长 沙 学 院

软件工程基础实训Ⅱ 实训

|  |  |
| --- | --- |
| **题目** | **基于C/S架构的餐饮管理系统** |
| **学院** | **计算机科学与工程** |
| **专业(班级)** | **软件工程（21软件01）** |
| **姓名** | **赵毅军** |
| **学号** | **B20210304108** |
| **指导教师** | **潘怡、杨刚** |
| **起止日期** | **2023.06.05～2023.06.16** |

实训任务书

**课程名称：软件工程基础实训II**

**实训题目：基于C/S架构的餐饮管理系统**

### 已知技术参数和设计要求：

1. 问题描述（功能要求）：

本次案例将完成一个基于C/S架构的餐饮管理系统。

（1）.进行餐饮管理系统的功能设计。

（2）.设计餐饮管理系统的数据库。

（3）.利用JAVA的GUI类设计开发友好的人机接口。

（4）.利用JAVA语言设计监听用户操作和JDBC进行数据库的处理。

餐饮管理系统主要有6大模块：

（1）.员工管理：对员工实现增删改查。

（2）.客户管理：对客户实现增删改查。

（3）.餐台管理：对餐台实现增删改查。

（4）.菜品管理：对菜品分类、菜品实现增删改查。

（5）.点菜管理：服务员对某客户、某一空闲餐台实行开台，同时实现点菜，将餐台号与所点的菜品对应起来，分别显示出来，并记录开台时间。

（6）.结账管理：收银员对某一餐台通过统计显示消费的菜品清单统计出消费金额，通过手动输入实收金额进行找零的计算，并显示，完成结账的操作，并记录成统计数据。

基础实训II有2周时间40学时，学生基本必做任务：独立完成基于C/S架构的餐饮管理系统。

选做扩展：菜品推荐、销售统计。

**2. 运行环境要求：**

（1）客户端：

windows操作系统，JDK1.6，Idea或Eclipse。

（2）服务器：

MySQL5.6或 SQL Server 数据库服务器。

3. 技术要求：

1） 掌握软件工程的需求分析和系统设计方法。

2) 掌握数据库的分析与设计，完成餐饮管理系统数据库的设计。

3) 掌握JAVA的GUI设计方法，完成登录、主界面、菜品分类管理界面、菜品管理界面等人机接口。

4） 掌握JAVA面向对象程序设计方法，设计系统所需的各种类。

5) 掌握JDBC方法，对数据库进行操作，完成主程序和菜品分类、菜品管理等功能。

### 实训工作量：

40课时

### 工作计划：

1.班级

21软件1-4

2.课时及教室安排

见实训课程工作计划表。

实训报告

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学生姓名 | 赵毅军 | 学号 | B20210304108 | 班级 | 21软件工程01 |
| 专业 | 软件工程 | | 指导教师姓名 | 杨刚、潘怡 | |
| 项目名称 | 基于C/S架构的餐饮管理系统 | | | | |
| **实训过程**：（说明实训各阶段的工作过程，包括需求分析、系统设计、编码实现、系统测试、交付实施等阶段的主要工作。）  需求分析阶段：  （1）.明确餐饮管理系统的业务目标，就是完成点菜功能以及各表的增删改查  （2）.编写需求说明书，包括文档编写、用例分析、流程图等；  （4）.进行需求评审，修正需求说明书。  系统设计阶段：  （1）.进行系统架构设计，确定系统使用的技术栈和平台；  （2）.进行数据库设计，定义数据结构和关系，并进行数据流程图的绘制；  （3）.进行界面设计，规划系统界面的结构和交互逻辑；  （4）.编写详细设计文档，总结规划结果。  编码实现阶段：  （1）.使用所选定的技术栈和设计文档实现功能；  （2）.对核心代码进行测试，并进行缺陷修复；  （3）.编写程序说明文档。  系统测试阶段：  （1）.进行单元测试，对于每个模块进行测试；  （2）.进行系统测试，测试整个系统的功能和稳定性；  （3）.进行压力测试，模拟多用户请求进行测试；  （4）.测试报告编写，总结测试结果。  交付实施阶段：  （1）.部署和安装餐饮管理系统；  （2）.进行使用说明，向所有用户传达操作方法；  （3）.进行维护和支持，对于发现的问题进行及时处理； | | | | | |
| **实训成果**：  参见基于C/S架构的餐饮管理系统设计说明书。 | | | | | |
| **实训总结**：（包括心得体会、存在的问题和改进方向。）  一、心得体会：  1．学习了mybatis框架。对xml配置文件有了一个熟练的了解。  2.学习了java swing 的各种类型的使用  3.对于debug一步一步调试有了比较大的提升。  4.工程能力得到了较大提升。  二、存在的问题  1．增删改查界面还未设置完成。  2. 数据库持久性不足。  3. 业务逻辑还存在部分缺陷。  4.前端界面不够美观。  三、改进方向  1. 尝试使用一些web技术来构建前端可以尝试vue之类的技术。  2. 完善好各类业务逻辑。  3.进一步学习mybatis和mybatis-plus再加上springboot构建一个完整的后端。 | | | | | |

基于C/S架构的餐饮管理系统

设计说明书

作者：赵毅军 (B20210304108)

计算机科学与工程学院

2023年 06 月 15日

**摘要**

摘要应概括反映出课程设计的内容、方法、成果和结论。结果和结论性字句是摘要的重点，在文字论述上要多些，以加深读者的印象；用精炼、概括的语言来表达，每项内容不宜展开论证或说明，要客观陈述，不宜加主观评价；要独立成文，选词用语要避免与全文尤其是前言和结论部分雷同；摘要中不宜使用公式、图表，不标注引用文献编号。避免将摘要写成目录式的内容介绍。

关键词是供检索用的主题词条，应采用能覆盖全文主要内容的通用技术词条(参照教材附录A的名词索引或其它技术术语标准)。关键词一般列3~5个，按词条的外延层次排列（外延大的在前面）。

本次实训是基于Java语言和C/S架构的餐饮管理系统，系统中使用到的数据库为MySql。该系统旨在帮助学生深入了解软件开发的全流程，通过实践掌握相关技能。系统功能包括点单、后厨制作、服务、结算等，可以大大提高餐厅管理的效率。

在实训过程中，学生需要自行进行需求分析、系统设计、编码实现、系统测试和交付实施等阶段的工作。在需求分析阶段中，需要明确系统需求，明确系统功能和特性；在系统设计阶段中，应该对系统架构、数据库和界面进行规划；在编码实现阶段中，将会使用Java语言和MySql数据库完成系统开发；在系统测试阶段中，将对系统进行功能、稳定性和压力测试；在交付实施阶段中，将完成部署安装、操作文档编写和用户培训等工作。

本次实训所用的技术栈为Java语言、C/S架构、MySql数据库，并以餐饮管理系统为例进行设计和开发。通过实践，学生可以获得完整的软件开发周期经验，同时熟悉我国餐饮行业的工作方式和管理方式。在这个过程中，学生还能够提高沟通能力，从而更好地适应未来的职场。

**关键词：**java ，C/S架构，餐饮管理系统，MySql, mybatis.

目录

[1 引言 1](#_Toc11882504)

[1.1 编写目的 1](#_Toc11882505)

[1.2 参考资料 1](#_Toc11882506)

[2 需求规约 2](#_Toc11882507)

[2.1 功能需求 2](#_Toc11882508)

[2.2 界面需求 6](#_Toc11882509)

[2.3 数据需求 7](#_Toc11882510)

[3 系统设计 10](#_Toc11882511)

[3.1 运行环境 10](#_Toc11882512)

[3.2 系统静态结构设计 10](#_Toc11882513)

[3.3 人机接口设计 11](#_Toc11882514)

[3.4-3.11各类界面操作 19](#_Toc11882515)

[4 数据库设计 32](#_Toc11882517)

[4.1 数据库环境说明 32](#_Toc11882518)

[4.2 数据库的命名规则 32](#_Toc11882519)

[4.3 逻辑结构设计 34](#_Toc11882520)

[4.4 物理结构设计 44](#_Toc11882521)

[4.5 安全性设计 44](#_Toc11882522)

[5 测试用例设计 44](#_Toc11882523)

[附录 48](#_Toc11882524)

[附录1 程序运行结果 48](#_Toc11882525)

[附录2 程序源代码 15](#_Toc11882526)

# 1 引言

## 1.1 编写目的

本文档是“基于C/S架构的餐饮管理系统”的软件设计说明书。编写目的是：明确软件需求、概要设计、详细设计和功能测试用例，为软件后续开发和维护提供指导。本文档的读者对象是需求分析人员、系统设计人员、系统开发人员、系统测试人员。

## 1.2 参考资料

列出有关资料的名称、作者、文件编号或版本等。

[1] 文东. 数据库系统开发[M]. 北京: 北京科海电子出版社，2011．

[2] 戴远泉，程宁，胡文杰. Java高级程序设计[M]. 北京：人民邮电出版社，2022.

# 2 需求规约

## 2.1 功能需求

提示：详细描述程序的功能需求，分析功能需求，写出每个功能的说明。画出功能模块图，如图2.1所示。

（1）.员工管理：对员工实现增删改查。

（2）.客户管理：对客户实现增删改查。

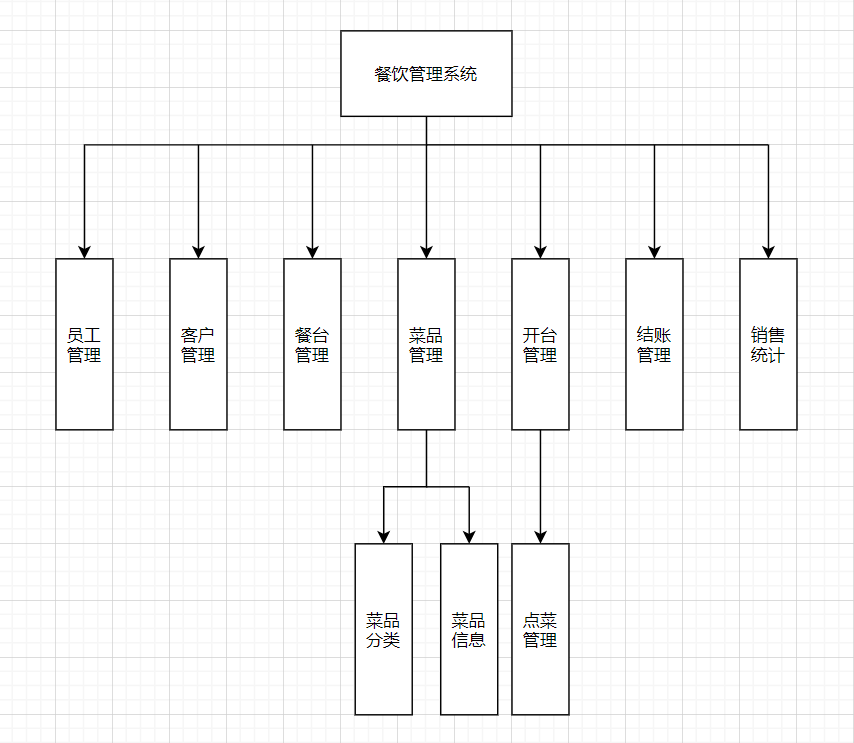
（3）.餐台管理：对餐台实现增删改查。

（4）.菜品管理：对菜品分类、菜品实现增删改查。

（5）.点菜管理：服务员对某客户、某一空闲餐台实行开台，同时实现点菜，将餐台号与所点的菜品对应起来，分别显示出来，并记录开台时间。

（6）.结账管理：收银员对某一餐台通过统计显示消费的菜品清单统计出消费金额，通过手动输入实收金额进行找零的计算，并显示，完成结账的操作，并记录成统计数据。

（7）.销售统计：对某一段时间各菜品进行销售统计，记录销售数量及总金额



（**图名： 仿宋，小四。**

**图内文字：宋体，五号。）**

图2.1 餐饮管理系统功能模块图

2.1.1 员工管理用例说明

用例名称：员工管理。

用例描述：管理员对员工实现增删改查操作。

前置条件：管理员账号和密码必须正确。

后置条件：操作成功。

活动步骤：

1.管理员注册账号。

2.管理员登录。

3.选择菜单中的员工管理。

4.进行增删改查操作。

5.操作成功。

异常处理：

1.如果管理员账号密码输入错误，提示登录失败。

2.如果添加的员工工号已存在，则提示该员工已存在。

2.1.2 客户管理用例说明

用例名称：客户管理。

用例描述：管理员对客户实现增删改查操作。

前置条件：管理员账号和密码必须正确。

后置条件：操作成功。

活动步骤：

1.管理员注册账号。

2.管理员登录。

3.选择菜单中的客户管理。

4.进行增删改查操作。

5.操作成功。 1.

异常处理：

1.如果管理员账号密码输入错误，提示登录失败。

2.如果添加的客户id已存在，则提示该客户已存在。

2.1.3 餐台管理用例说明

用例名称：餐台管理。

用例描述：管理员对餐台实现增删改查操作。

前置条件：管理员账号和密码必须正确。

后置条件：操作成功。

活动步骤：

1.管理员注册账号。

2.管理员登录。

3.选择菜单中的餐台管理。

4.进行增删改查操作。

5.操作成功。

异常处理：

1.如果管理员账号密码输入错误，提示登录失败。

2.如果添加的餐台序号已存在，则提示该餐台已存在。

2.1.4 开台管理用例说明

用例名称：开台管理。

用例描述：管理员对客户开台，并进行点菜。

前置条件：管理员账号和密码必须正确，客户必须是在客户表中的客户，所用的餐台必须是处于空闲状态。

后置条件：操作成功。

活动步骤：

1.管理员注册账号。

2.管理员登录。

3.选择菜单中的开台管理。

4.进行开台操作。

5.进行点菜操作。

6.点菜成功。

扩展点：

1.

异常处理：

1.如果管理员账号密码输入错误，提示登录失败。

2.如果添加的流水号已存在，则提示该订单已存在。

2.1.5 菜品管理用例说明

用例名称：菜品管理。

用例描述：管理员对菜品和菜品分类管理。

前置条件：管理员账号和密码必须正确，菜品所属的菜品类必须是在菜品类别表中的菜品类。

后置条件：操作成功。

活动步骤：

1.管理员注册账号。

2.管理员登录。

3.选择菜单中的菜品管理。

4.选择菜品类别管理或者菜品管理。

5.对菜品类别或者菜品进行增删改查操作。

6.操作成功。

扩展点：

1.

异常处理：

1.如果管理员账号密码输入错误，提示登录失败。

2.如果添加的流水号已存在，则提示该订单已存在。

2.1.6 结账管理用例说明

用例名称：结账管理。

用例描述：管理员结账管理。

前置条件：管理员账号和密码必须正确，输入已存在的订单序号和实收金额和优惠。

后置条件：结账成功。

活动步骤：

1.管理员注册账号。

2.管理员登录。

3.选择菜单中结账管理。

4.对未结账的订单进行结账操作。

5.操作成功。

扩展点：

1.

异常处理：

1.如果管理员账号密码输入错误，提示登录失败。

2.如果填写的订单号不存在，则无法结账。

2.1.7 销售统计用例说明

用例名称：销售统计。

用例描述：管理员查看销售统计。

前置条件：管理员账号和密码必须正确，输入开始日期和截至日期。

后置条件：出现销售统计表。

活动步骤：

1.管理员注册账号。

2.管理员登录。

3.选择菜单中销售统计。

4.输入开始日期和截止日期。

5.出现销售统计表。

扩展点：

1.

异常处理：

1.如果管理员账号密码输入错误，提示登录失败。

## 2.2 界面需求

在餐饮管理系统中，用户对交互界面的要求通常包括以下几个方面：

易用性：用户希望界面操作简单、直观易懂，能够快速完成任务，并且不需要太长时间的学习成本。

可视化效果：用户希望界面能够提供清晰、美观的图形展示，以便更好的了解数据。

响应速度：用户希望界面能够快速响应他们的操作请求，不会出现卡顿、延迟等情况，提高工作效率。

界面定制化：用户希望有一定的自定义界面的权利，如可更改界面主题、字体设置等。

数据采集与汇总：界面是餐饮管理系统的窗口，用户希望界面能够方便快捷地进行数据采集和数据汇总、查询等操作。

数据分析统计：餐饮管理系统中包括了大量的数据信息，用户希望通过界面能够方便地进行数据分析和统计操作，以便更好地了解经营状况。

安全性：作为一个运行在网络上的管理系统，用户希望界面具有安全保障，确保数据不受到恶意攻击或外泄的风险。

总之，餐饮管理系统中的界面应该既能够满足用户的需求，又能够为其提供全面的数据支持，以便更好地进行运营管理。

## 2.3 数据需求

表2.1 用户实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| username | varchar | 20 | 否 | 是 | 用户名 |
| password | varchar | 20 | 否 | 否 | 密码 |

表2.2员工实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 客户号 |
| name | varchar | 20 | 是 | 否 | 姓名 |
| sex | varchar | 2 | 是 | 否 | 性别 |
| birthday | datetime | 8 | 是 | 否 | 出生日期 |
| identityId | varchar | 18 | 是 | 否 | 身份证号 |
| address | varchar | 40 | 是 | 否 | 地址 |
| tel | varchar | 11 | 是 | 否 | 电话 |
| position | varchar | 4 | 否 | 否 | 职位 |
| freeze | varchar | 4 | 否 | 否 | 是否在职 |

表2.3客户实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| name | varchar | 20 | 否 | 否 | 姓名 |
| sex | varchar | 2 | 是 | 否 | 性别 |
| company | varchar | 20 | 是 | 否 | 单位 |
| tel | varchar | 11 | 是 | 否 | 电话 |
| cardId | varchar | 10 | 否 | 是 | 贵宾卡号 |

表2.4菜品分类实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| name | varchar | 20 | 否 | 否 | 名称 |
| describ | varchar | 20 | 是 | 否 | 描述 |

表2.5菜品实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| name | varchar | 20 | 否 | 否 | 菜名 |
| categoryId | int | 10 | 否 | 否 | 类别编号 |
| price | double | 6 | 否 | 否 | 价格 |
| status | varchar | 4 | 否 | 否 | 状态 |

表2.6餐台实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| no | varchar | 8 | 否 | 是 | 餐台编号 |
| saeting | int | 4 | 否 | 否 | 座位数 |
| status | varchar | 10 | 否 | 否 | 状态 |

表2.7订单实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| orderNo | varchar | 20 | 否 | 是 | 订单编号 |
| deskId | int | 10 | 否 | 否 | 餐台号 |
| createtime | datetime | 40 | 否 | 否 | 就餐时间 |
| money | double | 6 | 否 | 否 | 金额 |
| customerId | int | 10 | 否 | 否 | 客户编号 |
| status | varchar | 4 | 否 | 否 | 状态 |
| number | int | 4 | 否 | 否 | 就餐人数 |

表2.8订单明细实体

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| orderId | varchar | 20 | 否 | 是 | 订单编号 |
| dishId | varchar | 20 | 否 | 否 | 菜品编号 |
| amount | int | 4 | 否 | 否 | 数量 |

# 3 系统设计

这里陈述总体设计思路。说明系统需求中的每个功能实现思路。

总体设计思路就是就是一般系统的操作对所有表实现增删改查。然后就是用mybatis来代替了JDBC总体上，然后根据点菜的具体业务逻辑实现了一些接口方式。点菜逻辑大概就是先是客户点单，然后实现开台确定桌号和订单好，然后再在orderitem这个order和desk对应表中添加对应的内容。然后最后再实现一个结账功能就可以了。

## 3.1 运行环境

简要说明对本系统的运行环境（包括硬件环境和支持软件环境）的规定。

操作系统：Windows 11

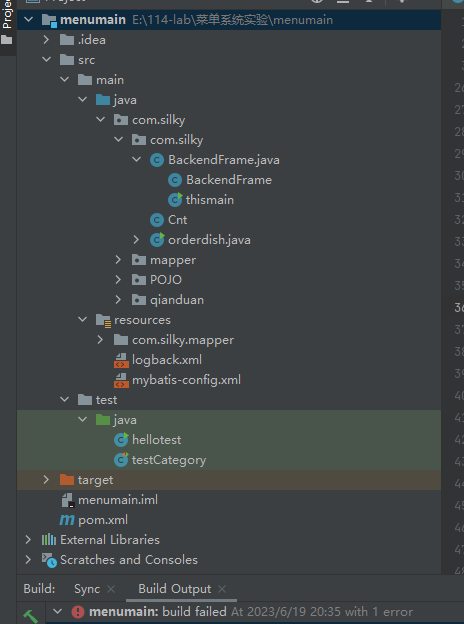
数据库：MySQL 8.0.33

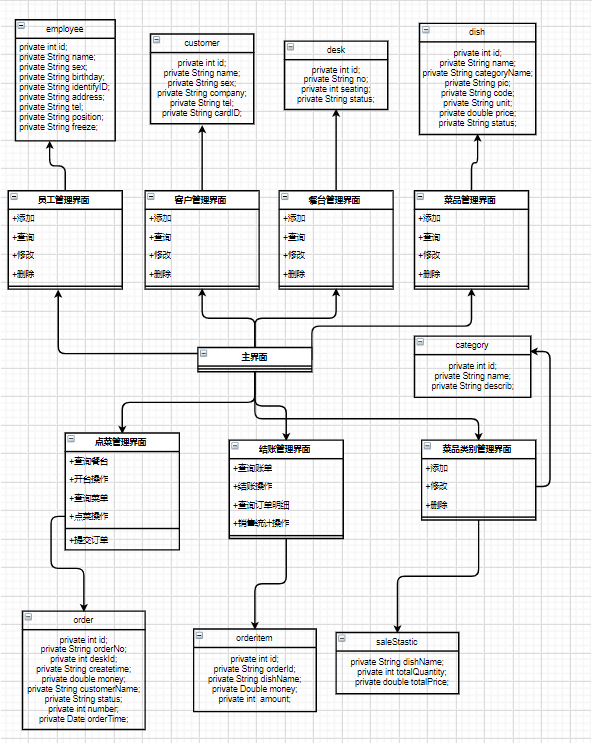
java：Java 10.0.2

## 3.2 系统静态结构设计

采用类图方式给出系统静态结构设计，说明本系统的各软件类的划分。类图要求详细描述，必须有属性和操作，不能仅仅是类名。如果一张图画不下，可以分模块叙述，也可以先画类的简图，在简图之后逐个画出每个类的类图。

例如，图2.1的航空售票系统中，要实现预售票，则系统的软件静态结构如下：





## 3.3 人机接口设计

3.3.1 设计应用类

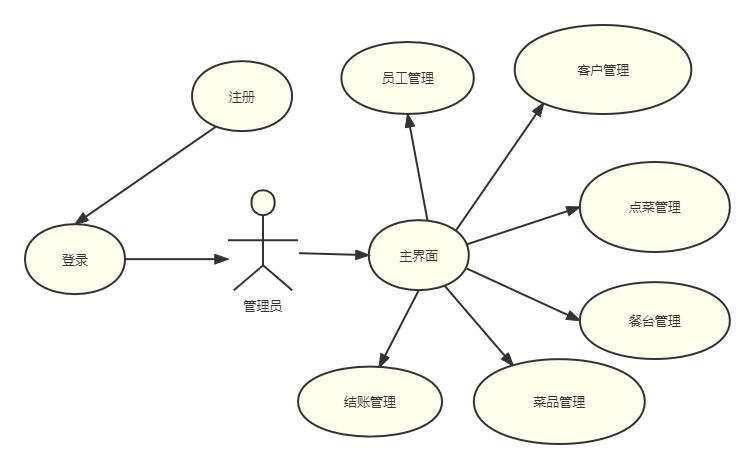


图3.2餐饮管理系统设计类图

3.3.2 用户接口1（具体标识符）设计

说明本接口的界面设计，给出界面布局。例如：图3.1

3.3.3 用户接口2（具体标识符）设计

**界面主要控件：**

JMenu，菜单栏按钮，触发子菜单。

JMenuItem，子菜单按钮，触发按钮事件

**实现功能：**

1. 点击员工管理，下拉子菜单，点击查询跳转查询界面，点击管理跳转管理界面。
2. 点击客户管理，下拉子菜单，点击查询跳转查询界面，点击管理跳转管理界面。
3. 点击餐台管理，下拉子菜单，点击查询跳转查询界面，点击管理跳转管理界面。
4. 点击菜品管理，下拉子菜单，点击查询跳转查询界面，点击管理跳转管理界面，点击菜品类别管理进入菜品类别管理界面。
5. 点击点菜管理，下拉子菜单，点击开台跳转开台界面，点击点菜进入点菜界面
6. 点击结账管理，下拉子菜单 ，点击结账跳转结账界面，点击销售统计进入销售统计界面。
7. 点击帮助，下拉子菜单，点击切换账号进入登录界面，点击退出，退出餐饮管理系统。

3.3.5 用户接口4（员工管理界面）设计

**查询界面主要控件：**

JButton，按钮，触发点击事件。

JTextField，输入框，接收员工id、员工姓名、住址、职位等。

**实现功能：**

1. 输入要查询的信息
2. 点击查询按钮，将对员工进行查询。

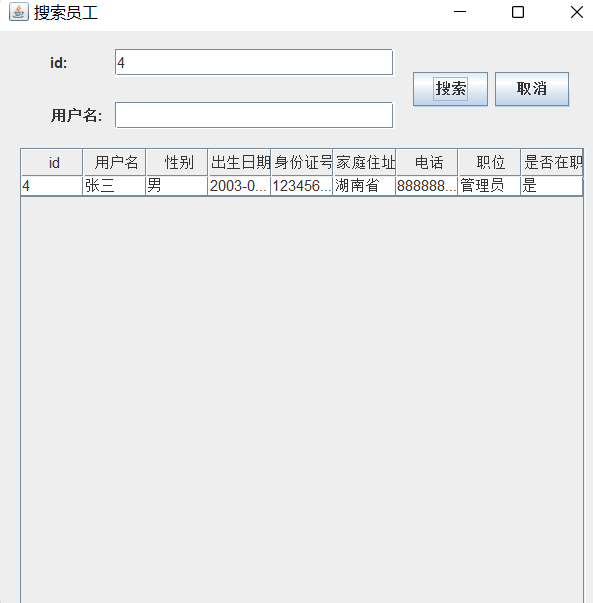


图3.6员工查询界面

**管理界面主要控件：**

JButton，按钮，触发点击事件。

JTextField，输入框，接收员工信息。

**实现功能：**

1. 在输入框输入要添加的员工信息，点击添加按钮，将对员工进行增添。
2. 在输入框输入要修改的员工信息，点击修改按钮，将对员工进行修改。
3. 在输入框输入要删除的员工序号，点击删除按钮，将对员工进行删除。

客户管理、餐台管理、菜品管理、菜品类别管理与员工管理的操作界面一样。



图3.7员工管理界面

3.3.9 用户接口8（点菜管理界面）设计

**开台管理界面主要控件：**

界面1：

JButton，按钮，触发点击事件。

JTextField，输入框，接收餐台查询信息。

界面2：

JButton，按钮，触发点击事件。

JTextField，输入框，确定餐台人数

JComboBox，下拉列表，对客户名和餐台号信息进行筛选。

**实现功能：**

界面1：

1. 主界面即可查看是否有空闲餐台。
2. 输入餐台序号、餐台号、就餐人数。
3. 点击开始开台跳转至界面2

界面2：

1. 选择点菜
2. 进入点菜界面



图3.10开台管理界面（1）



图3.10点菜管理界面

**点菜管理界面主要控件：**

界面1：

JButton，按钮，触发点击事件。

JTextField，输入框，接收菜品查询信息。

界面2：

JButton，按钮，触发点击事件。

JTable，表格，确定菜品数量

JComboBox，下拉列表，对订单编号信息进行筛选。

**实现功能：**

界面1：

1. 输入菜品信息进行查询是否菜品有货，
2. 点击开始点菜跳转至界面2

界面2：

1. 选择订单编号，确定菜品数量
2. 点击添加，下单成功
3. 点击购物车可查询已添加的菜品

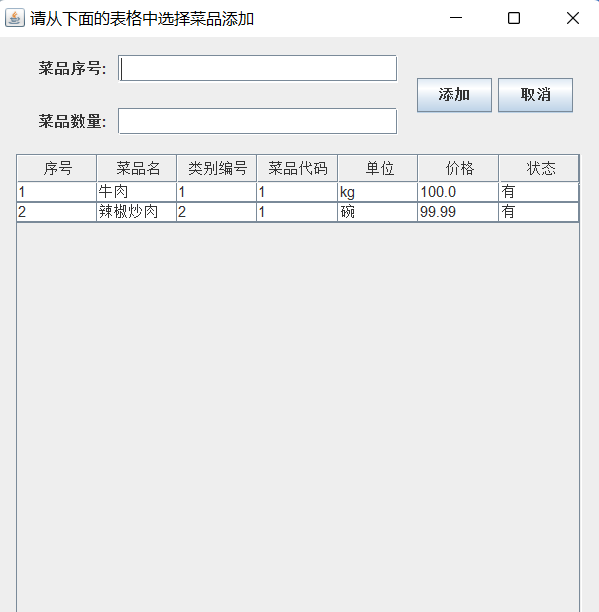


图3.10点菜管理界面

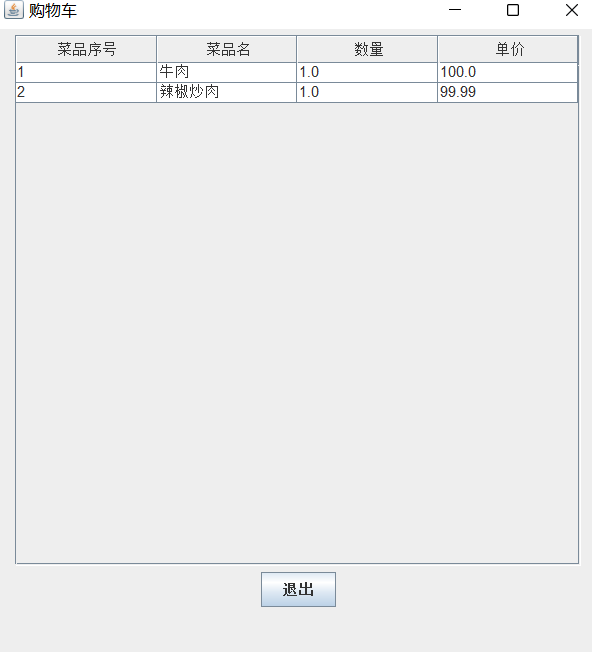


图3.10购物车界面

3.3.10 用户接口9（结账管理界面）设计

**结账界面主要控件：**

界面1：

JButton，按钮，触发点击事件。

JTextField，输入框，输入订单信息。

界面2：

JTextField，输入框，输入金额。

JComboBox，下拉列表，对订单编号信息进行筛选。

界面3

JComboBox，下拉列表，对订单编号信息进行筛选。

Jtable,表格，显示点菜信息。

**实现功能：**

界面1：

1. 输入订单编号进行查询订单金额
2. 点击确定显示应收金额
3. 点击结账出现结账成功



## 图3.11结账管理界面

3.3.11 用户接口10（销售统计界面）设计

**结账界面主要控件：**

界面1：

JButton，按钮，触发点击事件。

JTextField，输入框，输入订单信息。

界面2：

JTextField，输入框，输入金额。

JComboBox，下拉列表，对订单编号信息进行筛选。

界面3

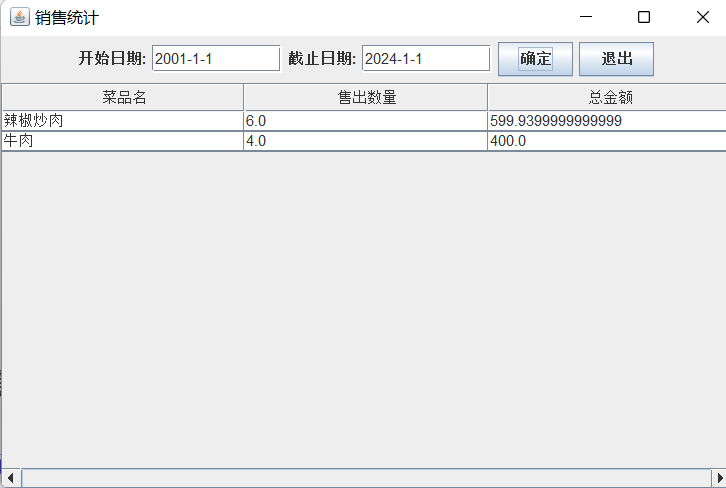
JComboBox，下拉列表，对订单编号信息进行筛选。

Jtable,表格，显示点菜信息。

**实现功能：**

界面1：

1. 输入开始时间和截至时间
2. 点击确定出现销售统计表。



## 图3.12销售统计界面

## 3.4 注册模块设计说明

（1）功能：该模块是餐饮系统对管理员进行登录注册操作。

（2）测试要点：

1. 输入的账号或密码为空将提示账号或密码为空。

2. 输入的账号密码错误则提示账号密码有误。

3. 输入账号不存在，进行注册

3.5 登录模块设计说明

（1）功能：该模块主要是管理员对餐饮管理系统进行登录操作。

（2）测试要点：

1. 输入的账号或密码为空将提示账号或密码为空。

2. 输入的账号密码错误则提示账号密码有误。

## 3.6 员工管理模块设计说明

（1）功能：实现管理员对员工的增删改查。

1. 新增员工，程序流程图如图3.14所示：

测试要点：

1. 点击新增员工按钮，弹出新增员工窗口。
2. 输入员工的信息，若员工信息无误且完整，则提示新增成功。
3. 若信息不完整，且信息有误，系统会给出提醒。

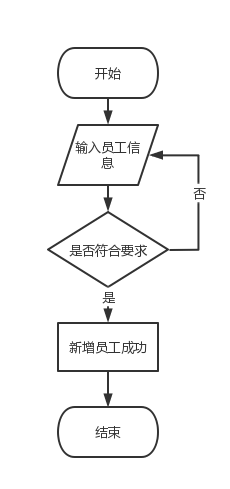


图3.14新增员工信息流程图

1. 删除员工，程序流程图如图3.15所示：



图3.15删除员工流程图

测试要点：

1. 点击表格中员工的信息，点击删除，系统提示删除成功。
2. 修改员工，程序流程图如图3.16所示：

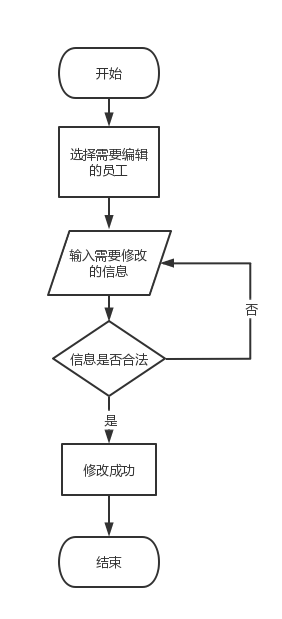


图3.16修改员工信息流程图

测试要点：

1. 点击表格中员工的信息，弹出新的界面。
2. 在新的界面中填写要修改的信息，若输入的信息合法则提示修改成功
3. 若输入的信息不合法，则提示信息不合法。
4. 查询员工，程序流程图如图3.17所示：

测试要点：

1. 在文本框中输入员工的姓名和id，点击查询按钮。
2. 查询信息将在表格中显示。

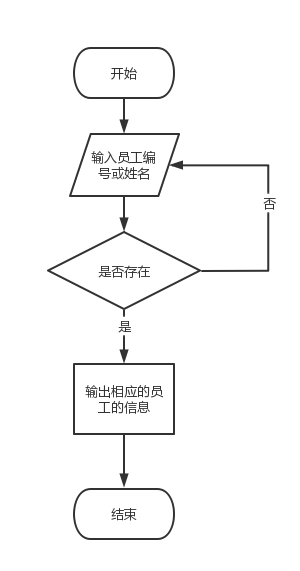


图3.17查询员工信息流程图

## 3.7 客户管理模块设计说明

（1）功能：实现管理员对客户的增删改查。

1. 新增客户，程序流程图如图3.18所示：

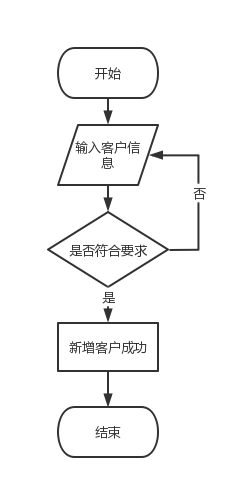


图3.18新增客户流程图

测试要点：

1. 点击新增客户按钮，弹出新增客户窗口。
2. 输入客户的信息，若客户信息无误且完整，则提示新增成功。
3. 若信息不完整，且信息有误，系统会给出提醒。
4. 删除客户，程序流程图如图3.19所示：



图3.19删除客户流程图

测试要点：

1. 点击表格中客户的信息，点击删除，系统提示删除成功。
2. 修改客户，程序流程图如图3.20所示：

测试要点：

1. 点击表格中客户的信息，弹出新的界面。
2. 在新的界面中填写要修改的信息，若输入的信息合法则提示修改成功。
3. 若输入的信息不合法，则提示信息不合法。
4. 查询客户，程序流程图如图3.21所示：

测试要点：

1. 在文本框中输入客户的姓名和id，点击查询按钮。
2. 查询信息将在表格中显示。

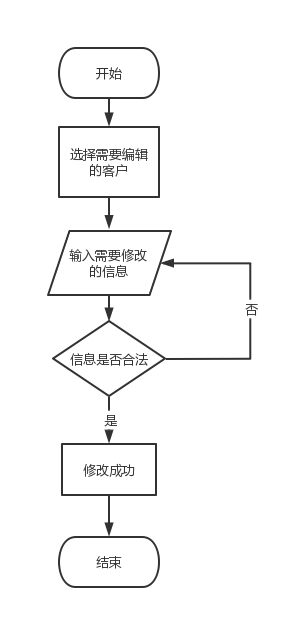
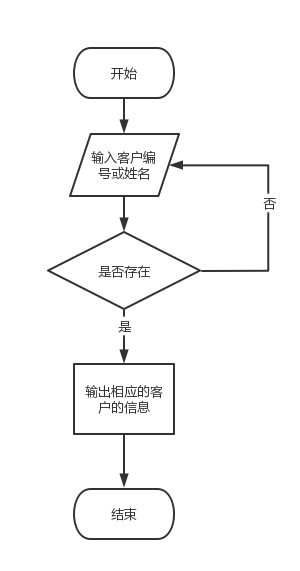
 

图3.20修改客户流程图 示：

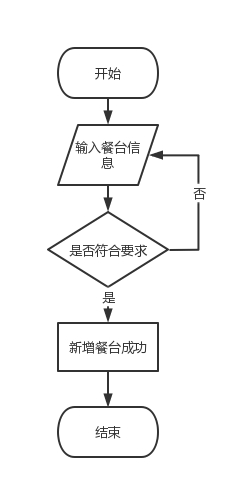


图3.22新增餐台流程图

测试要点：

1. 点击新增餐台按钮，弹出新增餐台界面。
2. 输入餐台的座位数，若输入为数字且合法，则新增餐台成功，否则提示新增失败。
3. 删除餐台，程序流程图如图3.23所示：



图3.23删除餐台流程图

测试要点：

1. 点击餐台信息，点击删除按钮，系统提示删除成功
2. 修改餐台，程序流程图如图3.24所示：

测试要点：

1. 点击表格中餐台的信息，弹出新的界面。
2. 在新的界面中填写要修改的信息，若输入的信息合法则提示修改成功。
3. 若输入的信息不合法，则提示信息不合法。
4. 查询餐台，程序流程图如图3.25所示：

测试要点：

1. 在文本框中输入餐台号，点击查询按钮。
2. 查询信息将在表格中显示。

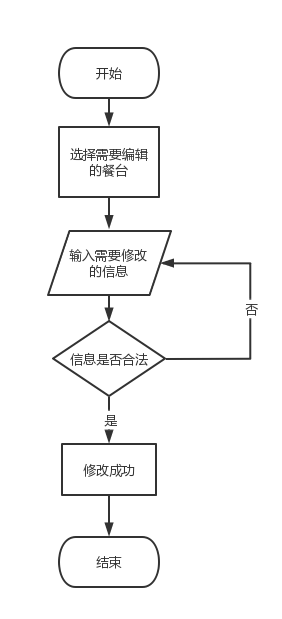
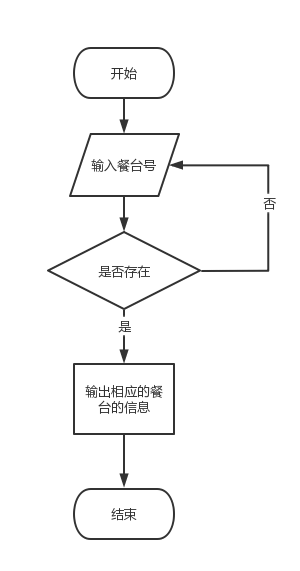
 

图3.24修改餐台流程图 图3.25查询餐台流程图

## 3.9 菜品管理模块设计说明

（1）功能：实现管理员对菜品信息的增删改查

1. 新增菜品，程序流程图如图3.26所示：

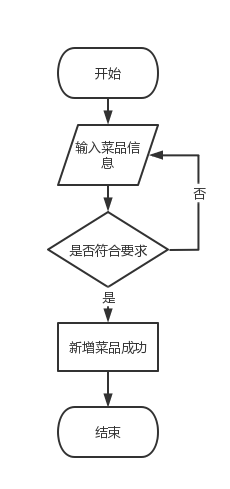


图3.26新增菜品程序流程图

测试要点：

1. 点击新增菜品按钮，弹出新增菜品窗口。
2. 输入菜品的信息，若菜品信息无误且完整，则提示新增成功。
3. 若信息不完整，且信息有误，系统会给出提醒。
4. 删除菜品，程序流程图如图3.27所示：



图3.27删除菜品流程图

测试要点：

1. 点击菜品信息，点击删除按钮，系统提示删除成功。
2. 修改菜品，程序流程图如图3.28所示：

测试要点：

1. 点击表格中餐台的信息，弹出新的界面。
2. 在新的界面中填写要修改的信息，若输入的信息合法则提示修改成功。
3. 若输入的信息不合法，则提示信息不合法。
4. 查询菜品，程序流程图如图3.29所示：

测试要点：

1. 在文本框中输入菜品名或菜品id，点击查询按钮。
2. 查询信息将在表格中显示。

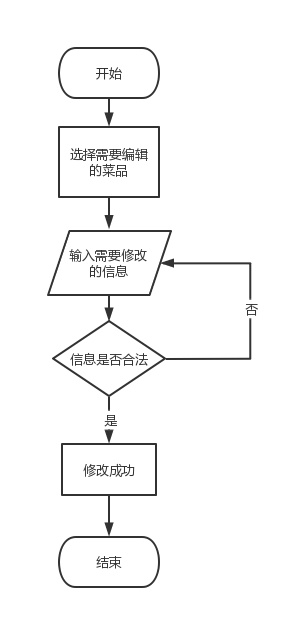
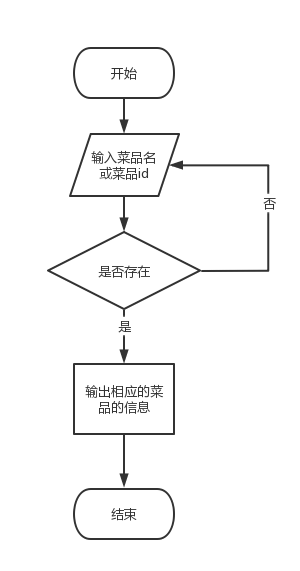
 

图3.28删除菜品程序流程图 图3.29查询菜品程序流程图

## 3.10 点单管理模块设计说明

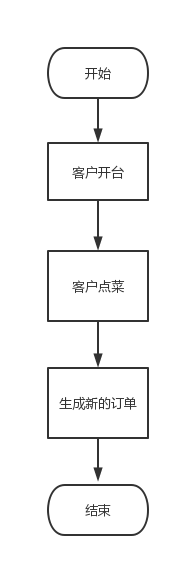
（1）功能：实现管理员对客户进行开台，点单操作。

（2）测试要点：

1. 输入客户的客户号以及人数，为客户开台。

2. 客户选菜，生成新的订单。

（3）程序流程图如图3.30所示：



3.30点菜管理程序流程图

## 3.11 结账管理模块设计说明

（1）功能：实现管理员对订单进行结账管理。

（2）测试要点：

1. 在输入框中输入订单编号或客户编号，查询的订单将在表格中显示。

2. 点击订单，跳转至结账界面，将会根据客户的身份对客户进行优惠。

3. 在新的界面输入金额，找零

（3）程序流程图如图3.31所示：

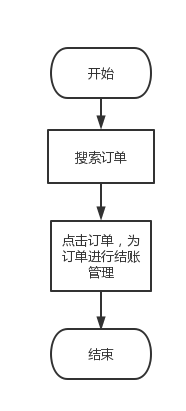


图3.31订单管理流程图

# 4 数据库设计

## 4.1 数据库环境说明

数据库版本：MySQL 8.0.28

运行环境：Windows 11

## 4.2 数据库的命名规则

数据库名：rms

数据表命名规范：

（1）采用26个英文字母（区分大小写）和0-9的自然数（经常不需要）加上下划线’\_’组成，命名简洁明确，多个单词用下划线’\_’分隔。

（2）全部小写命名，禁止出现大写。

（3）禁止使用数据库关键字，如：name,time, datetime,password等。

（4）表名称不应该取得太长（一般不超过三个英文单词）。

（5）表的名称一般使用名词或者动宾短语。

（6）用单数形式表示名称。

（7）表必须填写描述信息（使用SQL语句建表时）。

字段命名规范：

（1）采用26个英文字母（区分大小写）和0-9个自然数（经常不需要）加下划线’\_’组成，命名简洁明确，多个单词用下划线’\_’分隔。

（2）全部小写命名，禁止出现大写。

（3）字段必须填写描述信息。

（4）禁止使用数据库关键字。

（5）字段名称一般采用名词或动宾短语。

（6）采用字段的名称必须是易于理解，一般不超过三个英文单词。

（7）在命名表的列时，不要重复表的名称。

（8）不要在列的名称中包含数据类型。

（9）字段命名使用完整命名，禁止缩写。

## 4.3 逻辑结构设计

关系模型（主键用下划线标注，双下划线表示既是主键又是外键）

管理员（用户名，密码）

客户（客户编号，客户姓名，性别，出生日期，联系方式，身份）

菜品（菜品编号，菜品名，菜品价格，菜品类别，菜品口味，烹饪方式）

员工（员工编号，员工姓名，员工性别，出生日期，联系方式，就职日期）

餐台（餐台编号，餐台状态，餐台人数）

餐台消费记录（订单编号，餐台编号）

客户消费记录（订单编号，消费时间，客户编号）

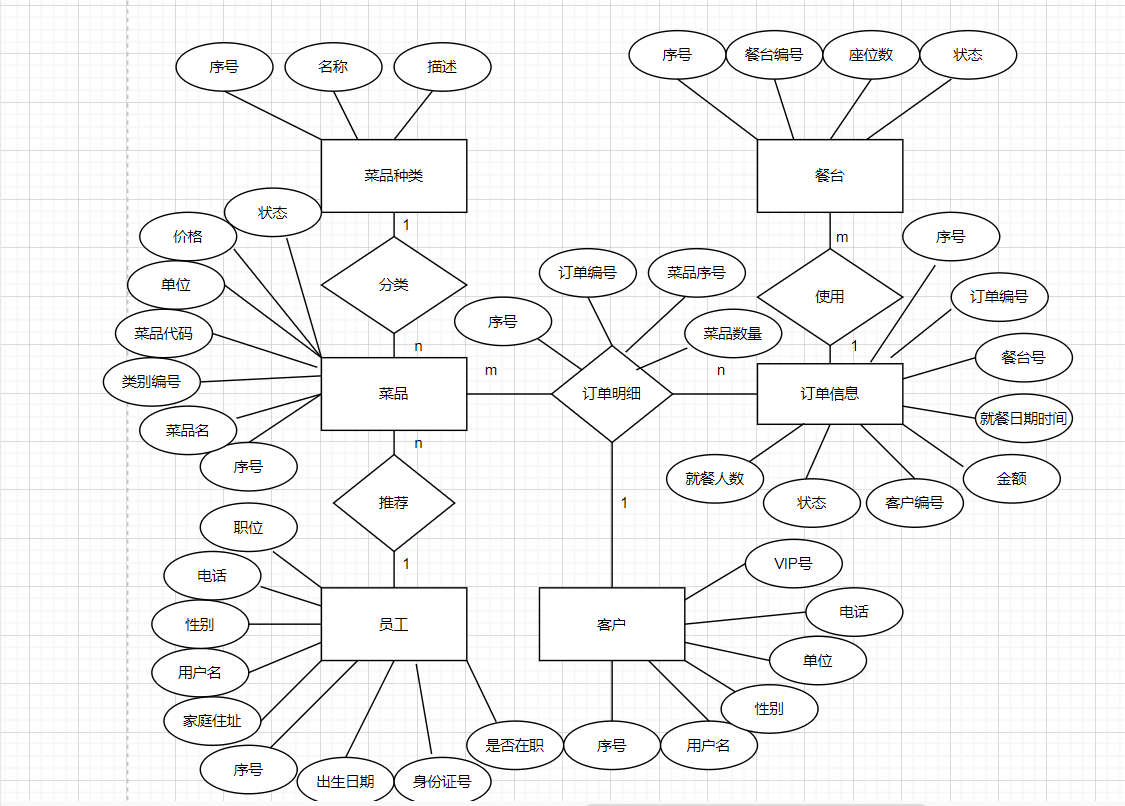
点单（订单编号，菜品编号，数量）

订单（订单编号，餐台编号，客户编号，开台时间，消费金额，订单状态，人数）

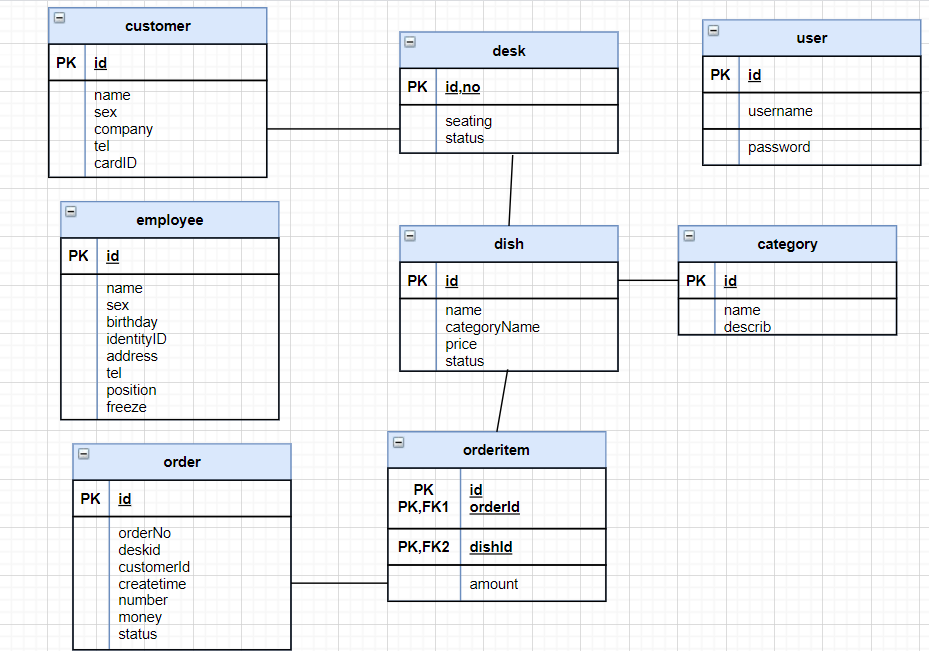
范式分析

所列关系均为第三范式，因为它消除了非主属性对主属性的传递依赖

E-R图



逻辑模型图



## 4.4 物理结构设计

4.4.1 数据库表

表4.1 表汇总

|  |  |
| --- | --- |
| 表 名 | 功 能 说 明 |
| user | 存储管理员信息 |
| employee | 存储员工信息 |
| customer | 存储客户信息 |
| category | 存储菜品分类信息 |
| dish | 存储菜品信息 |
| desk | 存储餐台信息 |
| order | 存储订单信息 |
| orderitem | 存储订单明细信息 |

表4.2 管理员信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | user | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | id | int | N | Y |  |  | 自增/序号 |
| 2 | username | Varchar(20) | N | N |  |  | 用户名 |
| 3 | password | Varchar(20) | N | N |  |  | 密码 |
| sql脚本 | | create table restaurant\_management.category  (  id int auto\_increment comment '序号'  primary key,  name varchar(20) not null comment '名称',  describ varchar(20) null comment '描述'  )  collate = utf8\_czech\_ci; | | | | | |
| 备注 | |  | | | | | |

表4.3 员工信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | Employee | | | | | |
| 数据库用户 | | Root | | | | | |
| 主 键 | | Id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id,name | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Id | Int | N | Y |  |  | 自增/序号 |
| 2 | Name | Varchar(20) | N | N |  |  | 姓名 |
| 3 | Sex | Varchar(2) | Y | N |  |  | 性别 |
| 4 | Birthday | Datetime(8) | Y | N |  |  | 出生日期 |
| 5 | identityID | Varchar(18) | Y | N |  |  | 身份证号 |
| 6 | Address | Varchar(40) | Y | N |  |  | 家庭住址 |
| 7 | Tel | Varchar(11) | Y | N |  |  | 电话 |
| 8 | Position | Varchar(4) | N | N |  |  | 职位 |
| 9 | Freeze | Varchar(4) | N | N |  |  | 是否在职 |
| sql脚本 | | create table restaurant\_management.employee  (  id int auto\_increment comment '序号'  primary key,  name varchar(20) null comment '用户名',  sex varchar(2) not null comment '性别',  birthday datetime not null comment '出生日期',  identityID varchar(18) not null comment '身份证号',  address varchar(40) not null comment '家庭住址',  tel varchar(11) not null comment '电话',  position varchar(4) null comment '职位',  freeze varchar(4) null comment '是否在职'  )  charset = utf8; | | | | | |
| 备注 | |  | | | | | |

表4.4 客户信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | Customer | | | | | |
| 数据库用户 | | Root | | | | | |
| 主 键 | | Id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id,name | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Id | Int | N | Y |  |  | 自增/序号 |
| 2 | Name | Varchar(20) | N | N |  |  | 姓名 |
| 3 | Sex | Varchar(4) | Y | N |  |  | 性别 |
| 4 | Company | Datetime(20) | Y | N |  |  | 单位 |
| 5 | Tel | Varchar(11) | Y | N |  |  | 电话 |
| 6 | cardID | Varchar(10) | N | N |  |  | 贵宾卡号 |
| sql脚本 | | create table restaurant\_management.customer  (  id int auto\_increment comment '序号'  primary key,  name varchar(20) not null comment '用户名',  sex varchar(4) null comment '性别',  company varchar(20) null comment '单位',  tel varchar(11) null comment '电话',  cardID int not null comment '贵宾卡号'  )  collate = utf8\_czech\_ci; | | | | | |
| 备注 | |  | | | | | |

表4.5菜品分类信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | Category | | | | | |
| 数据库用户 | | Root | | | | | |
| 主 键 | | Id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id,name | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Id | Int | N | Y |  |  | 自增/序号 |
| 2 | Name | Varchar(20) | N | N |  |  | 名称 |
| 3 | Describe | Varchar(20) | Y | N |  |  | 描述 |
| sql脚本 | | create table restaurant\_management.category  (  id int auto\_increment comment '序号'  primary key,  name varchar(20) not null comment '名称',  describ varchar(20) null comment '描述'  )  collate = utf8\_czech\_ci; | | | | | |
| 备注 | |  | | | | | |

表4.6 菜品信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | Dish | | | | | |
| 数据库用户 | | Root | | | | | |
| 主 键 | | Id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id,name | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Id | Int | N | Y |  |  | 自增/序号 |
| 2 | Name | Varchar(20) | N | N |  |  | 菜品名 |
| 3 | CategoryId | int(10) | N | N |  |  | 类别编号 |
| 4 | Pic | varchar(50) | N | N |  |  | 图片 |
| 5 | Code | Varchar(8) | N | N |  |  | 菜品代码 |
| 6 | Unit | Varchar(4) | Y | N |  |  | 单位 |
| 7 | Price | double(6) | Y | N |  |  | 价格 |
| 8 | status | Varchar(4) | Y | N |  |  | 状态 |
| sql脚本 | | create table restaurant\_management.dish  (  id int auto\_increment comment '序号'  primary key,  name varchar(20) not null comment '菜品名',  categoryId int not null comment '类别编号',  pic varchar(20) not null comment '图片',  code varchar(8) not null comment '菜品代码',  unit varchar(4) null comment '单位',  price double(6, 2) null comment '价格',  status varchar(4) null comment '状态'  )  collate = utf8\_czech\_ci; | | | | | |
| 备注 | |  | | | | | |

表4.7餐台信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | Desk | | | | | |
| 数据库用户 | | Root | | | | | |
| 主 键 | | Id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id,no | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Id | Int | N | Y |  |  | 自增/序号 |
| 2 | No | Varchar(8) | N | Y |  |  | 餐台编号 |
| 3 | Seating | Int(4) | N | N |  |  | 座位数 |
| 4 | Status | Varchar(10) | N | N |  |  | 状态 |
| sql脚本 | | create table restaurant\_management.desk  (  id int auto\_increment comment '序号',  no varchar(8) not null comment '餐台编号',  seating int not null comment '座位数',  status varchar(10) not null comment '状态：已预定，就餐中，已结账',  primary key (id, no)  )  collate = utf8\_czech\_ci; | | | | | |
| 备注 | |  | | | | | |

表4.8 订单信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | Order | | | | | |
| 数据库用户 | | Root | | | | | |
| 主 键 | | Id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id,orderNo | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Id | Int | N | Y |  |  | 自增/序号 |
| 2 | orderNo | Varchar(20) | N | Y |  |  | 订单编号 |
| 3 | Deskid | int(10) | N | N |  |  | 餐台号 |
| 4 | Createtime | date(40) | N | N |  |  | 就餐日期时间 |
| 5 | Money | double(6) | N | N |  |  | 金额 |
| 6 | Customer | int(10) | N | N |  |  | 客户编号 |
| 7 | Status | varchar(4) | N | N |  |  | 状态 |
| 8 | Number | int(4) | N | N |  |  | 就餐人数 |
| sql脚本 | | create table restaurant\_management.`order`  (  id int auto\_increment comment '序号',  orderNo varchar(20) not null comment '订单编号(当前日期时间+4位随机数)',  deskId int not null comment '餐台号，外键',  createtime datetime not null comment '就餐日期时间',  money double(6, 0) not null comment '金额',  customerId int not null comment '客户编号',  status varchar(4) not null comment '状态：已支付，未支付',  number int not null comment '就餐人数',  primary key (id, orderNo)  )  collate = utf8\_czech\_ci; | | | | | |
| 备注 | |  | | | | | |

表4.9订单明细信息表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | Orderitem | | | | | |
| 数据库用户 | | Root | | | | | |
| 主 键 | | Id | | | | | |
| 其他排序字段 | |  | | | | | |
| 索引字段 | | Id,no | | | | | |
| 序号 | 字段名称 | 数据类型（精度范围） | 允许为空Y/N | 唯一Y/N | 区别度 | 默认值 | 约束条件/说明 |
| 1 | Id | Int | N | Y |  |  | 自增/序号 |
| 2 | Orderid | int(10) | N | Y |  |  | 订单编号 |
| 3 | Dishid | Int(10) | N | N |  |  | 菜品序号 |
| 4 | Amount | double(4) | N | N |  |  | 菜品数量 |
| sql脚本 | | create table restaurant\_management.orderitem  (  id int auto\_increment comment '序号'  primary key,  orderId int not null comment '订单编号，外键',  dishId int not null comment '菜品编号，外键',  amount double(4, 0) not null comment '菜品数量'  )  collate = utf8\_czech\_ci; | | | | | |
| 备注 | |  | | | | | |

# 5 测试用例设计

表5.1 登录-注册测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 登录 | yl-001 | 输入‘用户名’和‘密码’ | admin，123456 | 登录成功 | 通过 |
| yl-002 | admin，1234567 | 登录失败 | 通过 |
| yl-003 | admin，1237236182 | 登录失败 | 通过 |
| yl-004 | admin,空 | 登录失败 | 通过 |
| yl-005 | 空,1237236182 | 登录失败 | 通过 |
| yl-006 | admin,一个空格 | 登录失败 | 通过 |
| yl-007 | 一个空格，1237236182 | 登录失败 | 通过 |
| yl-008 | admin，密码长度超过32 | 登录失败 | 通过 |
| yl-009 | 用户名超过36，1237236182 | 登录失败 | 通过 |
| 注册 | yl-010 | 用户名，密码 | 8888888888,23216323 | 创建成功 | 通过 |
| y1-011 | 用户名，密码 | 8888888888,1 | 创建失败 | 通过 |

表5.2 员工管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 增加员工 | y2-001 | 点击添加按  钮，输入信息 | Anish，女，15273751972,  2002-08-03,2022-06-18 | 新增成功 | 通过 |
| 删除员工 | y2-002 | 点击删除按钮 | Anish | 删除成功 | 通过 |
| 修改员工 | y2-003 | 点击修改按钮 | Anish->A | 修改成功 | 通过 |
| 查询员工 | y2-004 | 输入员工姓名，点击查询  按钮 | A | 查询成功 | 通过 |

表5.3 客户管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 增加客户 | y3-001 | 点击添加按  钮，输入信息 | 啦啦啦，女，2022-06-22，12345678901，是 | 新增成功 | 通过 |
| 删除客户 | y3-002 | 点击删除按钮 | 啦啦啦 | 删除成功 | 通过 |
| 修改客户 | y3-003 | 点击表格 | 女->男 | 修改成功 | 通过 |
| 查询客户 | y3-004 | 输入客户姓名，点击查询  按钮 | 220622787 | 查询成功 | 通过 |

表5.4 菜品管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 增加菜品 | y4-001 | 点击添加按  钮，输入信息 | 2313，50，家禽类，酱香味，烧 | 新增成功 | 通过 |
| 删除菜品 | y4-002 | 点击删除按钮 | 2313 | 删除成功 | 通过 |
| 修改菜品 | y4-003 | 点击表格 | 50->40 | 修改成功 | 通过 |
| 查询菜品 | y4-004 | 输入菜名，点击查询  按钮 | D220623595 | 查询成功 | 通过 |

表5.5 餐台管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 增加餐台 | y5-001 | 点击添加按  钮，输入‘座位数’ | 10 | 新增成功 | 通过 |
| 删除餐台 | y5-002 | 点击删除按钮 | T220623148 | 删除成功 | 通过 |
| 修改餐台 | y5-003 | 点击表格 | 10->20 | 修改成功 | 通过 |
| 查询餐台 | y5-004 | 输入餐台号，点击查询  按钮 | T220623148 | 查询成功 | 通过 |

表5.6 点菜管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 开台 | y6-001 | 输入客户id和人数 | 220622545，1 | 开台成功 | 通过 |
| 生成订单 | y6-002 | 点菜后点击保存 | 无 | 操作成功 | 通过 |

表5.7 结账管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 删除订单 | y7-001 | 点击订单后点击删除按钮 | Z220626767 | 删除成功 | 通过 |
| 结账操作 | Y7-002 | 点击订单后输入金额 | 500 | 支付成功 | 通过 |

表5.8 销售统计测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 功能 | 用例编号 | 操作步骤  （输入条件） | 测试数据 | 预期结果 | 实际结果/判定 |
| 查询全部订单 | Y8-001 | 点击全部订单 | 无 | 查询成功 | 通过 |
| 查询按日统计 | Y8-002 | 点击按日统计 | 无 | 查询成功 | 通过 |
| 查询按月统计 | Y8-003 | 点击按月统计 | 无 | 查询成功 | 通过 |
| 查询按年统计 | Y8-004 | 点击按年统计 | 无 | 查询成功 | 通过 |

# 附录

## 附录1 程序运行结果

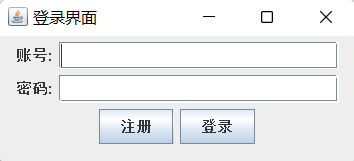
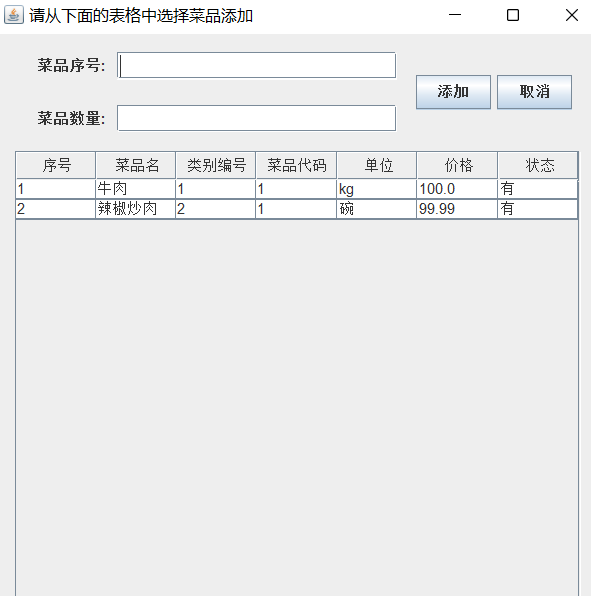


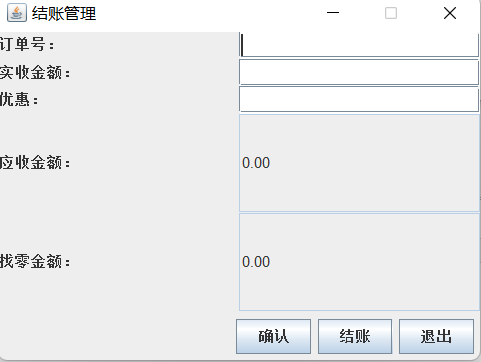
图1 登录注册界面



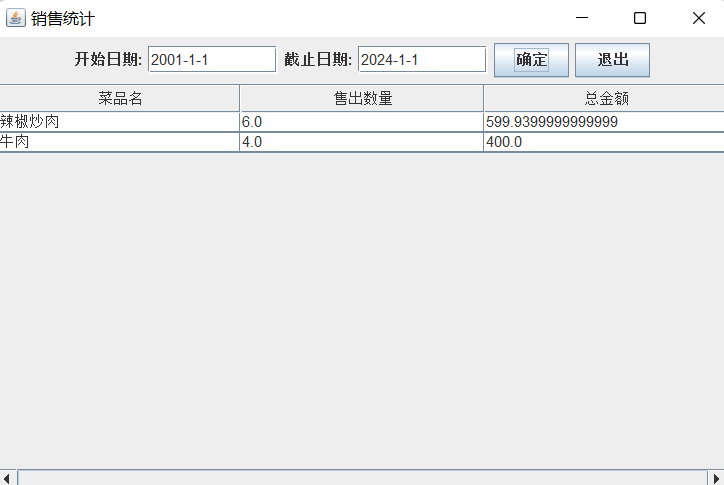
图三员工管理界面



图五 点菜界面



图六 结账界面



图七 销售统计界面

## 附录2 程序源代码

列出自己设计的主要代码

