# “开饭了”线上点餐软件

# 系统设计说明书

姓名： 崔红梅

姓名： 包凤梅

姓名： 冯永萍

姓名： 王 芬

目录

[1.软件系统总体结构设计 1](#_Toc24871)

[1.1需求规定 1](#_Toc30677)

[1.2运行环境 1](#_Toc22135)

[1.3基本设计概念 1](#_Toc14715)

[1.4处理流程 2](#_Toc7908)

[1.5结构 3](#_Toc25019)

[2.接口设计 3](#_Toc9394)

[2.1用户接口 3](#_Toc19273)

[2.2其他软硬件接口 3](#_Toc4926)

[2.3各模块之间的接口 3](#_Toc10659)

[3.系统数据结构设计 5](#_Toc30553)

[3.1逻辑结构设计 5](#_Toc2165)

[3.2物理结构设计 7](#_Toc8091)

# 软件系统总体结构设计

## 1.1需求规定

“开饭了”线上点餐系统按需求可以分为四个模块，一是商家模块，二是用户模块，三是管理员模块；每个大模块又由具体的小模块组成。

## 1.2运行环境

1. 操作系统: Windows7/10

（2）服务器: Tomcat5.5以上

（3）数据库服务器: Oracle/MySql，能够处理数据并发访问，访问回馈时间短。

## 1.3基本设计概念

系统整体设计方案

（1）“开饭了”线上点餐系统主要特性

①用户界面的复杂度:数据静态显示/可制定视图

②用户界面的部署约束:基于独立的桌面电脑活专用工作网站⑧用户数量和类型:组织内的日常使用者，总共几百人

④系统接口类型:通过HTTP协议提供服务

⑤性能:主要是独立的数据更新，有少量并发处理

基于上述特性，我们可以判断“开饭了”线上点餐系统是属于中大型管理项目，因此我们采用SSH框架的分层架构设计方案。

（2）架构分层

“开饭了”线上点餐系统项目架构设计中，我们采用分层模式。具体说，我们将毕业论文管理系统在职责上分成3层:表示层、持久层和业务层。每个层在功能上都应该是十分明确的，而且每层要相互独立，通过一个通信接口而相互联系。

（3）模式和框架使用

在分层设计基础上，使用设计模式和框架，它们是可重用资源。

1) MCV模式

MCV模式即模型—视图—控制器模式。

①模型端

模型端是执行某些任务的代码,而这部分代码并没有任何逻辑决定用户端的表示方法。模型只是一系列的公共方法，通过这些公共方法，便可以取得模型端的所有功能。

②视图端

一个模型可以有几个视图端,而实际上多个视图端是使用MCV的原始动机。使用MCV模式可以允许多于一个的视图存在，并可以在需要的时候动态注册所需要的视图。

③控制端

视图端是与控制器结合使用的。当用户端与相应的视图发生交互时，用户可以通过视窗更新模型的状态，而这种更新是通过控制器端进行的。

2)框架

根据项目特点，使用三种开源架构:表示层、业务层、持久层。

①表示层

一个典型的Web应用的前端应该是表示层，表示层主要负责:管理用户的请求，做出相应的响应、提供一个流程控制器，委派调用业务逻辑和其它上层处理、处理异常、为显示提供一个数据模型、用户界面的验证。

②持久层

典型的Web应用的后端是持久层，持久层主要负责:如何查询对象的相关信息;如何存储、更新、删除数据记录。

③业务层

典型的Web应用中间部分应该是业务层或者服务层，业务层主要负责:处理应用程序的业务逻辑和业务校验、管理事务、提供与其它层相互作用的接口、管理业务级别的对象的依赖、在表示层和持久层之间增加了一个灵活的机制，使得他们不能直接联系在一起、通过揭示从表示层到业务层之间的上下文来得到业务逻辑、管理程序的执行。

## 1.4处理流程

1. 商家模块

商家模块的处理流程：商家权限-查看用户订单、更新菜单-确认用户订单

1. 用户模块

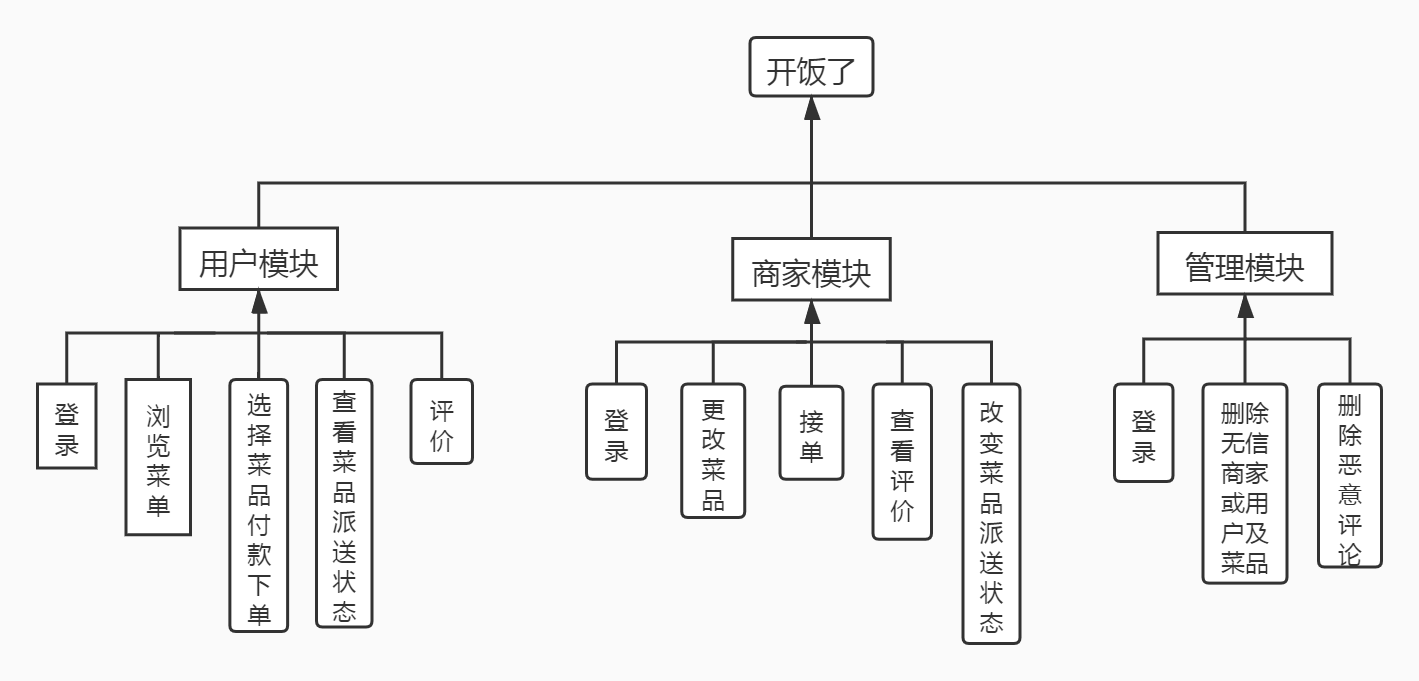
用户模块的处理流程：浏览菜单-查看评价-选择菜单-下单-查看订单；

1. 管理模块

管理模块的处理流程：基本设置-管理权限-查看商家-查看用户

## 1.5结构

下面是“开饭了”线上点餐系统的总体结构图：



（1）商家模块：登录、信息统计、菜单更新、订单统计；

（2）用户模块：订单上传、订单查看、填写评论、查看评论、删除评论；

（3）管理模块：确认供选菜单、查看商家和用户的操作日志、权限管理、模块设置。

# 2接口设计

## 2.1用户接口

用户接口以基于浏览器的图形用户界面的方式提供。本系统GUI及内部处理模块均采用JAVA实现，数据库采用orecal。

## 2.2其他软硬件接口

软件接口：本系统提供其他软件使用的函数，以达到软件的复用。

硬件接口：本次鸥汀通过JVM和底层硬件层交流。

## 2.3各模块之间的接口

本西通数据交互层通过DBConnection类和数据库进行数据交互，主要包括数据插入、删除、更新等操作。DBConnection类提供了一个接口给上层的Restaurant类及BookingSystem类，MainTable类对BookingSystem类有一个引用，每个功能模块，包括登陆、解锁、结账、点菜、添加删除订单、都有一个对 BookingSystem类的引用，即 BookingsSystem类提供了一个公共接口，并且可以说是整个系统的核心。本系统整体采用的模式为观察者模式，被观察者为 BookingSystem类，观察者为MainTable类，当 BookingSystem类有所变化时，MainTable类就会自动及时更新，并将更新后的信息.显示给用户。

# 3. 系统数据结构设计

## 3.1逻辑结构设计

表1 收货地址表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 编号 | 数据项名 | 数据项含义说明 | 数据类型 | 长度 |
| 1 | id | address表中数据项编号 | int | 10 |
| 2 | name | 用户的name | varchar | 255 |
| 3 | phonenumber | 用户的电话号码 | varchar | 255 |
| 4 | address | 用户的收货地址 | varchar | 255 |
| 5 | uid | 用户表中的编号id | int | 10 |

表2 business(商家表)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | business表中数据项编号 | int | 11 |
| 2 | name | 商家的姓名 | varchar | 255 |
| 3 | password | 商家密码，登陆商家系统 | varchar | 255 |

表3 cart(购物车表)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | cart表中数据项编号 | int | 10 |
| 2 | name | 加入购物车中的菜名 | varchar | 255 |
| 3 | price | 加入购物车中菜的价格 | float | 255 |
| 4 | num | 加入购物车中菜的数量 | int | 10 |
| 5 | food | 加入购物车中菜的图片 | varchar | 255 |

表4 evaluate(评价表)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | evaluate表中数据项编号 | int | 10 |
| 2 | uid | 用户表中的数据项编号 | int | 10 |
| 3 | sid | 商店表中的数据项编号 | int | 10 |
| 4 | context | 评价的内容 | varchar | 255 |
| 5 | grade | 评分等级 | int | 10 |

表5 foods（菜品表）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | foods表数据项编号 | int | 10 |
| 2 | name | 菜名 | varchar | 255 |
| 3 | price | 菜的价格 | float | 10 |
| 4 | food | 菜品图片 | varchar | 255 |
| 5 | sid | 店家的数据项编号 | varchar | 255 |

表6 manager(管理员)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | manager表的数据项编号 | int | 11 |
| 2 | name | 管理员名称 | varchar | 255 |
| 3 | password | 管理员设置的密码 | varchar | 255 |

表7 orders(订单表)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | orders表的数据项编号 | int | 10 |
| 2 | name | 订单的用户名 | varchar | 255 |
| 3 | price | 订单中菜品价格 | float | 10 |
| 4 | num | 菜品数量 | int | 10 |
| 5 | shopname | 商家姓名 | varchar | 255 |
| 6 | address | 订单提交的地址 | varchar | 255 |
| 7 | remarks | 订单备注（如不要香菜） | varchar | 255 |
| 8 | food | 菜品图片 | varchar | 255 |
| 9 | fid | 菜品表中的数据项编号 | int | 10 |
| 10 | uid | 用户表中数据项编号 | int | 10 |
| 11 | sid | 商店表中的数据项编号 | int | 10 |
| 12 | totalprice | 该订单总价 | varchar | 255 |
| 13 | time | 下单时间 | varchar | 255 |

表8 shop（商店）

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | shop表中数据项编号 | int | 10 |
| 2 | name | 商店姓名 | varchar | 255 |
| 3 | cover | 商店的店面图 | varchar | 255 |
| 4 | qualification | 店家的营业执照 | varchar | 255 |
| 5 | bid | 商家表中的数据项编号 | int | 10 |

表9 user(用户)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 | id | user表中数据项编号 | int | 10 |
| 2 | name | 用户姓名 | varchar | 255 |
| 3 | password | 用户设置的密码 | varchar | 255 |

## 3.2物理结构设计

本系统使用Oracle 数据库，以便满足每个数据结构中的每个数据的存储要求、访问方法、存取单位、存取的物理关系、设计考虑和保密条件。Oracle是一个多用户、多线程的数据库，是一个客户机/服务器结构的应用，它对于商业和个人用户是免费的。它的执行速度块简单有效。