

蛋糕商城

|  |  |
| --- | --- |
| **课程名称：** | JAVAWEB程序设计 |
| **实验项目：** | 蛋糕商城 |
| **小组成员：** | 李若瑛、冯丽芷 |
| **实验班级：** | 22大数据1班 |
| **总 份 数：** | 1 |
| **指导教师：** | 张锐 |

2025年 1 月 1 日

**目录**

**[一、 引言 1](#_Toc5307)**

[1. 1选题背景 1](#_Toc8880)

[1.2目的和意义 1](#_Toc31192)

[1.3开发工具 1](#_Toc17786)

**[二、 可行性分析 1](#_Toc12981)**

[2.1 技术可行性 1](#_Toc28752)

[2.2 经济可行性 2](#_Toc1677)

[2.3 操作可行性 2](#_Toc31194)

**[三、 需求分析 2](#_Toc21594)**

[3.1任务概述 2](#_Toc25684)

[3.2功能需求 3](#_Toc24551)

[3.3运行需求 5](#_Toc3285)

**[四、系统概要设计 5](#_Toc132)**

[4.1系统功能模块设计 5](#_Toc27068)

[4.2数据库设计 6](#_Toc1592)

[4.3界面设计 9](#_Toc31117)

**[五、系统详细设计 11](#_Toc4515)**

[5.1系统架构 11](#_Toc19918)

[5.2类体系设计 14](#_Toc15354)

**[六、系统实现 21](#_Toc14747)**

[6.1系统登录模块实现 21](#_Toc6074)

[6.2系统注册模块实现 23](#_Toc10763)

[6.3 前台模块实现 23](#_Toc6142)

[6.4 后台模块实现 26](#_Toc7843)

**[七、 系统测试 36](#_Toc7082)**

**[八、结论展望与体会 41](#_Toc10849)**

**[九、小组成员及分工 41](#_Toc10849)**

1. 引言
2. 1选题背景

随着互联网的普及和电子商务的迅猛发展，越来越多的传统行业开始向线上转型，食品行业也不例外。蛋糕作为一种广受欢迎的甜品，其在线销售市场潜力巨大。传统的蛋糕销售方式主要依赖于实体店，存在地域限制、营业时间限制等问题，无法满足消费者随时随地购买的需求。而网上蛋糕商城可以突破这些限制，为消费者提供更加便捷、多样化的购买渠道，同时也为蛋糕商家拓展市场、提高销售额提供了新的机遇。因此，开发一个功能完善、用户体验良好的网上蛋糕商城具有重要的现实意义和广阔的应用前景。

1.2目的和意义

目的：

不仅为消费者提供一个方便快捷的蛋糕购买平台，能够随时随地浏览和购买各种口味、风格的蛋糕，满足不同场景和需求；还为蛋糕商家提供一个展示和销售产品的窗口，通过网上商城可以扩大市场覆盖范围，提高品牌知名度和销售额，同时也可以收集用户反馈和数据，优化产品和服务。

意义：

对于消费者而言，网上蛋糕商城可以提供更加丰富多样的蛋糕选择，满足个性化需求，同时购物过程更加便捷、高效，节省时间和精力。

对于蛋糕商家来说，网上商城可以突破地域限制，拓展销售渠道，提高销售业绩，同时也可以更好地了解市场需求和消费者偏好，促进产品创新和服务升级。

1.3开发工具

编程语言：Java，作为项目的主要开发语言，用于编写后端逻辑和处理业务功能。

前端技术：HTML、CSS、JavaScript，用于实现页面布局和交互效果；Bootstrap框架，提供丰富的UI组件和样式，加快前端开发速度。

数据库：MySQL，用于存储项目中的数据，如用户信息、商品信息、订单信息等。

开发工具：IntelliJ IDEA，作为项目的开发环境，提供代码编写、调试、版本控制等功能。

服务器：Tomcat，作为Web应用服务器，用于部署和运行项目。

辅助工具：Navicat，用于数据库管理和数据操作。

1. 可行性分析

2.1 技术可行性

Java Web技术成熟稳定：Java语言具有跨平台、高性能、安全性强等特点，广泛应用于Web开发领域。Java Web开发技术经过多年的发展，已经形成了完整的生态系统，拥有丰富的开发框架和工具，如Spring、Spring Boot、MyBatis等，这些框架和工具能够帮助开发者快速构建稳定、高效的Web应用。

前端技术不断进步：随着HTML5、CSS3、JavaScript ES6等新技术的推出，前端开发能力得到了极大的提升，可以实现更加丰富、流畅的页面效果和交互体验。同时，Bootstrap等前端框架的出现，使得前端开发更加高效和便捷。

数据库支持强大：MySQL是一种开源的关系型数据库管理系统，具有性能高、可靠性强、易于维护等优点。它能够满足网上蛋糕商城对数据存储和查询的需求，支持高并发访问和大数据量处理。

开发工具完善：IntelliJ IDEA、Git、Tomcat等工具为项目开发提供了强大的支持，从代码编写、版本控制到应用部署，都有相应的工具可以使用，极大地提高了开发效率和项目质量.

2.2 经济可行性

开发成本可控：Java Web开发所需的软件和工具大多为开源免费，如Java开发环境、Spring Boot框架、MySQL数据库等，降低了开发成本。同时，网上蛋糕商城的开发周期相对较短，人员投入相对较少，总体开发成本在可接受范围内。

运营成本较低：与传统的实体店相比，网上蛋糕商城不需要租赁实体店面，节省了租金、装修、水电等费用。同时，通过线上销售，可以实现订单的自动化处理和物流配送的优化，降低了运营成本。

盈利潜力大：网上蛋糕商城可以面向全国乃至全球的消费者，市场容量巨大。通过合理的定价策略和营销推广，可以吸引大量用户，实现较高的销售额和利润。

2.3 操作可行性

对于用户而言，网上蛋糕商城的操作流程遵循常见电子商务平台的设计规范，用户注册、登录、浏览商品、添加购物车、下单支付等流程简洁明了，贴合大众日常网络购物习惯，无需复杂的学习过程即可上手使用。商家端的后台管理系统界面设计简洁直观，功能布局合理，管理员只需经过简单培训，即可熟练掌握商品管理、订单处理等操作，能够高效地运营商城业务，保障项目在实际应用中的操作可行性。

1. 需求分析

3.1任务概述

网上蛋糕商城项目旨在开发一个集商品展示、在线购物、订单管理、用户管理等功能于一体的Web应用平台。该平台需要满足消费者和商家的多样化需求，为消费者提供便捷的蛋糕购买服务，为商家提供高效的店铺运营工具。项目开发任务包括需求分析、系统设计、编码实现、测试与部署等环节，最终实现一个稳定、易用、安全的网上蛋糕商城系统.

3.2功能需求

此套系统首先分为四大部分，分别为登录、管理员、用户、注册。首页为登录页面；管理员页面下不仅有用户页面的功能块，还设订单管理、商品管理、客户管理等功能模块；用户页面包含商品分类、我的订单、购物车等功能；注册页面为单独功能模块，可以注册新用户。

蛋糕商城的主要功能是实现用户在蛋糕商城进行自主选购商品，管理人员对商品进行更新分类、修改用户信息等。

3.2.1 前台功能需求

前台的功能可以总结为：（1）用户注册，（2）登录，（3）购物车，（4）商品分类查询，（5）商品搜索，（6）节日公告管理。

（1）用户注册

用户需填写基本信息，包括用户名、手机号码（用于接收商品，确保账号真实与后续订单通知）、电子邮箱（方便接收商城促销信息、订单详情等）、设置密码、收货人和收货地址。填写完成后，点击“提交”按钮，用户的信息插入到 MySQL 数据库的用户表中，注册成功后自动跳转至登录页面，并提示用户注册成功。

（2）登录

登录页面提供用户名与密码输入框。用户输入登录凭据后，点击“登录”按钮，若登录成功，创建会话（Session）保存用户登录状态，记录用户 ID 等关键信息，并跳转至商城首页；若登录失败，返回相应错误提示信息，前端展示错误提示，引导用户重新输入或找回密码。

（3）购物车

当用户在商品详情页点击“加入购物车”按钮时，通过 Ajax 技术异步向服务器端发送添加购物车请求。服务器端查询数据库购物车表，判断该商品是否已存在于购物车中，若已存在，则更新商品数量与小计金额；若不存在，则插入一条新的购物车记录。操作完成后，服务器返回更新后的购物车商品总数，前端实时更新购物车图标上显示的数字，提示用户购物车商品数量变化。用户进入购物车页面后，服务器端查询购物车表与商品表关联数据，获取购物车中商品的详细信息，包括图片、名称、单价、数量、订单总金额等，通过 JSP 页面动态展示，用户在此可修改商品数量、删除商品，并实时查看购物车商品总价，方便用户核对购物成本。

（4）商品分类查询

根据特点和制作材料的不同，可以将蛋糕分为不同的类型，蛋糕商城应当提供商品查询功能，以满足用户挑选需求。单击导航栏中的“商品分类”，选择分类，可以展示该分类下的所有商品。

（5）商品搜索

用户在浏览商品时，可以通过导航栏选择查看不同分类的商品，但是由于蛋糕商城的商品数量众多，并不便于用户快速查找和购买意向商品，因此商城还提供商品搜索功能。搜索功能可以让用户根据商品名称模糊查询商品，满足用户快速搜寻意向商品的需要。商城首页顶部显眼位置设置搜索框，用户输入关键词后，点击搜索按钮或按下回车键便可以进行搜索。

（6）节日公告管理

在后台管理系统中，管理员可提前针对不同节日（如情人节、圣诞节、母亲节等）编辑节日专属公告内容，公告内容是节日促销活动介绍（如特定蛋糕折扣力度、满减优惠、赠品活动等）、新品推荐、限时抢购时段等信息。编辑完成后，将公告信息存储至数据库的公告表中，关联对应的节日字段。前台页面展示时，系统根据当前日期或临近节日自动检索并提取相应的节日公告。

3.2.2 后台功能需求

通过超级管理员admin登录网上蛋糕商城，单击前台导航栏中的“后台管理”按钮，进入后台管理系统，它的主要功能模块有：（1）订单管理模块，（2）客户管理模块，（3）商品管理模块，（4）类目管理模块，（5）节日管理。

（1）订单管理模块

订单管理模块可按订单的状态查询订单，订单状态包括未付款、已付款、配送中、已完成。管理员可对订单的状态进行修改。例如，当客户提交订单并付款后，管理员选择已付款订单，单击“发货”按钮，这样订单的状态就会变为配送中。配送中状态的订单，若客户线下签收完成，管理员可单击“完成”按钮，将订单状态改为已完成。管理员还可以单击“删除”按钮，删除订单。

（2）客户管理模块

网上蛋糕商城中的客户管理是指对客户信息的管理，例如收货人、收货电话、收货地址等。客户管理模块的主要功能包括添加客户、修改客户信息、删除客户和重置客户密码。

（3）商品管理模块

网上蛋糕商城中的商品管理指的是对蛋糕信息的管理，如蛋糕的名称、蛋糕的价格、蛋糕的分类等，通过后台系统中的商品管理模块可以实现蛋糕信息在前台网站上的动态展示。网上蛋糕商城后台管理系统中的商品管理模块主要实现的是添加商品信息、编辑商品信息、删除商品信息、加入推荐条幅、加入热销推荐、加入新品推荐这6个功能。

超级管理员进入后台管理系统后，单击导航栏中的“商品管理”，即可进入商品管理模块的列表页面。商品管理可分别对全部商品、条幅推荐商品、热销推荐商品、新品推荐商品进行管理。由于编辑商品、修改商品、加入/移出条幅、加入/移出热销、加入/移出新品是针对单件商品操作的功能，所以在商品列表中的每个商品后面，还带有针对该商品的编辑、修改、加入/移出条幅、加入/移出热销、加入/移出新品等按钮。

（4）类目管理模块

网上蛋糕商城后台管理系统中的商品类目管理模块主要功能包括添加商品类目、修改商品类目和删除商品类目。

（5）节日管理

提前规划各个重要节日（如情人节、圣诞节、母亲节、中秋节等）的促销活动方案。在节日管理页面，管理员可以编辑活动主题、活动时间范围（精确到起止时刻，确保活动按时开启与结束）和折扣。

3.3运行需求

页面设计风格应契合蛋糕商城的甜蜜、温馨主题。整体色调以棕色系为主，营造出诱人食欲、愉悦舒适的购物氛围。同时，搭配简洁流畅的线条与精致的图标，提升页面的美观度与专业性。在字体选择上，字号适中，确保不同屏幕尺寸下文字显示清晰，便于用户浏览信息。

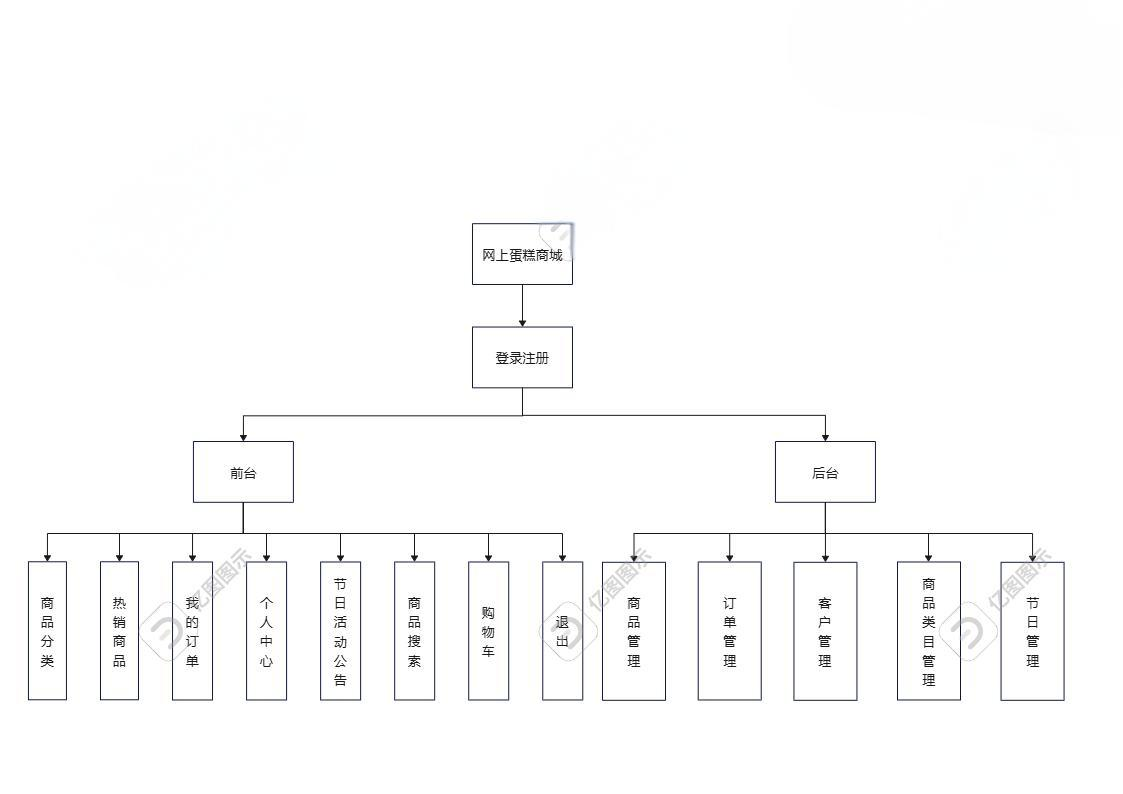
设置清晰明确的导航栏，涵盖首页、商品分类、购物车、个人中心等基本模块。商品分类导航采用下拉菜单形式，细分蛋糕类别，方便用户快速查找所需商品。在页面滚动过程中，导航栏保持固定位置，方便用户随时跳转至其他页面。

四、系统概要设计

4.1系统功能模块设计

此套系统首先分为四大部分，分别为登录、管理员、用户、注册。首页为登录页面；管理员页面下不仅有用户页面的功能块，还设订单管理、商品管理、客户管理等功能模块；用户页面包含商品分类、我的订单、购物车等功能；注册页面为单独功能模块，可以注册新用户。

蛋糕商城的主要功能是实现用户在蛋糕商城进行自主选购商品，管理人员对商品进行更新分类、修改用户信息等。



4.2数据库设计

4.2.1数据库的需求分析

根据功能需求分析，要有一个用户表，用户具有用户编号、姓名、用户名、用户密码、用户邮箱、用户联系方式、地址等属性，身份识别用于区分读者和管理员账户，用户账号、用户密码用于登陆系统；

整个商城最多的是商品，那么就需要商品表，商品表拥有序号、商品名称、商品价格等属性；

除了商品外最多的就是订单，故需要一个订单表，订单表拥有id、用户姓名、支付方式、商品数量等属性；

除此以外还有订单项实体，因此订单项有id、价格、数量等属性；

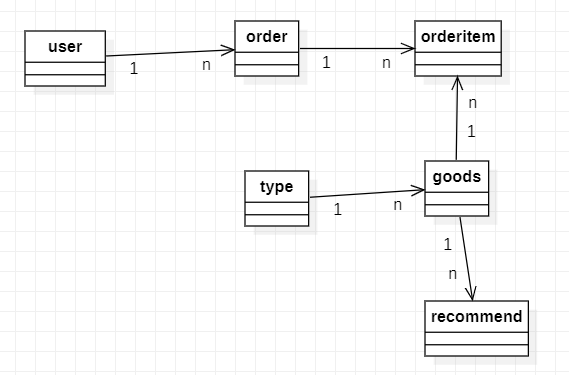
另外还有商品分类和推荐栏，商品分类表有商品id、商品类型这两个属性；推荐栏表有商品id、商品类型、id属性。

而节日管理表有id、节日、折扣等属性。

这便是整套系统的数据库需求。

4.2.2数据库的概念设计

各表的关系都是一对多的关系。一个商品分类对应多个商品、一个商品对应多个推荐栏、一个商品对应多个订单项、一个订单对应多个订单项、一个用户对应多个订单。



4.2.3数据库的逻辑设计

由于数据库概念模型独立于任何特定的数据库管理系统，因此，需要根据具体使用的数据库管理系统的特点进行转换。即转化为按计算机观点处理的逻辑关系模型,E-R模型向关系数据库模型转换应遵循下列原则：

1.每一个实体要转换成一个关系

2.所有的主键必须定义非空（NOT NULL）

3.对于二元联系应按照一对多、弱对实、一对一、多对多等联系来定义外键。

根据E-R模型，图书管理系统建立了以下逻辑数据结构，下面是各数据表的详细说明：

1. user表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 是否为空 | 是否为主键 | 说明 |
| id | int(11) | 否 | 是 | 用户表主键 |
| username | varchar(45) | 否 | 否 | 用户名 |
| password | varchar(45) | 否 | 否 | 用户密码 |
| name | varchar(45) | 是 | 否 | 用户姓名 |
| email | varchar(45) | 否 | 否 | 用户邮箱 |
| phone | varchar(45) | 是 | 否 | 用户电话 |
| address | varchar(45) | 是 | 否 | 用户地址 |
| isadmin | bit(1) | 否 | 否 | 是否为管理员 |
| isvalidate | bit(1) | 否 | 否 | 账户是否有效 |

1. goods表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 是否为空 | 是否为主键 | 说明 |
| id | int(11) | 否 | 是 | 商品表主键 |
| name | varchar(45) | 是 | 否 | 商品名称 |
| cover | varchar(45) | 是 | 否 | 商品封面图 |
| image1 | varchar(45) | 是 | 否 | 商品详情图1 |
| image2 | varchar(45) | 是 | 否 | 商品详情图2 |
| price | float(0) | 是 | 否 | 商品价格 |
| intro | varchar(300) | 是 | 否 | 商品描述 |
| stock | int(1) | 是 | 否 | 商品库存 |
| type\_id | int(11) | 是 | 否 | 商品类型 |

1. orderitem表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 是否为空 | 是否为主键 | 说明 |
| id | int(11) | 否 | 是 | 订单项表主键 |
| price | float(0) | 是 | 否 | 商品价格 |
| amount | int(11) | 是 | 否 | 商品数量 |
| goods\_id | int(11) | 是 | 否 | 商品id |
| order\_id | int(11) | 是 | 否 | 订单id |

1. order表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 是否为空 | 是否为主键 | 说明 |
| id | int(11) | 否 | 是 | 订单表主键 |
| total | float(0) | 是 | 否 | 商品总额 |
| amount | int(6) | 是 | 否 | 商品数量 |
| status | tinyint(1) | 是 | 否 | 支付状态 |
| paytype | tinyint(1) | 是 | 否 | 支付方式 |
| name | varchar(45) | 是 | 否 | 用户名称 |
| phone | varchar(45) | 是 | 否 | 用户电话 |
| address | varchar(45) | 是 | 否 | 用户地址 |
| datetime | datetime | 是 | 否 | 订单日期 |
| user\_id | int(11) | 是 | 否 | 用户id |

1. type表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 是否为空 | 是否为主键 | 说明 |
| id | int(11) | 否 | 是 | 商品类别表主键 |
| name | varchar(45) | 是 | 否 | 商品名称 |

1. recommend表

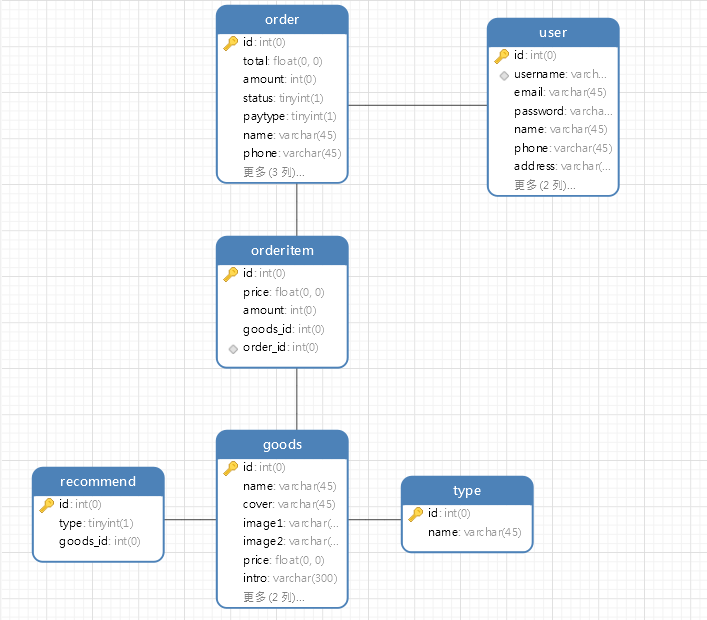
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 是否为空 | 是否为主键 | 说明 |
| id | int(11) | 否 | 是 | 推荐栏表主键 |
| type | tinyint(1) | 是 | 否 | 商品类别 |
| goods\_id | int(11) | 是 | 否 | 商品id |

1. holiday表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 字段名 | 类型 | 是否为空 | 是否为主键 | 说明 |
| id | int(11) | 否 | 是 | 节日管理表主键 |
| holiday\_name | varchar(255) | 否 | 否 | 节日名称 |
| start\_date | date | 否 | 否 | 开始日期 |
| end\_date | date | 否 | 否 | 结束日期 |
| discount | int | 否 | 否 | 折扣 |
| description | text | 是 | 否 | 描述 |

4.2.4数据库的物理设计

设计数据库物理结构，并利用数据库管理系统导出数据库关系图，如下：



4.3界面设计

4.3.1登录/注册界面设计

该注册登录界面是通用的，输入客户或管理员账号便可以进入相关界面。



4.3.2前台界面设计

用户界面：

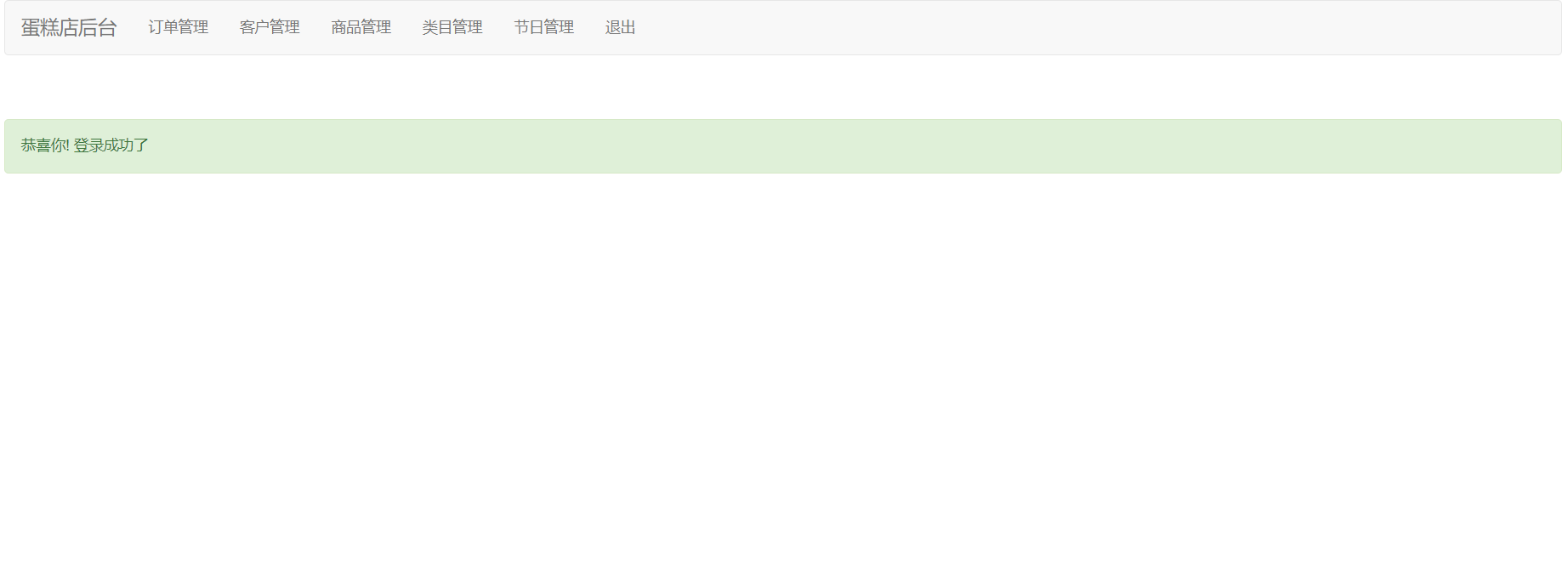


管理员界面：



4.3.3后台界面设计

只有管理员才能进入后台



五、系统详细设计

5.1系统架构

5.1.1逻辑架构

本蛋糕商城系统基于 B/S 架构，并采用 MVC 模式进行设计，各部分相互协作，共同实现系统功能。

（1）**视图层（View）**：

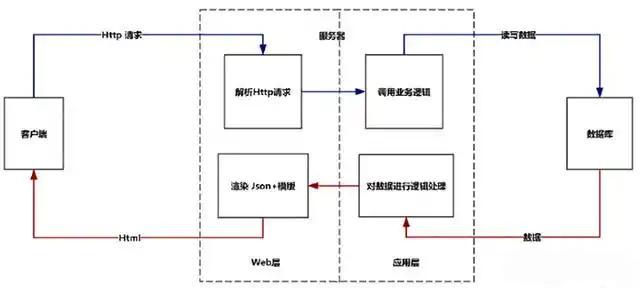
主要通过浏览器向用户展示蛋糕商城的各种页面，包括首页、商品列表页、商品详情页、购物车页面、用户登录注册页面以及订单相关页面等。这些页面以直观的方式呈现商品信息、用户账户信息、购物流程引导等内容，让用户能够方便地浏览商品、查看订单状态、管理购物车等。同时，接收用户在页面上的各种操作，如点击商品、输入数量、提交订单、登录注册等，并将这些操作转化为请求发送给控制器。

（2）**控制器层（Controller）**：

负责接收视图层传来的用户请求，根据请求的类型和参数进行判断和处理。例如，当用户在商品列表页点击某个商品分类时，控制器会调用相应的业务逻辑方法来获取该分类下的商品数据；当用户在购物车页面点击结算按钮时，控制器会处理订单创建的相关逻辑。在处理完业务逻辑后，根据结果选择合适的视图页面反馈给用户，如显示商品列表、订单成功页面或错误提示页面等，确保用户操作与系统响应之间的顺畅交互。

（3）**模型层（Model）**：

模型层包含业务逻辑类和数据访问对象（DAO）。业务逻辑类处理与蛋糕商城业务相关的核心逻辑，如商品的库存管理、促销活动的计算、订单的处理流程等。数据访问对象则负责与数据库进行交互，执行数据的查询、插入、更新和删除操作。例如，从数据库中获取商品信息以展示在商品列表页，将用户的订单数据保存到数据库，以及更新商品库存数量等，为整个系统提供数据支持和业务规则的实现。



5.1.2物理架构

（1）**客户端**：

用户使用各种浏览器（如常见的 Chrome、Firefox 等）访问蛋糕商城系统。浏览器是用户与系统交互的前端工具，它发送 HTTP 请求获取服务器上的资源，并将服务器返回的页面和数据展示给用户。用户在浏览器中输入商城网址后，即可浏览商品、进行购物操作，如将商品加入购物车、提交订单、查看个人信息等。

（2）**Web 服务器**：

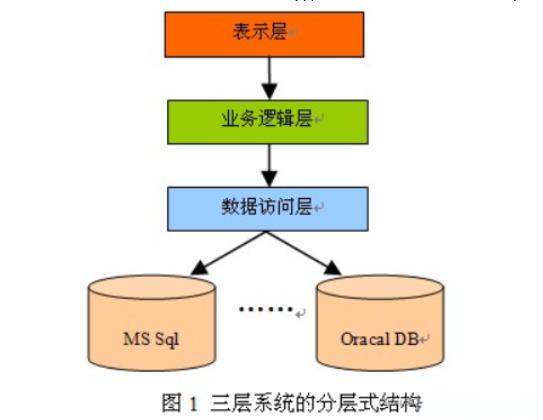
采用如 Tomcat 等 Web 服务器，主要接收客户端浏览器发送的 HTTP 请求。它能够解析请求的 URL，确定请求的资源类型。对于静态资源（如 HTML 页面、CSS 样式文件、JavaScript 脚本文件、图片等），Web 服务器直接从文件系统中读取并返回给客户端，提高系统对静态资源的响应速度。对于动态请求（如涉及业务逻辑处理的请求），则将请求转发给应用服务器进行处理。

（3）**应用服务器**：

与 Web 服务器紧密配合，运行服务器端的应用程序（如处理业务逻辑的 Servlet 和生成动态页面的 JSP）。它负责执行系统的业务逻辑，如处理用户登录验证、商品搜索、购物车操作、订单处理等功能。应用服务器管理系统运行所需的资源，确保在多用户并发访问时系统的稳定运行，高效处理各种业务请求。

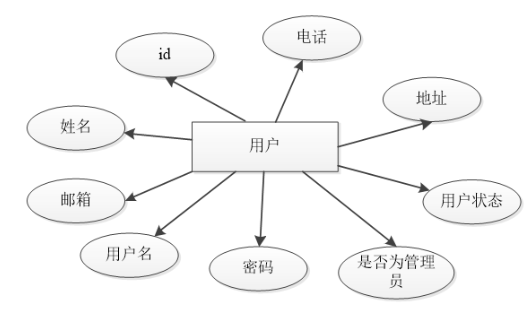
（4）**数据库服务器**：

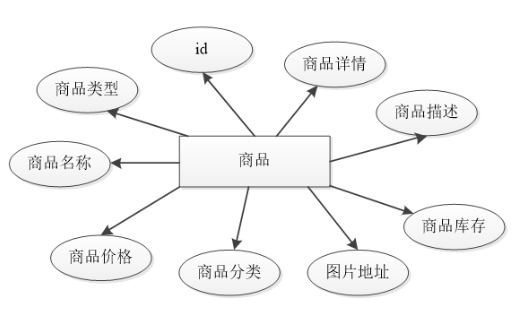
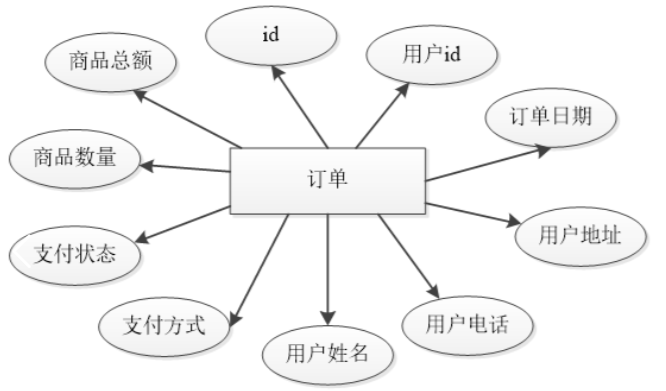
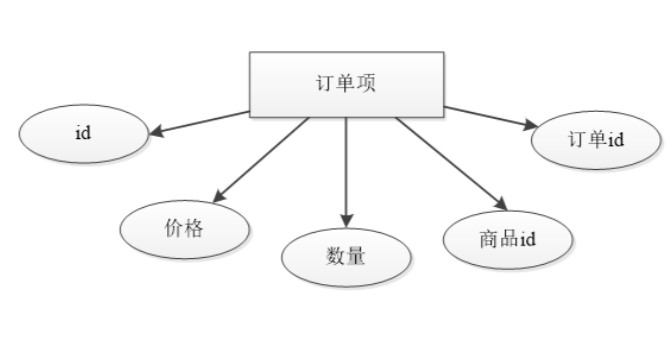
使用如 MySQL 等数据库管理系统，运行在独立的服务器上。数据库服务器存储蛋糕商城系统的所有数据，包括商品的详细信息（如名称、价格、描述、图片路径等）、用户信息（如用户名、密码、地址、联系方式等）、订单信息（如订单编号、商品明细、订单金额、下单时间、订单状态等）以及商品分类信息等。它能够快速响应对数据的各种操作请求，确保数据的一致性、完整性和安全性，为系统的正常运行提供数据基础

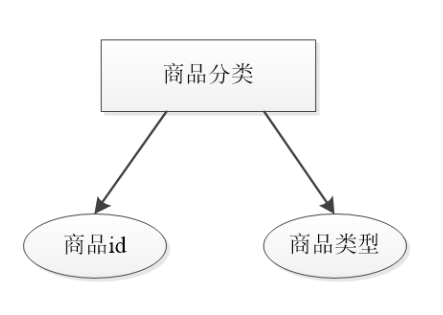
根据B/S的架构的需要，我们需要三层结构为**表现层、业务逻辑层、数据访问层** 。如上图所示本系统的物理架构基于 B/S 架构的常见部署方式，由不同的服务器和组件协同工作。

5.2类体系设计

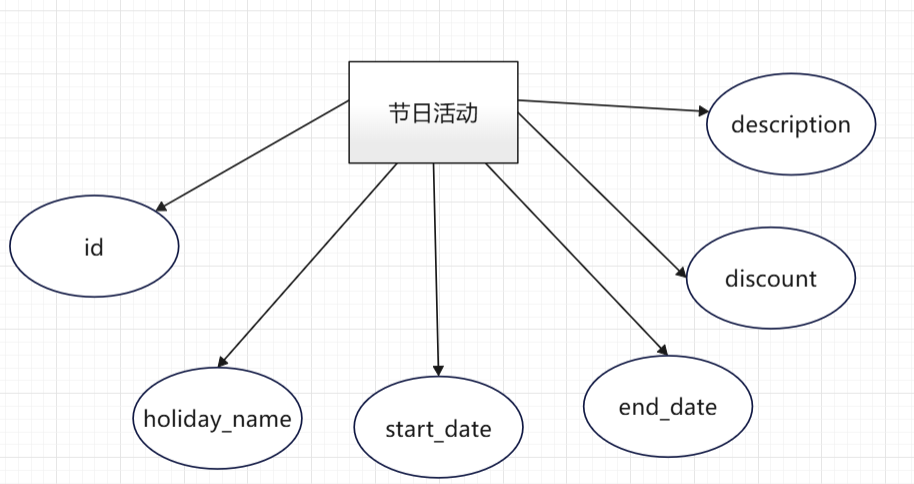
5.2.1 实体类的设计

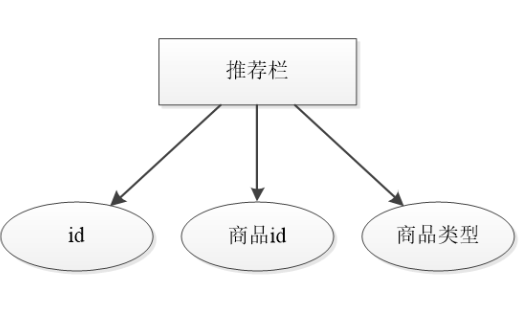
**1、User类**

1. **goods类**
2. **order类**
3. **order项实体类**
4. **type类**

****

1. **holiday类**

****

1. **推荐栏**

5.2.2 DAO类的设计

DAO是Data Access Object的缩写，它的作用是进行数据库的访问操作，负责与数据库进行交互操作，实现数据的持久化存储与读取，简单来说，DAO类就是操作数据库的实体类。DAO类包括两个工具类和多个操作类.对数据库进行操作首先连接数据库，用到了JDBC技术，JDBC 是 Java 提供的链接数据库的技术标准，它是专门执行 SQL 的 API.编写DBUtil类连接数据库，连接数据库的流程如下：

1、DBUtil数据库工具类

DBUtil是一个工具类，它的作用是对数据库进行连接，对数据库进行操作的时候都需要对数据库进行连，而连接需要提供数据库的名称、数据库的账号、数据库的密码、数据库的编码格式.如果每次都输入这些数据库信息会让代码十分冗余，所以把它封装成一个类，只需调用这个工具类就可以获取连接。

2、DateUtil日期工具类

日期工具类将java.util.Date类与java.sql.Timestamp 类的互相转换.在实体类中，用java.util.Date获取，所以需要将它们进行转换才能获取到正确的数据。

3、GoodsDao类

GoodsDao类包括多个方法，这多个方法都是基本的数据库增加、更新、查询和操作的操作，在后续的类中也会有这多个基本的方法。

基本的流程：1.调用DBUtil类对数据库进行连接2.使用sql语句对数据库进行增删查改操作3.使用executeQuery()把查询结果存放在ResultSet类对象4.在rs中取出需要的数据并且赋给对应的变量.5.释放资源6.将获取到的数据return。

(1)增加

public void add(Goods bean) ，使用insert into语句对goods表进行插入数据的操作。

(2)删除

public void delete(int id) ，使用delete语句对goods表进行删除数据的操作。

(3)修改

public void update(Goods bean) ，使用update语句对goods表进行删除数据的操作。

(4)根据id获取

public Goods get(int id) ，使用select 语句对goods表进行查询数据的操作。

(5)分页查询

public List<Goods> list(int start, int count) ，使用select 语句对goods表进行查询数据的操作。

(6)查询所有

public List<Goods> list() ，使用select 语句对goods表进行查询数据的操作。(7)获取总数

public int getTotal()，使用select 语句对goods表进行查询数据的操作。

4、TypeDao类

TypeDao类包括多个方法，以下是对基本数据库操作方法。

(1)获取所有类型

public List<Type> GetAllType() ，使用select语句对type表进行查询数据的操作。

(2)根据id获取类型

public Type select(int id) ，使用select语句对type表进行查询数据的操作。

(3)添加类型

public void insert(Type t) ，使用insert into语句对type表进行插入数据的操作。

(4)更新类型

public void update(Type t) ，使用update语句对type表进行更新数据的操作。

(5)删除类型

public void delete(int id) ，使用delete语句对type表进行删除数据的操作。

5、UserDao类

UserDao类包括多个方法，以下是对基本数据库操作方法。

(1)添加用户

public void addUser(User user) ，使用insert into语句对user表进行插入数据的操作。

(2)检查用户名是否存在

public boolean isUsernameExist(String username) ，使用select语句对user表进行查询数据的操作，判断用户名是否存在。

(3)检查邮箱是否存在

public boolean isEmailExist(String email) ，使用select语句对user表进行查询数据的操作，判断邮箱是否存在。

(4)根据用户名和密码查询用户

public User selectByUsernamePassword(String username, String password) ，使用select语句对user表进行查询数据的操作，根据用户名和密码获取用户信息。

(5)根据邮箱和密码查询用户

public User selectByEmailPassword(String email, String password) ，使用select语句对user表进行查询数据的操作，根据邮箱和密码获取用户信息。

(6)根据用户ID查询用户

public User selectById(int id) ，使用select语句对user表进行查询数据的操作，根据用户ID获取用户信息。

(7)更新用户地址

public void updateUserAddress(User user) ，使用update语句对user表进行更新数据的操作，更新用户的地址信息。

(8)更新用户密码

public void updatePwd(User user) ，使用update语句对user表进行更新数据的操作，更新用户的密码信息。

(9)获取用户总数

public int selectUserCount() ，使用select语句对user表进行查询数据的操作，获取用户总数。.

(10)获取用户列表（分页）

public List<User> selectUserList(int pageNo, int pageSize) ，使用select语句对user表进行查询数据的操作，获取用户列表并进行分页。

(11)删除用户

public void delete(int id) ，使用delete语句对user表进行删除数据的操作，根据用户ID删除用户。

(12)更新用户头像路径

public void updateAvatarPath(int userId, String avatarPath) ，使用update语句对user表进行更新数据的操作，更新用户的头像路径。

(13)获取用户头像路径

public String getAvatarPath(int userId) ，使用select语句对user表进行查询数据的操作，根据用户ID获取用户的头像路径。

6、OrderDao类

OrderDao类包括多个方法，以下是对基本数据库操作方法。

(1)插入订单

public void insertOrder(Connection con, Order order) ，使用insert into语句对order表进行插入数据的操作。

(2)获取最后插入的订单ID

public int getLastInsertId(Connection con) ，使用select last\_insert\_id()语句获取最后插入的订单ID。

(3)插入订单项

public void insertOrderItem(Connection con, OrderItem item) ，使用insert into语句对orderitem表进行插入数据的操作。

(4)查询用户的所有订单

public List<Order> selectAll(int userid) ，使用select语句对order表进行查询数据的操作，根据用户ID查询用户的所有订单。

(5)查询订单的所有订单项

public List<OrderItem> selectAllItem(int orderid) ，使用select语句对orderitem表进行查询数据的操作，根据订单ID查询订单的所有订单项。

(6)获取订单数量

public int getOrderCount(int status) ，使用select语句对order表进行查询数据的操作，根据订单状态获取订单数量。

(7)查询订单列表（分页）

public List<Order> selectOrderList(int status, int pageNumber, int pageSize) ，使用select语句对order表进行查询数据的操作，根据订单状态查询订单列表并进行分页。

(8)更新订单状态

public void updateStatus(int id,int status) ，使用update语句对order表进行更新数据的操作，根据订单ID更新订单状态。

(9)删除订单

public void deleteOrder(Connection con ,int id) ，使用delete语句对order表进行删除数据的操作，根据订单ID删除订单。

(10)删除订单项

public void deleteOrderItem(Connection con ,int id) ，使用delete语句对orderitem表进行删除数据的操作，根据订单ID删除订单项。

7、HolidayDAO类

HolidayDAO类包括多个方法，这多个方法都是基本的数据库增加、更新、查询和操作的操作。

(1)添加节日

public void addHoliday(Holiday holiday) ，使用insert into语句对holiday\_activities表进行插入数据的操作。

(2)删除节日

public void deleteHoliday(int id) ，使用delete语句对holiday\_activities表进行删除数据的操作。

(3)修改节日

public void updateHoliday(Holiday holiday) ，使用update语句对holiday\_activities表进行更新数据的操作。

(4)查询所有节日

public List<Holiday> getAllHolidays() ，使用select语句对holiday\_activities表进行查询数据的操作，获取所有节日信息。

(5)根据ID查询节日

public Holiday getHolidayById(int id) ，使用select语句对holiday\_activities表进行。

5.2.3 业务类的设计

业务层（Service层）是蛋糕商城系统的核心部分，它封装了系统的业务逻辑，为表现层提供服务接口，同时调用数据访问层（DAO层）进行数据的持久化操作.业务层的设计主要包括以下几个方面：

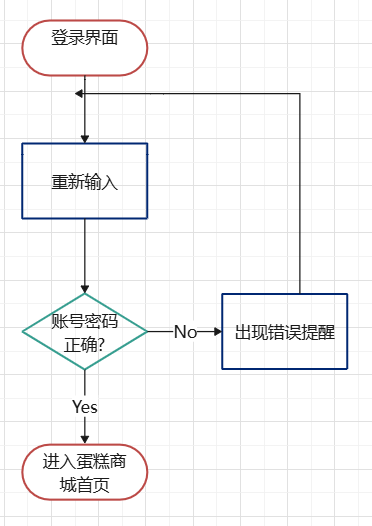
1. GoodsService实现类：实现了GoodsService接口定义的方法，通过调用GoodsDao进行数据访问操作，实现了商品的查询、添加、修改、删除等业务逻辑.例如，在getGoodsList()方法中，调用GoodsDao的getGoodsList()方法获取推荐类型的商品列表；在insert()方法中，调用GoodsDao的insert()方法将商品信息插入数据库。
2. OrderService实现类：实现了OrderService接口定义的方法，通过调用OrderDao进行数据访问操作，实现了订单的创建、查询、修改、删除等业务逻辑.例如，在addOrder()方法中，首先调用OrderDao的insertOrder()方法创建订单，然后获取订单ID，再调用OrderDao的insertOrderItem()方法插入订单项；在selectAll()方法中，调用OrderDao的selectAll()方法查询用户的所有订单，并为每个订单查询其订单项。
3. UserService实现类：实现了UserService接口定义的方法，通过调用UserDao进行数据访问操作，实现了用户的注册、登录、信息修改等业务逻辑.例如，在register()方法中，首先校验用户名和邮箱是否已存在，若不存在则调用UserDao的addUser()方法添加用户；在login()方法中，根据传入的用户名或邮箱和密码，调用UserDao的selectByUsernamePassword()或selectByEmailPassword()方法查询用户信息。
4. HolidayService实现类：实现了HolidayService接口定义的方法，通过调用HolidayDao进行数据访问操作，实现了节日的添加、删除、修改、查询等业务逻辑.例如，在addHoliday()方法中，调用HolidayDao的addHoliday()方法将节日信息插入数据库；在getAllHolidays()方法中，调用HolidayDao的getAllHolidays()方法查询所有节日信息。

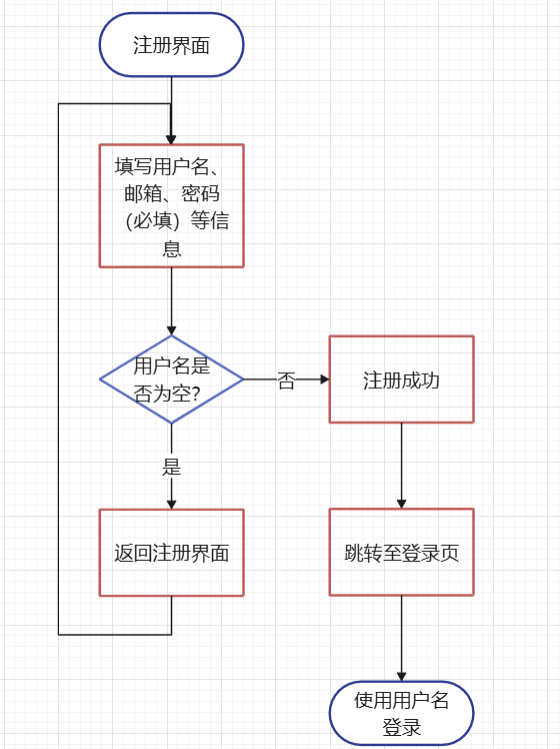
5.TypeService实现类：实现了TypeService接口定义的方法，通过调用TypeDao进行数据访问操作，实现了商品类型的查询、添加、修改、删除等业务逻辑.例如，在GetAllType()方法中，调用TypeDao的GetAllType()方法查询所有商品类型；在insert()方法中，调用TypeDao的insert()方法将商品类型插入数据库。

六、系统实现

6.1系统登录模块实现

在登录界面输入账号密码，系统判断账号密码，正确则进入蛋糕商城首页，错误则出现错误提醒并要求你重新输入。

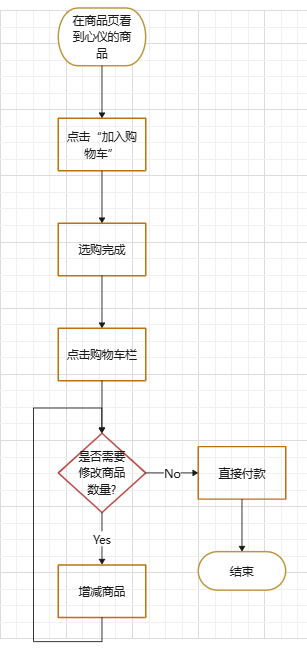


6.2系统注册模块实现

6.3 前台模块实现

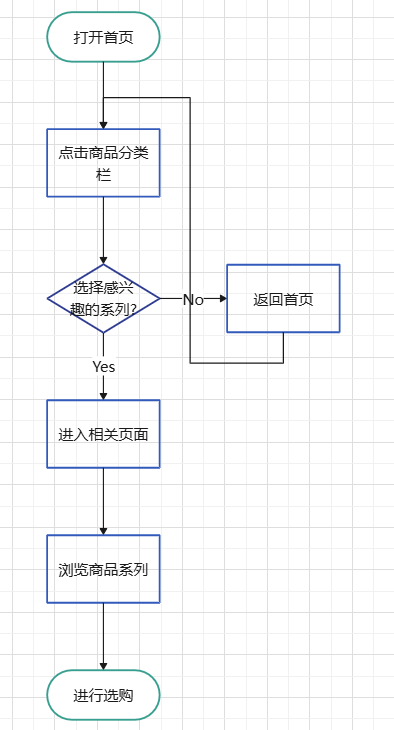
（1）购物车

在商品页看到心仪的商品，点击“加入购物车”，选购完成后点击购物车栏，可以在购物车栏对商品进行增减，如果不需要便可以直接付款。



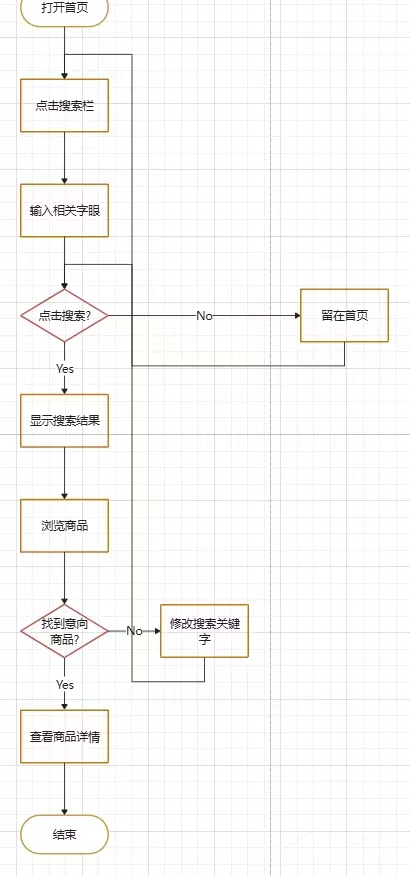
1. 商品分类查询

在首页点击商品分类栏，点击自己感兴趣的系列进入相关页面，便可以浏览自己感兴趣的商品系列并进行选购。



1. 商品搜索

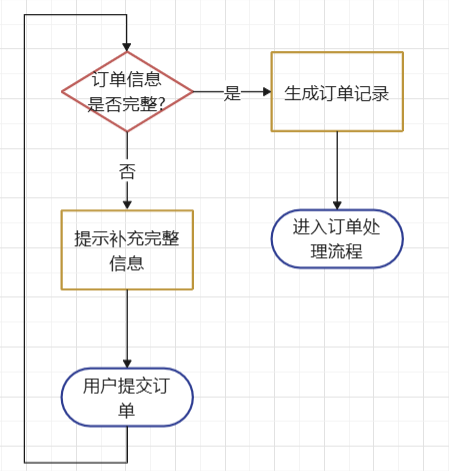
在首页点击搜索栏，输入相关字眼再点击搜索，便可以搜到意向商品。



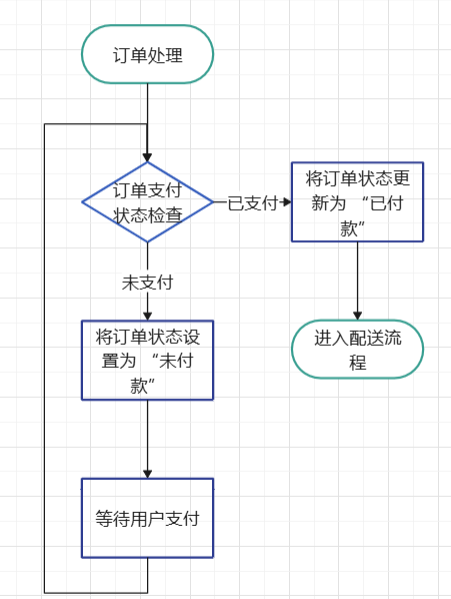
6.4 后台模块实现

6.4.1订单管理模块

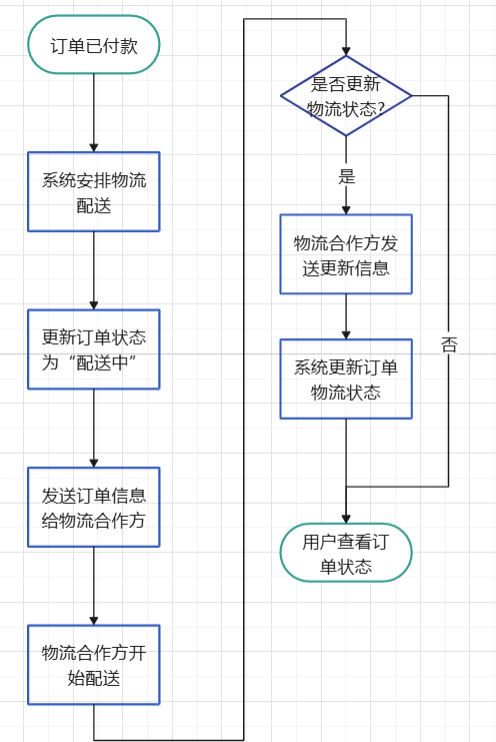
1. 订单提交

用户在前端界面提交订单，系统接收到订单信息后，首先判断订单信息是否完整（包括商品信息、用户信息等）。如果订单信息不完整，系统提示用户补充完整订单信息，并返回订单提交界面。如果订单信息完整，系统生成订单记录，进入订单处理流程。

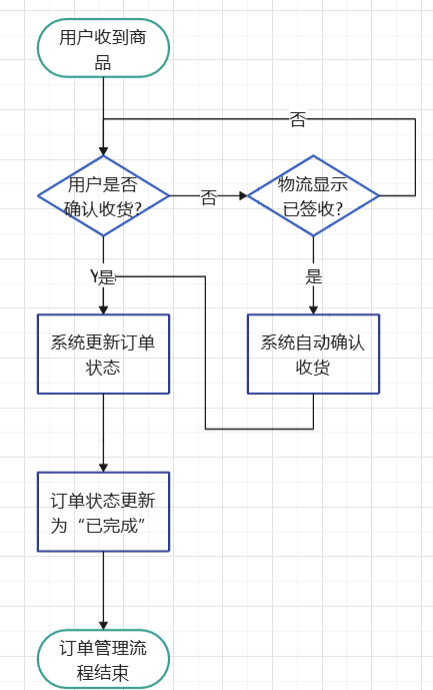
1. 订单处理

在订单提交界面，用户在前端界面提交订单。系统接收到订单信息后，首先判断订单信息是否完整（包括商品信息、用户信息等）。如果订单信息不完整，系统提示用户补充完整订单信息，并返回订单提交界面。如果订单信息完整，系统生成订单记录，进入订单处理流程。

1. 订单配送

系统对新生成的订单进行支付状态检查。如果订单未支付，系统将订单状态设置为 “未付款”，等待用户支付。如果订单已支付，系统将订单状态更新为 “已付款”，进入配送流程。

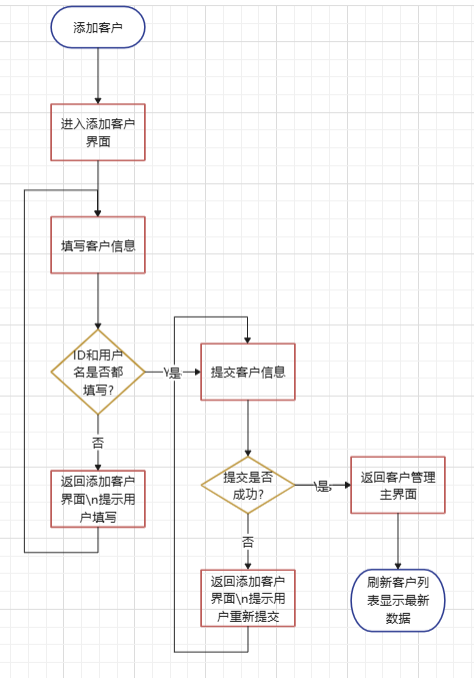
1. 订单完成

对于 “已付款” 的订单，系统安排物流配送。系统将订单状态更新为 “配送中”，同时将订单相关信息发送给物流合作方。物流合作方开始配送商品，并在配送过程中可能会更新物流状态信息给系统（如已发货、在途中等），系统相应地更新订单状态供用户查看。

6.4.2客户管理模块

1. 添加客户

点击 “添加客户” 按钮（位于界面右上角橙色按钮），进入添加客户界面。在添加客户界面中，填写客户信息，包括 ID、用户名、邮箱、收货人、电话、地址等。判断所有必填项（假设 ID 和用户名为必填项）是否都已填写，若有必填项未填写，则返回添加客户界面提示用户填写，若必填项已填写，则进入下一步。提交客户信息，判断信息提交是否成功，若提交失败（例如数据库连接错误等），则返回添加客户界面提示用户重新提交，若提交成功，则返回客户管理主界面，并刷新客户列表显示最新数据。



1. 对客户操作管理



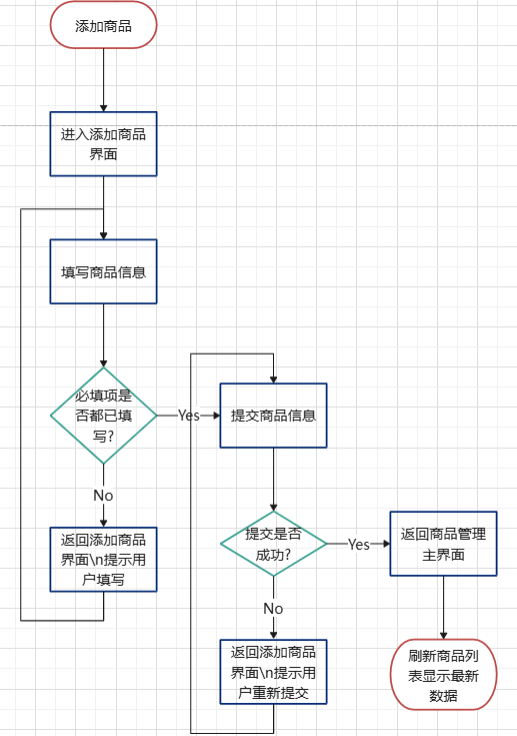
6.4.3商品管理模块

1. 添加商品

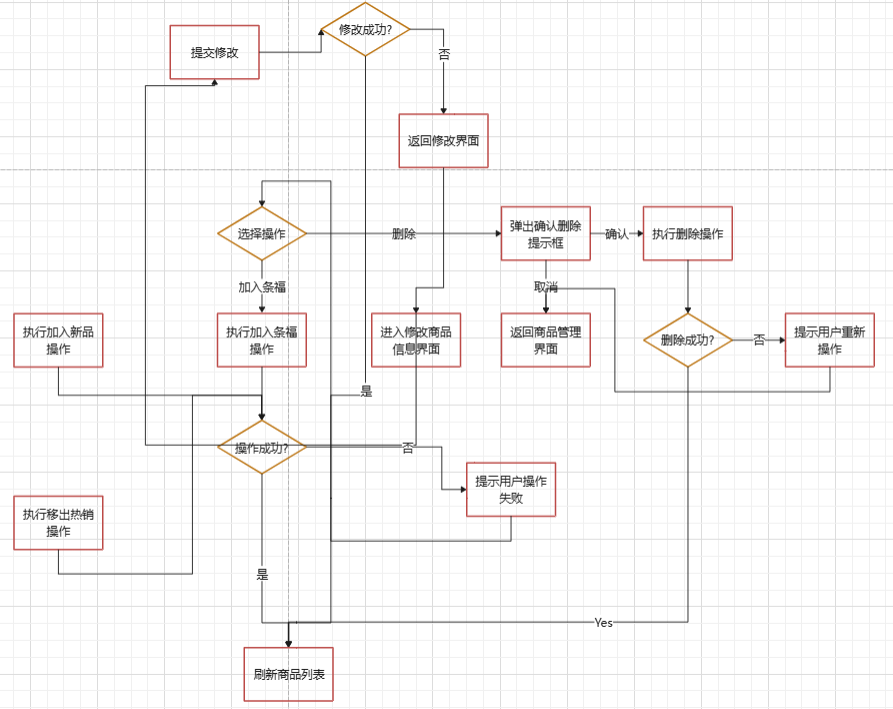
点击 “添加商品” 按钮（位于界面右上角橙色按钮），进入添加商品界面。

在添加商品界面中，填写商品信息，包括 ID、图片、名称、介绍、价格、类目等（假设 ID 和名称为必填项）。判断所有必填项是否都已填写，若有必填项未填写，则返回添加商品界面提示用户填写，若必填项已填写，则进入下一步。

提交商品信息，判断信息提交是否成功，若提交失败（例如数据库连接错误等），则返回添加商品界面提示用户重新提交，若提交成功，则返回商品管理主界面，并刷新商品列表显示最新数据。

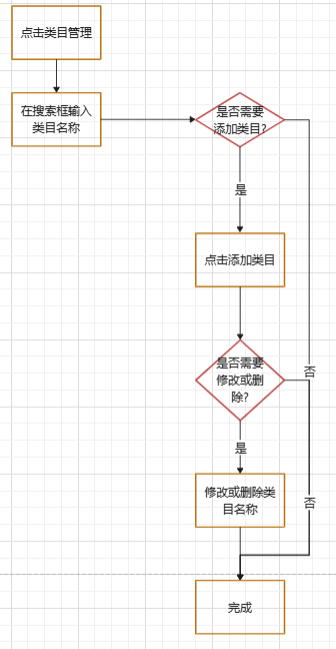


1. 商品操作



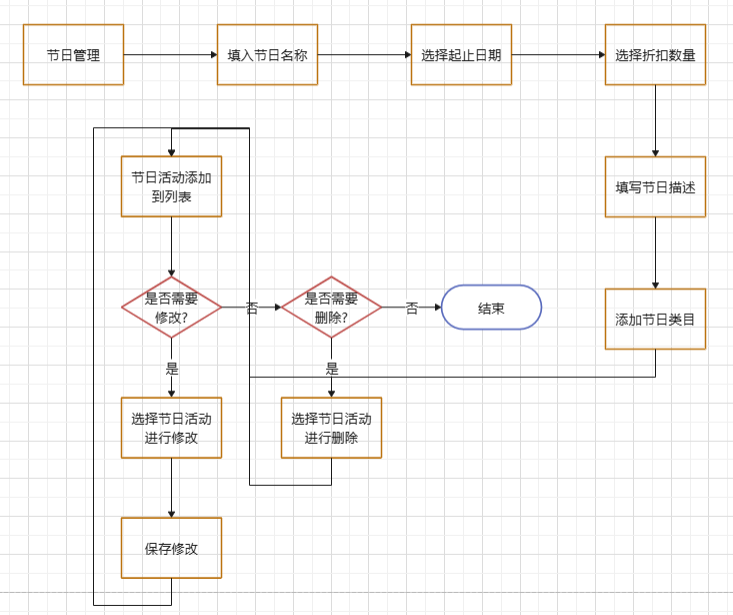
6.4.4类目管理模块

点击类目管理，在搜索框输入类目名称，点击添加类目即可添加成功，如有需要可以对类目名称进行修改或删除



6.4.5节日管理

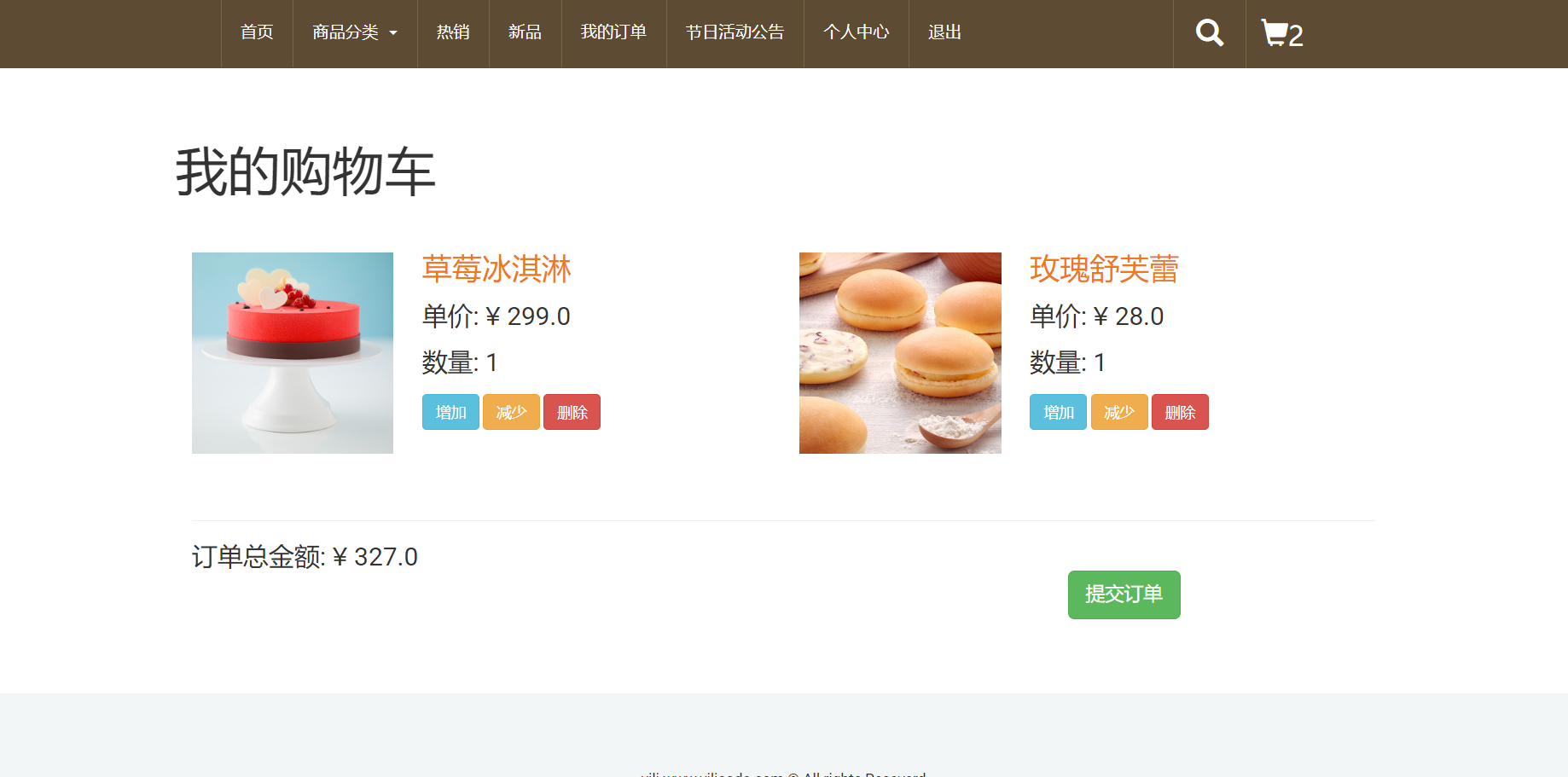
点击节日管理，在框里填入节日名称，选择起止日期，选择折扣数量，填写节日描述，添加节日类目。在列表中，可以对已有的节日活动进行修改和删除



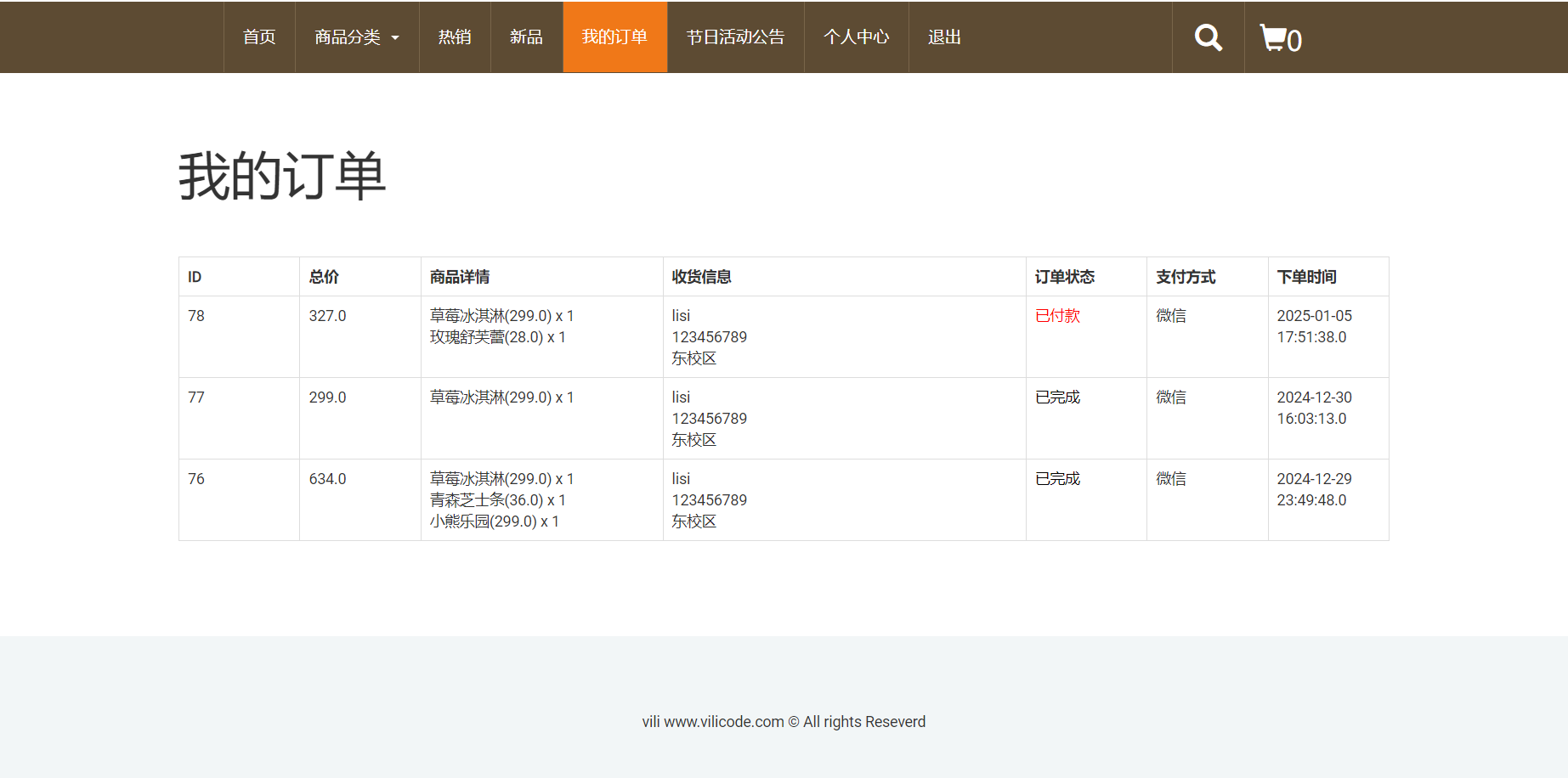
1. 系统测试

对系统功能进行测试，确保各项功能可以运行。关键功能如下图所示。

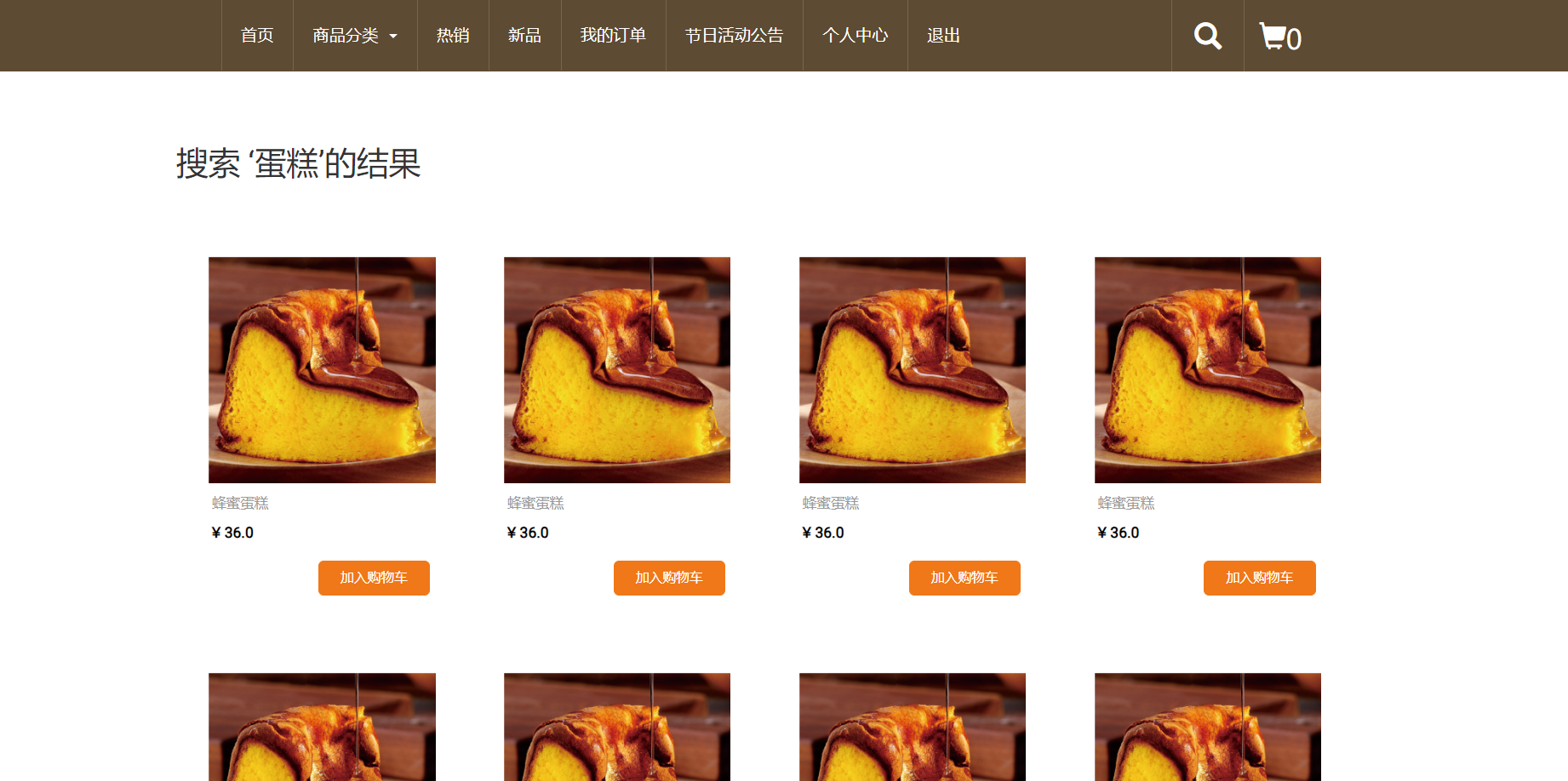
1. 购物车



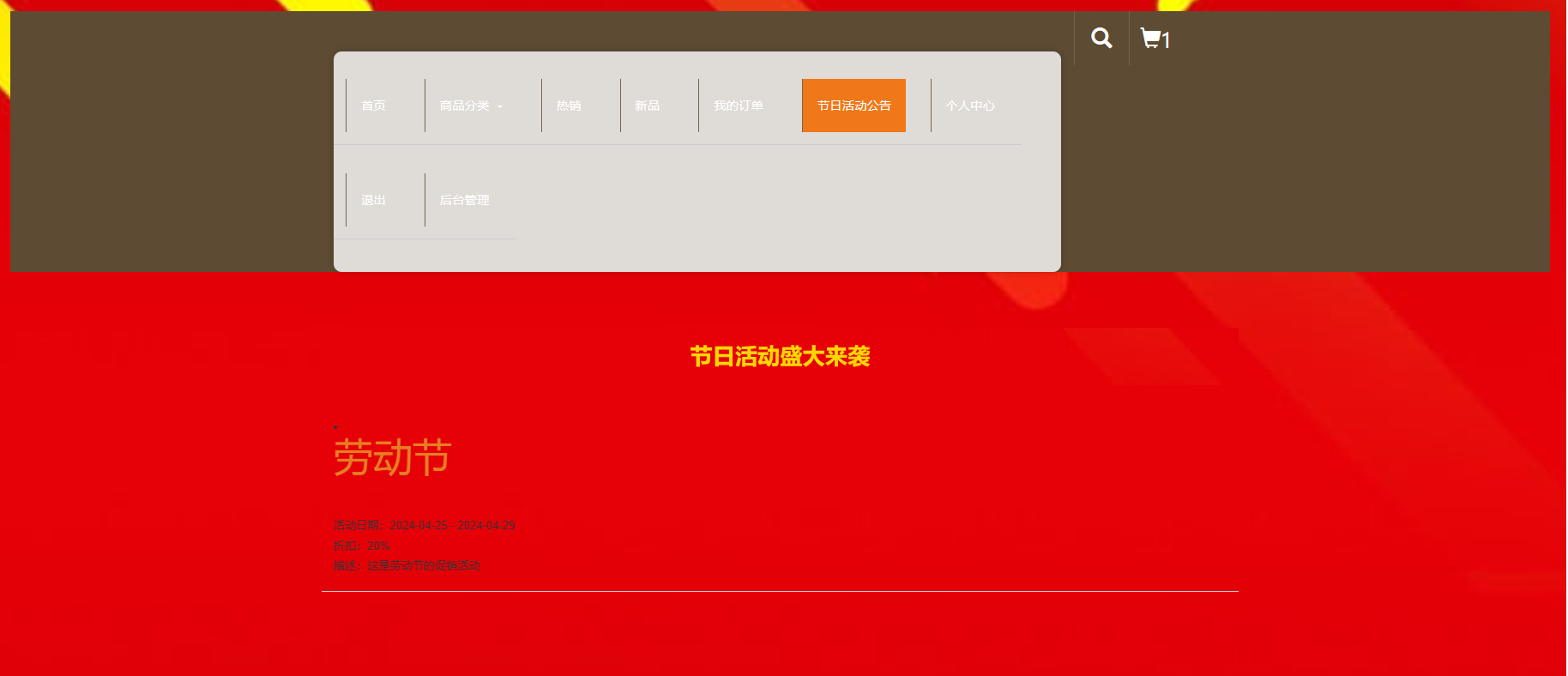
1. 订单支付



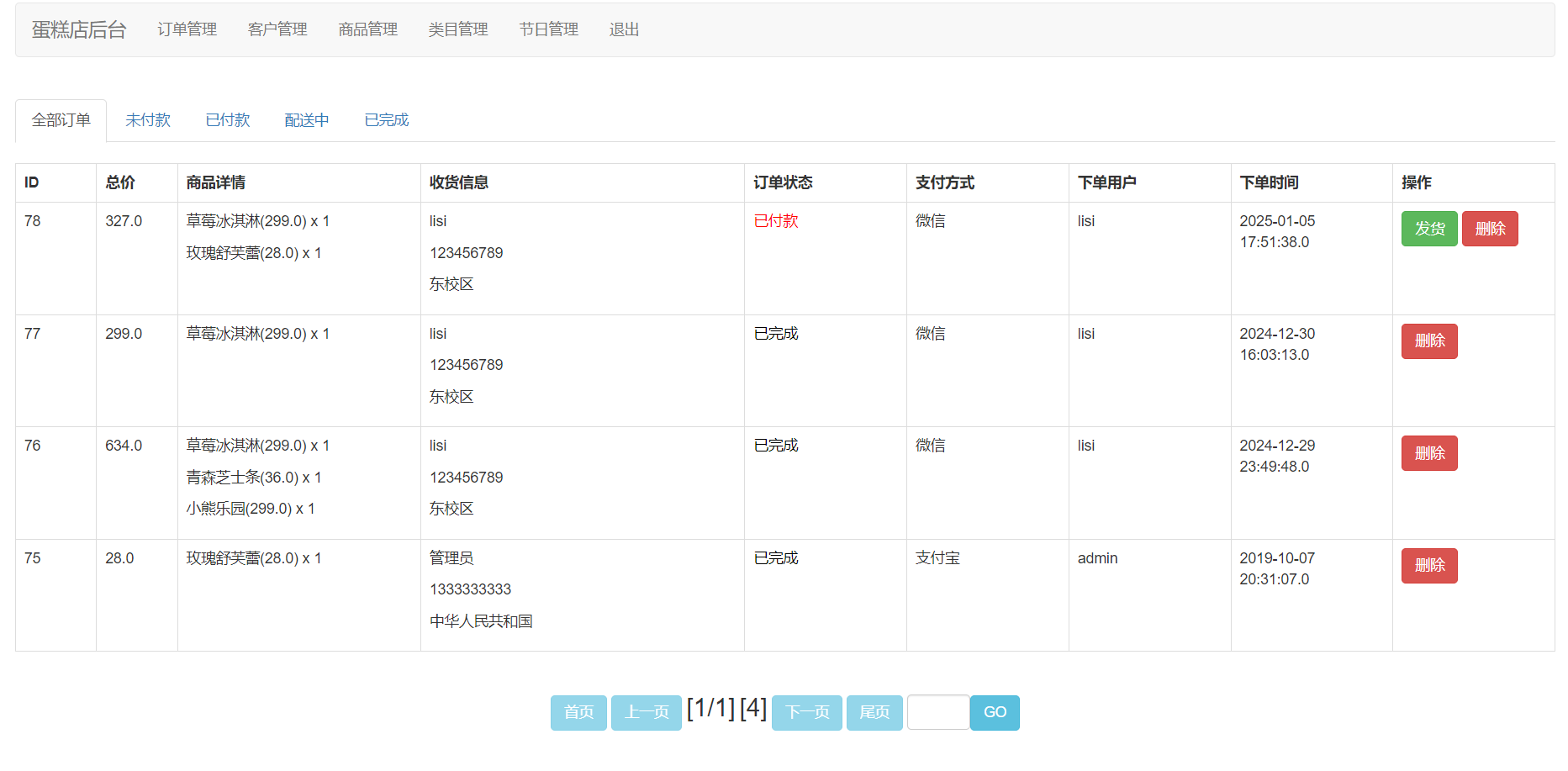
1. 商品搜索



1. 节日活动公告



1. 订单管理



1. 客户管理



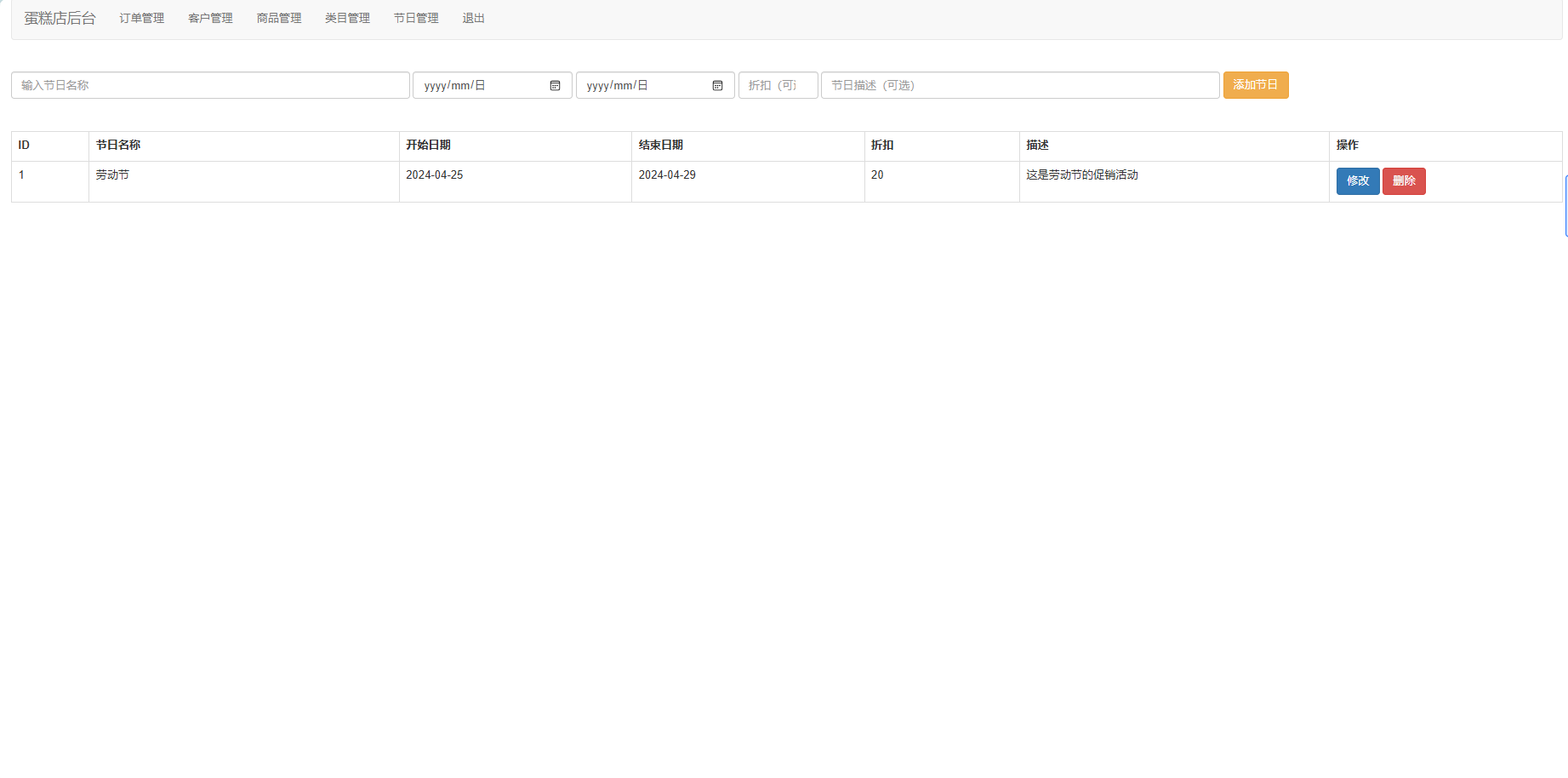
1. 商品管理



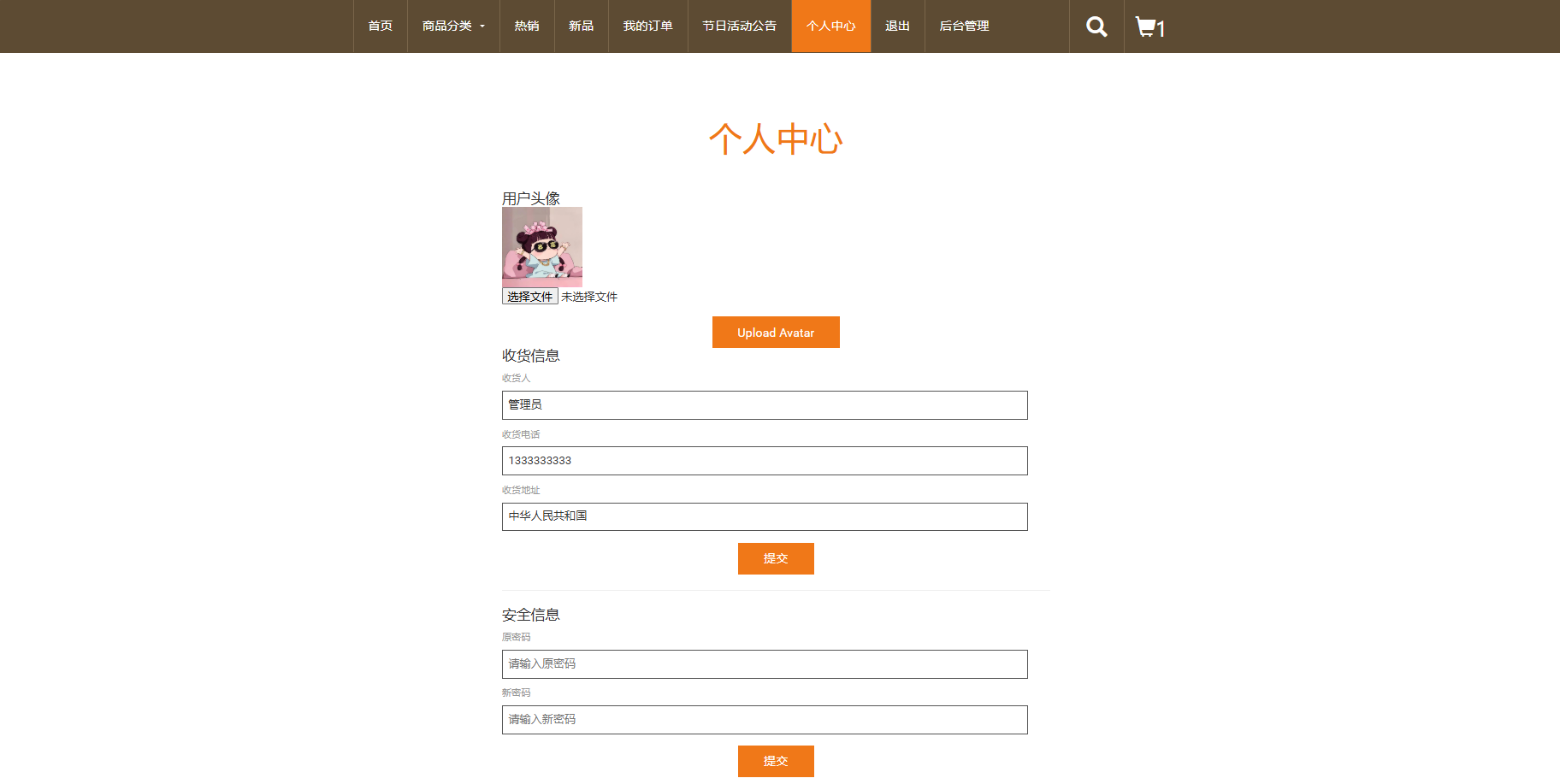
1. 类目管理



1. 节日管理



1. 个人中心



八、结论展望与体会

通过本次网上蛋糕商城项目的实践，我们收获颇丰。网上蛋糕商城的基本功能已完善，但是还有许多不完善的地方。由于技术的不足，我们也遇到了诸多技术难题，如头像的上传更新、节日管理同步到公告等，通过查阅资料、反复调试，不仅成功攻克，更提升了自己解决实际问题的能力。同时，对团队协作也有了全新认识，与小组成员分工合作，沟通需求、整合代码，明白了集思广益、优势互补在项目推进中的关键作用。

九、小组成员分工

李若瑛：新增功能代码、技术文档部分画图编写

冯丽芷：技术文档内容编写、节日活动公告代码初始框架