

**2018级综合实训课**

□嵌入式系统综合实训（计算机科学与技术专业学生适用）

☑软件开发综合训练（软件工程专业学生适用）

□网络工程综合训练（网络工程专业学生适用）

**课题研究报告**

**课题名称： 订餐小程序**

**专业班级及分组：软件工程专业6班第4组**

**学生团队成员：吴金月、周佳怡、吴鸣明、林晓凯**

**指导教师： 陈绪行**

**计算机科学与工程学院**

**2021年7月**

**综合实训课团队成员分工表(2018级)**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 课  题  情  况 | 课题名称 | 订餐小程序 | | | | |
| 教师姓名 | 陈绪行 | 职称 | 讲师 | 学位 | 硕士 |
| 课程  类别 | □嵌入式系统综合实训（计算机科学与技术专业学生适用）  ☑软件开发综合训练（软件工程专业学生适用）  □网络工程综合训练（网络工程专业学生适用） | | | | |
| 团队成员分工安排 | **所在班级**：**2018级软件工程6班**  **队长：吴金月**  **组员：周佳怡、吴鸣明、林晓凯**  **团队成员具体分工安排如下：（下文需根据项目的实际完成情况，对团队成员各自承担的任务进行详细描述，必须要细致，工作任务落实到每个人，不得简化叙述。）**  吴金月任务内容:  1、编写需求分析的任务概述、需求规定  2、编写概要设计的总体设计  3、编写详细设计  4、编写数据库设计的外部设计  5、负责编码用户端的登录页面、主页面、商家页面及相关接口  周佳怡任务内容：  1、编写需求分析的涉众分析、约束、性能需求  2、编写概要设计的接口设计  3、编写详细设计  4、编写数据库设计的结构设计  5、负责编码用户端的确认订单页面、订单页面、订单详情页面、搜索商家页面及相关接口  吴明鸣任务内容：  1、编写需求分析的接口需求、公司需求  2、编写概要设计的运行设计、系统数据结构设计、系统出错处理设计  3、编写详细设计  4、编写数据库设计的运用设计  5、负责编码商家端的注册页面、登录页面、忘记密码页面、主页面及相关接口  林晓凯任务内容：  1、编写项目简介  2、编写需求分析的系统用例、数据描述  3、编写详细设计的14-18  4、编写功能测试  5、负责编码商家端的商家信息页面、商家信息修改页面、添加餐品页面、餐品管理页面、商家订单页面及相关接口  6、负责整合全部代码，测试功能 | | | | | |

（续表）

**订餐小程序**

作者姓名：吴金月、周佳怡、吴鸣明、林晓凯 专业班级：18软件工程6班

指导教师：陈绪行

**摘 要**

由于新冠疫情的原因，许多高校在开学后采取半封闭式管理，严格控制校园的人员流动，很多高校不允许学生和教职工之外的人员随意进出校园。由于外卖配送员无法进入学校，许多校外的餐饮店无法将餐品送到学生手中，导致学生在饮食上的选择减少很多，很多学生不得不选择在学校食堂或开设在校内的餐饮店就餐。

民以食为天，本项目的目标是致力于提供一个校园订餐平台，主要的服务对象是广大的学生和校园餐厅商户。一方面，协助校园餐厅商户改善运营方式，提高运营效率和服务质量。另一方面，服务于高校学生，为学生提供便捷的订餐方式，减少学生点餐后等待取餐的时间，优化学生的用餐体验。

**关键词：**订餐系统 微信小程序

Food Ordering Platform

Author: Jinyue Wu, Jiayi Zhou, Mingming Wu, Xiaokai Lin

Specialty: Software Engineering Tutor: Xuhang Chen

**Abstract**

Due to the epidemic situation of COVID-19, many colleges and universities adopt semi-closed management after the beginning of the term, strictly control the flow of personnel on campus, and many colleges and universities do not allow students and staff to enter and leave the campus at will. Due to the delivery staff can not enter the school, many off-campus restaurants can not deliver meals to the students, resulting in a lot less food choices for students, many students have to choose to eat in the school canteen or set up in the school restaurant.

Food is the most important thing for the people, the goal of this project is to provide a campus food ordering platform, the main service object is the majority of students and campus restaurant merchants. On the one hand, to assist campus restaurant merchants to improve the mode of operation, improve operational efficiency and service quality. On the other hand, serve college students, provide students with a convenient way to order meals, reduce the time for students to wait for meals, and optimize students' dining experience.

**Keywords：**food ordering platform Wechat Applet

**目录**

[1 绪论 1](#_Toc102550110)

[1.1 研究背景 1](#_Toc102550111)

[1.2 研究意义 1](#_Toc102550112)

[1.3 系统概述 1](#_Toc102550113)

[2 关键技术介绍 2](#_Toc102550114)

[2.1 spring框架 2](#_Toc102550115)

[2.1.1 spring框架简介 2](#_Toc102550116)

[2.1.2 特征 2](#_Toc102550117)

[2.2 相关技术栈简介 2](#_Toc102550118)

[3 需求分析 4](#_Toc102550119)

[3.1任务概述 4](#_Toc102550120)

[3.1.1目标用户分析 4](#_Toc102550121)

[3.1.2 应用环境 4](#_Toc102550122)

[3.2环境规定 5](#_Toc102550123)

[4 系统分析 6](#_Toc102550124)

[4.1 系统概述 6](#_Toc102550125)

[4.2系统处理流程 6](#_Toc102550126)

[4.3 系统结构 8](#_Toc102550127)

[5 系统设计 9](#_Toc102550128)

[5.1程序系统的结构 9](#_Toc102550129)

[5.2用户授权登录设计说明 10](#_Toc102550130)

[5.2.1程序描述 10](#_Toc102550131)

[5.2.2功能 10](#_Toc102550132)

[5.2.3性能 10](#_Toc102550133)

[5.2.4输入项 10](#_Toc102550134)

[5.2.5输出项 11](#_Toc102550135)

[5.2.6接口 11](#_Toc102550136)

[5.2.7存储分配 11](#_Toc102550137)

[5.2.8注释设计 11](#_Toc102550138)

[5.2.9限制条件 11](#_Toc102550139)

[5.3浏览餐品设计说明 11](#_Toc102550140)

[5.3.1程序描述 11](#_Toc102550141)

[5.3.2功能 11](#_Toc102550142)

[5.3.3性能 11](#_Toc102550143)

[5.3.4输入项 11](#_Toc102550144)

[5.3.5输出项 12](#_Toc102550145)

[5.3.6接口 12](#_Toc102550146)

[5.3.7存储分配 12](#_Toc102550147)

[5.3.8注释设计 12](#_Toc102550148)

[5.4选购餐品设计说明 12](#_Toc102550149)

[5.4.1程序描述 12](#_Toc102550150)

[5.4.2功能 12](#_Toc102550151)

[5.4.3性能 12](#_Toc102550152)

[5.4.4输入项 12](#_Toc102550153)

[5.4.5输出项 13](#_Toc102550154)

[5.4.6接口 13](#_Toc102550155)

[5.4.7存储分配 13](#_Toc102550156)

[5.4.8注释设计 13](#_Toc102550157)

[5.5取消选购设计说明 13](#_Toc102550158)

[5.5.1程序描述 13](#_Toc102550159)

[5.5.2功能 13](#_Toc102550160)

[5.5.3性能 13](#_Toc102550161)

[5.5.4输入项 13](#_Toc102550162)

[5.5.5输出项 14](#_Toc102550163)

[5.5.6接口 14](#_Toc102550164)

[5.5.7存储分配 14](#_Toc102550165)

[5.5.8注释设计 14](#_Toc102550166)

[5.6结算支付设计说明 14](#_Toc102550167)

[5.6.1程序描述 14](#_Toc102550168)

[5.6.2功能 14](#_Toc102550169)

[5.6.3性能 14](#_Toc102550170)

[5.6.4输入项 14](#_Toc102550171)

[5.6.5输出项 15](#_Toc102550172)

[5.6.6接口 15](#_Toc102550173)

[5.6.7存储分配 15](#_Toc102550174)

[5.6.8注释设计 15](#_Toc102550175)

[5.6.9限制条件 15](#_Toc102550176)

[5.7用户查询订单设计说明 15](#_Toc102550177)

[5.7.1程序描述 15](#_Toc102550178)

[5.7.2功能 15](#_Toc102550179)

[5.7.3性能 15](#_Toc102550180)

[5.7.4输入项 15](#_Toc102550181)

[5.7.5输出项 16](#_Toc102550182)

[5.7.6接口 16](#_Toc102550183)

[5.7.7存储分配 16](#_Toc102550184)

[5.7.8注释设计 16](#_Toc102550185)

[5.7.9限制条件 16](#_Toc102550186)

[5.8搜索商家设计说明 16](#_Toc102550187)

[5.8.1程序描述 16](#_Toc102550188)

[5.8.2功能 16](#_Toc102550189)

[5.8.3性能 16](#_Toc102550190)

[5.8.4输入项 17](#_Toc102550191)

[5.8.5输出项 17](#_Toc102550192)

[5.8.6接口 17](#_Toc102550193)

[5.8.7存储分配 17](#_Toc102550194)

[5.8.8注释设计 17](#_Toc102550195)

[5.9商家注册登录设计说明 17](#_Toc102550196)

[5.9.1程序描述 17](#_Toc102550197)

[5.9.2功能 17](#_Toc102550198)

[5.9.3性能 17](#_Toc102550199)

[5.9.4接口 18](#_Toc102550200)

[5.9.5存储分配 18](#_Toc102550201)

[5.9.6注释设计 18](#_Toc102550202)

[6 平台实现 18](#_Toc102550203)

[6.1 运行环境及开发工具 18](#_Toc102550204)

[6.2 系统界面 18](#_Toc102550205)

[7 系统测试 22](#_Toc102550206)

[7.1测试概要 22](#_Toc102550207)

[7.1.1测试目的 22](#_Toc102550208)

[7.1.2测试环境 22](#_Toc102550209)

[7.1.3测试执行 22](#_Toc102550210)

[7.2 测试结论 22](#_Toc102550211)

[8 结论与展望 23](#_Toc102550212)

[8.1 结论 23](#_Toc102550213)

[8.2 展望 23](#_Toc102550214)

[参考文献 24](#_Toc102550215)

[附录 25](#_Toc102550216)

# 1 绪论

## 研究背景

由于新冠疫情的原因，许多高校在开学后采取半封闭式管理，严格控制校园的人员流动，很多高校不允许学生和教职工之外的人员随意进出校园。由于外卖员无法进入学校，许多校外的餐饮店无法将餐品送到学生手中，导致学生在饮食上的选择减少很多，很多学生不得不选择在学校食堂或开设在校内的餐饮店就餐。

据了解，由于在食堂或校内餐饮店就餐的人数突然增多，导致在就餐高峰期会出现“排长龙”的情况。店家出餐的速度赶不上点餐量，因此点餐后至少要等侯二十多分钟，有时甚至要等候四十分钟以上，极大地影响了同学们的就餐质量，也浪费了同学们的时间。

## 研究意义

据调查，在用餐高峰期，有超过百分之六十的学生取餐时间超过十分钟，有超过百分之三十五的学生取餐时间超过二十分钟。总体而言等待取餐的时间较长，加上点餐人数过多，如此长的排队等待取餐时间必定造成大量拥挤。

同时，目前校内的餐饮店虽然采用了扫码点餐的方式，简化了点餐的过程，但是商家基本都是保持着学生到店扫码点餐后，商家再制作餐品的情况，学生每次点餐都需要排队等待商家出餐，极大地浪费了学生的时间。而且每个商家的二维码不同，当想吃某种餐品时，必须到对应的商家扫描该商家的二维码后才能进行点餐，减少了对学生对餐品的选择空间。

## 1.3 系统概述

民以食为天，本项目的目标是致力于提供一个校园订餐平台，主要的服务对象是广大的学生和校园餐厅商户。一方面，协助校园餐厅商户改善运营方式，提高运营效率和服务质量。另一方面，服务于高校学生，为学生提供便捷的订餐方式，减少学生点餐后等待取餐的时间，优化学生的用餐体验。

# 2 关键技术介绍

本博客系统是基于spring框架进行设计的，因而有必要知道什么是spring框架，有什么优缺点,并了解一些相关的技术栈。

## 2.1 spring框架

### 2.1.1 spring框架简介

Spring框架是[Java平台](https://baike.baidu.com/item/Java%E5%B9%B3%E5%8F%B0/3793459)上的一种开源应用框架，提供具有控制反转特性的容器。尽管Spring框架自身对编程模型没有限制，但其在Java应用中的频繁使用让它备受青睐，以至于后来让它作为[EJB](https://baike.baidu.com/item/EJB/144195)（EnterpriseJavaBeans）模型的补充，甚至是替补。Spring框架为开发提供了一系列的解决方案，比如利用控制反转的核心特性，并通过依赖注入实现控制反转来实现管理对象生命周期容器化，利用面向切面编程进行声明式的事务管理，整合多种持久化技术管理数据访问，提供大量优秀的Web框架方便开发等等。

### 2.1.2 特征

SpringBoot所具备的特征有：

（1）可以创建独立的[Spring](https://baike.baidu.com/item/Spring/85061)应用程序，并且基于其Maven或Gradle插件，可以创建可执行的JARs和WARs；

（2）内嵌Tomcat或Jetty等Servlet容器；

（3）提供自动配置的“starter”项目对象模型（POMS）以简化[Maven](https://baike.baidu.com/item/Maven/6094909)配置；

（4）尽可能自动配置Spring容器；

（5）提供准备好的特性，如指标、健康检查和外部化配置；

（6）绝对没有代码生成，不需要XML配置。

## 2.2 相关技术栈简介

技术栈：springboot、mysql、spring sercurity、mybatis、thymeleaf、Bootstrap、 Redis、layui、ActiveMQ、MongoDB、editorMarkDown

* Web端使用thmeleaf、bootstrap来实现响应式页面渲染、后端使用mybatis来实现数据持久化和PageHelper实现数据分页，中间件Redis、ElasticSearch、kafka来集成加强项目功能
* 使用Spring Security 做权限控制，替代拦截器的拦截控制，并使用自己的认证方案替代Security 认证流程，使权限认证和控制更加方便灵活。
* 使用Redis进行数据缓存，缓解数据库压力，使用Redis的数据类型set实现点赞，zset实现关注，并使用Redis存储ticket和验证码。
* 使用Redis高级数据类型HyperLogLog统计访问量,使用Bitmap统计网站日流量。
* 使用Elasticsearch做全局搜索
* 使用editorMarkDown实现富文本编辑器
* 使用Quartz定时同步缓存数据。
* 期望优化使用Kafka处理发送评论、点赞和关注等系统通知，构建异步消息系统。

# 3 需求分析

## 3.1任务概述

### 3.1.1目标用户分析

订餐者是指使用本系统进行订餐的用户。涉众的主要特征可以概括为：因时间问题或不想堂食，需要进行预约让商家先准备好餐品的用户。主要人群为高校学生或教职工。以下为该涉众的期望：

涉众期望1：系统操作的简便性。即人性化的人机交互界面。

涉众期望2：使用订餐系统的过程中用户信息高度安全。

涉众期望3：支持通过系统查询、浏览所需要的餐品信息。

涉众期望4：支持通过该系统进行订餐。

涉众期望5: 支持查询用户在本系统中的订餐订单信息。

商家是指在本系统中为订餐者提供餐品的用户。主要为高校食堂或入驻校内的餐饮店。

涉众期望1：系统操作的简便性。即人性化的人机交互界面。

涉众期望2：使用订餐系统的过程中用户信息高度安全。

涉众期望3：支持在本系统中对商家自身的信息进行修改、完善。

涉众期望4：支持在本系统中对添加商家的餐品信息。

涉众期望5: 支持在本系统中对商家已添加的餐品进行管理，包括餐品的上架、下架、删除。

涉众期望6: 支持在本系统中管理商家的营业状态，包括开店及打烊。

涉众期望7: 支持在本系统中查询商家的订单信息。

### 3.1.2 应用环境

CPU：8核心cpu，两颗，主频2.0GZ以上

内存：32GB

存储：高速存储，250G，普通存储，500G及以上

软件运行环境：

操作系统：Linux / Windows 7及以上

支持环境：JAVA（JDK1.8以上） Tomcat6以上

数 据 库：Mysql 5.6 / JDBC及以上

## 3.2环境规定

系统环境技术如表3-3-1所示

表3-3-1 系统环境

|  |  |
| --- | --- |
| ***工具*** | ***版本号*** |
| JDK | 1.8 |
| Mysql | 5.7 |
| Redis | 3.2 |
| layui |  |
| AticeMQ | 5.10.0 |
| MongoDB | 4.4.6 |

支持环境：JAVA（JDK1.8以上） Tomcat9以上

开发软件：idea（2020）

# 4 系统分析

## 4.1 系统概述

订餐小程序旨在帮助用户建立一个基于互联网的网络订餐平台，商家能够通过有效的方式，将自己的商品发布到订餐小程序系统中；而用户能够通过浏览、关键字检索等手段，获取自己所喜爱的餐品，然后进行预约。

1)基本概念设计

订餐小程序的核心目标是为用户提供餐品预约渠道，从而帮助用户减少用餐前的等待时间。所以，从目标出发，系统如何给订餐用户提供更为方便、好用的订餐环境，是系统设计上，需要重点思考的基础。系统在设计概念上，将完全考虑到以下因素：

本系统在设计上，考虑到用户的使用环境，整体结构将采用B/S结构进行搭建，它将使得用户完全忽略客户端所带来的困扰。

2)系统处理流程

订餐小程序需要实现四大核心流程：用户订餐流程、商家注册流程、商家上货流程。

用户订餐流程是订餐小程序的核心业务流程，它提供给用户一个标准化、简单易用的订餐过程，让用户能够方便的完成自己的订餐行为。

商家注册流程是订餐小程序用来获取商家的入口，它通过注册过程，完成商家的积累。

商家上货流程提供给订餐小程序的商家，帮助商家将自己通过网店销售的餐品上架到网店中，以达到餐品销售的目的。

下文将通过流程图，详细描述上面三个核心流程。

## 4.2系统处理流程

1. **用户订餐流程**

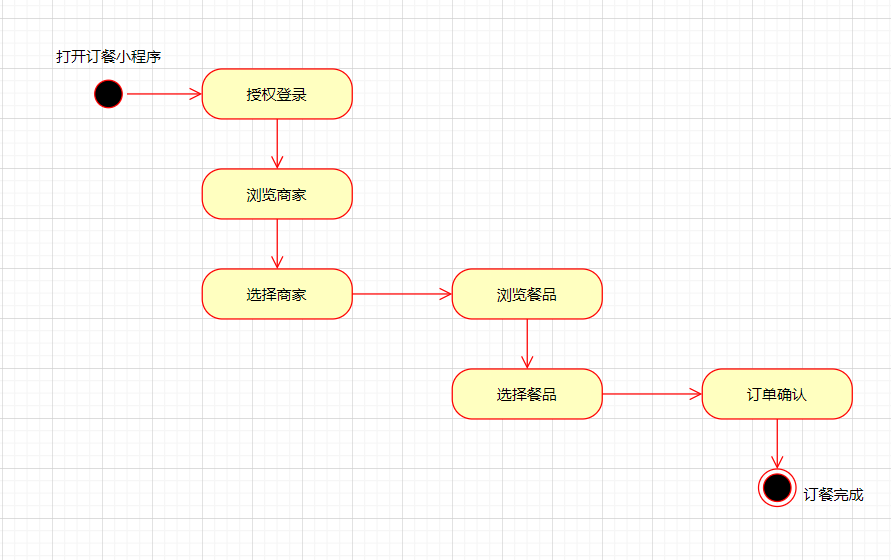


图4-2-1 用户订餐流程图

其流程包括以下步骤：

1. 用户打开订餐
2. 用户在系统中浏览商家
3. 用户选择商家并进入商家页面
4. 用户在商家页面流程商家的餐品
5. 用户选择餐品并加入购物车
6. 用户确认
7. 订餐流程完成
8. **商家注册流程**

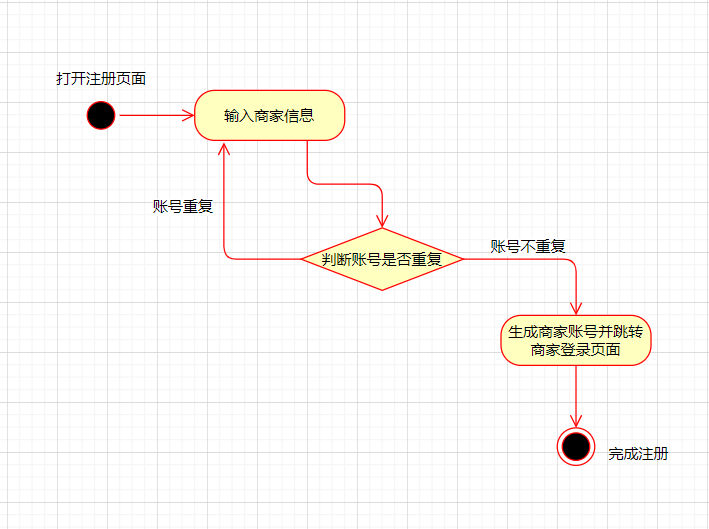


图4-2-2 商家注册流程图

其流程包括以下步骤：

1. 商家打开注册页面
2. 商家输入商家的注册信息
3. 当商家输入信息完毕之后，判断商家账号是否重复
4. 如果商家账号重复，则返回输入信息，如果商家账号不重复，则完成生成用户并跳转登录页面
5. 完成注册
6. **商家上货流程**

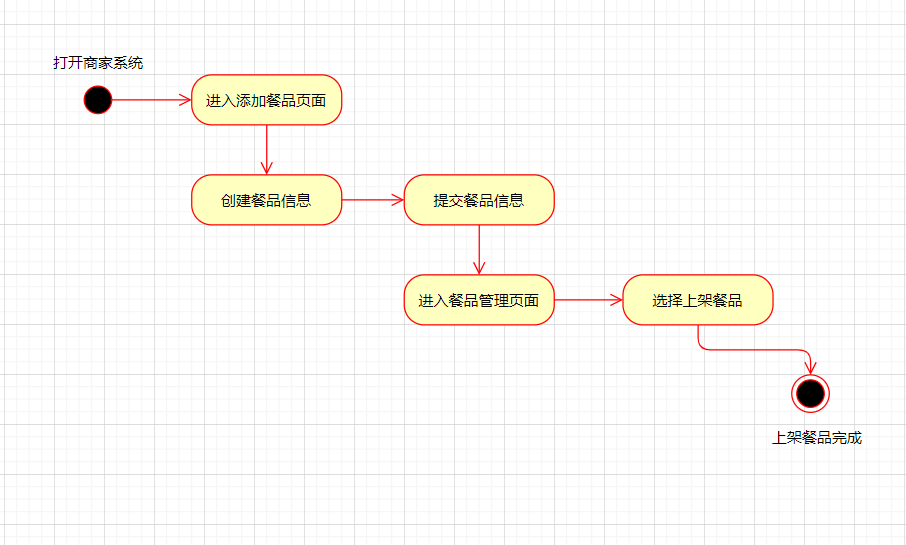


图4-2-3 商家上货流程图

其流程包括以下步骤：

1.商家打开商家系统

2.进入添加餐品的页面

3.商家输入餐品的信息

4.商家提交餐品的信息

5.商家进入餐品管理页面

6.商家选择要上架的餐品

7.商家餐品完成

## 4.3 系统结构

系统在软件使用模式上，为了方便用户使用，降低用户使用成本，采用B/S架构进行构建。同时，在系统软件架构上，采用MVC的分层模式对系统架构进行实现，确保系统安全稳定运行。其整体模块结构图如下：

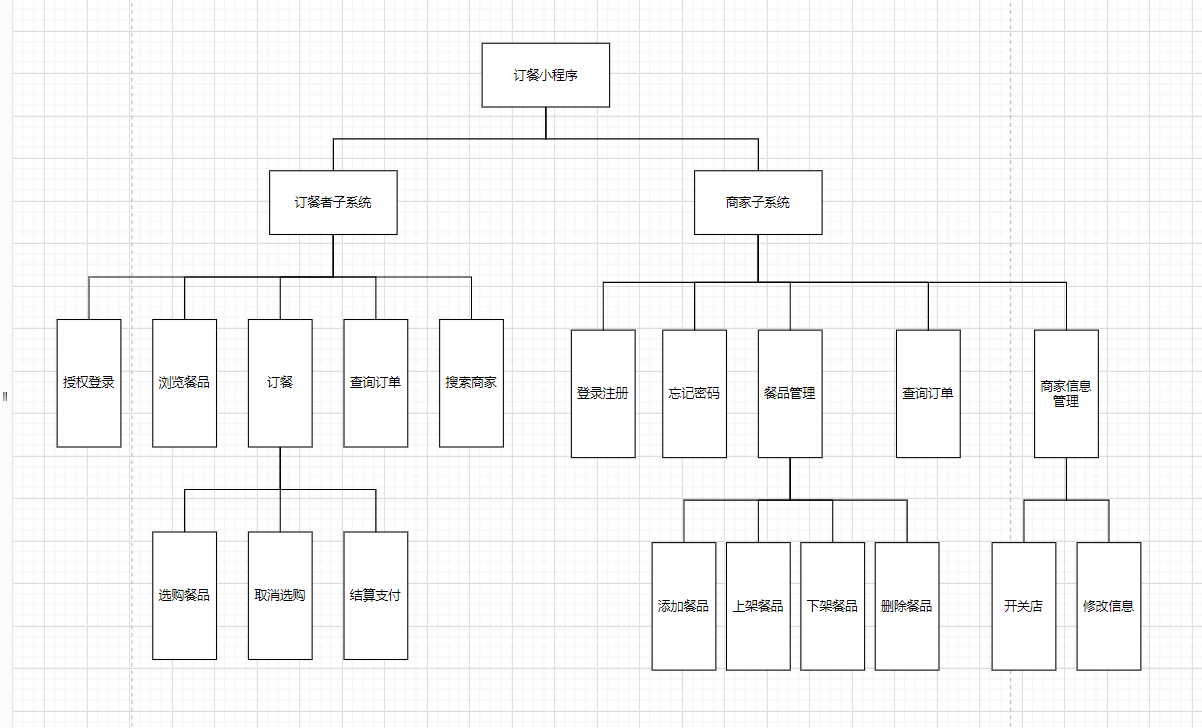


图4-3-1 系统模块结构图

整个系统分为五个层级：用户界面层、功能服务层，软件实现层，基础软件层和硬件服务层。

其中，功能服务层通过浏览器或微信小程序，对各类不同用户，提供不同的功能使用。

软件实现层提供了软件的MVC实现，它在每个层级都采用标准技术来完成层次的构建，并且因为层级存在，每个层次都可以根据需求不同进行调整而不影响其它层级运行。

基础软件层为系统的运行提供基本的环境，它包括软件运行的应用服务器中间件环境、JDK环境、数据库环境以及操作系统环境。

硬件服务层则为系统运行提供基本的硬件基础，它包括网络环境以及硬件服务器环境。

1）订餐者子系统

订餐者子系统能够处理用户订餐的整个过程。在进行订餐的时候，用户需要先授权系统使用用户的信息。用户授权后进入系统主页面浏览商家，选择商家后进入商家页面进行餐品浏览。然后用户可以将自己想要的餐品加入购物车中，确认后进行提交。最后，订餐者子系统能够将用户的餐品选购信息、用户的基本信息自动生成用户订单，通过与用户的交互确认，从而形成一次用户订餐活动。

2）商家子系统

商家子系统主要是给商家对餐品进行管理以及对商家信息的管理。商家在注册后登录系统中，可管理自身的营业状态，也可添加新的餐品并对餐品进行上下架处理，还可查询自家的所有订单，根据订单为订餐者准备餐品。

# 5 系统设计

## 5.1程序系统的结构

该博客系统采用多模块方式来开发，主要是一个主模块，里面包含多个子模块。每个子模块由页面包、公共包、数据库操作包、管理业务service包，实体类Model包组成，其包图如下

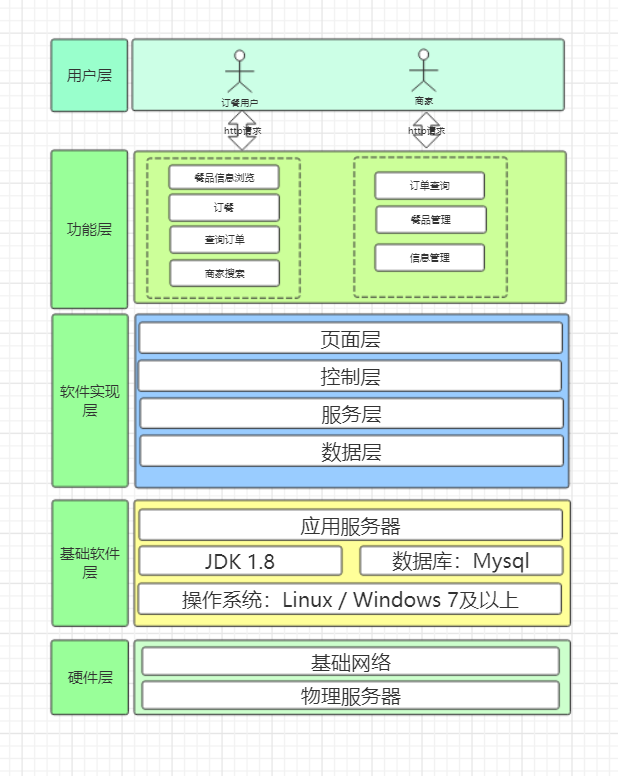


图5-1-1 系统结构图

## 5.2用户授权登录设计说明

### 5.2.1程序描述

本模块主要在用户首次使用本系统时记录下用户的信息，作为用户以后信息查询的依据。

### 5.2.2功能

用户在授权页面点击“授权”按钮，向微信官方获取临时凭证，获取后将临时凭证及用户信息发送到后端，系统根据用户的临时凭证向微信官方获取用户的唯一凭证（用户id）。判定用户信息是否在本系统中已存在，不存在则保存用户信息，存在则进入用户主页面。

### 5.2.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.2.4输入项

用户的临时凭证、用户头像、用户昵称、用户性别

### 5.2.5输出项

用户id、用户头像、用户昵称、用户性别存入用户信息表

### 5.2.6接口

本程序不使用外部接口。

### 5.2.7存储分配

本系统无存储描述。

### 5.2.8注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

### 5.2.9限制条件

暂无限制条件

## 5.3浏览餐品设计说明

### 5.3.1程序描述

本模块用于给用户查看所选择商家当前正在售卖的餐品。

### 5.3.2功能

用户进入主页面后可查看当前处于营业状态的商家列表，用户选择列表中任一商家后进入商家页面，在该页面中浏览商家当前正在售卖的餐品。

### 5.3.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.3.4输入项

用户的操作指令。

### 5.3.5输出项

主页面的输出项：商家列表（其中每个列表项包含商家名，商家图标，商家总销售量以及商家评分）。

商家页面的输出项：商家名、商家图标、餐品列表（其中每个列表项包含餐品名、餐品描述、餐品价格、餐品销售量）。

### 5.3.6接口

本程序不使用外部接口。

### 5.3.7存储分配

本程序无存储描述。

### 5.3.8注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

## 5.4选购餐品设计说明

### 5.4.1程序描述

本模块主要是用户在浏览餐品是使用，用于给后续的结算支付提供信息。

### 5.4.2功能

用户在商家页面浏览餐品时，若有想要购买餐品可点击餐品右下角的“+”按钮，即可将餐品添加进购物车中。

### 5.4.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.4.4输入项

用户的操作指令。

### 5.4.5输出项

页面信息或提示。

### 5.4.6接口

本程序不使用外部接口。

### 5.4.7存储分配

本系统无存储描述。

### 5.4.8注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

## 5.5取消选购设计说明

### 5.5.1程序描述

本模块用于用户在商家页面浏览餐品时，将已选购餐品从购物车中删除。

### 5.5.2功能

用户在商家页面浏览餐品时，若购物车中已有选购的餐品，可点击餐品右下角的“-”按钮取消该餐品的选购，或点击下方的购物车打开购物车面板，在购物车面板可点击“-”按钮取消选购，或直接点击右上角的“清空”按钮直接清空购物车。

### 5.5.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.5.4输入项

用户的操作指令。

### 5.5.5输出项

页面信息或提示。

### 5.5.6接口

本程序不使用外部接口。

### 5.5.7存储分配

本系统无存储描述。

### 5.5.8注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

## 5.6结算支付设计说明

### 5.6.1程序描述

本模块用于用户选购餐品完毕后进价格统计及支付订单。

### 5.6.2功能

用户在商家页面选购完毕后点击“去结算”按钮进入支付页面，支付页面显示当前订单中的餐品信息及总价格。用户确认无误后点击“去支付”按钮支付，完成后提示用户并自动跳转至订单页面。

### 5.6.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.6.4输入项

商家id、用户id、结算时间、用户选购的餐品信息(包括餐品id、餐品名、价格、数量），总价格，餐品总数量。

### 5.6.5输出项

页面提示及订单页面。

### 5.6.6接口

本程序不使用外部接口。

### 5.6.7存储分配

本系统无存储描述。

### 5.6.8注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

### 5.6.9限制条件

暂无限制条件

## 5.7用户查询订单设计说明

### 5.7.1程序描述

本模块用于用户查询自己已有的订单信息。

### 5.7.2功能

用户在授权登录进入主页后，可点击下方菜单栏的“订单”进入订单页面，订单页面显示用户已有的订单列表。用户选择任一订单后进入订单详情页面，该页面中显示用户订单的详细信息。

### 5.7.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.7.4输入项

用户的操作指令。

### 5.7.5输出项

订单页面的输出项：用户订单列表（每一项包括订单的商家名、商家图标、订单时间、总价格、餐品总数量、主要的餐品名）

订单详情页面的输出项：商家名、商家图标、订单号、订单时间、总价格、餐品列表（每一项包括餐品名、数量、价格）。

### 5.7.6接口

本程序不使用外部接口。

### 5.7.7存储分配

本系统无存储描述。

### 5.7.8注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

### 5.7.9限制条件

暂无限制条件

## 5.8搜索商家设计说明

### 5.8.1程序描述

本模块主要用于用户搜索商家。

### 5.8.2功能

用户授权登录进入主页后，可点击上方的输入框中输入要搜索的商家名，再点击“搜索”按钮进入搜索页面。进入搜索页面后显示商家名与用户检索条件相似的商家列表。

### 5.8.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.8.4输入项

用户要搜索的商家名。

### 5.8.5输出项

若有商家名相似的商家则输出商家列表（每一项包括商家名、商家图标、商家总销售量、评分）；

若没有商家名相似的商家则给出相应提示。

### 5.8.6接口

本程序不使用外部接口。

### 5.8.7存储分配

本系统无存储描述。

### 5.8.8注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

## 5.9商家注册登录设计说明

### 5.9.1程序描述

本模块主要在商家首次使用本系统时记录下商家的信息，作为商家以后信息查询和各项操作的依据。

### 5.9.2功能

商家在登录页面点击“登录”按钮上方的蓝色字体“点我注册为商家”进入商家在注册页面填写商家的信息后点击下方的“填写完毕”，前端验证填写的信息是否合法后，系统验证商家的账号是否已被注册，若是提示商家错误信息，若不是则保存商家信息并提示“注册成功”后跳转登录页面。

### 5.9.3性能

程序要求各功能用户发出操作指令后，能够在5秒内能返回操作结果。程序要求各层级之间的数据传输在0,1s内完成。

### 5.9.4接口

本程序不使用外部接口。

### 5.9.5存储分配

本系统无存储描述。

### 5.9.6注释设计

本程序中安排的注释如下：

* 1. 加在类首部的注释；
  2. 加在方法首部的注释
  3. 加在各分枝点处的注释；
  4. 对各变量的功能、范围、缺省条件等所加的注释；
  5. 对使用的逻辑所加的注释等等。

# 6 平台实现

## 6.1 运行环境及开发工具

服务器：基于Intel构架的企业服务器

支持环境：JAVA（JDK1.8以上） Tomcat9以上

数 据 库：MySQL5.7以上

编程语言：Java,html,js

设计工具：intellij idea、redis、powerdesign、archi等

## 6.2 系统界面

1） 主页面

主页面主要显示商家列表。界面如图6-1所示。



图6-1 用户主界面

1. 商家界面

商家页面中显示该商家的餐品信息。界面如图6-2所示。



图6-2 商家界面

1. 结算页面

结算页面中显示用户选择的餐品信息以及订单的总价格，如图6-3所示。



图6-3 结算界面

1. 订单页面

订单页面中显示用户的订单列表，点击任一订单可查看订单详细信息，如图6-4所示。



图6-4 订单页面

1. 订单详情页面

订单详情页面中显示用户订单的详细信息，包括商家名、餐品信息、订单号、下单时间等，如图6-5所示。



图6-5 订单详情页面

6）搜索商家页面

搜索商家页面中根据用户的检索条件显示商家列表，如图6-6所示。

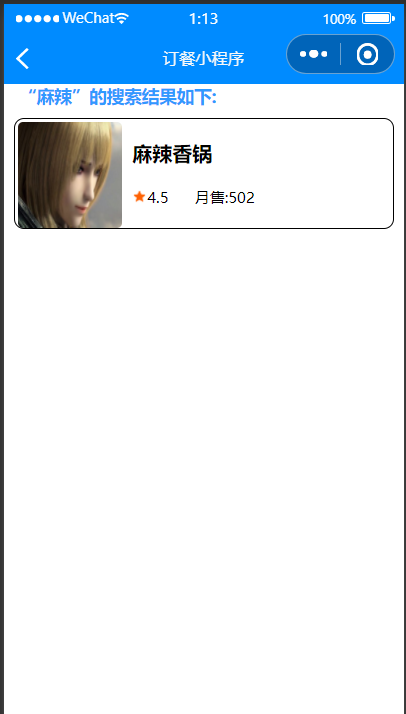


图6-6 搜索商家页面

# 7 系统测试

## 7.1测试概要

### 7.1.1测试目的

在于执行测试提供用例，指导测试的实施，查找分析缺陷，评估测试质量并执行测试用例后，需要跟踪故障，以确保开发的产品适合需求。

测试声明：测试人员在软件开发过程中的任务

① 寻找BUG；

② 避免软件开发过程中的缺陷；

③ 衡量软件的品质；

④ 关注用户的需求。

### 7.1.2测试环境

系统测试环境如表7-1所示

表7-1 测试环境

|  |  |
| --- | --- |
| *环境* | *所有环境都是部属在单机上的* |
| 硬件配置 | CPU：奔腾G5500  内存：16G  硬盘：金士顿固态240G |
| 软件配置 | Win10、Jmeter测试工具 |
| 网络配置 | 无要求 |

### 7.1.3测试执行

主要采用黑盒测试，以需求分析文档中的功能模块为单位，测试的重点集中在基本数据录入、业务流程及各功能模块间的接口。

## 7.2 测试结论

1）功能

系统正确的实现了用户登录功能等功能，已存在一些账户和密码的约束问题。例如：要用户名不能重复……

2）性能

系统在单机部署的情况下，能承载20000个用户发起的20000次请求。但视不同的用户主机有一些小小的差异，当线程开太多的时候容易产生冲突，出现异常。

# 8 结论与展望

## 8.1 结论

由于新冠疫情的原因，许多高校在开学后采取半封闭式管理，外卖配送员无法进入学校，许多校外的餐饮店无法将餐品送到学生手中，导致学生在饮食上的选择减少很多，很多学生不得不选择在学校食堂或开设在校内的餐饮店就餐。对此，我们想开发一个用于校园内订餐的平台（微信小程序），通过提前订餐并在规定时间内取餐的方式能够尽量避免必须到店才能点餐的情况，减少就餐高峰期因点餐人数过多而导致“排长龙”情况的发生，尽量节约同学们从点餐后到取得餐品中间所浪费的时间。

## 8.2 展望

一个完美的产品是希望它能适应各式各样的需求，本系统框架也是不尽完美的，也是有需要改进的地方，这也是展望之处：

1）减少不必要的冗余代码；

2）实现更多的个性化设置、功能；

3）优化前端页面；

# 参考文献

[1]黄文毅.springboot 微服务项目实战[M].

[2]王珊.数据库系统概论[M].

[3]软件设计开发文档【国家标准】-GB8567-88[S].

[4]朱少明.软件质量保证和管理[M].

[5]黑马.Java实用教程[M].

[6]尚硅谷.数据库系统概论[M].

[7]张奇思.数据库设计三大范式[R].

[8]钟鸣.MySQL必知必会[M].

[9]谢希仁.计算机网络[M].

[10]耿祥义,张跃平.Java 2实用教程[M].

[11]黄文毅.SpringBoot微服务项目实战[M].

[12]赵丰年.网页设计与制作HTML5+CSS3+JavaScript[M].

[13]黄文毅.一步一步学Spring Boot微服务项目实战[M].

附录