



本科生毕业论文（设计）

题目：中山大学

本科毕业论文（设计）

姓 名 朱 龙 凯

学 号 17341222

院 系 计算机学院

专 业 计算机科学与技术

指导教师 毛明志 (副教授)

2021 年 4 月 2 日

中山大学
本科毕业论文（设计）

**基于 Android 前端和 Springboot 后台的
食堂订餐系统的设计与实现**

姓 名	朱 龙 凯
学 号	17341222
院 系	计算机学院
专 业	计算机科学与技术
指导教师	毛明志 (副教授)

2021 年 4 月 2 日

学术诚信声明

本人郑重声明：所呈交的毕业论文（设计），是本人在导师的指导下，独立进行研究工作所取得的成果。除文中已经注明引用的内容外，本论文（设计）不包含任何其他个人或集体已经发表或撰写过的作品成果。对本论文（设计）的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。本论文（设计）的知识产权归属于培养单位。本人完全意识到本声明的法律结果由本人承担。

作者签名：

日 期： 年 月 日

【摘 要】

目前国内的很多食堂中，仍然采用顾客在吃饭时间前往食堂排队取餐的运作模式，这种模式的弊端很明显，首先，用餐高峰期顾客在排队取餐的过程中，会造成很多时间的浪费；其次，由于食堂对顾客数量很难准确评估，造成提前预备饭菜时对量的把握不够准确，往往会造成食物的浪费。

本系统旨在开发一个包含后端、前端和客户端的食堂订餐系统，做到食堂管理员通过前端发布菜品信息，并且可以管理用户信息、菜品信息和订单信息等，用户通过客户端访问后端服务器，浏览食堂即将提供的菜品，由于可能存在多个食堂进行选择，用户还可以选择浏览不同食堂不同日期将有提供的菜品，根据个人意愿，实现下单订餐这一核心功能。为了加大用户粘性，为此系统添加了社交属性，实现用户之间的交流。同时为了提高食堂服务质量和顾客满意度，系统开通了顾客投诉和建议渠道，食堂可以根据顾客反馈进行整改。

此系统采用 SpringBoot 作为后端服务器框架，采用 EasyUI 框架实现前端，采用 MyBatis 实现数据持久化，客户端 App 则采用原生 Java 开发，使用 Material Design 原则美化部分界面，同时采用 Glide 框架进行图片加载，LitePal 框架进行持久化存储，Gson 库进行 json 数据与对象之间的转换等。

此系统已经开发完毕，设计功能都已经完成，运行效果良好。

【关键词】 食堂订餐系统，SpringBoot 框架，Android

[ABSTRACT]

At present, many canteens in China still adopt the mode of operation where customers go to the canteen to line up to pick up meals during meal time. The disadvantages of this model are obvious. First of all, customers will waste a lot of time when queuing up to pick up meals during peak dining periods. ; Secondly, because the canteen is difficult to accurately assess the number of customers, it is not accurate enough to grasp the amount when preparing meals in advance, which often results in food waste.

This system aims to develop a canteen ordering system that includes a back-end, a front-end and a client, so that the canteen administrator can publish dish information through the front-end, and can manage user information, dish information, order information, etc. The user accesses the back-end server through the client , Browse the upcoming dishes in the canteen. Since there may be multiple canteens to choose from, the user can also choose to browse the dishes that will be available in different canteens on different dates, and realize the core function of ordering and ordering according to personal wishes. In order to increase user stickiness, social attributes are added to the system to achieve communication between users. At the same time, in order to improve the service quality of the canteen and customer satisfaction, the system has opened channels for customer complaints and suggestions, and the canteen can be rectified based on customer feedback.

This system uses SpringBoot as the back-end server framework, EasyUI framework to realize the front-end, MyBatis to realize data persistence, and the client App is developed in native Java. The Material Design principle is used to beautify part of the interface. At the same time, Glide framework is used for image loading and LitePal framework. For persistent storage, the Gson library performs conversion between json data and objects.

The system has been developed, the design functions have been completed, and the operation effect is good.

[Keywords] Canteen ordering system, springboot framework, Android

目录

一、 绪论	1
1.1 选题背景与意义	1
1.2 国内外研究现状和相关工作	2
1.3 本文的论文结构与章节安排	2
二、 相关理论与技术	3
2.1 客户端技术	3
2.2 前端技术	4
2.3 后端技术	5
三、 食堂订餐系统需求分析与设计	7
3.1 系统边界	7
3.2 食堂订餐系统需求与系统分析	8
3.3 登录模块	10
3.4 菜品管理模块	11
3.5 订单管理模块	13
3.6 用户管理模块	15
3.7 食堂信息管理模块	16
3.8 系统设计	18
四、 系统功能模块的实现	21
五、 总结和展望	22
六、 实验与结果	23
参考文献	24
致谢	25

附录 A 补充更多细节	26
A.1 补充图	26

插图目录

3-1 订餐系统边界图	7
3-2 管理员用例图	8
3-3 用户用例图	9
3-4 系统架构图	18
3-5 后端系统功能模块图	19
3-6 系统数据表关系图	19
A-1 一个配有彩色表格的插图	26

表格目录

3.1	管理员登录模块需求描述	10
3.2	用户登录模块需求描述	10
3.3	查询菜品信息需求描述	11
3.4	创建菜品信息需求描述	11
3.5	修改菜品信息需求描述	12
3.6	下架菜品信息需求描述	12
3.7	创建订单需求描述	13
3.8	修改订单需求描述	13
3.9	查看订单需求描述	14
3.10	禁用用户需求描述	15
3.11	启用用户需求描述	15
3.12	修改食堂信息需求描述	16
3.13	增加食堂需求描述	16
3.14	修改管理员信息需求描述	17
3.15	新增管理员需求描述	17

一、绪论

1.1 选题背景与意义

食堂就餐是我们很多人生活中都会出现的场景，尤其是学生和企业员工，除了在家就餐，很多时候都会选择在食堂进行就餐。在网络和智能手机并不是很普及的时候，大家都选择到食堂排队就餐的方式好像无可厚非，虽然会出现排队时间过长、偶尔有人不文明的插队行为，大家也都觉得很正常。但是随着移动互联网的发展和智能手机的普及，很多传统生活中的生活方式都被颠覆着，电子商务使得购物不再需要频繁的前往实体店，给了用户更多的选择和便利、网络的发展也促进了视频行业的发展，越来越快的网络使得人们可以观看更加高清的视频而不会卡顿，也促生了很多视频内容平台，比如抖音、快手等、移动网络的发展也极大的促进了即时通信，方便了人们之间的交流和沟通。

然而在人们生活当中非常普遍的食堂就餐场景，却还几乎没有被科技进步的春风吹拂，食堂的运作模式也是十年如一日，鲜有改进。对于就餐高峰期，顾客们由于漫长的排队等待时间滋生的不满情绪，食堂管理人员熟视无睹，好像认为浪费时间排队是必然的一件事，没有积极主动去寻求解决办法，可想而知，由于人们不愿排队，必然会造成食堂就餐人数的减少，食堂收入必然受到影响。食堂长期以来习惯于通过经验来评估需要提前准备多少饭菜来供应给顾客，但如果仅仅依靠经验，很容易造成对饭菜预备数量评估不准，如果准备少了，很多顾客就没有办法就餐，准备多了就会造成浪费，这是一个急需解决的问题。

互联网的兴起已经给人们的就餐方式带来了一些改变，外卖越来越多的被人们接受，但是外卖的很多缺点也为人们所诟病，比如价格比较贵，很多商家生产制造的过程不卫生等。如果将诸如美团外卖或者饿了么这样的外卖平台与生产经营比较规范的食堂结合起来，会有很大的好处。

本文描述的基于 Android 前端和 SpringBoot 后台的食堂订餐系统就是为了将订餐功能引入食堂的经营方式，希望食堂顾客在使用此系统进行订餐以后，疏解食堂就餐高峰期的压力，食堂的管理者也能够通过订餐人数做到备餐数量和需求之间的合理匹配减少不必要的浪费。另一个益处是订餐系统可以减少不必要的人群聚集，对于应对类似 2020 年的疫情非常有利。

由于食堂拥有固定的就餐客户群体，客流量稳定而集中，此系统每给一个人带来便利和每给一个食堂带来管理效率的提高，那总的来说，收益都将是非常巨

大的。

1.2 国内外研究现状和相关工作

目前外卖已经为越来越多的人接受，外卖系统的用户心智也愈发成熟，但是入驻外卖系统的通常是一些餐厅和个体经营者，传统的食堂并没有入驻，一方面是因为食堂的用户群体比较固定，比如学校食堂通常只为师生提供餐饮服务，公司食堂只为本单位员工提供餐饮服务，另一方面是食堂中很大一部分是学生食堂，为了减少学生使用手机不能接入订餐系统，基于以上原因，国内的学校食堂基本上没有使用订餐食堂的。

但是目前国内已经有少数公司为了方便员工就餐，已经开始尝试使用订餐系统了，比如快手公司，这是一家经营短视频社区的互联网公司，公司拥有多个食堂，每个食堂提前几天将准备提供的饭菜发布到订餐系统，员工可以选择不同的食堂查看不同日期的菜品，根据个人选择提前订餐或者直接去食堂就餐，如果选择订餐，那食堂会派人将饭菜送到楼层取餐柜，员工直接去取就可以了。此系统可以使得员工很方便确认自己想去哪里就餐，因为就餐之前就已经知道不同的食堂分别会提供什么饭菜。

但在这个系统仍然存在问题，由于公司餐饮免费，所以很多人订餐以后并没有去取餐，造成浪费和不卫生的情况。

总的来说，食堂订餐系统并不是一个成熟的领域，也并无太多经验可以借鉴。

1.3 本文的论文结构与章节安排

本文共分为六个章节，各章节内容安排如下

第一章绪论。简单说明了本文章的选题背景、意义和国内外研究现状。

第二章相关理论和技术。主要介绍开发此系统所使用的具体技术，简要说明技术的原理和技术方法。

第三章为需求分析与设计。对系统需要实现的功能和模块进行介绍。

第四章为功能模块的实现。通过核心代码辅助介绍各个功能模块是如何实现的，以及如何关联的。

第五章为系统成果演示。演示系统运行的结果。

第六章总结和展望。总结系统开发和论文写作过程中的收获，以及对食堂订餐系统进一步演进的构想。

二、相关理论与技术

2.1 客户端技术

2.1.1 Glide

Glide 是一个用于 Android 的快速高效的开源媒体管理和图像加载框架，它将媒体解码、内存和磁盘缓存以及资源池封装到一个简单易用的界面中，Glide 支持获取、解码和显示视频静像、图像和动画 gif。在本次订餐系统开发的过程中，主要用于从服务器请求菜品图片并显示到图片控件 ImageView 上。

2.1.2 Gson

Gson 是一个 Java 库，通过反射的方式，可用于将 Java 对象转换为其 JSON 表示。它还可以用于将 JSON 字符串转换为等效的 Java 对象。反射是指程序可以访问、检测和修改它本身状态或行为的一种能力通过反射可以获取类型的相关信息、动态调用方法、动态构造对象、从程序集中获得类型。在订餐系统开发中，由于订餐 App 从服务器请求到的数据都是 json 格式的，需要做数据解析处理，得到 json 中每个字段的值，使用 Gson 库考验很方便的将每个字段映射为预先定义的一个 Java 对象的每个属性的值。

2.1.3 OkHttp

HTTP（超文本传输协议）是计算机网络协议的应用层协议用于客户端和服务端之间交换数据。高效地使用 HTTP 可以加快加载速度并节省带宽。OkHttp 是默认情况下高效的 HTTP 客户端，具有以下特性：1、HTTP/2 支持允许对同一主机的所有请求共享一个套接字；2、连接池减少了请求延迟（如果 HTTP/2 不可用）；3、透明 GZIP 缩小了下载大小；4、响应缓存完全避免了重复请求的网络。在订餐系统的开发中，由于需要客户端 App 需要与服务器进行网络通信，交换数据，使用 OkHttp 将用于网络请求的方法提取到一个类中，非常简单高效，而且可以多次复用。

2.1.4 LitePal

LitePal 是一个开源的 Android 库，使用对象关系映射（ORM）模式，配置简单，允许开发人员非常容易地使用 SQLite 数据库。甚至不需要编写 SQL 语句就可以完成大多数数据库操作，包括创建或升级表、数据库的增删改查操作、聚合函数等。在订餐系统开发中，从服务器请求到的数据需要做本地存储，以实现数据复用的目的，这样可以避免每次用户使用 App 都重新从服务器请求数据，造成用户流量的浪费。LitePal 可以方便的将继承了 LitePalSupport 类的类对象进行持久化存储，在需要的时候从本地磁盘读出。

2.1.5 Material Design

Material Design 是由 Google 的设计工程师基于传统优秀的设计原则，结合丰富的创意和科学技术所发明的一套全新的界面设计语言，包含了视觉、运动、互动效果等特性。2015 年的 Google 的 I/O 大会上推出了一个 Design Support 库，这个库将 Material Design 中最具代表性的一些控件和效果进行了封装，使得开发者在不了解 Material Design 的情况下也能轻松的将自己的应用 Material 化。在订餐 App 中，浮动按钮、下拉刷新、可折叠式标题栏都是使用 Material Design 的体现。

2.2 前端技术

2.2.1 jQuery

jQuery 是一个高效、精简并且功能丰富的 JavaScript 工具库。它提供的 API 易于使用且兼容众多浏览器，这让诸如 HTML 文档遍历和操作、事件处理、动画和 Ajax 操作更加简单。在 jQuery 的基础之上已经开发出了众多优秀的前端框架，比如 jQuery Easy UI、Bootstrap 和 Vue 等。

2.2.2 Easy UI

Easy UI 是以 jQuery 为核心开发的前端 UI 插件，目的是让 web 开发者快捷方便地构建出功能丰富且美观的用户界面。开发者无需编写复杂的 JavaScript，也无需对 CSS 样式有深入了解，只需要有一些 HTML 和 jQuery 的基础，就可以轻松开发出一些较好的软件界面。在订餐系统中，使用 Easy UI 开发用于食堂管理员登录操作的前端界面，Easy UI 的每个组件都有属性、方法和事件，因此对于解析后端传回的数据和发起异步请求都非常方便。

2.3 后端技术

2.3.1 SpringBoot

SpringBoot 基本上是 Spring 框架的扩展，它消除了设置 Spring 应用程序所需的繁琐的配置，为更快，更高效的开发生态系统铺平了道路。也仍然继承了 Spring 的核心特性，依赖注入 (DI) 和面向切面编程 (AOP)，使得程序组件高度解耦，提高了程序的可维护性和可扩展性。在 SpringBoot 项目中，摒弃了以往 Spring 项目中大量繁琐的配置，遵循约定大于配置的原则，通过自身默认的配置，极大的降低了项目的复杂度，可以使得开发者更加专注于程序代码的开发。订餐系统后端主要就是基于 SpringBoot 进行开发的，对于应用的创建、调试、部署等都非常友好。

2.3.2 MyBatis

MyBatis 是 Apache 开发的一个持久层开源框架，支持普通的 SQL 查询、存储过程、关联映射。MyBatis 消除了几乎所有的 JDBC 代码和参数的手工配置以及对结果集的检索。MyBatis 可以使用简单的 XML 文件或者注解进行配置和映射基本数据类型，将借口和普通 Java 对象进行映射成数据库中的记录。MyBatis 支持动态 SQL、关联映射和多级缓存，功能强大且高效。

在开发订餐系统过程中，MyBatis 主要是用来将前端传回的表单数据以对象的形式写入数据库，也会进行一些更新、查询和删除的操作。

2.3.3 JSR303

这是 Java 为 Bean 数据合法性校验所提供的的一个框架。JSR303 支持 XML 风格和注解风格的验证，通过在 Bean 属性上标注类似于 @NotNull(不能为空)、@Email(邮件格式) 等的标准注解指定校验规则，并通过标准的验证借口对 Bean 进行验证。这样，当客户端提交的某些表单是非法的时候，可以给出提示而不是直接接受。

2.3.4 Tomcat

Tomcat 是 Apache 软件基金会的 Jakarta 项目中的一个核心项目，是一款轻量级应用服务器。因为 Tomcat 技术先进、性能稳定，而且免费，因而深受 Java 爱好者的喜爱并得到了部分软件开发商的认可，成为目前比较流行的 Web 应用服务器。订餐系统后端程序就是运行在 Tomcat 服务器上。

adjustbox caption subfigure

三、食堂订餐系统需求分析与设计

本章主要介绍食堂订餐系统的总体概述和需求分析，然后对系统中重要的构成模块登录模块、菜品管理模块、订单管理模块、用户管理模块、食堂和管理员信息管理模块进行详细的需求分析，最后对本文涉及的模块进行了概要设计。

3.1 系统边界

食堂订餐系统是提供给某个单位进行订餐的系统，订餐顾客需要登录验证身份，才能浏览菜品并进行下单等操作。管理员同样需要登录，身份验证通过以后才能够对菜品信息、订单信息、用户信息等进行操作。

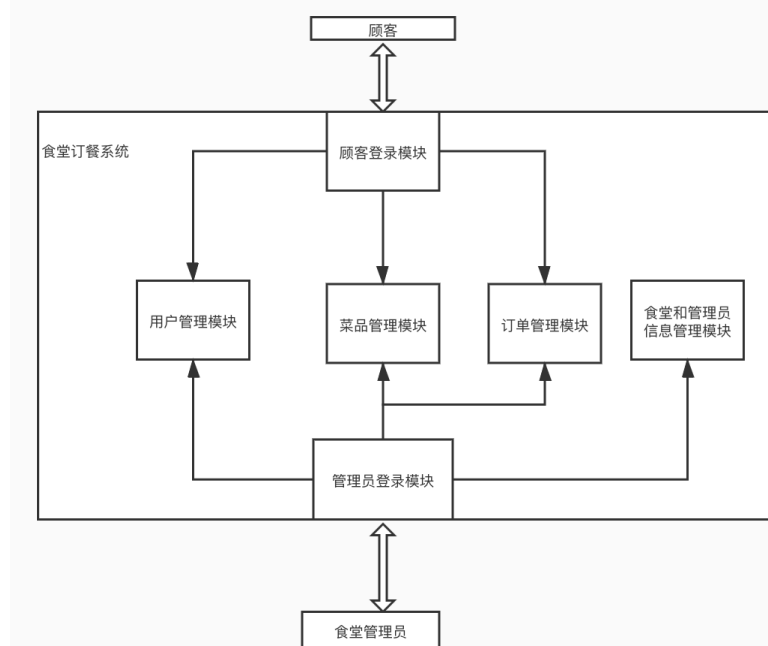


图 3-1 订餐系统边界图

图 3-1 是食堂订餐系统边界图，顾客和管理员都只有在登录以后才能够访问其他模块进行操作，顾客虽然也能同时访问用户管理、菜品管理和订单管理等模块，但是只能进行读操作，不能像管理员一样可以进行权限更高的写操作。

3.2 食堂订餐系统需求与系统分析

食堂订餐后台管理系统用于管理员登录后，对菜品信息、菜品类型、订单信息、顾客信息和食堂信息进行管理。在这个系统中，管理员用例图如图 3-2。根据需求分析，管理员拥有如下功能权限：

- 1) 菜品管理，包括添加菜品、菜品下架、修改菜品信息、查询菜品。
- 2) 菜品类型管理，包括添加菜品类型、删除菜品类型、修改菜品类型。
- 3) 订单管理，包括查看订单，删除订单，查看订单明细；
- 4) 用户管理，查询用户，启用和禁用用户。
- 5) 食堂管理，添加食堂、修改食堂营业状态。

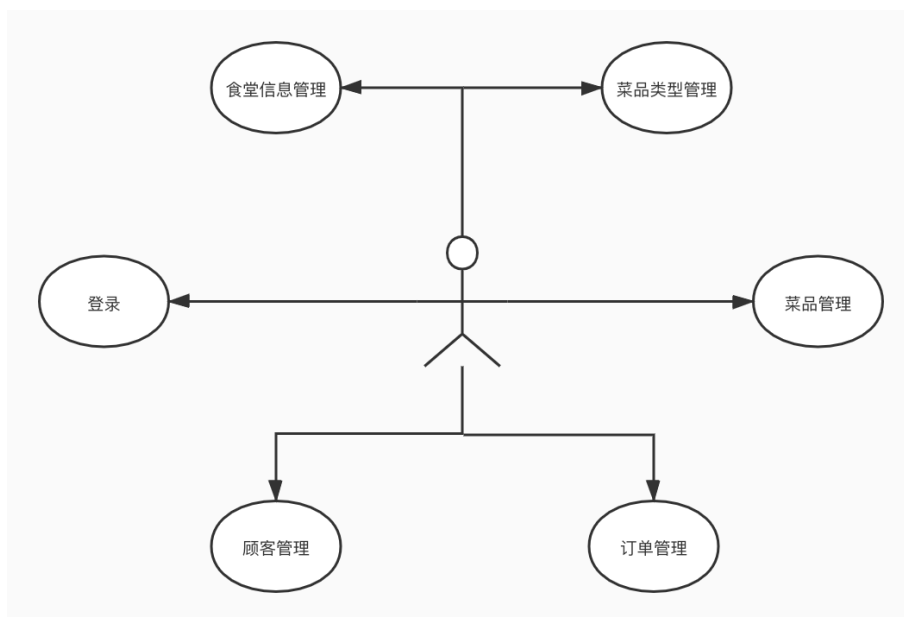


图 3-2 管理员用例图

订餐 App 是食堂订餐系统的用户终端，用户通过订餐 App 接入食堂订餐系统。

图 3-3 是用户实例图，用户通过访问后端接口提交数据，拥有如下功能：

- 1) 菜品管理，查询菜品。
- 2) 订单管理，创建订单，删除订单，查看订单明细。
- 3) 用户管理，修改用户信息。
- 4) 食堂管理，提交反馈建议。

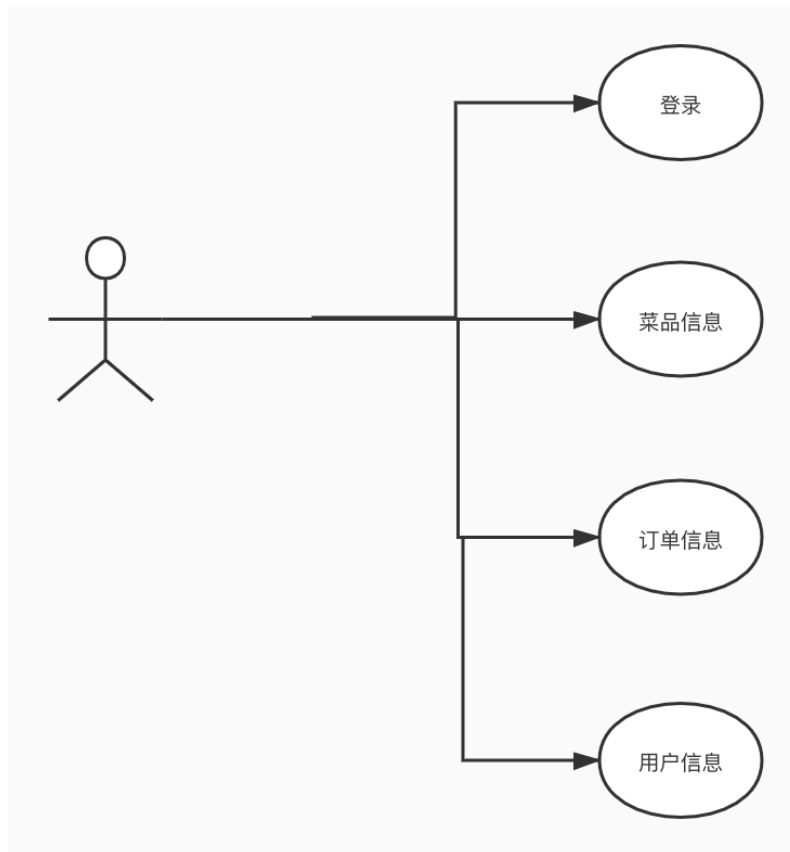


图 3-3 用户用例图

3.3 登录模块

此模块作为管理员和用户访问系统的入口模块，旨在对管理员和用户的身份进行合法性验证，却不其具备访问系统的权限，而不会对系统造成破坏。登录模块分为管理员登录模块和用户登录模块，只有进行了登录验证以后才能进入功能页面进行其他操作。

表 3.1 管理员登录模块需求描述

名称	登录订餐后台
参与者	管理员
触发条件	发起登录请求
前置条件	提交账户和密码
后置条件	登录成功进入管理员首页，否则给出登录失败提示
优先级	高
正常流程	发起登录请求，输入账户和密码，提交信息，验证是否登录成功

表 3.2 用户登录模块需求描述

名称	登录订餐 App
参与者	用户
触发条件	在登录页发起登录请求
前置条件	提交账户和密码
后置条件	登录成功进入列表页，否则给出登录失败提示
优先级	高
正常流程	发起登录请求，输入账户和密码，提交信息，验证是否登录成功

3.4 菜品管理模块

此模块管理员与用户都可以访问，不过权限功能不同，管理员可以对菜品数据进行查询、删除、修改和增加操作，但是用户只能浏览查询菜品列表信息。

表 3.3 查询菜品信息需求描述

名称	查询菜品信息
参与者	管理员、用户
触发条件	发起查询请求
前置条件	已经登录
后置条件	返回所有菜品信息
优先级	高
正常流程	管理员和用户查询菜品信息，服务器返回菜品列表

表 3.4 创建菜品信息需求描述

名称	创建菜品信息
参与者	管理员
触发条件	发起创建请求
前置条件	菜品信息必要项不为空
后置条件	单项菜品信息添加到列表
优先级	高
正常流程	管理员发起添加请求，将菜品信息加入菜品列表并重新加载页面

创建菜品信息：当食堂即将提供新的菜品样式时，管理员将菜品信息录入数据库中进行增加，用户就可以看到新的菜品信息了。

表 3.5 修改菜品信息需求描述

名称	修改菜品信息
参与者	管理员
触发条件	发起修改请求
前置条件	菜品信息必要项不为空，将要修改的项进行修改
后置条件	修改后的菜品信息添加到列表
优先级	高
正常流程	管理员发起修改请求，将菜品信息加入菜品列表并重新加载页面

修改菜品信息：由于菜品可能会出现价格浮动、库存减少等情况，管理员可以通过修改菜品信息，让用户了解。

表 3.6 下架菜品信息需求描述

名称	下架菜品信息
参与者	管理员
触发条件	发起删除请求
前置条件	菜品状态处于再架上状态
后置条件	用户不再能浏览这一菜品信息
优先级	高
正常流程	管理员发起下架请求，将菜品状态改为下架重新加载页面

下架菜品：如果出现某一菜品因为某些原因，不再继续供应，为了避免造成用户错误下单，管理员应该将对应的菜品下架，设置为用户不可见的状态。

3.5 订单管理模块

此模块用于管理用户提交了的有效订单，当用户下单订餐以后，在系统生成有效的订单，管理员可以查看并且删除订单，用户可以创建、修改订单。

表 3.7 创建订单需求描述

名称	创建订单信息
参与者	用户
触发条件	发起创建订单请求
前置条件	余额足够、菜品库存足够
后置条件	订单生效、用户余额减少、库存减少
优先级	高
正常流程	判断余额余库存并进行相应减少，生成订单，重新加载页面

创建订单：当用户在选择好需要预定的菜品的时候，可以通过客户端确认下单按钮提交订单，在确认用户余额和菜品库存足够的情况下，订单将是有效订单。

表 3.8 修改订单需求描述

名称	修改订单信息
参与者	用户
触发条件	发起修改订单请求
前置条件	订单未进入处理环节
后置条件	订单生效、之前的订单失效
优先级	高
正常流程	判断订单状态，如果订单状态为未处理则进入修改状态，否则不能修改

修改订单状态：修改订单的权限只能用户拥有，而且只能是在订单提交以后，食堂未将订单设置为处理状态的情况下进行修改，这是为了避免用户随意修改订单给食堂管理工作造成麻烦。

表 3.9 查看订单需求描述

名称	查看订单信息
参与者	管理员、用户
触发条件	发起查看订单请求
前置条件	已经登录
后置条件	管理员取得所有用户订单列表，用户取得个人历史订单列表
优先级	高
正常流程	发起查看请求，管理员直接返回所有订单列表，用户返回个人订单列表

3.6 用户管理模块

此模块的功能访问者是食堂管理员，这是为了引入一种处罚机制，如果用户违背食堂订餐系统的某些规则，那么食堂管理员有权利禁用客户使用食堂订餐系统的功能，处罚期满以后食堂管理员再将用户设置为启用状态，用户才能继续使用食堂订餐系统。

表 3.10 禁用用户需求描述

名称	禁用用户
参与者	管理员
触发条件	发起禁用用户请求
前置条件	已经选择禁用的用户且该用户目前是启用状态
后置条件	该用户在启用之前将不能下单
优先级	中
正常流程	发起禁用已选择用户的禁用请求，如果该用户为启用状态，则设置为禁用

表 3.11 启用用户需求描述

名称	启用用户
参与者	管理员
触发条件	发起启用用户请求
前置条件	已经选择用户并且该用户处于禁用状态
后置条件	该用户设置为启用状态，可以正常使用订餐系统的功能
优先级	中
正常流程	发起启用已选择用户的启用请求，如果该用户为禁用状态，则设置为启用

3.7 食堂信息管理模块

此模块用于管理食堂信息和管理员信息，因为会存在食堂暂停营业、新增食堂或者某个食堂永久性停止营业等情况，也会存在管理员信息改变，删除管理员等情况，为了顾客可以准确的联系到管理员，应该保证管理员信息的准确性和时效性。

表 3.12 修改食堂信息需求描述

名称	修改食堂信息
参与者	管理员
触发条件	发起修改食堂信息
前置条件	已经选择需要修改的食堂
后置条件	食堂信息发生变更
优先级	中
正常流程	发起修改食堂信息请求，显示原有信息，提交修改，重新加载修改后信息

修改食堂信息是为了变更食堂名称、地址或者是否营业等，是否营业状态属性的及时显示有利于用户了解食堂的营业状态，避免造成用户花费不必要的时间到食堂就餐。

表 3.13 增加食堂需求描述

名称	增加食堂信息
参与者	管理员
触发条件	发起增加食堂信息
前置条件	
后置条件	新增食堂加入食堂列表
优先级	中
正常流程	发起新增食堂信息请求，确认提交食堂必要信息以后，重新加载食堂列表

表 3.14 修改管理员信息需求描述

名称	修改管理员信息
参与者	管理员
触发条件	发起修改信息请求
前置条件	
后置条件	管理员信息变革
优先级	中
正常流程	发起管理员信息变更请求，确认提交修改信息以后，重新加载管理员信息列表

修改管理员信息：管理员信息应该保证准确无误，如果发生变更，应该及时修改，这是为了顾客向食堂反馈提交反馈建议或者投诉时能够准确的将信息发到管理员邮箱或者手机。

表 3.15 新增管理员需求描述

名称	新增管理员信息
参与者	管理员
触发条件	发起新增管理员请求
前置条件	管理员必要信息无误
后置条件	新增管理员提交到列表
优先级	中
正常流程	发起管理员新增请求，确认信息提交新增信息以后，重新加载管理员信息列表

新增管理员信息：考虑的系统的可扩展性，未来单位内可能会新增其他食堂，这一需求是为了如果有新的食堂投入使用，可以及时降食堂信息添加进入系统。

3.8 系统设计

3.8.1 系统架构

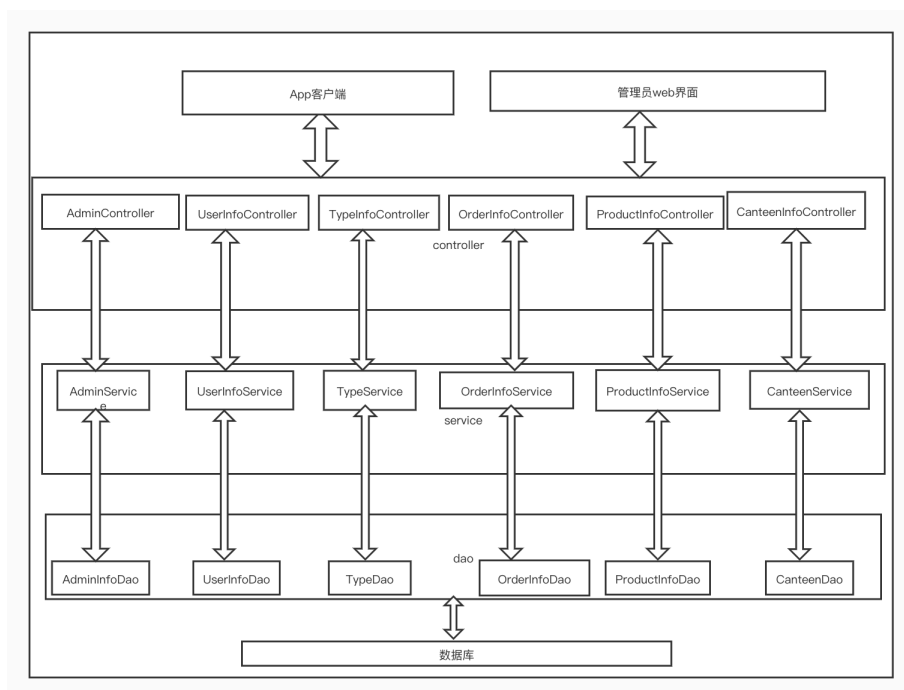


图 3-4 系统架构图

食堂订餐系统采用前后端分离的架构，前端、客户端和后端分开开发、测试和部署，后端对外提供统一的接口给前端访问，如果前端需要后端返回数据，后端会对数据统一封装为 json 格式以后返回，

前端准确来说分为管理员使用的 Web 界面和用户使用的订餐 App，Web 界面使用 Easy UI 开发，订餐 App 使用 Java 开发，都是通过 get/post 方法与后端发生交互。

后端使用 SpringBoot 框架开发，程序运行服务器为 tomcat，主体程序分为 controller 层，service 层和 dao 层，controller 层提供前端访问的接口，前端和客户端通过链接访问运行在服务器的程序，这些链接如果能与 controller 层的某个 controller 中的方法相匹配，那这个请求就会被这个方法处理，计算结果并返回所需数据。service 进行业务逻辑的处理，供 controller 层调用，dao 层访问数据库，由 service 层调用。使用三层架构的优势很多，首先使得合作开发成为可能，由于各层相互独立，一个小组只需负责一小块就可以，其次就是上一层依赖于下一层，如果测试下一层没有问题，那么问题就只可能发现在本层了，便于发现和改正 BUG 体现了“高内聚，低耦合”的思想，再有就代码的复用和劳动成本的减少。

3.8.2 系统功能模块图

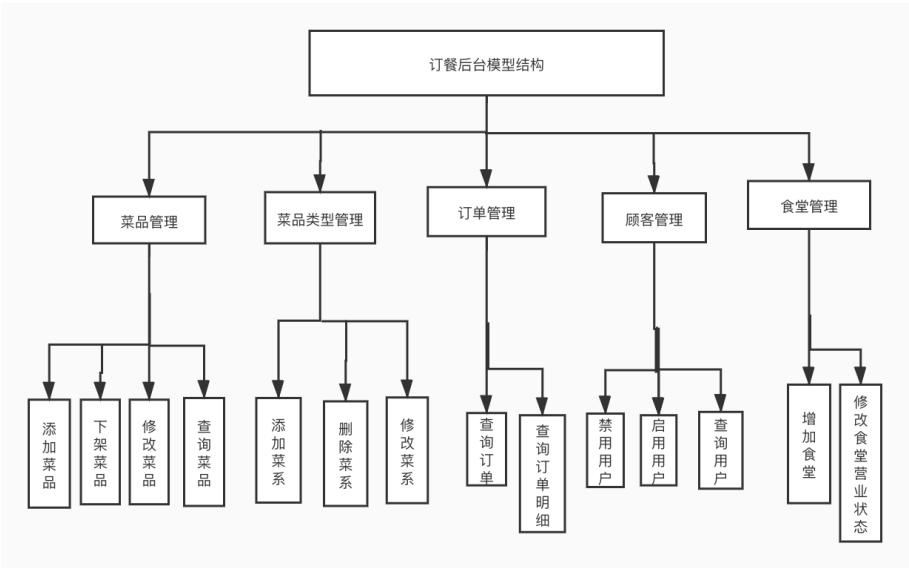


图 3-5 后端系统功能模块图

3.8.3 数据库设计

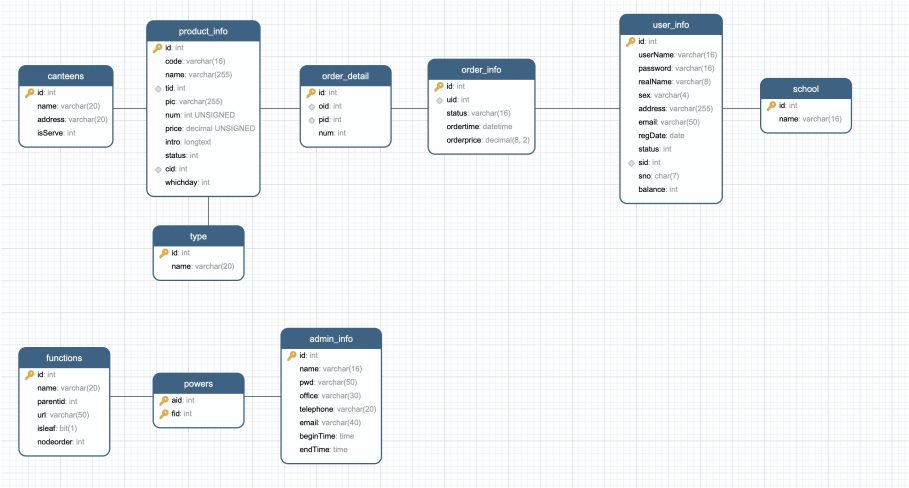


图 3-6 系统数据表关系图

院系表			菜品类型表			权限表		
字段名	数据类型	字段注释	字段名	数据类型	字段注释	字段名	数据类型	字段注释
id	int		id	int		aid	int	
name	varchar		name	varchar		fid	int	

食堂表

字段名	数据类型	字段注释
id	int	
name	varchar	
address	varchar	
isServe	int	

订单详情表

字段名	数据类型	字段注释
id	int	订单明细 id
oid	int	订单 id
pid	int	产品 id
num	int	购买数量

订单信息表

字段名	数据类型	字段注释
id	int	
uid	int	
status	varchar	
ordertime	datetime	
orderprice	decimal	

菜品信息表

字段名	数据类型	字段注释
id	int	
code	varchar	菜品编号
name	varchar	菜品名称
tid	int	菜品类别
pic	varchar	商品图片
num	int	菜品库存
price	decimal	菜品小图
intro	longtext	菜品简介
status	int	菜品状态
cid	int	
whichday	int	

用户信息表

字段名	数据类型	字段注释
id	int	
userName	varchar	
password	varchar	
realName	varchar	
sex	varchar	
address	varchar	
email	varchar	
regDate	date	
status	int	
sid	int	学院
sno	char	
balance	int	

管理员功能表

字段名	数据类型	字段注释
id	int	
name	varchar	功能菜单
parentid	int	
url	varchar	
isleaf	bit	
nodeorder	int	

管理员信息表

字段名	数据类型	字段注释
id	int	
name	varchar	
pwd	varchar	
office	varchar	
telephone	varchar	
email	varchar	
beginTime	time	
endTime	time	

四、系统功能模块的实现

五、总结和展望

六、实验与结果

参考文献

致谢

四年时间转眼即逝，青涩而美好的本科生活快告一段落了。回首这段时间，我不仅学习到了很多知识和技能，而且提高了分析和解决问题的能力与养成了一定的科学素养。虽然走过了一些弯路，但更加坚定我后来选择学术研究的道路，实在是获益良多。这一切与老师的教诲和同学们的帮助是分不开的，在此对他们表达诚挚的谢意。

首先要感谢的是我的指导老师王大明教授。我作为一名本科生，缺少学术研究经验，不能很好地弄清所研究问题的重点、难点和热点，也很难分析自己的工作所能够达到的层次。王老师对整个研究领域有很好的理解，以其渊博的知识和敏锐的洞察力给了我非常有帮助的方向性指导。他严谨的治学态度与辛勤的工作方式也是我学习的榜样，在此向王老师致以崇高的敬意和衷心的感谢。

最后我要感谢我的家人，正是他们的无私的奉献和支持，我才有了不断拼搏的信息的勇气，才能取得现在的成果。

王小明

2021 年 4 月 2 日

附录 A 补充更多细节

A.1 补充图

A.1.1 补充图

这是附录内容，应该用宋体小四号字体。



图 A-1 一个配有彩色表格的插图