****

**毕业论文**

基于J2EE的网上在线调查平台

的设计与实现

**学生姓名**： 薛源 **学号**：142085243

**系 部**： 理学系

**专 业：** 数学与应用数学

**指导教师：**王晋明（讲师） 王建军（教授）

二○一八年六月

|  |
| --- |
| 诚信声明  本人郑重声明：本设计（论文）及其研究工作是本人在指导教师的指导下独立完成的，在完成设计（论文）时所利用的一切资料均已在参考文献中列出。    本人签名： 年 月 日 |

**毕业设计（论文）任务书**

毕业设计（论文）题目： 基于J2EE的网上在线调查平台的设计与实现

系部： 理学系 专业： 数学以应用数学 学号： 142085243

学生： 薛源 指导教师（含职称）： 王晋明（讲师） 王建军（教授）

1．课题意义及目标

问卷调查是社会调查中收集数据资料的一种常用的手段方法。随着互联网科技的飞速发展和人民生活水平的不断提高社会迫切的需要一种覆盖面广、高效、安全的调查方式。通过此课题的设计与实现，使学生熟练掌握所学知识，将理论运用与实践。最终目标是实现网上在线调查平台的所有功能。

1. 主要任务

根据课题进行调研，查阅资料；汇总文献查阅结果确定系统完成方案；完成网上在线调查平台的需求分析，确定系统的功能；完成网上在线调查平台的详细设计以及数据库设计和实施；实现网上在线调查平台的各个功能。

3．主要参考资料

1. 孙卫琴.李洪成.Tomcat与Java Web开发技术详解[M].电子工业出版 社，2003年10月.
2. 李宁.Java Web编程实战宝典[M].清华大学出版社,2014: 2-188.
3. 刘增杰.MySQL 5.7从入门到精通[M].清华大学出版社,2016: 1-256.

4．进度安排

|  |  |  |
| --- | --- | --- |
|  | 设计（论文）各阶段名称 | 起 止 日 期 |
| 1 | 问题定义、可行性研究 | 2017年12月27日~2018年2月01日 |
| 2 | 查阅、学习文献资料并准备开题报告 | 2018年02月02日~2018年3月09日 |
| 3 | 需求分析、系统建模 | 2018年03月10日~2018年3月20日 |
| 4 | 系统设计、数据库设计 | 2018年03月21日~2018年3月31日 |
| 5 | 系统实现、调试运行、编写文档 | 2018年04月01日~2018年5月15日 |
| 6 | 撰写论文 | 2018年05月16日~2018年5月30日 |
| 7 | 准备答辩 | 2018年06月01日~2018年6月18日 |

基于J2EE的网上在线调查平台的设计与实现

摘要：问卷调查时社会调查中收集数据资料最常用和最直接的手段之一。随着互联网科技的飞速发展和人民生活水平的不断提高社会迫切的需要一种覆盖面广、高效、安全的调查方式。因此网上问卷调查以其覆盖面广、信息收集迅速、信息采集更安全更真实、成本更低等特点被广泛应用。

本系统的设计开发模式采取基于WEB的B/S架构模式，系统采用J2EE作为开发语言，系统后台数据库采用MySQL，运用STS开发环境，采用Tomcat服务器为系统服务器。系统最终实现用户的注册、登录、发布问卷、填写问卷、数据统计等功能模块。

关键词：问卷调查， J2EE，B/S架构，MySQL数据库，STS，Tomcat

**Design and implementation of online survey platform based**

**on Java EE**

**Summary**: One of the most common and direct means of collecting data in social surveys during questionnaires. With the rapid development of Internet technology and the continuous improvement of people's living standards, the society urgently needs a broad, efficient and safe way of investigation. Therefore, the online questionnaire survey is widely used for its wide coverage, rapid information collection, more secure information collection, more real and lower cost. The design and development mode of this system adopts B/S model based on Web, the system uses Java-EE as the development language, the system backstage database uses MySQL, uses the STS development environment, uses the Tomcat server as the system server. The system finally realizes the user's registration, registers, publishes the questionnaire, fills in the questionnaire, the data statistics and so on function module.

**Keywords**: **questionnaire survey**,**J2EE**,**B/S architecture**, **MySQL database**, **Spring Tool Suite**, **Apache Tomcat.**

目录

[1 绪论 1](#_Toc515825676)

[1.1 前言 1](#_Toc515825677)

[1.2 研究目的及意义 1](#_Toc515825678)

[1.3 国内外研究进展 2](#_Toc515825679)

[1.4 开发工具 2](#_Toc515825680)

[1.5 开发技术 3](#_Toc515825681)

[2 系统设计 5](#_Toc515825682)

[2.1 需求分析 5](#_Toc515825683)

[2.1.1 可行性分析 5](#_Toc515825684)

[2.1.2 系统功能需求描述 5](#_Toc515825685)

[2.2 系统整体设计 6](#_Toc515825686)

[2.2.1 系统功能分析 6](#_Toc515825687)

[2.2.2 用户功能分析 9](#_Toc515825688)

[2.3 系统数据库设计 10](#_Toc515825689)

[2.4 系统模块设计 14](#_Toc515825690)

[2.4.1 系统管理模块 14](#_Toc515825691)

[2.4.2 问卷管理模块 21](#_Toc515825692)

[2.4.3 答卷管理模块 25](#_Toc515825693)

[3 系统测试 27](#_Toc515825694)

[3.1 单元测试 27](#_Toc515825695)

[3.2 系统整体测试 27](#_Toc515825696)

[3.2.1 部分模块测试用例 27](#_Toc515825697)

[4 成果展示 29](#_Toc515825698)

[4.1 部分功能成果展示 29](#_Toc515825699)

[5 结束语 31](#_Toc515825700)

[5.1 设计总结 31](#_Toc515825701)

[5.2 心得体会 31](#_Toc515825702)

[6 致谢 32](#_Toc515825703)

[7 参考文献 43](#_Toc515825704)

[8 附录 33](#_Toc515825705)

[8.1 附录A登录控制器 33](#_Toc515825706)

[8.2 附录B题库控制器 35](#_Toc515825707)

[8.3 附录C题目数据实现 41](#_Toc515825708)

# 绪论

## 前言

问卷调查时社会调查中收集数据资料最常用和最直接的手段方法之一，具有适用群体广、快捷的获取数据、研究成本底等特点。通常是以问题的形式记录调查内容，通过问卷调查表的填写的答案的收集整理用于分析被调查人对于某个特定问题的态度和观念[7]。

随着互联网科技的飞速发展和人民生活水平的不断提高，以及人民隐私保护意识的提升，传统的纸质问卷调查由于传播难、统计难、不利于日常管理、涉及个人信息尤其是对与某些敏感问题的调查往往会暴露被调查人的隐私信息，而越来越不被接受和使用。社会迫切的需要一种覆盖面广、高效、安全的调查方式。因此网上问卷调查以其覆盖面广、信息收集迅速、信息采集更安全更真实、成本更低等特点被广泛应用。

本文介绍的网上在线调查平台是专门为适应学校用户的需求而设计开发的，系统采用动态网页设计技术，构建了一个基于动态框架的调查平台。系统包含了问卷管理、答卷管理、系统管理三大功能模块，问卷管理模块包括出卷和发布、修改问卷、删除问卷、数据统计四个子模块；答卷管理包括查询问卷和参与答卷两个子模块；系统管理包括用户管理和数据管理两个子模块。

本系统与传统的问卷调查相比，具有效率高、成本低的优势；由于填写问卷的用户不需要注册登录就可以参加填写，因此本系统还具有获取信息的真实性相比传统的问卷调查更真实的优势。

## 研究目的及意义

问卷调查时社会调查中收集数据资料最常用和最直接的手段方法之一。随着互联网科技的飞速发展和人民生活水平的不断提高社会迫切的需要一种覆盖面广、高效、安全的调查方式。因此网上问卷调查以其覆盖面广、信息收集迅速、信息采集更安全更真实、成本更低等特点被广泛应用。本系统最终实现用户的注册、登录、发布问卷、填写问卷、数据统计等功能模块。

通过此课题的设计与实现，使学生熟练掌握所学知识，将理论运用与实践。同时，提升学生自主思考、独立分析、解决问题的能力，并使学生熟悉并投入现如今的大数据时代，为后续工作打下良好基础。

## 国内外研究进展

目前国内外有许多同类型的网站，如国内比较好的有：问道网，是目前国内专业的调查问卷服务平台[18]；数字100调查网[18]，数字100市场调研公司提供权威市场调研、在线调研、市场调查、在线调查专业研究报告；我们做[18]，我们做的口号是做企业级的调研、心理测评和投票系统[18]；问卷星[18]，是一个专业的[在线问卷调查](http://baike.sogou.com/v10205973.htm)、测评、投票平[18]；麦客平台，是一款在线表单制作工具，同时也是强大的客户信息处理和关系管理系统，应用动态网页技术，用户可以随意制定自己想要的问卷。国外比较好的调查平台有<http://questionform.com> 和<http://www.wufoo.com>。

但是这些平台对于学生使用来说，或多或少都有些不方便。如国内的大多平台都是某些公司的子系统或某些系统的子模块，功能太多，找起来不方便；并且，很多平台都是收费的。因此迫切需要一款免费对学生开放的平台。

## 开发工具

本系统设计主要采用STS， Tomcat服务器，MySQL数据库等开发工具。下面对其进行简要说明：

STS

STS（Spring Tool Suite）就是一款被包装过的Eclipse[19]，是一个基于Eclipse开发环境中的用于开发Spring应用程序的工具[19]。提供了开箱即用的环境用于实现，调试和部署你的Spring应用，包括为关键的服务器和云计算，Git，Maven，AspectJ,和最新的Eclipse版本提供整合支持[19]。

Tomcat服务器

Tomcat 服务器是一个免费的开放源代码的Web 应用服务器，属于轻量级应用服务器，在中小型系统和并发访问用户不是很多的场合下被普遍使用，是开发和调试JSP 程序的首选[1]。

MySQL

MySQL是一种关系数据库管理系统，关系数据库将数据保存在不同的表中，而不是将所有数据放在一个大仓库内，这样就增加了速度并提高了灵活性[3]。MySQL使用了多种[编译器](https://baike.baidu.com/item/%E7%BC%96%E8%AF%91%E5%99%A8" \t "_blank)进行测试，保证了[源代码](https://baike.baidu.com/item/%E6%BA%90%E4%BB%A3%E7%A0%81)的可移植性[3]。

MySQL所运用的 SQL 语言是用于访问[数据库](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%BA%93" \t "_blank)的最常用的数据库标准化语言。因为其体积小、速度快、总体成本低，最重要的是其[开放源码](https://baike.baidu.com/item/%E5%BC%80%E6%94%BE%E6%BA%90%E7%A0%81)的特点，使得普遍的中小型网站的开发所使用的数据库都选择 MySQL。

## 开发技术

本系统设计主要运用J2EE、SSM、JavaScrip、JQuery等技术。下面对主要技术进行简要介绍：

J2EE

J2EE的核心是一组技术规范与指南，其中所包扩的各类组件、服务架构及技术层次，均有共同的标准及规格，让各种依循J2EE架构的不同平台之间，存在良好的兼容性[1]

SSM

SSM（Spring+SpringMVC+MyBatis）框架集由Spring、SpringMVC、MyBatis三个开源框架整合而成其中spring是一个轻量级的控制反转（IoC）和面向切面（AOP）的容器框架[5]

SpringMVC分离了控制器、模型对象、分派器以及处理程序对象的角色，这种分离让它们更容易进行定制[5]。

Spring MVC 是目前相对于Struts 2 更好的控制层设计框架，它本身的设计理念是针对方法的编程，即这一理念支持Restful设计理念，而且本身也是在MVC模式的基础上创建出来的，这是符合现代软件开发模式的选择。

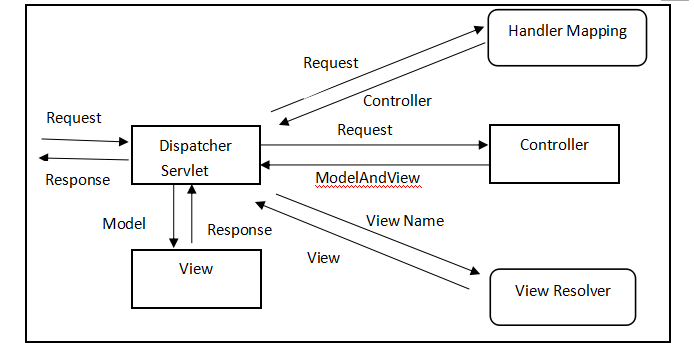


图1.4.1 Spring MVC原理流程图

MyBatis是一个支持普通SQL查询、存储过程和高级映射的优秀持久层框架[5]。

Mybatis，是一个和数据库交流的使者，和其他的数据库管理框架相比（例如：Hibernate、JPA等等），Mybatis相对简单，并且也支持动态SQL。Spring MVC通过与MyBatis的结合，提高了网站的可发展性。

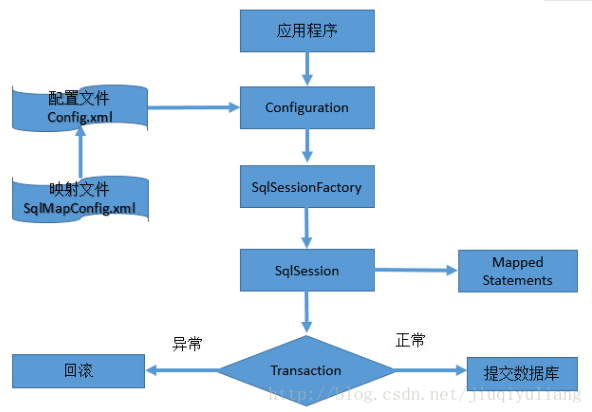


图1.4.2 Mybatis原理流程图

J2EE和SSM都具有简单易学、整合性良好、具有强大的延展性、开源且跨平台等优势。因此本系统选择其作为技术支持。

# 系统设计

## 需求分析

### 可行性分析

软件开发前对其进行可行性分析是很有必要的。可行性分析能确定该项目是否值得开发，开发周期大概需要多久，有没有能力开发等前提条件。

1. 技术可行性

本系统主要使用的开发工具是STS和MySQL数据库，开发语言是Java。Java是面向对象开发语言，具有简单易学的特点。大三时期就已经系统的学习了Java，问卷调查系统总体上开发难度并不大，数据库设计和权限的设置，是本系统设计的难点。而大学期间学习过软件工程课程，也有过开发经历。因此，开发本系统在技术上是完全没问题的。

1. 经济可行性

选择的开发工具和开发语言以及服务器都是免费且开源的软件。并且，开发的是成本较低的基于WEB的B/S模式的系统，而不是成本较高的C/S模式软件。所以经济也完全没有问题。

1. 操作可行性

本系统的设计方案遵从能选的绝不填的方案，操作简单，对操作员几乎没有技术要求，只要会用电脑就会用本系统。而且，本系统讲究操作简单、界面简洁清晰，可视性良好。因此，操作也没什么问题。

### 系统功能需求描述

系统管理员所使用的功能：

用户管理：管理员批量添加问卷会员帐号（问卷发布者）；对于违规用户，管理员可视其违规严重程度进行禁用或删除此用户。

会员用户（问卷发布者）所使用的功能：

用户管理：会员用户可自主注册登录本系统；可修改自己的信息；可自主注销自己的账号。

问卷管理：会员用户登录系统后，可发布、修改、删除自己的问卷；可查看自己问卷的统计结果。

普通用户（任意访问该系统且未登录的用户）所使用的功能：

答卷模块：任意未通过登录访问本系统的用户均可参与问卷答卷。

## 系统整体设计

### 系统功能分析

系统按功能模块分为问卷管理、答卷管理、系统管理三大功能模块，问卷管理包括出卷、修改问卷、查询结果、删除问卷、数据管理五个子模块；答卷管理包括查询问卷和答卷两个子模块；系统管理主要包含用户管理。

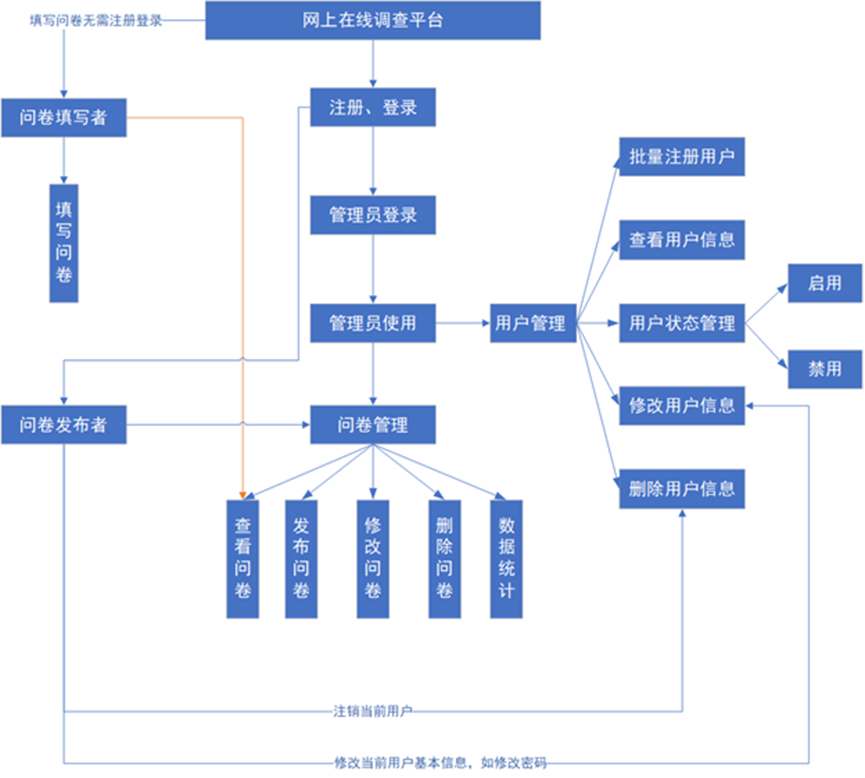


图2.2.1系统功能设计结构图

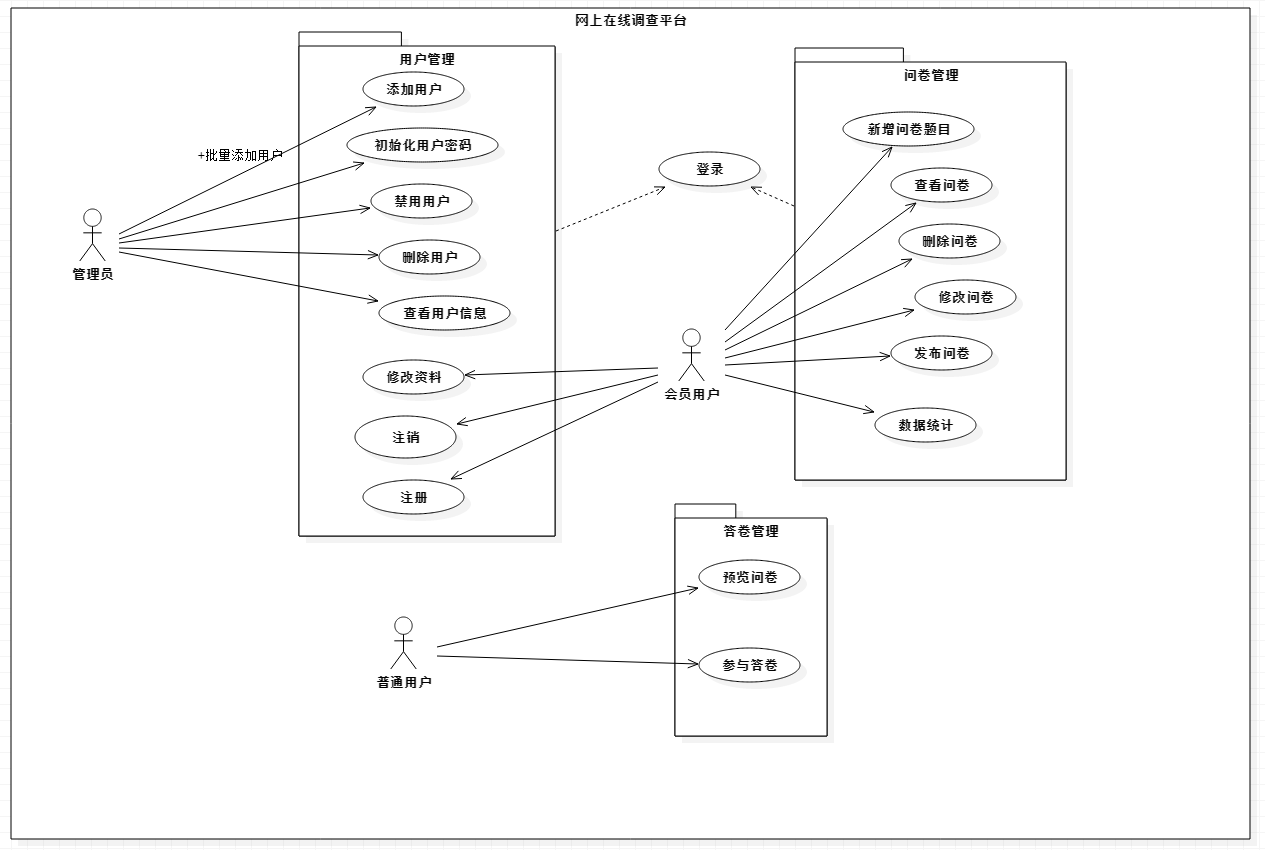
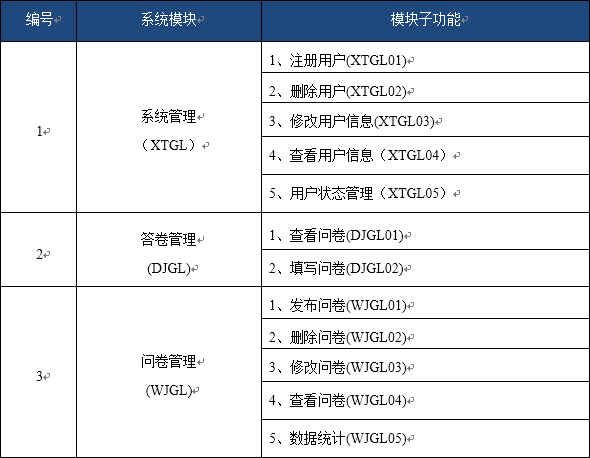


图2.2.2 系统整体功能用例图

表2.2.1 网上调查平台系统功能分析表



### 用户功能分析

管理员对应功能模块用例图：

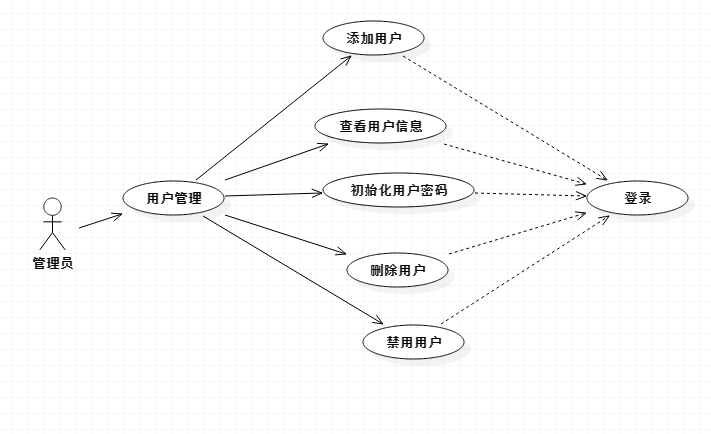


图2.2.3 管理员用例图

会员用户（问卷发布者）对应功能模块用例图：

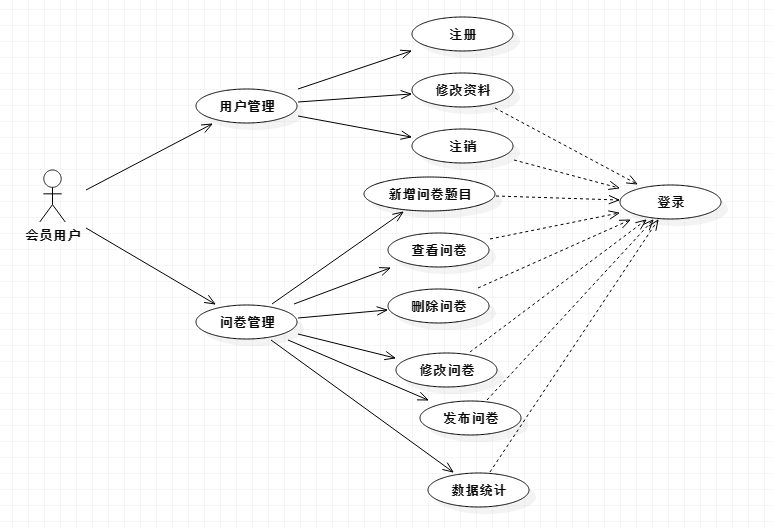


图2.2.4 会员用户用例图

普通用户（任意未登录访问本系统的用户）对应功能模块用例图：

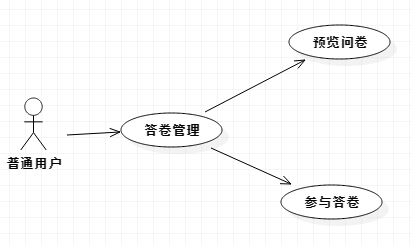


图2.2.5 普通用户用例图

## 系统数据库设计

本系统的设计过程，采用MySQL数据库。旨在设计本系统中各项功能和非功能性需求实现时所需要的数据库，并以此数据库为根据实现具体的数据存储。

系统使用的数据库环境为MySQL5.0。

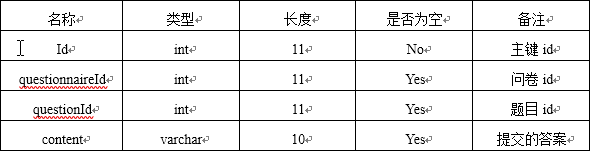


图2.3.1 数据库物理模型

数据库表和字段说明：

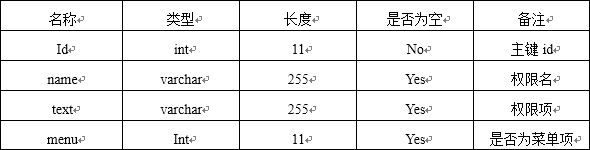
1. Answer答案表，存储用户参与调查时所填写的答案。包含主键id、问卷id、题目id、提交的答案。

表2.3.1 问卷答案表



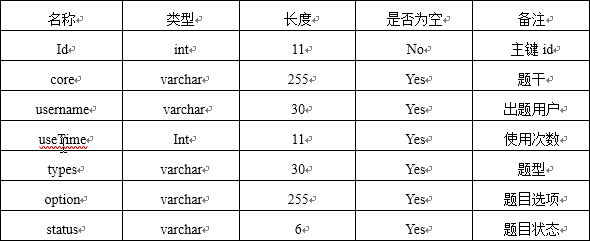
1. Permission权限表，存储用户权限，用于控制用户使用的功能。包含主键id、权限名称、权限项、是否为菜单项

表2.3.2 权限表



1. Question 题目表，存储问卷调查表中的题目。包含主键id、题干、出题用户、使用次数、题型、题目选项、题目状态。

表2.3.3 问卷题目表



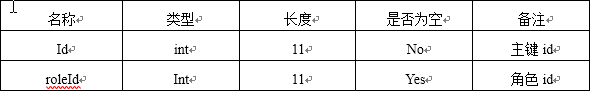
1. Role角色表，存储用户角色。包含主键id、角色名称。

表2.3.4 角色表



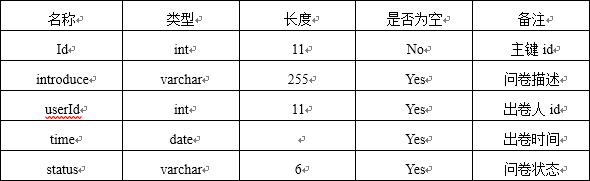
1. role\_permission 角色权限关系表，存储角色拥有的权限。包含主键id、角色id、权限id。

表2.3.5 角色权限关系表



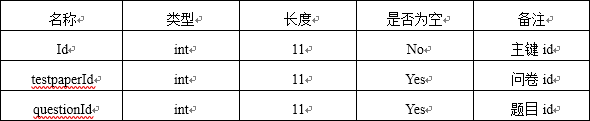
1. testpaper 问卷表，存储问卷。包含主键id、问卷描述、出卷人、出卷时间、状态。

表2.3.6 问卷表



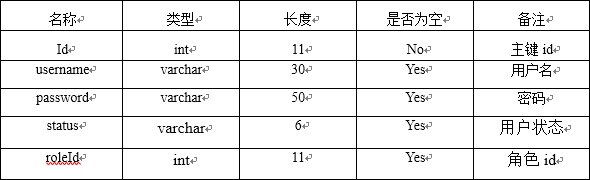
1. testpaper\_question 问卷和题目关系表，存储问卷和题目的对应关系。包含主键id、问卷id、题目id。

表2.3.7 问卷和题目关系表



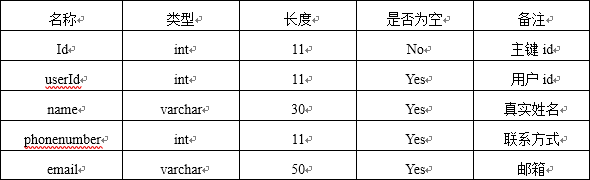
1. User 用户表，存储用户帐号信息。包含主键id、用户名、密码、状态、角色。

表2.3.8用户表



1. Userinfo 用户信息表，存储用户基本信息。包含主键id、用户id、用户真实姓名、联系方式、邮箱。

表2.3.9 用户信息表



## 系统模块设计

### 系统管理模块

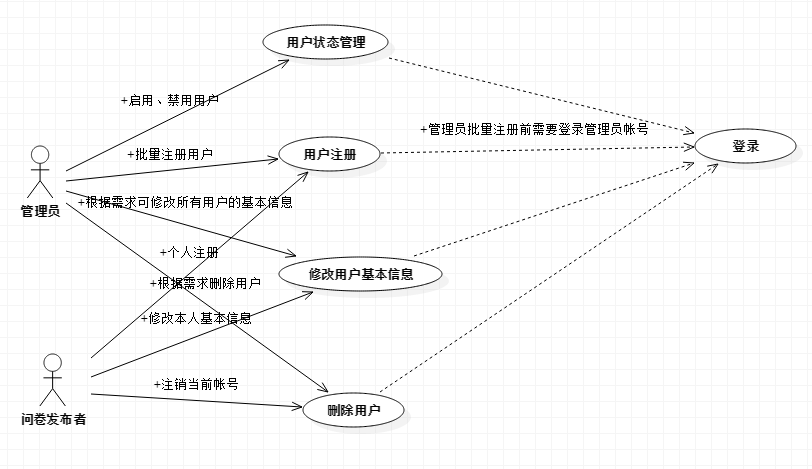


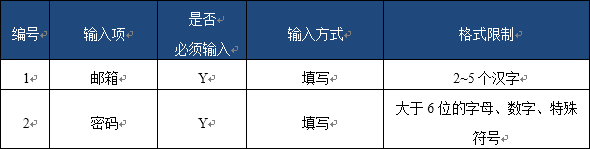
图2.4.1 系统管理模块功能用例图

##### 系统管理模块功能说明

###### 注册用户

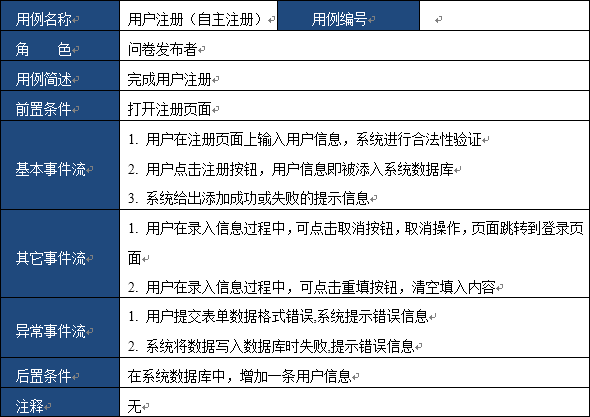
用户可自主注册帐号，管理员可批量注册用户帐号。注册信息包括：邮箱、密码。

表2.4.1 注册用户信息输入说明表



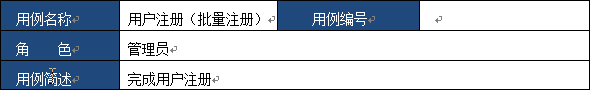
用户注册用例描述（自主注册）

表2.4.2 用户注册用例描述（自主注册）

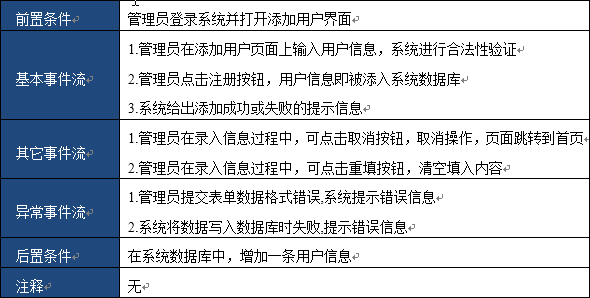


用户注册用例描述（批量注册）

表2.4.3 用户注册用例描述（批量注册）



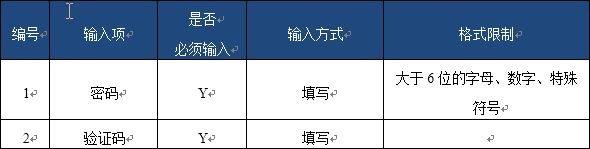
续表2.4.3



###### 删除用户

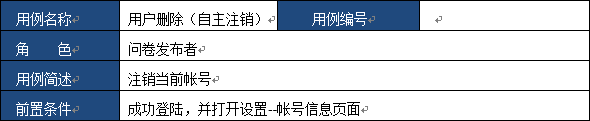
用户可自行注销自己的账号；管理员根据需求可将失效帐号信息（已被注销的帐号）或严重违规的用户进行删除操作。

表2.4.4 删除用户信息输入说明表

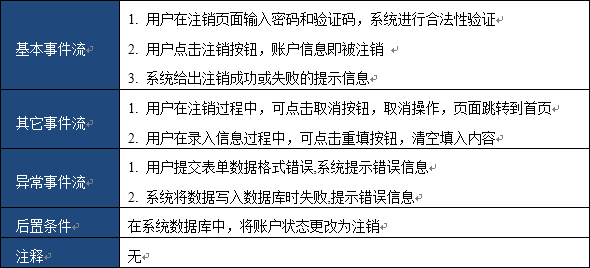


删除用户用例说明（用户自主注销）

表2.4.5 用户删除用例描述（自主注销）

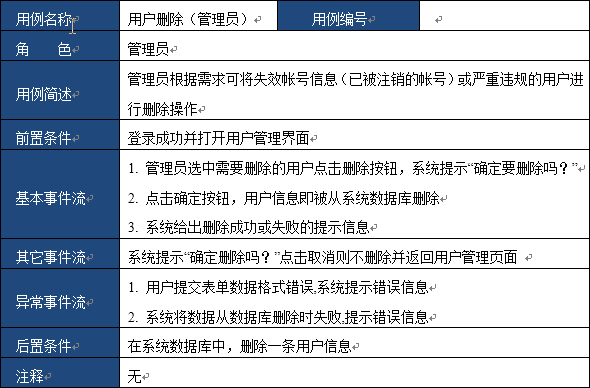


续表2.4.5



删除用户用例说明（管理员删除其他用户）

表2.4.6 用户删除用例描述（管理员操作）



###### 修改用户信息

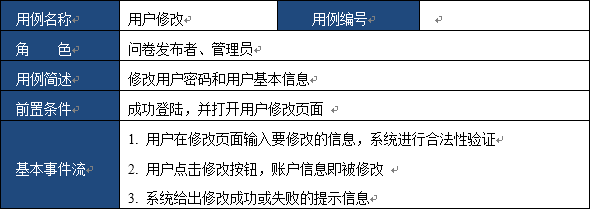
用户可修个帐号信息。普通用户只可修改自己的帐号基本信息和和更改密码；管理员可修改所有用户的密码。

表2.4.7 修改用户信息输入说明表

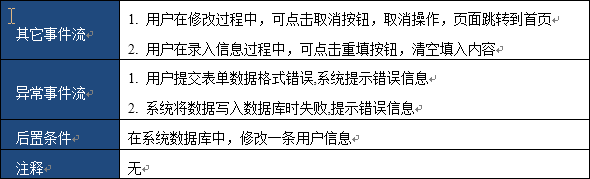


修改用户信息用例描述

表2.4.8 用户修改用例描述



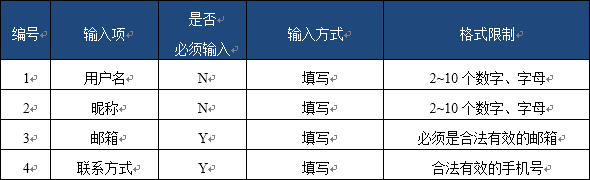
续表2.4.8



###### 查询用户

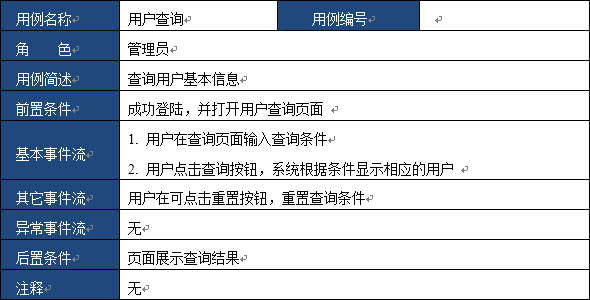
管理员可查看已注册用户的基本信息。

表2.4.9 查询用户输入说明表



用户查询用例描述

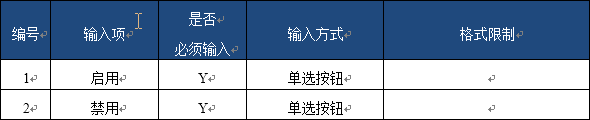
表2.4.10用户查询用例描述



###### 用户状态管理

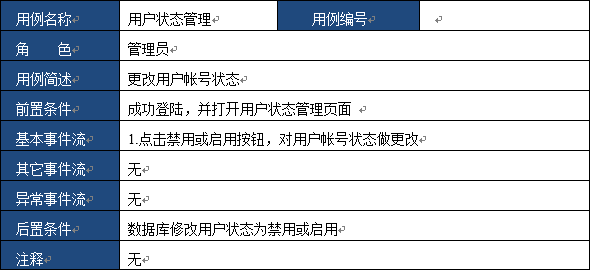
管理员根据普通用户的使用情况，可对普通用户的账号状态做禁用或启用的更改，限制其登录。

表2.4.11 用户状态管理输入说明



用户状态管理用例描述

表2.4.12 用户状态管理用例描述



### 问卷管理模块

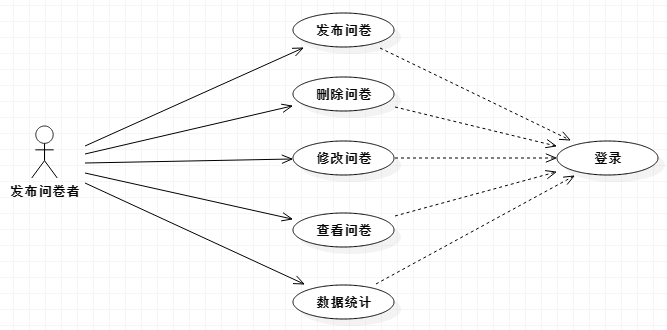


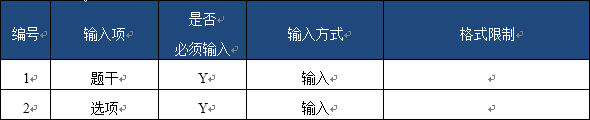
图2.4.2 问卷管理模块功能用例图

##### 问卷管理模块功能说明

###### 发布问卷

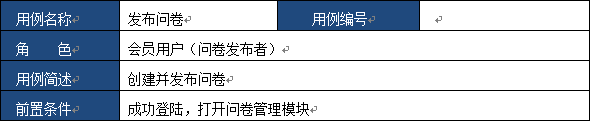
问卷发布者登录系统后，通过问卷管理模块发布问卷。主要实现创建问卷和发布问卷功能。

表2.4.13 发布问卷输入说明

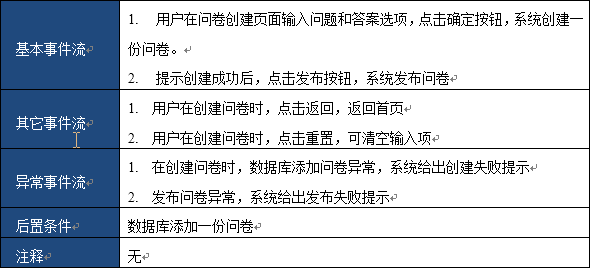


发布问卷用例描述

表2.4.14 发布问卷用例描述



续表2.4.14

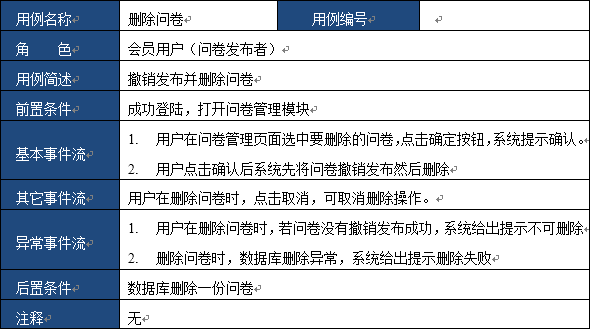


###### 删除问卷

用户登录系统后，可对自己创建的问卷进行删除操纵。

删除问卷用例描述

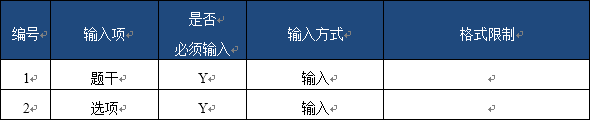
表2.4.15 删除问卷用例描述



###### 修改问卷

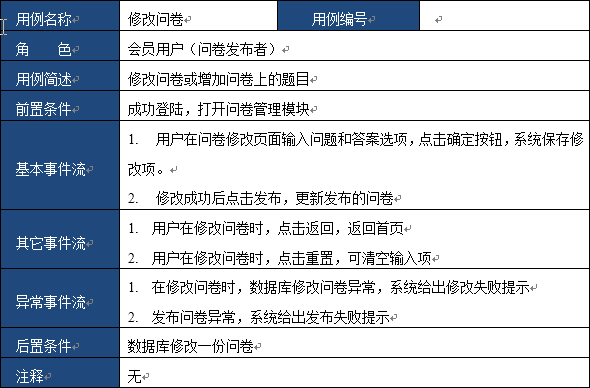
问卷发布者登录系统后，通过问卷管理模块修改问卷。主要实现问卷不合理处的更改和增加问卷上的题目。

表2.4.16 修改问卷输入说明



修改问卷用例描述

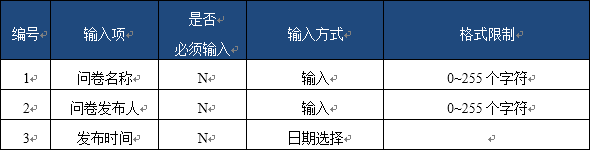
表2.4.17 修改问卷用例描述



###### 查询问卷

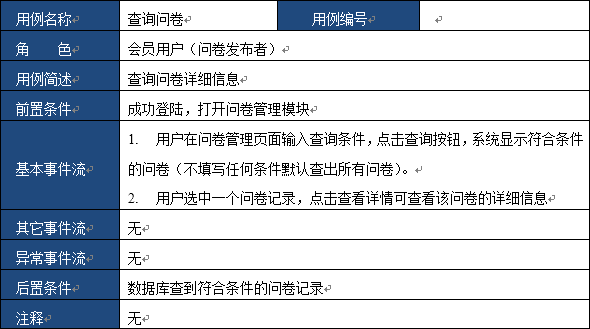
问卷发布者登录系统后，通过问卷管理模块查询问卷。主要实现问卷详情查询功能。

表2.4.18 查询问卷输入说明



查询问卷用例描述

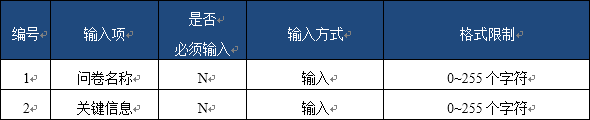
表2.4.19 查询问卷用例描述



###### 数据统计

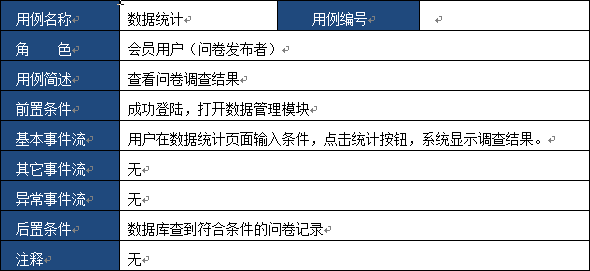
根据重点问题的回答情况得到发布者所需要的结果。用户可查看调查结果。

表2.4.20 数据统计输入说明



数据统计用例描述

表2.4.21 数据统计用例描述



### 答卷管理模块

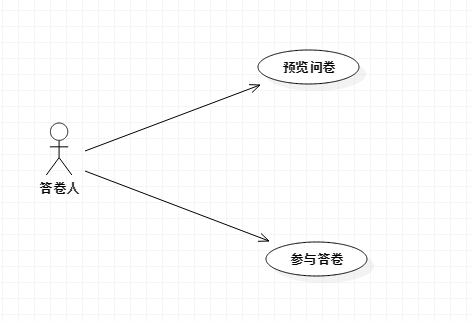


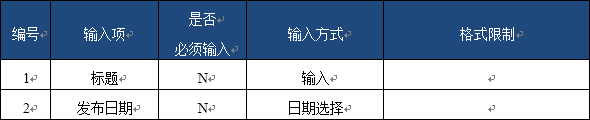
图2.4.3 答卷管理模块功能用例图

##### 答卷管理模块功能说明

###### 预览问卷

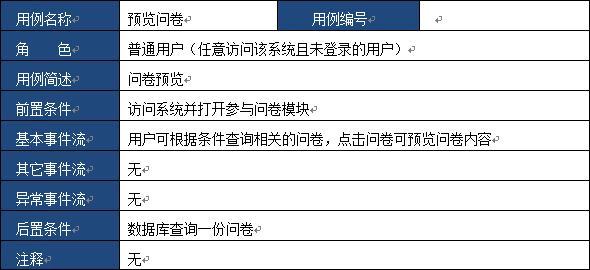
用户可以查看当前已发布的问卷，也可通过条件查询到特定的问卷预览该问卷。

表2.4.22 发布问卷输入说明



预览问卷用例描述

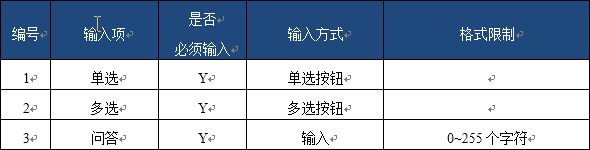
表2.4.23 预览问卷用例描述



###### 参与答卷

主要实现参与问卷调查的答卷操作。

表2.4.24 参与答卷输入说明

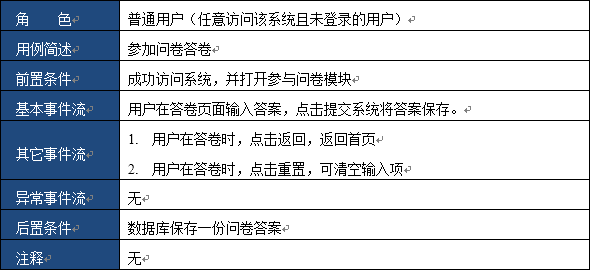


参与答卷用例描述

表2.4.24 参与答卷用例描述



续表2.4.24



# 系统测试

## 单元测试

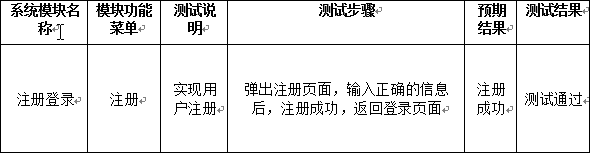
开发完每一个功能模块后，运用Junit插件进行单元测试，并测试通过。保证后台代码逻辑功能上没有问题。

## 系统整体测试

系统开发完成后，进行了整体的测试，保证每一个工能都可以正常使用。

### 部分模块测试用例

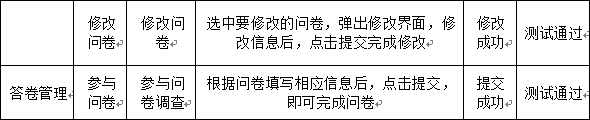
表3.2.1 部分功能模块测试用例表



续表3.2.1



续表3.2.1



# 成果展示

## 部分功能成果展示

问卷管理功能示意图：



图4.1.1 问卷管理模块示意图

添加题目功能示意图：

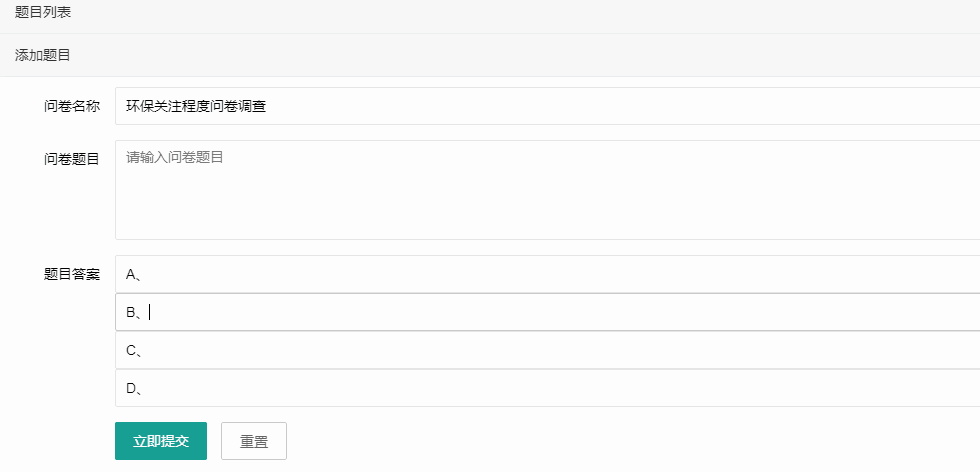


图4.1.2 添加题目示意图

添加问卷功能示意图：

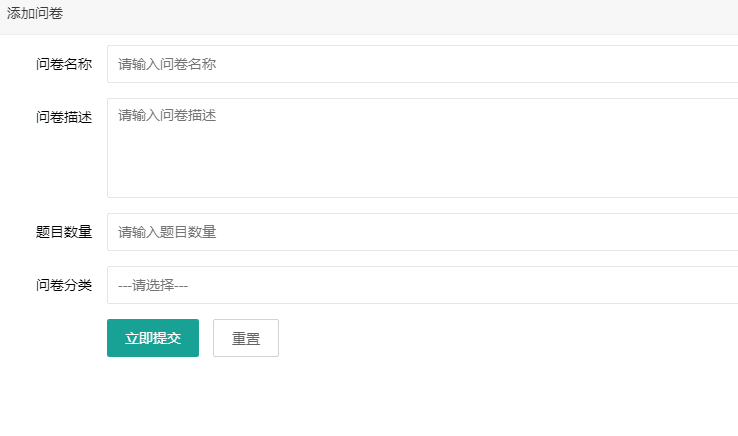


图4.1.3 添加问卷示意图

题库示意图：



图4.1.4 题库示意图

# 结束语

## 设计总结

对于此次基于J2EE的网上在线调查平台的设计与实现的完成，我严格按照软件工程导论的思想，进行设计开发，系统经过前期调研和完成系统需求分析，确定系统所需要完成的基本功能；通过数据库设计确定和建立系统数据库；经过总体设计、详细设计之后进行开发和测试，网站后端管理系统基本达到预期目标，前端答题系统略有缺陷，还有待继续完善。同时系统的稳定性、完整性、可拓展性、实用性、安全性均达到了一定标准。因此后续有新的功能需求可在此基础上添加。系统后台经过测试，运行良好，已达期望。

目前的调查平台的开发，其实还存在着大量的不足之处，加之实习期边工作边做设计与开发，时间上的冲突，使得系统部分功能没有按预期的想法实现，而退而求其次的用另外的方式去实现。例如：创建问卷模块，本想做成动态添加网页的形式，而由于技术和时间的关系没有实现，进而做成了静态的。另，参与问卷调查，本想做成不登录参与的，而由于权限和答题的统计没有做好，进而做成了登录之后参与答题。还有一些能提高网站效率和实用性的功能，由于技术不过关和时间原因，均没来得及实现。而且随着访问人数的增加而暴露出的一些问题，如：多线程并发的问题，都没来的及解决。这些都是本次系统开发所遗留的遗憾。虽然毕业设计的时间已经结束了，而我的系统并没有按照预期完成。但是我之后还是可以继续开发的。这也不代表我会放弃这个系统，我会后续继续完善，将其遗留的问题都要解决，争取做出一个符合期望的系统。

## 心得体会

此次论文的完成宣告着大学生活的即将结束，因此我十分珍惜和回味这一过程，尽管过程非常困难，我也觉得很享受。回顾已经过去的四年大学生活，从懵懂的大一新生到现在成熟沧桑的自己，我有过认真也有过浮躁，过成就也有过失落，有过熬夜学习、有过泡图书馆，也有过逃课打游戏，有过挂科等等这些经历。我成熟稳重了很多，也渐渐成长了。我很开心在即将离开大学的时候，运用自己所学，并且是自己爱好的知识完成这次的设计，我很欣慰自己没有沉沦在游戏世界，没有自甘堕落，始终坚持自己所好，并将其变成自己的职业，让自己的大学生活没有留下遗憾。

通过这次系统的设计与开发，我更加熟练的掌握了所学知识，也学会了怎样系统的设计一个系统。更重要的是将所学知识运用于实践，让我收获颇多。这使得我在工作中，熟练运用所学知识是非常大的帮助。

本次系统的开发过程中也有过很多的困难，也暴露了很多问题。比如技术上的不成熟导致很多东西都得现学，虽然克服了，但也浪费了很多时间；其次是时间规划和设计是的考虑欠佳使得系统很多功能开发出来没有预想的效果，这是很不好的事情。通过这些问题，也暴露了我在设计过程中的不严谨，这个结果虽然给了我不小的打击，但对于以后的工作确是一件好事，早暴露出问题早解决，会使我以后工作中不会再出现类似的问题。

总体来说，这次的设计还是成功的，给我的收获也很大。并且由于是自主开发，让我更加认识到了自学的重要性和会自学的重要性。也让我明白了，只要肯学习、会学习，就没有什么事是能难道你。

# 参考文献

[1]孙卫琴.李洪成.Tomcat与Java Web开发技术详解[M].电子工业出版社，2003.10.

[2]李宁.Java Web编程实战宝典[M].清华大学出版社,2014.

[3]刘增杰.MySQL 5.7从入门到精通[M].清华大学出版社,2016.

[4]张孝祥.徐明华.JAVA基础与案例开发详解[M].清华大学出版社,2009.

[5]康牧.JSP动态网站开发实用教程[M].清华大学出版社,2009.

[6]H.M.Deitel,P.Jdeitel著.Java程序设计教程(第五版)[M].清华大学出版社.2004.

[7] 高欣.张建莉.茹小光.刘环.李明.《网络安全技术与应用》[J]2016年.第11期.73-73.

[8]杨学瑜,高立军.软件开发过程与项目管理[M].电子工业出版社.2008.

[9]李书振. MySQL数据库的安全机制[J].计算机应用.2002.

[10]张海潘.牟永敏.软件工程导论（第六版）[M].清华大学出版社.2013.

[11]薛鹤娟.基于WEB的大学选课系统分析.希望月报(上半月)[J].2007年11期.

[12]江孝宜，胡海璐，米海生.信息系统开发实例精粹[M].电子工业出版社.2006.

[13]杜娟.信息系统分析与设计[M].北京:清华大学出版社.2008.

[14] 王趾成.软件工程(第二版)[M].大连理工大学出版社.2008.

[15] 美)惠腾(Jeffrey L. Whitten),美)本特利(Lonnie D. Bentley)系统分析与设计方法[M].机械工业出版社.2007.

[15]刘腾红等.信息系统分析与设计[M].科学出版社.2006.

[16]孙晓悦.精通Java、 JDK、 数据库系统开发、Web开发[M].人民邮电出版社.2007年2月.

[17] 向传杰.Java编程案例教程[M].电子工业出版社.2004年5月.

[18] 几个专业的问卷调查网站的评测[EB/OL].阿里云资讯网.2014.

https://www.aliyun.com/zixun/content/2\_6\_371421.html.

[19] springboot搭建[EB/OL].阿里云.云栖社区.2018年3月.

# 致谢

本设计在王晋明老师和王建军老师的悉心指导和严格要求下已完成，从课题选取、方案论证到具体设计和调试，无不凝聚着老师的心血和汗水，在四年的本科学习和生活期间，也始终感受着导师的精心指导和无私的关怀，我受益匪浅。在此向两位老师表示深深的感谢和崇高的敬意。

不积跬步何以至千里，本设计能够顺利的完成，也归功于各位任课老师的认真负责，使我能够很好的掌握和运用专业知识，并在设计中得以体现。正是有了他们的悉心帮忙和支持，才使我的毕业论文工作顺利完成，在此向各位指导过我的老师表示由衷的谢意。感谢他们四年来的辛勤栽培。

老师的悉心指导固然重要，同学间的互帮互助也不容忽视，正是有了同学间的互相帮助和互相交流，才使得我论文工作过程总少走很多弯路。在此向小二班的全体同学表示感谢，感谢他们四年来的陪伴和关照。

# 附录

## 附录A登录控制器

/\*\*

\* 登录控制器类

\* @author SourceHsueh

\*

\*/

@Controller

public class LoginController {

private Logger log = Logger.getLogger(this.getClass());

/\*\*

\* 用户管理业务类

\*/

@Resource(name = "userService")

private IUserService userService;

/\*\*

\* 跳转到注册页面

\*

\* @return

\*/

@RequestMapping(value = "/register", method = RequestMethod.GET)

public String register() {

return "register";

}

@RequestMapping(value = "/register", method = RequestMethod.POST)

public String register(HttpServletRequest request,String username, String roleId, String status, String password) {

// 处理密码加密

String sha1password = new SHA1().getString(password.getBytes());

User user = new User();

Role role = new Role();

role.setId(Integer.parseInt(roleId));

user.setRole(role);

user.setStatus(status);;

user.setUsername(username);

user.setPassword(sha1password);

boolean result = userService.addUser(user);

if (result) {

request.getSession().setAttribute("message", "(注册成功！请登录)");

return "redirect:login";

} else {

request.getSession().setAttribute("message", "注册失败！");

return "redirect:message";

}

}

/\*\*

\* 去登录页面

\*

\* @param school

\* 去增加页面初始对象

\* @return

\*/

@RequestMapping(value = "/login", method = RequestMethod.GET)

public String login() {

return "login";

}

/\*\*

\* 处理登录

\*

\* @param username

\* 用户名

\* @param password

\* 用户输入的密码

\* @param isSavePassword

\* 是否保存密码（1是，0否）

\* @param request

\* @param session

\* @param response

\* @return

\*/

@RequestMapping(value = "/login", method = RequestMethod.POST)

public String login(String username, String password, String isSavePassword, HttpServletRequest request,

HttpSession session, HttpServletResponse response) {

String result = "login";

if (username != null && username != "" && password != null && password != "") {

log.debug("请求登录的用户是：" + username);

String sha1password = new SHA1().getString(password.getBytes());

User returnUser = userService.loginService(username, sha1password);

log.info("返回的用户是：" + returnUser);

// 查到用户切用户状态为可用

if (returnUser != null && "可用".equals(returnUser.getStatus())) {

result = "redirect:index";

session.setAttribute(Constants.SESSION\_USER, returnUser);

// 创建cookie对象,添加到response

if (isSavePassword != null && isSavePassword.equals("1")) {

Cookie usernameCookie = new Cookie("username", username);

Cookie passwordCookie = new Cookie("password", password);

// 设置cookie的有效时间

usernameCookie.setMaxAge(604800);

passwordCookie.setMaxAge(604800);

response.addCookie(usernameCookie);

response.addCookie(passwordCookie);

}

} else if (returnUser != null && !"可用".equals(returnUser.getStatus())) {

request.setAttribute("message", "(该用户已被禁用！)");

} else {

request.setAttribute("message", "(用户名或密码错误！)");

}

}

return result;

}

@RequestMapping(value = "/logout", method = RequestMethod.GET)

public String logout(HttpServletRequest request) {

request.getSession().invalidate();

return "redirect:login";

}

public IUserService getUserService() {

return userService;

}

public void setUserService(IUserService userService) {

this.userService = userService;

}

}

## 附录B题库控制器

/\*\*

\* 题库控制器类

\* @author SourceHsueh

\*

\*/

@Controller

@RequestMapping("/question")

public class QuestionController {

/\*\*

\* 题库管理业务类

\*/

@Resource(name="questionService")

private IQuestionService questionService;

/\*\*

\* 答案管理业务类

\*/

@Resource(name="answerService")

private IAnswerService answerService;

private Logger logger = LoggerFactory.getLogger(QuestionController.class);

/\*\*

\* 去增加题目页面

\* @param question 去增加页面初始对象

\* @return

\*/

@RequestMapping(value = "/add", method = RequestMethod.GET)

public String add(@ModelAttribute Question question) {

return "questionAdd";

}

/\*\*

\* 增加题目

\* @param question 页面提交的学校对象

\* @param model

\* @return

\*/

@RequestMapping(value = "/add", method = RequestMethod.POST)

public String add(Question question, Model model,HttpSession session) {

User user = (User) session.getAttribute(Constants.SESSION\_USER);

question.setUsername(user.getUsername());

boolean result = questionService.add(question);

if (result) {

return "questionManager";

}else{

return "false";

}

}

/\*\*

\* 手动组卷时 根据 阶段和题型选择题目

\* @param response

\* @param types 题型

\* @throws JsonGenerationException

\* @throws JsonMappingException

\* @throws IOException

\*/

@RequestMapping(value="/searchRandom",method=RequestMethod.GET)

public void searchRandom(HttpServletResponse response,String types) throws JsonGenerationException, JsonMappingException, IOException {

logger.info("手动组卷 条件是 ======" + types);

String json = "";

//设置查询条件

Map<String,Object> map = new HashMap<String, Object>();

map.put("types",types);

List<Question> searchRandomres = questionService.search(map);

if(searchRandomres != null) {

ObjectMapper mapper = new ObjectMapper();

json = mapper.writeValueAsString(searchRandomres);

response.getWriter().print(json);

}

}

/\*\*

\* 试题页面详情

\*/

@RequestMapping(value="/particular",method=RequestMethod.GET)

public String particular() {

return "rMTestPage";

}

/\*\*

\* 查询题目并分页显示

\* @param response

\* @param map 查询条件

\*/

@RequestMapping(value = "/search", method = RequestMethod.GET)

public void search(HttpServletResponse response,int page,int rows,String order,String id,String types,String username) {

logger.info("条件是\*\*\*\*\*\*" + id + types + username);

//设置查询条件

Map<String , Object> map = new HashMap<String, Object>();

int begin = (page-1)\*rows; //起始页

//查询条件

map.put("id", id);

map.put("types", types);

map.put("username", username);

map.put("begin", begin);

map.put("rows", rows); //每页显示的条数

map.put("order", order); // 排序

// 根据条件返回符合条件的题目条数

int total = questionService.searchTotal(map);

List<Question> searchResult = questionService.searchPage(map);

if (searchResult != null) {

// 将查询得到的数组转换成json数据

String json = null;

ObjectMapper mapper = new ObjectMapper();

try {

json = mapper.writeValueAsString(searchResult);

json = "{\"total\":"+total+",\"rows\":" + json + "}";

// 把查询到的记录传到页面进行展示

logger.info("json =" + json);

response.getWriter().print(json);

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

/\*\*

\* 删除题目

\* @param id 参数（题目id）

\* @param session

\* @return

\*/

@RequestMapping(value = "/{id}/remove", method = RequestMethod.GET,

produces="application/json;charset=utf-8")

@ResponseBody

public String remove(@PathVariable int id, HttpSession session) {

Question question = new Question();

question.setId(id);

boolean removeResult = questionService.remove(question);

String json = null;

if (removeResult) {

json = "{\"success\":true,\"msg\":\"删除成功!\"}";

//result = "permissionManager";

} else {

//session.setAttribute("errmsg", "删除失败");

//result = "error";

json = "{\"success\":true,\"msg\":\"删除失败!\"}";

}

return json;

}

/\*\*

\* 去修改题目页面

\* @param question 去增加页面初始对象

\* @return

\*/

@RequestMapping(value = "/{id}/modify", method = RequestMethod.GET)

public String modify(@PathVariable int id, HttpSession session) {

Question question = new Question();

question = questionService.searchById(id);

String result = null;

if(question != null) {

session.setAttribute("question", question);

result = "questionModify";

}else {

session.setAttribute("message", "没有此题目");

result = "message";

}

return result;

}

/\*\*

\* 修改题目

\* @param id 参数（题目id）

\* @param session

\* @return

\*/

@RequestMapping(value = "/modify", method = RequestMethod.POST)

public String modify(Question question, Model model,HttpSession session) {

User user = (User) session.getAttribute(Constants.SESSION\_USER);

question.setUsername(user.getUsername());

boolean result = questionService.modify(question);

if (result) {

return "questionManager";

}else{

session.setAttribute("message", "修改失败");

return "message";

}

}

/\*\*

\* 去题目详情页面

\* @param question 去增加页面初始对象

\* @return

\*/

@RequestMapping(value = "/{id}/searchById", method = RequestMethod.GET)

public String searchById(@PathVariable int id, HttpSession session) {

Question question = new Question();

question = questionService.searchById(id);

String result = null;

if(question != null) {

session.setAttribute("question", question);

result = "questionDetail";

}else {

session.setAttribute("message", "没有此题目");

result = "message";

}

return result;

}

/\*\*

\* 重名验证

\* @param core

\* @param response

\* @throws IOException

\*/

@RequestMapping(value="/rename",method = RequestMethod.POST)

private void rename(String core,HttpServletResponse response) throws IOException {

Question result = questionService.searchByCore(core);

ObjectMapper mapper = new ObjectMapper();

String json = mapper.writeValueAsString(result);

response.getWriter().print(json);

}

/\*\*

\* 提交答案，保存答案并判卷

\* @param question 去增加页面初始对象

\* @return

\*/

@RequestMapping(value = "/submit", method = RequestMethod.GET)

public String submit(HttpServletRequest request) {

String coreRes = request.getParameter("coreRes");

String res = request.getParameter("res");

System.out.println("我对方对方的 "+coreRes+"====="+res);

//按“；”分割字符串

String cores[] = coreRes.split(";");

String options[] = res.split(";");

logger.info ("题目id" + cores);

logger.info ("答案" + options);

return "home";

}

public IQuestionService getQuestionService() {

return questionService;

}

public void setQuestionService(IQuestionService questionService) {

this.questionService = questionService;

}

public IAnswerService getAnswerService() {

return answerService;

}

public void setAnswerService(IAnswerService answerService) {

this.answerService = answerService;

}

}

## 附录C题目数据实现

<?xml version="1.0" encoding="UTF-8" ?>

<!DOCTYPE mapper

PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"

"http://mybatis.org/dtd/mybatis-3-mapper.dtd">

<mapper namespace="com.ssm.survey.dao.IQuestionDao">

<!-- 添加题目 -->

<insert id="insert">

insert into

question(core,option1,option2,option3,option4,username,types,useTime,status)

values(#{core},#{option1},#{option2},#{option3},#{option4},#{username},#{types},0,'可用')

</insert>

<!-- 查询题目总数 -->

<select id="selectTotal" resultType="int" parameterType="Map">

select count(\*) as num from question

<where>

<if test="id != null and id != ''">

and id = #{id}

</if>

<if test="phases != null and phases != ''">

and phases = #{phases}

</if>

<if test="types != null and types != ''">

and types = #{types}

</if>

<if test="username != null and username != ''">

and username like %#{username}%

</if>

</where>

</select>

<!-- 按条件查询题目并分页显示 -->

<select id="selectPage" resultType="Question" parameterType="Map">

select \*

from question

<where>

<if test="id != null and id != ''">

and id = #{id}

</if>

<if test="types != null and types != ''">

and types = #{types}

</if>

<if test="username != null and username != ''">

and username like %#{username}%

</if>

</where>

<if test="order == 'asc'">

order by useTime asc,id

</if>

<if test="order == 'desc'">

order by useTime desc,id

</if>

limit #{begin},#{rows}

</select>

<!-- 条件查询 -->

<select id="select" resultType="Question" parameterType="Map">

select \*

from question

<where>

<if test="id != null and id != ''">

and id = #{id}

</if>

<if test="types != null and types != ''">

and types = #{types}

</if>

<if test="username != null and username != ''">

and username like %#{username}%

</if>

</where>

</select>

<!-- 按id查询题目 -->

<select id="selectById" resultType="Question">

select \* from question where

id = #{id}

</select>

<!-- 按题干查询题目 -->

<select id="selectByCore" resultType="Question">

select \* from question

where core = #{core}

</select>

<!-- 按id删除题目 -->

<delete id="delete">

delete from question where id=#{id}

</delete>

<!-- 按id修改题目 -->

<update id="update">

update question set

core=#{core},option1=#{option1},option2=#{option2},option3=#{option3},

option4=#{option4},username=#{username},types=#{types},

status=#{status} where id=#{id}

</update>

<!-- 按id修改题目的使用次数 -->

<update id="updateUseCount">

update question set useTime = #{useTime} where id=#{id}

</update>

</mapper>