

摘 要

伴随着物流的兴起，餐饮业也从实体店面转向了外卖，现在传统的订餐模式明显不能满足顾客的需要，服务员不能直接上门为顾客提供订餐服务。随着网络技术的快速发展，以及在餐饮服务方面的持续革新，外卖行业也在跟上时代步伐。因此，外卖管理系统迅速发展起来。本论文以 SpringCloud 为基础，采用 Java 语言并结合 MySQL 数据库而实现的 SpringCloud 的外卖管理系统。它包括：用户信息管理、商家信息管理、系统公告管理、配送单管理、菜品信息管理、订单信息管理、骑手信息管理等功能。经过对系统功能的测试，表明本系统操作简单，可视化非常好，没有繁琐的流程，可以让用户轻松上手。最终实现为用户和商家提供一个更加灵活、方便、安全的一个外卖平台。

关键词：外卖管理；Java 开发语言；MySQL 数据库；SpringCloud

ABSTRACT

Along with the increase in logistics, the restaurant business has moved away from the physical store to the delivery. Now the traditional ordering mode can not meet the needs of customers. With the rapid development of network technology and the continuous innovation in catering service, the take-out industry is keeping pace with the times. Therefore, the take-out management system developed rapidly. Based on Spring Cloud and using Java language, this paper designs and develops a take-out management system with MySQL database, and develops and realizes the user information management, merchant information management, system bulletin management, delivery order information management, food information management, order information management and rider information management. After the system function test, indicated this system operation is simple, the visualization is very good, does not have the tedious flow, may let the user easily start. The final implementation for users and businesses to provide a more flexible, more convenient, more secure a reservation platform.

Keywords: Food delivery management;Java; MySQL; SpringCloud

目 录

第 1 章 绪论	1
1.1 开发背景和意义	1
1.2 国内外研究现状	1
1.3 研究内容	2
第 2 章 关键技术及开发环境简介	3
2.1 Java 语言	3
2.2 SpringCloud 架构	3
2.3 MySQL 数据库	3
2.4 IDEA 简介	3
2.5 开发环境	4
第 3 章 系统分析	5
3.1 可行性分析	5
3.1.1 经济可行性	5
3.1.2 技术可行性	5
3.1.3 操作可行性	5
3.2 功能需求分析	5
第 4 章 系统设计	9
4.1 系统功能模块设计	9
4.1.1 技术架构设计	9
4.1.2 系统功能	9
4.2 数据库设计	9
4.2.1 概念结构设计	10
4.2.2 逻辑结构设计	12
4.2.3 数据库的连接	15
第 5 章 系统实现	17
5.1 注册与登录	17
5.1.1 注册	17
5.1.2 登录	18
5.2 用户信息管理	19
5.3 骑手信息管理	23
5.4 商家信息管理	24
5.5 菜品信息管理	25
5.6 订单信息管理	26

5.7	配送单信息管理	28
5.8	商品评价管理	29
5.9	我的收藏管理	30
5.10	系统公告管理	30
第6章	系统测试	32
6.1	测试目的	32
6.2	功能测试	32
6.2.1	用户信息管理	32
6.2.2	商家信息管理	33
6.2.3	系统公告管理	33
6.2.4	配送单信息管理	34
6.2.5	菜品信息管理	34
6.2.6	订单信息管理	35
6.2.7	骑手信息管理	35
6.2.8	我的收藏管理	35
6.2.9	商品评价管理	36
6.3	回归测试	36
结论	37
参考文献	38
致谢	39

第 1 章 绪论

1.1 开发背景和意义

近年来，O2O（online to offline，外卖是最典型的 O2O 模式）得到了快速的发展。许多人喜欢在家里买东西，而在这个时候，餐饮业也希望与网络合作，能够共同发展。同样，人们习惯了这种快节奏的生活方式，也喜欢在家里就能吃到美味的食物。就是因为人们的这种思维的转变，于是，便产生了一种新的餐饮形式——外卖。在我们国家，最早引进的是国外的快餐，像是肯德基，麦当劳，必胜客等等。然后就是中国人的生日蛋糕，还有现在的海底捞火锅，甚至是中式快餐，红烧大虾等等都能送到。从这一点可以看出，外卖服务已成为一个快速增长的产业。可是阿里巴巴、淘宝的商品竞争也是愈演愈烈，很多商品都已经达到了饱和状态就是这个原因造成了很多小型卖家的损失。因为商品的过量，导致了价格竞争愈演愈烈。但是，以餐饮为主导的服务行业则与此不同，由于食品的保质期、运输以及味道等问题，如今，餐饮行业开始向互联网发展。但是，在外卖行业中，目前还只是占据其中的一小部分，这一行业还亟待发展。而且还是有着很大的发展前景的一个群体，所以外卖行业也是在不断探索中发展起来的。

从最初的电话点餐，到如今的网络点餐，都得到了顾客的认可。网上点餐被公认具有以下优势：（1）容易使用；（2）节省电话费用；（3）可视化的菜单。

1.2 国内外研究现状

随着社会的快速发展，在互联网的大环境下，信息传播的速度已经不是任何线下途径可以比拟的了。它可以更快更广的让人了解产品。同样在餐饮行业也可设计外卖平台将商铺有机结合来缓解就餐困扰。在 2016 年刘冰冰和郭群的《综合型外卖网站设计与实现》中写到采用面向对象设计和分析的方法对整个系统进行研究与设计,使用 B/S 体系架构,有利于增加系统的可读性,提高系统的性能和效率^[3]。在 2018 年蔡丹的《基于 Android 的校园外卖系统设计与实现》指出外卖系统为 C/S 结构,客户端程序运行在安卓系统上,基于用户设备当前的定位信息,将周围的外卖商家的产品展示给用户^[1]。在 2018 年崔育帅和王春江的《基于 O2O 的校园“食堂外卖”微信小程序的开发及推广》中提到系统采用 C/S 模式开发,分为三层,视图层、业务逻辑层、数据层,这种结构的好处是能处理大量数据、响应速度快和交互性强等^[2]。

在 2015 年 Varsha Chavan, Priya Jadhav, Snehal Korade and Priyanka Teli 的《Implementing Customizable Online Food Ordering System Using Web-Based Application》中指出基于 Web 的应用程序实现可定制的在线食品订购系统通过应用程序为订购添加了一个模块，同时解释了高效查看实时订单的方法，引入了 SMS 系统，在注册时要求客户输入联系电话和其他信息，并将其保存在数据库中^[25]。在 2014 年 ReshamShinde, Priyanka Thakare, Neha Dhomne, Sushmita Sarkar 的《Design and Implementation of Digital dining in Restaurants using Android》中提到组织环境中构建基于网络的食物配送，Web 应用程序通过使用迭代开发方法构建，并使用 MVC 概念，该概念利用了带有 CI 框架和 MySQL 数据库的 PHP 编程语言。这个系统在订购配送食品的过程中提供了食品订单、搜索食品菜单、查看余额、回顾采购和销售交易历史以及打印销售交易历史报告等功能^[26]。

1.3 研究内容

本论文设计目的在于给商家和用户提供更好的订餐互动的方式，通过 MySQL 数据库开发了一个基于 SpringCloud 的外卖管理系统。该系统可以处理用户信息管理、商家信息管理、系统公告管理、配送单管理、菜品信息管理、订单信息管理、骑手信息管理等功能。本系统的关键任务是：

- (1) 通过不同的角度与层次判断该系统存在的意义，并收集多方的需求。
- (2) 通过对需求的分析与对本系统的认识，划分它的功能模块。
- (3) 选择相关的数据库来存储数据。
- (4) 做好系统中的每一个功能模块的设计。
- (5) 做好系统测试，发现系统中的潜在问题并加以优化。

第 2 章 关键技术及开发环境简介

2.1 Java 语言

Java 作为开发语言是静态的，相比之下它主要有以下几个优点：（1）操作简单（2）不局限于一个平台，可以实现多个不同平台之间的跨越（3）可实现多线程，即在同一个程序内可以允许多个任务同时进行等。正是因为 Java 的这些特点，所以不管是在 APP 的程序编写上，还是在页面端的游戏开发等方向，它都是目前软件开发行业中使用最多的语言。

2.2 SpringCloud 架构

SpringCloud 微服务架构是以 SpringBoot 为基础建立并运行应用软件的一种方式，它是一组技术系统和方法论的集合，同时操作简单且配置方便的一个工具开发包。但是，在进行业务更新的时候，还需要把分布式数据库的一致性列入考虑的范围之内。因为数据之间可能存在某种联系，不能轻易改变，否则就会导致其他数据可能无法正常使用。

2.3 MySQL 数据库

MySQL 是一种可以结合目前市场上使用最普遍的编程语言（如 Java，C++等）的高效率、轻量化的数据库，同时它可以在不同的平台上提供服务且具有一定的安全性。不仅如此，该数据库可实现与管理系统的链接，数据通常可以存储在多个表上，而不是将所有数据都存在同一个库中。正是它的这个特点使系统的运行速度得到了明显的提高，也使操作更加方便。MySQL 在实际的开发编程过程中得到了广泛的应用和支持主要是出于以下几个原因：（1）采用集合的操作方式；（2）具有很高的可操作性（3）实现了一个具有多种用法的结构化语法，因此它是一种操作方便、容易被人接受的语言。

2.4 IDEA 简介

这是目前大部分编程人员喜欢使用的一个开发的软件。因为这个软件相比较来说，它所拥有的功能更加全面，操作起来十分简单且容易上手，特别值得提出的就是这个软件具有代码的自动查错和提示等功能，在一定程度上为工作人员减轻了很多工作的压力，

同时，也加快了开发速度和项目进程。此外，软件的界面也是简洁优美，给人一种身心愉悦的感受。综上所述，它成为当前使用率最早的开发软件。

2.5 开发环境

开发环境见下表所示。

表 2.1 开发环境介绍表

环境类型	使用名称
操作系统	Microsoft Window10
编程语言	Java
编译器	IntelliJ IDEA 2020.3.3 x64
服务器	Apache-Tomcat-9.0.451

第 3 章 系统分析

3.1 可行性分析

3.1.1 经济可行性

由于互联网和科技的持续发展，我们可以快速的设计和开发新系统。这是因为支持系统的软件可以直接加载和使用。因此，可以在相对较短的时间内满足用户的所有要求。所以，该系统的初始成本非常低，但其使用却显著提高了消费者的效率。

3.1.2 技术可行性

技术可行性研究旨在研究开发该系统所需的架构，软件和设备配置环境水平等方面。系统中使用的配置环境很简单，只需要一台带有 Win7 及以上系统的计算机即可完全执行系统，而 Tomcat 服务器则擅长一次更新所有数据，使用 IDEA 工具编写更容易。以上都表明该系统在技术上是实现的。

3.1.3 操作可行性

本系统使用方便，大多数的资料都是以下拉式的形式输入，有些页面甚至可以不需人工输入就能自动生成资料，对于操作者来说简单，方便，灵活。所以，在操作方面也是可以实现。

3.2 功能需求分析

本系统设计并开发了外卖管理系统，商家可以使用此系统添加或删除自己的菜品，用户也可以通过此系统来订购外卖，骑手可以系统来配送外卖。本系统中所有角色及操作介绍如下：

1. 用户

- (1) 注册登录：可以提供用户在本系统中添加自己的账号并且使用。
- (2) 个人中心：用户可以在这个模块中对自己账号的信息进行更改。
- (3) 订单管理：用户可以在此查看自己下单的情况。
- (4) 商品评价管理：用户可以表达自己对商家的商品体验感。

-
- (5) 配送单管理：用户可以查看自己下单的订单状态。
 - (6) 我的收藏管理。

用户用例图如图 3.1 所示。

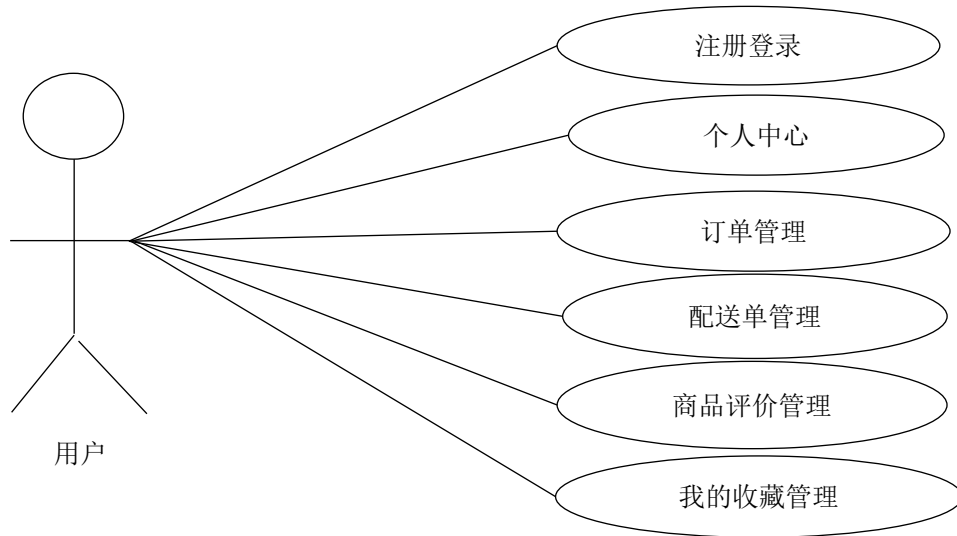


图 3.1 用户用例图

2. 商家

- (1) 登录：可以提供商家在本系统中登录并使用账号。
- (2) 个人信息：商家可以在这个模块中对自己账号的信息进行编辑。
- (3) 订单信息管理：商家可以在此查看用户对本店商品下单的情况。
- (4) 菜品信息管理：商家在此可以对本店的菜品进行编辑。
- (5) 商品评价管理：可以查看用户对商家的商品的体验感和意见等。
- (6) 配送单管理。

商家用例图如图 3.2 所示。

3. 管理员

- (1) 登录页面：管理员可以进入到后台系统。
- (2) 个人中心：管理员可以在这个模块中对自己账号的信息进行编辑。
- (3) 用户信息管理：可以查看和维护所有用户的个人信息情况。
- (4) 商家信息管理：可以为商家建立自己的账号，也可对商家信息查看和维护。
- (5) 系统公告管理：该模块用来展示餐饮行业相关的最新新闻资讯等内容。
- (6) 菜品信息管理：管理员可以查看商家发布的菜品信息。
- (7) 骑手信息管理：可以对骑手信息查看和维护，也可帮助骑手建立自己的账号。
- (8) 配送单管理：主要是用来查看和监督用户订单的状态。
- (9) 商品评价管理：可以查看用户对商家的商品的体验感和意见等。

管理员用例图如图 3.3 所示。

4. 骑手

- (1) 注册登录：可以提供骑手在本系统中添加自己的账号并且使用。
- (2) 个人信息：骑手可以在这个模块中对自己账号的信息进行编辑。
- (3) 订单信息管理：骑手可以在此查看用户对商家商品下单的情况。
- (4) 商品评价管理：可以查看用户对商家的商品的体验感和意见等。
- (5) 配送单管理。

骑手用例图如图 3.4 所示。

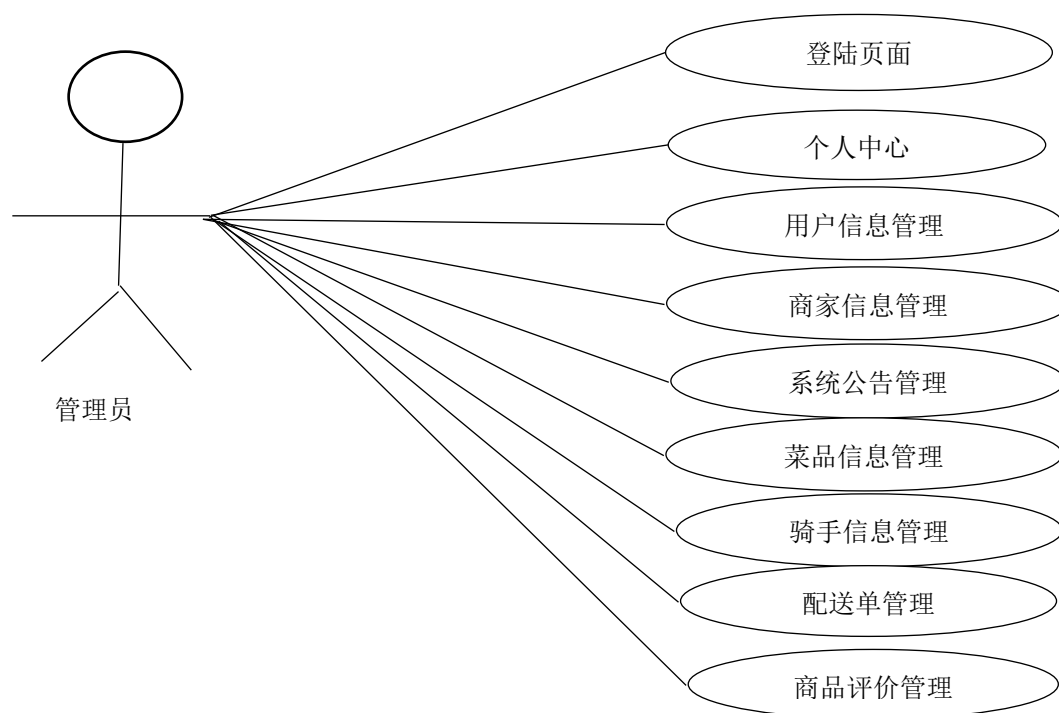


图 3.3 管理员用例图

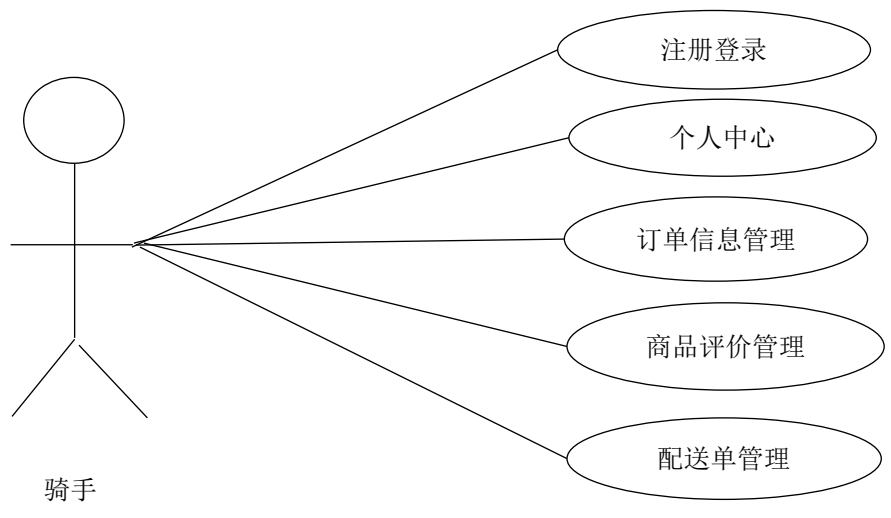


图 3.4 骑手用例图

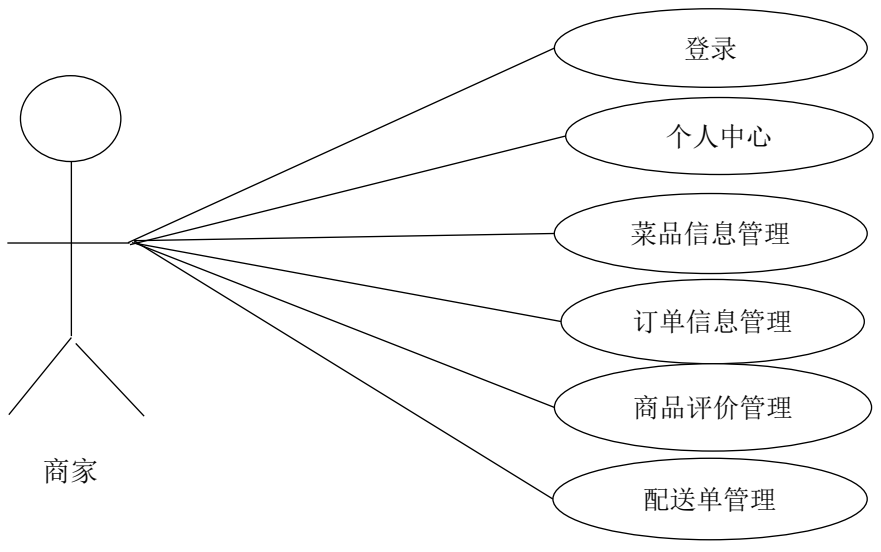


图 3.2 商家用例图

第 4 章 系统设计

4.1 系统功能模块设计

4.1.1 技术架构设计

Spring Cloud 微服务体系结构，是对 SOA 体系结构的一种改良和扩展，其目的是将业务分解，并使服务可以独立运作。登录注册的服务组件为 Spring Cloud Eureka，通过注册服务进行注册后，能够实现单点和高可用注册中心的效果。网关服务的服务组件为 Spring Cloud Zuul，主要包含两方面的功能：一是实现路由规则和服务实例的维护；二是通过过滤器实现登录校验^[4]。Spring Cloud Config 是这个分布式配置的一个服务组件，这个组件也恰好为统一管理、动态调整和自动更起到了很好的促进作用。其中，SpringCloud 总线是一个服务部件，它通过与 RabbitMQ 的集成，使消息能够及时更新。资料资讯服务透过 MySQL 资料库，将资料资讯处理，以确保使用者资料的安全性及资料的完整性。

4.1.2 系统功能

在该系统设计之初，通过分析使用该系统的人员的多边需求，以及由于连接方式不同而导致的不同连接访问权限，共定义了四种连接方式。连接有四种方式：第一种是用户端，第二种是商家端，第三种是骑手端，第四种是管理员端。该系统模块的方案如图 4.1 所示。

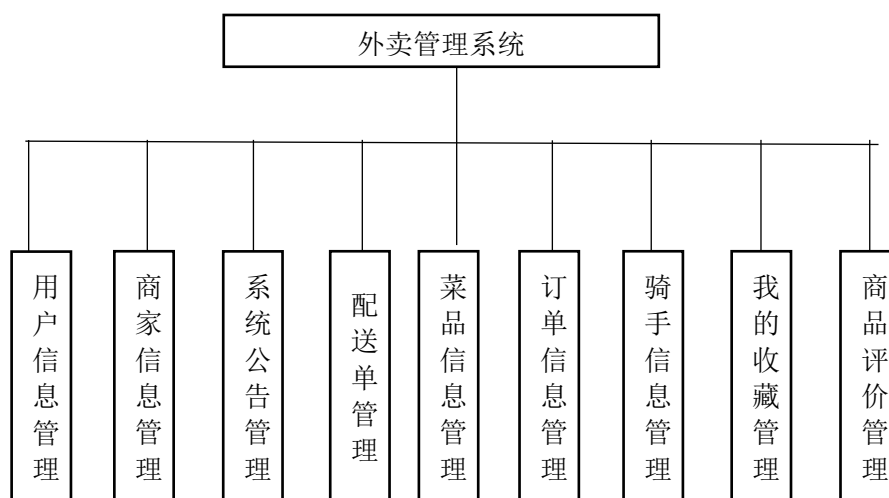


图 4.1 系统功能模块图

4.2 数据库设计

由于概念模型的标准无法插入到特定的数据库存储中，因此需要使用系统的特定功能来实现它们。从计算机的角度来看，应遵守以下将 E-R 模型转换为逻辑关系的原则：

- (1) 每一个实体要转换成一个关系。
- (2) 所有的主键必须定义非空 (NOT NULL)。
- (3) 用一对多、弱对实、一对一、多对多等联系来定义二元关系中的外键。

4.2.1 概念结构设计

该系统能正常运行的前提是有足够完整的数据信息，这就必须要做到必须填写的内容中每一项都有数据，否则将会无法访问。同时，我们需要确保相关板块之间的数据准确无误，不存在数据冲突。还要注意要使用正确的数据库格式存储，不能在格式规范上出现问题。各关系如下表：

表 4.1 关系表

功能模块	主码	外码
用户信息	用户账号	姓名、密码、性别、手机号、地址、编号
商家信息	商家账号	姓名、密码、性别、手机号、地址、编号
系统公告	公告编号	公告名称、公告内容
菜品信息	菜品编号	名称、单价、类型、图片、商家编号、商家名称、商家地址
订单信息	订单编号	菜品名称、单价、数量、类型、总价、配送费、下单时间、商家编号、商家名称、用户账号、用户手机号
骑手信息	骑手账号	姓名、密码、性别、手机号、地址
我的收藏	收藏编号	商家名称、菜品名称
商品评价	订单编号	菜品名称、数量、商家编号、用户账号、菜品评分、配送评分、商家评分、评价内容
配送单信息	订单编号	菜品名称、图片、数量、配送费、商家编号、用户账号、骑手账号、骑手姓名、地址、下单时间、骑手

使用 E-R 图解的方法可以通过概念化对数据化进行设计。因此，该系统可以分解并设计成各种 E-R 方案，包括全部图和局部图。先将局部图设计出来再整合成一个全部图，再把定义模型转化成概念模型。

(1) 用户信息实体属性见图 4.2 所示。

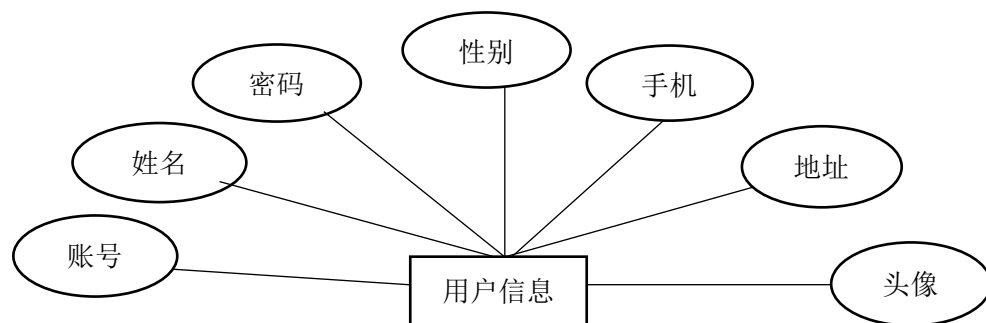


图 4.2 用户信息实体属性图

(2) 商家信息实体属性见图 4.3 所示。

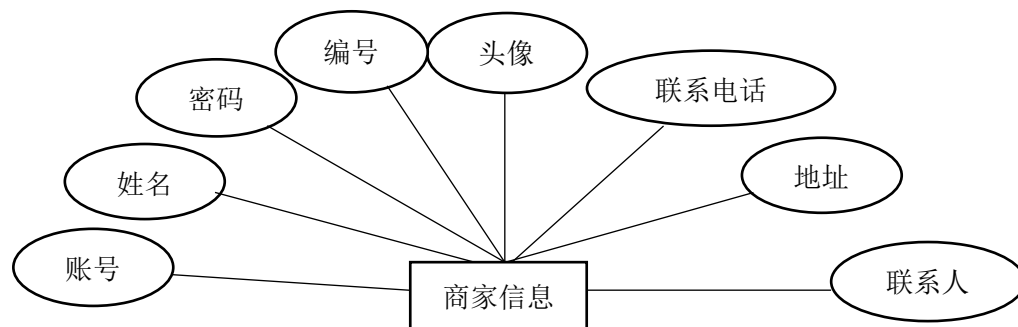


图 4.3 商家信息实体属性图

(3) 订单信息实体属性见图 4.4 所示。

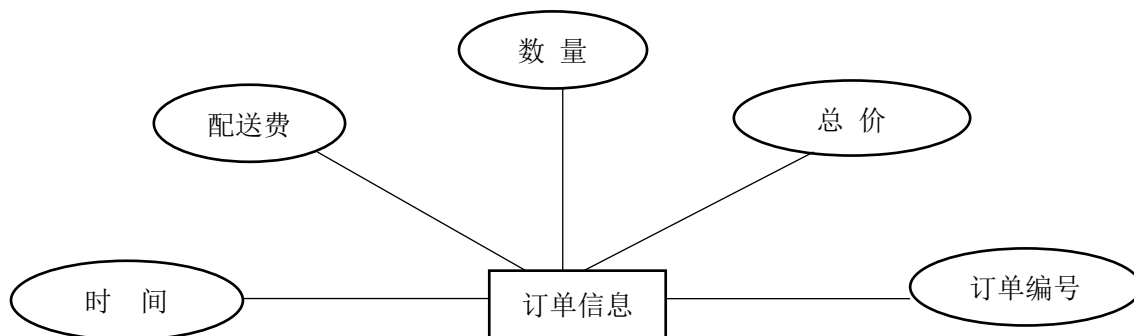


图 4.4 订单信息实体属性图

(4) 核心设计 E-R 图见图 4.5 所示。

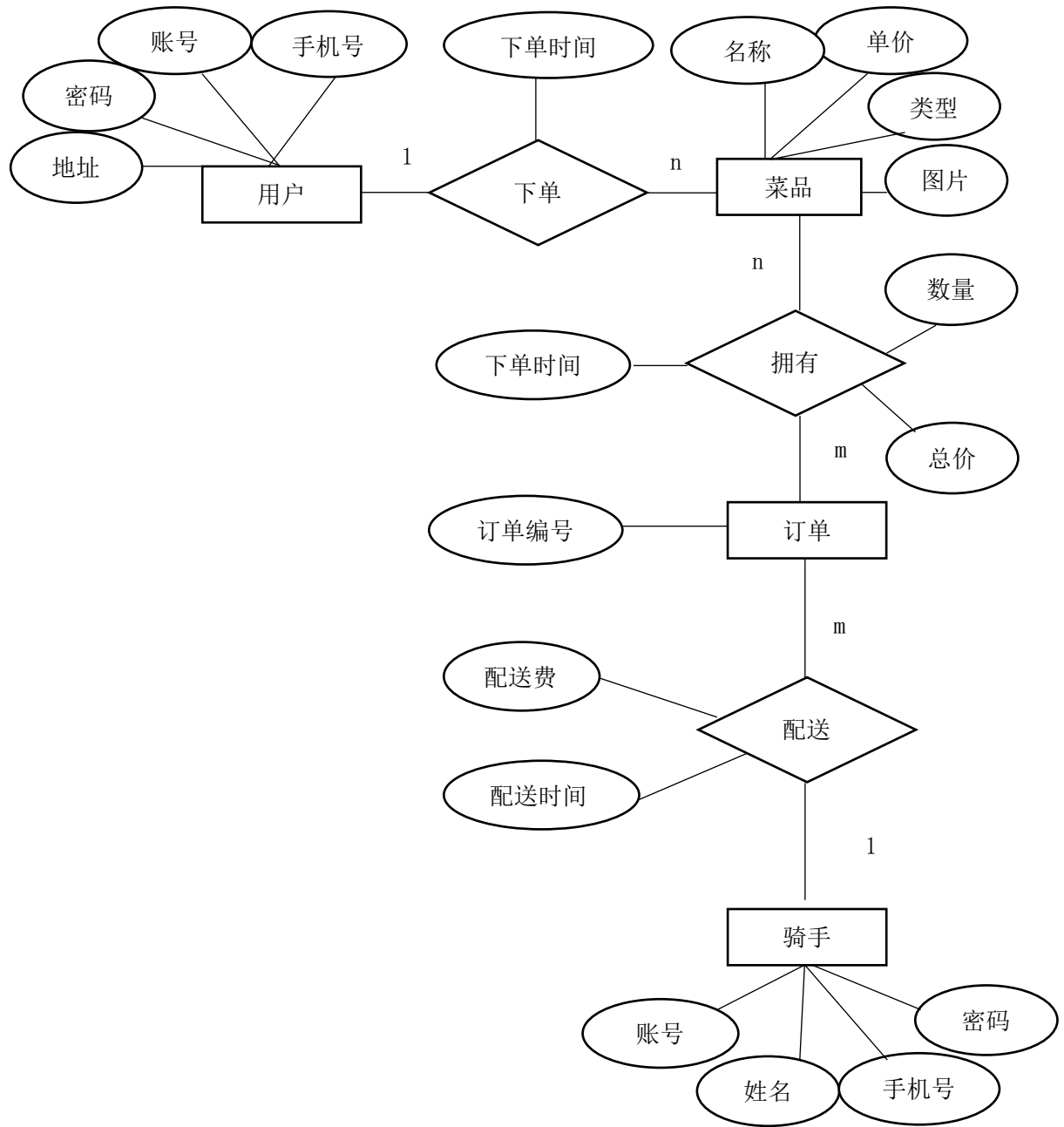


图 4.5 核心 E-R 图

4.2.2 逻辑结构设计

在功能需求的基础上，给本系统建立了一个专门的数据库，同时，提出了以下数据结构。

(1) 用户信息表：表结构如表 4.1 所示。

表 4.1 用户信息表

字段类型	字段名称	字段说明
Bigint	Id	用户账号
varchar(50)	name	姓名
varchar(100)	password	密码
varchar(50)	sex	性别
varchar(100)	phone	手机号
varchar(255)	address	地址
varchar(225)	touxiang	头像

(2) 商家信息表：表结构如表 4.2 所示。

表 4.2 商家信息表

字段类型	字段名称	字段说明
Bigint	id	商家账号
varchar(50)	shangjianame	名称
varchar(100)	password	密码
varchar(255)	touxiang	头像
varchar(100)	phone	联系电话
varchar(255)	address	地址
varchar(50)	bianhao	编号
varchar(50)	name	联系人

(3) 公告信息表：表结构显示在表 4.3 中。

表 4.3 公告信息表

字段类型	字段名称	字段说明
Bigint	Id	公告编号
varchar(50)	Name	公告名称
Longtext	Content	公告内容

(4) 配送单信息表：表结构如表 4.4 所示。

表 4.4 配送单信息表

字段类型	字段名称	字段说明
Bigint	id	订单编号
varchar(50)	name	菜品名称
varchar(255)	tupian	图片
varchar(50)	foodnum	数量

varchar(50)	bianhao	商家编号
varchar(50)	yonghuid	用户账号
varchar(50)	peisongfei	配送费
varchar(50)	qishouid	骑手账号
varchar(50)	qishouname	骑手姓名
varchar(50)	address	地址
varchar(50)	time	下单时间
varchar(100)	phone	骑手手机号

(5) 菜品信息表：表结构如表 4.5 所示。

表 4.5 菜品信息表

字段类型	字段名称	字段说明
Bigint	id	菜品编号
varchar(50)	name	名称
varchar(255)	tupian	图片
varchar(50)	foodmoney	单价
varchar(50)	foodstyle	类型
varchar(50)	bianhao	商家编号
varchar(50)	shangjianame	商家名称
varchar(255)	address	商家地址

(6) 订单信息表：表结构如表 4.6 所示。

表 4.6 订单信息表

字段类型	字段名称	字段说明
Bigint	id	订单编号
varchar(50)	name	名称
varchar(50)	foodmoney	单价
varchar(50)	foodnum	数量
varchar(50)	account	总价
varchar(50)	shangjiaid	商家编号
varchar(50)	shangjianame	商家名称
varchar(50)	yonghuid	用户账号
varchar(100)	phone	用户手机号
varchar(50)	distribution	配送费
varchar(50)	foodstyle	类型
varchar(50)	time	下单时间

(7) 骑手信息表：表结构如表 4.7 所示。

表 4.7 骑手信息表

字段类型	字段名称	字段说明
Bigint	id	骑手账号
varchar(50)	name	姓名
varchar(100)	password	密码
varchar(50)	sex	性别
varchar(100)	phone	手机号
varchar(255)	address	地址

(8) 我的收藏信息表：表结构如表 4.8 所示。

表 4.8 我的收藏信息表

字段类型	字段名称	字段说明
varchar(20)	code	收藏编号
varchar(50)	caipinname	菜品名称
varchar(50)	shangjianame	商家名称

(9) 商品评价信息表：表结构如表 4.9 所示。

表 4.9 商品评价信息表

字段类型	字段名称	字段说明
Bigint	dingdanid	订单编号
varchar(50)	caipinname	菜品名称
varchar(50)	foodnum	数量
varchar(50)	shangjiabianhao	商家编号
varchar(50)	yonghuid	用户账号
varchar(50)	caipinpingfen	菜品评分
varchar(50)	peisongpingfen	配送评分
varchar(50)	Shangjiapingfen	商家评分
Longtext	content	评价内容

4.2.3 数据库的连接

本系统将数据库与 Mybatis 框架相连，并通过 Mybatis 框架来管理数据库，此处仅给出 Dao 层的一段代码，其代码如下：

```
import com.entity.YonghuEntity;
import com.baomidou.mybatisplus.mapper.BaseMapper;
```

```
import java.util.List;
```

以上内容仅为本文档的试下载部分，为可阅读页数的一半内容。如要下载或阅读全文，请访问：

<https://d.book118.com/457000150123006055>