期 |
| Test02 | 小程序登录页面 | 输入错误账号或密码 | 提示“账号或密码输入错误，请重新输入” | 符合预期 |
| Test03 | 小程序登录页面 | 输入正确账号和密码 | 提示“登录成功”，跳转到首页 | 符合预期 |

### 小程序端数据展示功能测试

用户小程序端数据展示页面只有登录成功后携带token返回值的用户才能查看，具体测试用例如表 6-4所示。

表 6-4 数据展示测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例ID | 3 | 用例名称 | 小程序数据页面展示 | |
| 用例描述 | 用户查看数据页面是否具有权限 | | | |
| 用例入口 | 小程序端点击tabbar跳转数据页面 | | | |
| 测试场景ID | 测试场景 | 测试步骤 | 预期成果 | 实际结果 |
| Test01 | 小程序数据页面 | 不登录点击数据页面 | 提示“尚无权限，请先登录” | 符合预期 |
| Test02 | 小程序数据页面 | 登录后点击数据页面 | 展示监护老人数据 | 符合预期 |

### 小程序端提交预约来访功能测试

用户小程序端具有预约来访功能，游客或已用账号用户都可提交，提交后将上传到后台管理系统的在线预约模块中，具体测试用例如表 6-5所示。

表 6-5 提交预约来访测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例ID | 4 | 用例名称 | 小程序预约来访提交 | |
| 用例描述 | 用户提交预约来访上传至后台 | | | |
| 用例入口 | 小程序端点击首页“预约来访”中的提交按钮 | | | |
| 测试场景ID | 测试场景 | 测试步骤 | 预期成果 | 实际结果 |
| Test01 | 小程序预约来访页面 | 填写信息点击提交 | 提示“提交成功”，后台系统在线预约模块中出现该预约内容 | 符合预期 |

### 后台系统老人模块搜索功能测试

对老人模块的检索功能进行测试，验证该功能的稳定性和准确性，具体测试用例如表 6-6所示。

表 6-6 老人模块搜索测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例ID | 5 | 用例名称 | 后台系统老人模块搜索 | |
| 用例描述 | 进入老人模块顶部搜索框输入关键字搜索 | | | |
| 用例入口 | 老人模块顶部搜索按钮 | | | |
| 测试场景ID | 测试场景 | 测试步骤 | 预期成果 | 实际结果 |
| Test01 | 后台系统老人模块 | 在姓名输入框中输入“刘奶奶”，点击搜索 | 展示1条结果，结果为刘奶奶的个人信息 | 符合预期 |
| Test02 | 后台系统老人模块 | 在姓名输入框中输入“刘爷爷”，点击搜索 | 展示0条结果，当前信息不存在 | 符合预期 |
| Test03 | 后台系统老人模块 | 在手机号输入框输入“19874064526”，点击搜索 | 展示1条结果，结果为手机号码为“19874064526”的老人的个人信息 | 符合预期 |

### 后台系统来访管理是否预约功能测试

在来访管理中如若填写的数据是已预约，则对应在线预约模块中的数据中的完成状态将动态更改，具体测试用例如表 6-7所示。

表 6-7 来访模块预约测试用例

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 用例ID | 6 | 用例名称 | 后台系统来访模块预约 | |
| 用例描述 | 进入来访模块输入已预约的数据 | | | |
| 用例入口 | 点击在线预约对应的数据完成状态 | | | |
| 测试场景ID | 测试场景 | 测试步骤 | 预期成果 | 实际结果 |
| Test01 | 后台系统来访模块、在线预约模块 | 来访模块中新增一条预约状态为“是”的数据 | 在线预约模块中匹配的预约数据的完成状态自动更新为“已完成” | 符合预期 |
| Test02 | 后台系统来访模块、在线预约模块 | 来访模块中新增一条预约状态为“否”的数据 | 在线预约模块无发生变化 | 符合预期 |

## 测试总结

本次测试主要针对小程序端用户权限功能以及后台系统一些相关联数据动态变化的测试，目的在于确保本智慧养老系统能够合理且流畅地运行，能基本达到预期。通过一次次的测试与修改，对功能的各个方面进行完善，努力带给用户本系统最好的体验。

# 总结与展望

## 总结

三农E家系统的实际应用在小型农业企业中将为农民用户带来更大的时间和空间的自由。尽管系统在实际应用中仍需进一步改进，但它已经取得了一些显著的优势：

首先，中成国面临着严重的农村问题，而三农E家系统的出现有望为农村居民信息获取和社交互动的新途径，助力农业发展。

其次，系统成功解决了农村信息不对称和商品供应链短缺等问题，为提高农业效益和农民收入提供了可行的解决方案。

再者，系统的前端采用了微信小程序和Vant-Weapp组件库，后台管理系统使用了Vue3+TypeScript+Element-Plus，利用Node.js提供后端接口服务，体现了先进的技术应用，提升了用户体验。

尽管系统已取得一定的成功，但仍有进一步改进的空间。在未来，可以考虑添加更多功能，如品牌的引入，社区的维护等，以提升系统的全面性。

系统的安全性需要更多关注，未来的发展需要进一步加强系统的安全性，确保用户数据和交易隐私的充分保护。

## 展望

虽然系统已完成初步规划和设计，但在实际应用中，仍需投入更多时间、精力和经济支持。在未来的发展中，将持续深入调查和学习，致力于提升系统的功能性和实用性，推动农村居民生活的便捷性，助力农村经济的可持续发展。

参考文献

**[1]陈维榕,彭志良,李莉婕等.基于微信小程序的贵州12316三农服务系统设计与实现[J].农技服务,2020,37(05):46-48.**

**[2]叶桄希. 三农信息服务系统设计[D].电子科技大学,2012.**

**[3]蓝小龙. 基于PHP+MySQL的三农信息系统的设计与实现[D].吉林大学,2014.**

**[4]刘菁.关于加强“三农”资金管理的思考[J].农村经济,2010(05):56-58.**

**[5]陈巧玲. 农村改革中农民推动作用研究[D].西北农林科技大学,2012.**

**[6]白家铭.微信在农业信息传播中的应用研究[J].山西农经,2022(01):152-154.**

**[7]王璐.“三农”政务微信提升策略[J].新闻前哨,2018(10):64-65.**

**[8]姜全坤.基于Vue的通用后台管理系统设计与实现[J].现代信息科技,2022,6(08):33-35+38.**

致 谢

在完成三农E家系统的设计与实现过程中，我要向许多人表示由衷的感谢。首先，我要感谢我的指导教师--赵乐，对我的毕业设计提供了悉心的指导和宝贵的建议。您的耐心与专业性使得整个项目能够顺利推进，我深感荣幸能在您的指导下完成毕业设计。感谢开源社区提供的丰富资源，为我提供了在技术实现上的支持与启发。在开发过程中，能够借助开源工具和框架，大大提高了系统的开发效率。最后，感谢我的家人，你们的支持和理解使我能够专心致志地完成这个项目。在这个毕业设计中，收获了许多宝贵的经验和知识，为我的未来奠定了坚实的基础。