

# B/S 课程 - 问卷应用设计文档

---

By 3170102412 戴将来

## 1. 引言

---

### 1.1 编写目的

为了规范问卷网页应用的编码过程，编写本需求与设计说明书以供软件开发者在实现过程中进行参考。本需求说明书描述了本系统的各项功能和性能需求，明确阐述了各功能的实现过程、实用范围和客户的使用场景，并为其提供了一个遵循的标准。另外，本需求与设计说明书也为程序员和客户提供了一个沟通的桥梁，使得开发者和客户对需求有明确的了解，消除歧义，也是为了使软件开发过程更顺利。

本说明书的预期读者包括：

- 客户
- 项目经理
- 本系统及其他子系统的软件开发人员
- 软件质量分析员
- 软件测试人员

### 1.2 定义

**用户权限：**用户所能进行的操作，例如查询、修改成绩和个人信息。

**用例：**每个用例提供了一个或多个场景，该场景说明了系统是如何和最终用户或其它系统互动，即谁可以用系统做什么，从而获得一个明确的业务目标。

**IPO 图：**输入加工输出图的简称，用来描述每个模块的输入、输出和数据加工过程。

**数据流图：**能全面地描述信息系统逻辑模型，用少数几种符号综合地反映出信息在系统中的流动、处理和存储情况。

**状态图：**描述一个实体基于事件反应的动态行为，显示了该实体如何根据当前所处的状态对不同的事件做出反应。

**CRC 卡：**标准索引卡集合的集合，包括类名、类的职责、类的协作关系三部分，每一张卡片表示一个类。

**数据词典：**定义数据流图中的各个成分的具体含义，对每一个数据流、文件、加工给出详细定义。

**系统集成：**根据用户的需求，通过优化和连接技术将分离的子系统整合成一个有效的整体，使性能达到最优。

## 1.3 项目概述

一个完整的问卷 web 应用，应实现以下功能模块：

### 1. 登陆与注册模块

- 邮箱格式验证
- 用户名格式验证
- 密码格式验证
- 保证用户名与邮箱在数据库中唯一
- 登陆验证
- 忘记密码找回功能

### 2. 账户管理模块

- 修改用户名
- 修改绑定邮箱
- 修改密码
- 申请修改用户会员等级

### 3. 问卷管理模块

- 查找问卷
- 创建问卷
- 删除问卷
- 修改问卷

### 4. 问卷编辑模块

- 单选
- 多选
- 单行文本填写
- 多行文本填写
- 数字填写，支持定义数字类型（整数、小数）
- 评分（单选的特殊形式，界面有所不同）
- 级联选择（对于单选的每个选择，可以动态显示其他一个或多个问卷项目）

### 5. 问卷发送模块

- 动态生成问卷 url
- 自由选择填写模式：
  - 仅限注册用户
  - 无需注册，可填写n次
  - 无需注册，每天可填写n次
- 快速复制填写入口（url）

### 6. 答卷填写模块

- 答卷者账户认证
- 答卷者 ip 认证
- 答卷输入验证
- 答卷提交

### 7. 问卷分析模块

- 查看特定答卷
- 查看特定问题统计数据
- 查看答卷元信息统计
- 删除特定答卷

## 2. 总体设计

---

### 2.1 需求规定

#### 2.1.1 系统功能

账户管理界面，包括所有用户登录界面，找回密码界面，显示/更改账户信息界面；功能主界面，包括问卷总览界面，侧边栏面板，其中侧边栏面板提供对账户管理界面的接口，问卷总览界面提供对创建问卷界面、删除问卷界面、编辑问卷界面、发送问卷界面、统计问卷界面的接口；编辑问卷界面包含对不同种类的问题组件的接口；统计问卷界面包含答卷展示界面的接口与数据统计界面的接口。

#### 2.1.2 系统性能

本系统的性能将由网络数据传输延迟、服务器响应时间内、数据库响应时间以及并发访问该系统的用户数量决定。本系统将采用 React + Material-UI + Spring Boot 的技术栈，应用前后端分离理念，提高维护效率，减轻服务器压力，增加带宽利用率，并向用户提供良好而现代的交互体验。

#### 2.1.3 输入输出要求

网页客户端前端采用 React 开发框架，将组件理念应用到前端，并使用谷歌设计系的 Material-UI 库，力求为用户提供一个美观，友善的操作界面。用户通过点击按钮或者提交表单的方式向服务器下达指令与提供数据。服务器后台处理后将结果显示到用户的网页界面上。

#### 2.1.4 数据管理能力要求

安全：服务器将予以数据库最高等级的保护，防止黑客从后台下载数据库，防止通过网页 SQL 注入的方式从数据库中获取信息或者破坏数据库。

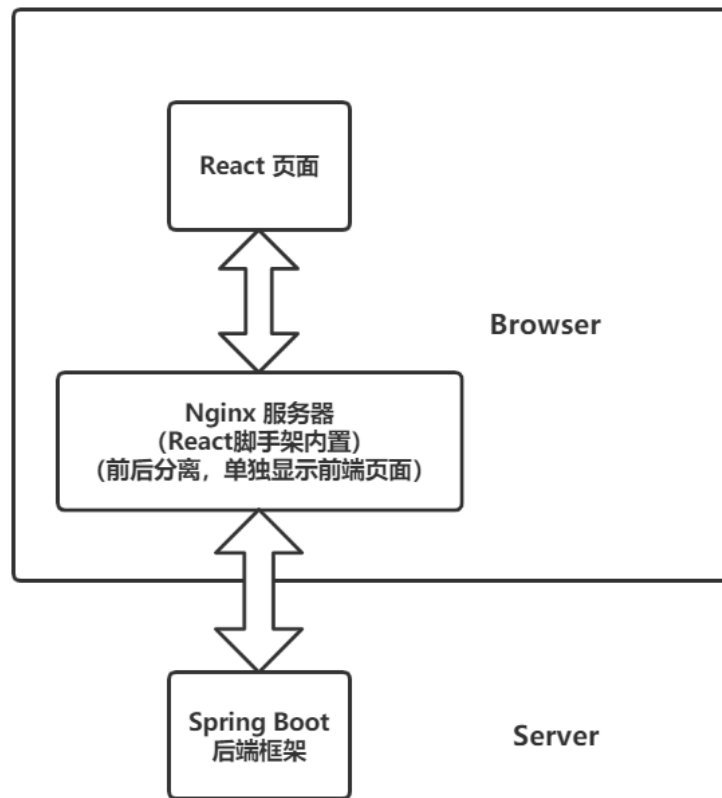
性能：对于频繁的访问数据库的操作，后台建立持久的数据库连接，以避免重复连接数据库耗费资源，并将常用数据保存在 cookies 中。

### 2.2 基本设计概念和处理流程

服务器：采用 Spring Boot 搭建后端，服务器端采用 Java 语言编写，数据库采用 MySQL。

客户端：完美兼容桌面端和手机端 Chrome、Safari、IE8 以上浏览器。采用 React 前端框架，界面简洁美观，兼容性强。以及 Ajax 技术（尽量使用原生方法）减轻服务器压力，增加服务器带宽利用率。

处理流程图如下：



**Browser:** 通过 React 前端框架，采用 Material-UI，设计一个简洁，美观的界面，并通过组件化的设计，在每个组件内完成与后端 api 的连接与互动。

**Server:** 使用 Java 访问/修改数据库。此模块通过 url 接口从 Browser 得到数据与命令，并且对其进行安全检测，而后按照要求访问/修改数据库，并且返回操作结果或是查询结果。

## 3. 详细设计

### 3.1 登录模块设计说明

#### 3.1.1 模块概述

本模块是用户见到的第一个界面，在这个界面进行身份认证成功后方可使用 Future Survey APP 的功能。存在的目的是对用户进行认证，从而控制用户对于 Future Survey APP 的访问。

#### 3.1.2 IPO 图



3.1.3 功能

输入账号密码后登陆，继而进行身份认证。

3.1.4 输入项

名称	标识	数据类型与格式	输入方式
邮箱	mail	varchar(40)	输入框键入
密码	password	varchar(30)	输入框键入
登录	submit_button	<button></button>	鼠标点击

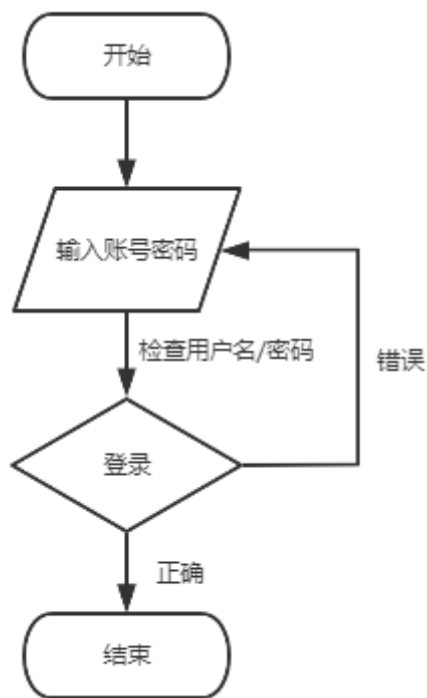
3.1.5 输出项

名称	标识	数据类型与格式	输出方式
提示信息（登录失败）	message	char(100)	警告框文字
主功能界面	home_page	page	新页面

3.1.6 设计方法（算法）

功 能	视 图	控制器	模型	结 果 视 图	说 明
用 户 登 录	登 录 界 面	输入登录信息，获得输入信息并送至Model 进行判断，得到判断结果送至View	存储学生信息，并执行信息的判断方法	显示登录结果	登录有两种结果：成功；失败

3.1.7 流程逻辑

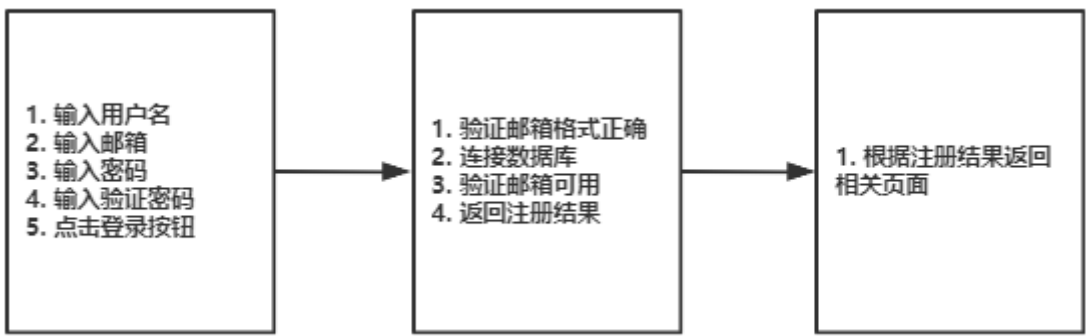


## 3.2 注册模块设计说明

### 3.2.1 模块概述

提供用户注册功能，并对用户进行验证。

### 3.2.2 IPO 图



### 3.2.3 功能

注册验证，更新账户数据库。

### 3.2.4 输入项

名称	标识	数据类型与格式	输入方式
用户名	username	varchar(20)	输入框键入
邮箱	mail	varchar(40)	输入框键入
密码	password	varchar(30)	输入框键入
验证密码	repeat password	varchar(30)	输入框键入

名称	标识	数据类型与格式	输入方式
注册按钮	submit_button	<button></button>	鼠标点击

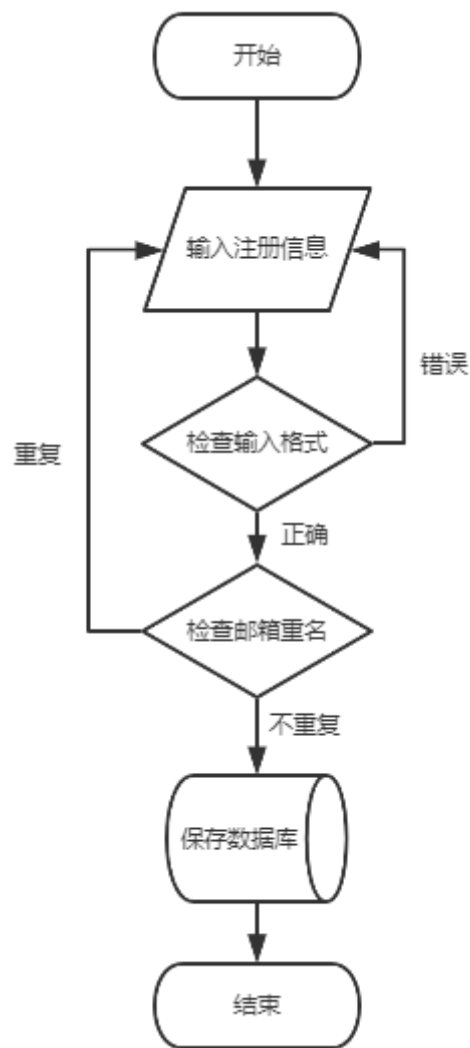
3.2.5 输出项

名称	标识	数据类型与格式	输出方式
提示信息（格式错误）	helpertext	string	前端组件显示 helpertext
提示信息（邮箱重复）	helpertext	string	前端组件显示 helpertext
注册成功	home_page	page	新页面

3.2.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
用户注册	注册界面	输入注册信息，获得输入信息并送至Model 进行判断，得到判断结果送至View	存储用户信息，并执行信息的判断方法	显示注册结果	注册有两种结果：成功；失败

3.2.7 流程逻辑



### 3.3 密码找回模块设计说明

#### 3.3.1 模块概述

此模块以网页表单的形式显示，整合于登陆模块界面中，是为了使暂时回忆不出密码的用户有途径找回密码。

#### 3.3.2 IPO 图



#### 3.3.3 功能

用户可以通过回答验证问题来找回密码，若回答成功则返回密码。



3.3.4 输入项

名称	标识	数据类型与格式	输入方式
找回密码按钮	forget_pwd_btn	<button></button>	鼠标点击
验证问题答案	answer	varchar(45)	输入框键入

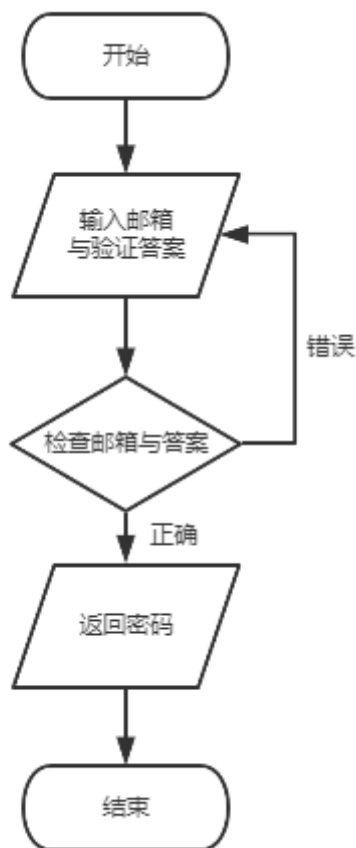
3.3.5 输出项

名称	标识	数据类型与格式	输出方式
原密码	password	varchar(30)	表单文字
找回密码逐步提示界面	query_page	form	网页表单
找回密码验证问题	message_1	varchar(45)	表单文字
成功找回密码提示信息	message_1	varchar(45)	表单文字
找回密码失败提示信息	message_2	varchar(45)	表单文字

3.3.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
找回密码	密码找回界面	输入验证信息，获得输入信息并送至 Model 进行判断，得到判断结果送至View	存储用户信息，并执行信息的判断方法	显示验证结果	通过验证找回密码有两种结果：成功；失败

3.3.7 流程逻辑



## 3.4 修改账户信息模块设计说明

### 3.4.1 模块概述

此模块以网页表单的形式显示，是在登陆界面后用户可使用的功能模块，整合于查询用户信息界面中，存在目的是让用户得以修改自己的个人信息。

### 3.4.2 IPO 图



### 3.4.3 功能

满足权限要求的用户只要选择在每条用户信息记录右侧的修改用户信息功能按钮，按键点击，页面就可弹出默认值为用户原信息的表单。系统可接收用户在表单中的修改内容，更新数据库，并在原页面下方显示修改更新后的结果。

### 3.4.4 输入项

名称	标识	数据类型与格式	输入方式
用户名	user_name	varchar(35)	表格文字
密码	password	varchar(30)	表格文字
联系号码	phone	varchar(15)	表格文字
联系邮箱	email	varchar(45)	表格文字

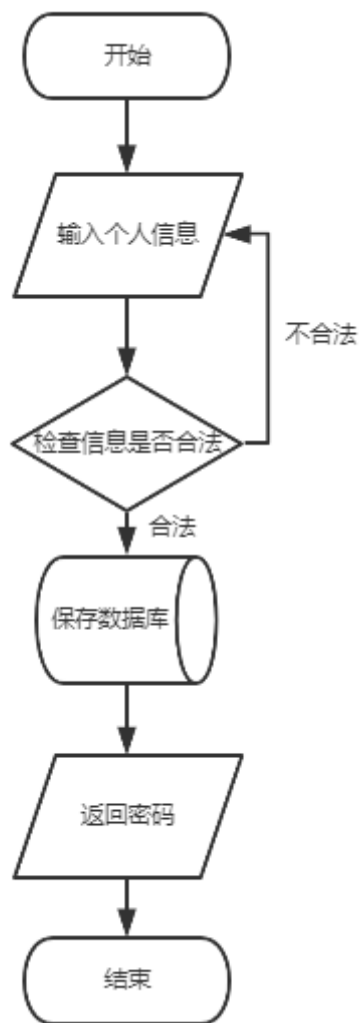
3.4.5 输出项

名称	标识	数据类型与格式	输入方式
用户名	user_name	varchar(35)	表格文字
密码	password	varchar(30)	表格文字
联系号码	phone	varchar(15)	表格文字
联系邮箱	email	varchar(45)	表格文字

3.4.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
修改账户信息	账户信息界面	输入更新的信息，获得输入信息并送至 Model 进行判断，得到判断结果送至View	存储账户信息，并执行信息的判断方法	显示修改结果	修改账户信息有两种结果：成功；失败

3.4.7 流程逻辑

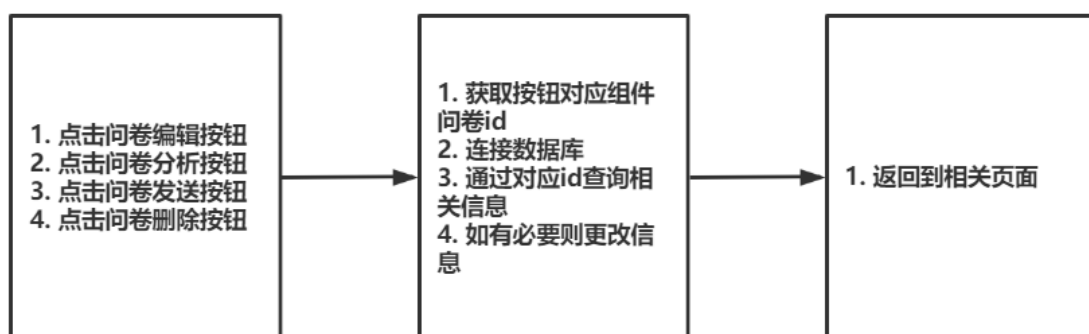


## 3.5 问卷总览模块设计说明

### 3.5.1 模块概述

属于功能主界面的主要组成部分，用于展示用户保存的所有问卷。

### 3.5.2 IPO 图



### 3.5.3 功能

展示用户所有问卷的主要信息，并给用户提供对各个问卷进行不同操作的接口。

### 3.5.4 输入项

名称	标识	数据类型与格式	输入方式
用户ID	user_id	varchar(5)	从 session 中获取
问卷ID	form_id	varchar(7)	从前端 components 中获取
编辑按钮	edit_btn	<button></button>	鼠标点击
分析按钮	analyze_btn	<button></button>	鼠标点击
发送按钮	send_btn	<button></button>	鼠标点击
删除按钮	删除按钮	<button></button>	鼠标点击

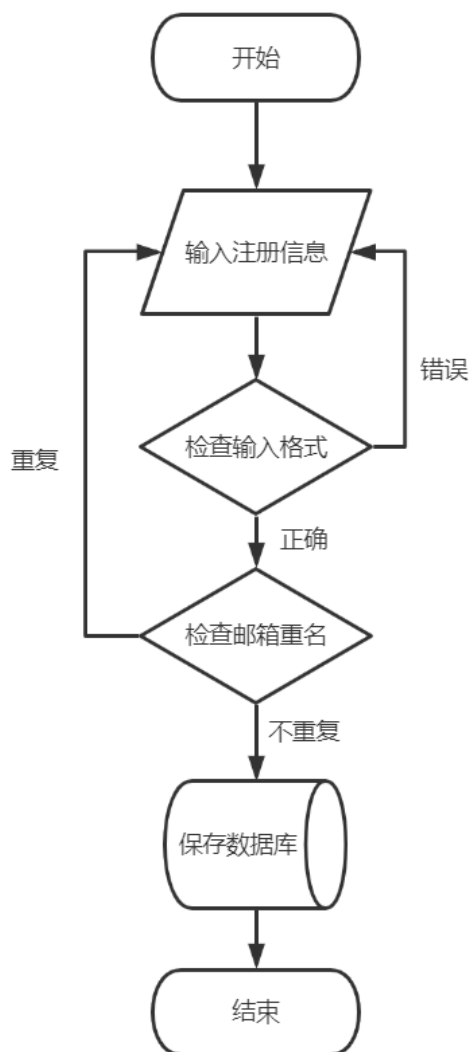
### 3.5.5 输出项

名称	标识	数据类型与格式	输出方式
问卷卡片组件	surveyCard	<React.Fragment />	主功能界面渲染
编辑页面	editPage	Page	新页面
分析页面	analyzePage	Page	新页面
问卷url	form_url	String	发送框内渲染
发送对话框	sendDialog	<Dialog></Dialog>	主功能界面渲染
删除对话框	deleteDialog	<Dialog></Dialog>	主功能界面渲染

### 3.5.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
问卷总览与增删改查	主功能界面问卷板块	输入对问卷的操作，获得输入信息并送至 Model 进行判断，得到判断结果送至View	存储问卷信息，并执行信息的判断方法	显示对应交互	问卷总览有两种结果：成功；失败

### 3.5.7 流程逻辑



## 3.6 问卷创建模块设计说明

### 3.6.1 模块概述

属于主界面提供的模块功能之一，以一个上方按钮提供这个模块的用户接口。

### 3.6.2 IPO 图



### 3.6.3 功能

为用户创建新问卷，返回问卷编辑页面，并在数据库中保存相应条目。

### 3.6.4 输入项

名称	标识	数据类型与格式	输入方式
添加问卷按钮	add_form_btn	<button></button>	鼠标点击

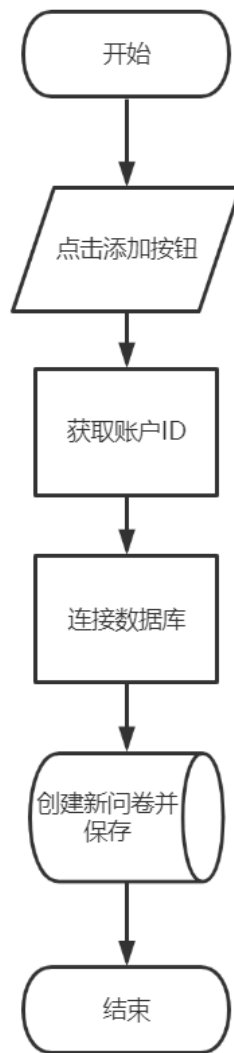
### 3.6.5 输出项

名称	标识	数据类型与格式	输出方式
新问卷编辑页面	editPage	Page	新页面
失败提示框	failToAddDialog	<Dialog></Dialog>	对话框

### 3.6.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
添加新问卷	主功能界面添加问卷按钮	点击添加问卷按钮发送输入信息，获得输入信息并送至 Model 进行判断，得到判断结果送至 View	新建并存储问卷信息，并执行信息的判断方法	新问卷编辑界面或错误提示	添加新问卷有两种结果：成功；失败

### 3.6.7 流程逻辑

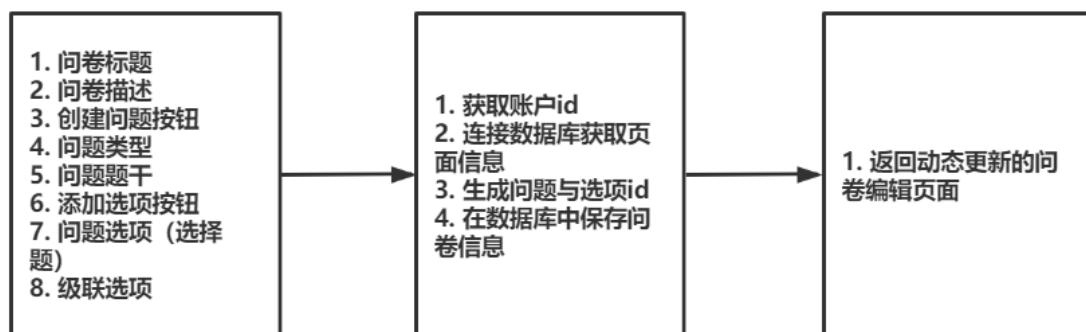


## 3.7 问卷编辑模块设计说明

### 3.7.1 模块概述

通过创建新问卷或者点击问卷卡片的编辑按钮进入该模块，对问卷进行编辑。

### 3.7.2 IPO 图



### 3.7.3 功能

对问卷标题、提示、题目内容等信息进行增删改查。

### 3.7.4 输入项



名称	标识	数据类型与格式	输入方式
问卷标题	name	varchar(80)	输入框键入
问卷描述	description	varchar(200)	输入框键入
创建问题按钮	add_qstn_btn	<button></button>	光标单击
问题类型	qstn_type	varchar(1)	光标选择
问题题干	qstn_body	varchar(300)	输入框键入
添加选项按钮	add_option_btn	<button></button>	光标单击
问题选项（选择题）	options	varchar(300)	输入框键入
级联选项	cascade_qstnid	varchar(4)	输入框键入
提交编辑按钮	submit_btn	<button></button>	光标单击

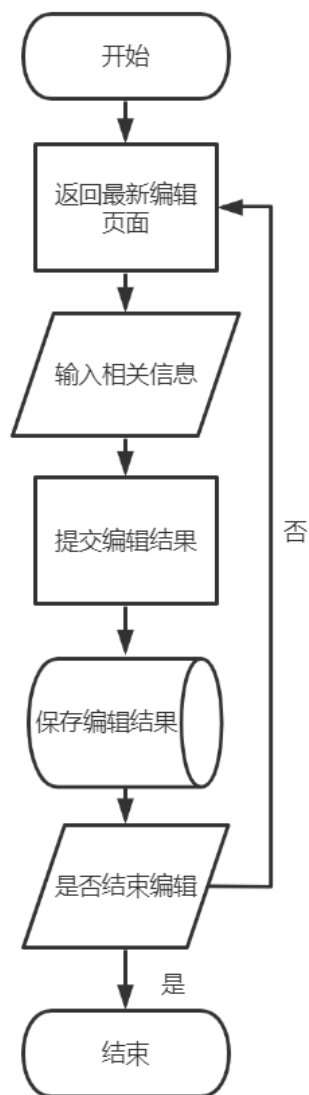
### 3.7.5 输出项

名称	标识	数据类型与格式	输出方式
问卷编辑页面	editPage	Page	页面渲染

### 3.7.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
问卷编辑	问卷编辑页面	输入对问卷具体内容的操作，获得输入信息并送至 Model 进行判断，得到判断结果送至View	存储问卷信息，并执行信息的判断方法	问卷编辑页面	问卷编辑有两种结果：成功；失败

### 3.7.7 流程逻辑



## 3.8 问卷发送模块设计说明

### 3.8.1 模块概述

功能主界面与问卷编辑界面均能调用此模块，提供用户完成编辑后的问卷填写入口地址。

### 3.8.2 IPO 图



### 3.8.3 功能

向用户提供获取问卷填写入口的接口。

### 3.8.4 输入项

名称	标识	数据类型与格式	输入方式
复制按钮	copy_btn	<button></button>	光标单击

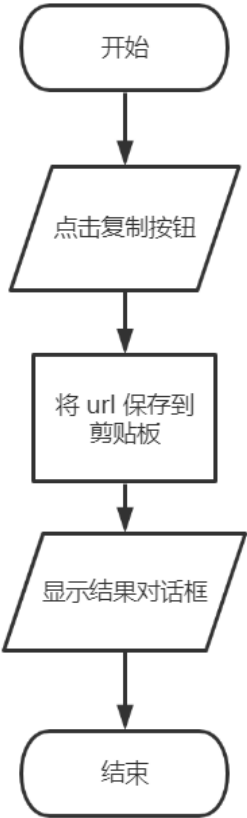
3.8.5 输出项

名称	标识	数据类型与格式	输出方式
问卷url	form_url	String	保持至剪贴板
复制结果对话框	copy_dialog	<Dialog></Dialog>	对话框

3.8.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
问卷发送	问卷发送对话框	输入问卷id与用户id，获得输入信息并送至 Model 进行判断，得到判断结果送至View	输出问卷信息，并执行信息的判断方法	问卷发送对话框	获取问卷发送入口有两种结果：成功；失败

3.8.7 流程逻辑

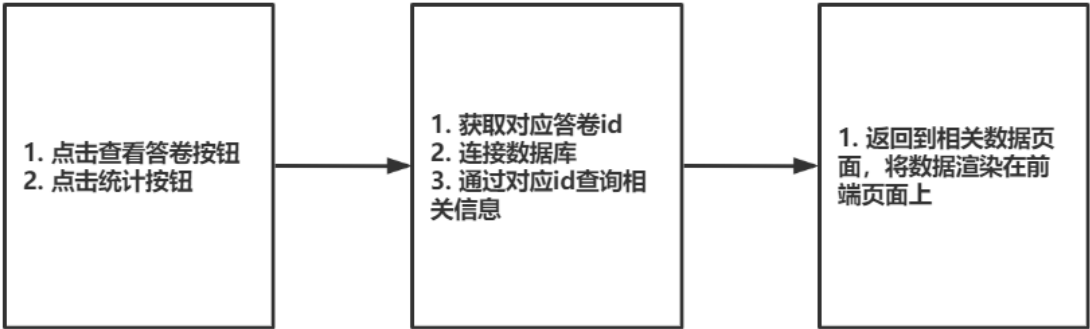


### 3.9 答卷分析模块设计说明

#### 3.9.1 模块概述

由主功能界面进行调用，提供对问卷结果的简单统计。

#### 3.9.2 IPO 图



#### 3.9.3 功能

在这个模块，用户将能看到已发布问卷收回的结果的简单统计数据，乃至查看具体的问卷答复。

#### 3.9.4 输入项

名称	标识	数据类型与格式	输入方式
查看答卷按钮	view_resp_btn	<button></button>	光标单击
答卷id	form_id	varchar(7)	前端 components 提供
统计按钮	stats_btn	<button></button>	光标单击

#### 3.9.5 输出项

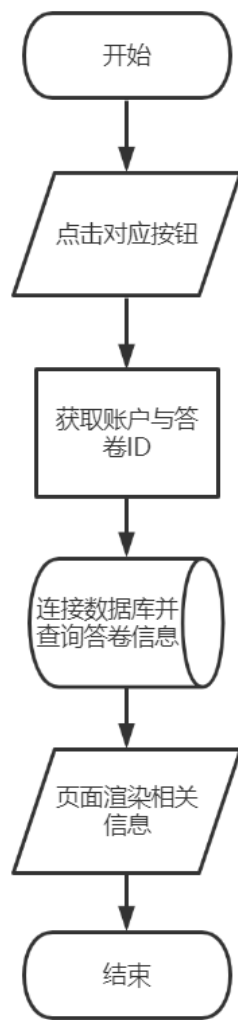
名称	标识	数据类型与格式	输出方式
答卷查看页面	viewPage	Page	新页面
结果统计页面	statsPage	Page	新页面

#### 3.9.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
答卷查看	答卷查看			答卷查看	

答卷功能分析	答卷分析页面	输入查看特定答卷或查看统计数据的需求，获得输入信息并送至 Model 进行判断，得到判断结果送至View	存储答卷信息，并执行模型判断方法	答卷分析结果页面	答卷分析有两种结果：成功；失败
--------	--------	--	------------------	----------	-----------------

3.9.7 流程逻辑



3.10 问卷删除模块设计说明

3.10.1 模块概述

由问卷总览模块调用，实现问卷删除。

3.10.2 IPO 图



3.10.3 功能

给用户接口，删除自己账号下的问卷。

3.10.4 输入项

名称	标识	数据类型与格式	输入方式
确认按钮	conf_delete_btn	<button></button>	光标单击
取消按钮	cancel_delete_btn	<button></button>	光标单击

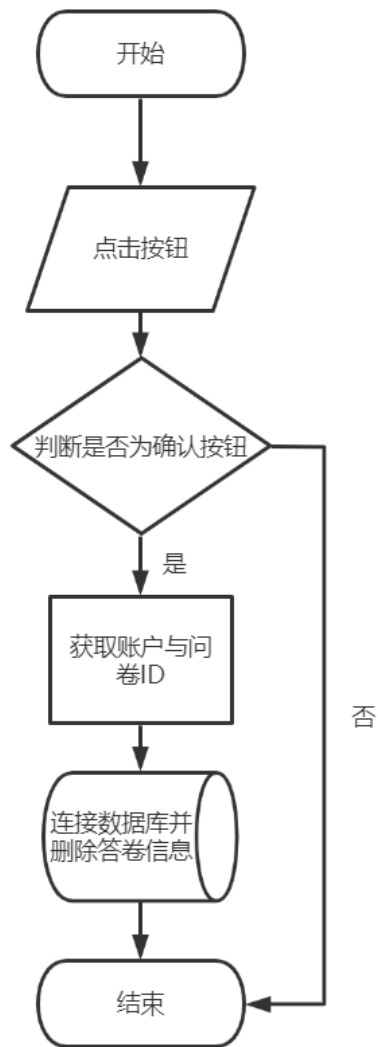
3.10.5 输出项

名称	标识	数据类型与格式	输出方式
更新后的问卷总览页面	homePage	Page	新页面

3.10.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
删除问卷	问卷删除页面	输入删除特定答卷的请求，获得输入信息并送至 Model 进行判断，得到判断结果送至View	删除答卷信息，并执行信息的判断方法	更新后的问卷总览页面	问卷删除有两种结果：成功；失败

3.10.7 流程逻辑

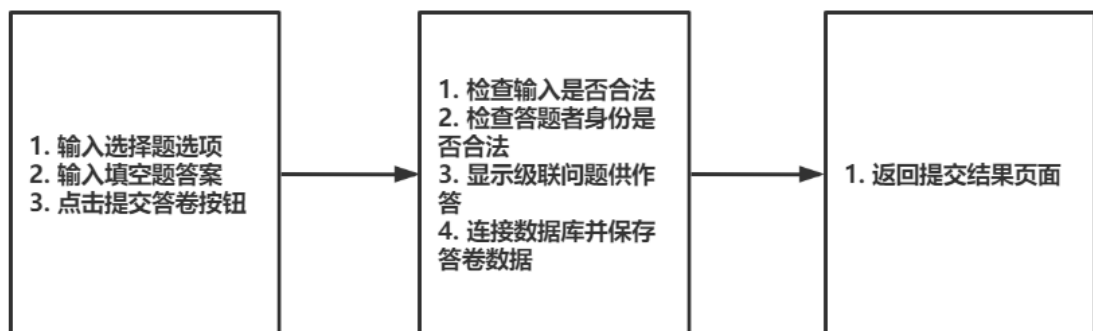


## 3.11 问卷填写模块设计说明

### 3.11.1 模块概述

该模块通常直接由答题者通过 url 访问。

### 3.11.2 IPO 图



### 3.11.3 功能

使访问者能够在该页面上作答并提交答卷，答卷数据将存入数据库，供答卷所有者调取查看。

### 3.11.4 输入项

名称	标识	数据类型与格式	输入方式
问题选项（选择题）	selection	varchar(200)	光标单击
问题答案（填空题）	answer	varchar(200)	输入框键入
问卷提交按钮	resp_submit_btn	<button></button>	光标单击

### 3.11.5 输出项

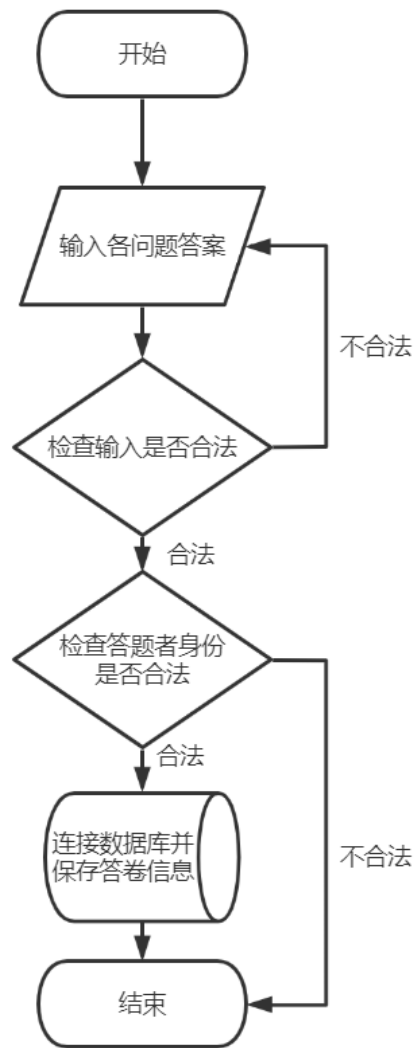
名称	标识	数据类型与格式	输出方式
级联问题	cascade_qstn	<QuestionCard></QuestionCard>	题目卡片
提交结果界面	submittedPage	Page	新页面

### 3.11.6 设计方法（算法）

功能	视图	控制器	模型	结果视图	说明
问卷填写	问卷填写页面	输入各个题目的答案，获得输入信息并送至 Model 进行判断，得到判断结果送至View	存储答卷信息，并执行信息的判断方法	答卷分析页面	答卷分析有两种结果：成功；失败

### 3.11.7 流程逻辑






## 4. 接口设计

---

### 4.1 用户接口

#### 4.1.1 登录界面



## Sign in

No account yet? [Sign up here](#)

Email

root@root.com


Password

\*\*\*\*\*

SIGN IN

[Forgot password?](#)

### 4.1.2 注册界面



## Sign up

Already have an account? [Sign in here](#)

Username \*

root@root.com

Email \*

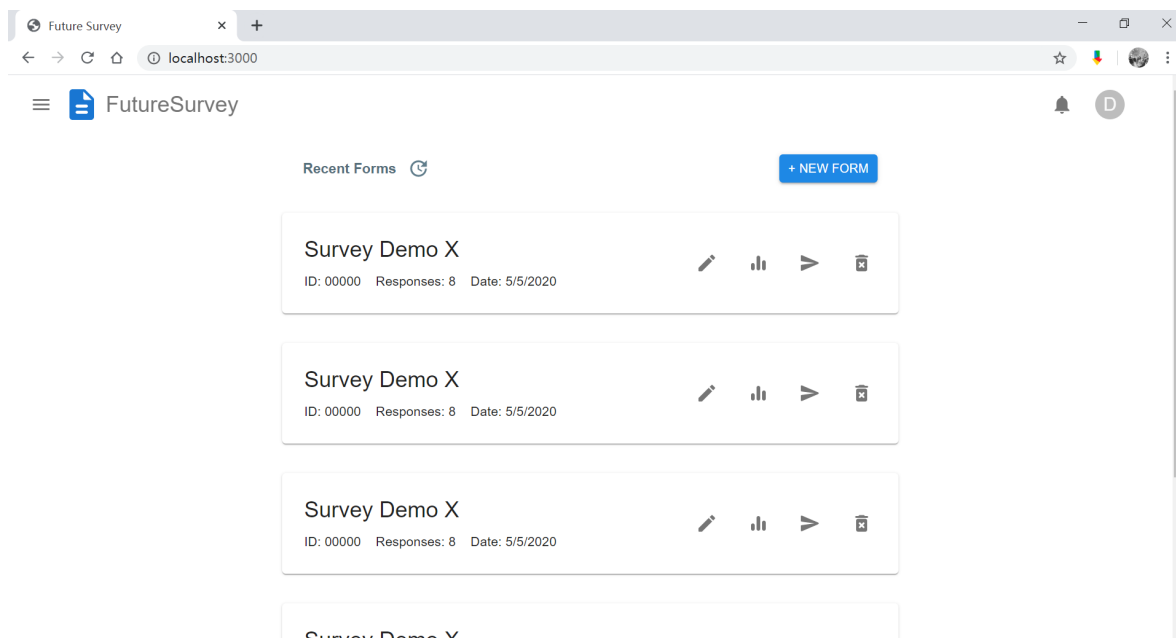
Password

\*\*\*\*\*

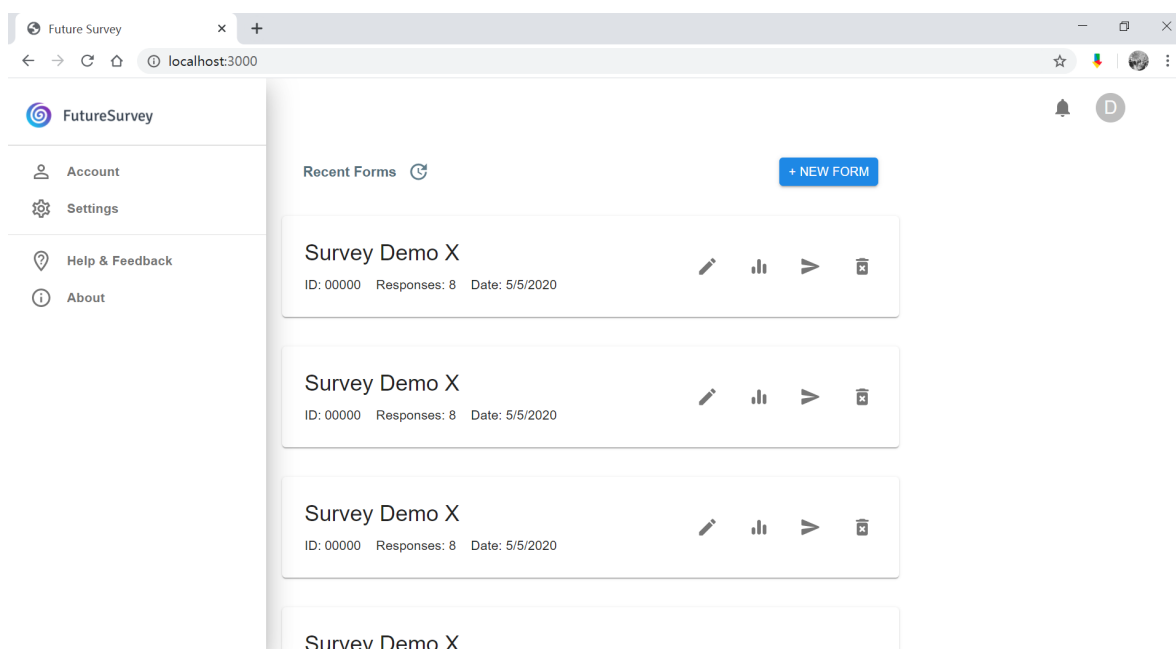
Repeat Password \*

SIGN UP

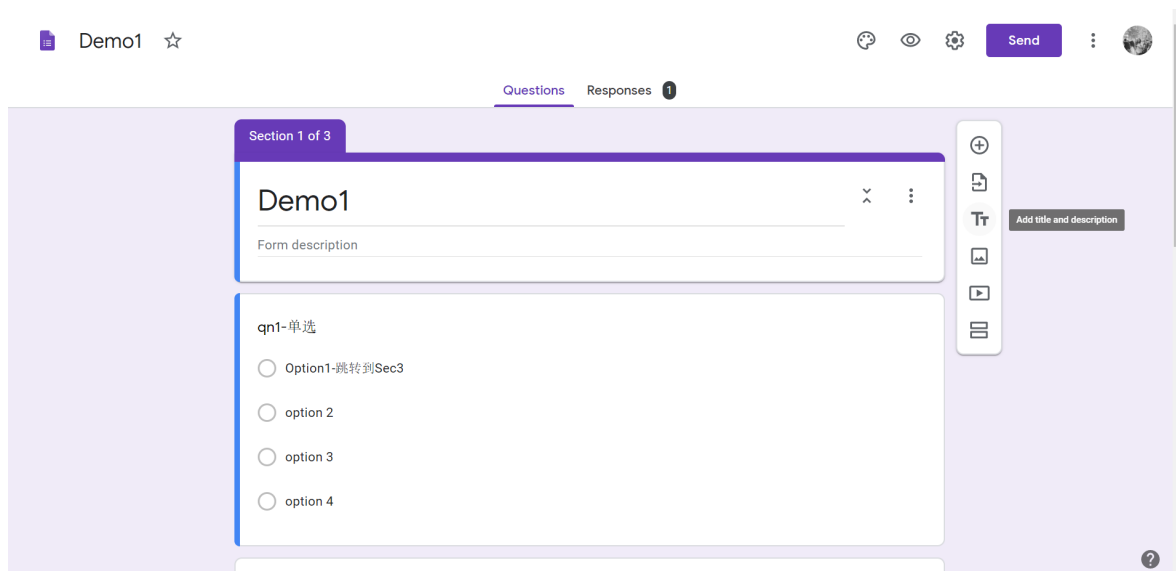
### 4.1.3 应用主界面



#### 4.1.4 侧边栏选项界面



#### 4.1.5 问卷编辑界面 (效果图)



### 4.1.6 问卷填写界面（效果图）

Demo1

qn1-单选

☐ Option1-跳转到Sec3

☐ option 2

☐ option 3

☐ option 4

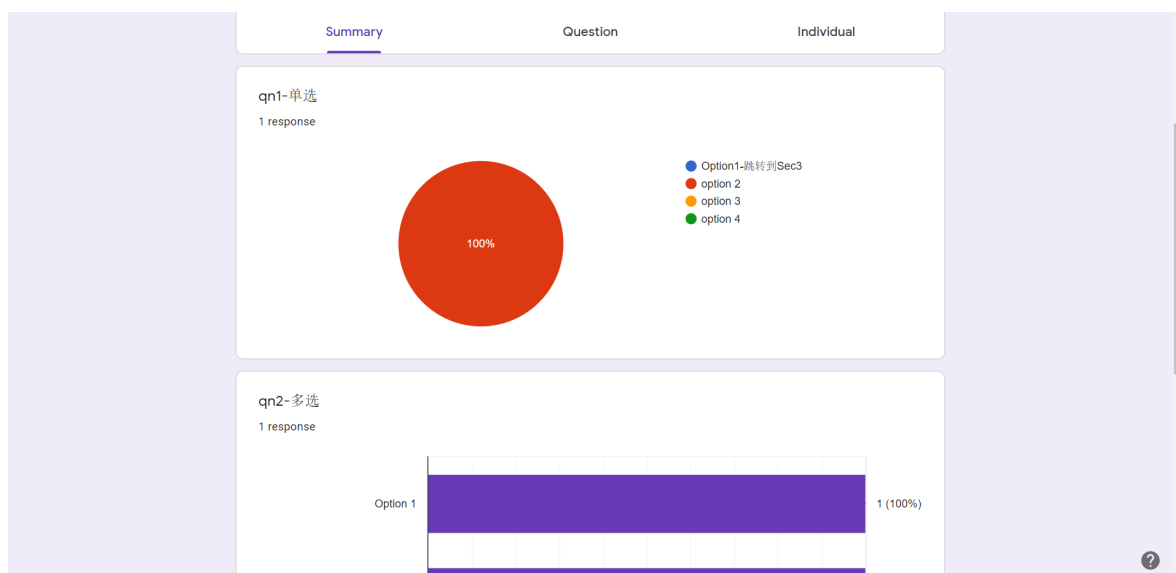
qn2-多选

☐ Option 1

☐ Option 2

qn3-填空

### 4.1.7 问卷统计界面（效果图）



## 4.2 外部接口

在 Spring Boot 中使用 JdbcTemplate 与 MySQL 建立连接。

## 4.3 内部接口

账户管理界面，包括所有用户登录界面，找回密码界面，显示/更改账户信息界面；功能主界面，包括问卷总览界面，侧边栏面板，其中侧边栏面板提供对账户管理界面的接口，问卷总览界面提供对创建问卷界面、删除问卷界面、编辑问卷界面、发送问卷界面、统计问卷界面的接口；编辑问卷界面包含对不同种类的问题组件的接口；统计问卷界面包含答卷展示界面的接口与数据统计界面的接口。

## 5. 系统数据结构设计

## 5.1 逻辑结构设计

账号管理：accounts ( user\_id, mail, password, name, user\_type )

问卷元信息：forms ( form\_id, description, name, date )

问题管理：questions ( qstn\_id, form\_id, type, content )

选项管理：options ( opt\_id, qstn\_id, form\_id, content, cascade\_qtnid )

答卷元信息：responses ( resp\_id, form\_id, submit\_time, ip )

答案管理：answers ( resp\_id, form\_id, qstn\_id, content )

## 5.2 物理结构设计

### 5.2.1 账号管理

字段	类型	能否为空	是否主键	备注
user_id	varchar(5)	N	Y	用户id
mail	varchar(30)	N	N	邮箱
password	varchar(30)	N	N	密码
username	varchar(35)	N	N	用户名
user_type	varchar(1)	N	N	用户等级

### 5.2.2 问卷元信息

字段	类型	能否为空	是否主键	备注
form_id	varchar(7)	N	Y	问卷id
description	varchar(200)	Y	N	描述
name	varchar(80)	N	N	标题
date	varchar(20)	N	N	修改日期

### 5.2.3 问题管理

字段	类型	能否为空	是否主键	备注
qstn_id	varchar(4)	N	Y	题目id
form_id	varchar(5)	N	Y	问卷id
type	varchar(1)	N	N	问题类型
content	varchar(300)	N	N	题干

5.2.4 选项管理

字段	类型	能否为空	是否主键	备注
opt_id	varchar(2)	N	Y	选项id
qstn_id	varchar(4)	N	Y	题目id
form_id	varchar(5)	N	Y	问卷id
content	varchar(300)	N	N	题干
cascade_qstn_id	varchar(4)	N	N	级联问题id

5.2.5 答卷元信息

字段	类型	能否为空	是否主键	备注
resp_id	varchar(10)	N	Y	答卷id
form_id	varchar(5)	N	Y	问卷id
submit_time	varchar(40)	N	N	提交时间
ip	varchar(30)	N	N	答卷ip地址

5.2.6 答案管理

字段	类型	能否为空	是否主键	备注
resp_id	varchar(10)	N	Y	答卷id
form_id	varchar(5)	N	Y	问卷id
qstn_id	varchar(4)	N	Y	题目id
content	varchar(200)	N	N	提交的答案

6. 运行设计

6.1 运行模块组合

本子系统按照功能划分模块，每个模块又按照流程划分为客户端界面，客户端脚本，服务器后台程序。出了主界面之外，功能模块之间相互不共享界面和客户端脚本，服务器后台程序只共享建立数据库连接的方法。

6.2 运行控制

详见详细设计。

## 6.3 运行时间

由于采用 Ajax 技术，实现客户端静态更新信息，静态分页等等，因此，服务器的带宽利用率大大提高，唯一会影响服务器的运行时间的，应该只是客户端频繁建立销毁数据库连接所带来的时间开销

# 7. 系统出错设计

## 7.1 出错信息

系统输出信息的形式	含义	解决办法
未登录时功能页面无法打开	没有登录时访问用户信息、课程信息等详情页面时无法转入相应页面	提示“非法连接”，并提示用户登录
输入错误	在任何有输入框的地方输入非法的字符串，例如 SQL 语句，企图通过 SQL 注入来破坏数据库	对表单的输入进行过滤，阻止非法输入
数据库无法连接	数据库配置不对	在源码中修改数据库配置
暂时无法连接网站服务器	由于短时间内出现大量攻击可能导致的服务器瘫痪	联系系统管理员进行紧急处理
账户密码错误	由于用户忘记密码或者被人恶意篡改密码造成用户无法登录	通过登录界面的“找回密码”功能对密码进行修改
数据库编码错误	由于用户端编码方式不一样，在数据库读出写入的过程中也会涉及数据的编码问题	统一编码为 utf-8

## 7.2 补救措施

确保数据完整性和一致性：将数据库进行多处备份，并及时校对、更新；除线上保存之外，还可以进行线下存档。工程代码通过 GitHub 进行完善的版本管理。

恢复技术：如果系统被攻击，数据库被修改，通过查看系统日志，获得更多信息并导入数据库备份文件进行恢复。

紧急处理：若网站被攻击瘫痪后，需要修改数据，联系系统管理员在后台进行数据更新。

## 7.3 系统维护设计

- 在可能出错的地方使用 try-catch 语句捕获异常，并输出相应的出错信息和可能的处理方法提示。

- 系统管理员定时查看系统日志，统计非法攻击来源和次数，并针对相应攻击加强安全防范措施。
- 系统管理员的登录不仅需要账户密码，还需要识别 IP，禁止非白名单 IP 的任何访问。
- 系统维护人员及时更新技术漏洞，增强代码的可靠性。