



湖南电子科技职业学院
HUNAN VOCATIONAL COLLEGE OF ELECTRONIC AND TECHNOLOGY

产品设计	方案设计	工艺设计
√		

信息工程学院

毕 业 设 计

题目： 基于 SpringBoot 的堂食管理系统的设计与实现

学生姓名 张佳凤

学生学号 010425190833

班级名称 软件 G32204

专业名称 软件技术

指导教师 周先文

2025 年 5 月

毕业设计真实性承诺及指导教师声明

本人郑重声明：所提交的毕业设计是本人在指导教师的指导下，独立进行研究工作所取得的成果，内容真实可靠，不存在抄袭、造假等学术不端行为。除文中已经注明引用的内容外，本设计不含其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。如被发现设计中存在抄袭、造假等学术不端行为，本人愿承担相应的法律责任和一切后果。

学生（签名）： 张佳凤 日 期：2025.5.10

指导教师关于学生毕业设计真实性审核的声明

本人郑重声明：已经对学生毕业设计所涉及的内容进行严格审核，确定其成果均由学生在本人指导下取得，对他人成果的引用已经明确注明，不存在抄袭等学术不端行为。

指导教师（签名）： 周敏 日 期：2025.5.10

（注：本页学生和指导教师须亲笔签名。）

目录

一、项目概述.....	1
(一) 设计背景.....	1
(二) 设计意义.....	1
(三) 设计思路.....	1
二、需求分析.....	3
(一) 技术可行性分析.....	3
(二) 系统需求分析.....	3
三、系统概要设计.....	8
(一) 功能模块设计.....	8
1.前台模块.....	错误! 未定义书签。
2.管理员后台模块.....	错误! 未定义书签。
四、数据库设计.....	10
(一) 数据库概念模型设计.....	10
(二) 数据库表结构设计.....	11
五、系统详细设计与实现.....	14
(一) 前端模块.....	14
1.前端首页.....	14
2.用户登录.....	14
3.用户注册.....	16
4.商品详情.....	17
(二) 后台模块.....	19
1.管理员登录.....	19
2.用户管理.....	20
3.地址管理.....	20
六、系统测试.....	21
(一) 测试方法.....	21
(二) 系统测试用例.....	21
1.用户登录测试.....	21
2.用户注册.....	22
3.用户管理测试.....	23
表 6.5 用下单户表.....	25
(三) 测试结论.....	25
总结.....	26
参考资料.....	27
附录后台接口源码.....	28

一、项目概述

（一）设计背景

随着经济的稳步增长，人们的生活水平显著提高，这导致了消费目标和水平的逐步提升。在饮食和娱乐方面，消费者的需求变得更加多样化和个性化，推动了餐饮行业的快速发展和创新。

中国的餐饮业正在紧跟时代的步伐，通过调整产业结构和扩大企业规模来满足消费者日益增长的需求。许多餐饮企业通过建立自己的网站，利用网络的力量，展示特色服务，吸引顾客关注。这些网站因其受欢迎程度而受到广泛欢迎，而那些未能与时俱进的网站则面临被淘汰的风险。

餐饮行业与人们的日常生活密切相关，随着科技的进步，电子商务平台的兴起为餐饮业带来了新的机遇。各具特色的企业文化在这一平台上大放异彩，形成了独特的竞争力。然而，随着社会的发展，企业需要不断创新，加大投入，以适应和满足消费者的需求。

（二）设计意义

在用户浏览网站时，美食页面的首页是他们接触的第一个界面，因此其设计对于吸引用户至关重要。首页应迅速引导用户获取他们感兴趣的信息，例如最新的美食趋势，并提供便捷的注册和登录机制，以使用户能够迅速加入网站社区。

首页的布局应以丰富的美食内容为核心，这些内容不仅包括本地美食信息，也应涵盖全球美食动态，为用户提供全面的餐饮资讯。通过展示最新的美食资讯，网站能够吸引用户的注意力，并激发他们进一步探索的兴趣。

用户完成注册并登录后，可以在首页进行各种操作，如搜索特定美食、查看推荐餐厅、阅读美食博客或参与社区讨论。这种设计不仅提升了用户体验，也增强了用户对网站的忠诚度。

（三）设计思路

这些年来。社会上的人们生活上的快速的进步，我们已经快速的踏入了信息大众化的时代。我们的社会中的大众的消费的观念与之人类的生活的改变，互联网上其中不同的平台的商品是可以着凭借简单的、快捷的、方便的特点来融入我

们人们的生活其中，而且快速适应了当今的社会中的快节奏的生活，互联网上商品的购物已经是人们生活的一部分，商店的东西可以让我们用户们不用出去就可以方便的快捷的轻松的买自己的喜欢的商品，所以，我这个开发的美食中管理的系统是融合了当前 IT 的技术，设计开发了一款美食的管理中系统，给不同的使用钱的消费的人来购买商品，我们去完成着基本信息的管理，为商店的货在互联网上的售出拉高了成绩，去帮助后台的商品中管理拉高了工作的效率。

二、需求分析

（一）技术可行性分析

Java 技术因其在 Web 系统开发中的广泛应用而成为 Web 应用程序开发的主要语言之一。Java 的面向对象特性、高安全性、可靠性、强可移植性以及高平台适配度使其成为实现美食管理系统的理想选择。

在本项目中，我们采用 Java 语言开发美食管理系统。系统后端基于 Java Web 技术，核心控制采用 Servlet 技术，结合 Maven 快速搭建系统环境和数据库管理依赖。前端则利用 JSP、HTML、CSS、AJAX、JavaScript 和 Font Awesome 等技术，实现基于 B/S 架构的系统设计。后端数据存储选用 MySQL 数据库，实现前后端数据交互。

美食管理系统采用 B/S 架构设计，基于三层架构（Maven）实现工作流程，便于管理和调整前端和后台业务。架构分为以下三层：

表现层：作为 B/S 架构的第一层，面向客户端，负责实现输入输出功能。表现层将后台信息封装成 API 接口，供前端 View 层调用并返回数据。

逻辑层：作为 B/S 架构的第二层，位于 Web 服务器中，负责实现数据传递和业务逻辑处理。逻辑层主要设计美食管理系统的核心功能实现。

数据层：作为 B/S 架构的第三层，后台数据库服务器负责接收用户请求，根据逻辑层的业务逻辑，通过 SQL 语句对后台数据库中的数据进行处理。美食管理系统使用 MySQL JDBC 处理数据持久层操作，将处理结果反馈给 Web 服务器进行响应和对接。

（二）系统需求分析

美食中的管理的系统现在的用户区分成二种，前端的使用用户为基本使用的用户还有后台的使用的用户就是管理员，大部分都是通过基本的用户其中的行为进行了详细的说明了系统中的需求然后画出来用户中的用例图。

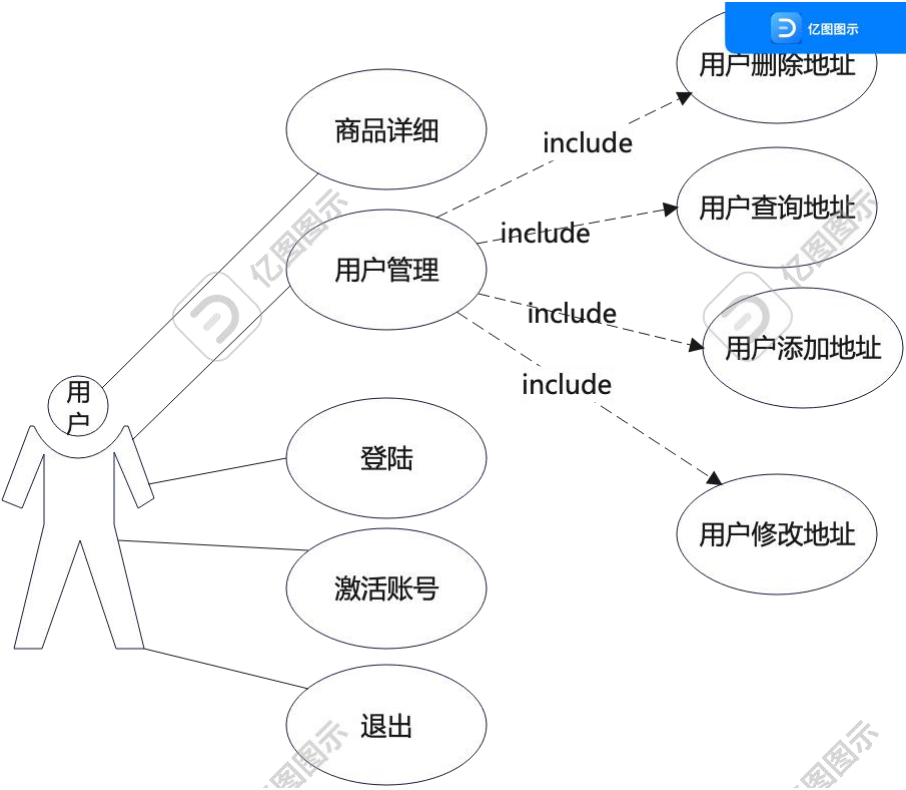


图 2.1 管理员用例图

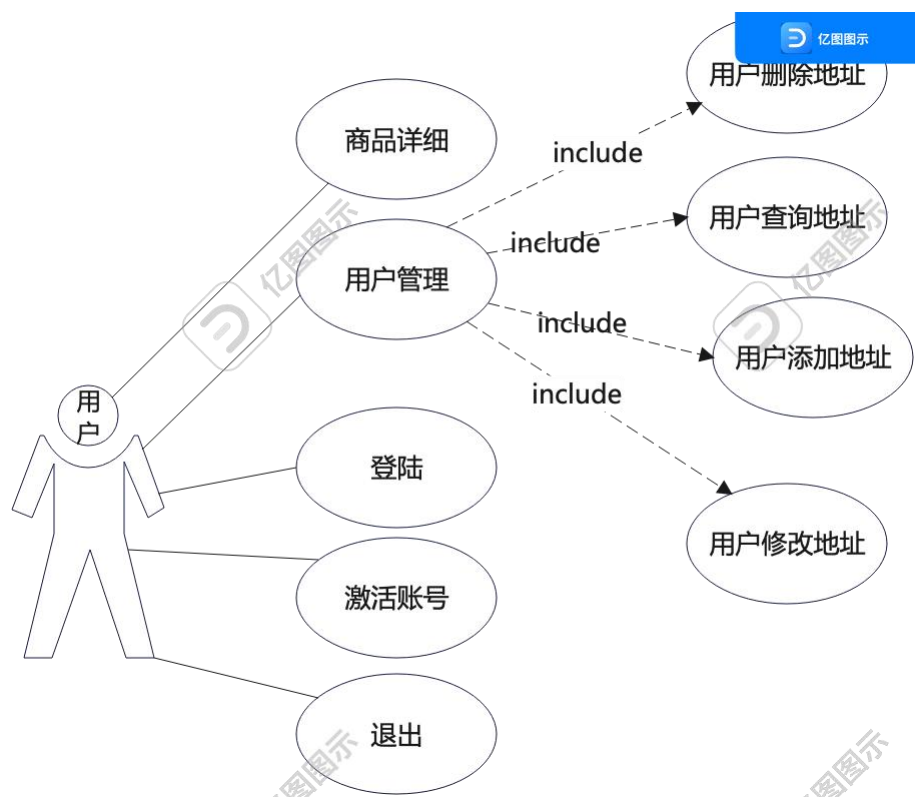


图 2.2 普通用户用例图

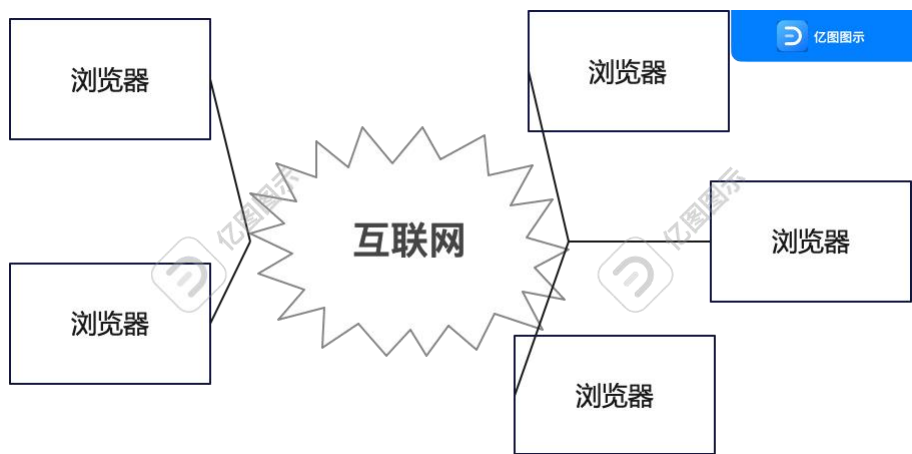


图 2.3 B/S 结构

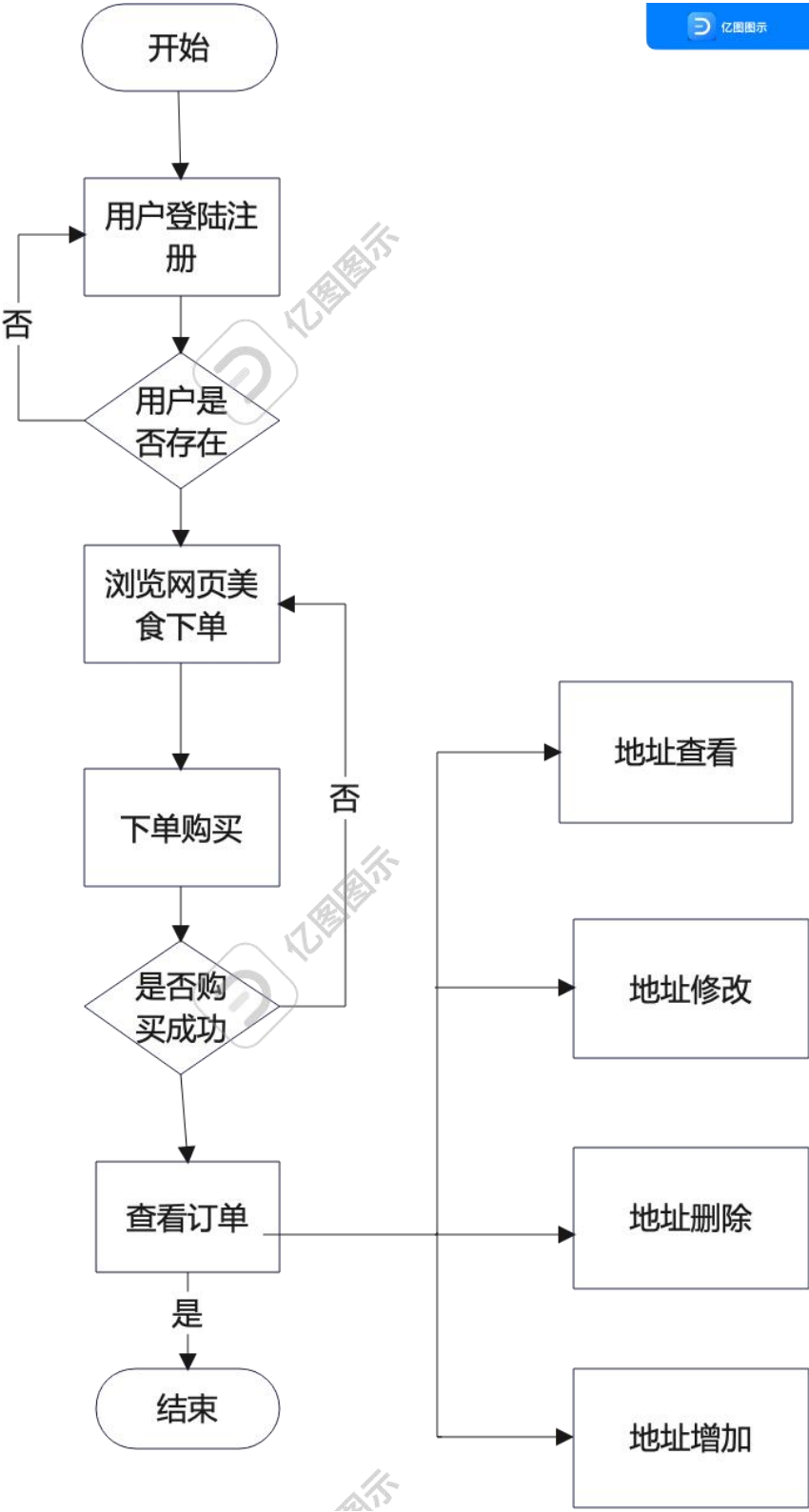


图 2.3 前端用户操作流程图

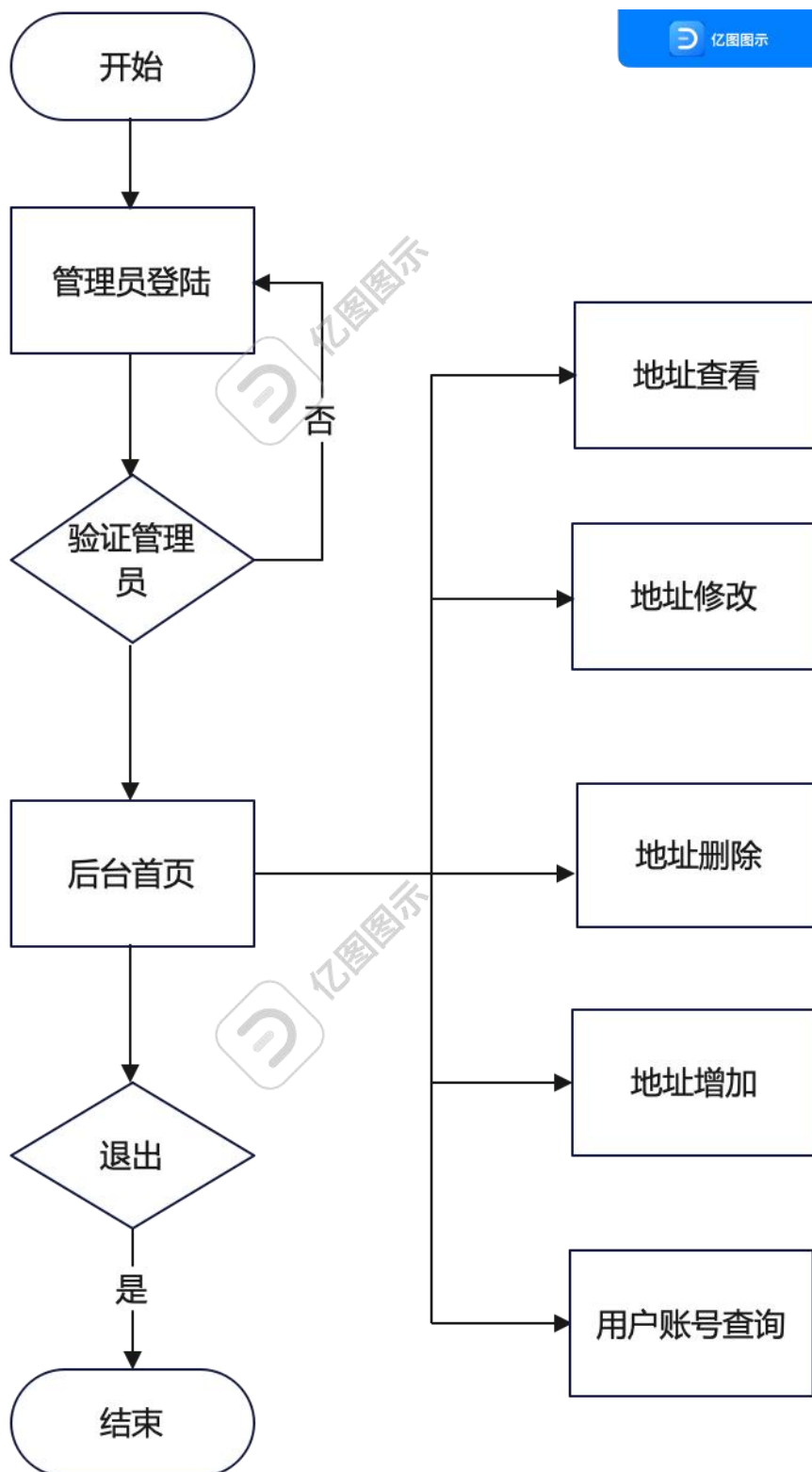


图 2.4 管理员模块流程图

三、系统概要设计

（一）功能模块设计

我们对于在对管理的美食购物的系统功能上的需求的功能分析，给出了我们设计的美食的管理系统制作出的功能的框图。如下图 3.1，3.2 所示。

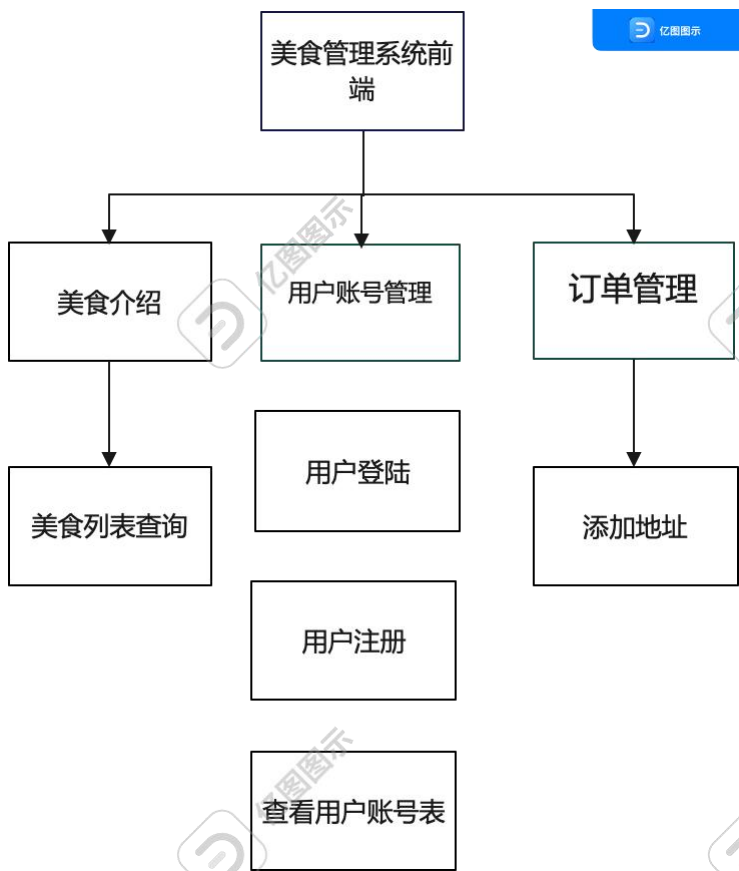


图 3.1 系统前端功能模块图

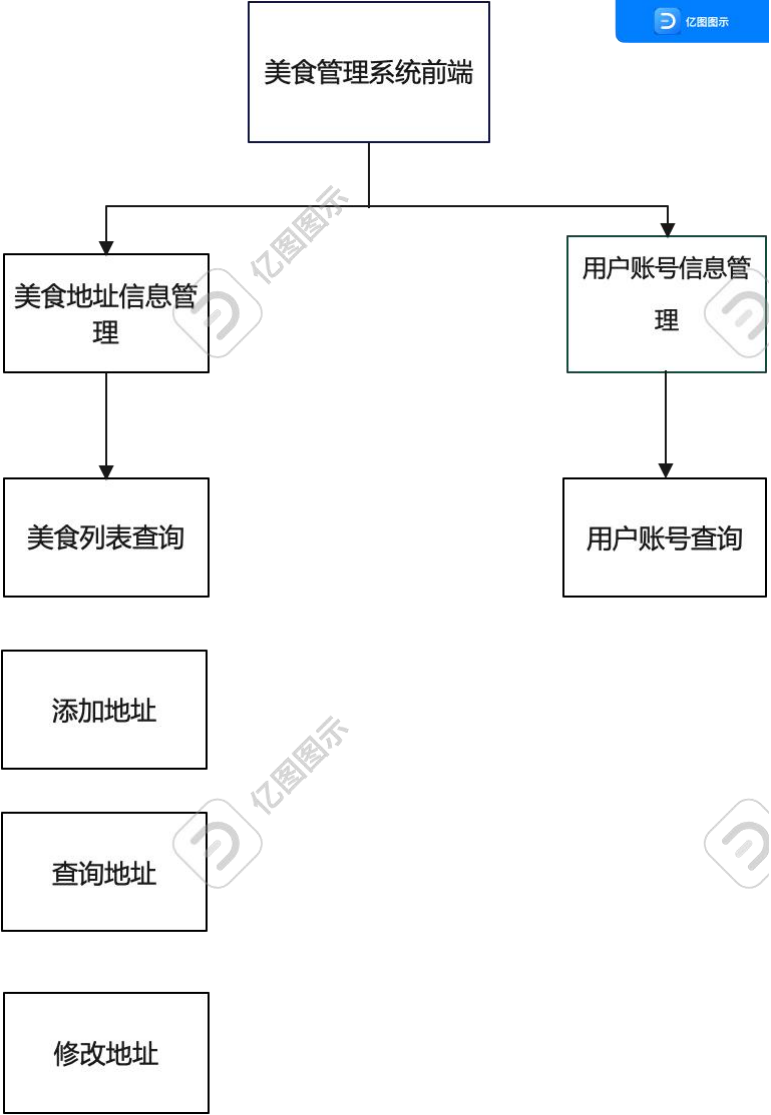


图 3.2 系统后台功能模块图

四、数据库设计

（一）数据库概念模型设计



图 4.3 普通用户 E-R 图

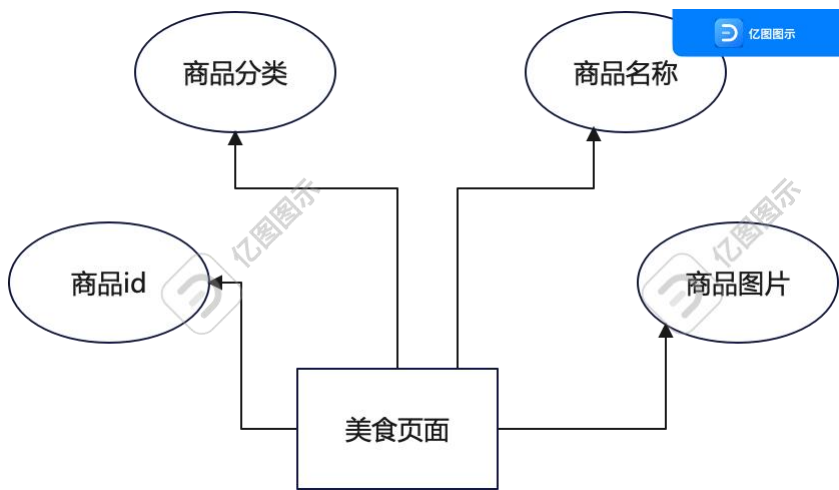


图 4.5 商品 E-R 图



图 4.8 管理员 E-R 图

(二) 数据库表结构设计

系统中的设计其中的根据的需求中的基本的表格：用户的表格、美食的地址的表、管理员所需要显示的表格
计如下。

用户表（user），该表主要存储了普通用户信息，具体设计如表 4.1 所示。

表 4.1 用户表

字段名	字段说明	类型	长度	备注
name	账号	varchar	255	
password	密码	int	255	

美食地址表主要存储记录本系统中地址信息表，详情见下表 4.2 所示

表 4.2 美食地址表

字段名	字段说明	类型	长度	备注
id	ID	int	11	pk
name	名字	varchar	32	
password	密码	varchar	32	
Email	邮箱	varchar	32	
country	地址	varchar	32	

表 4.3 管理员表

字段名	字段说明	类型	长度	备注
name	账号	varchar	255	

password	密码	int	255
----------	----	-----	-----

我们设计的美食的系统的管理中管理员所主要需要存储有关美食的系统的管理员的信息数据。详细设计参照表 4.3

五、系统详细设计与实现

（一）前端模块

1. 前端首页

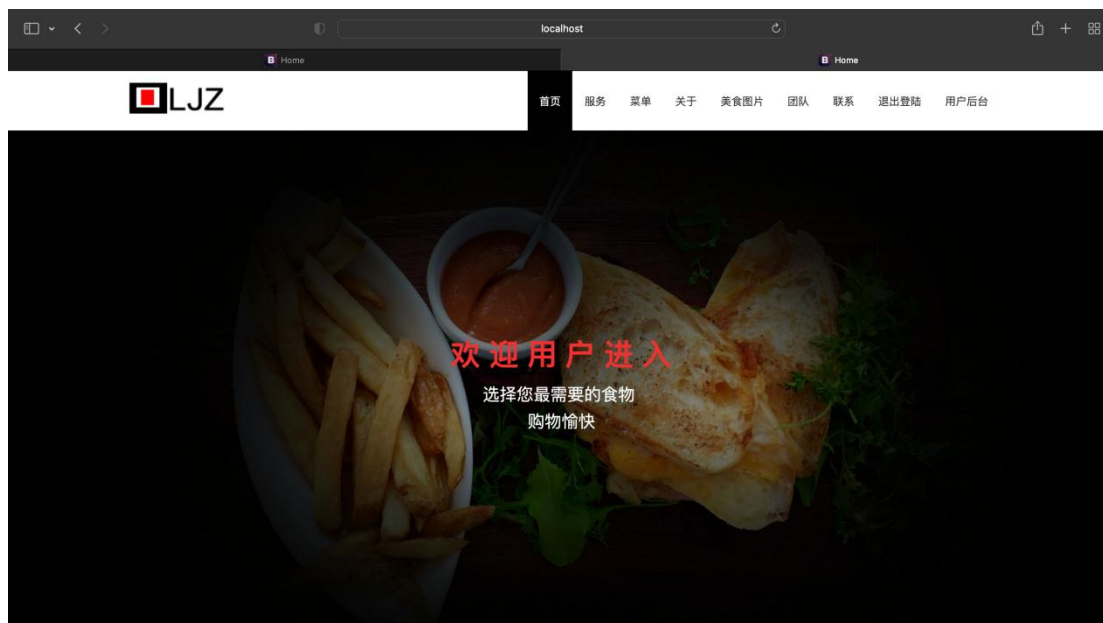


图 5.1 首页图

2. 用户登录

保证我们这次美食的管理中的系统中的数据的安全，我们的美食的管理中的系统就需要去访问我们这里的用户访问我们美食的管理中系统的时候只能从美食的管理中系统的登录的页面访问，只能在输入对的用户的账号和密码才可以访问到美食的系统，详细的用户的登录方法如如下所示。

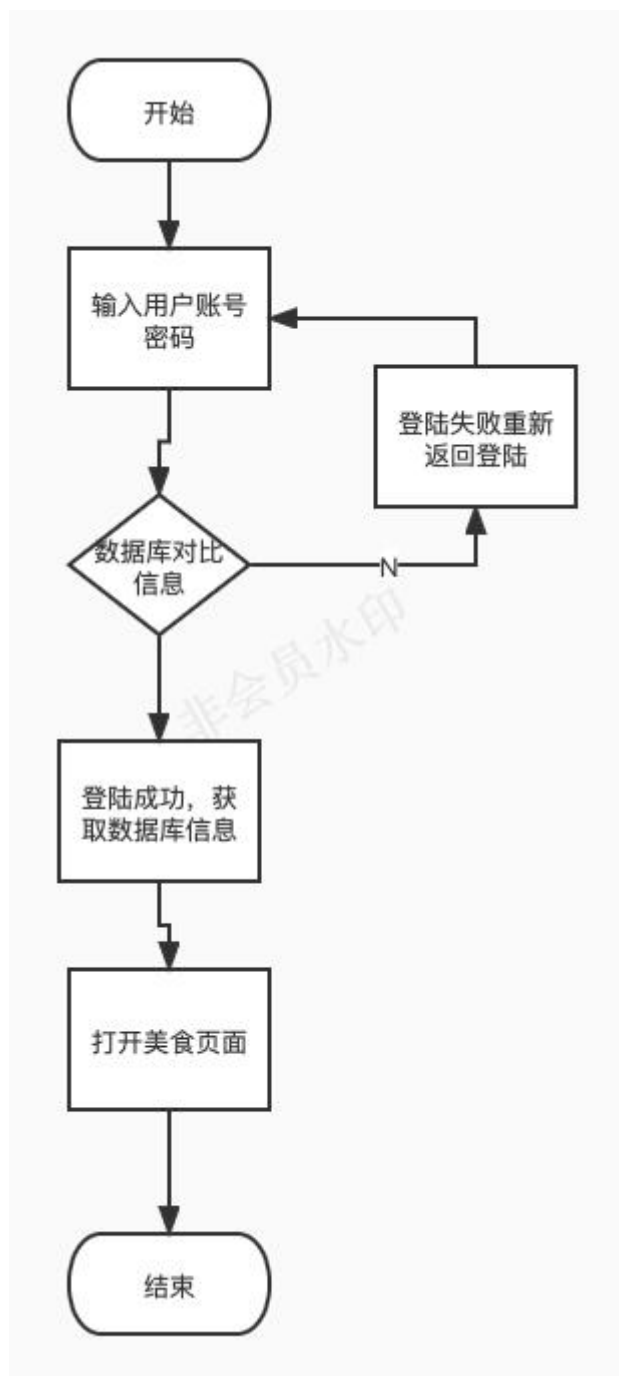


图 5.2 登录流程图

我们设计的面美食的管理系统中用户在登陆我们所设计的首页进行输入自己的账号信息和密码信息后, 点击我们的提交按钮然后提交到我们所需要到的后台进行反馈。再后端的代码进行相对应的判断再进入我们存储信息的数据库里面的信息对比后将结果返回到前端登录页面, 用户登录界面设计如图 5.3 所示。

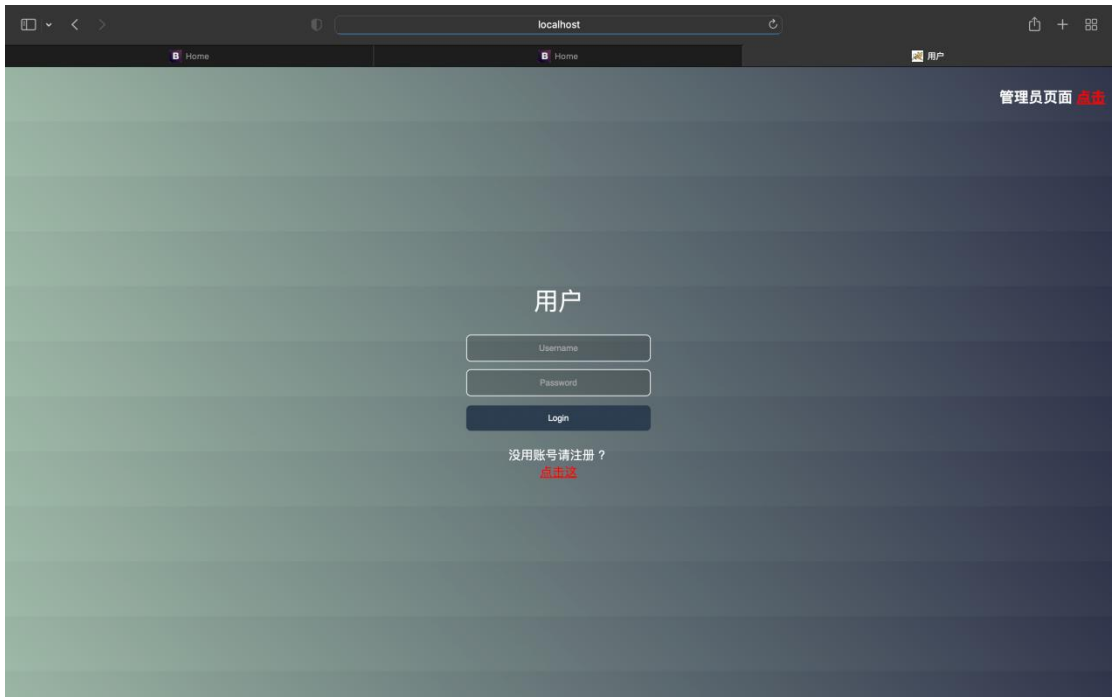


图 5.3 登陆页面

3. 用户注册

在我们这次的用户中登录的访问前面就只能先去先注册好用户才能，在用户注册的时候的页面，去输入我们用户的账号还有密码，然后去点击注册的按钮给我们的后台去传达注册请求，操作的流程下面所示

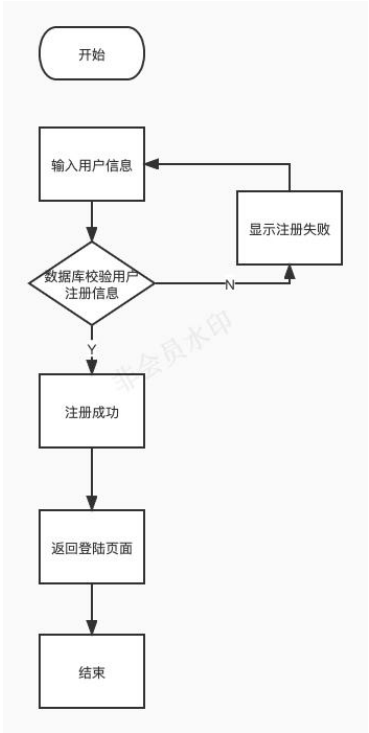


图 5.4 注册页面

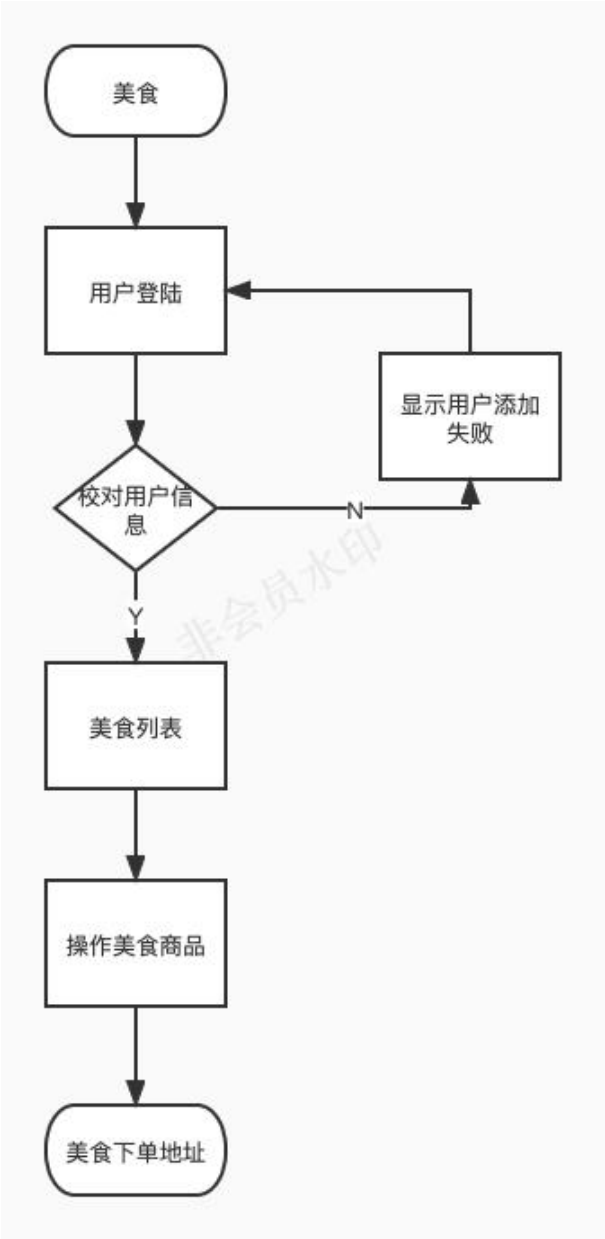


图 5.7 购买下单流程图

4. 商品详情

在我们所设计的首页中下单的页面中单击我们页面的菜单上的产品，完成后进行转跳到要我们所需查看的美食商品的详细信息的页面。在菜单中的商品显示在我们首页的页面上，可以添加下单地址、增加下单地址，修改下单地址、删除下单地址

图 5.7 示出了接收操作的具体设计。

需要查看我们所需要查看我们台数据库所查询的用户信息的下单的地址，必须在登陆的页面进行检查使用的用户是否已经处于我们的购买美食的前端首页。如果我们的使用的用户没有进行登录，就不能在主页上订购下单进行反馈。

在我们的用户登录反馈后，我们的使用的用户就可以在需要点击的地方获取到用户自己想要的食物，进入菜单的食物选择，在食物列表中做出选择。

添加、删除和修改项目中的项目。具体设计如图 5.8 和 5.9 所示。

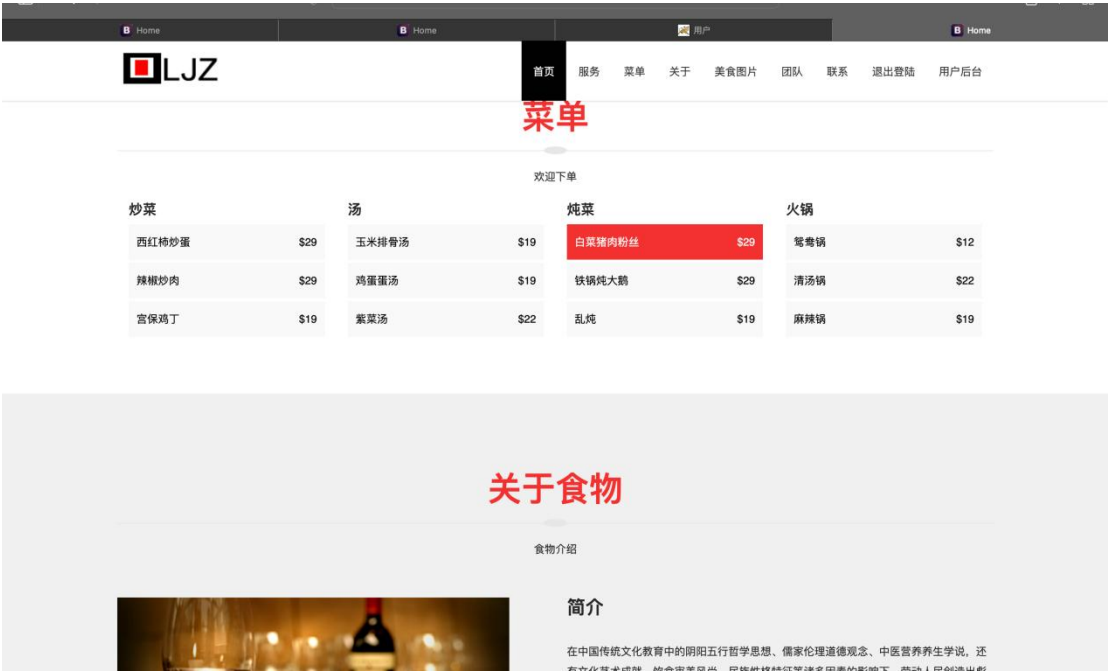


图 5.8 菜单下单页面



购买地址

名字:

密码:

邮箱:

地址:

提交

图 5.9 下单地址页面

(二) 后台模块

1. 管理员登录

管理员是系统权限最高的用户，进入后台必须通过登录校验管理员的合法性，具体设计如图 5.11 所示。

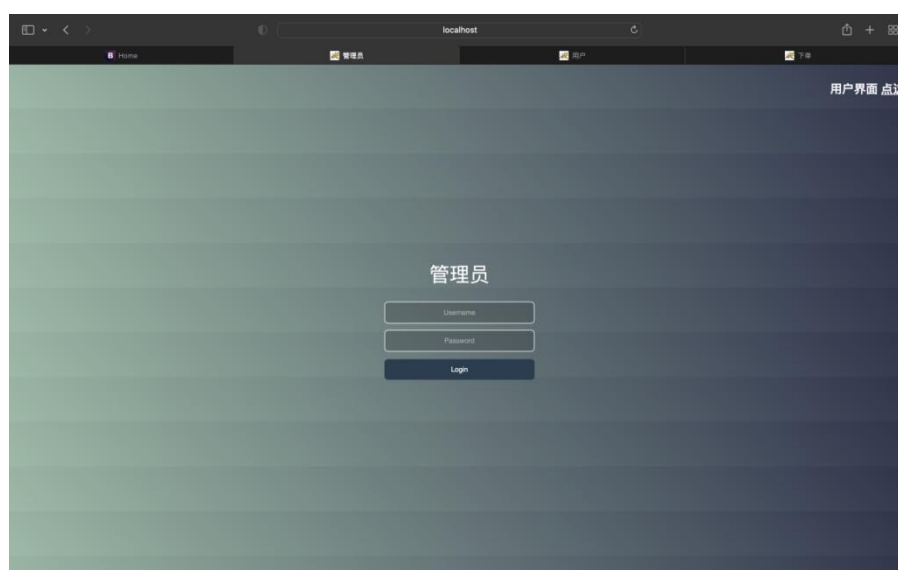


图 5.11 管理员登陆页面

2. 用户管理

我们的用户的账号信息和密码信息都是由我们管理员进行存储的让用户在忘记密码的时候进行反馈给用户的账号信息和密码信息具体设计如图 5.13 所示：

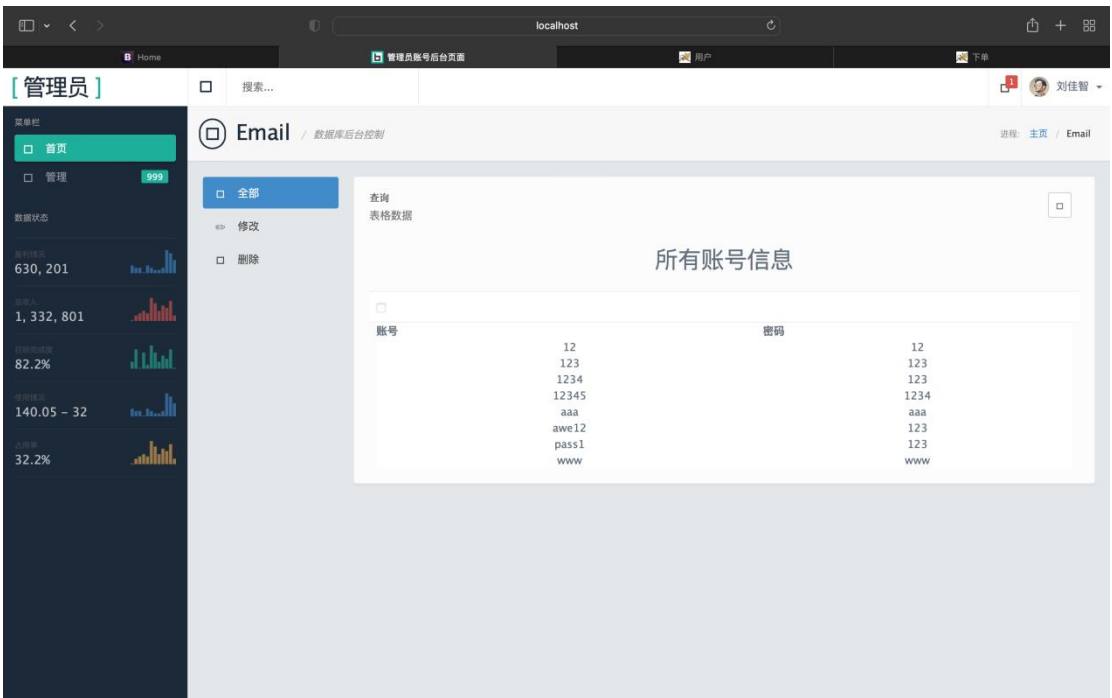
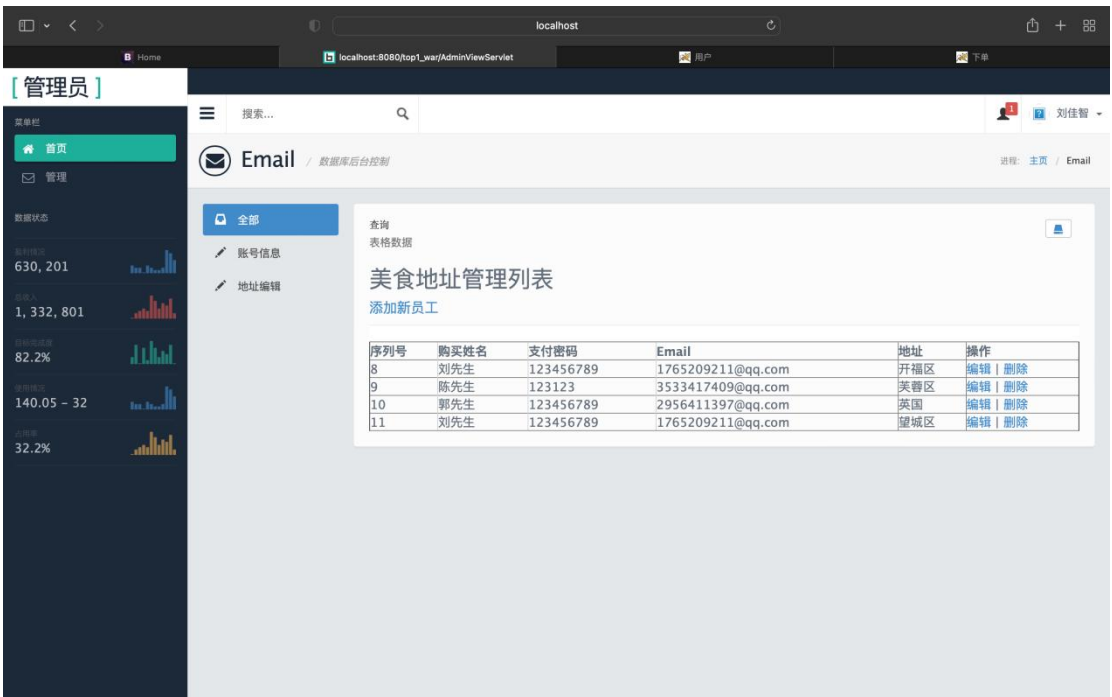


图 5.13 管理员查询用户账号页面

3. 地址管理

我们所设计的用户在管理自己的账号信息时候是包括显示我们的用户帐户信息和密码的信息，用户在操作自己帐户的信息的生活可以根据用户自己帐户进行查询的以备不时的时候需要的。5.13 视图。



六、系统测试

（一）测试方法

美食购物中的系统来进行的测试中想要的目我们还是进行开发不错的项目是不是根据我们所需求分析中的和说明文档里面进行完全的功能进行开发，是不是里面存在安全的隐患，我们的用户是不是体验友好，我们后台中的数据是不是存储合理。我们在这次系统中在测试的前面，需要提前去搭建好系统的环境。

（二）系统测试用例

1.用户登录测试

我们所测试的流程： 我们的使用的用户进入我们的用户登录的界面，输入用户的账号信息和密码信息后，进行对比，选择完我们使用的用户身份，点击我们登陆的按钮进行反馈，如表 6.1 所示。

表 6.1 用户登录用例表

测试用例	预期结果	实际结果	是否通过
用户账号：123 密码：123	成功登录，进入系统 首页	成功登录，进入 程序首页	通过
用户账号：1111 密码：1234	账号错误， 跳转错误	账号信息为空， 跳转错误	通过
用户账号： 123456@qq.com 密码：	输入密码，跳转错误	输入密码，跳转 错误	通过

当我们的使用的用户输入的账号信息不是在后端数据的库中的信息不正确的时候我们的登陆首页会进行反馈，跳转到新的页面中会显示登陆失败，页面停留三秒后转跳，在用户不输入密码信息的时候会显示我们用户需要输入密码信息，我们所测试的内容结果显示如下图 6.1 所示。



图 6.1 用户登录测试

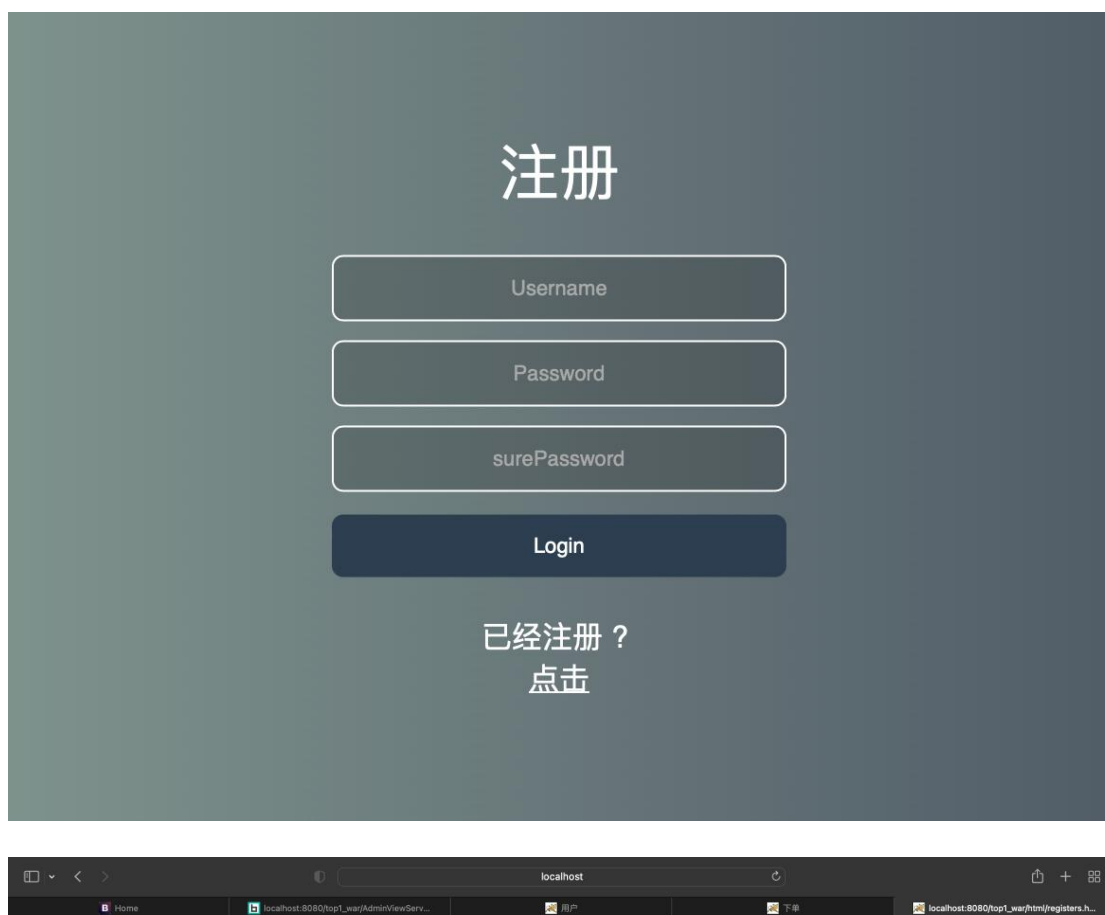
2.用户注册

我们美食管理的系统测试流程：我们使用的用户在进入用户的注册页面中，进入到用户所需要注册的页面，输入密码信息的用户信息，然后再点击注册进行反馈，具体用例表设计如下表 6.2。

表 6.2 用户注册用例表

测试用例	预期结果	实际结果	是否通过
未输入账号信息	账号不能为空	账号不能为空	通过
未输入密码	请输入密码	请输入密码	通过
验证码错误	请输入验证码	请输入验证码	通过

当我们使用的用户中去输入的我们用户所使用的账号并不是正确的邮箱时候，我们这次等登录中的账号中的框就会去显示用户中账号不正确，我们不去输入密码的时候会转跳到注册失败重新注册页面，会去显示框里面的密码不一样的，所以测试的结果中如下显示图 6.2。



注册失败，页面只停留3秒.....

图 6.2 用户注册测试

3.用户管理测试

我们测试的美食的管理系统中测试流程：我们这个管理员会进入用户的所在的管理中列表的页面里面，数据库中的账号我们可以通过管理员的账号来进行这

次普通的用户的账号中的信息，密码的信息所去查询中我们所需要其中的信息。
下面显示的例表设计 6.4，6.5。

表 6.4 用户管理用例表

测试用例	预期结果	实际结果	是否通过
查询用户	显示用户下单地址信息	显示结果	通过
删除用户	点击删除按钮，选择确认，删除对应的用户地址	删除成功	通过
用户修改	点击修改按钮，修改用户地址的信息	修改成功	通过

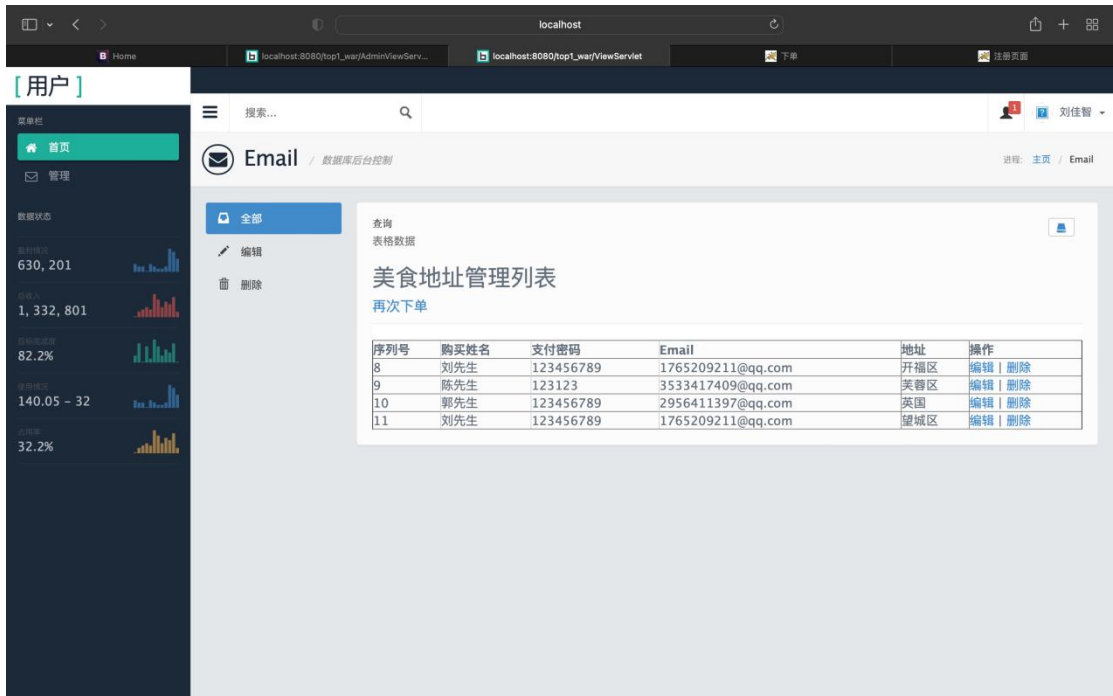


表 6.5 用下单户表

(三) 测试结论

我们这次系统行检查，对于美食项目中的功能进行了测试，首先我们通过设计出美食系统中模块，根据不同的功能来实现出它测试中的用例，我们这次主要通过浏览器中来进行测试来验证是否正常，浏览器的测试中，从用户的体验来看，发现少部分的缺陷，经过几番常识修改代码，再经过回归测试，系统中项目，美食项目的功能上是比较完善的。

我们这次系统的开发后，通过这次实现出了测试的用例，方便了我们测试的人员全方面的进行了的测试，对于系统中功能中的测试具体中描述测试的过程和功能，这次是通过浏览器中的检测，来实现验收我们的系统中功能是不是完整的，后面结尾中的总结测试，在后面结尾中的总结测试中测试中我们所遇到的里面问题，去解决问题，然后理解问题，我们最后完成了我们这次最终的测试。

总结

从确定写这本书的食物主题来看，写这篇的文本并不容易。不像我想的那样，因为这个文档与项目和程序直接相关，包括项目，决定编写哪种编程语言以及用于建立美食管理结构的技术。配置后台服务器。在我确定了这情况，将美食的管理的系统。我选择 Java 语言作为我全部使用的技术支持和我们所使用的应用程序来使用作为美食的管理系统的后台支持大众。首先，我们线进行安装 Win 系统的环境的本地 Java 的 JAR 包进行系统配置，数据库的系统环境配置和使用我们常用的软件环境配置，我也很多问题。我必须在互联网上找到每个人，解决他们，直到项目完成。在这个项目的过程中，反向连接系统的技术不是很好，它可以很快完成，但在开发过程中有放弃的想法。所以，尤其是在后方。我在观看时也了解到了这一点，因为互联网上仍然有开放资源，最终完成了后端数据库和 JAV 代码中的 JDBC 辛迪加。通过了这次所操作的这个美食管理的项目中，我对与这次所需要开发的美食的管理系统程序、所使用的框架和一系列的操作数据的库有了很新的理解和思路。在设计过程中，E-R 示意图是在需求阶段和详细过程中绘制的。另一方面，这一设计也是我多年来读到的一个法案，是美妙过程的一部分。我觉得最重要的是项目的实验，因为在完成项目后，下一个优先事项是执行实验项目，分析用于执行每个章节的详细描述和文本的密码，根据软件技术的流程编写，绘制大量流程图，并使用案例示意图进行观察，以支持项目的最重要部分。尤其是它前后的开发机制，被包裹在标题中并称为标题记录，所有这些都不清楚。我们办法停下我们的步伐随着我们的社会上的实践发展，虽然毕业生即将到来，但我对这个项目非常满意。我们这一代人的使命是推动我学习。这个过程很困难，包括我上学时学到的知识，以及我在外面学到的新技术。言到这里，我很感谢我们现在有机会的现在照顾了我的所有老师和帮助我所需要解决的解决问题的同学们。

参考资料

- [1] 座山茂密。在使用美食中管理的系统设计与实现。JAVA 的编程语言在使用计算机中 IDEA 中的软件开发 的应用。16-115;
- [3] 坎沙库。应用和 JSP 的设计的分析解析的书。2021: 12-22;
- [4]李艳杰。使用与 java 的技术和 MySQL 数据库移动应用系统[J]黑龙江科学, 2022, 13 (02) 31-35:
- [5]哈塔的起源。适用于 html5 中技术的互联网中接口的设计与开发。195-199
- [6] 日野久弥坚、日野森仓和张江。在浏览的情况下的 B 和 S 中的模式, 很标准化, 正确化的管理中信息的美食的管理系统的设计与实现。168-156
- [7]JUnit 框架 Java 单元测试[J]电子技术与工程的实现

附录后台接口源码

```
public static int login(Login login) {
    int flag = -1;
    String URL="jdbc:mysql://localhost:3306/javadata?userSSL=true&userUnicode=true&characterEncoding=utf-8";
    String NAME = "root";
    String PASSWORD = "liu220500";

    Connection connection = null;
    PreparedStatement pstmt = null;
    ResultSet rs = null;

    try {
        // 1、加载驱动
        Class.forName( className: "com.mysql.cj.jdbc.Driver");
        // 2、建立连接
        connection = DriverManager.getConnection(URL, NAME, PASSWORD);

        // 3、操作数据库
        String sql = "select * from user where name=? and password=?";

        pstmt = connection.prepareStatement(sql);
        pstmt.setString( parameterIndex: 1, login.getName());
        pstmt.setString( parameterIndex: 2, login.getPassword());
    }
}
```

附录图 5.1 登录链接数据库查询表格图

```
1 usage
public static int register(Login login) {
    int count = -1;
    String URL="jdbc:mysql://localhost:3306/javadata?userSSL=true&userUnicode=true&characterEncoding=utf-8";
    String NAME = "root";
    String PASSWORD = "liu220500";
    //String URL = "jdbc:sqlserver://localhost:1433;databaseName=test";

    Connection connection=null;
    PreparedStatement pstmt=null;
    ResultSet rs=null;

    try {
        //1、加载驱动
        Class.forName( className: "com.mysql.cj.jdbc.Driver");
        //2、连接
        connection=DriverManager.getConnection(URL, NAME, PASSWORD);
        //操作数据库
        String sql="insert into user values(?,?)"; //插入数据
        pstmt=connection.prepareStatement(sql);
        pstmt.setString( parameterIndex: 1, login.getName());
        pstmt.setString( parameterIndex: 2, login.getPassword());
    }
}
```

附录图 5.2 用户注册链接数据库插入表格图

```

public class EmployeeDao {
    public static Connection getConnection() { return ConnectDb.getConnection(); }
    1 usage
    public static int save(Employee e) {
        int status = 0;
        try {
            Connection con = EmployeeDao.getConnection();
            PreparedStatement ps = con
                .prepareStatement( sql: "insert into xn_employee(name,password,email,country)values(?,?,?,?)");
            ps.setString( parameterIndex: 1, e.getName());
            ps.setString( parameterIndex: 2, e.getPassword());
            ps.setString( parameterIndex: 3, e.getEmail());
            ps.setString( parameterIndex: 4, e.getCountry());
            status = ps.executeUpdate();
            con.close();
        } catch (Exception ex) {
            ex.printStackTrace();
        }
        return status;
    }
}

```

附录图 5.3 用户下单链接数据库插入表格图

```

1 usage
public static int delete(int id) {
    int status = 0;
    try {
        Connection con = EmployeeDao.getConnection();
        PreparedStatement ps = con.prepareStatement( sql: "delete from xn_employee where id=?");
        ps.setInt( parameterIndex: 1, id);
        status = ps.executeUpdate();
        con.close();
    } catch (Exception e) {
        e.printStackTrace();
    }
    return status;
}

```

```

public static Employee getEmployeeById(int id) {
    Employee e = new Employee();
    try {
        Connection con = EmployeeDao.getConnection();
        PreparedStatement ps = con.prepareStatement( sql: "select * from xn_employee where id=?");
        ps.setInt( parameterIndex: 1, id);
        ResultSet rs = ps.executeQuery();
        if (rs.next()) {
            e.setId(rs.getInt( columnIndex: 1));
            e.setName(rs.getString( columnIndex: 2));
            e.setPassword(rs.getString( columnIndex: 3));
            e.setEmail(rs.getString( columnIndex: 4));
            e.setCountry(rs.getString( columnIndex: 5));
        }
        con.close();
    } catch (Exception ex) {
        ex.printStackTrace();
    }
}

```

附录图 5.4 用户下单地址增删改查数据库语句