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** | **高晶晶** |
| **学号** | **B20210304114** |
| **指导教师** | **杨刚、潘怡** |
| **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 | 高晶晶 | 学号 | B20210304114 | 班级 | 21软件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。）  本栏正文排版要求：宋体5号，行距18磅，段首缩进2字符。  本栏可跨多页。  第一阶段（1—4课时）   1. 熟练掌握软件项目可行性分析、开发计划制定及需求分析的方法。 2. 根据待开发系统特点，进行可行性研究并制定有效的开发计划及验收标准，如:实现技术上是否存在无法解决的问题、什么样的实施方案更可行、进度安排是否满足开发要求等。 3. 讨论或借鉴、查找相关资料，深入分析项目需求，撰写需求规格说明书。   第二阶段（5—12课时）   1. 根据需求分析完成餐饮管理系统的系统设计。   第三阶段（13—34课时）   1. 根据第二阶段的系统设计用编码实现餐饮管理系统。   第四阶段（35—38课时）   1. 根据第三阶段的编码进行系统测试，完成实训报告文档。   第五阶段（39—40课时）   1. 交付实施。 | | | | | |
| **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。）  本栏正文排版要求：宋体5号，行距18磅，段首缩进2字符。  本栏可跨多页。  本次实训是本学期最后一次的专业实训﹐回想着刚刚过去的最后一个周﹐紧张沉默之后更多的是感慨﹐印在脑海里的每一个足迹都是那么的深﹐伴着时间﹐就像是在和自己的影子赛跑，不可能从真实的两面去看清它，只是经意不经意的感受着有种东西在过去，两周﹐短短的两周时间，我学到了很多以前并不了解的东西，实在是感受颇深。  作为一名大二的学生﹐经过差不多两年的在校学习，对程序设计有了理性的认识和理解。在校期间，一直忙于理论知识的学习，没有机会也没有相应的经验来参与项目的开发。所以在实训之前，系统开发对我来说是比较抽象的，一个完整的系统要怎么来分工以及完成该项目所要的基本步骤也不明确。然而我也清楚的明白：要学得知识，一靠学二靠实践。没有实践，学习就是无源之水，无本之木。这次为期两周的专业实训，更加让我明白了实践的重要性。在实训中我对如何用JAVA编程语言实现对数据库的操作存在困惑，同时也不太清楚数据库与界面之间的连接问题，但是通过一次次实践，以及查阅资料，我最终成功地完成了本次实训。  然而，本次实训系统还存在着一些问题，存在改进的空间，例如本次餐饮管理系统没有实现菜品推荐以及销售统计问题。可以通过设置菜谱管理系统详细介绍本店特色的菜品，包括每种菜品的唯一标识号、名称、所需材料 以及其所具有的保健功能等一系列详细的信息以供顾客等参考。可以在辅以图片显示所列的菜品 增加介绍的说服力，让顾客全面了解没种菜品的全面信息，根据顾客自身的情况合理安排其饮食。同时，可以通过增加成本和库房管理详细记录每种入库产品的详细信息，比如入库价格、数量、现存量等， 通过对库房存量、进价等的分析判断是否需要进货。 | | | | | |

基于C/S架构的餐饮管理系统

设计说明书

作者：高晶晶(B20210304114)

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） 。

随着我国市场经济的飞速发展，各行业都呈现出生机勃勃的发展景象，其中餐饮业尤为突出，随着餐饮企业规模的不断扩大和数量的不断增长，手动管理模式无论是在工作效率，还是提供决策信息方面都难以满足现代化经营管理的要求，甚至制约了整个餐饮业的规模化发展和整体服务水平的提升。为了使管理更高效、更科学，我们非常有必要开发一种餐饮管理系统。

如今，餐饮管理系统是大部分饭店，酒店等常用的一种平台操作, 这个是基于C/S架构的餐饮平台展现，我所使用的是Eclipse+mysql模式。餐饮管理操作是凌架于互联网系统基础上的一个平台，是对实体店餐饮管理系统的一种延伸和互补，也是直接和客户交易的途径之一。在我们国家，很多消费场所基本上都会使用系统平台，此外，管理系统在餐饮行业更是深受欢迎，在劳动力价格日渐上升的时候，各界行业都在为节省人工成本而努力。因此，餐饮管理系统，已经成为了未来必不可少的工具。

本餐饮系统是为了各大餐饮行业而创造的，有着多年的经营经验积累，无论是商家，顾客都做好了准备，设计主要包含六个模块的分析和设计，设计方向运用的是C/S架构，直接采用java的GUI类进行基本页面功能的设计与功能的实现，后台数据库使用MYSQL数据库。本系统的设计为餐饮管理系统，为餐饮的管理工作提供良好的条件。

**关键词：**java ，C/S架构，餐饮管理系统 ，点菜管理，增删改查

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 3](#_Toc11882509)

[2.3 数据需求 3](#_Toc11882510)

[3 系统设计 4](#_Toc11882511)

[3.1 运行环境 4](#_Toc11882512)

[3.2 系统静态结构设计 4](#_Toc11882513)

[3.3 人机接口设计 5](#_Toc11882514)

[3.4 系统管理模块设计说明 7](#_Toc11882515)

[3.5 业务管理模块设计说明 7](#_Toc11882516)

[3.6 结账管理模块设计说明 7](#_Toc11882516)

[4 数据库设计 8](#_Toc11882517)

[4.1 数据库环境说明 8](#_Toc11882518)

[4.2 数据库的命名规则 8](#_Toc11882519)

[4.3 逻辑结构设计 8](#_Toc11882520)

[4.4 物理结构设计 9](#_Toc11882521)

[4.5 安全性设计 10](#_Toc11882522)

[5 测试用例设计 12](#_Toc11882523)

[附录 14](#_Toc11882524)

[附录1 程序运行结果 14](#_Toc11882525)

[附录2 程序源代码 15](#_Toc11882526)

# 1 引言

## 1.1 编写目的

本文档是“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高级程序设计实战教程：微课版. 北京：人民邮电出版社，2022.

[3] 刘涛. 中国餐营业特许经营研究[D]. 北京：对外经济贸易大学, 2011.11.

[4]李春葆. 数据库系统设计与开发. 北京：清华大学出版社，2003.8.

# 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。

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。

选做扩展：菜品推荐、销售统计。

餐饮管理系统

%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%26lt%3Bfont%20face%3D%26quot%3BVerdana%26quot%3B%20style%3D%26quot%3Bfont-size%3A%2018px%26quot%3B%26gt%3B%E9%A4%90%E9%A5%AE%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22130%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%223%22%20value%3D%22%26lt%3Bp%26gt%3B%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E5%91%98%E5%B7%A5%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%26lt%3B%2Fp%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%2290%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%224%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E5%AE%A2%E6%88%B7%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22210%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%225%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E9%A4%90%E5%8F%B0%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22330%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%226%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E8%8F%9C%E5%93%81%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22450%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%227%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E4%B8%9A%E5%8A%A1%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22570%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%228%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E7%BB%93%E8%B4%A6%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22690%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%229%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%222%22%20target%3D%223%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22450%22%20y%3D%22370%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3CArray%20as%3D%22points%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22240%22%2F%3E%3CmxPoint%20x%3D%22150%22%20y%3D%22240%22%2F%3E%3C%2FArray%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2210%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%224%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22270%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2211%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%225%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22390%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2212%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%228%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22190%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3CArray%20as%3D%22points%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22240%22%2F%3E%3CmxPoint%20x%3D%22750%22%20y%3D%22240%22%2F%3E%3C%2FArray%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2213%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%226%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22510%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2214%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%227%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22630%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E%3CmxGraphModel%3E%3Croot%3E%3CmxCell%20id%3D%220%22%2F%3E%3CmxCell%20id%3D%221%22%20parent%3D%220%22%2F%3E%3CmxCell%20id%3D%222%22%20value%3D%22%26lt%3Bfont%20face%3D%26quot%3BVerdana%26quot%3B%20style%3D%26quot%3Bfont-size%3A%2018px%26quot%3B%26gt%3B%E9%A4%90%E9%A5%AE%E7%AE%A1%E7%90%86%E7%B3%BB%E7%BB%9F%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22380%22%20y%3D%22130%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%223%22%20value%3D%22%26lt%3Bp%26gt%3B%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E5%91%98%E5%B7%A5%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%26lt%3B%2Fp%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%2290%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%224%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E5%AE%A2%E6%88%B7%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22210%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%225%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E9%A4%90%E5%8F%B0%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22330%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%226%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E8%8F%9C%E5%93%81%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22450%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%227%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E4%B8%9A%E5%8A%A1%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22570%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%228%22%20value%3D%22%26lt%3Bfont%20style%3D%26quot%3Bfont-size%3A%2014px%26quot%3B%26gt%3B%E7%BB%93%E8%B4%A6%E7%AE%A1%E7%90%86%26lt%3B%2Ffont%26gt%3B%22%20style%3D%22rounded%3D0%3BwhiteSpace%3Dwrap%3Bhtml%3D1%3Brotation%3D90%3B%22%20vertex%3D%221%22%20parent%3D%221%22%3E%3CmxGeometry%20x%3D%22690%22%20y%3D%22370%22%20width%3D%22120%22%20height%3D%2260%22%20as%3D%22geometry%22%2F%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%229%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BexitX%3D0.5%3BexitY%3D1%3BexitDx%3D0%3BexitDy%3D0%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20source%3D%222%22%20target%3D%223%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22450%22%20y%3D%22370%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3CArray%20as%3D%22points%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22240%22%2F%3E%3CmxPoint%20x%3D%22150%22%20y%3D%22240%22%2F%3E%3C%2FArray%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2210%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%224%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22270%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2211%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%225%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22390%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2212%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%228%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22190%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3CArray%20as%3D%22points%22%3E%3CmxPoint%20x%3D%22440%22%20y%3D%22240%22%2F%3E%3CmxPoint%20x%3D%22750%22%20y%3D%22240%22%2F%3E%3C%2FArray%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2213%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%226%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22510%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3CmxCell%20id%3D%2214%22%20value%3D%22%22%20style%3D%22endArrow%3Dclassic%3Bhtml%3D1%3BentryX%3D0%3BentryY%3D0.5%3BentryDx%3D0%3BentryDy%3D0%3B%22%20edge%3D%221%22%20target%3D%227%22%20parent%3D%221%22%3E%3CmxGeometry%20width%3D%2250%22%20height%3D%2250%22%20relative%3D%221%22%20as%3D%22geometry%22%3E%3CmxPoint%20x%3D%22630%22%20y%3D%22240%22%20as%3D%22sourcePoint%22%2F%3E%3CmxPoint%20x%3D%22500%22%20y%3D%22320%22%20as%3D%22targetPoint%22%2F%3E%3C%2FmxGeometry%3E%3C%2FmxCell%3E%3C%2Froot%3E%3C%2FmxGraphModel%3E

（**图名： 仿宋，小四。**

**图内文字：宋体，五号。）**

结账管理

业务管理

菜品管理

餐台管理

客户管理

员工管理

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：餐饮系统管理员对员工的信息进行增删改查。对员工的姓名，性别，出生日期，身份证号，家庭住址，电话，职位，是否在职等信息进行操作。

前置条件：管理员请求管理员工信息。

后置条件：管理员对人员信息进行增删改查。

活动步骤：

1. 管理员登录餐饮管理系统。
2. 管理员请求管理员工信息。
3. 管理员选择如何操作员工信息。
4. 系统响应所有操作。
5. 管理员成功完成对员工信息的操作。

扩展点：

1. 可同时增加，删除，修改多位员工的信息。

异常处理：

1. 如果管理员没有选中需要删除的员工信息行，则用例终止。
2. 如果输入的信息不符合规范，则用例终止。

2.1.2 业务管理用例说明

[解释] 要逐一说明每个基本用例。其中活动步骤（流程）、扩展点、异常处理是重点描述内容。对于扩展用例和包含用例可以不单独列出，也可以单独列出。

用例名称：业务管理。

用例描述：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对应起来，分别显示出来，并记录开台时间。

前置条件：服务员请求开台和点菜。

后置条件：服务员对某客户、某一空闲餐台实行开台，同时实现点菜。

活动步骤：

1. 服务员登录餐饮管理系统。
2. 服务员请求开台和点菜操作。
3. 服务员进行开台和点菜操作。
4. 系统响应所有操作。
5. 服务员成功完成开台和点菜操作。

扩展点：

1. 可开台，以及查询餐台的信息。
2. 可实现点菜功能，同时可以对菜品进行增删改。

异常处理：

1. 如果服务员没有选中需要进行操作的菜品信息行，则用例终止。
2. 如果输入的信息不符合规范，则用例终止。

2.1.3 结账管理用例说明

[解释] 要逐一说明每个基本用例。其中活动步骤（流程）、扩展点、异常处理是重点描述内容。对于扩展用例和包含用例可以不单独列出，也可以单独列出。

用例名称：结账管理。

用例描述：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以及订单信息，通过手动输入餐台号，消费金额，实收金额点击计算进行找零的计算，并显示找零金额，然后点击结账完成结账的操作，并记录成统计数据。

前置条件：收银员请求结账。

后置条件：收银员为某餐台的客户进行结账。

活动步骤：

1. 收银员登录餐饮管理系统。
2. 收银员请求结账操作。
3. 收银员进行结账操作。
4. 系统响应所有操作。
5. 收银员成功完成结账操作。

扩展点：

1. 可查询客户的订单信息。
2. 可对订单信息进行增删改。

异常处理：

1. 如果收银员没有选中需要进行操作的订单信息行，则用例终止。

如果输入的信息不符合规范，则用例终止。

1. 如果收银员输入的餐台号不存在，则用例终止。

## 

## 2.2 界面需求

提示：说明用户对交互界面的要求。

1. 良好的界面感受，要求界面简洁，功能强大；
2. 实现系统所需的全部功能，
3. 便捷的信息检索功能；
4. 稳定又健壮的数据库管理系统；
5. 实时的信息交互；
6. 良好的可扩展性。

## 2.3 数据需求

提示：说明程序需处理的数据实体及其属性，数据名称，数据格式，数据结构，数据取值范围，数据存储要求，数据处理过程。例如，用户实体

学生实体属性有学号，姓名,…。

表2.1 管理员信息表（user）

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

表2.2 员工信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否允许为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| name | varchar | 20 | 否 | 否 | 用户名 |
| sex | varchar | 2 | 是 | 否 | 性别 |
| birthday | datatime | 8 | 是 | 否 | 出生日期 |
| identityID | varchar | 18 | 是 | 否 | 身份证号 |
| address | varchar | 40 | 是 | 否 | 家庭住址 |
| 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 客户信息表（customer）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否允许为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| name | varchar | 20 | 否 | 否 | 用户名 |
| sex | varchar | 4 | 是 | 否 | 性别 |
| company | varchar | 20 | 是 | 否 | 单位 |
| tel | varchar | 11 | 是 | 否 | 电话 |
| cardID | varchar | 10 | 否 | 否 | 贵宾卡号 |

表2.4 菜品分类信息表（category）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否允许为空 | 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 菜品信息表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否允许为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| name | varchar | 20 | 否 | 否 | 菜品名 |
| categoryid | int | 10 | 否 | 否 | 类别编号 |
| pic | blob | 50 | 否 | 否 | 图片 |
| code | varchar | 8 | 否 | 否 | 菜品代码 |
| unit | varchar | 4 | 是 | 否 | 单位 |
| price | datatime | 6 | 是 | 否 | 价格 |
| status | varchar | 4 | 是 | 否 | 状态 |

表2.6 餐台信息表（desk)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否允许为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| no | varchar | 8 | 否 | 是 | 餐台编号 |
| seating | int | 4 | 否 | 否 | 座位数 |
| status | varchar | 10 | 否 | 否 | 状态为：已预定、就餐中、已结账 |

表2.7 订单信息表（order）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否允许为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| orderNo | varchar | 20 | 否 | 是 | 订单编号（当前日期时间+4位随机数） |
| deskId | int | 10 | 否 | 否 | 餐台号，外键 |
| createtime | date | 40 | 否 | 否 | 就餐日期时间 |
| money | double | 6 | 否 | 否 | 金额 |
| customerId | int | 10 | 否 | 否 | 客户编号 |
| status | varchar | 4 | 否 | 否 | 状态为：已支付、未支付 |
| number | int | 4 | 否 | 否 | 就餐人数 |

表2.8 订单明细信息表（orderitem）

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 属性名称 | 类型 | 长度 | 是否允许为空 | 唯一标识 | 说明 |
| id | int | 10 | 否 | 是 | 序号 |
| orderId | int | 10 | 否 | 否 | 订单编号（当前日期时间+4位随机数）,外键 |
| dishId | int | 10 | 否 | 否 | 菜品序号，外键 |
| amount | double | 4 | 否 | 否 | 菜品数量 |

# 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。

本餐饮管理系统功能完善，操作简单，界面清晰，系统主要实现了员工管理，客户管理，餐台管理，菜品管理，开台管理，点菜管理，结账管理7个部分，每个功能实现的思路如下。

1. 员工管理

建立员工表结构类，员工信息管理类，对员工进行操作的类，对员工数据库进行操作的类，引用接口实现员工信息数据库操作的类。

1. 客户管理

建立顾客表结构类，顾客信息管理类，对顾客进行操作的类，对顾客数据库进行操作的类，引用接口实现顾客信息数据库操作的类。

1. 餐台管理

建立餐台表结构类，餐台信息管理类，对餐台进行操作的类，对餐台数据库进行操作的类，餐台查询界面类，引用接口实现餐台信息数据库操作的类。

1. 菜品管理

建立菜品和菜品类别表结构类，菜品信息管理管理类，对菜品进行操作的类，对菜品数据库进行操作的类，菜品添加界面类，引用接口实现菜品信息数据库操作的类。

1. 开台管理

建立开台管理界面类，进行开台操作的类，餐台查询界面类，引用接口实现开台信息数据库操作的类。

1. 点菜管理

建立点菜管理界面类，进行点菜操作的类，对菜品进行操作的类，引用接口实现点菜信息数据库操作的类。

1. 结账管理

建立订单信息表结构类，订单信息管理类，对订单进行操作的类，金额计算界面类，结账管理界面类，引用接口实现订单信息数据库操作的类。

## 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。

1. 客户端：windows操作系统，JDK10.0.2，Eclipse。
2. 服务器：MySQL8.0.28数据库服务器。

## 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3.1的餐饮管理系统中，系统的软件静态结构如下：

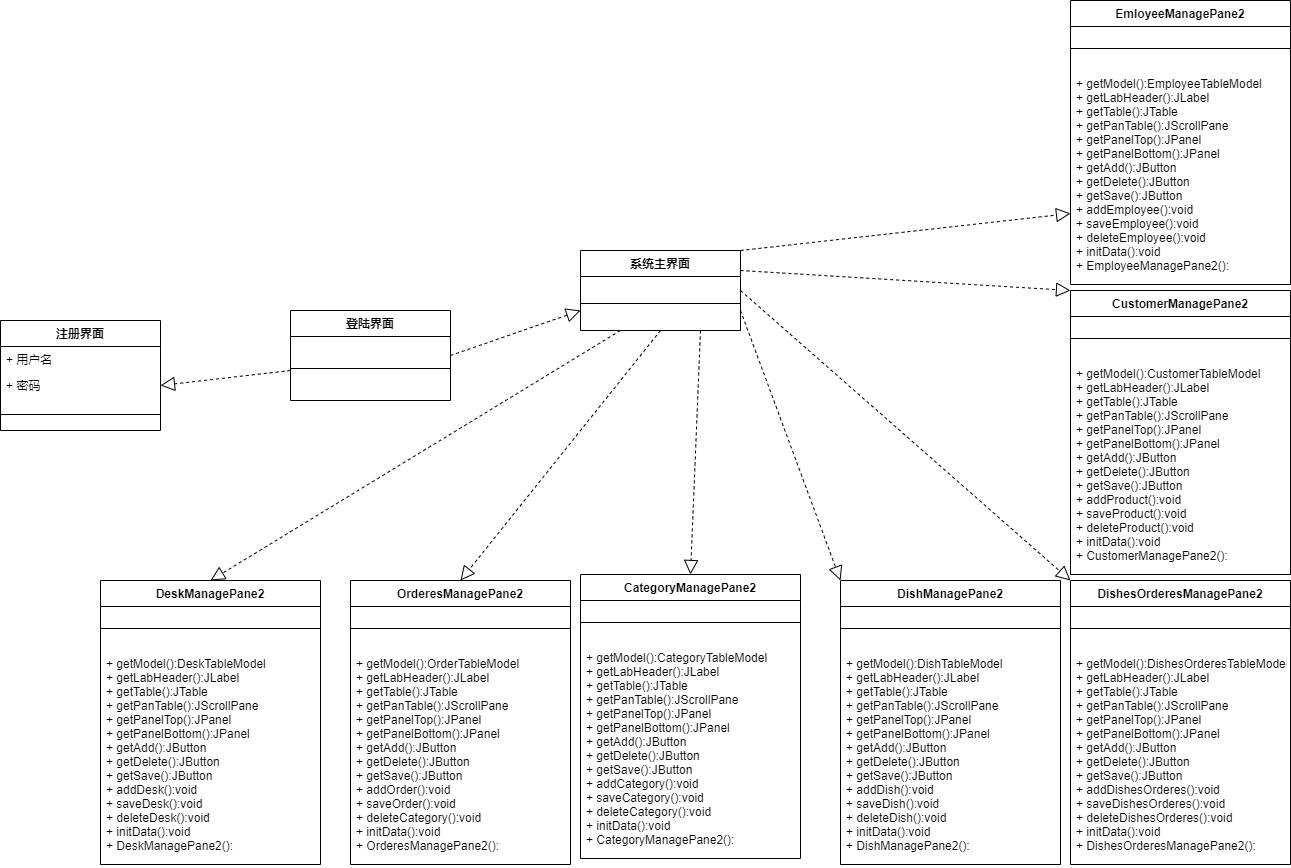
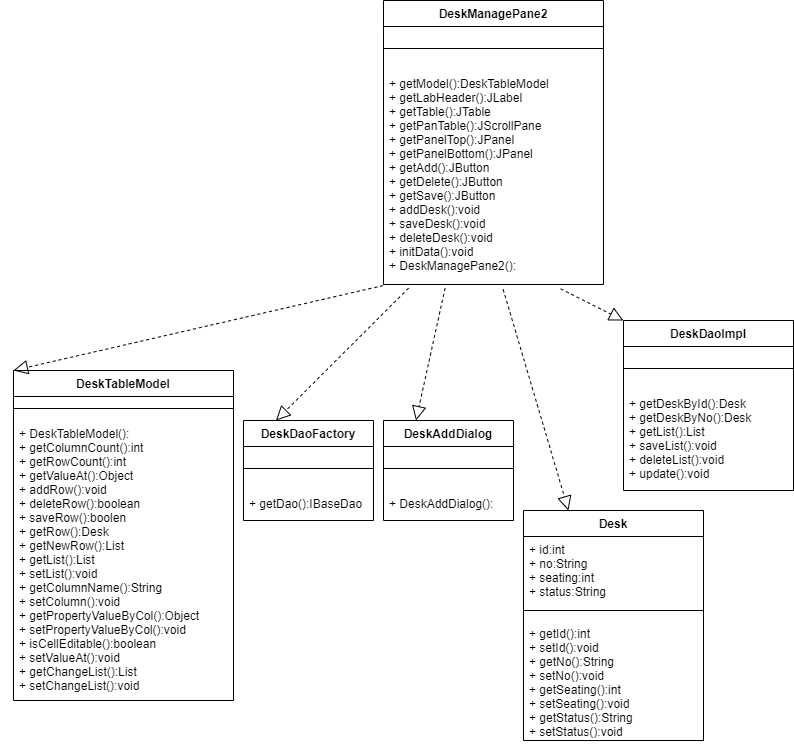
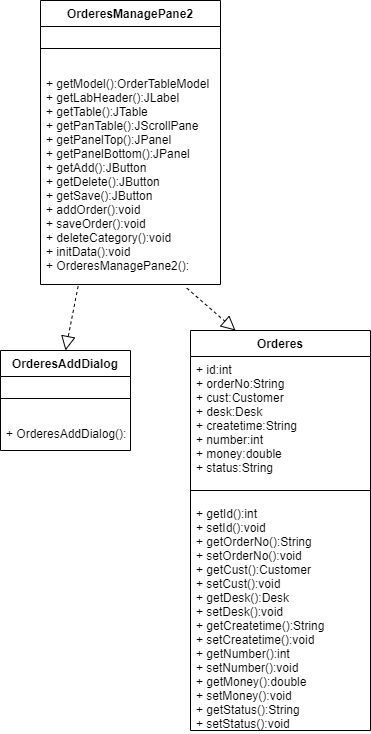
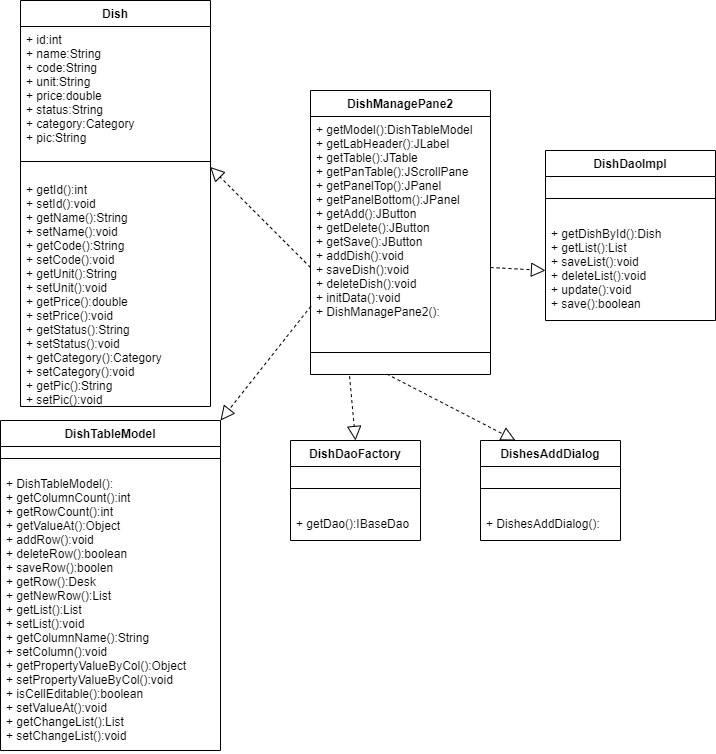
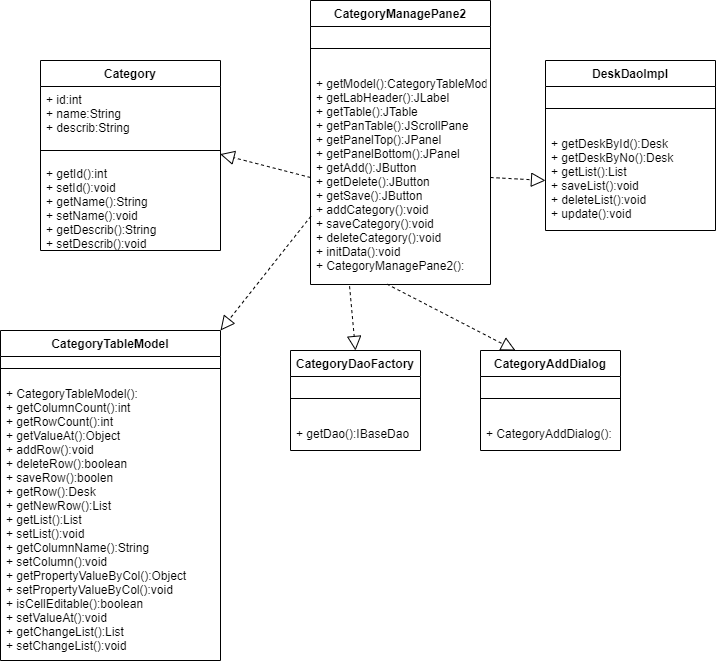


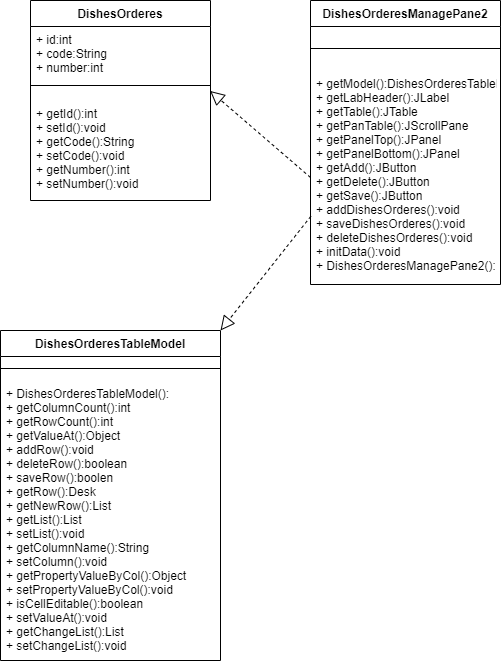
图3.1 餐饮管理系统总体类图

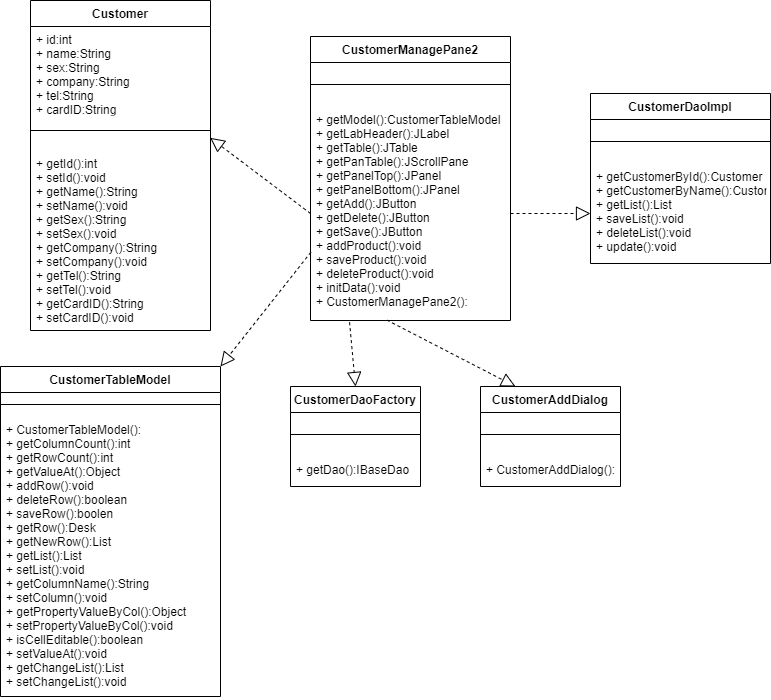
**各模块类图如下:**

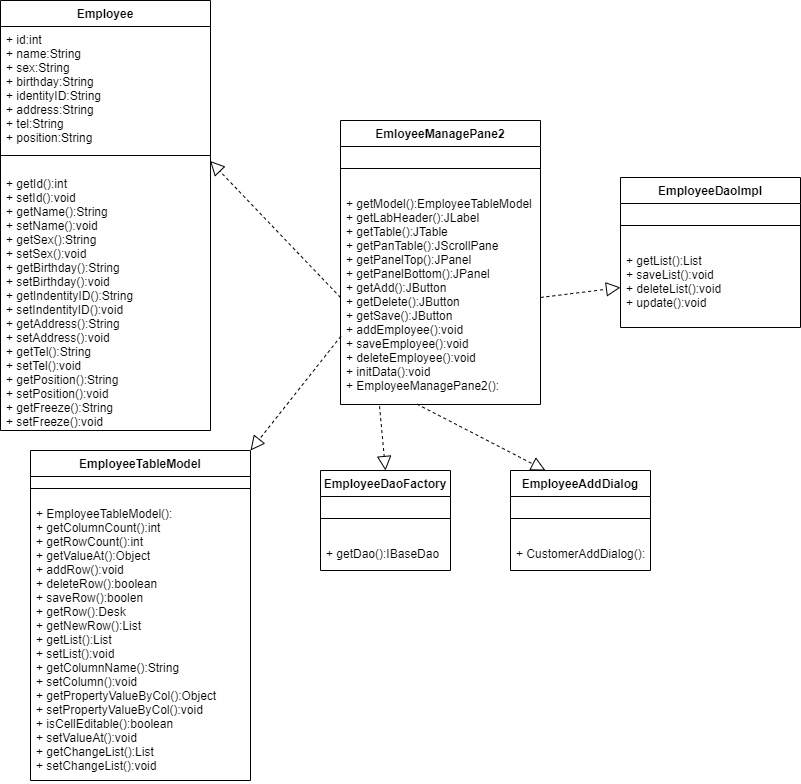












系统静态结构说明

针对3.1中绘制的类图说明在类设计是如何体现功能需求和非功能需求的。

类图（Class Diagram）将系统中不同的类抽象出来描述系统的静态结构，包括类以及它们之间的关系。在UML 的静态机制中类图是一个重点， 在UML 的各种图中占据了相当重要的地位，它不仅为设计人员所关心，同时，因所有软件系统都用对象或类作为其主要构造块，建模工具也主要依据类图来产生代码（正向）工程，因此系统的实现人员也非常关心类图。类是对具有共同性质的一组对象（从建模者的视角）的描述，是一组对象的低层结构和行为规范，即一种对本质相同事物的抽象，类中的每一个对象都是某个类的实例，一个特定类的所有对象具有相同的数据结构和行为，尽管它们的属性值可能互不相同，也就是说，每一个对象也都有标识（能够对它命名，以区别 于其他对象）、状态（通常有一些数据与它相联系）和行为（能对该对象做某些 事，它也能为其他对象做某些事）。在UML 中，类图显示了一组类、接口、协作以及它们之间的关系，类图包含一些具体的行为元素，操作他们的动态特征是在其他图中表示的，如状态图和协作图。在类图中类用矩形框来表示，它的属性和操作分别列在分格中。

· **功能需求**：规定开发人员**必须在系统中实现的软件功能**，满足业务需要

· **非功能需求**：系统必须具备的除功能需求外的特性，非功能性需求反映系统质量和特性的额外要求。根据系统的不同，非功能性需求的着重点也不同，在本系统中，非功能性需求的要求主要体现在冗余性,标准化，可移植性等方面，主要包括以下几点：

1. 系统的性能需求分析
2. 界面需求
3. 安全需求

## 

## 3.3 人机接口设计

3.3.1 设计应用类

给出了应用类模型，进一步说明有多少边界类。如有可能，最好进一步设计控制类和持久类，并通过应用类模型来说明它们之间的关系。

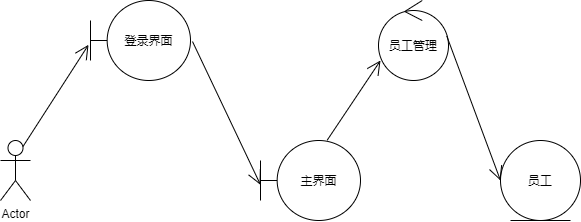


图3.2.1 员工管理的设计类图

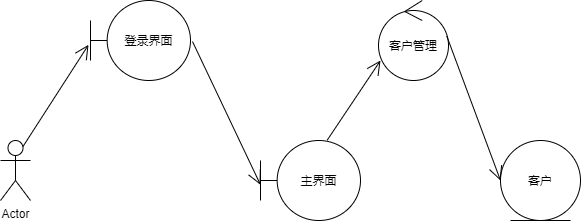


图3.2.2 客户管理的设计类图

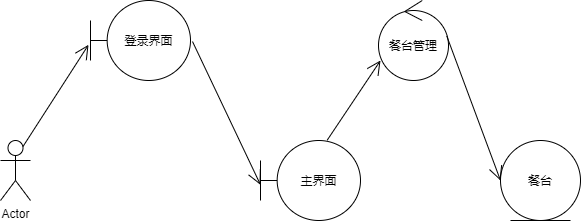


图3.2.3 餐台管理的设计类图

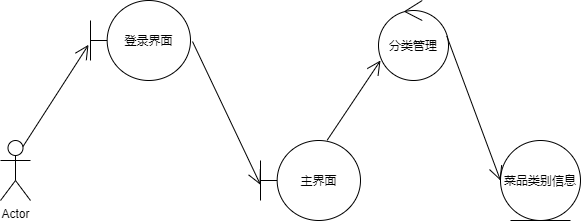


图3.2.4 分类管理的设计类图

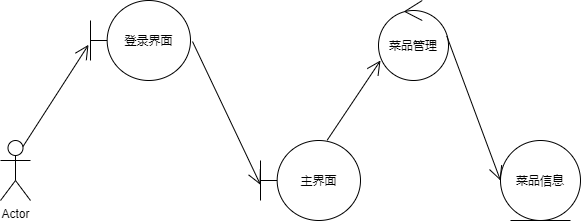


图3.2.5 菜品管理的设计类图

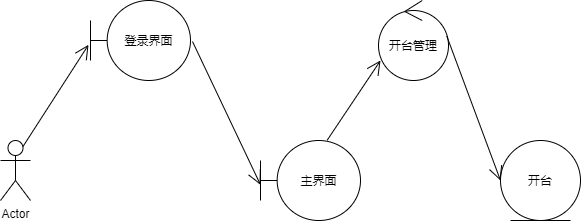


图3.2.6 开台管理的设计类图

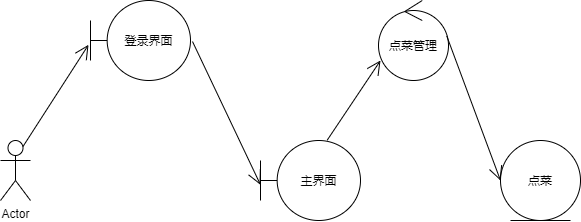


图3.2.7 点菜管理的设计类图

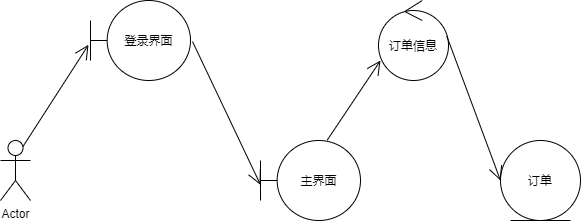


图3.2.8 订单信息的设计类图

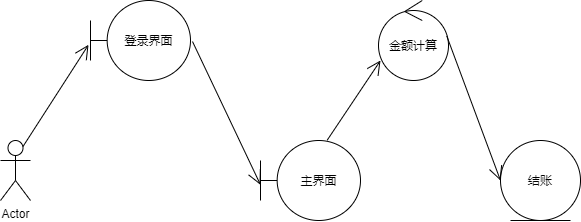


图3.2.9 金额计算的设计类图

3.3.2 登陆界面（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3.3



图3.3 登录界面布局设计

说明界面元素主要控件，包括控件类型、控件名称、控件作用、控件属性（仅说明使用的属性）。说明本用户接口所实现的功能，例如是否设计有密码位数控制，用户名是否设计有限制等等。

控件类型：界面名称

控件名称：餐饮管理系统

控件作用：显示界面标题

控件属性：决定界面标题

控件类型：按钮

控件名称：注册，登录

控件作用：触发事件

控件属性：实现功能

控件类型：文本框

控件名称：用户名，密码

控件作用：显示文字

控件属性：显示文字

3.3.3 结账管理（具体标识符）设计



控件类型：界面名称

控件名称：结账管理

控件作用：显示界面标题

控件属性：决定界面标题

控件类型：按钮

控件名称：计算，结账

控件作用：触发事件

控件属性：实现功能

控件类型：文本框

控件名称：餐台号，消费金额，实收金额，找零金额

控件作用：显示文字

控件属性：显示文字

## 3.4 系统管理模块设计说明

从小本节开始，详细说明每个模块的详细设计，逐个地给出需求规格的每个用例的设计考虑。要求给出所有用例的实现方法。

根据餐饮管理系统的职责范围和需求可以确定三个参与者:系统管理员、服务员、收银员。

1. 首先实现系统管理员管理员工信息，通过EmplyeeManagePane2实现对员工信息的增删改查操作；
2. 接着实现系统管理员管理客户信息，通过CustomerManagePane2实现对客户信息的增删改查操作；
3. 最后实现修改密码界面，通过PasswordManagePane实现对密码的修改。

## 3.5 业务管理模块设计说明

用类似第3.4小节的方式，说明第2个模块乃至第N个模块的设计考虑。

根据餐饮管理系统的职责范围和需求可以确定三个参与者:系统管理员、服务员、收银员。

1. 首先实现服务员为某一客户开台，通过OrderesAddDialog实现对开台操作；
2. 接着实现服务员为某一客户点菜，通过DishesOrderesManagePane2实现为客户点菜操作；

## 3.6 结账管理模块设计说明

用类似第3.4小节的方式，说明第2个模块乃至第N个模块的设计考虑。

根据餐饮管理系统的职责范围和需求可以确定三个参与者:系统管理员、服务员、收银员。

1. 首先实现收银员管理订单信息，通过OrderesManagePane2实现订单信息管理操作；
2. 接着实现收银员为某一客户结账，通过CheckoutDialog实现对开台操作；

# 4 数据库设计

## 4.1 数据库环境说明

提示：说明所采用的数据库系统，设计工具，编程工具等，说明其详细配置。

mysql-8.0.28-winx64,Eclipse,jdk10.0.2

## 4.2 数据库的命名规则

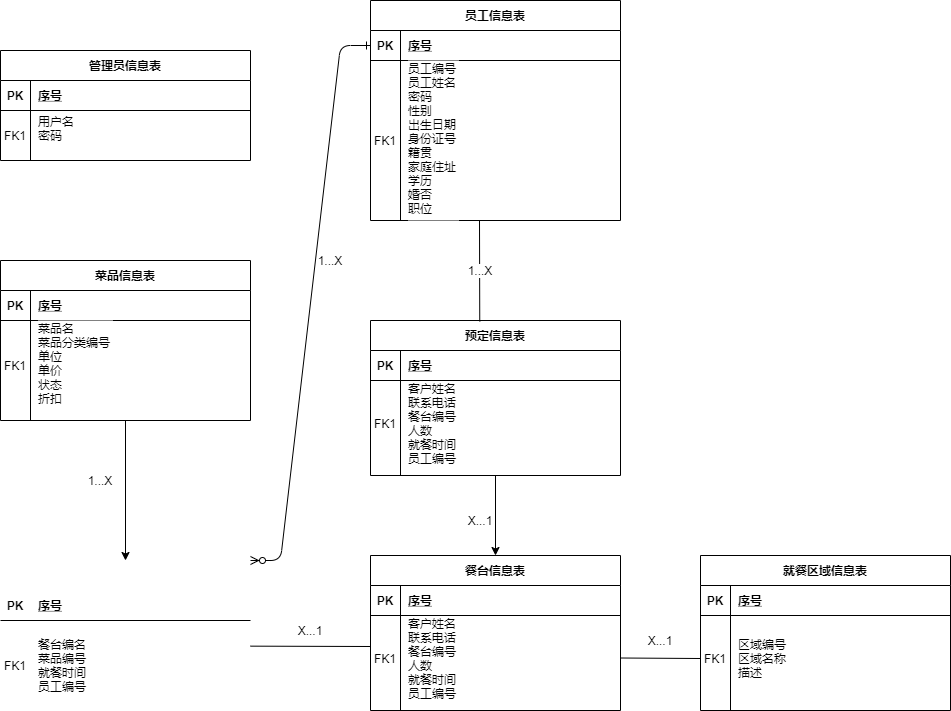
提示：完整并且清楚的说明本数据库的命名规则。如果本数据库的命名规则与机构的标准不完全一致的话，请作出解释。

本数据库设计完全按照《数据库设计规范》命名。

## 4.3 逻辑结构设计

提示：设计逻辑模型。说明把上述原始数据进行分解、合并后重新组织起来的数据库全局逻辑结构，包括所确定的关键字和属性、重新确定的记录结构和文卷结构、所建立的各个文卷之间的相互关系，形成本数据库的数据库管理员视图。

E-R图



**关系模型**

管理员信息(序号，用户名，密码)

菜品信息（序号，菜品名，菜品分类编号，单位，单价，状态，折扣）

点菜信息（序号，餐台编号，菜品编号，就餐时间，员工编号)

员工信息（序号，员工编号，员工信息，密码，性别，出生日期，身份证号，籍贯，家庭住址，学历，婚否，职位)

预定信息(序号，客户姓名，联系电话，餐台编号，人数，就餐时间，员工编号）

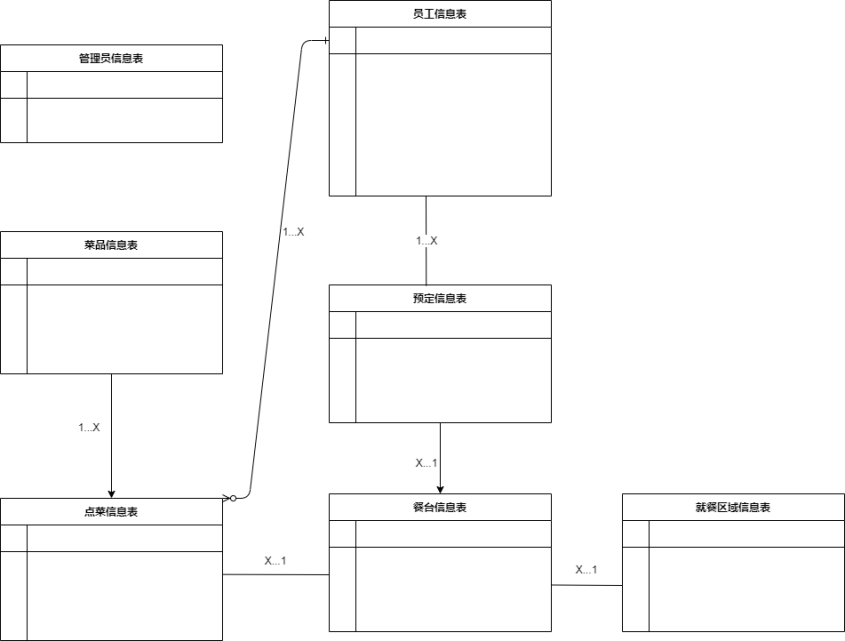
餐台信息（序号，餐台编号，餐台名称，人数，区域编号，人数，状态)

就餐区域信息（序号，区域编号，区域名称，描述）

**范式分析**

该关系模式属于1NF。

逻辑模型图



## 4.4 物理结构设计

提示：建立系统程序员视图，包括：数据在内存中的安排，包括对索引区、缓冲区的设计；所使用的外存设备及外存空间的组织，包括索引区、数据块的组织与划分；访问数据的方式方法。下列必选项必须说明：

（1）主要是设计表结构。一般地，实体对应于表，实体的属性对应于表的列，实体之间的关系成为表的约束。逻辑设计中的实体大部分可以转换成物理设计中的表，但是它们并不一定是一一对应的。

（2）对表结构进行规范化处理（第三范式）。

4.4.1 数据库表

说明数据库的表的设计。

表4.1 表汇总

|  |  |
| --- | --- |
| 表 名 | 功 能 说 明 |
| category | 存储菜品种类信息 |
| customer | 存储顾客信息 |
| desk | 存储餐台信息 |
| dish | 存储菜品信息 |
| emplyee | 存储员工信息 |
| orderes | 存储订单信息 |
| orderitem | 存储订单明细 |
| user | 存储用户信息 |

表4.2 category表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | category | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | name | varchar(20) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '名称' |
| 3 | describ | varchar(20) | Y | N | 低 | NULL | COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '描述' |
|  |  |  |  |  |  |  |  |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

表4.3 customer表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | customer | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | name | varchar(20) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '用户名' |
| 3 | sex | varchar(4) | Y | N | 低 | NULL | COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '性别' |
| 4 | company | varchar(20) | Y | N | 低 | NULL | COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '单位' |
| 5 | tel | varchar(11) | Y | N | 低 | NULL | COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '电话' |
| 6 | cardID | int(10) | N | N | 低 |  | NOT NULL COMMENT '贵宾卡号' |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

表4.4 desk表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | desk | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id,no | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | no | varchar(8) | N | Y | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '餐台编号' |
| 3 | seating | Int(4) | N | N | 低 |  | NOT NULL COMMENT '座位数' |
| 4 | status | varchar(10) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '状态：已预定，就餐中，已结账' |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

表4.5 dish表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | dish | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | name | varchar(20) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '菜品名' |
| 3 | categoryId | int(10) | N | N | 低 |  | NOT NULL COMMENT '类别编号' |
| 4 | pic | varchar(20) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '图片' |
| 5 | code | varchar(8) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '菜品代码' |
| 6 | unit | varchar(4) | Y | N | 低 | NULL | COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '单位' |
| 7 | price | double(6,2) | Y | N | 低 | NULL | DEFAULT NULL COMMENT '价格' |
| 8 | status | varchar(4) | Y | N | 低 | NULL | COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '状态' |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

表4.6 employee表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | employee | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | name | varchar(20) | Y | N | 低 | NULL | DEFAULT NULL COMMENT '用户名' |
| 3 | sex | varchar(2) | N | N | 低 |  | NOT NULL COMMENT '性别' |
| 4 | birthday | datetime | N | N | 低 |  | NOT NULL COMMENT '出生日期' |
| 5 | identityID | varchar(18) | N | N | 低 |  | NOT NULL COMMENT '身份证号' |
| 6 | address | varchar(40) | N | N | 低 |  | NOT NULL COMMENT '家庭住址' |
| 7 | tel | varchar(11) | N | N | 低 |  | NOT NULL COMMENT '电话' |
| 8 | position | varchar(4) | Y | N | 低 | NULL | DEFAULT NULL COMMENT '职位' |
| 9 | freeze | varchar(4) | Y | N | 低 | NULL | DEFAULT NULL COMMENT '是否在职' |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

表4.7 orderes表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | orderes | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id,orderNo | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | orderNo | varchar(20) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '订单编号(当前日期时间+4位随机数)' |
| 3 | deskId | int(10) | N | N | 低 |  | NOT NULL COMMENT '餐台号，外键' |
| 4 | createtime | date | N | N | 低 |  | NOT NULL COMMENT '就餐日期时间' |
| 5 | customerId | int(10) | N | N | 低 |  | NOT NULL COMMENT '客户编号' |
| 6 | status | varchar(4) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '状态：已支付，未支付' |
| 7 | number | int(4) | N | N | 低 |  | NOT NULL COMMENT '就餐人数' |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

表4.8 orderitem表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | orderitem | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | orderId | int(10) | N | N | 低 |  | NOT NULL COMMENT '订单编号，外键' |
| 3 | dishId | int(10) | N | N | 低 |  | NOT NULL COMMENT '菜品编号，外键' |
| 4 | amount | double(4,0) | N | N | 低 |  | NOT NULL COMMENT '菜品数量' |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

表4.9 user表

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 表 名 | | user | | | | | |
| 数据库用户 | | root | | | | | |
| 主 键 | | id | | | | | |
| 其他排序字段 | | 无 | | | | | |
| 索引字段 | | 无 | | | | | |
| 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(10) | N | Y | 低 |  | NOT NULL AUTO\_INCREMENT COMMENT '序号' |
| 2 | username | varchar(20) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '用户名' |
| 3 | password | varchar(20) | N | N | 低 |  | COLLATE utf8\_czech\_ci NOT NULL COMMENT '密码' |
| sql脚本 | | wys.sql | | | | | |
| 备注 | |  | | | | | |

## 4.5 安全性设计

提示：提高软件系统的安全性应当从“管理”和“设计”两方面着手。这里仅考虑数据库的安全性设计。必须说明在数据库的设计中，将如何通过区分不同的访问者、不同的访问类型和不同的数据对象，进行分别对待而获得的数据库安全保密的设计考虑。

4.5.1 防止用户直接操作数据库的方法

（1）用户标识和鉴定

对不同类型的用户帐号设置不同的访问角色，根据业务要求对不同的访问角色设计不同的权限。

1. 存取控制

用户只能用帐号登录到系统界面，通过系统界面访问数据库，而没有其他途径操作数据库。

（3）数据加密

对数据库数据进行加密操作。

4.5.2 用户帐号密码的加密方法

对用户帐号的密码进行加密处理，确保在任何地方都不会出现密码的明文。

4.5.1 角色与权限

可以考虑：（1）角色与权限。确定每个角色对数据库表的操作权限，如创建、检索、更新、删除等。每个角色拥有刚好能够完成任务的权限，不多也不少。在应用时再为用户分配角色，则每个用户的权限等于他所兼角色的权限之和。如表5.3所示，表格仅供参考。

表5.3 角色与权限

|  |  |  |
| --- | --- | --- |
| 角色 | 可以访问的表与列 | 操作权限 |
| 角色A | emplyee | 创建、检索、更新、删除 |
| customer | 创建、检索、更新、删除 |
| user | 更新 |
| 角色B | desk | 创建、检索、更新、删除 |
| dish | 创建、检索、更新、删除 |
| order | 创建、检索、更新、删除 |

列出角色与权限管理SQL脚本代码。

SET FOREIGN\_KEY\_CHECKS=0;

-- ----------------------------

-- Table structure for `category`

-- ----------------------------

DROP TABLE IF EXISTS `category`;

CREATE TABLE `category` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`name` varchar(20) COLLATE utf8\_czech\_ci NOT NULL COMMENT '名称',

`describ` varchar(20) COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '描述',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=5 DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci ROW\_FORMAT=DYNAMIC;

-- ----------------------------

-- Records of category

-- ----------------------------

INSERT INTO `category` VALUES ('1', '西餐', '欧式餐饮');

INSERT INTO `category` VALUES ('2', '湘菜系', '湖南菜系');

INSERT INTO `category` VALUES ('3', '粤菜', '广东潮州菜');

INSERT INTO `category` VALUES ('4', '鲁菜', '齐鲁风味');

-- ----------------------------

-- Table structure for `customer`

-- ----------------------------

DROP TABLE IF EXISTS `customer`;

CREATE TABLE `customer` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`name` varchar(20) COLLATE utf8\_czech\_ci NOT NULL COMMENT '用户名',

`sex` varchar(4) COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '性别',

`company` varchar(20) COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '单位',

`tel` varchar(11) COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '电话',

`cardID` int(10) NOT NULL COMMENT '贵宾卡号',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci ROW\_FORMAT=DYNAMIC;

-- ----------------------------

-- Records of customer

-- ----------------------------

INSERT INTO `customer` VALUES ('1', '小姜', '男', '湖北轻院', '156789', '1567896');

INSERT INTO `customer` VALUES ('2', '小喻', '男', '湖北轻院', '1357890', '1357890');

INSERT INTO `customer` VALUES ('3', '小卫', '男', '湖北轻院', '1345546', '1345546');

-- ----------------------------

-- Table structure for `desk`

-- ----------------------------

DROP TABLE IF EXISTS `desk`;

CREATE TABLE `desk` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`no` varchar(8) COLLATE utf8\_czech\_ci NOT NULL COMMENT '餐台编号',

`seating` int(4) NOT NULL COMMENT '座位数',

`status` varchar(10) COLLATE utf8\_czech\_ci NOT NULL COMMENT '状态：已预定，就餐中，已结账',

PRIMARY KEY (`id`,`no`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci ROW\_FORMAT=DYNAMIC;

-- ----------------------------

-- Records of desk

-- ----------------------------

INSERT INTO `desk` VALUES ('1', '2109', '10', '就餐中');

INSERT INTO `desk` VALUES ('2', '2110', '9', '已结账');

INSERT INTO `desk` VALUES ('3', '2110', '8', '已预定');

-- ----------------------------

-- Table structure for `dish`

-- ----------------------------

DROP TABLE IF EXISTS `dish`;

CREATE TABLE `dish` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`name` varchar(20) COLLATE utf8\_czech\_ci NOT NULL COMMENT '菜品名',

`categoryId` int(10) NOT NULL COMMENT '类别编号',

`pic` varchar(20) COLLATE utf8\_czech\_ci NOT NULL COMMENT '图片',

`code` varchar(8) COLLATE utf8\_czech\_ci NOT NULL COMMENT '菜品代码',

`unit` varchar(4) COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '单位',

`price` double(6,2) DEFAULT NULL COMMENT '价格',

`status` varchar(4) COLLATE utf8\_czech\_ci DEFAULT NULL COMMENT '状态',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci ROW\_FORMAT=DYNAMIC;

-- ----------------------------

-- Records of dish

-- ----------------------------

INSERT INTO `dish` VALUES ('1', '2', '1', '无', 'np', '份', '45.00', '在售');

INSERT INTO `dish` VALUES ('2', '3', '2', '无', 'hsr', '盘', '15.00', '在售');

INSERT INTO `dish` VALUES ('3', '4', '3', '无', '003', '盘', '28.00', '在售');

-- ----------------------------

-- Table structure for `employee`

-- ----------------------------

DROP TABLE IF EXISTS `employee`;

CREATE TABLE `employee` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`name` varchar(20) DEFAULT 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) DEFAULT NULL COMMENT '职位',

`freeze` varchar(4) DEFAULT NULL COMMENT '是否在职',

PRIMARY KEY (`id`)

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8;

-- ----------------------------

-- Records of employee

-- ----------------------------

INSERT INTO `employee` VALUES ('1', '小赵', '男', '2019-12-11 14:21:43', '123456', '北京市朝阳区', '13637379586', '员工', '是');

INSERT INTO `employee` VALUES ('2', '小钱', '男', '2019-12-22 14:23:06', '123456', '北京市朝阳区', '13637379587', '员工', '是');

INSERT INTO `employee` VALUES ('3', '小孙', '男', '2019-12-08 14:24:09', '123456', '北京市朝阳区', '13637379588', '组长', '是');

-- ----------------------------

-- Table structure for `orderes`

-- ----------------------------

DROP TABLE IF EXISTS `orderes`;

CREATE TABLE `orderes` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`orderNo` varchar(20) COLLATE utf8\_czech\_ci NOT NULL COMMENT '订单编号(当前日期时间+4位随机数)',

`deskId` int(10) NOT NULL COMMENT '餐台号，外键',

`createtime` date NOT NULL COMMENT '就餐日期时间',

`money` double(6,0) NOT NULL COMMENT '金额',

`customerId` int(10) NOT NULL COMMENT '客户编号',

`status` varchar(4) COLLATE utf8\_czech\_ci NOT NULL COMMENT '状态：已支付，未支付',

`number` int(4) NOT NULL COMMENT '就餐人数',

PRIMARY KEY (`id`,`orderNo`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci ROW\_FORMAT=DYNAMIC;

-- ----------------------------

-- Records of orderes

-- ----------------------------

INSERT INTO `orderes` VALUES ('1', '213213', '1', '2019-12-17', '455', '1', '已支付', '3');

INSERT INTO `orderes` VALUES ('2', '213214', '2', '2019-12-16', '566', '2', '已支付', '2');

INSERT INTO `orderes` VALUES ('3', '213215', '3', '2019-12-07', '365', '3', '未支付', '4');

-- ----------------------------

-- Table structure for `orderitem`

-- ----------------------------

DROP TABLE IF EXISTS `orderitem`;

CREATE TABLE `orderitem` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`orderId` int(10) NOT NULL COMMENT '订单编号，外键',

`dishId` int(10) NOT NULL COMMENT '菜品编号，外键',

`amount` double(4,0) NOT NULL COMMENT '菜品数量',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=4 DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci ROW\_FORMAT=DYNAMIC;

-- ----------------------------

-- Records of orderitem

-- ----------------------------

INSERT INTO `orderitem` VALUES ('1', '213213', '1', '4');

INSERT INTO `orderitem` VALUES ('2', '213214', '2', '5');

INSERT INTO `orderitem` VALUES ('3', '213215', '3', '6');

-- ----------------------------

-- Table structure for `user`

-- ----------------------------

DROP TABLE IF EXISTS `user`;

CREATE TABLE `user` (

`id` int(10) NOT NULL AUTO\_INCREMENT COMMENT '序号',

`username` varchar(20) COLLATE utf8\_czech\_ci NOT NULL COMMENT '用户名',

`password` varchar(20) COLLATE utf8\_czech\_ci NOT NULL COMMENT '密码',

PRIMARY KEY (`id`) USING BTREE

) ENGINE=InnoDB AUTO\_INCREMENT=2000514 DEFAULT CHARSET=utf8 COLLATE=utf8\_czech\_ci ROW\_FORMAT=DYNAMIC;

-- ----------------------------

-- Records of user

-- ----------------------------

INSERT INTO `user` VALUES ('2000511', '小吴', '123456');

INSERT INTO `user` VALUES ('2000512', '小王', '123456');

INSERT INTO `user` VALUES ('2000513', '小邓', '123456');

# 5 测试用例设计

对照需求中功能，写出功能的测试用例和测试结果。应考虑边界测试、默认测试、空测试、非法值测试。每个用例应说明下列内容：

(1)测试用例名称:登录-注册测试用例

(2)测试项：规定并简要说明本测试用例所要涉及的项和特性、对于每一项、可考虑引用需求说明书和设计说明书。

(3)输入说明：规定执行测试用例所需的各个输入。有些输入可以用值（允许适当的误差）来规定。而另一些输入，如常数表或事务文件可以用名来规定。规定所有合适的数据库、文件、终端信息、内存常驻区域和由操作系统传送的值。规定各输入间所需的所有关系（如时序关系等）。

(4)输出说明：规定测试项的所有输出和特性（如：响应时间）。提供各个输出或特性的正确值（在适当的误差范围内）。

(5)环境要求：规定执行本测试用例所需的硬件特征和配置、系统软件和应用软件。

(6)规程说明：描述对执行本测试用例的测试规程的一切特殊限制。这些限制可以包括特定的准备、操作人员干预、确定特殊的输出和清除过程。

(7)用例间的依赖关系：列出必须在本测试用例之前执行的测试用例名称，归纳依赖性质。建议采用表格描述每个用例，示例见表5.1。

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 | admin1，123456 | 登录失败 | 通过 |
| yl-004 | admin,空 | 登录失败 | 通过 |
| yl-005 | 空,123456 | 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，123456 | 登录失败 | 通过 |
| yl-008 | admin，密码长度超过20 | 登录失败 | 通过 |
| yl-009 | 用户名超过20，123456 | 登录失败 | 通过 |
| 注册 | yl-010 | 用户名，密码. | Tony,123456. | 创建成功 | 通过 |
|  |  |  |  |  |  |

表5.2 系统管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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 |
| 员工管理 | yl2-001 | 点击添加按钮，输入员工信息 | 小红，女，1999-05-20 13:14:00，123456，长沙市开福区，13537379580，员工，是 | 添加成功 | 通过 |
| yl2-002 | 小红，1234567 | 添加失败 | 通过 |
| yl2-003 | 小红，女，1999-05-20 13:14:00，123456，长沙市开福区，空，员工，是 | 添加失败 | 通过 |
| yl2-004 | 点击删除按钮 | 选中要删除的行 | 删除成功 | 通过 |
| yl2-005 | 未选中要删除的行 | 删除失败 | 通过 |
| yl2-007 | 点击保存按钮 | 选中要保存的行 | 保存成功 | 通过 |
| yl2-008 | 未选中要保存的行 | 保存失败 | 通过 |
| 客户管理 | yl2-009 | 点击添加按钮，输入顾客信息 | 小月，女，长沙学院，6666666，1999999 | 添加成功 | 通过 |
| yl2-010 | 小月，女，长沙学院，6666666，19999999999999 | 添加失败 | 通过 |
| yl2-011 | 小月，女，空 | 添加失败 | 通过 |
| yl2-012 | 点击删除按钮 | 选中要删除的行 | 删除成功 | 通过 |
| yl2-013 | 未选中要删除的行 | 删除失败 | 通过 |
| yl2-014 | 点击保存按钮 | 选中要保存的行 | 保存成功 | 通过 |
| yl2-015 | 未选中要保存的行 | 保存失败 | 通过 |

表5.3 业务管理测试用例

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 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 |
| 开台管理 | yl3-001 | 点击开台按钮，输入顾客，餐台，就餐人数 | 小月，2109，8 | 开台成功 | 通过 |
| yl3-002 | 小月，2109，0 | 开台失败 | 通过 |
| yl3-003 | 小月，2109，空 | 开台失败 | 通过 |
| yl3-004 | 点击查询按钮，选择餐台号 | 选中要查询的行 | 查询成功 | 通过 |
| yl3-005 | 未选中要查询的行 | 查询失败 | 通过 |
| 点菜管理 | yl3-006 | 点击添加按钮，输入菜品代码，数量 | np，1 | 添加成功 | 通过 |
| yl3-007 | np，0 | 添加失败 | 通过 |
| yl3-008 | np，空 | 添加失败 | 通过 |
| yl3-009 | 点击删除按钮 | 选中要删除的行 | 删除成功 | 通过 |
| yl3-010 | 未选中要删除的行 | 删除失败 | 通过 |
| yl3-011 | 点击保存按钮 | 选中要保存的行 | 保存成功 | 通过 |
| yl3-012 | 未选中要保存的行 | 保存失败 | 通过 |
|  |  |  |  |  |  |

# 附录

## 附录1 程序运行结果



图1 登录界面

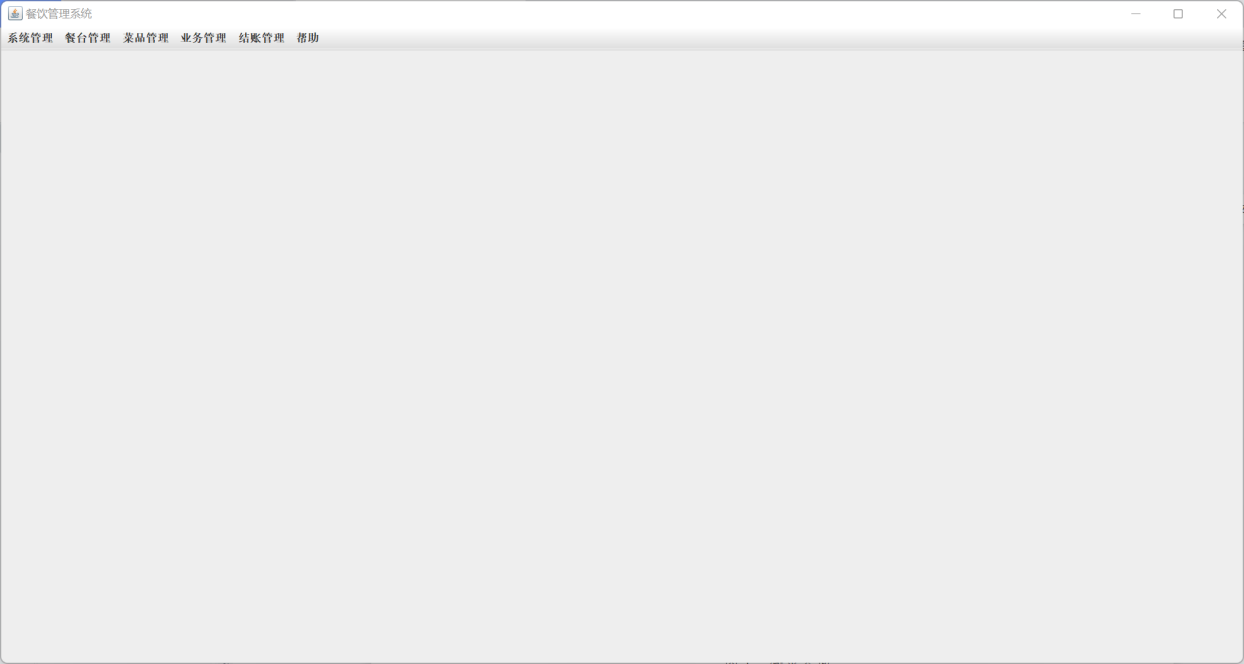


图2 主界面

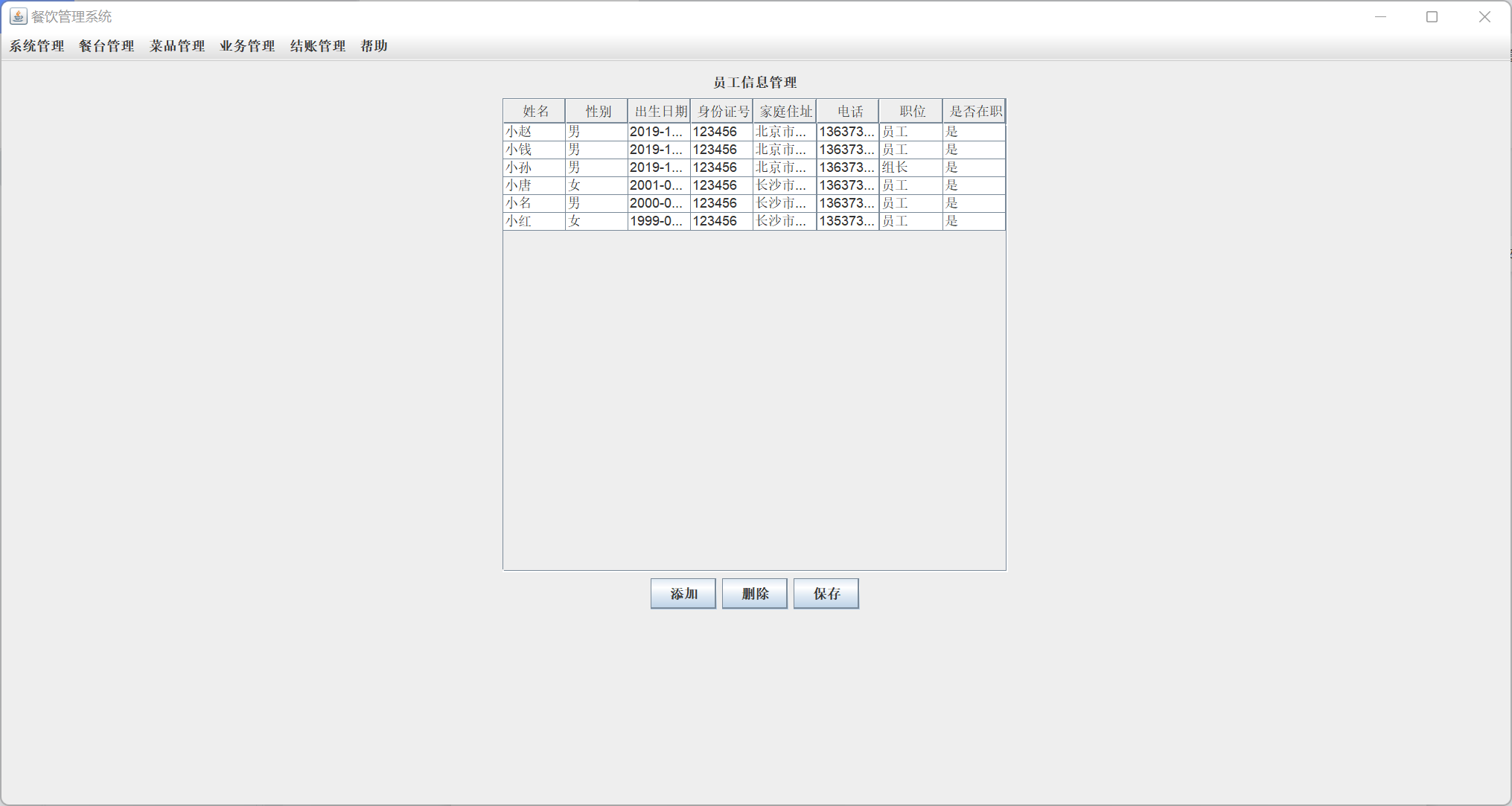


图3 员工管理界面



图4 客户管理界面

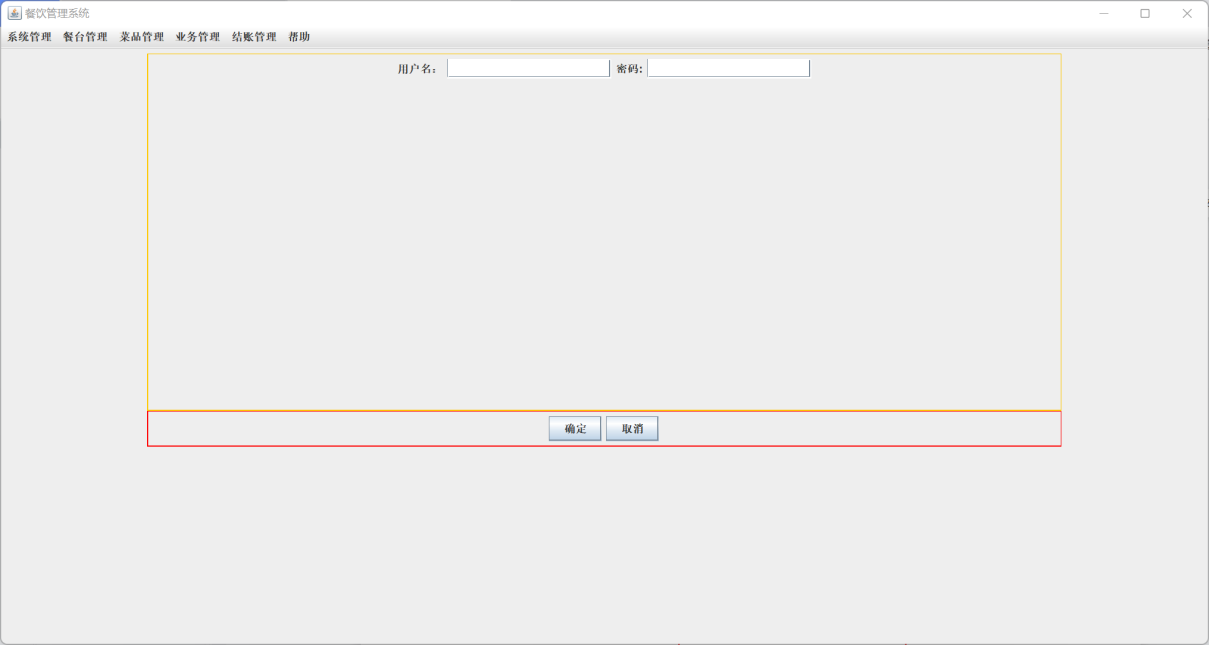


图5 修改密码界面

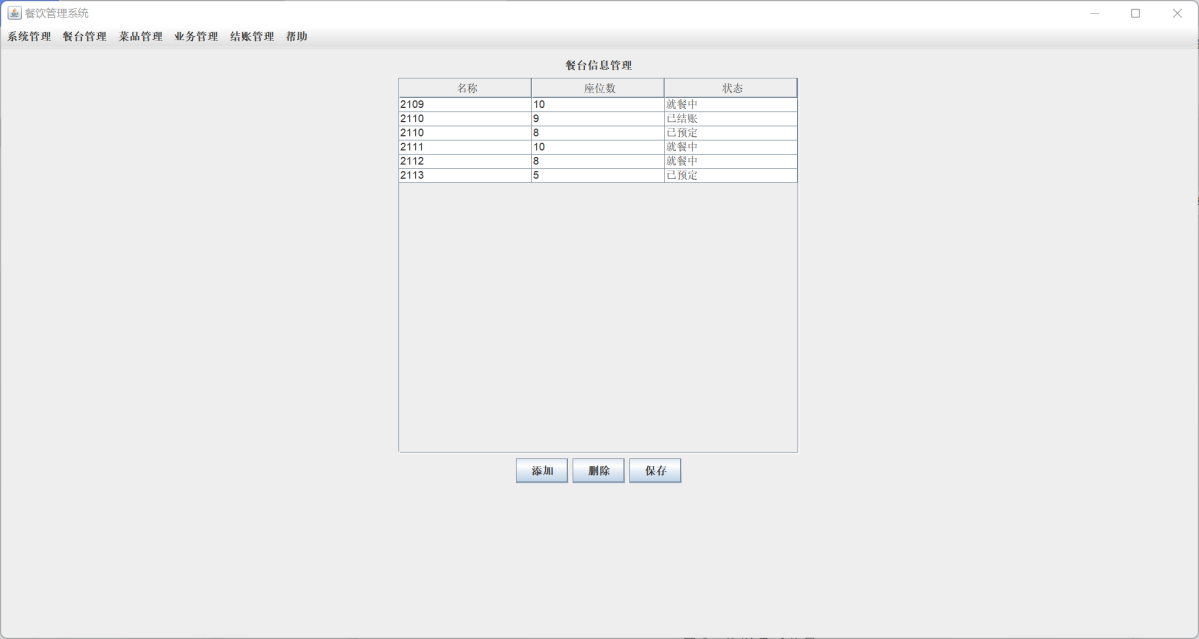


图6 餐台管理界面

## 

图7 分类管理界面

## 

图8 菜品管理界面

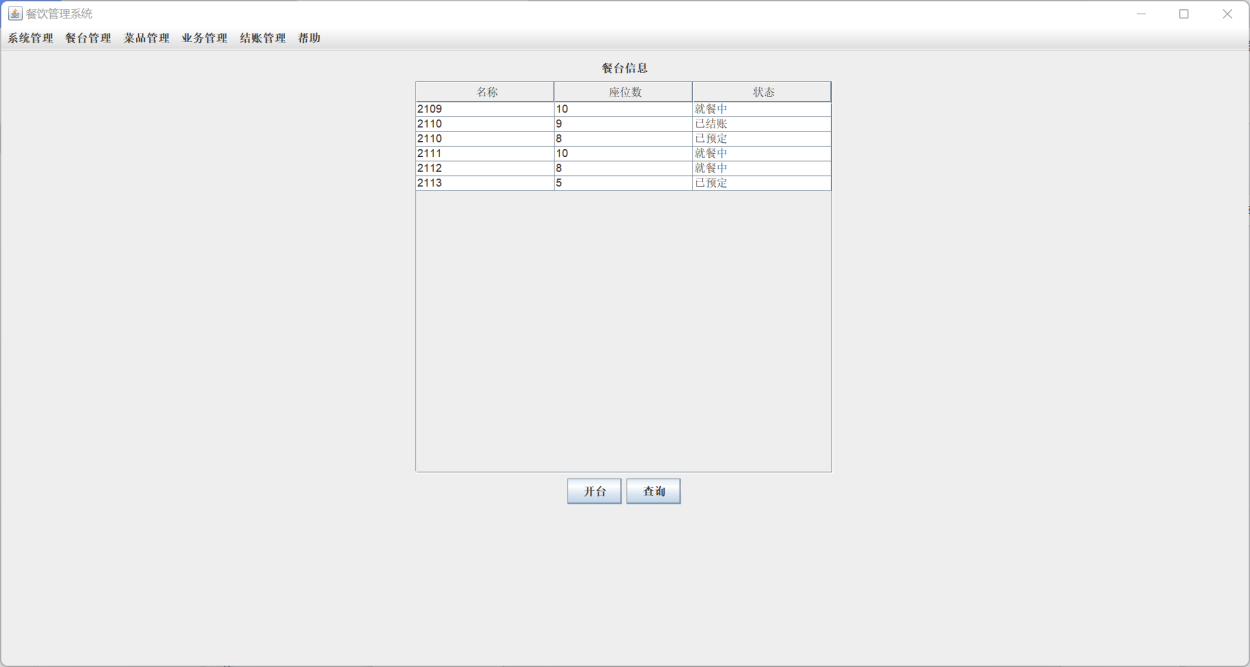


图9 开台管理界面

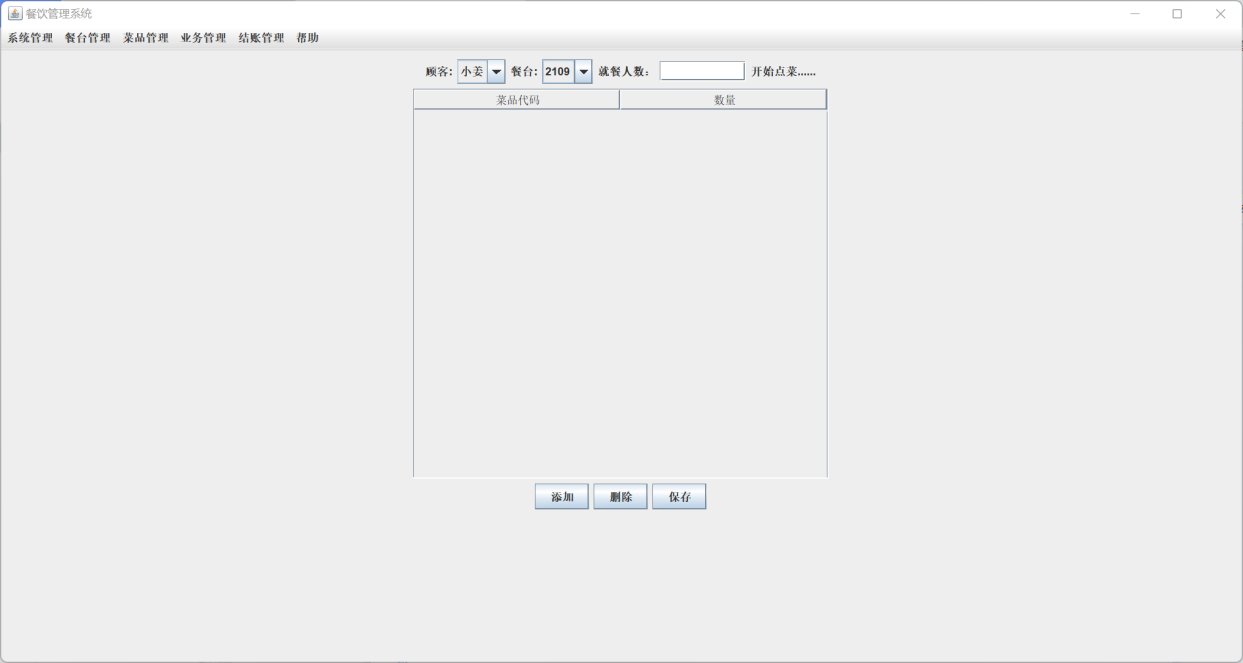


图10 点菜管理界面

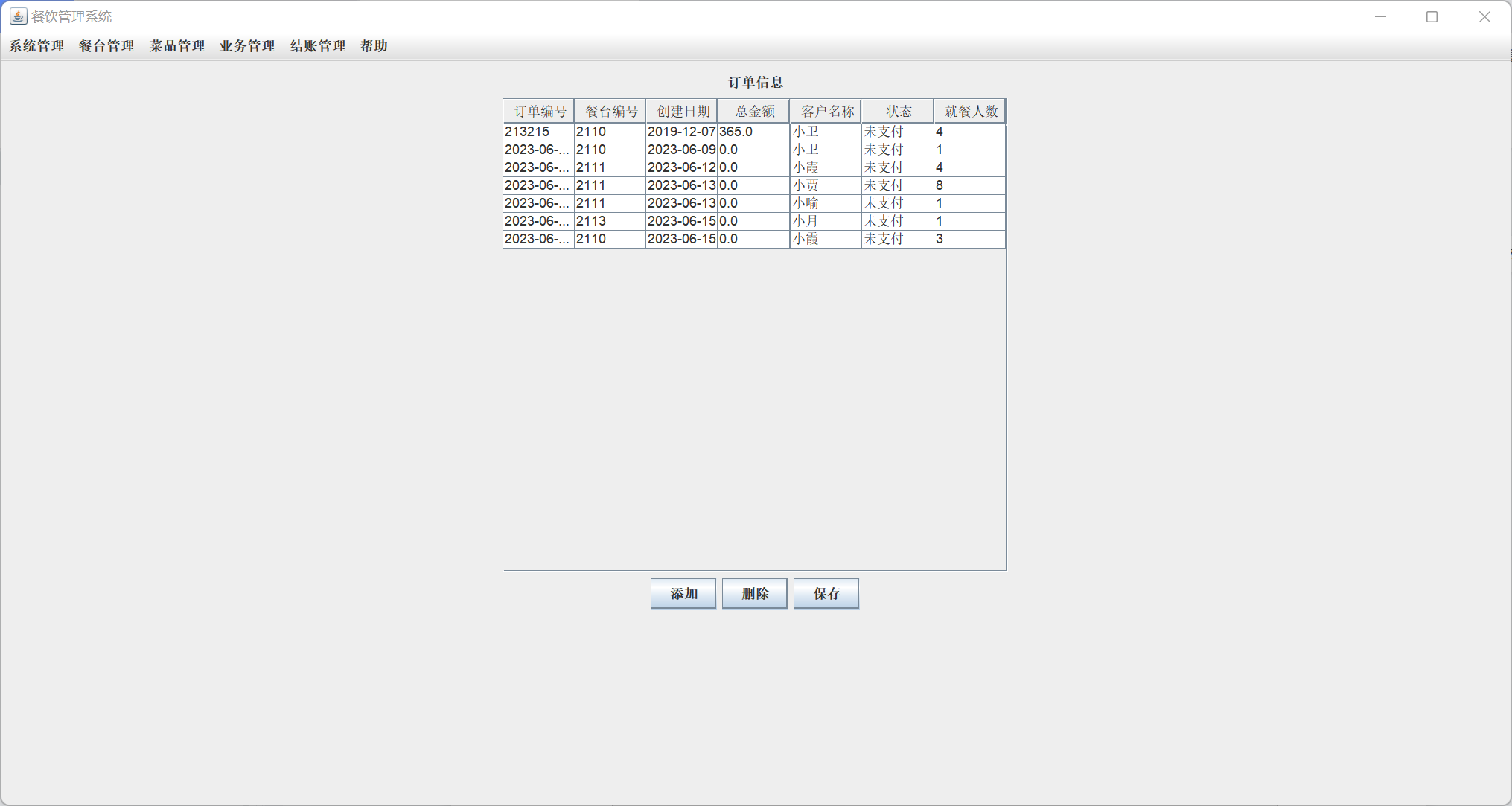


图11 订单信息界面

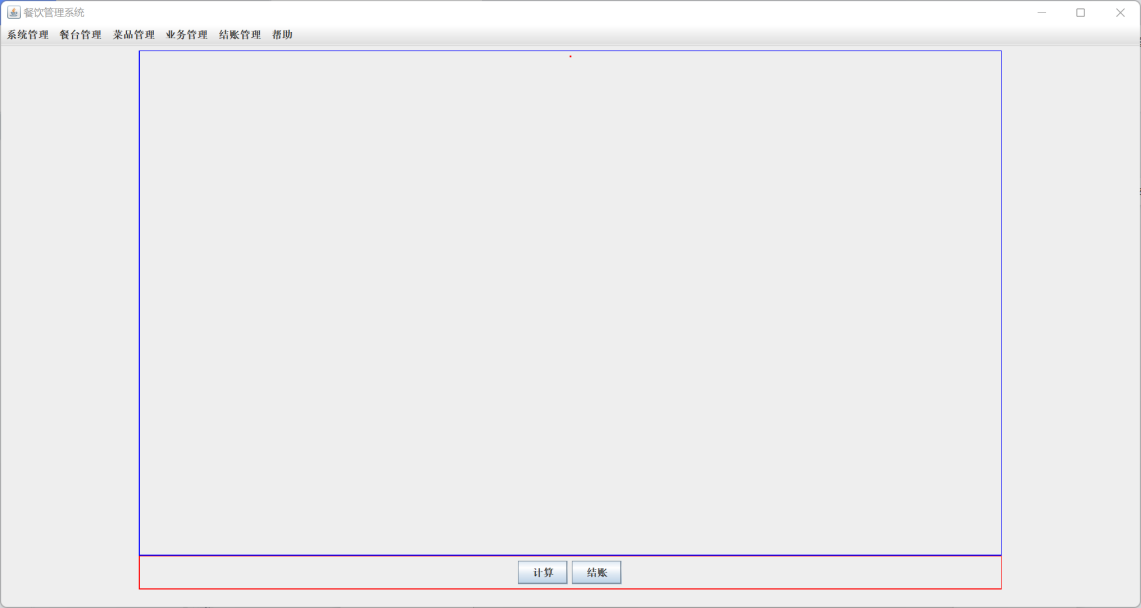


图12 金额计算界面

## 附录2 程序源代码

列出自己设计的主要代码

1.package SX;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JFrame;

import javax.swing.\*;

public class MainFrame extends JFrame{

private static final long serialVersionUID = 1L;

JPanel panel;

JMenuBar bar;

JMenu systemMenu,deskMenu,dishesMenu,orderMenu,checkoutMenu,helpMenu;

JMenuItem emplManage,custManage, pwdManage,categoryManage,dishesManage,aboutmeManage,orderManage,checkoutManage,orderItemManage,amountcalculation,deskManage;

FlowLayout layout = new FlowLayout();

EmployeeManagePane2 emplManaPane2;

//CustomerManagePane custManaPane;

CustomerManagePane2 custManaPane2;

PasswordManagePane pwdManaPane;

DeskManagePane deskManaPane;

DeskManagePane2 deskManaPane2;

CategoryManagePane2 cateManaPane2;

DishesManagePane dishManaPane;

DishManagePane2 dishManaPane2;

OrderManagePane orderManaPane;

OrderesManagePane2 orderesManaPane2;

DishesOrderesManagePane2 DishesOrderesManaPane2;

DeskManagePane3 deskManaPane3;

OrderItemManagePane orderItemManaPane;

CheckoutManagePane checkoutManaPane;

CheckoutFrame checkoutframe;

AboutmeManagePane aboutmeManaPane;

MainFrame() {

init();

//setBounds(100, 100, 200, 200);

setSize(1366,728);

setVisible(true);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

}

public void init() {

bar = new JMenuBar();

systemMenu = new JMenu("系统管理");

deskMenu=new JMenu("餐台管理");

dishesMenu=new JMenu("菜品管理");

orderMenu = new JMenu("业务管理");

checkoutMenu= new JMenu("结账管理");

helpMenu=new JMenu("帮助");

emplManage = new JMenuItem("员工管理");

custManage= new JMenuItem("客户管理");

pwdManage = new JMenuItem("修改密码");

deskManage=new JMenuItem("餐台管理");

categoryManage=new JMenuItem("分类管理");

dishesManage=new JMenuItem("菜品管理");

orderManage=new JMenuItem("开台管理");

orderItemManage=new JMenuItem("点菜管理");

checkoutManage=new JMenuItem("订单信息");

amountcalculation=new JMenuItem("金额计算");

aboutmeManage=new JMenuItem("关于我们");

bar.add(systemMenu);

bar.add(deskMenu);

bar.add(dishesMenu);

bar.add(orderMenu);

bar.add(checkoutMenu);

bar.add(helpMenu);

//系统管理菜单项

systemMenu.add(emplManage);

systemMenu.add(custManage);

systemMenu.add(pwdManage);

//餐台管理菜单项

deskMenu.add(deskManage);

//菜品管理菜单项

dishesMenu.add(categoryManage);

dishesMenu.add(dishesManage);

//业务管理菜单项

orderMenu.add(orderManage);

orderMenu.add(orderItemManage);

//结账管理菜单项

checkoutMenu.add(checkoutManage);

checkoutMenu.add(amountcalculation);

//帮助菜单项

helpMenu.add(aboutmeManage);

panel = new JPanel();

emplManaPane2=new EmployeeManagePane2();

//custManaPane=new CustomerManagePane();

custManaPane2=new CustomerManagePane2();

pwdManaPane=new PasswordManagePane();

deskManaPane=new DeskManagePane();

deskManaPane2=new DeskManagePane2();

deskManaPane3=new DeskManagePane3();

cateManaPane2=new CategoryManagePane2();

dishManaPane=new DishesManagePane();

dishManaPane2=new DishManagePane2();

orderesManaPane2=new OrderesManagePane2();

DishesOrderesManaPane2=new DishesOrderesManagePane2();

orderItemManaPane=new OrderItemManagePane();

checkoutManaPane=new CheckoutManagePane();

checkoutframe=new CheckoutFrame();

aboutmeManaPane=new AboutmeManagePane();

emplManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("ty", emplManaPane2);// 切换代码。

panel.validate();

repaint();

}

});

custManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("客户管理", custManaPane2);// 切换代码。

panel.validate();

repaint();

}

});

pwdManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("pmg",pwdManaPane);

panel.validate();

repaint();

}

});

deskManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("dmp",deskManaPane2);

panel.validate();

repaint();

}

});

categoryManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("cmp", cateManaPane2);

panel.validate();

repaint();

}

});

dishesManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent arg0) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("jk", dishManaPane2);

panel.validate();

repaint();

}

});

orderManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent arg0) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("omp", deskManaPane3);

panel.validate();

repaint();

}

});

orderItemManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent arg0) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("oimp", DishesOrderesManaPane2);

panel.validate();

repaint();

}

});

checkoutManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent arg0) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("jk", orderesManaPane2);

panel.validate();

repaint();

}

});

amountcalculation.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent arg0) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("金额计算", checkoutframe);

panel.validate();

repaint();

}

});

aboutmeManage.addActionListener(new ActionListener(){

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

panel.removeAll();

panel.add("gh", aboutmeManaPane);// 切换代码。

panel.validate();

repaint();

}

});

panel.setLayout(layout);

add(panel);

setJMenuBar(bar);

}

}

注：源代码以文件为单位附列在此。

