

# 基于Vue.js的智能匹配系统的设计与实现

作者姓名\_\_\_\_\_张培妍\_\_\_\_\_

学校导师姓名、职称\_\_\_\_\_李青山 教授\_\_\_\_\_

企业导师姓名、职称\_\_\_\_\_周璟 高工\_\_\_\_\_

申请学位类别\_\_\_\_\_工程硕士\_\_\_\_\_



学校代码 10701

分 类 号 TP311.5

学 号 1610122639

密 级 公开

# 西安电子科技大学

## 硕士学位论文

### 基于Vue.js的智能匹配系统的 设计与实现

作者姓名：张培妍

领 域：软件工程

学位类别：工程硕士

学校导师姓名、职称：李青山 教授

企业导师姓名、职称：周璟 高工

学 院：计算机科学与技术学院

提交日期：2019 年 6 月



# **Design and Implementation of Intelligent Matching System Based on Vue.js**

A thesis submitted to  
XIDIAN UNIVERSITY  
in partial fulfillment of the requirements  
for the degree of Master  
in Software Engineering

By

Zhang Peiyan

|                         |                        |
|-------------------------|------------------------|
| Supervisor: Li Qingshan | Title: Professor       |
| Supervisor: Zhou Jing   | Title: Senior Engineer |

June 2019



**西安电子科技大学**  
**学位论文独创性（或创新性）声明**

秉承学校严谨的学风和优良的科学道德，本人声明所呈交的论文是我个人在导师指导下进行的研究工作及取得的研究成果。尽我所知，除了文中特别加以标注和致谢中所罗列的内容以外，论文中不包含其他人已经发表或撰写过的研究成果；也不包含为获得西安电子科技大学或其它教育机构的学位或证书而使用过的材料。与我一同工作的同事对本研究所做的任何贡献均已在论文中作了明确的说明并表示了谢意。

学位论文若有不实之处，本人承担一切法律责任。

本人签名：\_\_\_\_\_ 日 期：\_\_\_\_\_

**西安电子科技大学**  
**关于论文使用授权的说明**

本人完全了解西安电子科技大学有关保留和使用学位论文的规定，即：研究生在校攻读学位期间论文工作的知识产权属于西安电子科技大学。学校有权保留送交论文的复印件，允许查阅、借阅论文；学校可以公布论文的全部或部分内容，允许采用影印、缩印或其它复制手段保存论文。同时本人保证，结合学位论文研究成果完成的论文、发明专利等成果，署各单位为西安电子科技大学。

保密的学位论文在\_\_\_\_年解密后适用本授权书。

本人签名：\_\_\_\_\_ 导师签名：\_\_\_\_\_

日 期：\_\_\_\_\_ 日 期：\_\_\_\_\_





## 摘要

在如今移动互联网发展的鼎盛时期，微信走入了千家万户，云服务为小型企业的发展带来颇多助益，所有的社会活动都被注入了信息时代的元素，使用信息技术手段辅助完成活动的策划与实施的智能活动匹配系统已经成为中小型商家发展客户关系的得力工具。近年来，商家举办活动越来越依靠计算机、互联网等现代化手段，五湖四海的人相聚在网上，依据不同的规则被划分成不同的组，共同参与感兴趣的活动。在信息爆炸和快节奏的新时代，人工统计和手动分组已经无法取得令人满意的效果。不仅如此，一般匹配系统的算法在设计开发之后无法自行调整，当举办不同的活动需要更改匹配规则时，重新开发活动匹配系统所产生的高昂费用令不少中小型商家望而却步。

在现今计算机飞速发展的潮流下，针对人工统计步骤繁琐、效率低、出错率高、随机率低、活动效果未知和活动数据难以复盘等传统活动匹配方式的不足，本文提出了适合中小型商家进行活动运营的智能匹配系统。本系统不仅提供了商家基本的活动宣传、报名、分组、举办等功能，还提供了活动数据的统计分析、活动表单一键生成和活动评价等功能。最重要的是，本系统为商家提供方便快捷的智能活动表单生成功能，选择合适的信息项、海报图片和匹配规则，可为不同活动生成不同报名表单，本系统一经开发，无需修改代码，表单模板可重复用于多次活动。本系统的设计与实现工作是在充分参阅技术文档前提下得以开展的，在系统的具体设计实现工作中采用了 Vue.js 前端框架、Java 开发技术以及 MySQL 数据库技术，在论文中对所涉及到的理论和技术进行了介绍与分析；对智能匹配系统所涉及到的业务流程进行详细的调研与分析，阐述各个业务内容与工作流程，搭配用例图来辅助说明，对系统正常运行所需满足的功能要求和性能要求进行详细的分析与说明，完成了系统的整体需求文档。在需求文档的指导下，对系统进行详细设计与实现工作，首先对组织结构、业务流程和数据库进行设计，详细阐述了系统的架构设计工作，然后对系统的核心功能模块逐一进行介绍，对系统的技术实现工作进行具体论述，最后对系统的具体设计与实现工作进行简要总结。根据系统实际运行环境对测试环境进行搭建和配置，列举说明了软硬件配置具体参数，编写各个功能的测试用例并执行功能测试，使用 LoadRunner 工具进行性能测试，对测试结果进行分析，考察系统是否实现了预期的功能目标和非功能目标。

对本系统进行分析可得，系统的主要用户角色包括活动参与者、商家以及系统管理员。参与者端即用户端，具有活动报名、活动分享、活动评价、活动查询等功能；商家端即为系统后台，主要包括活动管理模块、用户管理模块、选项库管理模块、商

家管理模块和统计管理模块;系统管理员对数据库、日志和密码进行管理。除此之外,本项目将数据库托管至阿里云服务器上,服务器端主要负责数据的查询、处理和存储。项目的后台使用 Java 语言进行开发,前端页面采用 Vue.js 框架进行编写,对微信公众平台做了相应的开发和连接,使用阿里云数据库实现对数据的高效存储且免于后期维护。

**关 键 词:** Vue, 匹配系统, 客户关系管理系统, 云平台

## ABSTRACT

In the heyday of the development of mobile Internet, WeChat has entered thousands of households. Cloud services have brought a lot of benefits to the development of small enterprises. All social activities have been injected into the elements of the information age, using information technology to assist in the completion. The intelligent activity matching system for the planning and implementation of activities has become a powerful tool for small and medium-sized businesses to develop customer relationships. In recent years, merchants have increasingly relied on modern means such as computers and the Internet. People from all corners of the world have gathered on the Internet and are divided into different groups according to different rules to participate in activities of interest. In the new era of information explosion and fast-paced, manual statistics and manual grouping have not been able to achieve satisfactory results. Moreover, the algorithms of the general matching system cannot be adjusted after the design and development. When the different activities need to change the matching rules, the high cost generated by the re-development of the activity matching system has discouraged many small and medium-sized businesses.

Under the current trend of rapid computer development, this paper proposes suitable small and medium-sized businesses for the cumbersome manual statistical steps, such as cumbersome steps, low efficiency, high error rate, low randomness rate, unknown activity effect and difficulty in replaying activity data. An intelligent matching system for performing activities. The system not only provides basic activities such as promotion, registration, grouping, and hosting of the business, but also provides statistical analysis of activity data, single key generation of activity tables, and activity evaluation. The most important thing is that the system provides merchants with convenient and quick intelligent activity form generation function, select appropriate information items, poster pictures and matching rules, and can generate different registration forms for different activities. Once the system is developed, no need to modify the code, the form Templates can be reused for multiple activities. The design and implementation of this system is carried out under the premise of fully referring to the technical documentation. The Vue.js front-end framework, Java development technology and MySQL database technology are used in the specific design and implementation of the system. The theory and technology are introduced and analyzed; the detailed research and analysis of the business processes involved in the

intelligent matching system, the description of each business content and workflow, and the use case diagram to assist in the description, the system needs to meet the normal operation. The functional requirements and performance requirements are analyzed and explained in detail, and the overall requirements document of the system is completed. Under the guidance of the requirements document, the system is designed and implemented in detail. Firstly, the organizational structure, business process and database are designed. The architecture design of the system is elaborated. Then the core functional modules of the system are introduced one by one. The technical implementation work is discussed in detail, and finally the specific design and implementation of the system is briefly summarized. According to the actual operating environment of the system, build and configure the test environment, enumerate the specific parameters of the software and hardware configuration, write the test cases of each function and perform the function test, use the LoadRunner tool to perform the performance test, analyze the test results, and check whether the system is implemented. Expected functional and non-functional goals.

The analysis of the system is available, the main user roles of the system include activity participants, merchants and system administrators. The participant end is the user end, which has the functions of activity registration, event sharing, activity evaluation, activity query, etc. The merchant side is the system background, which mainly includes the activity management module, the user management module, the option library management module, the merchant management module and the statistical management. Module; system administrator manages databases, logs, and passwords. In addition, the project will host the database to the Alibaba Cloud server, which is responsible for data query, processing and storage. The background of the project is developed using Java language. The front-end page is written by Vue.js framework. The WeChat public platform is developed and connected accordingly. The Alibaba Cloud database is used to realize efficient storage of data and avoid post-maintenance.

**Keywords:** Vue, Matching System, Customer Relationship Management system, Cloud Platform

## 插图索引

|        |                 |    |
|--------|-----------------|----|
| 图 2.1  | MVVM 框架图.....   | 7  |
| 图 2.2  | 微信公众号交互模式图..... | 11 |
| 图 3.1  | 系统总体用例图.....    | 14 |
| 图 3.2  | 用户管理用例图.....    | 15 |
| 图 3.3  | 活动管理用例图.....    | 16 |
| 图 3.4  | 选项库管理用例图.....   | 17 |
| 图 3.5  | 统计管理用例图.....    | 17 |
| 图 3.6  | 商家管理用例图.....    | 18 |
| 图 3.7  | 系统管理用例图.....    | 19 |
| 图 4.1  | 系统架构图.....      | 23 |
| 图 4.2  | 系统功能结构图.....    | 25 |
| 图 4.3  | 系统网络拓扑图.....    | 26 |
| 图 4.4  | 系统 E-R 图 .....  | 27 |
| 图 4.5  | 用户管理类图.....     | 31 |
| 图 4.6  | 用户管理时序图.....    | 32 |
| 图 4.7  | 活动管理流程图.....    | 33 |
| 图 4.8  | 活动管理类图.....     | 34 |
| 图 4.9  | 活动管理时序图.....    | 35 |
| 图 4.10 | 选项库管理流程图 .....  | 36 |
| 图 4.11 | 选项库管理类图.....    | 37 |
| 图 4.12 | 选项库管理时序图 .....  | 38 |
| 图 4.13 | 商家管理流程图 .....   | 39 |
| 图 4.14 | 商家管理类图 .....    | 40 |
| 图 4.15 | 商家管理时序图 .....   | 41 |
| 图 4.16 | 统计管理流程图 .....   | 42 |
| 图 4.17 | 统计管理时序图 .....   | 43 |
| 图 4.18 | 系统管理模块类图 .....  | 44 |
| 图 5.1  | 测试工作流程.....     | 48 |
| 图 5.2  | 用户信息管理界面.....   | 51 |
| 图 5.3  | 活动发布界面.....     | 53 |
| 图 5.4  | 选项新增窗口.....     | 54 |

|       |              |    |
|-------|--------------|----|
| 图 5.5 | 商家注册界面 ..... | 56 |
| 图 5.6 | 统计管理界面 ..... | 57 |
| 图 5.7 | 密码管理窗口 ..... | 58 |

## 表格索引

|       |                    |    |
|-------|--------------------|----|
| 表 2.1 | 自建数据库与云数据库对比表..... | 11 |
| 表 4.1 | 商家信息表.....         | 28 |
| 表 4.2 | 参与者信息表.....        | 28 |
| 表 4.3 | 活动信息表.....         | 28 |
| 表 4.4 | 选项信息表.....         | 29 |
| 表 4.5 | 规则项信息表.....        | 29 |
| 表 4.6 | 活动评价信息表.....       | 30 |
| 表 4.7 | 系统日志表.....         | 30 |
| 表 5.1 | 服务器端测试环境.....      | 47 |
| 表 5.2 | 用户端测试环境.....       | 47 |
| 表 5.3 | 用户管理测试用例表.....     | 50 |
| 表 5.4 | 活动管理测试用例表.....     | 51 |
| 表 5.5 | 选项库管理测试用例表.....    | 53 |
| 表 5.6 | 商家管理测试用例表.....     | 55 |
| 表 5.7 | 统计管理测试用例表.....     | 56 |
| 表 5.8 | 系统管理测试用例表.....     | 57 |
| 表 5.9 | 系统性能测试结果表.....     | 59 |





## 符号对照表



## 缩略语对照表

| 缩略语  | 英文全称                             | 中文对照       |
|------|----------------------------------|------------|
| API  | Application Programming Interfac | 应用程序编程接口   |
| CRM  | Customer Relationship Management | 客户关系管理     |
| CSS  | Customer Service and Support     | 客户服务和支持    |
| DW   | Data Warehouse                   | 数据仓库       |
| ERP  | Enterprise Resource Planning     | 企业资源计划     |
| MVVM | Model-View-ViewModel             | 模型-视图-视图模型 |
| SFA  | Sales Force Automation           | 销售能力自动化    |



# 目录

|                                      |     |
|--------------------------------------|-----|
| 摘要 .....                             | I   |
| ABSTRACT .....                       | III |
| 插图索引 .....                           | V   |
| 表格索引 .....                           | VII |
| 符号对照表 .....                          | IX  |
| 缩略语对照表 .....                         | XI  |
| 第一章 绪论 .....                         | 1   |
| 1.1 研究背景与意义 .....                    | 1   |
| 1.2 国内外研究现状 .....                    | 3   |
| 1.3 论文工作内容 .....                     | 5   |
| 1.4 论文组织结构 .....                     | 6   |
| 1.5 本章小结 .....                       | 6   |
| 第二章 基础理论与技术 .....                    | 7   |
| 2.1 Vue.js+Spring Boot 前后端分离框架 ..... | 7   |
| 2.1.1 Vue.js 框架概述 .....              | 7   |
| 2.1.2 Spring Boot 概述 .....           | 8   |
| 2.2 MVC 设计模式 .....                   | 9   |
| 2.3 微信公众平台 .....                     | 10  |
| 2.4 阿里云数据库 .....                     | 11  |
| 2.5 本章小结 .....                       | 12  |
| 第三章 需求分析 .....                       | 13  |
| 3.1 系统总体需求 .....                     | 13  |
| 3.1.1 需求背景 .....                     | 13  |
| 3.1.2 业务概述 .....                     | 13  |
| 3.2 系统功能需求分析 .....                   | 14  |
| 3.2.1 用户管理分析 .....                   | 14  |
| 3.2.2 活动管理分析 .....                   | 15  |
| 3.2.3 选项库管理分析 .....                  | 17  |
| 3.2.4 统计管理分析 .....                   | 17  |
| 3.2.5 商家管理分析 .....                   | 18  |
| 3.2.6 系统管理分析 .....                   | 19  |

|       |                    |    |
|-------|--------------------|----|
| 3.3   | 系统非功能性需求分析 .....   | 19 |
| 3.3.1 | 系统性能需求 .....       | 19 |
| 3.3.2 | 系统安全需求 .....       | 20 |
| 3.4   | 本章小结 .....         | 21 |
| 第四章   | 设计与实现 .....        | 23 |
| 4.1   | 系统总体设计 .....       | 23 |
| 4.1.1 | 系统架构设计 .....       | 23 |
| 4.1.2 | 系统功能结构设计 .....     | 24 |
| 4.1.3 | 系统网络拓扑设计 .....     | 25 |
| 4.2   | 系统数据库设计 .....      | 26 |
| 4.2.1 | E-R 模型设计 .....     | 26 |
| 4.2.2 | 数据字典设计 .....       | 27 |
| 4.3   | 系统功能设计与实现 .....    | 30 |
| 4.3.1 | 用户管理模块设计与实现 .....  | 30 |
| 4.3.2 | 活动管理模块设计与实现 .....  | 32 |
| 4.3.3 | 选项库管理模块设计与实现 ..... | 36 |
| 4.3.4 | 商家管理模块设计与实现 .....  | 39 |
| 4.3.5 | 统计管理设计与实现 .....    | 42 |
| 4.3.6 | 系统管理设计与实现 .....    | 43 |
| 4.4   | 本章小结 .....         | 45 |
| 第五章   | 系统测试与分析 .....      | 47 |
| 5.1   | 系统测试环境 .....       | 47 |
| 5.2   | 测试流程与方法 .....      | 48 |
| 5.2.1 | 测试流程 .....         | 48 |
| 5.2.2 | 测试方法 .....         | 49 |
| 5.3   | 系统功能测试 .....       | 50 |
| 5.3.1 | 用户管理模块测试用例 .....   | 50 |
| 5.3.2 | 活动管理模块测试用例 .....   | 51 |
| 5.3.3 | 选项库管理模块测试用例 .....  | 53 |
| 5.3.4 | 商家管理测试 .....       | 55 |
| 5.3.5 | 统计管理测试 .....       | 56 |
| 5.3.6 | 系统管理测试 .....       | 57 |
| 5.4   | 系统非功能测试 .....      | 58 |
| 5.5   | 本章小结 .....         | 59 |

|                 |    |
|-----------------|----|
| 第六章 总结及展望 ..... | 61 |
| 参考文献 .....      | 63 |
| 作者简介 .....      | 67 |





## 第一章 绪论

随着互联网的飞速发展，越来越多的信息充斥在复杂的网络上，人们可以很方便地通过互联网获取到各类信息，显然，互联网已经成为人们日常生活中一个重要的信息交互平台。

### 1.1 研究背景与意义

当今社会生活节奏快，互联网将四面八方的人们联系在一起，社交不再受地域的局限，人们可以与远在千里之外的陌生人进行交谈。打破了地域和时空的限制，互联网给社交活动中的分组匹配环节增加了难度。随着活动在网上的飞速传播，许多不同地域、不同时期的人们报名参加同一项活动，人工统计和分组显然不能满足随机和准确的要求，并且效率极低，极有可能导致活动无法如期举行，人工通知和真人客服在参与者大量涌入的新时期所暴露出来的弊端更加使得手工统计、人工分组的活动举办方式显得落后和不合时宜。

为了经营会员，大型商家只能选择付出高昂的费用自行开发活动管理系统<sup>[1]</sup>，对举办的活动以及参与的会员进行有效地管理，后期运营维护耗费人力物力财力，由于受到商家计算机专业知识的制约，往往效果欠佳，久而久之这种方式逐渐被放弃。中小型商家无力承担巨额开发运维费用，只能使用人工统计和店员宣传的方式去反馈老会员和吸引新会员，这种方式耗费人力，耗时较长，举办一次活动往往需要极长时间策划准备和大量人力投入，严重影响正常经营活动，并且由于效率低、出错高导致反响不佳。人工统计效率低、自行分组出错频繁、宣传渠道窄、缺乏数据分析无法复盘这些问题都影响了中小型商家自行举办活动。具体来说，对于中小型商家而言，举办活动的传统方式分为以下几个环节：

(1) 事前宣传：海报粘贴、传单印发。中小型商家没有专用的会员活动系统，宣传活动时大概率会选择将活动内容印刷成海报张贴在店外，或者印制传单雇佣店员进行派发。这种宣传方式耗费人力且传播范围有限，虽然付出不菲的费用，但是宣传效果不佳，性价比极低；

(2) 活动报名：在传统的活动举办模式中，客户参与活动需要填写报名表，商家手动进行登记，使用 Excel 等办公软件进行汇总统计，需要店员掌握一定的办公软件使用技能，并且人工统计耗时长，出错难以避免；

(3) 活动安排：所谓活动安排，是指将报名成功的成员按照规则进行分组、编号和排序。中小型商家在人工统计后，活动安排往往也选择了人工完成，分组规则形同

虚设，分组结果受到报名时间和空间的影响非常明显，无法保证分组的随机性和匹配的准确性；

(4) 活动统计：在活动结束后，使用传统活动举办方式的中小型商家难以进行活动复盘。由于活动各个环节均由人工完成，耗费时间久，活动记录材料往往遗漏缺失，几乎无法存档，活动举办效果无从得知。

随着移动互联网的普及，移动互联网企业争抢流量入口之战愈演愈烈，微信以“连接一切”的目标定位，坐拥全球超 10 亿用户<sup>[2]</sup>，已经成为现代生活中最为重要的社交工具之一。现如今，不少企业都拥有微信公众平台，不但在宣传产品和树立品牌方面起到了不可或缺的作用，而且能够为维护用户关系提供强有力的帮助，能够帮助企业更好地实现客户运营目标。微信公众平台已然成为各商家、组织与个人竞相使用的宣传发布平台和用户运营渠道。微信为商家带来了客流，为了维系与客户之间的关系，微信平台为各个商家提供了方便快捷的消息提醒、客服回复、自定义菜单和投票统计等非常实用的功能，微信公众号通过对消息、用户和素材三个方面的管理实现了操作快捷、简单易用的基本后台管理功能。

微信渗透进人们生活的方方面面，不少商家都使用微信群、微信公众号、微信小程序来提供更优惠的商品和更方便的服务，拉近与客户之间的距离，从而完成日常客户运营工作。微信公众平台普及后，相比微信群而言更适合商家进行商品和活动的宣传、树立品牌、进行投票调研以及对会员进行有效管理<sup>[3]</sup>。随着现代生活节奏越来越快，耗费人力和时间的传统活动举办方式已被淘汰。活动参与者希望活动报名便捷、活动进度可查询、能够发表评价增加参与感；活动举办者希望活动传播范围更广、活动分组更准确、活动安排效率更高、活动举办效果可统计。在此情况下，开发一个智能匹配系统来辅助完成活动举办可以满足上述要求，成本低廉，效果显著，为中小型商家举办活动提供了最佳解决方案。

以前，为了经营会员，大型商家只能选择自行开发活动管理系统对举办的活动以及参与的会员进行有效地管理，前期高昂资金投入，后期运营维护耗费人力物力财力，并且受到商家计算机专业知识的限制，往往效果欠佳，反而为商家造成了经济上的压力，这种方式性价比实在不高。中小型商家无力承担巨额开发运维费用，只能使用人工统计和店员宣传的方式去反馈老会员和吸引新会员，这种方式耗费人力，耗时较长，举办一次活动往往需要极长时间策划准备和大量人力投入，严重影响正常生产和经营活动，并且由于效率低、出错高经常导致反响不佳。人工统计效率低、自行分组出错频繁、宣传渠道窄无人参与、缺乏数据分析难以评估活动效果这些问题都阻碍了中小型商家自行举办活动，基于微信公众平台的智能匹配系统恰到好处地解决了这些问题。

## 1.2 国内外研究现状

多规则可变的活动匹配系统实质上是客户关系管理系统的一部分。国外 CRM<sup>[4]</sup>（Customer Relationship Management，客户关系管理）包含了对客户和活动的管理，功能点与本系统所涉及的成员管理、统计管理、活动管理等高度一致，故本系统重点参考 CRM 系统的发展历史、现状和趋势。国内关于活动匹配系统在企业生产经营活动中的研究和应用并不多见<sup>[5]</sup>，参考资料和文献甚少。根据对 CRM 系统国内外发展现状的分析可以得知，活动匹配系统的研究和发展在国内尚存在较大探索空间，尤其对于中小企业来说，在移动互联时代研究和发展基于微信公众平台的多规则可变的活动匹配系统对于活动运营和客户运营是十分必要的，也是提高企业核心竞争力的重要手段。本节对智能匹配系统相关的国内外研究现状进行详细介绍和说明。

本系统属于 CRM 系统，旨在实现活动与客户信息的管理。在激烈的市场角逐中，许多国外企业发现，客户关系是销售活动中极为重要的一环<sup>[6]</sup>，于是，销售自动化系统（SFA）于 1990 年诞生，在此基础上，专注于客户服务的重要工具客户服务系统（CSS）也应运而生。1996 年，在上述这两个系统的基础上，企业增加了营销与服务等内容，丰富了客户关系管理系统的功能，CRM 初具雏形。CRM 是用户运营的重要手段，包括了客户的获取、保持和增加的过程。企业将员工、技术和流程三者结合起来，将各个领域的业务和服务内容进行合理有效的集成，为企业维护客户关系提供了花费小、效率高的解决方案。不仅如此，CRM 实现一对一的客户服务和营销模式，让企业能够为客户提供最佳体验，保持良好的客户满意度，促进普通用户向忠实用户的转化，探索目标用户，吸纳对企业最有价值的客户群体。

CRM 的概念一经提出，就被许多企业竞相探索开发，市场获得了前所未有的开拓与惊人的增长速度。目前来说，国外具有代表性的 CRM 软件主要有 Siebel 公司开发的 Siebel System，SAP 公司开发的 My SAP CRM，Oracle 公司开发的 Oracle CRM 和微软公司开发的 Microsoft CRM 等产品。

值得注意的是，对于 CRM 系统，国外企业的研究方向主要是将 CRM 系统与其他生产经营活动中使用的软件相结合<sup>[7]</sup>，例如将 CRM 与电子商务软件相结合；将 CRM 和数据仓库（DW）、数据挖掘软件相融合，将 CRM 和 ERP 系统进行集成等等。与这些先进技术和生产软件的结合，使得 CRM 系统得到进一步的发展和壮大，应用更为广泛。

国外的 CRM 研究经过多年发展，已经处于相对比较成熟的时期，他们所开发的 CRM 软件具备以下几点优势：

(1) 实力雄厚，投入了大量人力物力培养优秀的技术人员，进行强有力的市场推广；

(2) 与知名硬件厂商、大型数据库公司,咨询行业知名公司进行合作,保证了 CRM 系统的良好发展;

(3) 国外 CRM 的设计思路受到先进的西方管理理念的影响,与市场的发展保持良好的协调性;

(4) 国外 CRM 概念提出较早,软件发展时间长,较为完善和成熟,为不同行业都能够提供行之有效的 CRM 产品;

(5) 国外的 CRM 系统不断探索与其他生产经营软件的集成,例如 ERP、SCM 系统,使得 CRM 应用更加灵活和广泛,突出其优势;

(6) 国外 CRM 软件积极寻找与互联网技术结合的可能,投入大量的人力物力进行研究,大大提高了 CRM 系统的易用性。

总而言之,国外 CRM 拥有起步早的优势<sup>[8]</sup>,历经多年试错与检验,不断发展与改进,在具体管理和实际使用方面积累了丰富的经验,更新换代诸多版本之后,如今的 CRM 已经非常成熟,有完善齐全的功能模块,具备很强的可扩展性,同时有较高的稳定性,是国内 CRM 的前辈。

然而,对于国外成熟的 CRM 的引进却不尽人意。由于国内在 CRM 系统的研究方面起步晚,自从 1999 年,CRM 进入中国,十几年的发展并没有让 CRM 成为真正意义上客户关系管理的一大帮手,迟迟无法融入中国市场,反而使得 CRM 一度被贴上了“中看不中用”的负面标签。除了银行系统、房地产行业和保险市场,CRM 在其他领域屡屡碰壁,与被大众所认可接受的 ERP 系统相比,CRM 高达 45% 的失败率一度使其发展停滞不前。

国外 CRM 软件向国内的引入存在一些阻碍:

(1) 国外 CRM 的许多功能在国内并不适用,引进的 CRM 无法真正落地为国内企业所用;

(2) 国外 CRM 软件价格高昂,对国内企业造成较大的经济压力;

(3) 国外 CRM 产品深受西方管理理念影响,与中国国情和本土风俗难免产生冲突,为实施企业可能带来负面影响;

(4) 由于计算机水平落后,国内技术力量无法支持 CRM 系统的研究工作。

现阶段,国内 CRM 系统的发展还处于初步的探索中,CRM 系统的市场尚未形成规模,没有得到良好的发展,无论是在行业前景、产品内容,还是销售渠道等方面来说都并不完善,不过,随着经济发展,企业销售模式与管理理念发生变化,我国的 CRM 系统也有望获得进一步发展的机会,日趋成熟。如今,国内对于 CRM 系统的研究重点主要是将客户关系管理与数据挖掘相结合,探索更精准可靠的客户关系管理系统。随着国内计算机技术水平稳步提升,对 CRM 重新重视,投入人力物力进行研究,国内的 CRM 系统也会迎来飞速发展并应用于各行各业的春天。

总而言之，用户运营是商业活动中不可或缺的一环，随着经济的发展，更多的企业将意识到使用客户关系管理系统的重要性，CRM 将成为商业竞争的下一个发展热潮。纵观 CRM 整个发展过程，CRM 未来的发展趋势如以下八点所述：

- (1) CRM 与云服务的结合将得到进一步发展；
- (2) 界面将成为 CRM 系统发展的重点；
- (3) CRM 将发展为全面的信息系统；
- (4) CRM 与其他办公系统合作，能够更好地完成企业生产活动；
- (5) CRM 将拥有对更多信息类目的管理功能和更灵活的使用场景；
- (6) CRM 的社交化属性在互联网时代将得到更好的发展；
- (7) 移动互联网时代的 CRM 给客户和员工提供更多功能；
- (8) CRM 将会具备根据使用者反馈及时作出调整的灵活性。

### 1.3 论文工作内容

本文对智能匹配系统的设计与实现工作做了详细的论述，在做了充分的研究准备工作后，查阅基础理论与技术文档，根据前后端分离的设计思想，使用 Vue.js 和 Spring Boot，采用 MVC 三层结构进行设计，使用 Java 语言完成编码开发，旨在为中小型商家举办活动提供多规则可变、效率高、信息化存档、活动效果可视化的智能匹配系统。论文主要进行了以下几方面内容的研究工作：

(1) 理论技术资料收集工作：本系统的设计与实现工作是在充分参阅技术文档前提下得以开展的，在系统的具体设计实现工作中采用了 Vue.js 前端框架、Java 开发技术、微信公众平台以及阿里云服务器，在论文中对所涉及到的理论和技术进行了介绍与分析；

(2) 需求分析工作：对智能匹配系统具体业务进行详细调研与分析，阐述各个业务内容与工作流程，搭配用例图来辅助说明，对系统正常运行所需满足的功能要求和性能要求进行详细的分析与说明，总结系统的整体需求文档。

(3) 设计与实现工作：按照系统需求分析文档完成系统的详细设计与实现工作，包括系统的总体设计、数据库设计和功能模块设计，通过对整体架构、组织结构和功能结构进行设计，简要介绍了系统的总体设计，其次通过列举数据表结构设计，对系统的数据库设计进行充分介绍，最后介绍各个功能模块的具体设计和技术实现工作。

(4) 测试工作：根据系统实际运行环境对测试环境进行搭建和配置，介绍了系统测试的软硬件环境，编写各个功能的测试用例并执行功能测试，使用 LoadRunner 工具进行性能测试，对测试结果进行分析，考察系统是否实现了预期的功能目标和非功能目标。

## 1.4 论文组织结构

根据智能匹配系统的工作内容，本文一共分为六章：

第一章，绪论。该章节主要介绍了智能匹配系统的研究背景，对研究目的和意义进行阐述，通过讨论与匹配系统相关的 CRM 系统在国内外发展现状，分析了发展趋势，最后给出了论文的工作内容，简要说明了论文的组织结构。

第二章，基础理论与技术。首先对智能匹配系统实现工作中所涉及的技术概念进行总体说明，然后对智能匹配系统构建所参考和使用的技术和理论分别作出简要介绍。

第三章，智能匹配系统的需求分析。该章节首先通过分析匹配系统的业务背景阐述了总体业务需求，主要对智能匹配系统的功能需求进行了分层次、有条理、深入细致的研究分析，接着通过功能结构图介绍业务模块，然后详述具体功能，配合对应的用例图进行了详尽的说明，并对性能需求和安全需求进行了补充说明，通过介绍详细的评判指标进行分析说明。

第四章，智能匹配系统的设计与实现。本章介绍了本文所研究的系统的架构设计与技术实现，其中总体设计包括整体架构、组织结构、业务流程三个方面，然后阐述了数据库的设计工作，最后对系统功能模块的设计实现工作进行介绍。

第五章，智能匹配系统的测试。本章介绍了本文所研究系统的测试工作，首先完成软硬件配置，搭建测试环境，接着编写各功能点的测试用例，然后执行功能测试和非功能测试，最后记录测试结果并进行分析，验证本系统是否满足预期的功能和性能需求。

第六章，工作总结及展望。该章节主要对智能匹配系统设计开发工作进行总结，并对今后工作作出规划和展望。

## 1.5 本章小结

本章介绍了论文的准备工作的准备工作，对系统的研究背景与意义、国内外研究现状和论文工作内容进行了充分的考察与研究，在此基础上完成了论文的组织结构安排，为后续工作梳理出基本思路作为指导和参考。

## 第二章 基础理论与技术

本章考察了系统开发所需基础理论与技术资料文档,对所使用的技术特性进行简明扼要的整理分析<sup>[9]</sup>,为系统后续的设计与实现工作提供必要且可靠的理论保障,本系统所使用的基础理论与技术包括 Vue.js 技术、Spring Boot 框架、MVC 三层架构、微信公众平台和阿里云数据库,本章对上述理论与技术进行介绍。

### 2.1 Vue.js+Spring Boot 前后端分离框架

智能匹配系统的开发采用前后端分离的思想<sup>[10]</sup>,在前端开发中使用 Vue.js 框架,后端使用基于 Java 语言的 Spring Boot 框架完成系统的整体实现<sup>[11]</sup>。本节对 Vue.js 和 Spring Boot 进行基本介绍,并对其应用于本系统开发工作的特点与优势进行说明。

#### 2.1.1 Vue.js 框架概述

Vue.js 是一个用来开发 web 界面的前端库,也是一个轻量级开发工具。Vue.js 不但属于响应式编程,还具有组件化的特点,恰如官方网站所言:通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。在 Vue.js、AngularJS 和 ReactJS 等前端 MVVM 框架都实现了使视图与状态保持一致性<sup>[12]</sup>(这种数据的同步更新即为数据绑定),MVVM 可以拆分成:Model---ViewModel---View 三部分,如图 2.1MVVM 框架图所示:



图2.1 MVVM 框架图

在图 2.1 中,左侧的 Model 相当于数据对象,右侧的 View 相当于页面 DOM 内容,中间用 ViewModel 进行连接,相当于页面的展示逻辑<sup>[13]</sup>。使用中间的监控者 ViewModel 对两侧的数据和操作进行监控,一旦一方发生改变,就通知相应的另一方完成数据的修改。Vue.js 使得前端开发人员不必再编写复杂的 DOM 操作通过 this 来回寻找相关节点,大大提高了开发的效率。通过 MVVM 框架<sup>[14]</sup>,自动完成了数据和视图的同步更新,并且与其他前端框架相比,Vue.js 非常简单易上手,没有太多新概念,学习成本较低,在对实例 `new Vue({data:data})` 进行声明后 data 中的数据自动与视图进行绑定,一旦 data 中的数据发生变化,视图中相应数据会随之更改。

Vue.js 的“一切皆为组件”理念,是指任意完成封装的代码块都可以被看作组件,

例如: `Vue.component('exa',exa)`, 能够以 `<exa></exa>` 的形式在模板中调用。进行合理的抽象, 就可以将网页划分成一个个区块, 区块由大大小小的组件组成。

除此之外, `Vue.js` 可以与其他工具搭配使用, 例如 `vue-resource` 和 `vue-router`, 对路由和异步请求提供支持, 这样就可以完成单页面的开发工作。

`Vue.js` 以易上手、轻量级的优点为人称道, `MVVM` 的开发模式为前端人员免去了原先冗余繁重的 `DOM` 操作, 代码更加易于维护, `Vue.js 2.0` 版本已经发布, 借助轻量级的 `virtual-DOM` 实现页面渲染<sup>[15]</sup>, 使得在大部分使用场景中初始化页面渲染速度可以提升 2~4 倍。作为非常轻量级的前端框架, 在前后端分离的项目中能够最大程度上发挥 `Vue.js` 的优势和特性, 本项目既有移动端也有 `PC` 端, `Vue.js` 在移动浏览器上压缩后只有 18KB 且没有其他的依赖, 相较于擅长构建 `APP` 的 `AngularJS` 和需要本地开发人员的 `ReactJS`, `Vue.js` 结合了它们的优势—灵活、简单和轻量, 更适合 `Web` 开发, 所以选择 `Vue.js` 最为合适。

### 2.1.2 Spring Boot 概述

新兴的动态语言衬托出使用 `Java` 语言进行开发工作的诸多不便: 配置繁多、开发效率低、部署流程复杂并且与第三方技术难以集成<sup>[16]</sup>。

`Spring Boot` 的出现解决了上述 `Java` 开发的难题。它带来了全新的配置理念, 在项目中已经预设了足够多的配置, 除此之外还为开发者内置了习惯性配置<sup>[17]</sup>, 免去手动配置的工作, 使得项目能够快速部署运行。这就是它所遵循的“习惯优于配置”。使用 `Spring Boot` 可以简单快速地开发一个内嵌 `Servlet` 容器、以 `jar` 包形式独立运行、准生产级别的项目, 使用 `Spring Boot` 的开发人员仅需要很少甚至无需进行 `Spring` 配置。

`Spring Boot` 发布于 2014 年 4 月, 截止 2014 年底, 在发布后仅半年时间, 使用 `Spring Boot` 作为 `Spring` 开发方案的项目已高达 34.1%。

`Spring Boot` 具有以下特征:

- (1) 根据“习惯优于配置”理念, 使用 `Spring Boot` 仅需进行少量配置<sup>[18]</sup>, 在大部分使用场景中默认配置完全可以满足需求。`Spring Boot` 根据类路径为 `jar` 包中的类进行自动配置相应的 `Bean`, 节省了开发人员的配置工作;
- (2) 快速搭建项目, 整合第三方框架时不必进行配置;
- (3) 简化 `maven` 配置, 使用 `starter pom` 简化了 `maven` 依赖;
- (4) 不使用 `xml` 配置, 仅通过 `Java Config` 和默认的习惯性配置, `Spring Boot` 通过条件注解而不是代码来完成配置;
- (5) 内嵌 `Servlet` (例如 `Tomcat`) 容器, 项目以 `jar` 包形式 (`java -jar`) 运行;
- (6) `Spring Boot` 能够基于 `http`、`ssh`、`telnet` 对运行中项目进行应用状态监控。



Spring Boot 是一个完全基于 Spring 的应用，没有使用别的技术，为 JavaEE 开发带来了类似于脚本语言开发的效率。对于本项目而言，快速构建、无配置集成、可独立运行、提供运行时监控、与云计算的天然集成等优点都使得 Spring Boot 成为不二之选。

## 2.2 MVC 设计模式

本文设计实现的智能匹配系统采用了 MVC 三层架构的设计模式<sup>[19]</sup>。MVC 设计思想与本项目前后端分离思路高度一致，将系统业务流程中数据从数据库到逻辑处理再到页面展示的过程进行合理拆分，分为三个部分：Model（模型层）、Controller（控制器）和 View（视图层）。Model 在 Java Web 里指的是 JavaBean<sup>[20]</sup>，除了为 Java 对象处理业务逻辑外，还负责进行数据访问，Java 对象有属性字段和行为方法；View 指的是负责展示的前端页面，在 Java Web 项目中一般由 JSP 充当 View 角色；Controller 指的是转发请求的中间层，将 Model 与 View 进行匹配，接收来自 View 层的输入，转交给 Model 层去处理，将返回的处理结果转发给 View 层，实现了控制器基本的转发功能，在 Java Web 项目中一般由 Servlet 充当 Controller 角色。

在 MVC 设计模式中，Web 应用根据不同的业务层次被划分成 Model、View 和 Controller 三层，详细说明如下：

(1) Model 层：模型层是 Web 项目中用于处理数据逻辑的部分，包括业务逻辑处理和数据访问，通常涉及到数据库中数据的读写操作<sup>[21]</sup>。模型层包括了项目的业务逻辑、数据处理规则 and 所有应用数据的读写访问。在 MVC 的三层结构中，Model 层所负责的处理任务最多，当业务流程被抽象为数据的流转后，应用程序中的业务逻辑处理和数据访问工作都由 Model 层来参与完成，例如对 Java 对象方法的调用、数据在数据库中的读取与写入操作等，这些操作都是在模型层中完成，通过设定好的业务逻辑对数据进行相应的处理。Model 层使用 Components 构件对象来处理数据库，由于 Model 层不负责展示，无需对数据进行格式处理，所以得到的数据是中立的，与模型无关，可以用于多个视图，增强了代码的复用性，提高了开发效率，降低了代码的维护难度。

(2) View 层：视图是与用户交互的界面，主要功能是展示数据和接收用户的输入数据。视图层不存在真正的数据处理操作，只是用户与系统进行交互的界面布局<sup>[22]</sup>。在视图层中，开发人员根据需求说明文档中规定的界面设计要求对系统的界面布局及其组成元素进行代码实现，当模型层中数据出现变化，视图层中数据应根据规则判断是否进行相应的改变，及时刷新，为用户展示合适的请求结果。

(3) Controller 层：控制层连接视图层与模型层，完成数据的流转。接收到来自视

图层的数据请求后，控制层选择合适的模型层进行转发，在模型层处理完毕后，控制层将其返回的结果转发给视图层，视图层将结果展示给用户。控制层只有接收并转发的功能，本身并不负责任何业务处理和数据输出，Controller 仅仅接收请求，并根据配置文件选择合适的模型构件进行转发，使其处理该请求，当返回结果后再选择合适的视图来完成数据的展示。

MVC 设计模式体现了逻辑处理与视图展示的分离<sup>[23]</sup>，允许在不修改模型层和控制层等后端处理代码的情况下，仅通过对视图层代码的修改完成数据展示的改变，同样，对模型修改后，只要数据返回格式正确，视图仍然能够正确地显示结果，代码的耦合度较低且易于维护。由于视图层与模型层分离，模型层返回的数据能够被多个视图使用，代码具有较高的复用性。本项目前后端分离，使用 MVC 设计模式非常合适，这使得开发人员能够集中精力进行各自负责部分的开发工作，显著提高了开发效率，开发时间大大缩减，用户接口变得更加易于开发和维护。

## 2.3 微信公众平台

现如今，微信成为大众生活中最主要的社交工具，微信公众平台也成为许多企业拓展业务和维护客户的最佳选择<sup>[24]</sup>。微信公众平台有以下六点优势：开发技术相对简单，周期较短，项目能够迅速上线；具有跨平台特性，无需考虑用户硬件设施的兼容性和适配性；针对特定用户提供信息和服务，定位精准，营销转化率更高；随时提供信息和服务，不受时间和空间限制，服务效率更高，客户体验更好；推送内容为微信链接，不受平台限制，便于拓展新客户；与传统企业官方网站相比功能更多，展示效果更适合移动互联网，便于浏览。

本项目中所涉及到的微信公众号为微信服务号，主要偏向于服务交互（例如消息推送、服务查询、信息提交），在活动举办过程中为商家提供宣传渠道，为参与者提供报名通道，为参与者提供活动进度查询服务等。

微信公众号的交互模式如图 2.2 所示，可以看成是用户端、服务器和后台三者之间的数据交互。

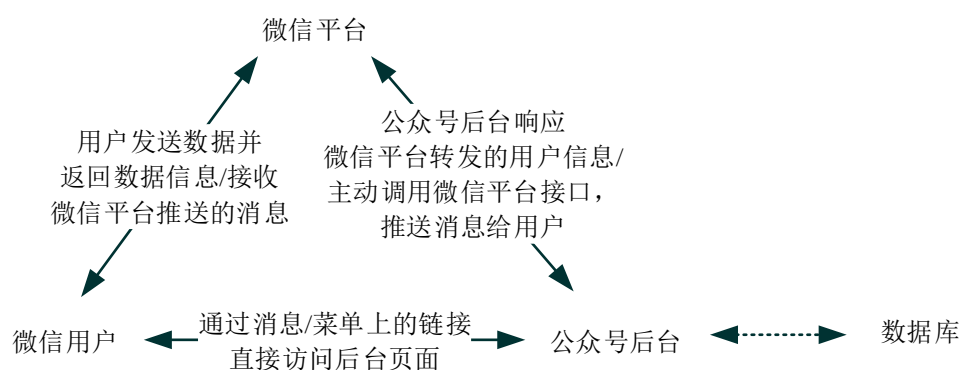


图2.2 微信公众号交互模式图

## 2.4 阿里云数据库

阿里云数据库在阿里云计算技术基础上发展兴起<sup>[25]</sup>，云计算的本质是通过整合管理网络中所有的计算资源，通过互联网来向大量用户提供高性能的服务。顾名思义，云数据库就是部署和虚拟化在云计算环境中的数据库。相较于传统自建数据库而言，云数据库拥有更多计算资源，存储能力更强；根据服务进行购买，具备动态可扩展性；企业根据自身需求购买，成本更低；由云数据库提供企业进行专业维护，免去购买者维护方面的投入<sup>[26]</sup>；云数据库使用简单，具备易用性；云数据库提供自动备份和多种技术措施保障数据库的正常运行，安全可靠。具体对比如表 2.1 所示。

表2.1 自建数据库与云数据库对比表

|       | 自购服务器自建数据库                | 云数据库                        |
|-------|---------------------------|-----------------------------|
| 服务可用性 | 自行保障，自建 RAID，自建主从复制等      | 99.95%                      |
| 数据可靠性 | 自行保障，自建 RAID，自建主从复制等      | 100.00%                     |
| 系统安全性 | 花费不菲自行部署，自行修复安全漏洞         | 防 DDoS 攻击，流量清洗；专业人员及时修复安全漏洞 |
| 数据库备份 | 自行实现，自行寻找备份空间，不能保证备份可恢复   | 自动备份                        |
| 软硬件投入 | 成本较高，使用 SQL Server 需购买许可证 | 无投入，按需购买服务                  |
| 系统托管  | 主从服务器每年花费超万元              | 免费托管                        |
| 维护成本  | 需要雇佣专业 DBA，花费人力财力         | 免维护                         |
| 部署扩容  | 自行进行准备工作，一旦部署不易改变，周期较长    | 即时开通，按需购买，弹性扩容              |
| 资源利用率 | 较低，需考虑峰值                  | 100.00%                     |

对于本项目而言,针对中小型商家对活动匹配系统的实际使用情况,采用阿里云数据库可以很好地满足要求。中小型商家考虑成本,不具备专业人员进行后期维护工作,并且举办活动具有明显的淡旺季,根据上述的分析可以得出:中小型商家的数据存储需求具有低成本、免维护、存储需求动态变化的特点<sup>[27]</sup>。阿里云数据库可以很好地满足中小型商家的要求。

在系统的数据库开发工作中,购买阿里云服务器,安装 MySQL 将其用于数据存储,在配置文件中填写相关阿里云账号和服务器信息,将数据库托管至云上,从而实现本项目与阿里云平台的结合。

## 2.5 本章小结

本章对智能匹配系统设计与开发工作中所需的相关理论与技术进行考察,作出简要介绍与分析,包括 Vue.js 技术、Spring Boot 框架、MVC 三层架构、微信公众平台和阿里云数据库等。

## 第三章 需求分析

本章首先对目前市场上的活动匹配业务发展情况进行调查和研究,分析现有解决方案的缺点,针对匹配系统的需求背景和业务概述,对智能匹配系统的功能需求进行一一列举和详细介绍,除此之外,对非功能性需求做出相应说明,为系统后续的设计与实现工作打下基础。

### 3.1 系统总体需求

#### 3.1.1 需求背景

在市场活动中,对商家而言,为了实现拉新、留存和促活等用户运营目的,举办活动是必不可少的运营手段。现阶段,很多人员匹配分组工作都是通过基于纸质的人工统计来完成,效率极低,频繁出错,耗时长,活动数据不方便存储和传播<sup>[28]</sup>。为了单一活动而开发一套会员分组系统耗费时间,成本高昂且无法重复使用,并不适合众多中小商家。相比于市面上现有的活动匹配系统,本论文中介绍的智能匹配系统具有低成本,高复用性的优点。中小型商家无力承担大型系统或多个小型系统的开发成本,该系统提供了拥有众多可选匹配规则、可编辑活动表单以及可视化活动统计的低成本解决方案。

#### 3.1.2 业务概述

活动组织者编辑活动信息项、上传活动图片并选择匹配规则项,发布活动后,活动信息存储至阿里云数据平台,生成活动链接返回至微信公众平台,经由微信公众号发送给活动参与者,活动参与者填写活动表单,用户信息同样存储至阿里云数据平台,参加活动后接收关于活动的微信提醒,可以分享活动,查看活动进度并在结束后评价活动,评价信息存储至阿里云数据平台,活动组织者可以查看活动参与情况的统计分析图表并导出相关文档。

该智能匹配系统包括以下核心功能:

- (1) 用户管理包括查询活动参与者信息、编辑参与者信息和手动设置禁用功能;
- (2) 活动管理包括活动发布、用户参与、匹配查询、活动查询、活动分享、活动评价、活动修改、活动删除功能;
- (3) 选项库管理包括对表单备选项的查询、新建、修改和删除功能;
- (4) 统计管理包括查看活动相关统计数据 and 活动信息文档的导出功能;
- (5) 商家管理包括对商家账号新增、启用、停用、编辑、修改密码功能。

为了实现使用本系统举办活动的目的，本系统对用户角色进行了有效的划分。系统中所存在的用户角色有：商家、活动参与者和系统管理员。系统总体用例情况如图 3.1 系统总体用例图所示。三种角色的用户在系统中具备的具体功能如下：

(1) 活动参与者用户：查看活动基本信息，填写个人信息，申请参加活动，分享活动，查看活动进度，评价活动；

(2) 商家用户：查看编辑活动参与者基本信息和扩展信息，禁止该用户参与活动，发布、修改和删除活动，新建、修改和删除备选信息项，统计活动数据，导出活动数据报表，绑定微信公众平台，编辑商家账号信息。

(3) 系统管理员：负责系统数据库的维护和管理、对系统日志查看和维护、对商家账号进行管理。

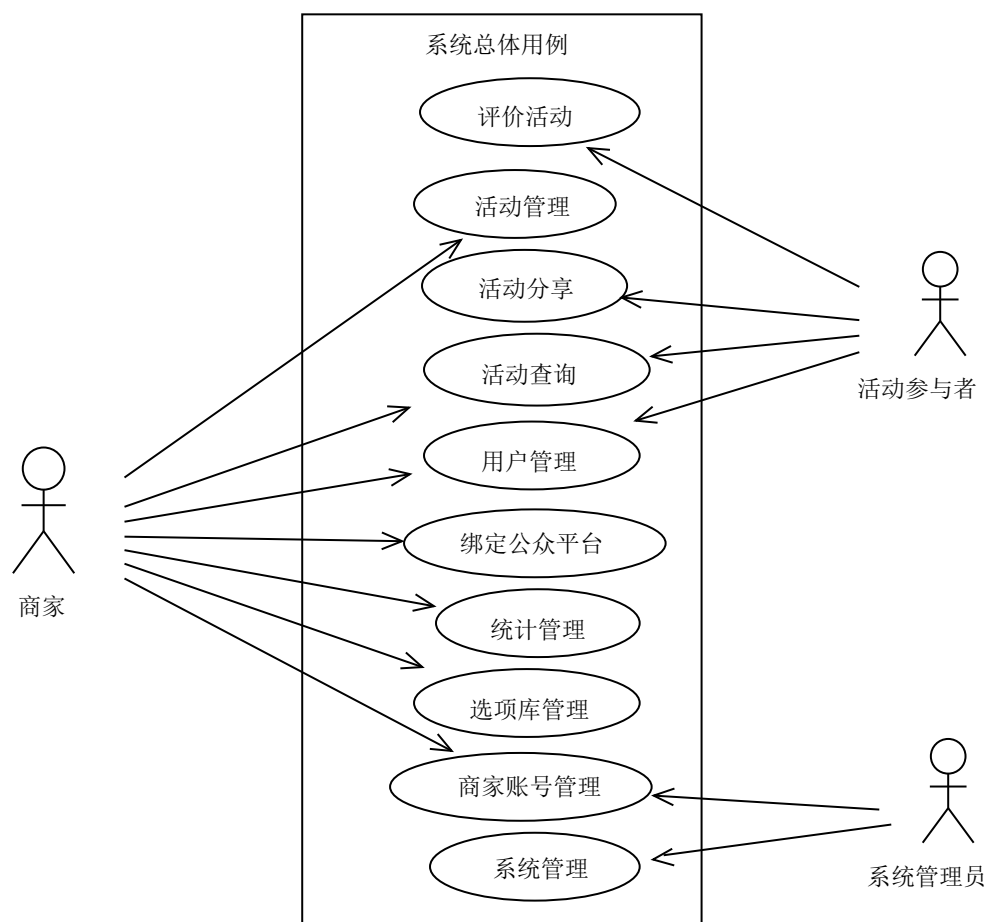


图3.1 系统总体用例图

## 3.2 系统功能需求分析

### 3.2.1 用户管理分析

用户管理模块中有基本信息管理和状态管理两个子模块。活动参与者在个人信息页面填写个人基本信息，对于基本信息用户可自行进行修改和新增。商家可以对活动

参与者信息、状态进行修改和查看，对行为不当用户可进行禁用处理，禁用后可以了解禁操作。用户管理的用例分析结果如图 3.2 所示。

#### (1) 用户基本信息管理

用户基本信息管理，包括对用户的微信号，姓名，性别，手机号，身份证号等信息进行维护。活动参与者和商家可以对参与者的基本信息进行编辑修改。

#### (2) 用户基本信息查看

用户基本信息查看是指活动参与者和商家能够对基本信息进行查看。

#### (3) 用户状态管理

用户状态管理，是指对活动参与者的启用用户状态和禁用用户状态的切换。商家可以对参与者的状态进行切换。

#### (4) 用户状态查看

用户状态查看是指商家对参与者的状态进行查看。活动参与者无法查看到状态信息。

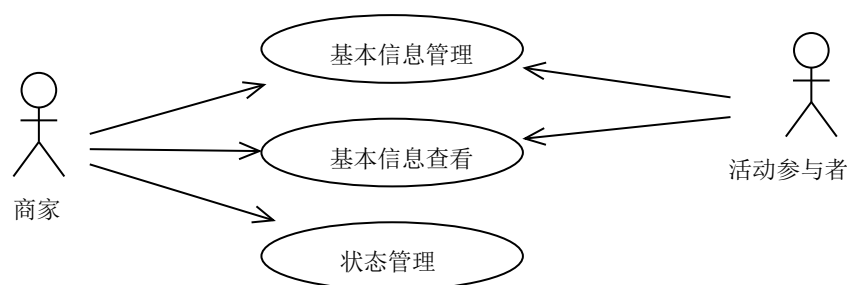


图3.2 用户管理用例图

### 3.2.2 活动管理分析

活动管理模块分为活动发布、用户参与、匹配查询、活动查询、活动分享、活动评价、活动修改、活动删除等八个子模块。活动参与者打开活动参与页面，可查看商家发布的活动信息，填写活动表单来报名参加活动，等待后台系统自动进行用户匹配后，返回微信提醒消息，查询活动参与结果，在结束后进行活动评价。在此期间，商家可进行对活动进行修改和删除。活动管理模块用例分析如图 3.3 活动管理用例图所示。

#### (1) 活动发布

商家对活动信息进行设置。选择合适的自行编辑的备选信息项、选择匹配规则项、添加少量的背景图片组，设置开始时间和报名截止时间并且附上相应的活动介绍形成活动表单。

#### (2) 用户参与

活动参与者根据自身情况在活动申请页面中填写活动表单，填报过程中用户基本

信息直接由微信后台读取生成，不可修改。被禁用户默认报名失败。

(3) 匹配查询

在报名结束后，系统根据商家设定好的匹配规则对报名用户进行自动分组匹配。被禁用户不参与分组匹配。活动参与者进入公众号个人页面能够进行匹配结果查询，商家在后台系统活动管理菜单中可以进行匹配结果查询。

(4) 活动查询

活动参与者点击微信消息进入个人页面进行活动查询，商家在后台系统活动管理中查看活动列表进行活动查询。

(5) 活动评价

用户参与活动完毕，对所参与活动进行评价。

(6) 活动分享

用户点击活动参与页面的分享按钮可以向微信好友发送活动链接或将该链接发布在微信朋友圈，他人可以通过点击该链接填写表单进行报名。

(7) 活动修改

活动发布后，商家可对活动信息进行修改。修改信息将以微信消息形式同步推送给该活动报名用户。

(8) 活动删除

活动发布后，商家可对活动进行删除。删除信息将以微信消息形式同步推送给该活动报名用户。

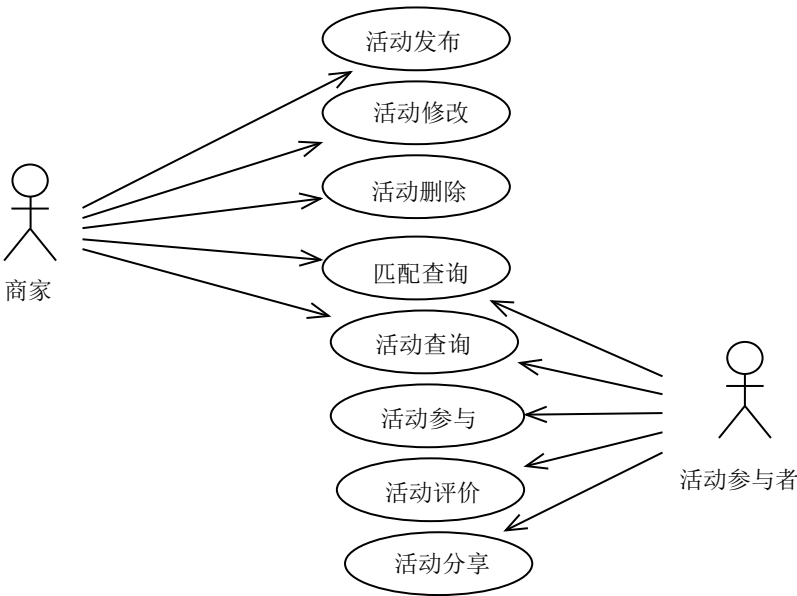


图3.3 活动管理用例图



### 3.2.3 选项库管理分析

选项库管理模块分为选项查询、选项新增、选项修改和选项删除等四个子模块。商家对用于构建活动表单的选项进行查询、新增、修改和删除。选项库管理模块用例分析结果如图 3.4 所示。

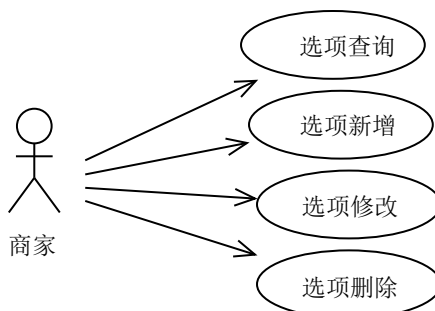


图3.4 选项库管理用例图

#### (1) 选项查询

商家对选项库中的选项进行查询。

#### (2) 选项新增

商家在选项库中新增选项用于后续添加至活动表单。新增时需要填写名称、选择选项类型和限制条件。

#### (3) 选项修改

商家在选项库中选择选项进行修改。

#### (4) 选项删除

商家在选项库中选择选项进行删除。

### 3.2.4 统计管理分析

统计管理包括统计数据查询和文档导出子模块。系统对举办完毕的活动的参与人数、转化率、好评率、用户偏好分析等数据作一系列相应的统计，以图表形式展示在统计管理页面，商家可浏览总体数据，选择时间范围来查询相应时段活动举办效果。商家查询完毕后，可以点击导出按钮将活动统计数据导出为本地文档。统计管理模块的用例分析结果如图 3.5 所示。

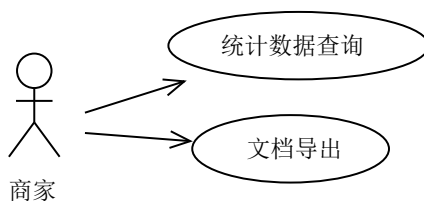


图3.5 统计管理用例图

### （1）统计数据查询

商家在统计管理页面可查看全部活动的相应数据，选择时间范围即可查看固定时段活动的统计数据。

### （2）文档导出

商家在查看完毕统计数据后，可选择以文档形式导出至本地进行信息备份。

## 3.2.5 商家管理分析

商家管理包括管理员对商家账号新增、启用、停用和商家对自身账号信息编辑、修改密码。商家管理模块用例分析如图 3.6 所示。

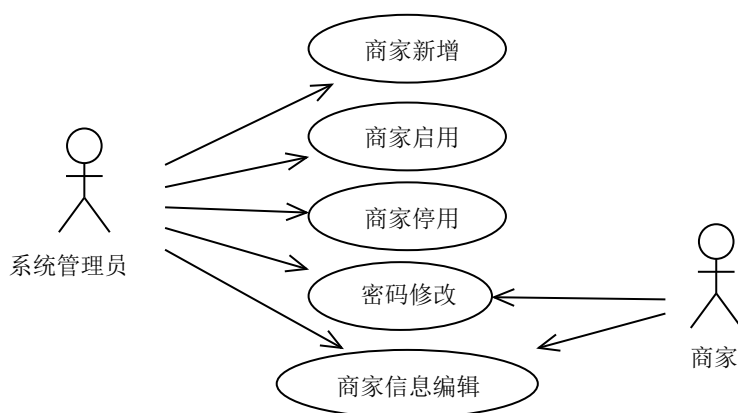


图3.6 商家管理用例图

### （1）创建商家账号

系统管理员能够对商家账号进行创建。创建的商家账号信息包括商家姓名、绑定微信公众号名称、联系方式、登录账号和登录密码。

### （2）启用商家账号

系统管理员能够对停用状态的商家账号进行启用处理。启用后的商家账号能够登录系统，使用商家用户所具备的功能。

### （3）停用商家账号

系统管理员能够对出现业务变更的商家账号进行停用处理。停用后的商家用户无法登录系统，也不能使用商家用户具备的任何功能。

### （4）商家账号信息编辑

商家可以对自身信息进行编辑，包括名称、联系方式。若绑定微信公众号名称有误，可联系系统管理员进行更改。

### （5）修改密码

商家通过输入原来的登录密码和新的登录密码来进行密码重设操作，系统对商家用户输入旧密码的正确性进行验证，如果验证通过，将输入的新密码设为登录密码，

如果验证失败，则提示修改失败信息。当商家忘记密码时，系统管理员需对其进行身份验证后方可重置密码。

### 3.2.6 系统管理分析

系统管理是指智能匹配系统向系统管理员提供的针对系统本身的管理功能，主要对数据库、日志和密码进行管理。系统管理的用例分析结果如图 3.7 所示。

#### (1) 数据库管理

数据库管理主要包括备份现有数据、还原备份数据和清理冗余数据等三个功能。系统管理员对系统应用数据进行备份处理的操作被称为数据备份，系统管理员为备份文件选择存储路径，系统记录备份时间并根据系统管理员提供的备份文件的存储路径对现有数据库完成备份处理。还原备份数据是指系统管理员使用备份数据库文件进行数据还原操作，系统将所选的备份数据库文件还原到本系统的数据库中。而清理冗余数据是指系统管理员选中待清理的活动管理数据并提交清理操作请求，系统在后台数据库中批量删除对应的活动管理数据并返回删除结果。

#### (2) 日志管理

系统后台将一切对系统产生影响的系统事件都记录在档，例如用户登录、系统数据的增删改查等，并且进行存储，形成了系统日志。系统管理员对系统日志的详细信息进行查看，包括系统操作的类型、时间和用户等，系统将日志文件信息以列表形式进行展示。

#### (3) 密码管理

密码管理是指系统管理员通过输入当前的登录密码与新密码来完成密码重置操作，系统对当前登录密码的正确性进行验证，验证通过则将输入的新密码设置为登录密码，验证失败则给出重设密码失败信息提示。

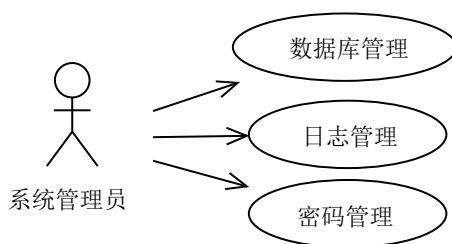


图3.7 系统管理用例图

## 3.3 系统非功能性需求分析

### 3.3.1 系统性能需求

智能匹配系统除了满足功能需求之外，还需要满足基本的非功能性需求。其中，

作为非功能性需求的核心，性能需求包括可用性、易用性、安全性、兼容性和可扩展性等。对于系统而言，非功能层面的需求也是必不可少的。对于系统是否满足上述性能需求的问题，具体以系统的四个指标表现为评判标准，即响应时间、并发性、网络通信能力和风控方案。

（1）响应时间：根据系统实际使用情况，操作的响应时间应限制在 3 秒之内，例如在多用户并发活动查询业务中，在正常运行环境下并发人数达到峰值时，每个并发查询操作都应满足在 3 秒之内能够获得响应。

（2）并发性：根据中小型商家举办活动业务的真实情况，考虑到该智能匹配系统需要满足适用于目前业务的流量要求，必须支持不少于 100 的并发用户。

（3）网络通信能力：智能匹配系统中存在数量可观的活动管理数据需要进行处理，目前中小商家的网络环境基本为百兆宽带网络或采用光纤网络，按照相关管理部门和商家的要求，智能匹配系统的数据传输速率不小于 10kbps。

（4）风控方案：确保智能匹配系统全天 24 小时为用户提供服务。每年累计宕机时间低于 8 个小时。一旦发生故障，维护人员必须在 2 小时之内完成系统恢复，确保系统能够正常运行。

从系统开发测试完成到部署上线的过程中，软硬件配置影响着上述四个方面的指标表现，甚至决定了系统的性能需求能否满足。因此，当系统开发工作完成后，选择性能优异、表现出色的软硬件平台，除此之外，对系统进行大量的针对性强且相互独立的性能测试，保证系统在实际投入使用时满足上述四个方面的性能需求。

### 3.3.2 系统安全需求

在系统实际运行过程中，不但要满足功能和性能方面的要求，还要在数据存储、系统操作和运行网络等方面具备高度可靠的安全保障，确保本系统在实际使用中稳定正常运行，本系统的安全需求具体包括以下几个方面：

（1）数据安全：在数据库操作过程中，系统限制字符长度，配备了全面完善的防止 SQL 语句注入的安全措施，除此之外，可以对数据库进行备份和还原。

（2）系统操作安全：严格控制系统中不同用户角色的权限，权限体系严格监督和控制用户的各项系统操作。比如商家在选项库修改中的各项操作仅仅针对选项库中的选项进行编辑，不得更改预设的匹配规则项。并且系统操作都有相应日志记录，以便排查问题。

（3）网络安全：本系统托管至阿里云服务器上，由专业人员负责维护系统的网络安全，配置最新的防火墙、提供入侵检测和 DDoS 攻击防御功能，对于常见的网络攻击、病毒、木马，本系统能够良好地应对和处理。

### 3.4 本章小结

本章介绍了智能匹配系统的需求分析工作，在目前市场上中小型商家举办活动需要对参与者进行分组匹配的业务背景下，对匹配系统的业务进行充分的分析和说明，不仅对智能匹配系统的功能需求进行了详细的阐述，而且简要介绍了系统的非功能性需求，为系统的详细设计与实现工作起到了重要的指导作用。



## 第四章 设计与实现

完成智能匹配系统的需求分析工作后，对系统的详细设计与实现工作结合软件设计技术、Vue.js 开发技术、云数据平台开发技术和微信公众平台开发技术进行分析阐述，首先对智能匹配系统的系统架构、系统功能结构和网络拓扑图进行设计，然后介绍了数据库设计工作，最后对具体功能模块的实现进行了详细的说明。本章对智能匹配系统的详细设计与实现工作进行具体介绍。

### 4.1 系统总体设计

#### 4.1.1 系统架构设计

智能匹配系统的架构采用 MVC 设计模式。系统架构如图 4.1 所示。

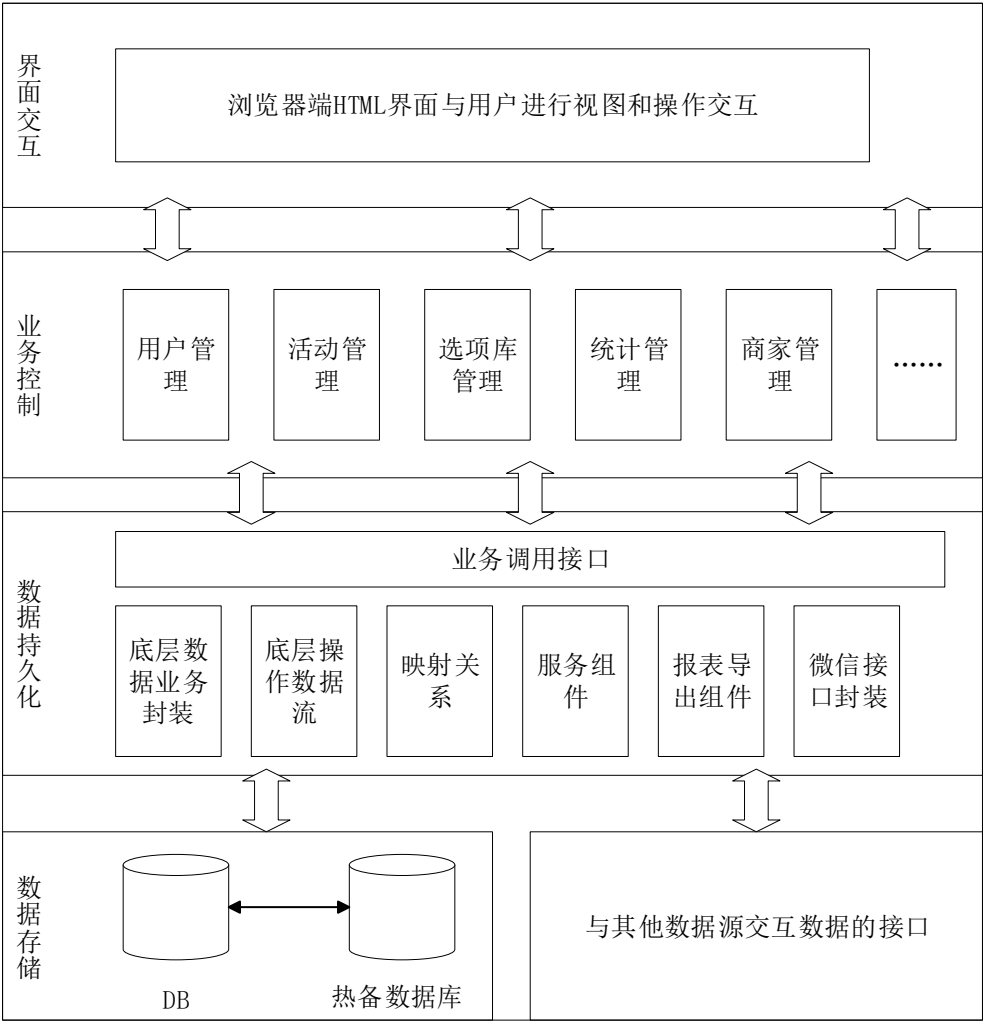


图4.1 系统架构图

在进行智能匹配系统的架构设计时，除了满足目前系统的业务需求，还要考虑系统未来的扩展和维护，因此选择分层架构非常合适。本系统的设计模式为 MVC，采用基于 Web 的三层架构，这是一种非常经典成熟的框架，在 B/S 架构的信息系统软件开发中<sup>[29]</sup>，被企业开发人员广泛使用，三层架构的设计思路是将数据的显示、处理和存储三者进行分离，减少系统间的依赖，实现高效开发的目的。智能匹配系统最底层是数据持久层，处于中间的是业务控制层，最顶层是界面交互层。智能匹配系统的架构图如图 4.2 所示。以下是系统各层次的简要描述：

(1) 数据持久层：最底层中不仅含有数据库管理系统<sup>[30]</sup>，还有涉及到数据读写的数据访问层。系统业务原始数据存储在数据库管理系统中，由数据库管理系统负责管理，对于本系统来说，数据访问层主要负责的是对 MySQL 数据库的原始数据接口完成封装，并且将新的数据访问接口提供给业务控制层。

(2) 业务控制层：业务控制层主要对用户管理、活动管理、选项库管理、统计管理商家管理以及系统管理的业务逻辑进行控制处理。业务控制层是连接负责用户交互的界面交互层和负责数据存储的数据持久层的中介，业务控制层获得来自界面交互层的数据，选择合适的业务逻辑规则对其进行运算处理后生成相应的数据请求，处理过程中可能访问数据持久层进行数据交换，处理完成后向界面交互层返回需要展示的数据。

(3) 界面交互层：界面交互层的主要任务包括与用户完成交互和为用户进行数据展示。在界面交互层部分一般包含了数据显示的格式和方法<sup>[31]</sup>。界面交互层读取到用户输入的数据后将其交给业务控制层进行业务处理，待处理完毕后，界面交互层获取到来自业务控制层的数据，根据自身内部预设的数据展示逻辑生成对应的代码，在浏览器中进行数据展示，在界面交互层的上一层，是 Web 浏览器，在界面交互层开发人员一般会使用 ASP、CSS、JavaScript 等相关技术来编写数据显示规则。

### 4.1.2 系统功能结构设计

基于用户角度而言，在智能匹配系统中，对于功能结构的设计应该采用功能模块的形式将业务流程进行拆分，对第三章需求分析提到的功能需求进行恰当地封装，由此整理得到的智能匹配系统的具体功能结构如图 4.2 所示。

根据图 4.2 所示的智能匹配系统的功能结构，本系统中具备的功能模块与功能需求项一一对应，其中，核心的功能模块有用户管理、活动管理、选项库管理、商家管理、统计管理和系统管理，综合上文第三章中功能需求分析所述，对各功能模块进行简要说明如下：

- (1) 用户管理包括查询活动参与者信息、编辑参与者信息和手动设置禁用；
- (2) 活动管理包含活动发布、用户参与、匹配查询、活动查询、活动分享、活



动评价、活动修改、活动删除功能；

- (3) 选项库管理包括对表单备选项的查询、新建、修改和删除功能；
- (4) 统计管理包括查看活动相关统计数据 and 活动信息文档的导出功能；
- (5) 商家管理包括对商家账号新增、启用、停用、编辑、修改密码功能；
- (6) 系统管理包括数据库管理、日志管理和密码管理功能。

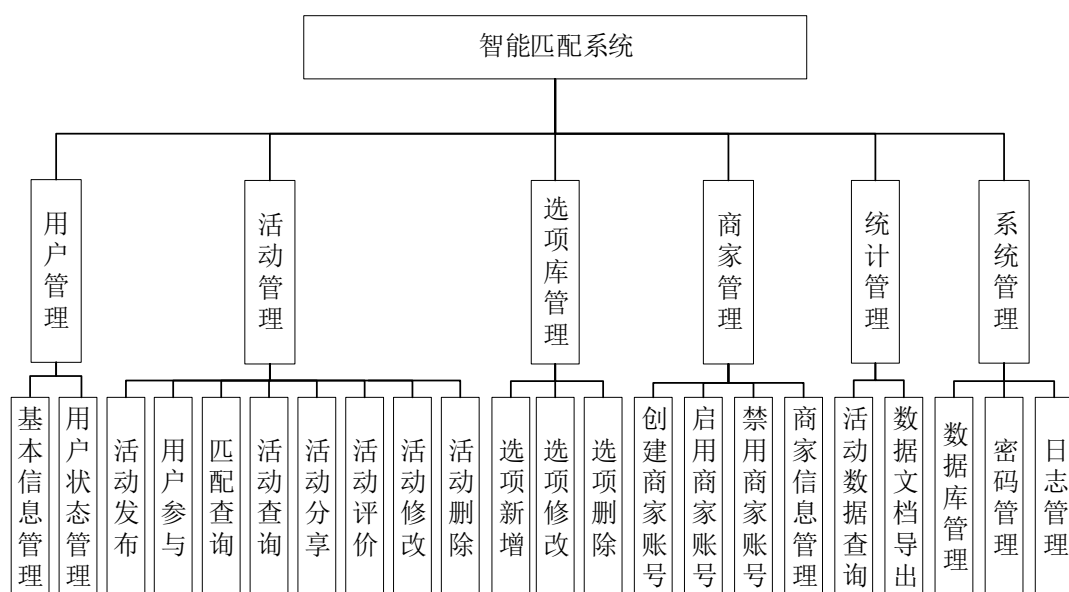


图4.2 系统功能结构图

### 4.1.3 系统网络拓扑设计

智能匹配系统的网络拓扑图如图 4.3 所示。本系统的网络架构主要采用了结合微信公众平台的基于 B/S 的浏览器端/服务器端模式<sup>[32]</sup>，主要使用场景仍为 PC 端（商家）和手机端（活动参与者），手机端使用微信跳转至 h5 页面，所以使用 B/S 模式十分适合。网络环境为一般家用或办公网络，通信节点包括用户端节点、数据库服务器和应用服务器等。由于活动需要存储图片元素，预计存储空间需求较大，为了不影响性能和使用，除本地数据库外，将本系统托管至阿里云服务器，使用阿里云数据平台。

在图 4.3 系统网络拓扑图中，最上方的商家、活动参与者和系统管理员所使用的主机属于用户端节点，系统本地的应用服务器和搭建在云端的数据库服务器两类型主机都属于服务器节点，应用服务器与数据库服务器使用双机并行的模式，在城市网络中使用公用网络信息通道进行交互，阿里云数据平台相对于自建数据库，价格低廉，无需运维投入，并且自建数据库存取图片较多会影响性能表现，由于该项目的商业性质，出于成本和性能考虑，使用阿里云数据平台而不是自建数据库。用户端与应用服务器之间通过城市网络进行通信，网络类型属于以太网，通信协议为 TCP/IP 传输协议，除此之外，使用硬件防火墙对所有数据通信进行必要的安全过滤。

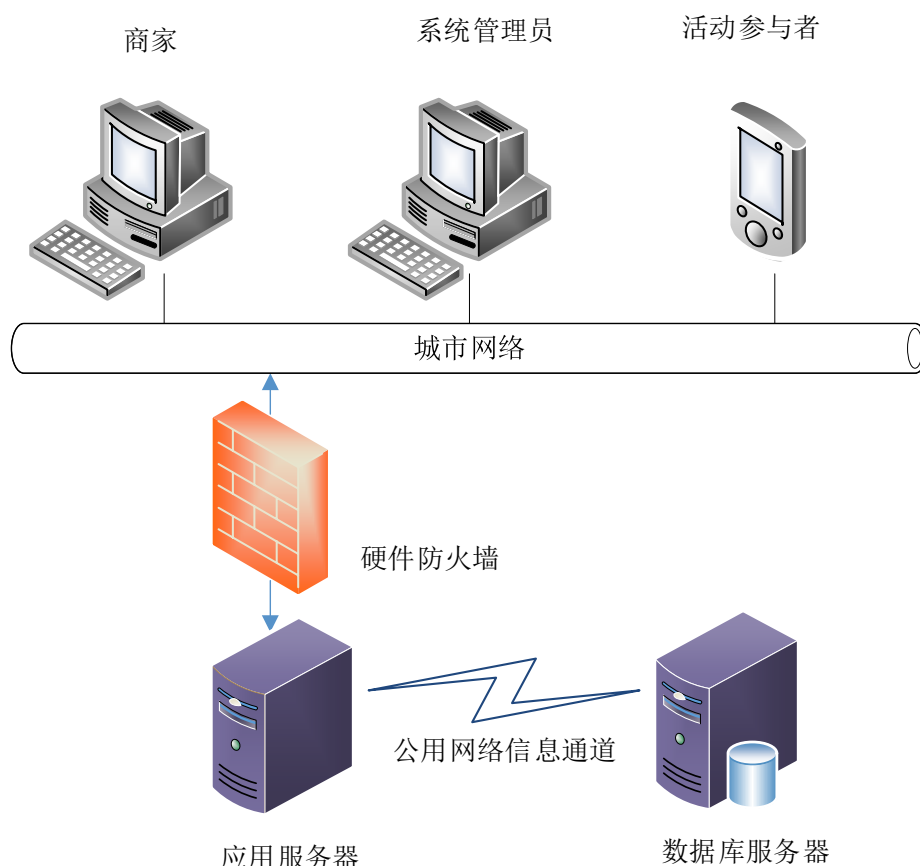


图4.3 系统网络拓扑图

## 4.2 系统数据库设计

### 4.2.1 E-R 模型设计

在本系统数据库的设计工作中,按照符合数据库版本的范式要求对该系统的数据库表结构进行设计,首先需要绘制 E-R 图<sup>[33]</sup>将系统中的各个对象和属性尽可能清晰地表达出来。在对数据模型进行设计的过程中,本系统采用外键将不同实体的属性关联起来,不但体现了数据库的一致性,而且充分展示了系统的业务逻辑和组织结构。本系统的概念模型如图 4.4 系统 E-R 图所示。

如图 4.4 所示,系统的实体-关系图中包含了商家、参与者、活动、选项、规则项、统计报告、系统日志等六个对象实体。在设计数据库时,严格遵循设计原则完成本系统数据字段的设计,为系统后期的编码工作奠定基础,以便能顺利完成系统的后续设计和实现。E-R 图中,标注了实体及其之间的关系,比如商家与活动之间是举办关系,它们的关联是一对多,一个商家可以举办多个活动,但一个活动只能由一个商家举办。参与者与活动之间是参与关系,它们的关联是多对多关联。一个参与者可以参加多个活动,一个活动也能有多个参与者参与。



名称、账号、密码、手机号码、商家状态等字段。其中商家 ID 是主键。

表4.1 商家信息表

| 字段名      | 字段类型        | 描述       | 允许为空  |
|----------|-------------|----------|-------|
| ID       | int         | 商家 ID    | 否（主键） |
| VX_ID    | int         | 微信公众号 ID | 否     |
| Name     | varchar(50) | 商家名称     | 否     |
| Account  | varchar(20) | 账号       | 否     |
| Password | varchar(20) | 密码       | 否     |
| Mobile   | varchar(20) | 手机号码     | 是     |
| Status   | int         | 商家状态     | 否     |

### （2）参与者信息表

参与者信息表如表 4.2 所示。参与者信息表中包含了参与者微信 ID、姓名、性别、生日、手机号码、参与者状态等字段。其中参与者微信 ID 是主键。

表4.2 参与者信息表

| 字段名      | 字段类型        | 描述       | 允许为空  |
|----------|-------------|----------|-------|
| ID       | int         | 参与者微信 ID | 否（主键） |
| Name     | varchar(50) | 姓名       | 否     |
| Gender   | int         | 性别       | 否     |
| Birthday | datetime    | 生日       | 否     |
| Mobile   | varchar(50) | 手机号码     | 否     |
| Status   | int         | 参与者状态    | 否     |

### （3）活动信息表

活动信息表如表 4.3 所示。活动信息表中有活动 ID、活动名称、活动表单选项 ID、活动规则项 ID、开始时间、截止时间、活动图片、活动评价、活动状态字段。其中活动 ID 是主键。

表4.3 活动信息表

| 字段名     | 字段类型        | 描述    | 允许为空  |
|---------|-------------|-------|-------|
| ID      | int         | 活动 ID | 否（主键） |
| Name    | varchar(50) | 活动名称  | 否     |
| Item_ID | int         | 选项 ID | 否     |

续表 4.3 活动信息表

| 字段名        | 字段类型        | 描述       | 允许为空 |
|------------|-------------|----------|------|
| Rule_ID    | int         | 活动规则项 ID | 否    |
| Start_time | datetime    | 活动开始时间   | 否    |
| End_time   | datetime    | 活动截止时间   | 否    |
| Picture    | varchar(50) | 活动图片     | 否    |
| Comment_ID | int         | 活动评价 ID  | 是    |
| Status     | int         | 活动状态     | 否    |

## (4) 选项信息表

选项信息表如表 4.4 所示。选项信息表中有选项 ID、选项名称、选项类型、选项描述字段。其中选项 ID 是主键。

表4.4 选项信息表

| 字段名     | 字段类型        | 描述    | 允许为空  |
|---------|-------------|-------|-------|
| ID      | int         | 选项 ID | 否（主键） |
| Name    | varchar(50) | 选项名称  | 否     |
| Type    | int         | 选项类型  | 否     |
| Comment | varchar(50) | 选项描述  | 是     |

## (5) 规则项信息表

规则项信息表如表 4.5 所示。规则项信息表中有规则项 ID、规则项名称、规则项描述、限制条件字段。其中规则项 ID 为主键。

表4.5 规则项信息表

| 字段名     | 字段类型        | 描述     | 允许为空  |
|---------|-------------|--------|-------|
| ID      | int         | 规则项 ID | 否（主键） |
| Name    | varchar(50) | 规则项名称  | 否     |
| Type    | int         | 规则项类型  | 否     |
| Comment | varchar(50) | 规则项描述  | 是     |
| Limit   | varchar(50) | 限制条件   | 否     |

## (6) 活动评价表

活动评价表如表 4.6 所示。活动评价表中有活动评价 ID、活动 ID、参与者 ID、

评价类型、评价内容字段。其中评价 ID 为主键。

表4.6 活动评价信息表

| 字段名     | 字段类型        | 描述      | 允许为空  |
|---------|-------------|---------|-------|
| ID      | int         | 活动评价 ID | 否（主键） |
| Act_ID  | int         | 活动 ID   | 否     |
| User_ID | int         | 参与者 ID  | 否     |
| Type    | int         | 评价类型    | 否     |
| Content | varchar(50) | 评价内容    | 是     |

（7）系统日志表

系统日志表如表 4.7 所示。系统日志表中有日志 ID、用户 ID、日志内容、日志类型、记录时间字段。其中日志 ID 是主键。

表4.7 系统日志表

| 字段名     | 字段类型        | 描述    | 允许为空  |
|---------|-------------|-------|-------|
| ID      | int         | 日志 ID | 否（主键） |
| User_ID | int         | 用户 ID | 否     |
| Content | varchar(50) | 日志内容  | 否     |
| Type    | int         | 日志类型  | 否     |
| Time    | datetime    | 记录时间  | 是     |

4.3 系统功能设计与实现

智能匹配系统的核心功能模块有用户管理模块、活动管理模块、选项库管理模块、统计管理模块、商家管理模块和系统管理模块。智能匹配系统的功能模块众多，功能点涉及较广，本节对系统的部分核心功能的设计和实现进行阐述说明。

4.3.1 用户管理模块设计与实现

用户管理模块包含一系列对参与者信息的管理操作，包括对参与者的基本信息与状态进行管理和查看。下面对用户管理的设计与实现工作进行具体介绍。

1.类图设计

智能匹配系统分为三个层次，自上而下依次是表现层、业务控制层与数据持久层，从设计思路落实到代码实现上，对应的功能实现通常包含了实体对象类、数据持久层数据处理类、数据持久层数据库处理类、表现层请求处理类和业务控制层逻辑处理类。

用户管理的类图设计如图 4.5 所示。

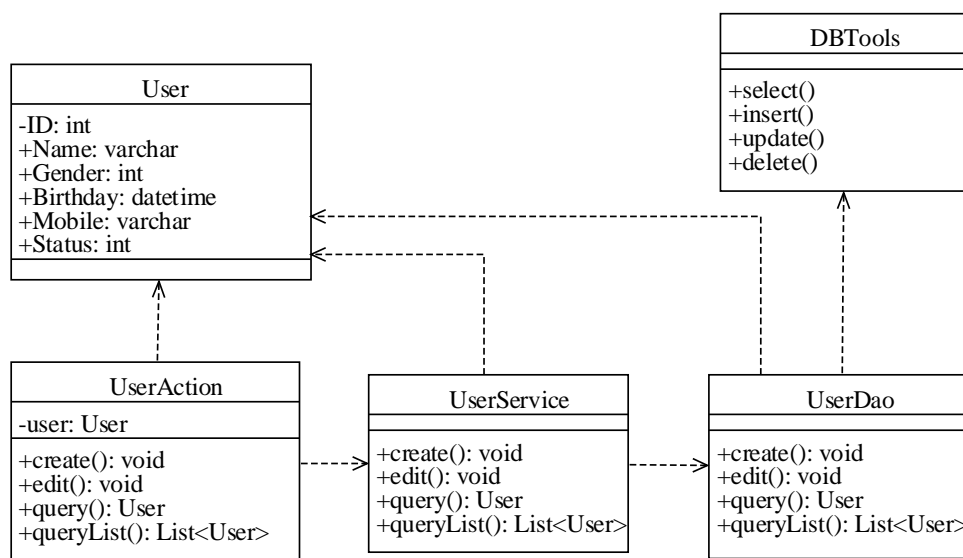


图4.5用户管理类图

### (1) User 类

**User** 类是用户实体对象类，其中包含用户个人基本信息和状态信息。在 **User** 类中包含了 ID、Gender、Name、Birthday、Mobile、Status 等字段以及相应 get 和 set 方法。

### (2) UserAction 类

**UserAction** 类属于用户管理模块的表现层，负责对页面请求进行处理。当 **UserAction** 类获得从浏览器端传来的请求后，对请求数据进行格式转换，将页面请求传递至合适的业务控制层，在业务控制层中根据相应的业务逻辑进行处理。在 **UserAction** 类中有创建用户信息 **create()**方法、编辑用户信息 **edit()**方法、查询用户信息 **query()**方法和批量查询用户信息 **queryList()**方法。

### (3) UserService 类

**UserService** 类属于用户管理模块的业务控制层，负责对业务逻辑进行处理。**UserService** 类接收到经过转换和校验后的表现层请求，选择合适的业务逻辑规则进行处理，与 **UserDao** 类共同完成对业务逻辑的运算处理，并且为表现层请求处理类返回处理结果。

### (4) UserDao 类

**UserDao** 类属于用户管理模块的数据持久层，负责对业务数据进行处理。数据从业务控制层传递给数据持久层后，都由 **UserDao** 类进行处理后提供给业务逻辑处理类，**UserDao** 类为 **UserService** 类提供数据操作支持，比如对用户信息进行增删改查等操作。在此之后，调用 **DBTools** 类实现数据的持久化。

### (5) DBtools 类

DBtools 类属于用户管理模块的数据持久层，负责在数据库中进行增删查改。DBtools 负责调用并执行各项数据库操作，完成数据持久化。比如进行数据库的新增 insert()操作、查询 select()操作、修改 update()操作、删除 delete()操作。

#### 2.时序图设计

动态调用用户管理类图，对用户管理进行时序设计。用户管理模块的时序设计如图 4.6 所示。

在图 4.6 的用户管理时序图中，用户在浏览器界面对用户信息进行修改，点击提交按钮，将用户信息提交给用户信息表现层请求处理类 UserAction，随后调用 edit() 方法。在表现层请求处理类 UserAction 中对请求数据进行转换和校验，校验通过后调用业务控制层逻辑处理类 UserService 类中的 edit()方法完成逻辑处理部分，下一步调用数据持久层数据处理类 UserDao 进行数据处理，调用 edit()方法生成符合 MySQL 数据库版本要求的 SQL 的 update 语句，最后调用数据库处理类 DBtools 的 update()方法将更新后的用户信息数据写入数据库。处理完成后返回调用结果，按照预设的页面显示逻辑将该响应结果显示在用户界面上。

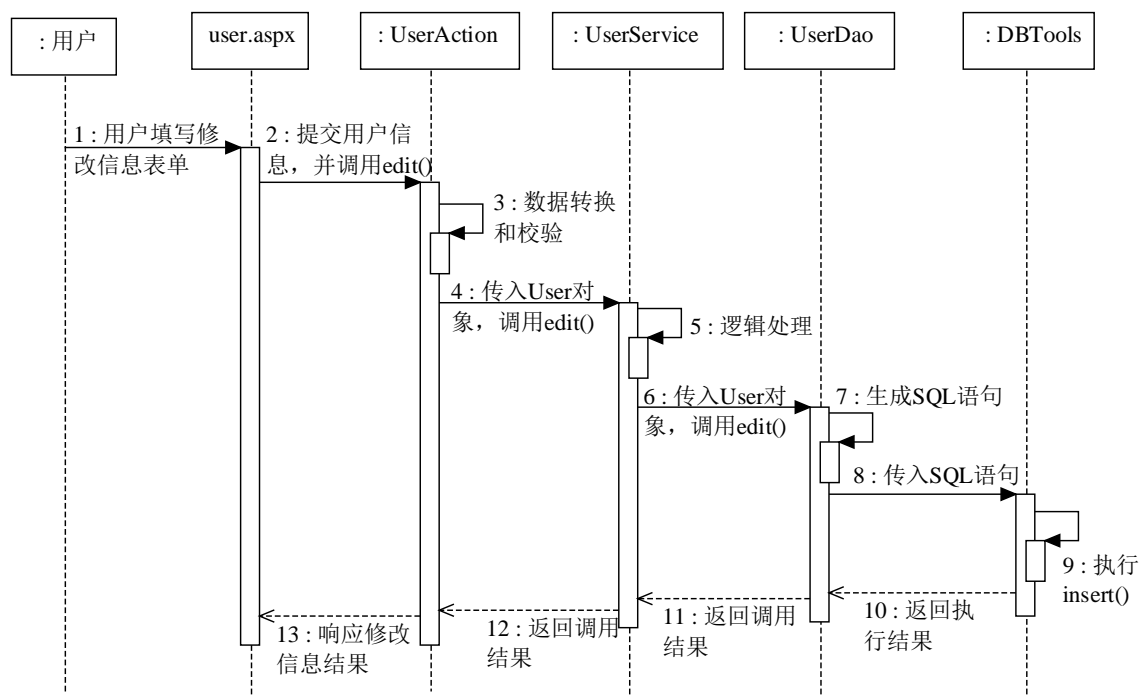


图4.6 用户管理时序图

### 4.3.2 活动管理模块设计与实现

活动管理主要包括活动发布、用户参与、用户匹配、活动查询、活动分享、活动评价、活动修改和活动删除。



## 1. 流程设计

活动管理模块的具体流程设计如图 4.7 活动管理流程图所示。

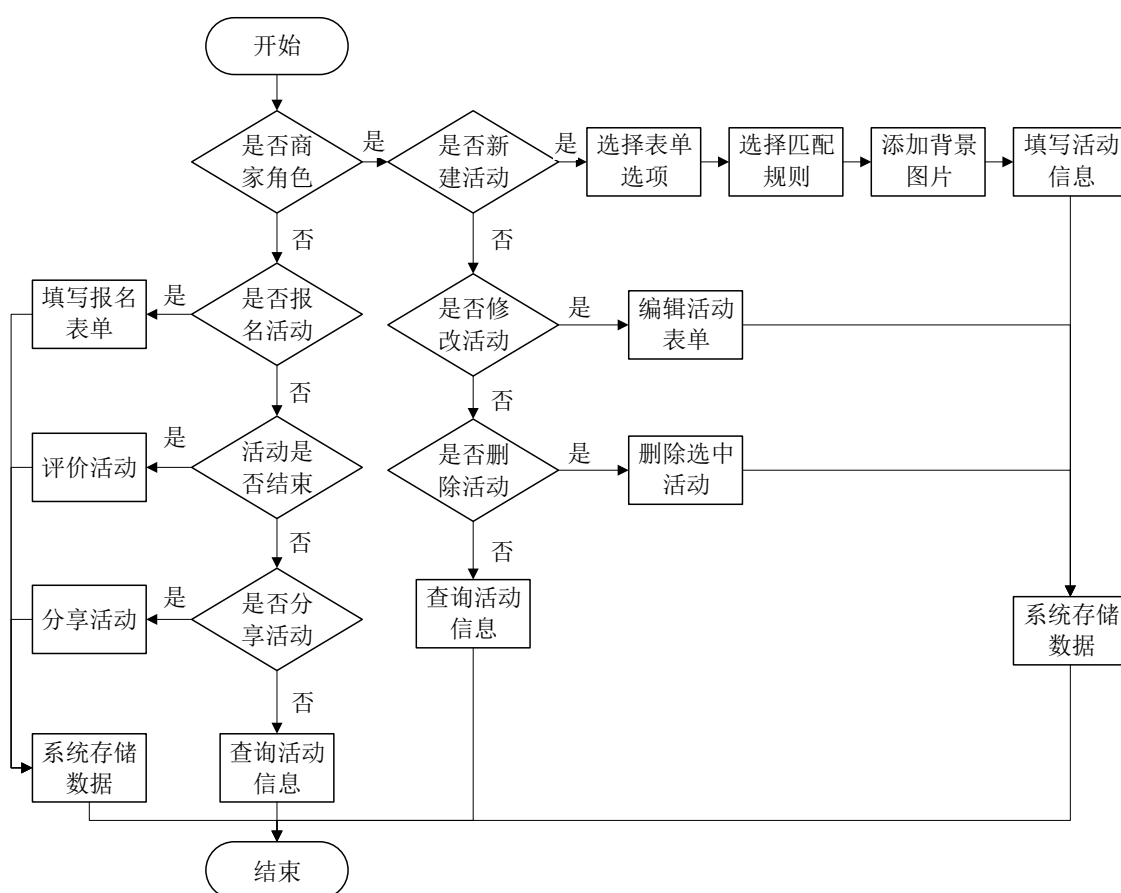


图4.7 活动管理流程图

在图 4.7 活动管理流程图中，用户成功登录系统后，系统根据用户角色显示相应系统界面。如果是商家角色登录系统，则进入商家系统展示商家权限相应界面。当商家点击新建活动时，进入发布活动界面，商家在新建活动界面中选择表单选项作为活动表单的基本信息项，选择匹配规则项作为该活动进行用户分组匹配的规则，为活动表单添加背景图片组成活动海报，填写活动信息，然后单击发布按钮，系统将活动信息数据存储到数据库中；当商家想要修改活动时，在活动列表中点击要修改的活动，进入要编辑的活动信息界面，对已发布的活动信息进行编辑，填写新的信息后点击保存，系统保存修改后的数据到数据库中；如果商家要删除活动，则在活动列表中对要删除的活动进行勾选，单击删除按钮，系统在数据库中对该活动进行删除标记；如果商家仅选择活动管理不进行上述操作，则系统查询出所有活动的信息以列表形式进行展示。如果是参与者角色登录系统，则进入参与者系统展示参与者权限相应界面。当参与者点击报名活动时，进入活动报名界面，参与者填写活动表单，然后单击提交按钮，系统将报名数据保存到数据库中；活动结束后，如果参与者想要评价活动，选择

已结束的活动，点击进入详情页面，添加评价，然后点击保存，系统保存评价信息到数据库中；如果参与者想要分享活动，进入要分享的活动界面，点击分享按钮，向微信好友发送活动链接或将该活动链接发布到朋友圈，并且系统保存分享数据到数据库中；如果参与者仅点击活动列表不进行上述操作，则系统查询出所有活动信息以活动列表形式进行展示。

## 2. 类图设计

活动管理类图如图 4.8 所示。

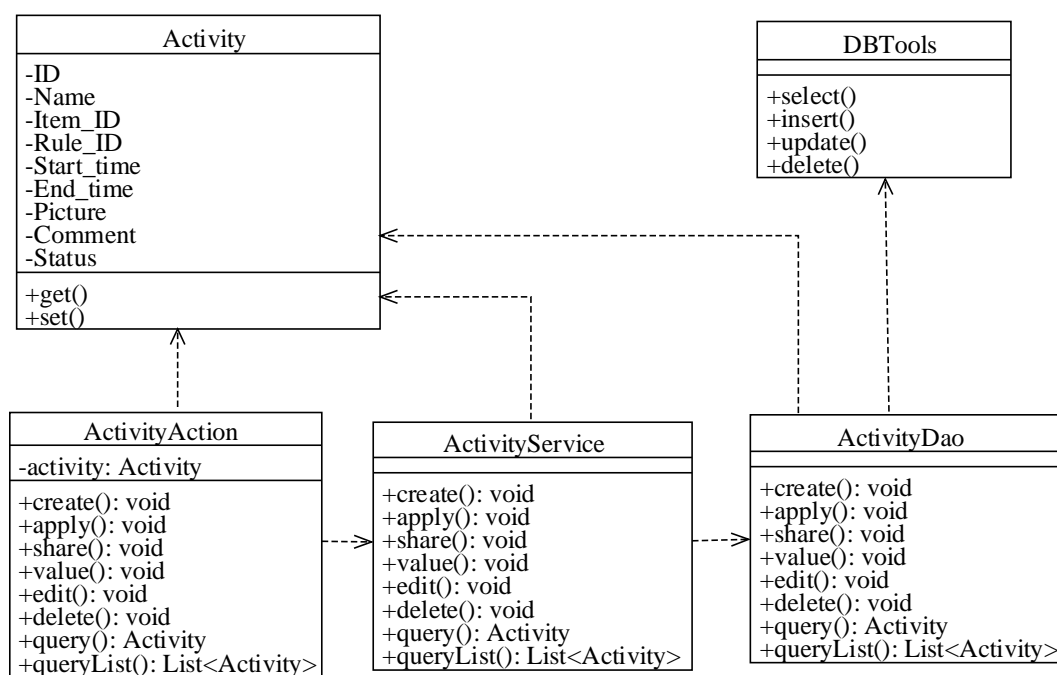


图4.8 活动管理类图

### (1) Activity 类

Activity 类是活动实体类，记录了活动信息。在 Activity 对象里有 ID、Name、Item\_ID、Rule\_ID、Start\_time、End\_time、Picture、Comment、Status 等字段及其 get 和 set 方法。

### (2) ActivityAction 类

ActivityAction 类属于活动管理模块的表现层，负责对页面请求进行处理。ActivityAction 类接收来自浏览器端的请求，将请求数据进行转换和校验，并且将页面请求转换成调用业务控制层的业务逻辑处理和运算。在 ActivityAction 类中有创建 create()方法、报名 apply()方法、分享 share()方法、评价 value()方法、修改 edit()方法、删除 delete()方法、查询 query()方法和批量查询 queryList()方法。

### (3) ActivityService 类

ActivityService 类属于活动管理模块的业务控制层，负责对业务逻辑进行处理。ActivityService 类接收到完成转换和校验的页面请求后，选择合适的业务逻辑规则进行处理，与 ActivityDao 类共同完成对业务逻辑的运算处理，并且为表现层请求处理类返回处理结果。

#### （4）ActivityDao 类

ActivityDao 类属于活动管理模块的数据持久层，负责对业务数据进行处理。数据从业务控制层传递给数据持久层后，都由 ActivityDao 类操作处理后提供给业务逻辑处理类，ActivityDao 类为 ActivityService 类提供数据操作支持，比如对活动信息的增删改查操作。在此之后，调用 DBTools 类实现数据持久化。

#### （5）DBtools 类

DBtools 类属于活动管理模块的数据持久层，负责在数据库中进行增删查改。DBtools 负责调用并执行各项数据库操作，完成数据持久化。比如进行数据库的新增 insert()操作、查询 select()操作、修改 update()操作、删除 delete()操作。

### 3.时序设计

动态调用活动管理类图，对活动管理进行时序设计。活动管理模块的时序设计如图 4.9 所示。

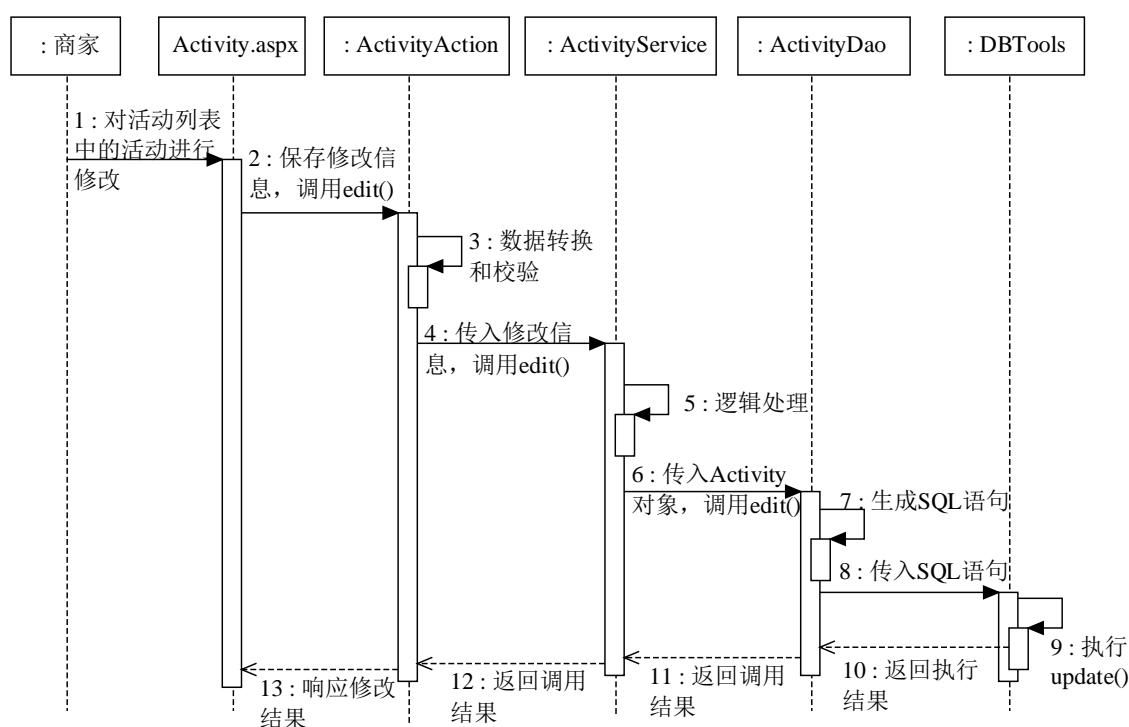


图4.9 活动管理时序图

在图 4.9 的活动管理时序图中，商家在活动管理界面选择活动列表中的活动，点击修改活动按钮，编辑活动表单，然后点击提交，将活动信息存入活动管理表现层处

理类 `ActivityAction`，并调用 `edit()`方法。在表现层处理类 `ActivityAction` 中先对请求数据进行格式转换和校验，然后调用逻辑处理类 `ActivityService` 中的 `edit()`方法。在 `ActivityService` 的 `edit()`方法中完成修改活动的业务逻辑处理部分，接着调用数据处理类 `ActivityDao` 对修改活动业务的数据进行处理，调用 `ActivityDao` 类中的 `edit()`方法生成符合数据库版本要求的 SQL 语句，随后将其作为参数调用活动管理模块的数据库处理类 `DBTools` 中的 `update()`方法，在数据库中对活动信息数据进行更新。处理结束后同步返回调用结果，最后将修改结果响应至商家界面。

### 4.3.3 选项库管理模块设计与实现

选项库管理模块包含选项新增、选项修改、选项删除和选项查询。

#### 1. 流程设计

选项库管理模块具体流程设计图如图 4.10 所示。

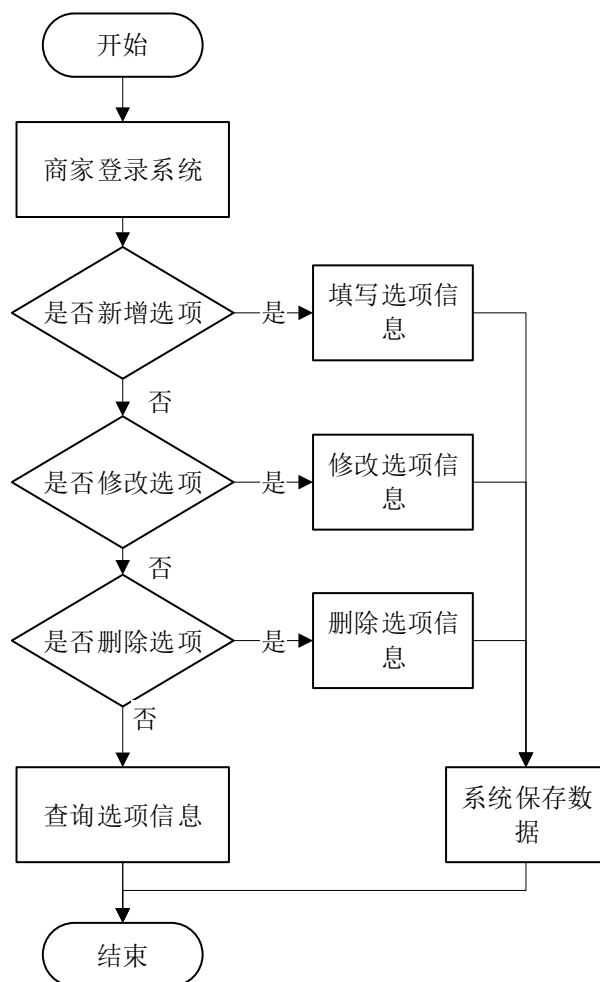


图4.10 选项库管理流程图

在图 4.10 选项库管理流程图中，只有商家用户角色可以进行选项库管理，所以

该流程图表示的是商家角色对选项库的一系列增删查改等管理操作。当商家点击新增选项按钮时,则弹出选项新增窗口,商家在选项新增窗口中填写选项名称、选项类型、选项描述和限制条件,单击保存按钮,系统将新增的选项信息数据存储到数据库中。当商家想要修改选项时,点击具体选项进入选项修改页面,商家编辑选项信息,点击保存按钮,系统将修改后的选项信息数据存储到数据库中。当商家想要删除选项时,选中待删除的选项,点击删除按钮,系统在数据库中为选中的选项做删除标记处理并保存。如果商家仅点击选项管理并未执行上述操作,输入查询条件对选项信息进行查询,那么系统查询出符合要求的所有选项信息,以列表形式进行展示。

## 2.类图设计

选项库管理类图如图 4.11 所示。

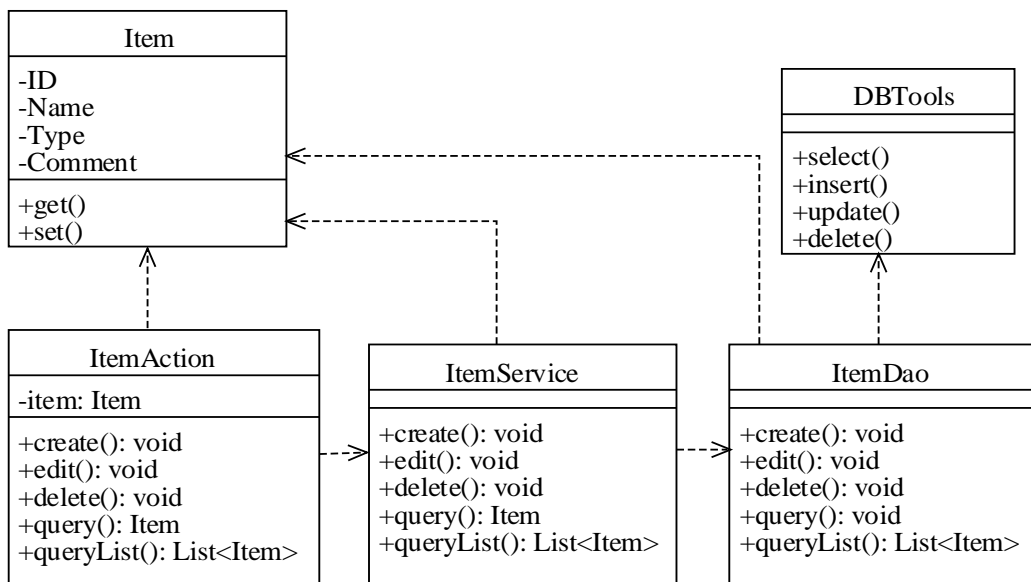


图4.11 选项库管理类图

### (1) Item 类

Item 类是选项对象,记录了选项信息。在 Item 对象里有 ID、Name、Type、Comment 等字段及其 get 和 set 方法。

### (2) ItemAction 类

ItemAction 类属于选项库管理模块的表现层,负责对页面请求进行处理。ItemAction 类获得从浏览器端传来的请求后,对请求数据进行格式转换,将页面请求传递至合适的业务控制层,在业务控制层中根据相应的业务逻辑进行处理。在 ItemAction 类中有创建 create()、修改 edit()、删除 delete()、查询 query()、批量查询 queryList() 5 个方法。

### (3) ItemService 类

ItemService 类属于选项库管理模块的业务控制层,负责对业务逻辑进行处理。

ItemService 类接收到完成转换和校验的页面请求后,选择合适的业务逻辑规则进行处理,与 ItemDao 类共同完成对业务逻辑的运算处理,并且为表现层请求处理类返回处理结果。

#### (4) ItemDao 类

ItemDao 类是属于选项库管理模块的数据持久层,负责对业务数据进行处理。数据从业务控制层传递给数据持久层后,都由 ItemDao 类操作处理后提供给业务逻辑处理类,ItemDao 类为 ItemService 类提供数据操作支持,比如对选项信息的增删改查操作。在此之后,调用 DBTools 类实现数据持久化。

#### (5) DBtools 类

DBtools 类属于选项库管理模块的数据持久层,负责在数据库中进行增删查改。DBtools 负责调用并执行各项数据库操作,完成数据持久化。比如进行数据库的新增 insert()操作、查询 select()操作、修改 update()操作、删除 delete()操作。

### 3.时序设计

动态调用选项库管理类图,对选项库管理进行时序设计。选项库管理模块的时序设计如图 4.12 所示。

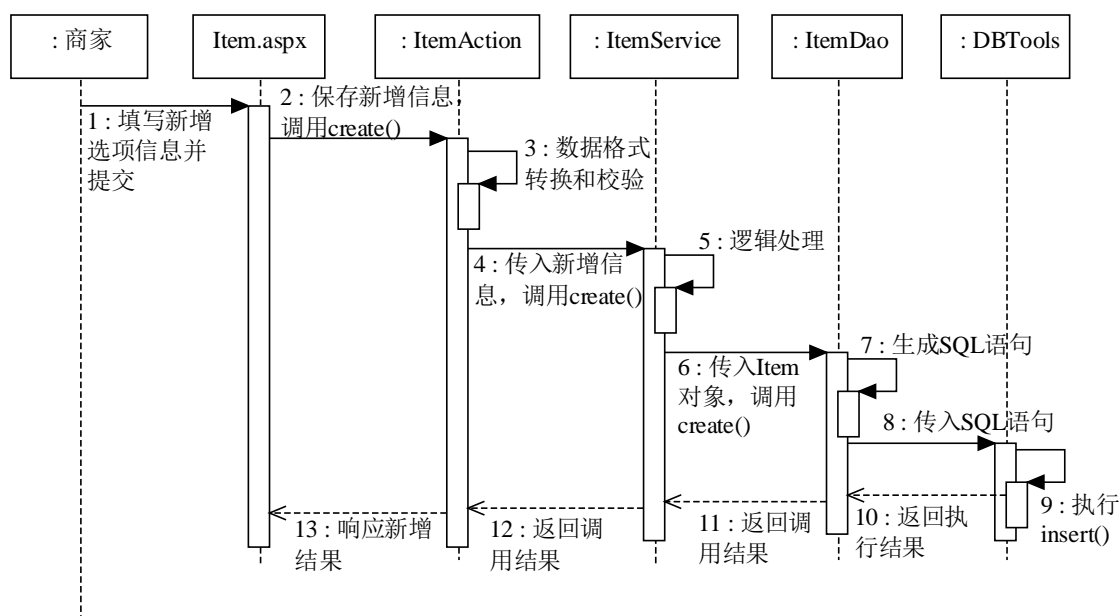


图4.12 选项库管理时序图

在图 4.12 的选项库管理时序图中,商家在选项库管理界面点击新建选项库按钮,填写选项信息,点击提交按钮,将选项信息存入选项库管理表现层请求处理类 ItemAction,对 create()方法进行调用。在表现层请求处理类 ItemAction 中先对请求数据进行格式转换和校验,然后调用逻辑处理类 ItemService 中的 create()方法。在 ItemService 的 create()方法中完成新增选项的业务逻辑处理部分,接着调用数据处理类 ItemDao 进行新增选项业务的数据处理,调用 ItemDao 类中的 create()方法生成符

合数据库版本要求的 SQL 语句，随后将其作为参数调用选项库管理模块的数据库处理类 DBTools 中的 insert()方法，在数据库中对选项信息数据进行更新。处理结束后同步返回调用结果，最后将新增结果响应至商家界面。

#### 4.3.4 商家管理模块设计与实现

商家管理包括对商家账号进行新增、启用、停用、编辑、修改密码等操作。

##### 1. 流程设计

在商家管理模块中，除了对商家基本信息的编辑，如名称、微信公众号、手机号，还有对商家账号进行启用、停用和密码重置的操作。商家管理模块流程设计如图 4.13 商家管理流程图所示。在商家管理模块的所有数据操作过程中，必须对数据进行完整性校验，加密处理关键数据，使用可靠的加密手段进行传输，确保传输过程中数据的安全性和完整性。

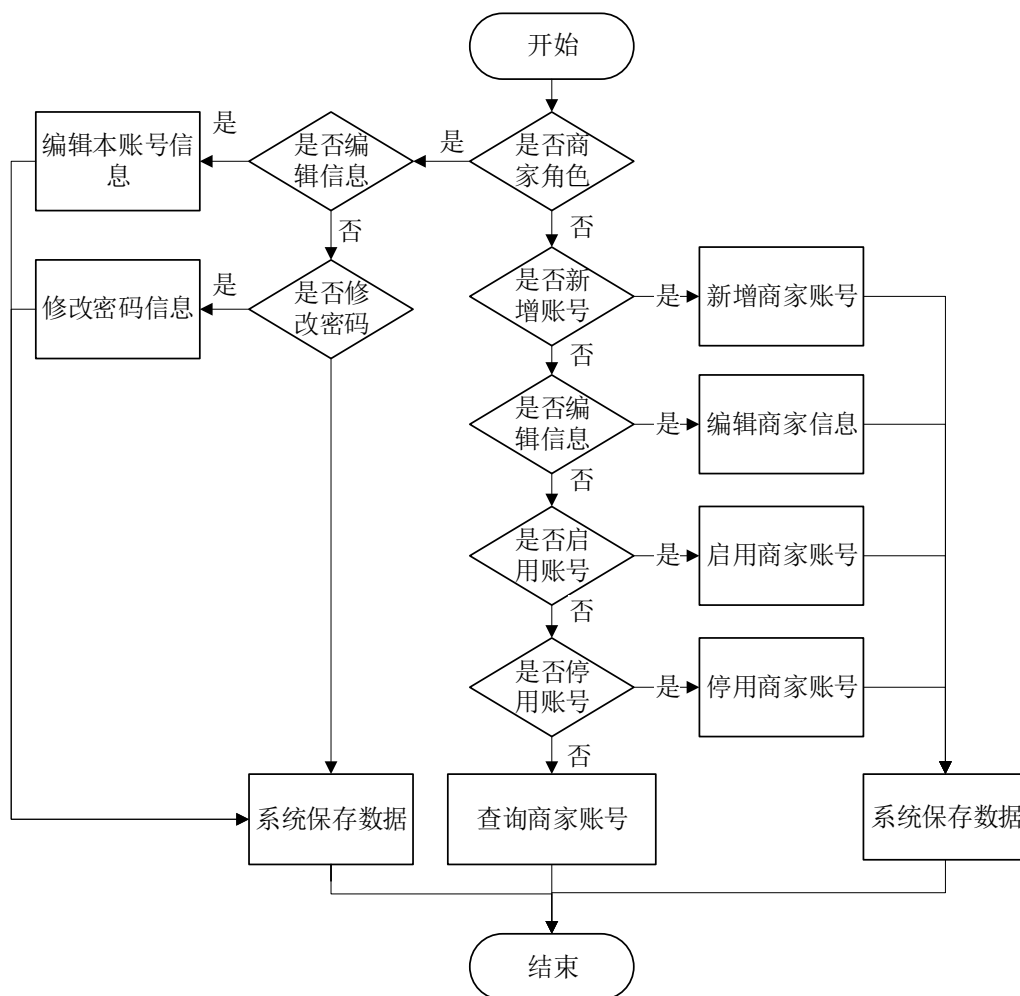


图4.13 商家管理流程图

在图 4.13 商家管理流程图中，用户成功登录系统，系统按照用户角色展示相应

的系统菜单。商家选择商家信息编辑，系统取得表单信息，商家对可编辑项进行修改，点击保存，页面检查必填项是否为空且输入项是否合法等校验通过后，系统保存修改内容到数据库；商家选择修改密码，填写原密码与新密码，页面校验新密码是否符合格式和强度要求且合法，点击提交，系统对用户输入的原密码与数据库中原密码进行比对校验，如果校验通过，系统在数据库中保存新密码并返回修改成功提示。管理员选择商家账号新增，填写新增账号信息，点击提交按钮，系统创建新商家账号并将账号信息存储到数据库中；管理员选择商家信息编辑，系统取得表单信息，管理员对可编辑项进行修改，点击保存，页面检查必填项是否为空且输入项是否合法等校验通过后，系统保存修改内容到数据库；管理员选择商家账号停用，选择商家账号并点击停用按钮，商家状态变更为停用，系统保存变更结果；管理员选择商家账号启用，选择已停用的商家账号并点击启用按钮，商家状态变更为启用，系统保存变更结果。

## 2.类图设计

商家管理类图如图 4.14 所示。

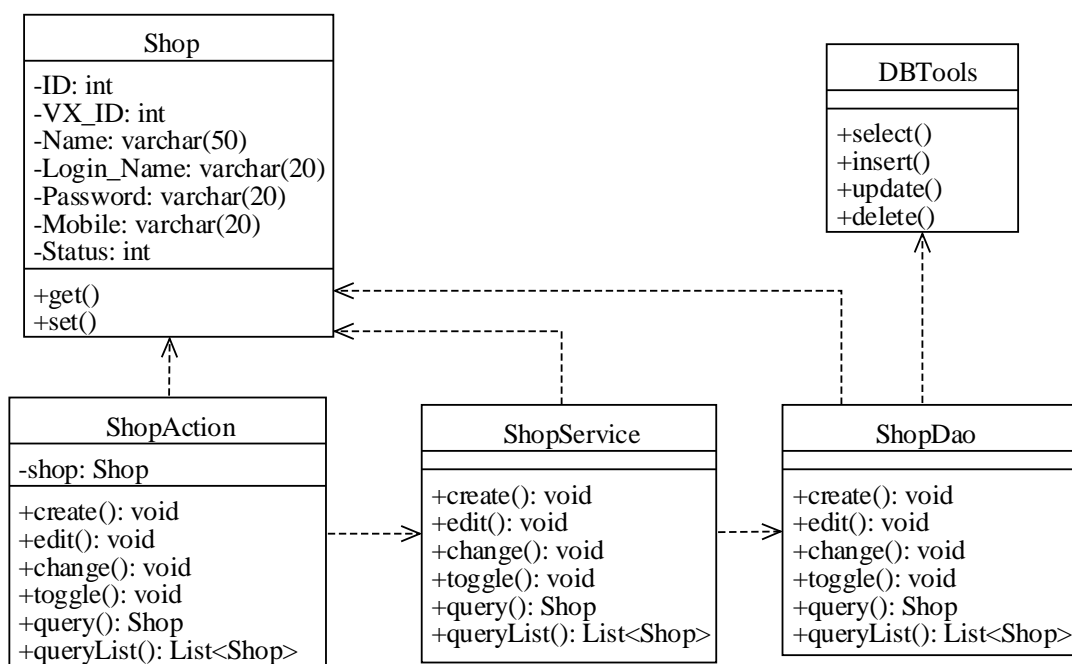


图4.14 商家管理类图

### (1) Shop 类

Shop 类是商家对象，记录了商家信息。在 Shop 对象里有 ID、VX\_ID、Name、Login\_Name、Password、Mobile、Status 等字段及其 get 和 set 方法。

### (2) ShopAction 类

ShopAction 类属于商家管理模块的表现层，负责对页面请求进行处理。ShopAction 类获得从浏览器端传来的请求后，对请求数据进行格式转换，将页面请求



传递至合适的业务控制层，在业务控制层中根据相应的业务逻辑进行处理。在 ShopAction 类中有创建 create()、编辑 edit()、修改密码 change()、切换状态 toggle()、查询 query()、批量查询 queryList() 5 个方法。

### (3) ShopService 类

ShopService 类属于商家管理模块的业务控制层，负责对业务逻辑进行处理。ShopService 类接收到完成转换和校验的页面请求后，选择合适的业务逻辑规则进行处理，与 ShopDao 类共同完成对业务逻辑的运算处理，并且为表现层请求处理类返回处理结果。

### (4) ShopDao 类

ShopDao 类属于商家管理模块的数据持久层，负责对业务数据进行处理。数据从业务控制层传递给数据持久层后，都由 ShopDao 类操作处理后提供给逻辑处理类，ShopDao 类为 ShopService 类提供数据操作支持，比如对商家信息的增删改查操作。在此之后，调用 DBTools 类实现数据持久化。

### (5) DBtools 类

DBtools 类属于商家管理模块的数据持久层，负责在数据库中进行增删查改。DBtools 负责调用并执行各项数据库操作，完成数据持久化。比如进行数据库的新增 insert()操作、查询 select()操作、修改 update()操作、删除 delete()操作。

## 3. 时序设计

使用商家管理的时序图对商家管理类图进行动态的调用说明。商家管理时序图如图 4.15 所示。

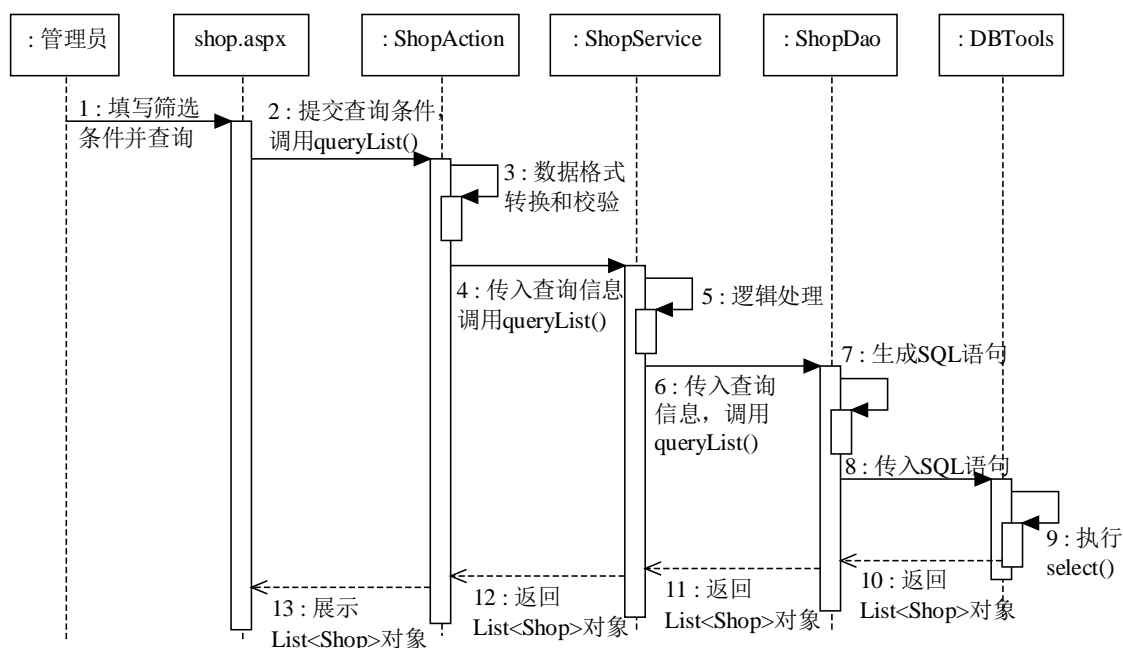


图4.15 商家管理时序图

商家管理在商家角色中的业务实现逻辑为：商家选择修改密码，填写修改密码表单，包括原密码与新密码，页面与系统共同进行格式校验、必填校验、正确性校验和唯一性校验等检查，校验通过后系统将新密码保存至数据库；商家选择个人信息管理，返回个人信息表单，对其进行编辑，页面进行一部分格式校验，点击保存后，对编辑的信息进行必填校验、唯一性校验和正确性校验等，校验通过后将修改后数据保存至数据库，返回结果并响应至用户界面。在管理员角色中的业务实现逻辑为：管理员选择商家管理菜单，输入查询商家条件，通过过滤筛选条件查询到目标商家，管理员选择停用/启用商家账号，则点击停用/启用按钮进行状态切换，系统保存状态数据至数据库；管理员选择编辑信息，则点击该商家获取其信息表单，对其进行编辑，页面进行一部分格式校验，点击保存后，对编辑的信息进行必填校验、唯一性校验和正确性校验等，校验通过后将修改后数据保存至数据库，返回结果并响应至用户界面。

#### 4.3.5 统计管理设计与实现

统计管理模块包括对活动及用户数据的可视化统计分析，并导出统计报表的功能。

统计管理模块与系统管理模块具有一定程度的关联，系统管理策略对统计分析策略具有决定性的影响，可通过开放不同等级的接口来控制为商家提供的统计分析内容，比如参与者性别比例、地域分布、年龄分布、收入分布和职业分布。统计管理流程图如图 4.16 所示。

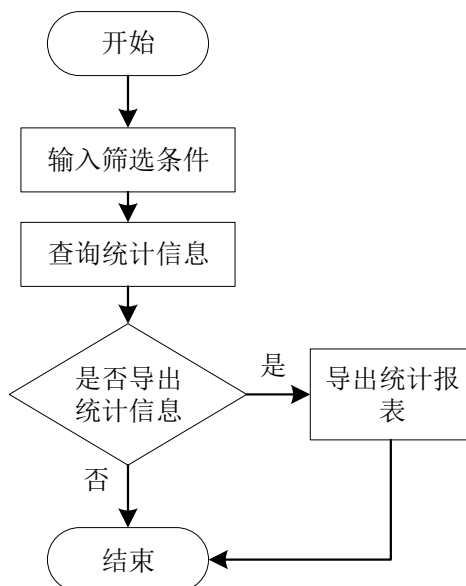


图4.16 统计管理流程图

在图 4.16 统计管理流程图中，商家选择统计管理菜单，系统默认返回一自然年内全部活动统计分析图表，包括按月分布的活动数量、相应活动的参与者数量、活动分享数量月分布、活动好评率月分布等统计信息。商家输入筛选条件进行查询，比如

时间范围选择 2018/06/01—2018/12/31，系统根据时间从数据库中查询到相关活动，利用数据库由预设统计分析组件进行统计分析后获得数据报表展示在用户界面上，商家可将当前查看的统计信息导出到本地文档中，点击导出按钮，系统导出组件将该统计报表导出到商家指定文件地址中。

## 2.时序设计

使用统计管理的时序图对统计管理进行动态的业务逻辑说明。统计管理时序设计如图 4.17 所示。

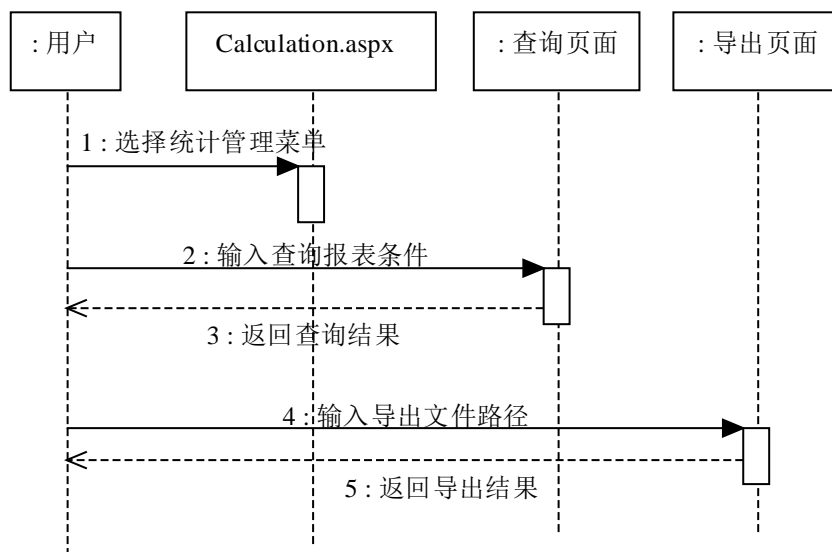


图4.17 统计管理时序图

统计管理业务实现逻辑为：商家选择统计管理菜单，查看默认返回的全部活动统计分析图表，包括按月分布的活动数量、相应活动的参与者数量、活动分享数量月分布、活动好评率月分布等统计信息。输入筛选条件，商家点击查询，系统根据筛选条件过滤数据后从数据库中取得相关活动信息，利用数据库由预设统计分析组件进行统计分析后获得数据报表展示在用户界面上，商家选择合适的筛选条件，系统展示经数据库组件加工后的数据报表，商家点击导出按钮，系统导出组件将该统计报表导出到商家指定文件地址中。

## 4.3.6 系统管理设计与实现

系统管理模块是对密码、系统日志和数据库进行管理的模块。

### 1.类图设计

系统管理模块具体类图设计如图 4.18 系统管理类图所示。

本系统所有功能模块均包含 DBTools 数据库操作类，在不同功能模块中分别用于数据请求处理、数据传输和后台数据库的增删改查等。与之前几个功能模块不同，系统管理功能模块业务实现过程中对数据库的交互请求更加灵活和频繁，涉及到关联多

个表的数据库操作，为了更方便地执行相关 SQL 语句，在系统管理模块中，取消设计 Action 和 Service 两个类,直接由请求处理层代码分析页面请求，转换数据格式并完成校验，调用 DBTools 数据库处理类进行数据读写和传输。

(1) DBTools 类：在 DBTools.cs 文件中对 DBTools 类进行定义和说明，DBTools 类主要负责后台数据库中各个表信息的管理操作，并且为系统正常稳定运行提供数据结构方面的保证。

(2) ResetpasswordAction 类：在 ResetpasswordAction.cs 代码文件中对 ResetpasswordAction 类进行定义和说明，ResetpasswordAction 类主要负责为用户修改密码功能提供数据的传输读写并将数据保存至数据库。

(3) LogAction 类：在 LogAction.cs 文件中对 LogAction 类进行定义和说明，LogAction 类主要负责系统的日志记录和查看，以及针对长时间运行后的冗余垃圾日志待清理的情况对用户进行提醒操作。

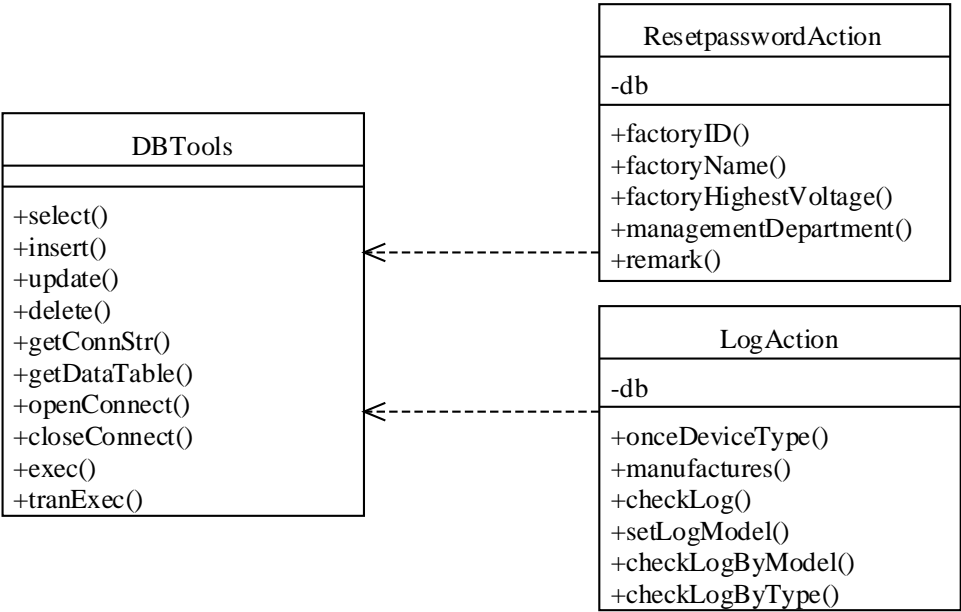


图4.18 系统管理模块类图

2.具体实现

系统管理主要指的是对日志、密码和数据库的管理。在一般的系统管理中，密码管理的实现基本均为通过解析管理员的管理请求，通过校验后调用数据库操作更新密码完成密码重置，因此工作主要在于日志管理和数据库管理两部分的运算处理，下面具体分析介绍这两部分的实现工作：

(1) 日志管理功能实现

日志管理包括对系统日志信息进行查看和后台记录的功能。系统的日志信息保存在系统后台数据库中，在实现过程中由请求处理层代码进行业务逻辑处理，以操作类

型、操作用户名称、操作用户 IP 地址、操作内容、操作时间和操作结果等作为传入参数，调用 `LogAction` 类中的 `manufacture()` 方法接口并且用以上参数构建相应 SQL 语句，调用 `DBToolsI` 类中的 `exec()` 方法将其写入数据库，从而实现系统日志信息的后台记录功能。在实现日志管理功能中的系统日志信息查看功能时，直接通过 `DBTools` 类对系统后台数据库中日志信息表进行查询检索返回日志信息对象列表即可。

#### （2）数据库管理功能实现

数据库管理功能包括数据库的备份功能、还原功能和清理功能，在实现过程中直接由请求处理层代码直接调用 `DBTools` 数据库处理类。实现备份功能需要系统管理员预先设置备份文件保存路径，通过调用 `exec()` 方法执行编写好的 `Restore SQL` 语句实现系统数据库备份功能。系统管理员调用 `exec()` 方法执行编写好的 `BACKUP` 和 `CLEAR` 等 SQL 语句实现数据库的还原和清理功能。

## 4.4 本章小结

本章介绍了智能匹配系统的设计和实现工作，主要包括了系统总体设计工作、系统数据库设计工作和系统功能设计与具体实现工作。首先对系统进行了深入研究，介绍了系统采用的三层架构、具体的网络结构和功能结构；其次对系统进行了详细分析，结合 E-R 图和数据字典阐述了系统的数据模型设计工作和数据表结构设计工作；最后对用户管理模块、活动管理模块、选项库管理模块、商家管理模块、统计管理模块和系统管理模块的详细设计与实现工作进行具体论述。



## 第五章 系统测试与分析

在完整详细的需求分析和系统设计工作的指导下，完成了系统的代码实现工作后，本章将对系统进行测试并分析结果。本章针对系统的功能和性能两个方面开展系统的测试工作，配合测试用例进行具体说明，根据测试结果给出相应的分析与评价。

### 5.1 系统测试环境

系统测试工作开始之前，应该先按照系统的功能定位和测试工作计划选择搭建合适的测试环境，这是顺利完成测试工作并保证分析结果准确性的前提。系统的测试环境包括网络和软硬件配置<sup>[35]</sup>。

系统测试工作中，通过使用用户常用软件、主流操作系统和软件平台来模拟本系统真实运行环境，在测试环境中配置两台服务器主机和一套外接磁盘阵列来完成对双机并行架构的模拟，其中一台主机用于部署系统的应用服务，另一台用于部署系统的数据库管理，同时，选择外置 RAID 磁盘阵列连接两台服务器构建双机集群运行环境。系统两台服务器测试环境具体参数如表 5.1 所示。

表5.1 服务器端测试环境

|         |                            |
|---------|----------------------------|
| 操作系统    | Windows10-64 位企业版          |
| 内存      | 4GB(DDR3 1600MHz)          |
| 处理器     | 英特尔 Intel i5 双核处理器 2.50GHz |
| 数据库管理系统 | MySQL 5.6                  |

本系统主要使用个人电脑完成测试工作，将两台电脑设置为测试机，使用 LoadRunner 工具进行性能测试，测试机的软硬件参数如表 5.2 所示。

表5.2 用户端测试环境

|        |                            |
|--------|----------------------------|
| 操作系统   | Windows10-64 位企业版          |
| 内存     | 4GB(DDR3 1600MHz)          |
| 处理器    | 英特尔 Intel i5 双核处理器 2.50GHz |
| 硬盘     | 2TB 7200 转机械硬盘             |
| 性能测试工具 | LoadRunner 11.0            |

5.2 测试流程与方法

智能匹配系统的一系列测试工作的有序进行需要测试流程的指导，因此必须对测试工作流程作出详细安排。考虑到系统的具体功能及使用场景，使用黑盒测试的方法进行测试，本节对测试工作的工作流程和选择的测试方法作出必要的介绍和说明。

5.2.1 测试流程

智能匹配系统的功能测试工作由测试人员来手动进行测试，在测试文档中编写详细的测试用例，根据测试用例对系统进行全面完善的测试，测试用户包括商家用户、参与者用户和管理员用户，对用户管理、活动管理、选项库管理、商家管理、统计管理和系统管理等核心功能模块进行测试。在完成测试后，将测试结果记录在测试文档中，对于测试中出现的 Bug 采用 Bug 报表的形式进行整理记录，并将维护工作作相应记录。测试工作具体安排如图 5.1 所示。

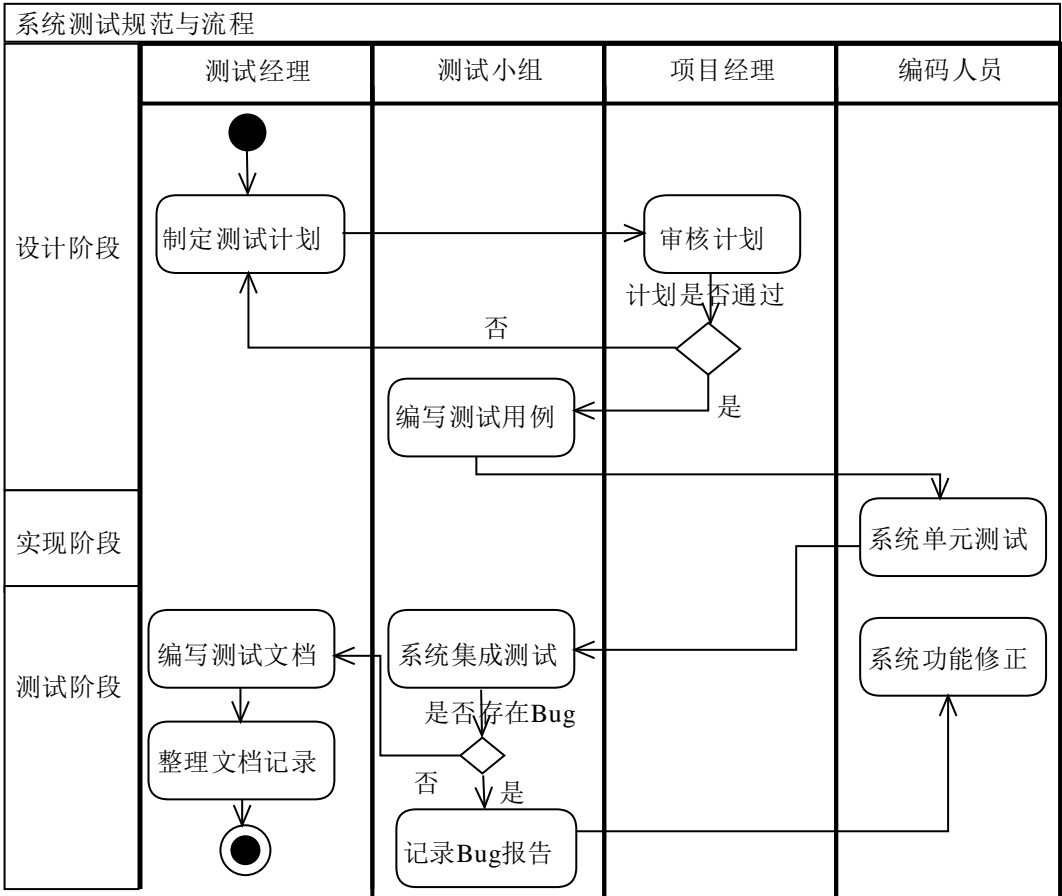


图5.1 测试工作流程

本系统的测试工作主要分为搭建测试环境、编写测试用例、执行功能测试、进行性能测试和记录 Bug 等环节。

1. 搭建测试环境



模拟系统的实际运行环境和使用场景，搭建测试环境，测试主机按照用户端测试环境要求进行搭建并部署相关应用程序来模拟客户端使用情况，按照系统服务器端测试环境要求选择合适的系统应用服务器主机并完成系统服务部署。

### 2.编写测试用例

为智能匹配系统编写完善齐全的用例，用于指导实际测试工作。测试用例用于功能测试，对系统六个功能模块具体测试的前置条件、步骤描述、期望结果、测试状态进行编写，性能测试使用专用性能测试工具测试，不需要撰写测试用例。

### 3.执行功能测试

编写完测试用例后，测试人员按照测试用例的内容执行功能测试。测试人员对系统各个功能进行测试，对测试情况进行记录，出现异常情况时测试人员必须及时作出反馈，最终汇总生成功能测试文档。

### 4.进行性能测试

功能测试结束后，测试人员对系统进行性能测试，性能测试一般包括负载测试、强度测试和容量测试。使用 LoadRunner 自动化测试工具对边界条件下系统的性能表现进行测试，观察并记录系统表现，完成性能检测报告。

### 5.记录 Bug

完成具体测试工作后，测试人员将测试中遇到的 Bug 信息反馈给开发人员，开发人员针对系统缺陷作出修改调整，必要时升级硬件配置。

## 5.2.2 测试方法

系统的功能测试除了伴随项目开发进行的单元测试，还有在开发完成后统一进行的集成测试。单元测试指的是系统开发人员在编写代码时，对最小可测试单元，比如单个模块、单个函数、单个类或者单个方法，进行正确性的检验，集成测试是在开发完成后，已通过单元测试的基础上，在系统集成时通过将代码单元和测试单元组合起来，对其组合结果进行正确性的检查。单元测试的测试重点在于单个可测试单元，比如类或函数的正确性；集成测试的测试重点在于全局工作中模块间的衔接、参数传递和方法调用的正确性，系统能否在集成后正常稳定地工作。集成测试包括黑盒测试和白盒测试两部分，黑盒测试是指在不关心代码逻辑是否符合业务逻辑情况下，只通过传入参数和输出结果对系统功能进行正确性的检验，从而确定系统是否存在功能上的缺陷；白盒测试是指测试人员主要对代码的具体实现过程进行检查，确保代码逻辑符合业务逻辑，避免隐性错误。

除了功能测试，系统测试还包括性能测试。使用自动化测试工具 LoadRunner 对用户实际使用过程中可能出现的多种正常、峰值和异常负载情况进行模拟，对系统进行测试，记录各项性能指标，比如系统 CPU 使用量、服务器响应时间和吞吐量等，

以此来评估系统的性能表现。

5.3 系统功能测试

系统的功能测试工作主要依赖测试人员的人工操作，测试人员按照编写好的测试用例完成各项操作，判断系统是否达到了预期的功能目标。本节通过列举多个测试用例对系统的功能测试具体工作进行介绍说明。

5.3.1 用户管理模块测试用例

用户管理模块包括用户信息管理和用户状态管理。用户管理模块基本测试用例表如表 5.3 所示。

表5.3 用户管理测试用例表

|        |  |            |                                     |        |
|--------|--|------------|-------------------------------------|--------|
| 设计目的   | 检验用户管理模块各项功能是否按照预期执行   |            |                                     |        |
| 测试指导   | 需求分析阶段关于“用户管理”的需求说明  |            | 用户角色                                | 商家、参与者 |
| 测试数据   | 用户基本信息和用户状态信息  |            |                                     |        |
| 功能名称   | 步骤描述   | 前置条件       | 预期结果                                | 测试状态   |
| 用户信息管理 | 1.测试人员进入用户信息管理界面；<br>2.测试人员修改基本信息；<br>3.测试人员单击保存按钮，完成用户信息管理。 | 测试人员成功登录系统 | 数据库中用户信息已被修改，在参与者列表中能够观察到修改后的参与者信息。 | 测试通过   |
| 用户状态管理 | 1.测试人员进入用户信息管理界面；<br>2.测试人员单击启用/禁用切换按钮；<br>3.系统显示切换后的用户状态。   | 测试人员成功登录系统 | 数据库中用户状态已完成切换，在列表中能够观察到参与者状态改变。     | 测试通过   |

在用户管理模块中，商家点击“用户管理”，系统显示用户列表，商家点击具体用户，进入用户信息管理界面。在页面加载完成后，系统展示用户基本信息。在用户信息管理界面，商家可以浏览并修改用户基本信息，完成后单击“保存”按钮，信息修改成功，修改后的用户信息被存储在数据库中。用户信息管理界面如图 5.2 所示。

用户基本信息

姓名

张三

微信号

12345678aa

性别

男

生日

1999/12/12

手机号码

13311112222

保存

图5.2 用户信息管理界面

5.3.2 活动管理模块测试用例

活动管理模块中包含活动发布、用户参与、匹配查询、活动查询、活动分享、活动评价、活动修改和活动删除等功能。活动管理模块的测试用例表如表 5.4 所示。

表5.4 活动管理测试用例表

|      |  |            |                                   |        |
|------|--|------------|-----------------------------------|--------|
| 设计目的 | 检验活动管理模块各项功能是否按照预期执行                                       |            |                                   |        |
| 测试指导 | 需求分析阶段关于“活动管理”的需求说明  |            | 用户角色                              | 商家、参与者 |
| 测试数据 | 活动发布、申请、举办、评价过程中的数据  |            |                                   |        |
| 功能名称 | 步骤描述   | 前置条件       | 预期结果                              | 测试状态   |
| 活动发布 | 1.测试人员进入活动发布界面；<br>2.测试人员填写基本活动信息；<br>3.测试人员单击提交按钮，完成活动发布。 | 测试人员成功登录系统 | 测试人员正常完成活动发布操作，在数据库中能够查看新发布的活动信息。 | 测试通过   |
| 用户参与 | 1.测试人员进入活动参与界面；<br>2.测试人员填写个人信息；<br>2.测试人员单击提交按钮，完成用户参与。   | 测试人员成功登录系统 | 测试人员正常完成用户参与操作，在数据库中能够查看参与者个人信息。  | 测试通过   |
| 匹配查询 | 1.测试人员进入匹配查询界面；<br>2.测试人员点击具体活动查看匹配结果；<br>3.系统显示活动匹配分组信息。  | 测试人员成功登录系统 | 查询结果与数据库中匹配结果信息一致。                | 测试通过   |

续表 5.4 活动管理测试用例表

| 功能名称 | 步骤描述   | 前置条件               | 预期结果                               | 测试状态 |
|------|--|--------------------|------------------------------------|------|
| 活动查询 | 1.测试人员进入活动查询界面;<br>2.测试人员输入查询条件;<br>3.测试人员点击查询按钮;<br>4.系统展示活动记录。                                   | 测试人员<br>成功登录<br>系统 | 结果符合筛选要求, 查询结果与数据库中活动信息一致。         | 测试通过 |
| 活动分享 | 1.测试人员进入活动参与界面;<br>2.测试人员点击分享按钮;<br>3.测试人员在弹出的分享窗口中选择分享至“微信好友/朋友圈/QQ 好友”;<br>4.测试人员点击发送按钮, 完成活动分享。 | 测试人员<br>成功登录<br>系统 | 测试人员正常完成活动分享操作, 并能查看分享链接。          | 测试通过 |
| 活动评价 | 1.测试人员进入活动评价界面;<br>2.测试人员填写评价信息;<br>2.测试人员单击提交按钮, 完成活动评价。  | 测试人员<br>成功登录<br>系统 | 测试人员正常完成活动评价操作, 在数据库中查询到评价信息。      | 测试通过 |
| 活动修改 | 1.测试人员进入活动管理活动修改界面;<br>2.测试人员编辑活动信息;<br>2.测试人员单击提交按钮, 完成活动修改。                                      | 测试人员<br>成功登录<br>系统 | 测试人员正常完成活动修改操作, 在数据库中查询到活动信息已完成修改。 | 测试通过 |
| 活动删除 | 1.测试人员进入活动管理界面;<br>2.测试人员点击某项活动的删除按钮;<br>3.测试人员在弹出的窗口中选择“确定删除”;<br>4.系统展示删除后的活动列表。                 | 测试人员<br>成功登录<br>系统 | 测试人员正常完成活动删除操作, 在数据库中查询到活动信息已完成删除。 | 测试通过 |

活动管理模块中, 商家点击“活动管理”, 选择“活动发布”, 进入活动发布界面。在页面加载完成后, 系统自动显示选项库中所有备选项和系统预设好的匹配规则项。在活动发布界面, 商家依次填写活动内容信息、选择基本信息选项和匹配规则项, 然

后上传图片，确认无误后点击“发布”按钮，活动发布成功，新增活动信息保存至数据库中。活动发布界面如图 5.3 所示。

姓名  
填写你的名字

微信  
填写微信号可以搜索添加，确保正确

邮箱  
确保填写正确

手机  
确保填写正确

• 选择项 + 添加选择项

性别  
男 or 女

匹配信息

• 选择项 + 添加选择项

图片信息

图5.3 活动发布界面

5.3.3 选项库管理模块测试用例

选项库管理模块中包含选项新增、选项修改、选项删除和选项查询等功能。选项库管理模块的测试用例表如表 5.5 所示。

表5.5 选项库管理测试用例表

|      |   |            |                  |      |
|------|---|------------|------------------|------|
| 设计目的 | 验证选项库管理模块各项功能是否按照预期执行   |            |                  |      |
| 测试指导 | 需求分析阶段关于“选项库管理”的需求说明  |            | 用户角色             | 商家用户 |
| 测试数据 | 选项基本信息  |            |                  |      |
| 功能名称 | 步骤描述  | 前置条件       | 预期结果             | 测试状态 |
| 选项新增 | 1.测试人员进入选项库管理界面；<br>2.测试人员单击新增按钮；<br>3.测试人员在弹出的选项新增窗口中填写选项名称，选择选项类型，填写选项描述；<br>4.测试人员单击“添加完成”按钮，完成选项新增。 | 测试人员成功登录系统 | 在数据库中能够查看新增选项信息。 | 测试通过 |

续表 5.5 选项库管理测试用例表

| 功能名称 | 步骤描述   | 前置条件       | 预期结果                             | 测试状态 |
|------|--|------------|----------------------------------|------|
| 选项修改 | 1.测试人员进入选项库管理界面；<br>2.测试人员单击具体选项对应的修改按钮；<br>3.测试人员在弹出的选项信息窗口中编辑选项信息；<br>4.测试人员单击提交按钮，完成选项修改。 | 测试人员成功登录系统 | 在数据库中<br>可以查询到<br>选项信息已<br>完成修改。 | 测试通过 |
| 选项删除 | 1.测试人员进入选项管理界面；<br>2.测试人员单击具体选项对应的删除按钮；<br>3.测试人员在弹出的确认窗口中选择“确定删除”；<br>4.系统展示删除后的选项列表。       | 测试人员成功登录系统 | 在数据库中<br>可以查询到<br>选项信息已<br>被删除。  | 测试通过 |
| 选项查询 | 1.测试人员进入选项管理界面；<br>2.测试人员在查询框中填写选项名称；<br>3.测试人员单击查询按钮；<br>3.系统展示选项信息查询结果。                    | 测试人员成功登录系统 | 查询结果与<br>数据库中选项信息一致。             | 测试通过 |

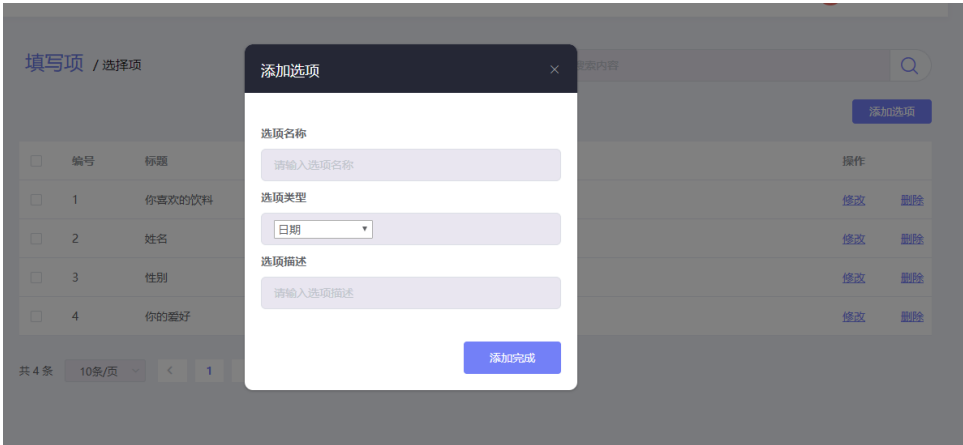


图5.4 选项新增窗口

商家登录系统，进入选项管理模块，点击选项新增按钮，在图 5.4 选项新增窗口中填写选项名称、选项类型、选项描述、限制条件。点击提交后，系统将新增的选项信息保存至数据库中，后续商家发布活动时即可选用。

### 5.3.4 商家管理测试

商家管理模块中包括商家账号新增、启用、停用、编辑和修改密码等功能。商家管理模块测试用例如表 5.6 所示。

表5.6 商家管理测试用例表

|      |   |                    |                    |      |
|------|---|--------------------|--------------------|------|
| 设计目的 | 验证商家管理模块各项功能是否按照预期执行  |                    |                    |      |
| 测试指导 | 需求分析阶段关于“商家管理”的需求说明   |                    | 用户角色               | 商家用户 |
| 测试数据 | 商家基本信息  |                    |                    |      |
| 功能名称 | 步骤描述  | 前置条件               | 预期结果               | 测试状态 |
| 账号新增 | 1.测试人员进入商家管理界面；<br>2.测试人员单击账号新增按钮；<br>4.测试人员填写新增账号信息；<br>5.测试人员单击提交按钮，完成账号新增。 | 测试人员<br>成功登录<br>系统 | 在数据库中能够查看新增商家账号信息。 | 测试通过 |
| 账号启用 | 1.测试人员进入商家管理界面；<br>2.测试人员单击具体商家的启用按钮，完成账号启用。                                  | 测试人员<br>成功登录<br>系统 | 数据库中商家账号已启用。       | 测试通过 |
| 账号停用 | 1.测试人员进入商家管理界面；<br>2.测试人员单击具体商家的停用按钮，完成账号停用。                                  | 测试人员<br>成功登录<br>系统 | 数据库中商家账号已停用。       | 测试通过 |
| 信息编辑 | 1.测试人员选择某一具体商家账号，点击进入商家信息编辑界面；<br>2.测试人员编辑商家信息；<br>3.测试人员单击提交按钮，完成商家信息编辑。     | 测试人员<br>成功登录<br>系统 | 数据库中商家信息已完成编辑。     | 测试通过 |
| 密码修改 | 1.测试人员进入商家管理密码修改界面；<br>2.测试人员填写密码修改信息；<br>3.测试人员单击提交按钮，完成商家密码修改。              | 测试人员<br>成功登录<br>系统 | 数据库中商家密码已变更。       | 测试通过 |

在商家管理模块中，商家账号新增的界面如图 5.6 商家注册界面所示。在商家信息表单中填写商家微信公众平台 ID（商家账号与该微信公众平台进行绑定）、商家名称、手机号码等，然后在账号信息中为新增的商家账号设置登录密码，如果不设置则默认为初始密码 123456789，最后点击注册，系统为商家分配商家账号。

智能匹配系统

首页 > 商家管理

admin 登

• 商家信息

姓名

填写你的名字

微信公众号

请填写正确微信公众号用于关联

手机

确保填写正确

• 账号信息

密码

默认设置123456789

图5.5 商家注册界面

5.3.5 统计管理测试

统计管理模块包括对活动及用户进行根据时间范围划分的多维度统计分析和对统计报表进行导出。统计管理模块测试用例如表 5.7 所示。

表5.7 统计管理测试用例表

|            |   |                    |  |      |
|------------|---|--------------------|--|------|
| 设计目的       | 验证统计管理模块各项功能是否按照预期执行  |                    |  |      |
| 测试指导       | 需求分析阶段关于“统计管理”的需求说明   |                    | 用户角色   | 商家用户 |
| 测试数据       | 活动发布、参与、分享和评价相关数据   |                    |  |      |
| 功能名称       | 步骤描述  | 前置条件               | 预期结果   | 测试状态 |
| 统计报表<br>查询 | 1.测试人员进入统计管理界面；<br>2.测试人员选择时间范围；<br>3 测试人员单击查询按钮；<br>4.系统展示符合条件的活动统计分析报表。             | 测试人员<br>成功登录<br>系统 | 图表内容<br>与数据库<br>中一致，<br>显示格式<br>正常。          | 测试通过 |
| 统计报表<br>导出 | 1.测试人员进入统计管理界面；<br>2.测试人员单击导出按钮；<br>2.测试人员在弹出窗口中输入导出文件存储路径；<br>3.测试人员单击确定按钮，完成统计报表导出。 | 测试人员<br>成功登录<br>系统 | 导出报表<br>内容与数<br>据库中一<br>致，显示<br>格式正常<br>无乱码。 | 测试通过 |

商家登录系统，选择统计管理菜单，进入统计管理界面，在图 5.6 统计管理界面的筛选条件中选择查询活动的时间范围。单击查询按钮后，系统从数据库中取出符合筛选条件的活动，自动进行统计分析生成图表，将结果响应到用户界面上。



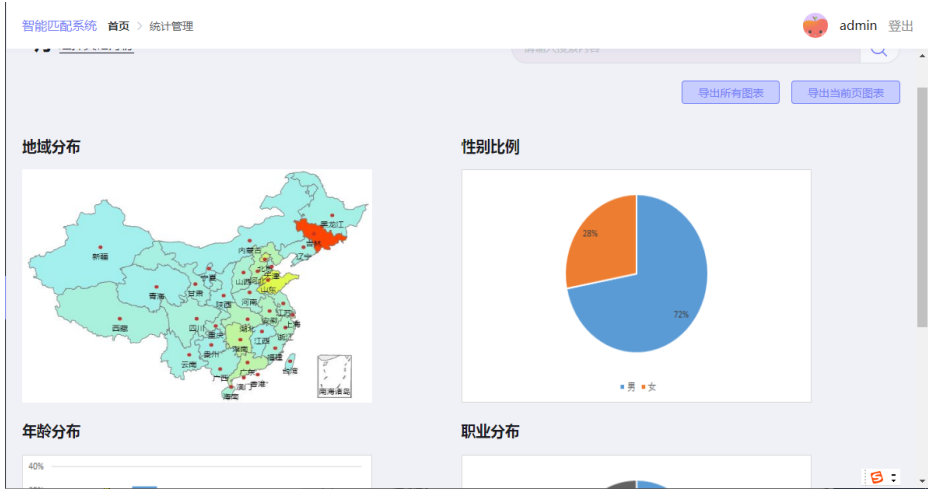


图5.6 统计管理界面

5.3.6 系统管理测试

系统管理模块主要包括对密码、日志和数据库的管理。系统管理功能测试用例如表 5.8 所示。

表5.8 系统管理测试用例表

|       |  |                    |                       |       |
|-------|--|--------------------|-----------------------|-------|
| 设计目的  | 验证系统管理模块各项功能是否按照预期执行   |                    |                       |       |
| 测试指导  | 需求分析阶段关于“系统管理”的需求说明  |                    | 用户角色                  | 系统管理员 |
| 测试数据  | 商家数据、活动数据以及系统操作数据  |                    |                       |       |
| 功能名称  | 步骤描述   | 前置条件               | 预期结果                  | 测试状态  |
| 密码修改  | 1.测试人员进入系统管理页面；<br>2.测试人员点击某用户密码修改按钮；<br>3.测试人员在弹出的密码管理页面输入新密码并确认新密码；<br>4.测试人员单击完成按钮。                     | 测试人员<br>成功登录<br>系统 | 数据库中密码已被修改为新密码。       | 测试通过  |
| 数据库管理 | 1.测试人员进入系统管理界面；<br>2.测试人员单击数据库备份按钮；<br>3.测试人员在弹出的窗口中选择备份文件存储路径，并单击提交按钮；<br>4.系统将数据库备份到指定路径并返回提示，完成数据库备份管理。 | 测试人员<br>成功登录<br>系统 | 备份文件与数据库中一致，内容格式均无错误。 | 测试通过  |
| 日志管理  | 1.测试人员进入系统管理界面；<br>2.测试人员点击查看日志按钮；<br>3.系统展示日志信息。  | 测试人员<br>成功登录<br>系统 | 日志信息与数据库中一致，显示格式正确。   | 测试通过  |

在系统菜单中点击“系统管理”，系统显示用户列表，点击具体用户的“密码管理”按钮，弹出密码管理窗口，如图 5.7 所示，可以对用户密码进行修改。

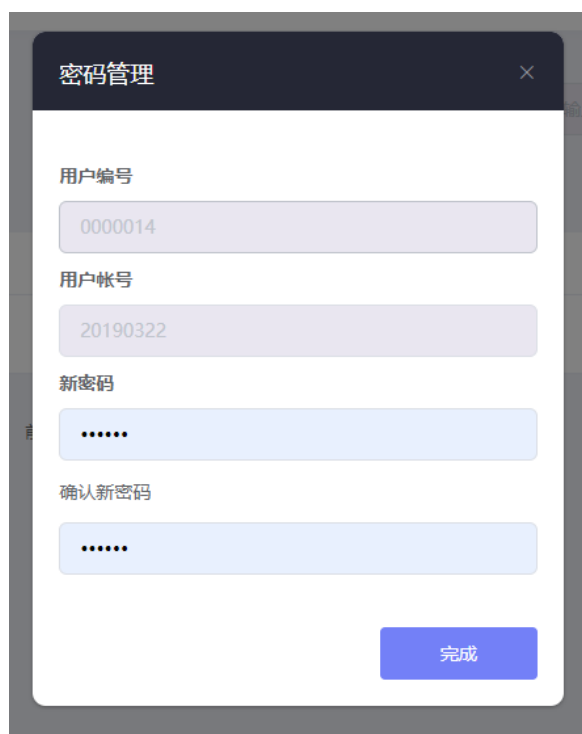


图5.7 密码管理窗口

## 5.4 系统非功能测试

系统的非功能测试主要为性能测试，通过编写测试脚本，在模拟实际使用场景中运行，对系统在正常运行环境和边界条件下的性能表现进行测试，观察并记录系统表现，完成性能检测报告。使用自动化测试工具 LoadRunner 对用户实际使用过程中可能出现的多种正常负载、峰值负载和异常负载情况进行模拟，对系统进行测试，监测记录各项性能指标，比如服务器响应时间、系统 CPU 使用量和吞吐量等，以此来评估系统的性能表现。本系统的性能测试使用 LoadRunner 工具，根据业务流程编辑脚本，使其参数化，设置场景，分析执行 Controller 后的结果，查看概要界面中的 Average 得到系统的平均响应时间。本系统性能测试的具体数值如表 5.9 所示。

在实际的测试工作中，计划使用 LoadRunner 获取系统中各个操作的响应时间，测试计划中规定设置初始执行用户数量为 20 个，逐步递增至 100。具体步骤为：首先在 LoadRunner 工具中选择 http 协议，录制方式为 HTML-based Script，默认浏览器为 IE，对脚本进行参数化编辑，然后将初始执行用户数设定为 20，循环步长为 20，根据业务流程设置启动和停止方式，当本系统的 11 个功能的操作相应时间超过 3 秒或者达到 100 个最大并发用户时停止并进行记录，最后查看测试结果，比如事务的响

应时间，以此分析系统性能指标是否符合要求。

表5.9 系统性能测试结果表

| 功能监测点  | 用户数为<br>20 的响应<br>时间 (s) | 用户数为<br>40 的响应<br>时间 (s) | 用户数为<br>60 的响应<br>时间 (s) | 用户数为<br>80 的响应<br>时间 (s) | 用户数为<br>100 的响应<br>时间 (s) |
|--------|--------------------------|--------------------------|--------------------------|--------------------------|---------------------------|
| 用户信息添加 | 1.232                    | 1.249                    | 1.256                    | 1.271                    | 1.283                     |
| 商家账号新增 | 1.313                    | 1.342                    | 1.375                    | 1.388                    | 1.401                     |
| 活动发布   | 1.435                    | 1.458                    | 1.473                    | 1.486                    | 1.510                     |
| 活动信息查询 | 1.162                    | 1.165                    | 1.171                    | 1.176                    | 1.177                     |
| 商家信息修改 | 1.215                    | 1.224                    | 1.237                    | 1.243                    | 1.252                     |
| 用户信息查询 | 1.127                    | 1.134                    | 1.146                    | 1.149                    | 1.154                     |
| 选项新增   | 1.374                    | 1.380                    | 1.387                    | 1.394                    | 1.398                     |
| 活动查询   | 1.125                    | 1.133                    | 1.135                    | 1.142                    | 1.145                     |
| 数据库备份  | 1.498                    | 1.513                    | 1.525                    | 1.531                    | 1.542                     |
| 密码修改   | 1.261                    | 1.268                    | 1.279                    | 1.282                    | 1.284                     |
| 日志维护   | 1.325                    | 1.329                    | 1.338                    | 1.341                    | 1.353                     |

通过使用 LoadRunner 工具进行测试,得出表 5.9 中所示的系统性能测试结果表,观察各事务响应时间可知,系统支持 100 个用户同时访问登录,各事务响应时间保持在 3 秒之内,系统的并发用户量和响应时间均符合系统需求分析工作中关于性能指标的要求。

测试人员不仅测试了系统的并发用户数和响应时间,还对系统的稳定性和可靠性进行了相关测试。当系统在模拟实际运行环境的条件下 7×24 小时持续不间断地运行时,系统并未出现任何异常,由于测试环境与实际运行环境的各项软硬件配置基本一致,测试结果足以说明系统是十分稳定可靠的,满足实际应用要求。

## 5.5 本章小结

本章对智能匹配系统的测试工作进行了详细介绍,对测试环境进行简要说明,阐述了系统的测试流程,介绍了所采用的测试方法,根据预定的测试计划对系统进行功能测试具体工作和性能测试具体工作后,分析测试结果并得出测试结论,完成了本系统的测试工作。



## 第六章 总结及展望

在如今移动互联网发展的鼎盛时期，微信走入了千家万户，云服务为小型企业的发展带来颇多助益，使用信息技术手段辅助完成活动的策划与实施的智能活动匹配系统已经成为中小型商家发展客户关系的得力工具。在现今计算机飞速发展的潮流下，针对人工统计繁琐、效率低、出错率高、随机率低、活动效果未知和活动数据难以复盘等传统活动匹配方式的不足，本文提出了适合中小型商家进行活动运营的智能匹配系统。不仅满足了商家基本的活动宣传、报名、分组、举办等需求，还提供了活动数据的统计分析、活动表单一键生成和活动评价等功能。最重要的是，本系统为商家提供方便快捷的智能活动表单生成功能，选择合适的信息项、海报图片和匹配规则，可为不同活动生成不同报名表单，本系统一经开发，无需修改代码，表单模板可重复用于多次活动，为不懂计算机的商家用户节省了开发多个活动系统的费用，解决了使用的后顾之忧。本文的目标是为举办活动的中小型商家开发一个基于微信公众平台的多规则可变的智能匹配系统。本文的主要内容包括：

（1）理论技术资料收集工作：本系统的设计与实现工作是在充分参阅技术文档前提下得以开展的，在系统的具体设计实现工作中采用了 Vue.js 前端框架、Java 开发技术以及 MySQL 数据库技术，在论文中对所涉及到的理论和技术进行了介绍与分析；

（2）需求分析工作：对智能匹配系统所涉及到的业务流程进行详细的调研与分析，阐述各个业务内容与工作流程，搭配用例图来辅助说明，对系统正常运行所需满足的功能要求和性能要求进行详细的分析与说明，完成了系统的整体需求文档。

（3）设计与实现工作：在需求文档的指导下，对系统进行详细设计与实现工作，首先对组织结构、业务流程和数据库进行设计，详细阐述了系统的架构设计工作，然后对系统的核心功能模块逐一进行介绍，对系统的技术实现工作进行具体论述，最后对系统的具体设计与实现工作进行简要总结。

（4）测试工作：根据系统实际运行环境对测试环境进行搭建和配置，列举说明了软硬件配置具体参数，编写各个功能的测试用例并执行功能测试，使用 LoadRunner 工具进行性能测试，对测试结果进行分析，考察系统是否实现了预期的功能目标和非功能目标。

在系统的考察研究、分析、设计、开发与测试工作中，基本按照预期的开发计划顺利执行，达到了相应目标，产出需求说明书、详细说明书、开发代码及说明和测试文档等阶段性成果。系统的测试结果表明本系统满足需求说明书中规定的功能需求与非功能需求。在基本完成开发目标后，对系统进一步分析，发现系统仍有改进的空间，未来系统的改进工作主要考虑以下两个方面：

（1）更丰富的匹配规则。活动人员匹配的效果很大程度上依赖于匹配规则的好坏以及算法的智能性，目前系统可选的匹配规则有限，只提供数字大小约束和字符串匹配的匹配规则可选，分组结果并不精准，直接影响到后续活动举办效果、用户评价甚至客户关系，提供更丰富的匹配规则和更精准的匹配算法可以有效地提升用户体验和满意度。

（2）多样化的统计分析。对活动的统计分析可以帮助商家改善产品和服务，及时调整运营投入；对参与者的统计分析可以帮助商家制定合适的营销计划，进一步了解市场和客户群体的需求。统计分析是活动举办的意义所在，系统目前只提供性别比例、年龄分布、地域分布和职业分布等寥寥几种统计分析图表，在后续发展中，提供更多维度的统计分析能够大大提高商家的活动复盘效果。

综上所述，系统未来主要在匹配规则和统计分析规则两方面进行不断探索。提供更丰富的匹配规则、对匹配算法不断进行改进、从更多维度对活动和用户进行统计分析，从而使系统更好地辅助商家改善产品及服务、制定营销计划和进行客户关系管理方面的决策。

## 参考文献

- [1] 朱子昂.基于微信开发平台的社交活动智能匹配系统[J].通讯世界,2018(08):44-45.
- [2] 候春俊.中小企业如何利用微信小程序提升企业品牌和利润[J].海峡科技与产业,2018(02):47-49.
- [3] 傅梓焕,朱晓凤,曾钟渝,马羨明.基于多数可变规则的校内交友平台的设计与实现[J].电脑与电信,2015(11):73-76.
- [4] Hendricks K B, Singhal V R, Stratman J K. The impact of enterprise systems on corporate performance: A study of ERP, SCM, and CRM system implementations[J]. Journal of operations management, 2007, 25(1): 65-82.
- [5] 王胜, 张靖. 基于 Vue.js 高速路政管理系统的设计与实现[J]. 电脑知识与技术, 2017, 13(7X):86-88.
- [6] 旷志光,纪婷婷,吴小丽.基于 Vue.js 的后台单页应用管理系统的研究与实现[J].现代计算机(专业版),2017(30):51-55.
- [7] 张丽,张静,黄宁,仲维一.基于 B/S 结构毕业答辩信息管理系统设计与实现[J].沈阳化工学院学报,2009,23(04):366-369.
- [8] 张凯飞,王泽华,王洪波,甄爱军.基于微信的图书借阅系统的设计与实现[J].电脑知识与技术,2018,14(06):63-64+67.
- [9] 邵慧威,孙宇成,杨楠,孙一笑,张玉军.本科毕业答辩分组系统的设计与实现[J].电脑知识与技术,2018,14(32):76-77.
- [10] 雷爱民,祖兆研,王家文,王冲.一种基于分组和注册的配置管理系统[J].数字通信世界,2017(11):173+176.
- [11] 王爽, 陈俊杰, 肖铮, 等. 应用阿里云搜索服务构建图书馆站内搜索引擎[J]. 数据分析与知识发现, 2013 (6): 85-89.
- [12] 张涛, 王海鹏, 胡正国. 基于 UML 用例图的软件产品线需求建模方法[D]., 2004.
- [13] Warmer J B, Kleppe A G. The object constraint language: Precise modeling with uml (addison-wesley object technology series)[J]. 1998.
- [14] Larman C. Applying UML and patterns: an introduction to object oriented analysis and design and iterative development[M]. Pearson Education India, 2012.
- [15] 孙娉娉, 李新, 史广军. 基于前后端分离的内容管理系统[J]. 科研信息化技术与应用, 2016, 4.
- [16] 徐頔, 朱广华, 贾瑶. 基于 VueJs 的 WEB 前端开发研究[J]. 科技风, 2017 (2017 年 14): 69-69.
- [17] 江庆, 叶浩荣. Vue+ Webpack 框架在银行 App 前端开发的应用[J]. 金融科技时代, 2016,

- 24(11): 15-19.
- [18] 郭杰. 攒蛋比赛自动分组判分系统的设计与实现[J]. 电脑知识与技术, 2016, 12(34): 91-92+114.
- [19] 夏玉敏, 华中. 基于 Ajax 技术的攒蛋比赛自动裁判系统设计与实现[J]. 内江科技, 2012, 33(07): 150-151.
- [20] 李永安, 蒋立兵, 张玉杰, 王磊. 乒乓球分组循环对抗赛组织管理系统软件设计与实现[J]. 安徽体育科技, 2016, 37(05): 28-31.
- [21] 饶国慧. 大学英语课堂随机分组系统[J]. 课程教育研究, 2015(21): 91-92.
- [22] 王立诺. 基于计算机技术的田径运动会编排管理系统分析[J]. 自动化与仪器仪表, 2016(06): 140-141.
- [23] 敖建华. 运动会编排系统中的分类算法设计[J]. 信息技术与信息化, 2014(09): 161-164.
- [24] 穆荣, 刘蒙蒙, 王晓路. 大学生社团活动管理系统的设计[J]. 电子世界, 2017(15): 159-159.
- [25] Myllymaki M, Talvitie T. Display screen with graphical user interface: U.S. Patent Application 29/584,008[P]. 2018-4-24.
- [26] Hung S Y, Hung W H, Tsai C A, et al. Critical factors of hospital adoption on CRM system: Organizational and information system perspectives[J]. Decision support systems, 2010, 48(4): 592-603.
- [27] Rababah K, Mohd H, Ibrahim H. Customer relationship management (CRM) processes from theory to practice: The pre-implementation plan of CRM system[J]. International Journal of e-Education, e-Business, e-Management and e-Learning, 2011, 1(1): 22-27.
- [28] Crockford D. The application/json media type for javascript object notation (json)[R]. 2006.
- [29] Lien C H, Cao Y. Examining WeChat users' motivations, trust, attitudes, and positive word-of-mouth: Evidence from China[J]. Computers in Human Behavior, 2014, 41: 104-111.
- [30] Gan C, Wang W. Uses and gratifications of social media: a comparison of microblog and WeChat[J]. Journal of Systems and Information Technology, 2015, 17(4): 351-363.
- [31] Lucassen J M, Maes S H. Mvc (model-view-controller) based multi-modal authoring tool and development environment: U.S. Patent 6,996,800[P]. 2006-2-7.
- [32] Lucassen J M, Maes S H. MVC (Model-View-Controller) based multi-modal authoring tool and development environment: U.S. Patent 7,900,186[P]. 2011-3-1.
- [33] You E. Vue.js[J]. Diakses dari <https://vuejs.org/>, pada tanggal, 2018, 17.
- [34] Webb P, Syer D, Long J, et al. Spring boot reference guide[J]. Part IV. Spring Boot features, 2013, 24.
- [35] Alex B, Taylor L, Winch R, et al. Spring Security Reference[J]. URL <https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/>. [utoljára megtekintve: 2017. 04. 21.], 2004.



## 致谢

时光如梭，流年似水。转眼间，为期三年的研究生生涯即将结束，回首这段青春奋斗的日子，我的心中充满了感慨。在我的论文即将完成之际，我首先要向我的母校西安电子科技大学致以深深的感谢，在这里我学会了最前沿的科学知识，得到了许许多多的实践锻炼机会，更拿到了敲开未来大门的钥匙，除此之外，母校良好的风气更是潜移默化地教会了我许多人生的道理，让我在懵懂和跌跌撞撞中不断成长，成为一个懂得坚持、积极向上的年轻人。

俗话说，“桃李不言，下自成蹊”，学院的导师们都是如此。严谨的学术作风、幽默风趣的教学方式、勤恳踏实的工作精神，跟随导师们进行学习的日子充实快乐，我有幸能够得到学院导师们的悉心指导，感到非常荣幸。带给我更多感动的是我的指导老师李青山老师，李老师是一位研究经验丰富的优秀教师，他对学术严谨的态度、对前沿科学的钻研精神都深深感染着我，他教会了我如何去发掘问题、思考问题。李老师对学生的关心爱护更是让我深深感动，在我遇到困难的时候，李老师积极地帮助我，在思想上开导我，让我学会坚强、学会如何去面对困难，这其实对我的一生有着更加深远的影响，在此我想对李老师真诚地说一声谢谢。

在研究生的生活中，最离不开的就是伴我成长的小伙伴了。我要感谢我的舍友庞萌和周辣辣，在我刚进入学校的时候，她们的热情和开朗使我感受到温暖和关怀，她们一直陪伴我进步，一起完成了很多有意义的学习和工作，对我的经验提升有了很大的帮助，而且也让我的研究生生活充满了乐趣。除此之外我想感谢我的男朋友，他在我骄傲时鞭策提醒我，在我消极低沉时开导鼓励我，为我的学习和生活提供了许多帮助，陪伴我共同进步。

父母恩，不可忘。在这里，我要郑重地感谢这么多年来含辛茹苦养育我的父母和给我精神力量的外祖父母，是他们扮演了我生活中坚强的后盾，让我可以顺利完成学业，在今后的生活中我一定会好好孝顺他们，回报他们的恩情。

最后，我要向所有关心和帮助过我的老师和同学们致以诚挚的感谢！向我的母校西安电子科技大学致敬，我当牢记校训，厚德，求真，砺学，笃行，将我所学贡献于社会，发挥自己的光和热！



## 作者简介

### 1. 基本情况

张培妍，女，陕西西安人，1994 年 10 月出生，西安电子科技大学计算机学院软件工程专业 2016 级硕士研究生。

### 2. 教育背景

2012.09～2016.07 西安电子科技大学，本科，软件工程

2016.09～2019.07 西安电子科技大学，硕士研究生，软件工程

### 3. 攻读硕士学位期间的研究成果

