



论文检测报告 (PDF版)

报告编号: D6A9097A2B2548CC8B4EACC2324C8C8A

送检文档: 基于javaSSM的微信点餐系统

论文作者: 林虎

送检单位: 湖北第二师范学院

文档字数: 17291

检测时间: 2019-04-10 23:39:25

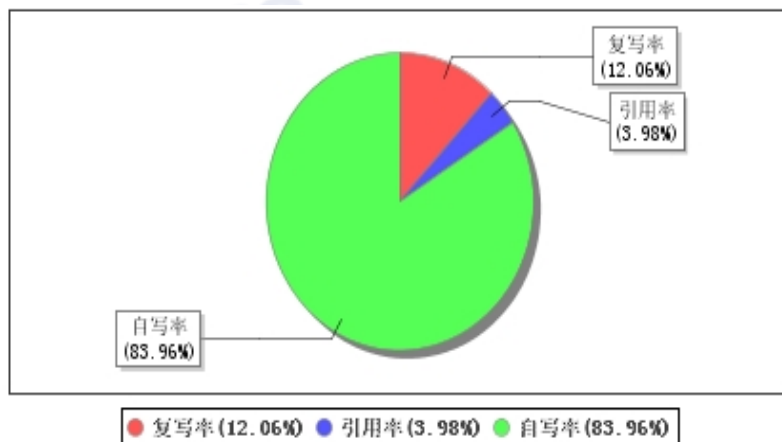
检测范围: 互联网, 中文期刊库 (涵盖中国期刊论文网络数据库、中文科技期刊数据库、中文重要学术期刊库、中国重要社科期刊库、中国重要文科期刊库、中国中文报刊报纸数据库等), 学位论文库 (涵盖中国学位论文数据库、中国优秀硕博论文数据库、部分高校特色论文库、重要外文期刊数据库如Emerald、HeinOnline、JSTOR等), 共享资源库, 参考文献库。

一、检测结果:

总相似比: 16.04% [即复写率与引用率之和]

检测指标: 自写率 83.96% 复写率 12.06% 引用率 3.98%

其他指标: 页数 (32) 图片数 (24) 表格数 (5) 引用句子数 (12) 公式 (0)



二、相似文献汇总:

| 序号 | 标题 | 文献来源 | 作者 | 出处 | 发表时间 |
|----|------------------------------|------|----|--------|------|
| 1 | 微信小程序getUserInfo解决方案-张小胖-博客园 | 互联网 | | 互联网 | 0 |
| 2 | 新经济时代企业信息系统开发和应用策略 | 学位论文 | 陈翔 | 硕博学位论文 | 2002 |
| 3 | 使用微信点餐系统,了解餐饮行业发展趋势_手机搜狐网 | 互联网 | | 互联网 | 0 |

| | | | | | |
|----|--|------|-----------------|------------|------|
| 4 | 餐饮与小程序走向新的模式!_网易订阅 | 互联网 | | 互联网 | 0 |
| 5 | 微信论文参考文献,参考文献_学术堂 | 互联网 | | 互联网 | 0 |
| 6 | Maven的几大优势!-Marcus丶的博客-CSDN博客 | 互联网 | | 互联网 | 0 |
| 7 | 拥有自己的外卖小程序,餐饮商家新方向-公众号+小程序一站式O2O... | 互联网 | | 互联网 | 0 |
| 8 | 基于微信公众平台的点餐系统的设计与实现_CNKI学问 | 互联网 | | 互联网 | 0 |
| 9 | 电子商务中的税收问题及其对策 | 学位论文 | 管浩 | 硕博学位论文 | 2005 |
| 10 | ...DictionaryServer):一个高性能的key-value数据库的..._CSDN博客 | 互联网 | | 互联网 | 0 |
| 11 | 云计算及应用的研究与实现 | 学术期刊 | 孙少陵 罗治国 徐萌 钱岭 王 | 电信工程技术与标准化 | 2009 |
| 12 | 2018年餐饮行业发展趋势如何?_手机搜狐网 | 互联网 | | 互联网 | 0 |
| 13 | SpringBoot-事件监听的4种实现方式 易学教程 | 互联网 | | 互联网 | 0 |
| 14 | 小程序的力量!_手机搜狐网 | 互联网 | | 互联网 | 0 |
| 15 | 深圳宝安小程序开发公司【紫鲸互联吧】_百度贴吧 | 互联网 | | 互联网 | 0 |
| 16 | 新型职业农民培育远程视频监管系统设计与实现 | 学术期刊 | 山红梅;陈曦;姚晨静;吴琦 | 中国市场 | 2016 |
| 17 | 网上花店管理系统的设计与实现 | 学位论文 | 张琪 | 硕博学位论文 | 2009 |
| 18 | 外卖领域国内外研究现状综述_毕业论文 | 互联网 | | 互联网 | 0 |
| 19 | 实现医保脱卡支付的医院手机APP设计与应用 | 学术期刊 | 陈辰;沈洁;汤振华;高喆;王诚 | 中国医疗设备 | 2017 |
| 20 | 2018年餐饮行业趋势【互联网头条】-中企动力 | 互联网 | | 互联网 | 0 |
| 21 | MVC架构在Asp.net中的应用和实现-随便记-CSDN博客 | 互联网 | | 互联网 | 0 |
| 22 | 基于微信小程序的点餐系统 | 互联网 | | 互联网 | 0 |
| 23 | 餐厅里吃出科技味_电脑报在线 | 互联网 | | 互联网 | 0 |
| 24 | 微信点餐系统对餐饮的优势_餐饮业_论坛_天涯社区 | 互联网 | | 互联网 | 0 |
| 25 | 基于微信公众平台的智能点餐系统的设计与实现-道客巴巴 | 互联网 | | 互联网 | 0 |
| 26 | 基于SpringMVC和Hibernate的企业人事管理系统-道客巴巴 | 互联网 | | 互联网 | 0 |
| 27 | 许昌微信小程序有什么优势许昌哪里可以制作小程序-资讯-新企报 | 互联网 | | 互联网 | 0 |
| 28 | 餐饮行业小程序,餐饮小程序为营销必备工具_Hi小程序 | 互联网 | | 互联网 | 0 |
| 29 | 浅谈从WebForm开发模式到ASP.NETMVC开发模式的发展 | 学术期刊 | 周泉 | 计算机光盘软件与应用 | 2012 |
| 30 | 舌尖上的创新与变革——2018新零售餐饮行业峰会_搜狐财经_搜狐网 | 互联网 | | 互联网 | 0 |

分类号
U D C

密级
编号

本科毕业论文(设计)

题目 基于javaSSM的微信点餐系统

学 院 计算机学院

专 业 名 称 物联网工程

年 级 2015级

学 生 姓 名 林虎

学 号 1550360012

指导教师名称 虞洋 孙延维

指导教师职称 工程师 副教授

二〇一九 年 四 月

湖北第二师范学院本科毕业论文(设计)原创性声明

本人郑重声明：所呈交的学位论文，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，本论文不含任何其他个人或集体已经发表或撰写过的作品成果。对本文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。本人完全意识到本声明的法律结果由本人承担。

论文作者签名： 年 月 日

摘要：在充分调查和分析现阶段中国餐饮行业的点餐方式和绝大多数是餐饮APP的运作流程的基础上，对于现代人的生活方式和传统APP的局限性进行分析。随着生活节奏的加快，有时候到餐厅点餐等待吃饭似乎不太尽人意，导致时间的浪费，不能提高餐桌的翻台率，工作效率的降低，未能提高营业额。而传统APP功能模块往往较多，难以适应针对性。针对这个现象，运用WEB开发技术，设计了基于微信小程序平台和B/S模式的点餐系统。该系统结合了微信小程序特有海量用户的优点，实现了用户登录、在线点餐，生成订单、用户评论、修改并管理收货地址等功能。同时后台也为商家提供了各种各样的商家服务，力求节约商家时间，降低餐厅成本，提高更多的利润。极大的便捷了人们的生活，也是未来信息化时代发展的大势所趋。

关键词：点餐 餐饮 java springBoot springMvc Mybatis 微信小程序

Abstract: On the basis of a full investigation and analysis of the ordering mode and the operation process of most catering apps in China's catering industry at the present stage, this paper analyzes the lifestyle of modern people and the limitations of traditional apps. With the acceleration of the pace of life, sometimes it seems to be unsatisfactory to go to the restaurant to order food and wait for dinner, which leads to the waste of time and cannot improve the table turnover rate, reduce the work efficiency and fail to improve the turnover. However, traditional APP function modules are often too many, and too many modules are not required by users, making it difficult to adapt to pertinence. In view of this phenomenon, using WEB development technology and springBoot background Mysql database, design based on WeChat small program platform and B/S mode ordering system. The system combines the advantages of weixin small program itself to realize the functions of user login, online ordering, order generation, user comments, modification and management of their own delivery address. At the same time, the background also provides businesses with a variety of business services, and strive to save business time, improve more profits. It greatly facilitates people's life and is the general trend of the development of the information age in the future.

Keywords: Order catering Java springBoot springMvc Mybatis WeChat small programs

目 录TOC \o "1-3" \h \u

第1章 引言1

1.1 选题背景1

1.2 国内外研究状况1

- 1.2.1 国外研究状况1
- 1.2.2国内研究状况2
- 1.3本文相关技术3
- 1.3.1java springBoot开发技术3
- 1.3.2微信小程序开发技术4
- 1.3.3 MVC后台开发模式4
- 1.3.4 Maven项目管理工具5

第2章 系统需求分析6

- 2.1 系统开发环境6
- 2.2系统需求6
- 2.2.1功能性需求7
- 2.2.2非功能性需求7
- 2.3本章小结8

第3章 系统详细设计与实现9

- 3.1系统网络拓扑结构9
- 3.2系统软件实现框架9
- 3.3功能模块实现10
- 3.3.1用户登录验证功能10
- 3.3.2菜单功能11
- 3.3.3购物车功能14
- 3.3.4订单模块14
- 3.3.5收货地址管理模块15
- 3.4后台页面的实现17
- 3.4.1商品管理17
- 3.4.2菜单管理18
- 3.4.3订单管理19
- 3.4.4老板查账19
- 3.4.5评价管理20
- 3.5后台数据库开发20
- 3.6本章小结23

第4章 系统测试24

- 4.1菜单模块测试24
- 4.2购物车功能测试：25
- 4.3收货地址管理测试25
- 4.4商家后台系统的测试26
- 4.5本章小结27

第5章 总结与展望28

- 5.1结论28
- 5.2展望28

参考文献30

致 谢31

第1章 引言

随着互联网的发展，信息技术的改革，互联网不再受时间和空间的限制。随着经济的发展，开始进入一个多元化的世界，标志着人类社会进入一个新的开始——电子商务时代。电子商务时代的到来，标志着一个依托互联网为基础的网上虚拟市场已经开始形成，这是一个具有全球性、数字化、跨时空等特点的飞速发展和潜力巨大的新兴市场。电子商务时代来临，对传统模式营销带来了冲击。人们开始在网上体验另一种购物的乐趣。

1.1 选题背景

随着市场经济的日益发展，百姓的生活水平愈来愈高，当今人们对饮食水平的要求也变得更更高了。尤其是随

着生活节奏的加快，许多顾客由于工作繁忙很少有空闲来餐厅享受美食。人们不仅仅局限于对饮食的种类和口味等方面的挑剔，对于就餐的时间、地点、环境和方式也提出了相对苛刻的要求[1]。这些需求促使网络电子商务和传统餐饮业发生了结合，也就是我们耳熟能详的互联网+的生产消费模式。

1.2 国内外研究状况

1.2.1 国外研究状况

国外餐饮品牌开展外卖业务是很容易接受的，外国人本身就崇尚快餐文化，绝大多数餐厅都是快餐餐厅。但是现在越来越多的传统美食也开始进行向外卖模式转变。无论是火锅还是佛跳墙，好像都和外卖这一概念根本搭不上任何关系，但是都和互联网密切合作，创造更大的效益。对于这些现象的研究也是越来越多，但是大多数是从消费需求和消费趋势来探索的，并没有从餐饮品牌建设角度来分析[2]。想要研究出与餐饮更加相符合的外卖管理策略，当然不能只研究成功品牌的餐饮业模式，毕竟其它很多类型的餐饮业因为自身的局限性而并不能走相同的成功之路。因此，必须综合各个方面，结合大小不同的餐饮企业，进行全方位的分析。传统餐饮业进驻外卖市场更具挑战性，同时也会更加艰辛。研究他们存在的问题对于在国内发展的餐饮互联网化的模式具有深远的重大意义。

其实在进行资料搜集过程当中，可以明显的发现国外关于外卖的研究很少，即使有研究其研究方向也是对于外卖餐饮的健康安全问题。从在国外的外卖市场就可以发现，国外以快餐业居多，而外卖服务对于外国快餐而言可以说是必不可少的。

1.2.2 国内研究状况

近年来，我国整个餐饮市场发展欣欣向荣，保持持续增长趋势。据相关报告数据显示：2016年全国餐饮收入实现35799亿元，较2015年增加3489亿元，同比增长10.8%，增速较快；2017年全国餐饮业收入超过3.9万亿元，同比增长10.70%左右[4]。

随着互联网的飞快发展，传统餐饮产业与互联网已经形成了不可分离的模式，发生了历史性的巨变。智慧化餐饮成为餐饮街的热门话题，许多商家为了谋求更好的发展都在各大外卖点餐平台或自己开展微信外卖等。各大商家转变经营发展模式，为顾客提供全新的点餐方式和用户体验。除了到店就餐，顾客还可选择更加方便的外卖方式，使用手机打开微信浏览菜单点餐、支付，坐等美食的到来，进而充分享受互联网带来的便捷和舒适的生活。

根据餐饮消费相关调查，可以明显发现，消费者更愿意去选择点评类网站、微信平台点餐，微信点外卖已然成为当今人们普遍接收并且经常使用的用餐方式。当下，餐饮商家应该把握机会紧跟时代步伐，坐上微信小程序的顺风车，全面发展智慧餐饮才能实现持续发展，选择微信系统再好不过！除了微信外卖，提供扫码下单、APP点单、移动收银等多种实用功能，满足您多种经营需求[4]。

1.3 本文相关技术

1.3.1 java springBoot开发技术

在当今社会，IT行业是一个发展迅速，更新频繁的行业。有Node JS、Ruby、Groovy、Scala、Python等很多种开发语言供该项目选择。并且每一种语言也有很多的技术框架或者技术支持可供选择。例如：就Python而言就有Django、Web.py、Flask等许多框架可以选择，但是该项目选择了java的SSM框架进行开发[5]。

那么至于为什么选择javaSSM是有很多方面的考虑：比如springBoot基层了Spring所有优秀的特点。并且SpringBoot需要的配置也是十分的简洁，不像是spring那样需要你对它有足够的了解才可以开发。SpringBoot可以采用零配置的注解开发模式进行开发。并且spring内嵌了servlet服务器，不像传统的web开发模式，需要配置Tomcat服务器才可以启动。并且自身内嵌的服务器要比配置的Tomcat启动要快。这样在开发期间可以节约服务器关闭/启动的时间。同样也方便我们的项目进行迭代开发。不断地扩展我们的项目功能[12]。

采用springBoot更因为它的部署也是十分方便快捷。只需要在服务器上安装所需要的JDK版本，然后将项目打包为War包，然后发表布到服务器上即可。同时springBoot也提供了非常强大的监控模式。其提供的监管模式有：

1. 手工向ApplicationContext中添加监听器方法2. 将监听器装载入spring容器模式3. 在application.properties中配置监听器模式4. 通过EventListener注解实现事件监听模式

这些模块的无不都是在为此项目保驾护航。并且由于其开源的维护模式，以及中国IT开发发现状模式——拥有大量的开发人员，方便解决项目在开发生产中的各种各样的问题[6]。

1.3.2 微信小程序开发技术

自从微信小程序问世以来，很多商家都在微信小程序这方面投入了巨大的心血。而且小程序也是十分受欢迎

，支付宝、各式浏览器也相继推出小程序。那微信小程序究竟有什么优势呢？

1、小程序与传统的手机APP是一种完全不同的体验，小程序无需下载安装，只需要打开微信然后找到相应的小程序即可使用，无需任何下载和安装，不占用手机一点内存。

2、开启方便，即用即走：用户扫一扫或者搜一下即可得到对应的小程序，用完退出即可，简单快捷的方式更容易提高使用率，增强用户体验。

3、功能更丰富：与传统的手机APP相比，它更轻便易用，与传统的服务号相比，它的功能也更丰富[11]。

4、系统迭代更新方便，无需像手机APP那样系统的更新必须重新下载对应的APP然后按照才可以体验最新版的体验。而小程序每一次的打开都是项目发布的最新版本。试想如此对比一下，哪一个更方便于用户，用户体验更好。

商户自营的餐饮小程序最大的优势是没有抽成。同时，小程序所有的后台操作都是自己管理，用户完全可以自己进行维护，不会遭遇“一言不合就下架”的尴尬和一些“店大欺客”的霸王条款。

只需要付出少量开发费用，同时无须添置额外硬件设备，中小餐饮商家就可以借助小程序这样的平台低成本实现加盟外卖平台一样的效果！餐饮的本质就如同是社交，小程序这样的平台将会使那些不重视社交的餐厅流失顾客，让那些重视社交的餐厅将获得巨大的红利。小程序会在客户最需要的时候出现，用完之后会自动消失，这对顾客来说是相当方便。用户的使用入口方便、操作简洁明确，体验感超好，你被认知的概率就越高，带来的利润也会越高。

1.3.3 MVC后台开发模式

在软件开发过程中将系统进行分层是一种常用的开发模式，软件系统的分层设计，例如目前应用广泛的MVC层结构可以高软件系统的开发效率，提高系统的稳定性和可维护性。MVC架构把数据处理，程序输入输出控制及数据显示分离开来，并且描述了不同部件的对象间的通信方式。使得软件可维护性，可扩展性，灵活性以及封装性大大提高；MVC(Model-View-Controller)把系统的组成分解为M（模型）、V（视图）、C（控制器）三种部件。优秀的MVC设计中，业务逻辑处理部分与显示部分逻辑上解耦，数据的获取通过接口实现，在不同的层中只完成各自的本质工作。良好的分层结构可以最大程度上简化后期系统升级维护的操作[7]。

1.3.4 Maven项目管理工具

另外此项目还采用了Maven工具对项目进行了管理，采用Maven的原因是：Maven让项目由之前的手动导入开发需要的各种各样的jar包转变为通过在pom.xml文件中的dependency标签中添加各种功能包。相比之前的手动导入jar,maven更能帮助我们解决各种各样的jar包之间的依赖和冲突。同时maven还可以小组之间建立私服务，然后使用通用的maven配置文件，不用手动去下载jar，pom文件会自动管理下载好的jar包。以上的种种有点都是为了让开发人员将所有的精力集中的专注于项目的业务逻辑开发上。

第2章 系统需求分析

2.1 系统开发环境

本文所开发设计的，微信点餐系统采用的是B/S结构。

本文设计的微信点餐系统的Web服务器和数据库，服务器都部署在同一个机器上。

选择这种开发场景是因为B/S架构要比C/S架构更加利于后期的维护或者迭代开发，不用每迭代一次就让客户重新安装一下我们的产品。而且这样的方式更容易方便我们的开发和维护[10]。

表2-1实际开发环境

操作系统

Window 10

CPU

8核2.5Hz及以上

内存

8G及以上

硬盘

128G及以上

Web服务器

Apache

数据库系统

Mysql+Redis

微信编辑器
微信开发者工具
数据库工具
Navcat10.2.1
Java开发工具
Eclipse Mar4, 5
测试工具

Android手机

2.2系统需求

软件的功能性需求分析就是分析在所有可能发生的情况下，当输入不断在可能取值间变化时，系统需要完成的功能和此时响应的效果是什么。软件的非功能性需求则是功能需求以外的要求，如软件的可靠性描述，软件的健壮性描述等等

微信点餐系统的需求较为复杂，需要同时考虑餐厅和消费者双方的需求结构本文采用了网上调研的方式分析消费者的需求，首先Internet调研，不同于传统调研方法，他没有时空和地域的限制，能够扩大信息收集的范围，获得更多的信息，其次，网上调研很好地解决了传统调研方法，中信息的实效性，具有信息的及时性和共享性，然后网上调研能够节约投入成本方便操作，且具有较强的准确性。对于餐厅方面的需求分析本文采用了与餐厅方面相关人员进行访谈与联合会议的方式来确认需求本文作者与餐厅相关负责人接触谈话，并商讨确认了某些特定业务，同时形成了系统的最终需求，将各个相关负责人召集在一起，联合讨论

2.2.1功能性需求

功能主要分为：客户对系统的使用要求和商家对系统的使用要求。

客户端的功能有：扫码登陆，菜单展示，菜品详情查看，查看菜品评价，对使用的菜品进行评价，购物车管理，订单(包含所有订单)查看，收货地址管理（地址的增加，删除，和修改）。

商家端后台的功能：账户密码登陆

商品管理（新增商品，查看所有商品）

菜单管理（菜单管理，分类管理）

订单管理（订单总览）老板查帐（收入统计）

评价管理（评价总览）

系统管理（店铺信息）

2.2.2非功能性需求

在系统工程及需求工程中，非功能性需求（Non-functional requirement）是指依一些条件判断系统运作情形或其特性，而不是针对系统特定行为的需求。和非功能性需求相对的是功能需求，后者会定义系统特定的行为或功能。非功能性需求也可以视为为了满足客户业务需求而需要匹配，但又不在于功能性需求以内的特性。

上面已经对整个系统工程有了详细的介绍，因此在非功能方面也是需要进行详细的需求设计，包含如下：

不论是用手机，还是用Pad(Android/IOS)都能够得到对应的微信小程序的前台页面。而且相应速度必须快，不能超过3秒，不能影响用户体验。

系统的前后台运行都必须要保证稳定，保持每年的服务器崩溃现象不超过两次。

前后台系统都要便于维护，可以进行迭代开发，可以为后期的业务扩展和新功能的开发提供良好的环境。

有良好的用户提示机制，方便用户学习和使用，做到用户一打开微信小程序就能明白小程序的各个功能。

并且做到对一些非本系统本身的意外情况做到预防，例如：租赁的服务器地区发生灾害迫不得已的宕机、部署的Redis服务集群出现断电情况等等。

2.3本章小结

本章节对项目的开发环境，以及项目的部署环境都进行了详细的分析和设计，分析了本项目的功能性需求和非功能性需求，综合商家和客户对微信点餐进行了各个方面的分析和把控。完成效果还是比较切合实际情况。

第3章 系统详细设计与实现

3.1系统网络拓扑结构

图3-1系统网络拓扑图

客户使用手机或者平板电脑上的微信APP，扫一扫点餐小程序对应的二维码，通过访问微信官方服务器获得

我们对应的微信小程序。然后通过点餐小程序请求开发者服务器得到商家的菜单。通过浏览和选择进而生成不同的订单。

3.2 系统软件实现框架

图3-2 系统软件实现框架图

3.3 功能模块实现

微信小程序前台的业务逻辑图：

图3-3 微信小程序前台的业务逻辑图

3.3.1 用户登录验证功能

本文所设计的微信点餐系统是基于微信小程序进行开发，本系统并未提供及维护自己的系统用户表，而是根据微信的 `getUserInfo()` 接口，进而获取用户的相关信息。只保留了微信提供的很少的一部分关于用户信息标识的数据。微信提供了两种获取用户信息的接口，一种是无感知的获取的信息较少（不需用户手动授权），另一种是有感知的获取的数据较多（需要用户手动点击页面授权，后台才有获取信息的权限）。

微信获取用户信息的代码实现：

```
getUserInfo: function (cb) {
  var _this = this
  if (this.globalData.userInfo) {
    typeof cb == "function" && cb(this.globalData.userInfo)
  } else {
    //调用登录接口
    wx.getUserInfo({
      withCredentials: false,
      success: function (res) {
        _this.globalData.userInfo = res.userInfo
        typeof cb == "function" && cb(_this.globalData.userInfo)
        console.log(_this.globalData.userInfo, 1)
      }
    })
  },
},
```

3.3.2 菜单功能

相关功能的代码实现：

```
@RequestMapping("/getAllMenu")
@ResponseBody
public Result getAllMenu() {
  List<Menu> allMenu = menuService.allMenu();
  return Result.ok(allMenu);}
public List<Menu> allMenu() {
  MenuExample example = new MenuExample();
  return menuMapper.selectByExample(example);
}
```

微信代码实现：

```
_this.setMenu()
app.getUserInfo(function (userInfo) {
  _this.setData({
    userInfo: userInfo,
    img: userInfo.avatarUrl,
    nickName: userInfo.nickName
  })
})
```

图3-4 微信小程序菜单显示效果图

菜单是将商家所有的菜品类目录入到开发者的服务器当中，通过微信小程序发生HTTP请求

127.0.0.1/menu/getAllMenu得到相应的数据，并通过微信小程序进行解析，用户可以通过上下滑动来查看不同的类目的商品分类，亦可以通过点击不同的商品进而查看相应的商品的详情和相关评价，进而对该商品有十分充分的了解。

图3-5 查看商品详情及评价的页面

详情及评价的代码实现：

```
onLoad: function (option) {  
  //根据菜的id请求菜的详细信息  
  wx.request({  
    url: app.getHeader() + '/disk/getDiskById',  
    data: {  
      diskId: option.id  
    },  
    header: {  
      'content-type': 'application/json' // 默认值  
    },  
    success: function (res) {  
      if (res.data.status == 200) {  
        console.log("请求菜的详情", res.data)  
        disk: res.data.data  
      } else {  
        app.showToast(res.data.msg)  
      }  
    },  
    fail() {  
      app.showToast("菜详情请求失败!")  
    }  
  })  
  //根据菜的id请求菜的评价  
  url: app.getHeader() + '/evaluate/getEvalsByDiskId',  
  console.log(res.data)  
  evals: res.data.data  
}  
},  
fail() {  
  app.showToast("评价请求失败!")  
} })
```

3.3.3购物车功能

当用户浏览菜单的时候，通过点击每一个菜品的右下角的加号，来添加到购物车当中。并且可以通过点击界面上对应的加号和减号来修改购物车当中对应所选的商品的数目。并可以对购物车中不同的商品数量以及不同的商品价格进行计算，产生不同的订单价格。

图3-6购物车对应的效果图

3.3.4订单模块

订单模块也是相对重要的一个模块，当用户将自己需要点的菜品放到购物车之后用户可以在购物车中对相应的商品进行付款，进而生成相应的订单，订单的状态又分为（未付款，交易完成，已取消）分别对应不同的后台操作状态，当订单为支付完成时会通知上商家后台，提醒商家及时的对订单进行相应的操作，比如制作菜品，配送菜品，完成菜品订单等一系列操作。当商家后台完成所有操作之时订单的状态就变成了交易完成的状态。

图3-7 订单模块的相关效果图

3.3.5收获地址管理模块

众所周知，客户的所在地可能是随时随地都在变化的，例如，一个客户可能在公司，医院，家中或者该客户想为另一个客户（比如自己的朋友、亲人）点餐亦可以在我们的微信点餐系统中进行各种各样的收餐地址的管理，当然我们的添加地址的方式也是十分方便快捷的，可以通过手动输入的方式进行添加收货地址，也可以通过调用地图接口的方式来添加收货地址。

图3-8 管理收货地址的功能页面

图3-9 地图定位获得地址功能页面

3.4后台页面的实现

店内点餐系统的主要业务逻辑：

后台的需求在很多地方和微信小程序的需求是不一样的。但是后台必须保证有十足的安全性能。必须是可以保证我们的后台只有我们的商家可以进行访问。同时需要保证能够给系统的小程序用户提供强大的数据传输功能，避免出现掉线，延时的一些造成不良影响的情况。其主要的逻辑及流程如图所示：

图3-10 商家后台业务逻辑流程图

3.4.1 商品管理

商品管理模块是对后台数据库中所有存在的菜品的管理，将所有数据都查出来然后通过前台的html渲染显示，当选择对应的商品时可以进行编辑、删除等操作。当并未选择相应的商品时可以进行添加商品操作。并且本系统将增、删、查、改的所有操作的出发事件都放在网页的最上端，方便商家的使用，提升用户体验。

图3-11 商品管理的页面效果图

3.4.2 菜单管理

菜单管理：在很多商品之中可能会在地域方面、或者口味方面存在相似之处，因此为了让商家更好的对菜品进行管理，让客户拥有更好的用户体验。因此很多设计了菜单管理功能。商家可以通过这个菜单管理迅速找出一类有共性的商品。方便自己管理。

图3-12 菜单管理页面效果图

3.4.3 订单管理

订单管理是为了方便商家及时地对前台小程序生成的订单进行及时处理，避免出现漏单，延误等现象的出现。

每个订单都有不同的状态，其包含：未付款、已付款、已接单、已发货、已完成、已取消。其中未/已付款是由前台小程序来控制的。但是订单的发货状态是由商户控制。商户可以对相应的订单进行接单、发货、查看、退单等操作。

图3-13 订单管理页面的效果图

3.4.4 老板查账

老板查账是特地为商家所设计的，商家可以通过系统的后台展示界面，选择餐厅正常营业的一段时间，可以得到这段时间的所有收入。方便用户的账务管理。

图3-14 老板查账功能效果图

3.4.5 评价管理

评价模块有助于用户掌握用户对各个菜品的喜好，帮助商家初步确定采购食材的数量，更有利于商家知晓自己的哪些菜品是好吃的，干净的，符合广大消费者的，哪些菜品是还需要不断提高不断加强的。评价管理模块首先是将所有评价展示，其实商户也可以通过选择好评、中评、差评这样的标签来区分不同的评价。

图3-15 评价管理功能效果图

3.5 后台数据库开发

数据库对一个系统来说是至关重要的，一个优秀的系统必定会有一个完美的数据库设计方案。为了系统的开发和运维更加贴近功能需求，做了如下的数据库设计。

由于内存数据库Redis，运行以及存取速度快，可以将内存中的数据保存在磁盘中。Redis不仅仅支持Key-value类型的数据，也同样支持list、set、hash等数据结构的数据存储。因此本项目才用了Redis作为缓存机制，来完成项目的购物车功能。

数据库表设计，如图3-16所示：

图3-16 数据库设计表

用户收货地址表，保存用户的收货地址，如图3-17所示：

图3-17 address表设计

菜品信息表，保存菜品信息，如图3-18所示：

图3-18 dish表设计

菜品评价表，保存用户对菜品的评价，如图3-19所示：

图3-19 evaluate表设计

菜品类目表，保存菜品的各种分类，如图3-20所示：

图3-20 menu表设计

订单地址表，保存所生成的点单对于的收货地址，如图3-21所示：

图3-21 order_address表设计

订单内容表，保存订单有哪些菜品的表，如图3-22所示：

图3-22 order_item表设计

3.6本章小结

本章对微信点餐项目的前台小程序和后台管理系统的功能模块都进行了详细的设计和划分。然后在功能划分的基础上，对每个模型进行了相应的系统模型设计以及数据库设计。用我最大的能力和专业技术水平实现各个模块的分里解耦，以希望创建一个更优秀、更健壮的系统。

第4章 系统测试

系统测试是软件开发环境中非常重要的一个环节，软件测试的主要目的是检查软件系统设计实现过程中呈现的问题，系统测试主要包括1、制定测试计划，2、编写测试用例，3、手工或自动化方法进行单元测试，4、记录并分析测试结果。首先，在系统测试前要根据目前项目的开发进度，系统部署上线的时间，系统部署环境与系统用户群体，拟定详实可靠的测试计划，根据系统将来的部署与运行环境，要求系统测试环境与之一致。例如操作系统的版本，服务类型等。其次，根据系统功能设计文档与系统模块间接口规范文档编写测试用例。测试用例要覆盖系统功能，尽量全的覆盖输入数据可能的类型包括控制边界值与异常指导输入，然后测试人员要严格按照测试用例完成系统各个功能模块的测试，记录模块输入输出结果，对模块间接口进行测试，测试接口间的联系，调用是否正确，最后详细记录测试结果分析并重现异常，结果分析其可能的原因。

4.1菜单模块测试

对菜单模块进行单元测试，测试用例与测试报告设计如下：

表4-1菜单模块测试表

测试用例

菜单模块测试

测试目的

测试菜单模块是否正常工作

测试过程

预计输出

实际输出

获取菜单页面

流畅高效获取并展示

可以畅高效获取并展示

菜目类筛选不同菜品

即时跳转，提升用户体验

能够通过用户对小程序的点击，对菜单的显示做出及时的调整。

点击菜品查看详情

可以及时跳转到菜品详情页面得到展示。

可以流畅的跳转到菜品详情页面。

点击菜品查看评价

获得对应菜品的相关所有用户评价

通过用户点击跳转到菜品的评价显示页面。通过后台访问得到所有关于当前商品的评价，并美好的展示在页面上。

4.2购物车功能测试：

对于微信端进行单元测试，测试用例与测试设计报告如下：

表4-2微信购物车功能测试表

微信端购物车功能测试

4.3收货地址管理测试

为了方便客户的使用和提高用户的用户体验，我们特地也在本项目做了收获地址管理的相关功能模块。为了保证此功能模块可以正常的运行和使用对其进行来相关的测试。测试用例与测试结果如下表所示：

表4-3收获地址管理测试表

微信端收获地址管理功能测试

测试收获地址管理模块是否正常工作

获取收获地址管理页面

响应地址编辑页面

可以快速响应，并对相关页面进行即时处理

添加新的收获地址

可以添加成功，并有相应的交互效果。

通过调用地图的接口方便快捷的实现收获地址的编辑和添加。

调用地图的接口正常

4.4 商家后台系统的测试

商家的后台虽然不如小程序那样需要极高的用户体验，以及交互能力，但是我们仍然需要保证我们所设计的系统的稳定性。因此我们对这个后台系统进行了整体化的测试。

表4-4商家后台系统测试表

点餐商家后台物测试

测试商家后台是否正常工作

对各个功能模块的页面进行，访问浏览

流畅高效获取并有效展示，

可以流畅高效获取并有效展示，拥有良好的使用体验，无延时。不存在页面崩溃或者页面找不到的情况。

展示所有菜品分类

即时展示，并且不存在卡顿现象。

能够流畅的展示菜品类目界面，抗议查看，编辑，管理所包含的菜品的功能。

查看所有菜品页面

可以即时响应，完成正常流程。

可以流畅的跳转到菜品详情页面。对响应的菜品进行查看编辑，预览等功能。

对订单进行管理

与微信小程序有及其紧密的联系，检查稳定性

商家对订单进行操作的时候及时同步到微信小程序。客户随时知道订单状态。

4.5 本章小结

通过这个测试环节，对于我们系统的功能模块的健壮性有很显著的提高，这就如同上学期间老师对我们的一次的摸底考试，让我对自己设计和开发出来的系统的业务逻辑方面和功能健壮性方面都有更加全方位的了解。知道自己系统在哪些方面更具优势，在哪些方面可能不是那样强大。

第5章 总结与展望

5.1 结论

随着，社会的进步，人类文明的推进，物质文化生活的不断提高。并且很多年轻人并不擅长也不愿意去做饭，故越来越多的人愿意去餐馆去享受美餐。而且微信平台的用户量是十分巨大的，当然依托于微信小程序平台本身就让我们的产品在某种意义上是倍具优势的[9]。当然，如今是一个互联网+的世界模式，很多商户也是十分愿意和互联网相融合，能够乘坐互联网的顺风车。力求将线上与线下服务结合在一起，留住已经拥有的消费者，挖掘潜在的客户，提高自己的营业额。

微信点餐的消费者都是使用微信的用户，用户使用系统完成自助点餐。微信点餐系统是客户和商家之间的联系桥梁，消费者可以通过微信自助选择菜品，提交订单，商家通过后台进行接单，制作，配送等交易流程。微信点餐系统能够为用户和商家提供全新的用户体验，同时节约客户点餐时间，降低商家的成本投入。

5.2 展望

本文所开发的微信公众平台的智能点餐系统设计与实现虽然能够为餐厅提供一种新的经营模式，为消费者提供新的就餐体验，但是由于个人开发水平有限，本微信点餐系统仍有很多不足和需要改进的地方。

随着科技的进步和餐厅经营理念的发展越来越多的餐饮企业开始寻求互联网信息化的有利帮助，跳出传统基因管理模式的方法，增强自身的竞争力，但是从长远来看，项目许多功能仍需移植移动化和多样化发展，下面对餐饮企业发展及微信点单系统的未来发展趋势提出以下几点展望：

餐饮企业为了提高自身的竞争力，将会有更多的餐厅开发自己的微信点餐系统平台。通过自己的维护，可以得到用户更多关于用餐的习惯和喜好数据。对于这些数据进行专业的分析，结合自身餐饮的优势推出客户更喜爱

的用餐模式或是对自己的不足进行弥补。进而获得更大的商业价值,推动社会的发展和进步。

点餐无线管理将是趋势,点餐系统接入互联网和移动互联网实现预约点餐催单结账的一条龙服务。随着科技的进步和餐厅经营理念的发展,使得未来餐厅点餐系统的服务将不仅仅只限于买卖食品,更多的混合经营模式将是餐饮运营企业的趋势[12]。

参考文献杨文博. 餐饮品牌的移动传播研究[D]. 湘潭大学,2017. 9-21

张涛. 必胜客在中国的营销战略制定研究[D]. 华东理工大学,2011. 20-78

刘国才. 不断创新的《中小学电教》[J]. 中小学电教,2003(11):7-11

刘广煌. 门禁一卡通系统行业现状及发展趋势[J]. 中国公共安全,2013(10):87-89.

Yu S Z. Drinking Water and Primary Liver Cancer[M]. Beijing: Beijing Springer, 1992:30~37.

郑杰, 钱育蓉, 范迎迎. 基于Spring MVC和Hibernate的企业人事管理系统. 电脑知识与技术, 2016. 92-141

JoshuaBloch. EffectiveJavaProgrammingLanguageGuide. Addison-WesleyProfessional. 2001 201-231

古丽敏. 餐饮服务类微信公众平台微交互设计研究[D]. 江南大学, 2016.

汪文君. Java高并发编程详解. 机械工业出版社. 2018 143-176

[10]严梅、高冠聪. 基于微信公众平台开发的大学生生活圈应用设计与实现[J]. 河北企业, 2016, 12:260-261.

[11]姜承尧. 高性能网站MySQL数据库实践. 维普中文科技期刊数据库. 2013 187-231

[12]京北方发布云测平台,打造一站式综合测试云平台服务[J]. 中国金融电脑,2018(08):89.

致 谢

在此,我要向在我制作毕业设计过程中帮助过我的同学和导师表示感谢,在这个过程中,导师帮助我选毕业设计的课题,耐心的给我做详细的讲解,经常询问我的进度与成果,有困难的时候给我点拨,帮助我顺利完成,没有他的指导,也不会有我今天说展示出来的成果。

然后是我的同学们,他们的集思广益,在我的制作过程中,提出各种各样的、好的、可用的想法帮助我查漏补缺,给我鼓励,其中,在这里我要着重感谢与我们学校合作的公司——武汉市软酷网科技有限公司。在我的整个项目中发挥到了举足轻重的作用,项目想法的产生、项目的立项、项目的需求分析、项目的需求分析、项目的系统详细设计与代码实现等等,都给予了我莫大的帮助和指导。

另外,我要感谢我自己,如若没有自己的辛苦付出,没有自己的检查不懈,没有自己绞尽脑汁加班加点的完成需求,我想项目也不可能完成的如初顺利。

四、指标说明:

1. 总相似比即类似于重合率。总相似比即送检论文中与检测范围所有文献相似的部分(包括参考引用部分)占整个送检论文的比重,总相似比=复写率+引用率。
2. 引用率即送检论文中被系统识别为引用的部分占整个送检论文的比重(引用部分一般指正确标示引用的部分)。
3. 自写率即送检论文中剔除雷同片段和引用片段后占整个送检论文的比重,一般可用于论文的原创新颖性评价,自写率=1-复写率-引用率。
4. 复写率即送检论文中与检测范围所有文献相似的部分(不包括参考引用部分)占整个送检论文的比重。
5. 红色字体代表相似片段;浅蓝色字体代表引用片段、深蓝色字体代表可能遗漏的但被系统识别到与参考文献列表对应的引用片段;黑色字体代表自写片段。

五、免责声明:

鉴于论文检测技术的局限性以及论文检测样本库的局限性,Gocheck.cn网站不保证检测报告的绝对准确,相关结论仅供参考,不做法律依据。

Gocheck论文检测服务中使用的论文样本,除特别声明者外,其著作权归各自权利人享有。根据中华人民共和国著作权法相关规定,Gocheck网站为学习研究、介绍、评论、教学、科研等目的引用其论文片段属于合理使用。除非经原作者许可,请勿超出合理使用范围使用其内容和本网提供的检测报告。