

西安欧亚学院

本科毕业论文（设计）

题 目 基于微信小程序的移动端第二课堂学分管理系统
设计与实现

学生姓名	黄森
学生学号	14610801152313
指导教师	张迪（专职） 任志宏（专职）
导师职称	讲师 副教授
所在分院	信息工程学院
专 业	软件工程
班 级	统本软件 1401 班
提交日期	二〇一八年四月

基于微信小程序的移动端第二课堂学分管理 系统设计与实现

摘要：随着信息时代的到来,各高校用计算机管理信息越来越普遍，现在微信小程序也在迅速发展,而且对开发人员来说,程序的开发成本更低，可以把有限的人、物资、财力置于具有经营价值的产品之上，可以放在如何运营产品上，对于用户来说，与各种 APP 相比，微信小程序操作简单、跨平台、不需要下载安装，会降低用户的使用难度，推广起来也更容易、更省成本。现在，传统的纸质管理方式，必然被以计算机为基础的信息管理系统所取代，为了解决纸质管理方式所带来的问题，基于微信小程序的移动端第二课堂学分管理系统具有可以在线活动报名、第二课堂学分查看、学分明细查看、学分申请进度查看、活动签到签退、活动的审核等功能，可以实现学生第二课堂管理工作的优化流程，加强部门之间的信息交流与协同工作，方便学生、方便教务处及分院，在提高工作效率的同时为学校师生提供更加便捷高效的服务。因此，该系统的设计与实现有着非常重要的意义。

关键词：微信小程序；第二课堂；APP

The Design and Implementation of Second class Credit management system for mobile terminal based on WeChat small program

Abstract: With the coming of information age, computer information management in colleges and universities are becoming more common, Now WeChat small programs are also developing rapidly, but for developers, program development is cheap. we can put the limits people, material and financial resources to those products which worth, for users, compared with the various APP, WeChat small program that can cross-platform, does't need to download and install, it is easy for users to use, The promotion is also easy and cost cheap. Now, the management mode of the traditional paper must be based on computer information management system replaced, in order to solve the problems brought by the paper management way, based on WeChat small program the mobile terminal of the second class for credit management system has online activity registration, the second class for credit view, credits the detail view, credit application progress view, activity checker is signed off., activities of the audit, and other functions, can realize the optimization of the management work of the second class process, strengthen the communication between departments and work together, convenient students, office and branch, while improve the work efficiency is to provide more convenient and efficient service to teachers and students. So, the design and implementation of the system is of great significance.

Keywords: WeChat small program; Second class; APP

目录

1	引言.....	1
1.1	研究背景与意义.....	1
1.2	研究内容与目标.....	1
2	需求分析.....	3
2.1	系统描述.....	3
2.2	系统功能需求分析.....	3
2.3	可行性分析.....	3
2.3.1	技术可行性分析.....	3
2.3.2	运行环境.....	4
2.4	经济可行性分析.....	4
3	技术栈介绍.....	5
3.1	开发环境.....	5
3.2	微信开发者工具介绍.....	5
3.3	ECMAScript6 介绍.....	5
3.4	WXML 和 WXSS 介绍.....	5
3.5	SpringMVC、Spring、MyBatis 介绍.....	6
3.6	RESTful 介绍.....	7
4	系统设计.....	9
4.1	系统架构.....	9
4.2	系统功能.....	10
4.3	业务流程.....	11
4.3.1	学生端登录模块.....	12
4.3.2	学生端活动报名模块.....	13
4.3.3	活动签到签退模块.....	14
4.3.4	活动审核模块.....	15
4.4	数据库设计.....	15
4.4.1	数据库概念设计.....	15
4.4.2	数据库表结构的设计.....	18
5	系统实现.....	28
5.1	请求资源实现.....	28
5.2	学生端.....	29
5.2.1	登录页面.....	29
5.2.2	系统主页.....	29

5.2.3	活动报名.....	30
5.2.4	活动预约查看.....	32
5.2.5	活动签到.....	33
5.2.6	学分申请查看.....	34
5.2.7	学分明细查看.....	35
5.3	教师端.....	36
5.3.1	登录.....	36
5.3.2	系统主页.....	36
5.3.3	活动审核和定位.....	37
5.3.3.1	活动审核列表.....	38
5.3.3.2	活动定位.....	41
6	系统测试.....	42
6.1	测试环境.....	42
6.2	系统后端 API 测试.....	42
6.3	系统功能模块测试.....	43
7	结论.....	46
7.1	总结.....	46
7.2	展望.....	46
	致谢.....	47
	参考文献.....	48

1 引言

1.1 研究背景与意义

随着社会的不断进步与改革,已经进入一个全新的时代,各国和地区之间的竞争日趋激烈。竞争的核心是人才的质量和竞争,随着社会的发展,高等教育在社会中的地位和作用越来越突出。在高等教育中,课堂教学是对学生实施教育的主渠道,但是课堂教学远远不能满足教育的要求,第二课堂的参与是有必要的。第二课堂是第一课堂的补充,它不受时间和空间的限制,它发展了各种形式,成为素质发展、创新意识培养和实践能力实现的重要载体,正是由于第二课堂教学的灵活性和新颖性,使其越来越受到高校的重视。高校以第二课堂的教育方式经过多年的发展,不仅是对第一课堂教学的补充,也是高校人才培养和教育的重要组成部分。第二课堂的学习与深化是培养大学生创新能力的客观要求,同时也是提高大学生综合素质的重要途径。

西安欧亚学院的第二课堂管理部门以各个分院、学生发展处、教务处为主,学生第二课堂学分认定涉及的种类繁多,认证流程较为复杂。这种现状导致学生的学分认定流程分散到各分院及学生发展处,老师不能够全面的了解到学生的情况,客观上造成各个管理部门之间信息的认知不一致以及局限性,具体表现为:

- 1、学校学生两万余人,学生的第二课堂事务复杂,信息量巨大,涉及多个部门和人员,对各个学院、部门之间的信息交流与协同工作要求很高。
- 2、学校目前人工纸介质、电子文档方式管理学生第二课堂学分的方式存在许多缺点,工作效率低下、协作性差、保密性差、人力浪费,产生大量的历史文件和数据,不方便进行历史数据查寻,不利于数据更新和维护。
- 3、目前的管理方式无法做到学生第二课堂的精细化管理,无法实时准确的了解学生的学分信息,无法对学生的第二课堂全历程进行管理与监督。
- 4、学生第二课堂没有一个统一的管理入口,造成求助难、处理问题慢、无法了解处理流程的问题,学生迫切需要统一第二课堂管理平台。

1.2 研究内容与目标

第二课堂学分管理系统有电脑端和手机端,电脑端包括学分申请,活动发布,活动审核,用户管理四个大的模块。手机端分为活动报名,活动预约查看,活动签到,学分申请查看,学分明细查看和学分审核六个模块,本文主要对手机端的六个模块进行分析与研究。

本文将按照以下的几个章节进行研究和分析:

第1章是引言,主要分析了当前高校纸质化管理的弊端,以及微信端第二课堂学分管理系统的优势和发展前景。

第 2 章是需求分析，主要对该系统的可行性以及功能展开分析和说明。

第 3 章是技术介绍，简单介绍 RESTful 架构、ECMAScript6、WXML、WXSS、ECMAScript6、SSM（SpringMVC、Spring、MyBatis）等开发时所用到的技术。

第 4 章是系统设计，主要对该系统的系统功能、业务流程、数据库设计和界面设计进行分析说明。

第 5 章是系统的实现，主要展现实现后的界面以及流程。

第 6 章是系统测试，主要介绍了系统中使用的测试方法、工具和测试用例。

第 7 章是总结和展望，本文主要对系统的完成进行了总结,并对未来的改进和展望作了简要的描述。

该系统目标采用了当前主流的 Java 开发技术，在 RESTful 架构的基础上采用前后端分离的开发模式进行开发，以及自动化测试来保证每个开发环节的正确性，运用 Git 来对项目的每个环节进行版本控制。

2 需求分析

2.1 系统描述

系统名称：基于微信小程序的移动端第二课堂学分管理系统。

系统用途：学生通过该系统，可以在系统上查询自己的第二课堂活动信息、报名信息、学分等一系列相关信息，还可以实现在线活动申请、活动预约查看等功能，方便了学生的第二课堂学分申请及获取。教师可以在手机端进行活动的审核，以及活动签到、签退的定位。

系统的功能要求：在 SpringMVC、Spring、MyBatis 的架构基础上通过前后端分离的开发方式实现第二课堂学分管理系统，系统设计简单，操作方便。前端使用微信小程序开发语言，系统的用户体验更加流畅。

2.2 系统功能需求分析

对于该系统，涉及到的角色有两个，老师(学生发展处)和学生，所以将该系统分为学生端和教师端，通过不同的账号登录进入系统。

1、学生端，学生端不支持自己注册账号，使用自己的学号和密码进行登录，学生进入该系统可以进行活动报名、活动预约查看、活动签到签退、学分申请查看、学分明细查看。在规定时间内可以进行活动报名，活动开始和活动结束时进行扫二维码和定位进行活动的签到或签退，可以对自己预约的活动进行查看，以及活动内容更改后，发的通知消息的查看，还可以查看自己申请学分的流程和学分明细的查看。

2、教师端，教师进入该系统后，可以查看学生参加活动报名情况，可以对参加这些活动的同学进行活动报名的审核，可以批量审核也可以单个的审核。教师和学生都不支持自行注册账号，管理员可以在 PC 端后台进行人员角色的分配。可以增加教师的数量，对审批流程的操作。

2.3 可行性分析

2.3.1 技术可行性分析

系统模式：B/S

开发工具：微信开发者工具

开发环境：Windows

运行环境：Android/IOS

数据库：MySQL

数据库工具：Navicat Premium

2.3.2 运行环境

第二课堂学分管理系统属于 B/S 架构，所以在访问的时候需要打开手机端微信，在微信小程序中进行操作。

2.4 经济可行性分析

本系统是基于 SpringMVC、Spring、MyBatis 的架构基础上实现前后端分离的开发方式实现，前台负责显示数据和逻辑交互，后端负责接收前端请求，处理业务逻辑。前后端任务分工明确，很大程度上提升了开发效率和项目的可维护性，提升效率的同时也减少人力成本，该系统是 B/S 架构，而且是基于微信小程序的，很大程度上节省了使用成本。

3 技术栈介绍

3.1 开发环境

- 1、开发工具：微信开发者工具
- 2、数据库：MySQL
- 3、开发语言：ECMAScript6
- 4、前端技术：WXML、WXSS、ECMAScript6、JSON
- 5、后端技术：SSM（SpringMVC、Spring、MyBatis）
- 6、开发模式：RESTful

3.2 微信开发者工具介绍

为了帮助开发者简单和高效地开发和调试微信小程序，在原有的公众号网页调试工具的基础上，推出了全新的微信开发者工具，集成了公众号网页调试和小程序调试两种开发模式。

使用公众号网页调试，开发者可以调试微信网页授权和微信 JS-SDK 详情，使用小程序调试，开发者可以完成小程序的 API 和页面的开发调试、代码查看和编辑、小程序预览和发布等功能。

可以通过编辑区左边的模拟器，实时预览编辑的情况。修改 WXSS、WXML 文件，会刷新当前 page；修改 JS 文件或者 JSON 文件，会重新编译小程序。在代码编写过程中，开发工具提供自动补全功能。JS 文件编辑时，会帮助开发者补全所有的 API，并给出相关的注释解释；WXML 文件编辑时，会帮助开发者直接写出相关的标签；JSON 文件编辑时，会帮助开发者补全相关的配置，并给出实时的提示。

3.3 ECMAScript6 介绍

ECMA 是一个以评估、开发和认可电信和计算机标准为目标的组织。ECMA 的总部位于日内瓦，靠近其他的标准化组织，如国际标准化组织(ISO)和国际电子技术协会(IEC)。ECMA 是"欧洲计算机制造商协会"简称中国计算机制造的欧洲联盟。

1996 年 11 月，网景，JavaScript 的创造者，决定将其提交给 ECMA 脚本语言的国际标准化组织，有望成为国际标准。2011 年对 ECMAScript5.1 版发布后，就开始制定 6.0 版了。因此，ES6（ECMAScript6）这个词的原意，就是指 JavaScript 语言的下一个版本，它已经在 2015 年 6 月正式发布。

3.4 WXML 和 WXSS 介绍

WXML 与 HTML 非常相似，有标签、属性等构成。但是也有很多不一样的地方。

比如标签名字不一样 往往写 HTML 的时候，经常会用到的标签是 `div`、`p`、`span`，开发者在写一个页面的时候可以根据这些基础的标签组合出不一样的组件，例如日历、弹窗等等。

WXSS 具有 CSS 大部分的特性，小程序在 WXSS 也做了一些扩充和修改。新增了尺寸单位。在写 CSS 样式时，开发者需要考虑到手机设备的屏幕会有不同的宽度和设备像素比，采用一些技巧来换算一些像素单位。WXSS 在底层支持新的尺寸单位 `rpx`，开发者可以免去换算的烦恼，只要交给小程序底层来换算即可，由于换算采用的浮点数运算，所以运算结果会和预期结果有一点点偏差。

提供了全局的样式和局部样式，你可以写一个 `app.wxss` 作为全局样式，会作用于当前小程序的所有页面，局部页面样式 `page.wxss` 仅对当前页面生效。此外 WXSS 仅支持部分 CSS 选择器。

3.5 SpringMVC、Spring、MyBatis 介绍

Spring、SpringMVC、MyBatis 简称 ‘SSM’，该框架集由 Spring、SpringMVC、MyBatis 三个开源框架整合而成，并经常被用来作为一个数据源的 Web 项目的框架。

Spring 是由 Rod Johnson 创建的一个开源框架，它是一个开放源代码的设计层框架，解决的是业务逻辑层和其它各层的耦合性问题，因此 Spring 是将面向接口编程贯穿了整个系统应用。Spring 是在 2003 年火热起来的，作为一个轻量级的 Java 开发框架，它是独一无二的：

- 1、它和其它流行的框架定位的领域是有所不同的。Spring 致力于打造一种管理业务实体的方法。

- 2、Spring 是模块化的和全面的，这也就造就了 Spring 的分层体系架构，在使用的时候可以选择任何一个模块使用。

- 3、Spring 是一个由测试来驱动工程进行的框架，这样的设计可以很好的帮助开发人员编写易于测试的业务代码。

- 4、Spring 是一个潜在的一站式解决方案，它位于与典型应用程序相关的大多数基础设施中。

SpringMVC 是一个后续的 Spring 开源框架开发的产品，它已经被集成到 Web 流量。Spring 框架提供了一个用于构建 Web 程序的功能 MVC 模块。使用 Spring 的 MVC 架构可以插入，所以使用 Spring 进行 WEB 开发的时候，你可以选择使用 Spring 的 SpringMVC 框架或整合其他的 MVC 开发框架，如 Struts1(现在一般不使用)、Struts2 等。

Spring MVC 是一种典型的 MVC 架构，对于新手学习也是非常容易的。通过相关的策略接口，Spring 框架也是高度可配置的，包括多种视图技术。Spring MVC 框架不知道视图在使用什么，所以它不会强迫开发人员只使用 JSP 技术。Spring MVC 分离了

与下岗人员、过滤器和处理对象相关的控制模块和模型的角色，从而使得它们更容易达成个性化的定制。

MyBatis 是 Apache 的一个开放源代码的项目 iBatis,后来由 Apache 软件基金会迁移到 Google Code 上,并改名为 MyBatis,最后又被迁移到 GitHub 上。iBATIS 一词的来源是"internet"和"abatis"的组合,iBATIS 提供的持久层框架包括 SQL Maps 和 Data Access Objects(DAOs),是一个基于 Java 的持久层框架。MyBatis 是公认的一个非常流行的持久层框架,支持存储过程、先进的映射,和自定义的 SQL。MyBatis JDBC 代码优化,并减少了参数的设置和结果集的检索问题。MyBatis 可以直接使用简单的 XML 或注解来实现实体和数据表的映射,将接口和 Java 的 POJOS 映射成数据库中的数据。

3.6 RESTful 介绍

越来越多的人开始意识到网站是软件,它是一种新型的软件。“网络软件”基于分布式系统的客户机/服务器模型,通过网络通信具有高延时 (high latency)和高并发性的特点。

REST 这个词是 Roy Thomas Fielding 在他的 2000 年的博士论文中提出来的。Fielding 是一个非常重要的人物。他是 HTTP 协议(1 和 1.1 版)的主要设计者,他是 Apache 服务器软件的作者之一,也是 Apache 基金会的首任主席。因此,它一出版,他的论文就受到了关注,并对因特网的发展产生了深远的影响。

REST, 即 Representational State Transfer 的缩写,如果体系结构符合 REST 原则的话,就称它为 RESTful 架构, REST 的名称“表现层状态转化”中,省略了主语。“表现层”其实指的是“资源”(Resources)的“表现层”。

所谓“资源”,就是网络上的一个实体,也可以说是一个具体的信息。它可以是一个文本、一首歌、一幅画、一个服务,总之,它是一个具体存在。用户可以使用 URL(统一资源定位器)指向它,它对应于每个资源的特定 URL。要获取此资源,访问其 URL 就可以了,因此 URL 成为每个资源的地址或唯一标识符。所谓上网,就是与互联网上一系列的“资源”互动,调用它的 URL。

所谓表现层 (Representation), “资源”它是一个信息实体,可以有多种表现形式。具体描述的形式,称为资源的“表示层”。例如,文本资源可以用 TXT 格式表示,也可以用 HTML、XML、JSON 和其他格式表示。甚至可以使用二进制格式。图像可以用 JPG、PNG 和其他格式表示。URL 只代表资源的实体,不代表它的形式。严格地说,大部分网址最后的“.html”等后缀名是没有必要的,因为这种后缀名表示的是资源文件的格式,属于“表现层”的范畴,而 URL 应该只代表“资源”所处的位置。HTTP 请求的特定形式应在接受和内容类型字段中指定。这两个字段分别是“表示层”和“表示层”的描述。

所谓状态转化(State Transfer),访问一个网站的时候,客户端和服务端发生着互动。在这个过程中,势必涉及到数据和状态的改变。HTTP 协议是一个无状态的互联网通信协议。它的无状态意味着所有的状态都保存在服务器端。因此,如果客户端想操作服务器,就必须通过某种手段使服务器端“状态转换”。而这种转换是基于性能层的,“性能层的状态转换了”,这就是它的方式。客户端只使用 HTTP 协议。具体而言,它是 HTTP 协议中代表操作的四个动词:GET、POST、PUT 和 DELETE。它们对应四种基本操作:GET 用于获取资源,POST 用于创建新资源,PUT 并用于更新资源,DELETE 用于删除资源。

4 系统设计

4.1 系统架构

本系统采用 RESTful 架构，如图 4-1 所示：

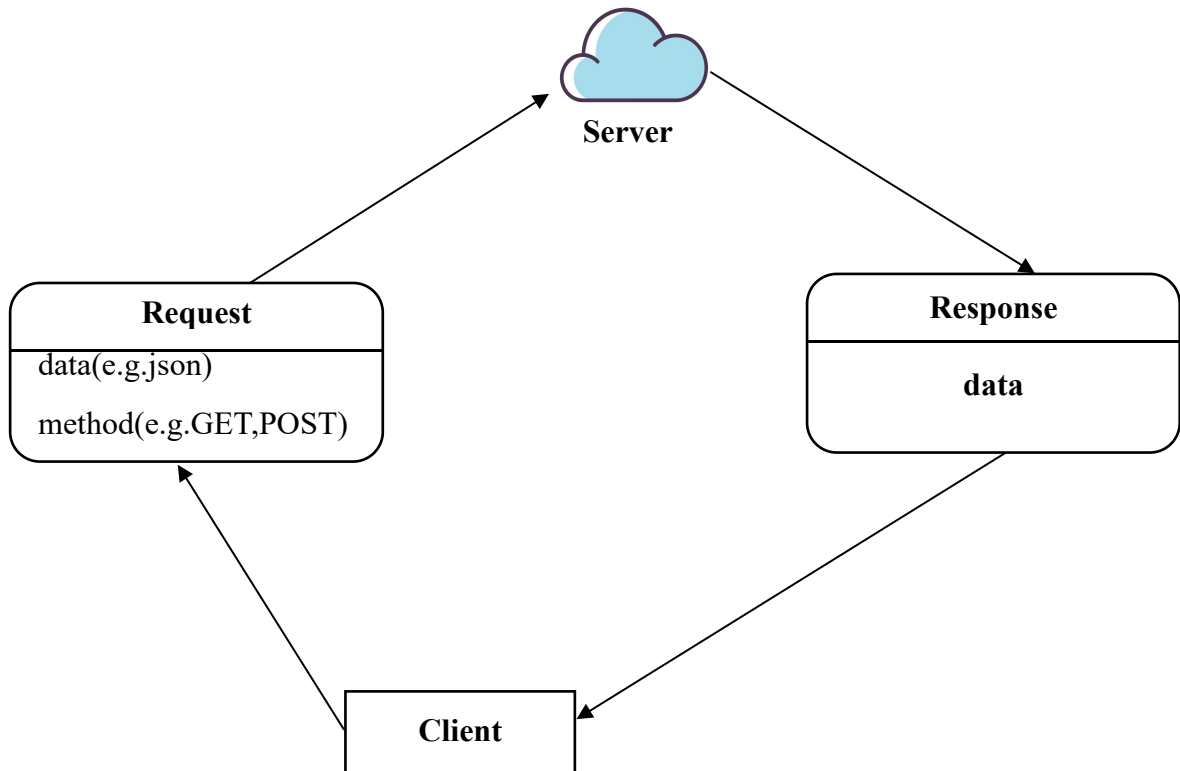


图 4-1 RESTful 架构图

如上图所示，服务端统一提供一套 RESTful API，Web 端、IOS 和 Android 可以共同调用这一套 API，各自的客户端都采用自己比较成熟的前端框架来帮开发者事半功倍。

在服务器端提供的 RESTful API 中，只有名词在 URL 中指定资源，原则上不使用动词。“资源”是 REST 架构或整个网络处理的核心。该系统的 API 定义如表 4-1 所示。

表 4-1 系统 API 表

接口	描述
/business/login	系统登录接口
/business/secClassRoom/event/bmEventListQry	查询活动列表接口
/business/secClassRoom/event/eventRegist	活动报名接口
/business/secClassRoom/clsEventListQry	查看预约活动接口
/business/secClassRoom/event/eventQDQTListQry	查询活动签到列表接口
/business/secClassRoom/event/eventQDQT	活动签退接口

/business/secClassRoom/event/eventQDQT	活动签到接口
/business/common/studentSQList	学分申请列表（审核中）接口
/business/common/endOrderList	学分申请列表（已完成）接口
/business/secClassRoom/approveQry	学分申请详情查看接口
/business/secClassRoom/event/clsEndXflistQry	学分明细查看接口
/business/secClassRoom/event/myEventListQry	活动审核列表接口
/business/common/homeTaskList	查询学生活动报名列表接口
/business/common/approve	单个活动审核接口
/business/common/batchApprove	批量活动审核接口

用 HTTP 协议里的动词来实现资源的添加, 修改, 删除等操作。即通过 HTTP 动词来实现资源的状态扭转:

GET 用来获取资源,

POST 用来新建资源,

PUT 用来更新资源,

DELETE 用来删除资源。

4.2 系统功能

第二课堂学分管理系统需要实现学生端和教师端两个模块, 学生端包括活动报名、活动预约查看、活动签到签退、学分申请查看、学分明细查看等功能。学生端功能图如图 4-2 所示。

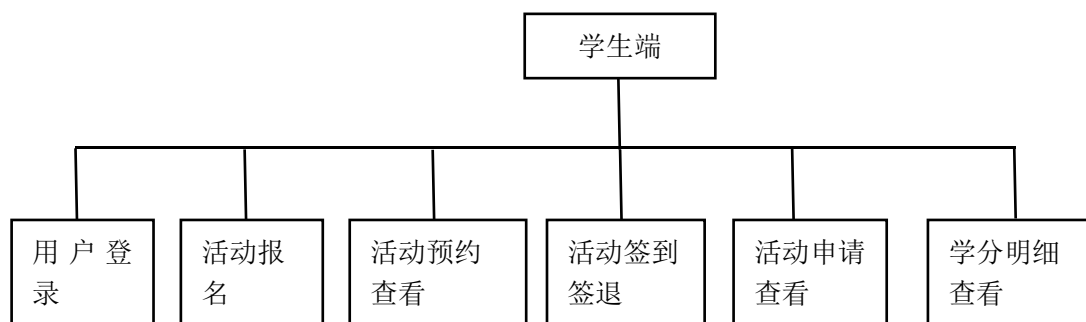


图 4-2 学生端功能图

教师端主要是活动的审核和活动开始和结束的时候发起定位信息的功能。学生在活动报名之后, 老师可以在线审核活动, 支持单个审核和批量审核, 审核通过之后, 在活动开始的时候, 需要教师在活动现场发起一个定位, 然后学生通过学生端进行活动签到, 根据老师定位的经纬度信息和学生定位的经纬度信息是否在一个范围内判断活动是否签到成功。在活动结束的时候, 老师需要在教师端再次发起一次定位, 根据老师定位的经纬度信息和学生定位的经纬度信息是否在一个范围内判断活动是否签退成功, 否则视为学生活动未签退。教师端功能图如图 4-3 所示。

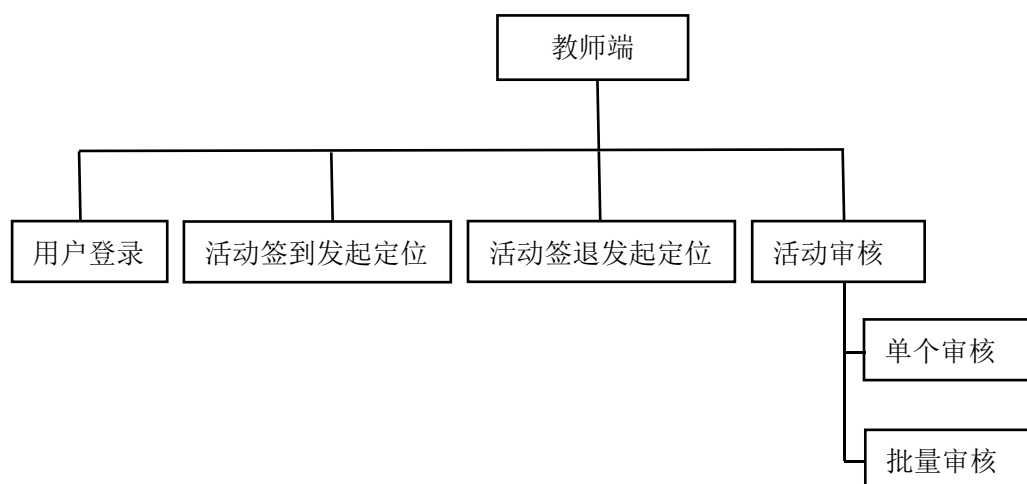


图 4-3 教师端功能图

4.3 业务流程

该系统需要电脑端和手机端结合才能实现一个完整业务流程，对于直接提交材料申请学分的，首先是学生提交申请，然后老师进行审核，审核不通过流程结束，进行下一次申请，审核成功，获得学分，本次流程结束。具体流程如图 4-4 所示。

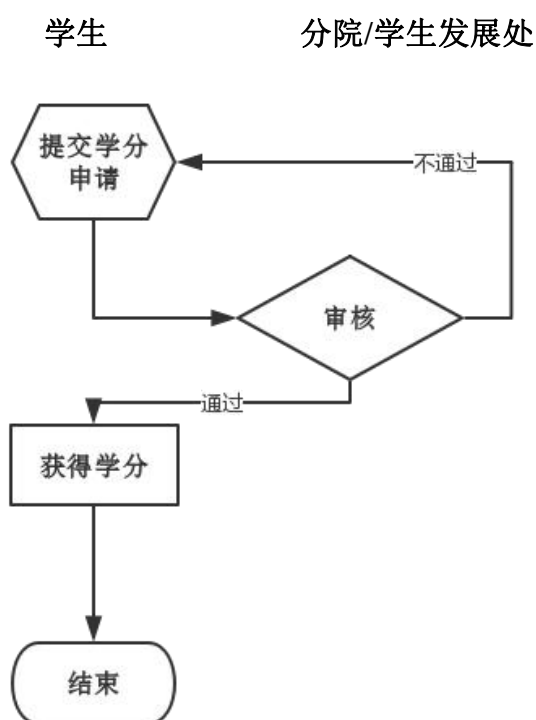


图 4-4 直接提交材料申请学分流程图

对于第二课堂活动发布及学生学分申请的业务流程首先是学生处老师或者分院老师在线发布一个活动，学生在电脑端或者手机端进行活动的报名，报完名后需要老

师审核，如果审核成功就参加活动，活动结束后不需要提交材料的，活动结束后直接获得学分，需要提交材料的，提交材料后审核成功才能获得学分。具体业务流程图如图 4-5 所示。

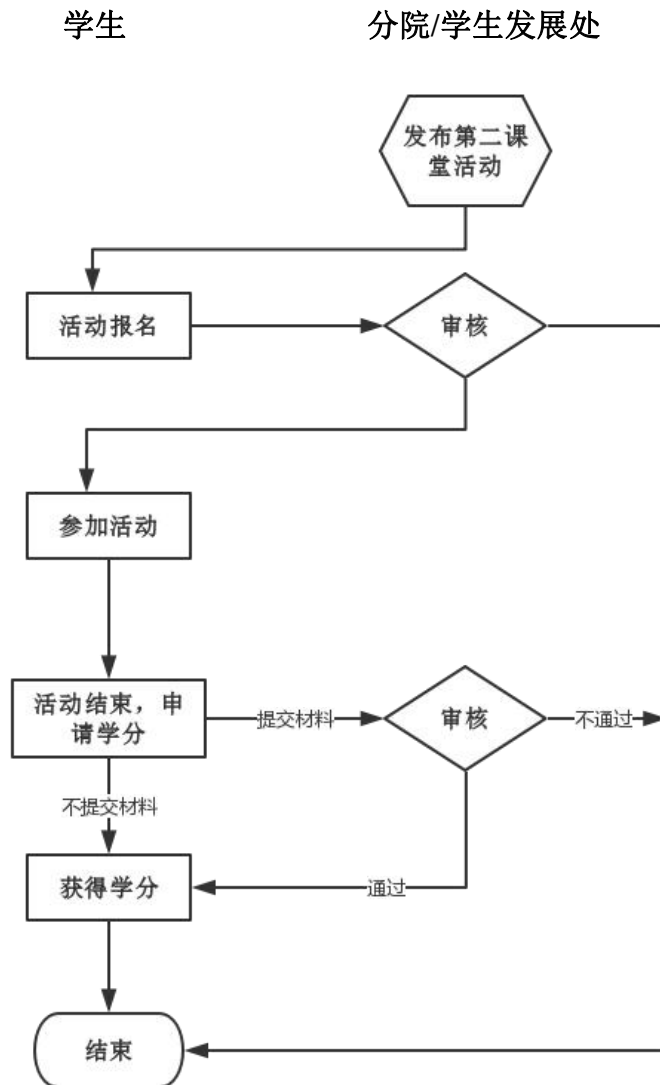


图 4-5 第二课堂活动发布及学生学分申请的业务流程

第二课堂学分管理系统主要的两大业务流程如上图所示，下面将对手机端的几个主要业务流程进行分析。

4.3.1 学生端登录模块

学生需要登录他们的学号和密码,输入用户名和密码,并在输入完成后点击登录按钮。系统前台会做非空校验,然后系统前台将请求“/business/login”资源,将用户名和密码发送给后台,后台会判断该用户是否存在、密码是否正确,如果用户不存在或者密码错误服务端会返回“-9999”状态码和错误的提示信息,如果用户名和密码正确,服务端会返回“0000”状态码,表示请求资源成功。如图 4-6 所示。

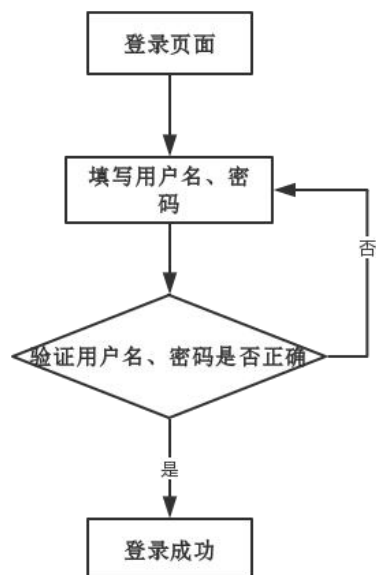


图 4-6 学生登录流程图

4.3.2 学生端活动报名模块

分院老师或者社区老师在电脑端发布活动，学生在手机端进行报名，如果已经报过名就不能重复报名，报完名审核成功后代表报活动名成功。具体流程如图 4-7 所示。

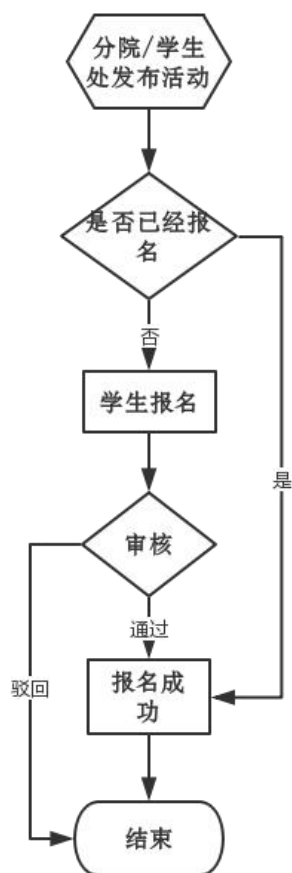


图 4-7 活动报名流程图

4.3.3 活动签到签退模块

在活动的开始需要老师在教师端发起一个活动的签到定位，然后学生才可以进行签到，学生端点击签到的时候，先判断学生定位是否在所规定范围之内，如果在范围之内，进行扫码签到，扫码成功，代表签到成功，否则签到失败。活动签退也需要老师在教师端发起一个签退的定位，学生在手机端点击签退，判断学生定位是否在所规定范围之内，如果在范围之内，签退成功，否则签退失败。具体流程如图 4-8 所示。

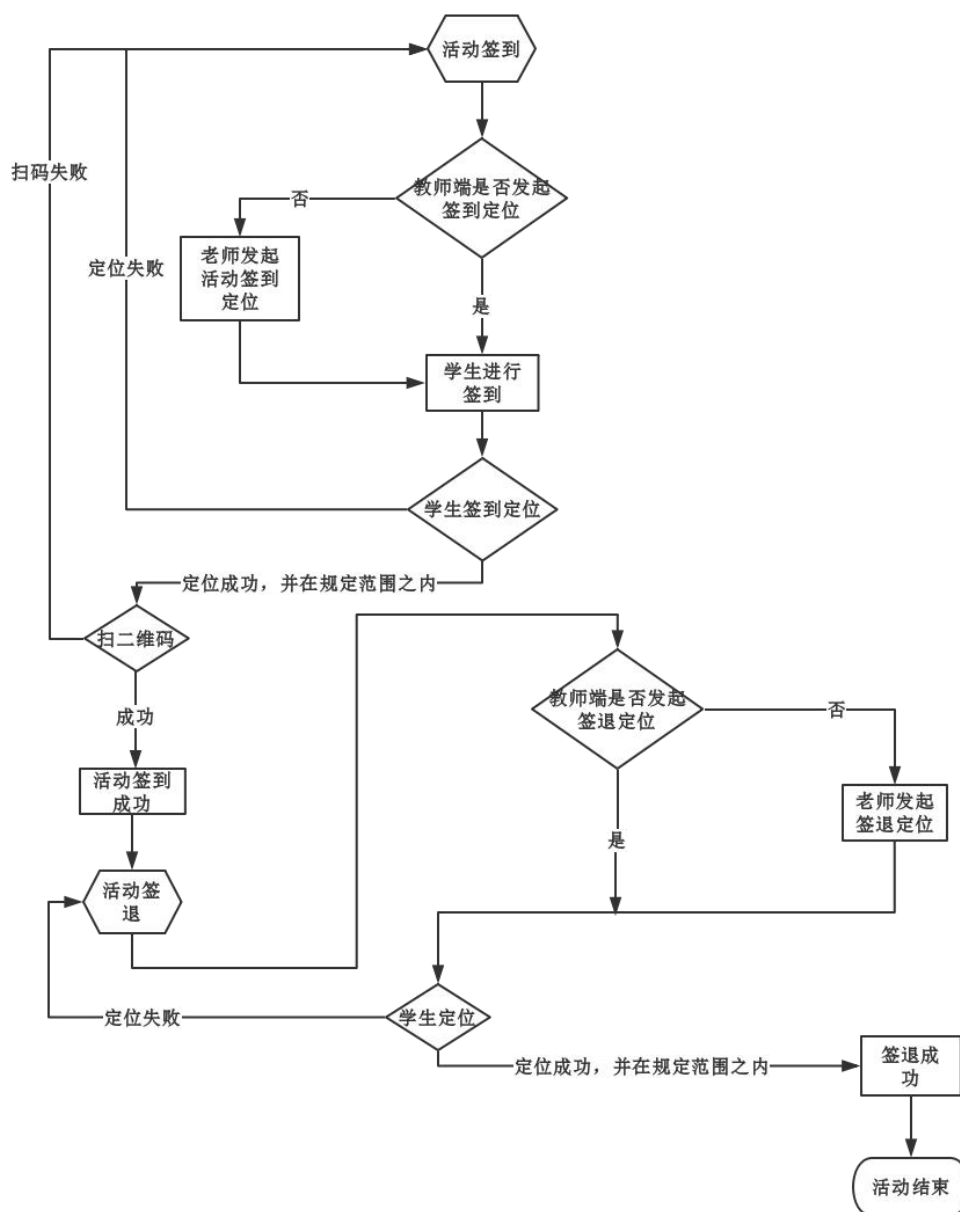


图 4-8 活动签到签退流程图

4.3.4 活动审核模块

活动审核属于教师端模块，主要对学生报名的活动进行审核，主要业务流程如图 4-9 所示。

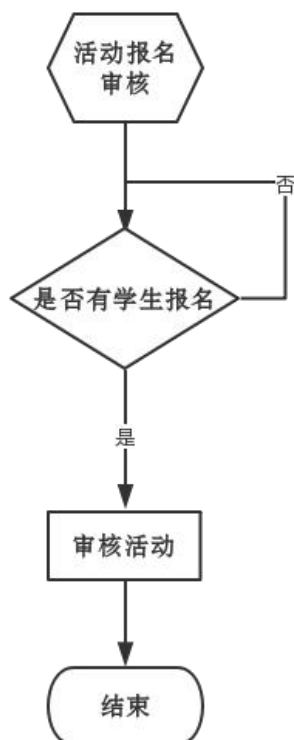


图 4-9 活动审核流程图

4.4 数据库设计

系统采用 MySQL 数据库,MySQL 是一个开源的关系数据库管理系统(RDBMS),使用最常用的数据库管理语言结构化查询语言(SQL)来管理数据库。

4.4.1 数据库概念设计

根据前期的需求分析和系统的流程设计，在该系统中表之间的关联关系比较复杂，所以在这只列举出以下几个实体。

1、用户实体图，如图 4-10 所示。

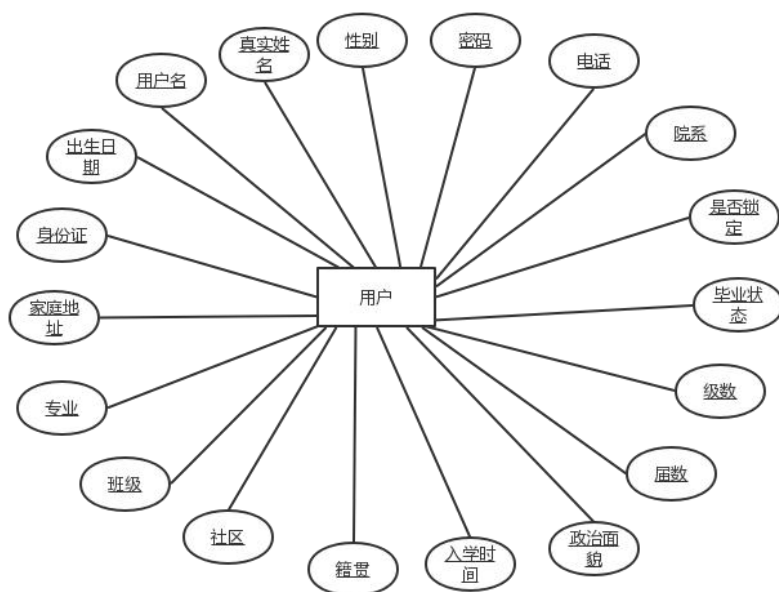


图 4-10 用户实体图

2、活动实体图，如图 4-11 所示。

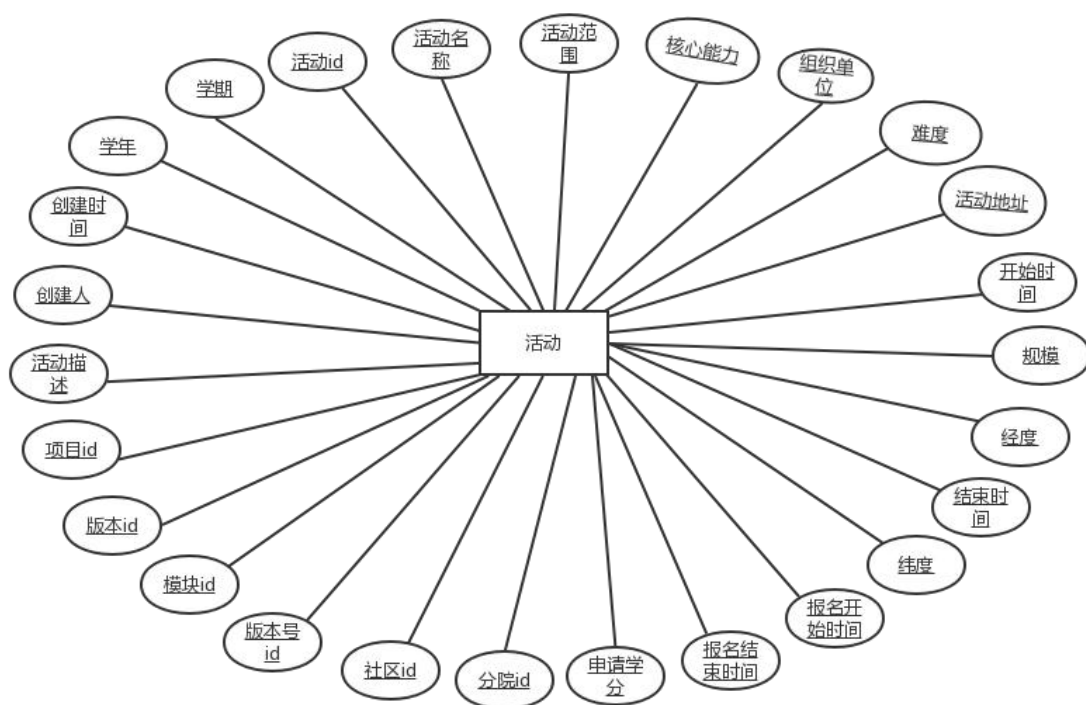


图 4-11 活动实体图

3、用户活动 E-R 图如图 4-12 所示。

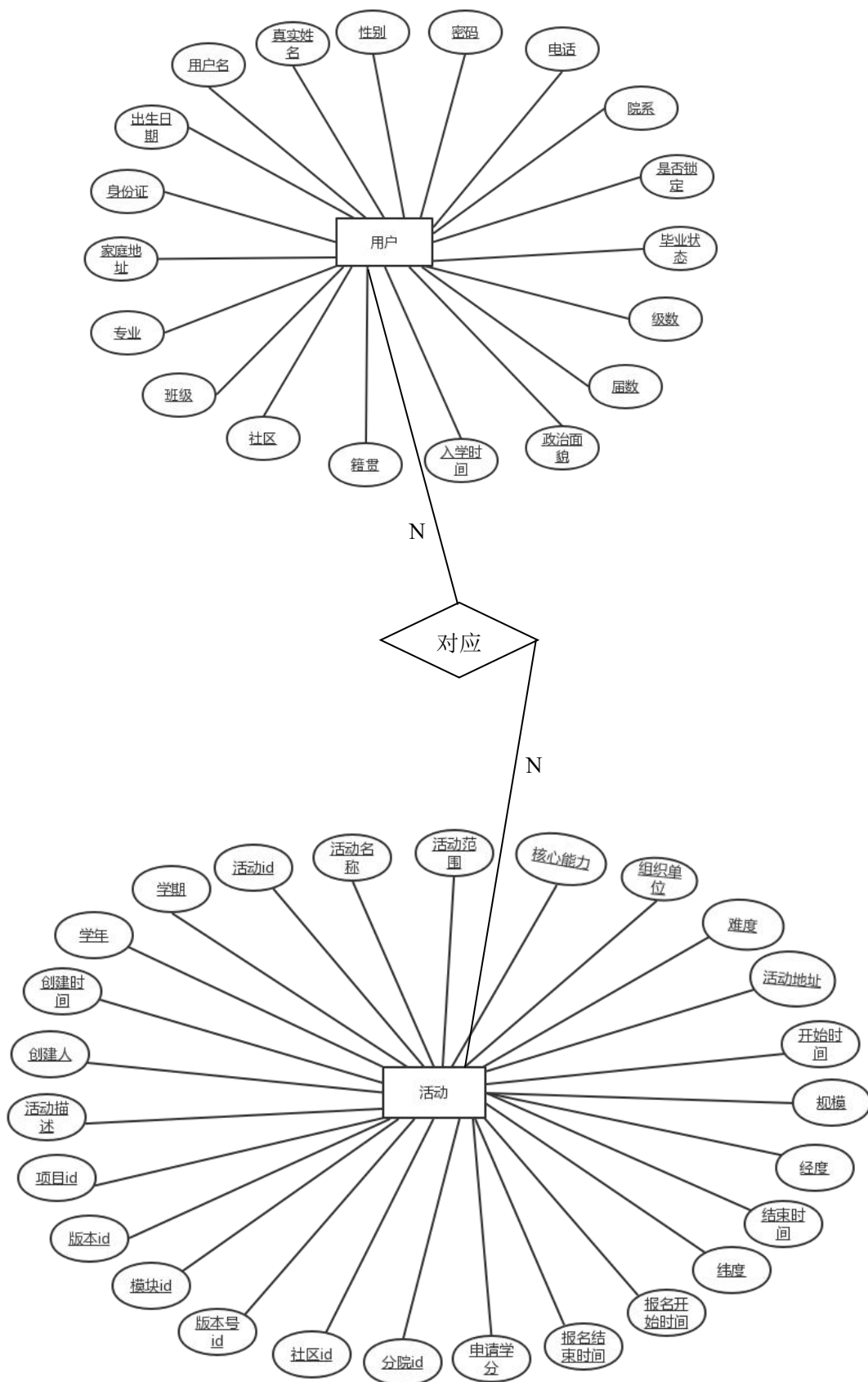


图 4-12 用户活动 E-R 图

4.4.2 数据库表结构的设计

1、bs_interface 接口配置表，如表 4-2 所示。

表 4-4 接口配置表

字段	数据类型	长度	描述
intf_id	bigint	20	id
intf_nm	varchar	20	接口名称
intf_cla_nm	varchar	40	接口处理类名称
intf_path	varchar	255	接口路径
rstfl_reqst_mth d_nm	varchar	40	接口请求方法名称 (GET,PUT,POST,D ELETE)
sys_sts_cd	varchar	10	接口状态: 0--不可用, 1--可用
crt_time	timestamp	0	创建时间
in_params	varchar	4000	请求参数示例
out_params	varchar	4000	返回参数示例

2、flow_secclassroom 学分申请流程记录表，如表 4-3 所示。

表 4-3 学分申请流程记录表

字段	数据类型	长度	描述
id	bigint	20	id
process_id	varchar	255	流程 Id
order_id	varchar	255	流程实例 ID
task_id	varchar	255	任务 id
mkId	varchar	50	模块 id
mk_name	varchar	50	模块名称
xmId	varchar	50	项目 id
xm_name	varchar	50	项目名称
jb_name	varchar	50	级别名称
start_xf	varchar	5	申请学分
end_xf	varchar	5	审批学分
cl_url	varchar	1000	学生上传材料路径
description	varchar	255	申请描述--便于快速 审核
operate_time	varchar	255	申请时间
operator	varchar	255	申请人

eventOperator	varchar	255	活动审核人
bossOperator	varchar	1000	初级审核人
endOperator	varchar	1000	终极审核人
version	varchar	20	申请版本--以学生入学年份为参考点
type	varchar	5	类型,0--普通学分申请, 1--活动学分申请
state	varchar	5	状态,start--开始, 0--活动同意, 1--活动驳回, 2--活动开始签到, 3--活动结束签退, 4--申请流程继续, end--流程结束
eventId	varchar	50	活动 id
xuenian	varchar	20	学年
xueqi	varchar	20	学期,1 第一学期,2 第二学期
juese	varchar	50	角色
fankui	varchar	255	反馈与建议

3、mk_version 模块版本关联表，如表 4-4 所示。

表 4-4 模块版本关联表

字段	数据类型	长度	描述
vId	varchar	40	版本 id
mkId	bigint	20	模块 id

4、process_operator 流程审核人关联表，如表 4-5 所示。

表 4-5 流程审核人关联表

字段	数据类型	长度	描述
processId	varchar	50	流程 id
roleIds	varchar	255	角色 ids,多个用逗号隔开
operatorName	varchar	20	审核人代码
desc	varchar	255	描述

5、sec_cls_event 学生活动表，如表 4-6 所示。

表 4-6 学生活动表

字段	数据类型	长度	描述
eventId	bigint	20	活动 id

evenName	varchar	200	活动名称
evenType	varchar	4	活动范围 1-分院，2-全院，3-学生处
startTime	varchar	30	开始时间
endTime	varchar	30	结束时间
bmStartTime	varchar	30	报名开始时间
bmEndTime	varchar	30	报名结束时间
startXf	varchar	10	申请学分
fyId	varchar	20	分院 id
shequId	varchar	50	社区 Id
qyId	varchar	50	全院中的分院 id,用逗号分割
vId	varchar	20	版本号 id
mkId	varchar	20	模块 id
xmlId	varchar	20	项目 Id
jbId	varchar	20	级别 id
eventDesc	varchar	1000	活动描述
crtTime	varchar	30	创建时间
operator	varchar	30	发布人
state	varchar	2	0-禁用,1-活动发布,2-活动结束
xuenian	varchar	20	学年
xueqi	varchar	20	学期
guimo	varchar	50	规模
hexinnl	varchar	50	核心能力
zuzhidw	varchar	50	组织单位
nandu	varchar	50	难度
lat	varchar	50	纬度
log	varchar	255	经度
address	varchar	255	活动地址
isHaveFile	varchar	5	是否需要提交附件。 1=是，0=否
eventFileUrls	varchar	1000	活动附件路径
isNotic	varchar	5	是否有通知。1=是， 0=否
noticStartTime	varchar	255	通知-开始时间

noticEndTime	varchar	255	通知-结束时间
noticAddress	varchar	255	通知-活动地点
noticDesc	varchar	1000	通知-其他描述

6、sec_cls_rdrule 学分规则字典表，如表 4-7 所示。

表 4-7 学分规则字典表

字段	数据类型	长度	描述
ruleId	bigint	20	规则 id
ruleName	varchar	40	规则名称
ruleDesc	varchar	1000	规则描述

7、sys_log 日志表，如表 4-8 所示。

表 4-8 日志表

字段	数据类型	长度	描述
log_id	bigint	20	id
user	varchar	255	用户
method_name	varchar	255	方法名称
method_description	varchar	4000	方法描述
request_ip	varchar	255	请求 ip
create_time	datetime	0	创建时间

8、version 学分规则字典表，如表 4-9 所示。

表 4-9 学分规则字典表

字段	数据类型	长度	描述
vId	varchar	40	版本号 Id
version	varchar	20	版本
fyId	varchar	40	分院 id

9、xf_zd_jb 学分级别配置表，如表 4-10 所示。

表 4-10 学分级别配置表

字段	数据类型	长度	描述
jb_id	bigint	20	级别 id
jb_name	varchar	40	级别名称
xm_id	bigint	20	项目 id

10、xf_zd_mk 学分模块配置表，如表 4-11 所示。

表 4-11 学分模块配置表

字段	数据类型	长度	描述
----	------	----	----

mk_id	bigint	20	模块 id
mk_name	varchar	40	模块名称
mkType	varchar	1	模块类型, 1--学科竞赛, 2--就业创业, 3--基本素养

11、xf_zd_xm 学分项目配置表, 如表 4-12 所示。

表 4-12 学分项目配置表

字段	数据类型	长度	描述
xm_id	bigint	20	项目 id
xm_name	varchar	40	项目名称
mk_id	bigint	20	模块 id
xm_type	varchar	1	项目类型 0:普通, 1:特殊
xm_ruleId	varchar	255	项目规则 Id
isEvent	varchar	4	是否活动, 0-否, 1-是

12、xf_zd_yq 学分要求配置表, 如表 4-13 所示。

表 4-13 学分要求配置表

字段	数据类型	长度	描述
yq_id	bigint	20	要求 id
jb_id	bigint	20	级别 id
yq_xf	varchar	5	要求学分

13、wf_cc_order 流程抄送表, 如表 4-14 所示。

表 4-14 流程抄送表

字段	数据类型	长度	描述
order_Id	varchar	32	流程实例 ID
actor_Id	varchar	100	参与者 ID
status	tinyint	1	状态
creator	varchar	50	发起人
create_Time	varchar	50	创建时间
finish_Time	varchar	50	完成时间

14、wf_hist_order 流程历史表, 如表 4-15 所示。

表 4-15 流程历史实例表

字段	数据类型	长度	描述
id	varchar	32	主键 ID

process_Id	varchar	32	流程定义 ID
order_State	tinyint	1	状态
creator	varchar	100	发起人
create_Time	varchar	50	创建时间
end_Time	varchar	50	完成时间
expire_Time	varchar	50	期望完成时间
priority	tinyint	1	优先级
parent_Id	varchar	32	父流程 ID
order_No	varchar	100	流程实例编号
variable	varchar	2000	附属变量 json 存储

15、wf_hist_task 流程历史任务表，如表 4-16 所示。

表 4-16 流程历史任务表

字段	数据类型	长度	描述
id	varchar	32	主键 ID
order_Id	varchar	32	流程实例 ID
task_Name	varchar	100	任务名称
display_Name	varchar	200	任务显示名称
task_Type	tinyint	1	任务类型
perform_Type	tinyint	1	参与类型
task_State	tinyint	1	任务状态
operator	varchar	100	任务处理人
create_Time	varchar	50	任务的创建时间
finish_Time	varchar	50	任务完成时间
expire_Time	varchar	50	任务期望完成时间
action_Url	varchar	200	任务处理 url
parent_Task_Id	varchar	32	父任务 ID
variable	varchar	2000	附属变量 json 存储

16、wf_hist_task_actor 流程历史任务处理人关联表，如表 4-17 所示。

表 4-17 流程历史任务处理人关联表

字段	数据类型	长度	描述
task_Id	varchar	32	任务 ID
actor_Id	varchar	100	参与者 ID

17、wf_task_actor 流程任务处理人关联表，如表 4-18 所示。

表 4-18 流程任务处理人关联表

字段	数据类型	长度	描述
task_Id	varchar	32	任务 ID
actor_Id	varchar	100	参与者 ID

18、wf_order 流程实例表，如表 4-19 所示。

表 4-19 流程实例表

字段	数据类型	长度	描述
id	varchar	32	主键 ID
parent_Id	varchar	32	父流程 ID
process_Id	varchar	32	流程定义 ID
creator	varchar	100	发起人
create_Time	varchar	50	发起时间
expire_Time	varchar	50	期望完成时间
last_Update_Time	varchar	50	上次更新时间
last_Updater	varchar	100	上次更新人
priority	tinyint	1	优先级
parent_Node_Name	varchar	100	父流程依赖的节点名称
order_No	varchar	100	流程实例编号
variable	varchar	2000	附属变量 json 存储
version	int	3	版本

19、wf_process 流程定义表，如表 4-20 所示。

表 4-20 流程定义表

字段	数据类型	长度	描述
id	varchar	32	主键 ID
name	varchar	100	流程名称
display_Name	varchar	200	流程显示名称
type	varchar	100	流程类型
instance_Url	varchar	200	实例 url
state	tinyint	1	流程是否可用
content	longblob	0	流程模型定义
version	int	2	版本
create_Time	varchar	50	创建时间

creator	varchar	50	创建人
---------	---------	----	-----

20、wf_task 流程任务表，如表 4-21 所示。

表 4-21 流程任务表

字段	数据类型	长度	描述
id	varchar	32	主键 ID
order_Id	varchar	32	流程实例 ID
task_Name	varchar	100	任务名称
display_Name	varchar	200	任务显示名称
task_Type	tinyint	1	任务类型
perform_Type	tinyint	1	参与类型
operator	varchar	100	任务处理人
create_Time	varchar	50	任务的创建时间
finish_Time	varchar	50	任务完成时间
expire_Time	varchar	50	任务期望完成时间
action_Url	varchar	200	任务处理的 url
parent_Task_Id	varchar	100	父任务 ID
variable	varchar	2000	附属变量 json 存储
version	tinyint	1	版本

21、role_permission 角色权限表，如表 4-22 所示。

表 4-22 角色权限表

字段	数据类型	长度	描述
role_id	bigint	20	角色 id
resource_id	bigint	20	资源 id
permission	varchar	100	权限

22、sys_organization 组织机构表，如表 4-23 所示。

表 4-23 组织机构表

字段	数据类型	长度	描述
id	bigint	20	主键 ID
name	varchar	100	名称
parent_id	bigint	20	父 id
parent_ids	varchar	100	父 ids
available	tinyint	1	是否可用
org_info_id	bigint	20	
class_id	varchar	50	班级 id

23、sys_resource 资源表，如表 4-24 所示。

表 4-24 资源表

字段	数据类型	长度	描述
id	bigint	20	主键 ID
name	varchar	100	名称
type	varchar	50	类型
url	varchar	200	地址
parent_id	bigint	20	父 id
parent_ids	varchar	100	父 ids
sn	varchar	100	简称
available	tinyint	1	是否可用
icon	varchar	50	图标

24、sys_role 角色表，如表 4-25 所示。

表 4-25 角色表

字段	数据类型	长度	描述
id	bigint	20	主键 ID
role	varchar	100	角色名称
description	bigint	20	角色描述
resource_ids	varchar	100	关联资源
available	tinyint	1	是否可用

25、sys_user_info 用户表，如表 4-26 所示。

表 4-26 用户表

字段	数据类型	长度	描述
id	bigint	20	主键 ID
username	varchar	100	用户名
realname	varchar	100	真实姓名
identityCard	varchar	50	身份证
sex	int	5	性别
password	varchar	100	密码
iphone	varchar	20	电话
department	bigint	100	院系
locked	tinyint	1	是否锁定
graduateStatus	tinyint	20	毕业状态
level	varchar	20	级数

years	varchar	20	届数
politicsStatus	tinyint	20	政治面貌
matriculationTime	varchar	200	入学时间
birthday	varchar	200	出生日期
nativePlace	varchar	255	籍贯
shequName	varchar	255	社区
className	varchar	255	班级
zyName	varchar	255	专业
address	varchar	255	家庭住址

26、sys_user_organization 用户组织机构关联表，如表 4-27 所示。

表 4-27 用户组织机构关联表

字段	数据类型	长度	描述
user_id	bigint	20	用户 id
org_id	bigint	20	组织机构 id

27、sys_user_roles 用户角色关联表，如表 4-28 所示。

表 4-28 用户角色关联表

字段	数据类型	长度	描述
user_id	bigint	20	用户 id
role_id	bigint	20	角色 id
sort	int	11	排序

5 系统实现

5.1 请求资源实现

由于该系统采用的 RESTful 架构，因此所有的客户端请求都属于对资源的操作。

以登录功能为例，当用户填写完账号、密码点击登录后，相当于客户端对 Server 发送了请求，请求代码如下：

```
wx.request({
  url: `${getApp().data.url}business/login`,
  method: 'POST',
  data: { "params":
    { "userName": this.data.username,
      "password": this.data.password }
    },
  header: {
    "Content-Type": "application/json"
  },
  success: function (res) {
    if (res.data.resultCode === '-9999') {
      console.log('网络异常')
    } else {
      if (res.data.object.resultCode === '-9999') {
        console.log('用户名或密码错误')
      } else {
        console.log('登录成功')
        wx.redirectTo({
          url: '../main/main',//跳转到主页
        })
      }
    }
  }
})
```

当 res.data.resultCode 返回的为 -9999 的时候，代表请求资源失败，res.data.object.resultCode 返回的为 -9999 的时候，代表业务处理失败，即‘用户名或密码错误’，否则代表业务处理成功。

5.2 学生端

5.2.1 登录页面

该页面有用户登录区域和帮助提示区域，用户可根据自己的角色，使用正确的账号登录，超级管理员可以在电脑端分配角色账号，本系统教师和学生端登录页面为同一个，根据账号角色的不同，进入不同的菜单，登录成功之后会跳进系统的主页，如果用户不清楚用什么账号登录，可点击页面上的帮助即可查看如图 5-1 所示。



图 5-1 登录页面

5.2.2 系统主页

在用户输入用户名和密码之后,点击立即登录，登录成功后，进入系统主页,页面由两部分组成,一部分是一张欢迎图和学校 Logo,另一部分是展示该系统的所有功能，主要功能有活动报名、活动预约查看、活动签到、学分申请查看、学分明细查看、退出等功能，如图 5-2 所示。



图 5-2 系统主页

5.2.3 活动报名

进入系统主页后点击【活动报名】进入活动报名页面，该页面展示所有的活动，会显示针对当前用户可以报名参加的活动列表，如果当前时间不在活动时间范围之内，那么该活动就不会在列表中显示，列表显示活动主题、活动时间、活动说明和报名按钮，点击【报名】进行活动的报名，报名成功后会显示已报名，如果你已经报名，就不能进行再次报名，系统会提示你已经报过名，请勿重复报名。

当该次活动有变更，管理员可在后台发布公告，学生端在活动列表就会显示有新的活动通知公告，如图 5-3 所示，点击可查看更改后的内容，显示活动具体的起止时间、地点和备注信息，如图 5-4 所示。



图 5-3 活动列表

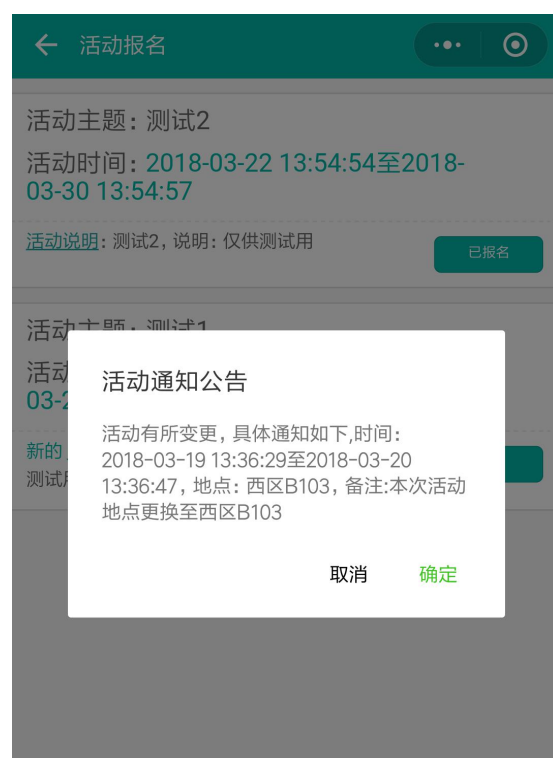


图 5-4 通知公告

5.2.4 活动预约查看

点击【活动预约查看】进入活动预约查看页面，该页面是以列表形式展示，顶部为一个 Tab 切换栏，可以查看正在进行的活动列表和已参加的活动列表，正在进行的活动就是已经报完名，管理员正在审核中的，已参加的活动就是管理员已经审核成功的。相对应的列表显示活动名称和活动状态，点击可查看详情，展示该活动的信息，模块、项目、级别等等，在活动没有开始之前，管理员发布的活动更改通知也会在列表中显示，如果有新的公告，会在列表显示一个小气泡，点击可查看更改后的活动内容信息，如图 5-5 所示。



图 5-5 活动预约查看

5.2.5 活动签到

点击【活动签到】进入活动签到页面，该页面是一个列表展示，显示该用户可以签到的活动列表，列表显示活动的标题，活动的开始时间和结束时间，还有一个签到按钮。活动签到是采用定位和扫二维码进行，在活动开始的时候需要教师发起一个活动地点的定位，然后学生端点击签到后先使用定位，根据学生定位点与教师定位点之间的距离小于一定的值来判定学生的签到，如果超出活动所规定的范围，则会定位失败，签到失败，如果定位成功之后，会弹出扫二维码的界面，扫码成功之后才意味着活动签到成功，签到成功之后按钮就会变成签退，签退的时候也需要教师发起一个定位，然后学生才可以进行签退，签退的时候只需要定位，只要在活动范围之内签退，都能签退成功，详细页面如图 5-6 所示。

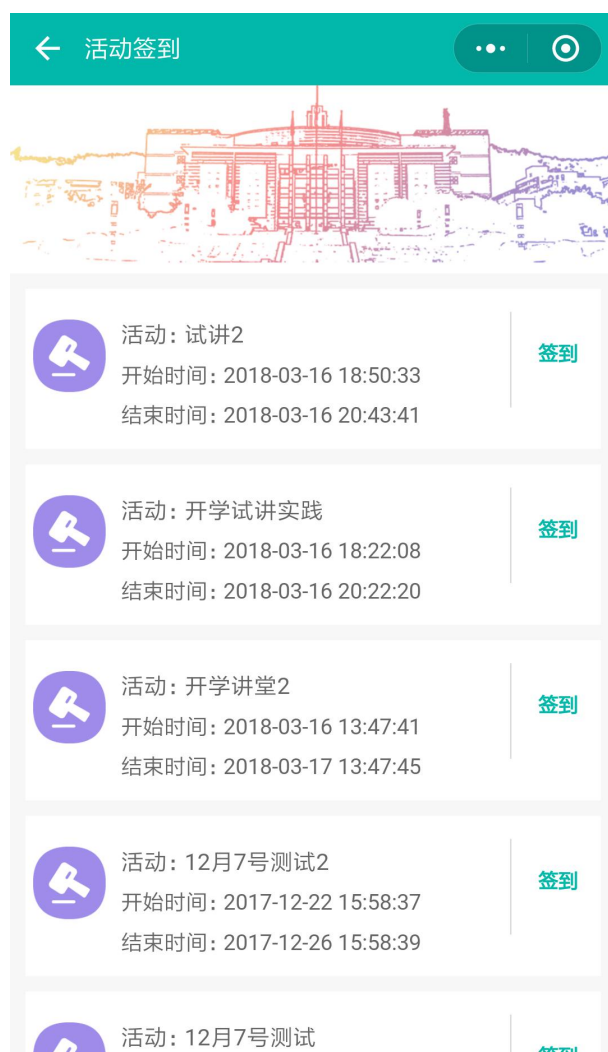


图 5-6 活动签到

5.2.6 学分申请查看

点击【学分申请查看】进入学分申请查看页面，学分申请的功能在本系统没有实现，在电脑端操作。该页面分为两部分，正在审核中的和审核完成的，包括学分申请的审核和活动报名的审核，也是以列表形式展示，列表展示学分或活动所在的模块，审核状态以及查看详情的按钮，如图 5-7 所示。点击查看详情即可查看学分或活动审核的流程，可以看到每一级审核的意见和状态，以一个流程树的方式展示，如图 5-8 所示。



图 5-7 学分审核列表



图 5-8 学分申请详情查看

5.2.7 学分明细查看

点击【学分明细查看】进入学分明细查看页面，该页面是以列表形式展示，顶部为一个 Tab 切换栏，可以查看分院学分和社区学分，分院的学分就是申请的第二课堂学分，社区的学分主要是学生报名参加活动后所得到的学分，列表主要展示活动的名称，点击可展开，展开后，可以查看所获得的学分，以及该学分所属的模块以及最后处理的时间，如图 5-9 所示。



图 5-9 学分明细查看页面

5.3 教师端

5.3.1 登录

教师端登录页面和学生端登录页面样式一致，登录成功后跳转到主页，如图 5-10 所示。



图 5-10 教师端登录

5.3.2 系统主页

对于教师端，教师使用自己的工号登录，登录成功后会跳转到系统的主页，教师端的主页和学生端的主页风格一致，只是显示的菜单项不同，只有一个管理员菜单项和退出功能如图 5-11 所示。



5-11 教师端系统主页

5.3.3 活动审核和定位

教师端进入系统主页后，点击【管理员】进入活动审核和定位页面，该页面主要有两个模块，分别活动的审核和活动的定位，如图 5-12 所示。

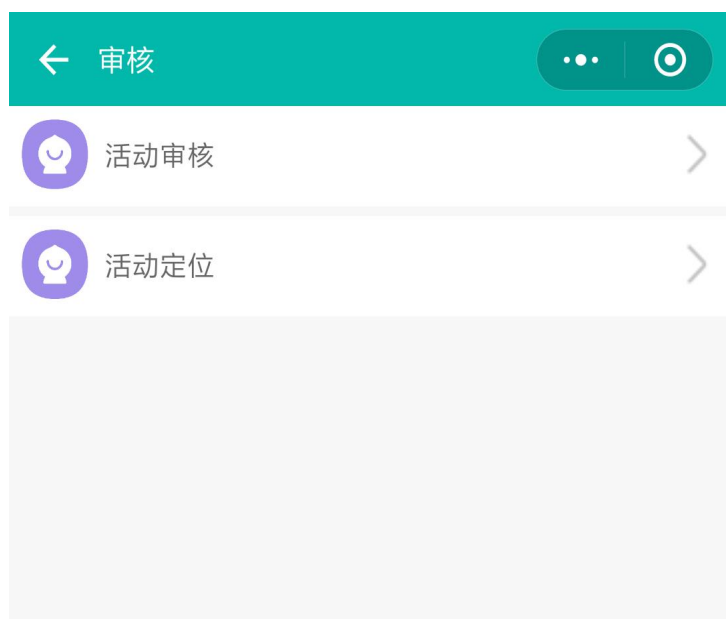


图 5-12 活动的审核和活动的定位页面

5.3.3.1 活动审核列表

点击【活动审核】进入活动审核列表，该页面展示所有的活动，老师可根据活动的名称去查看，如图 5-13 所示。

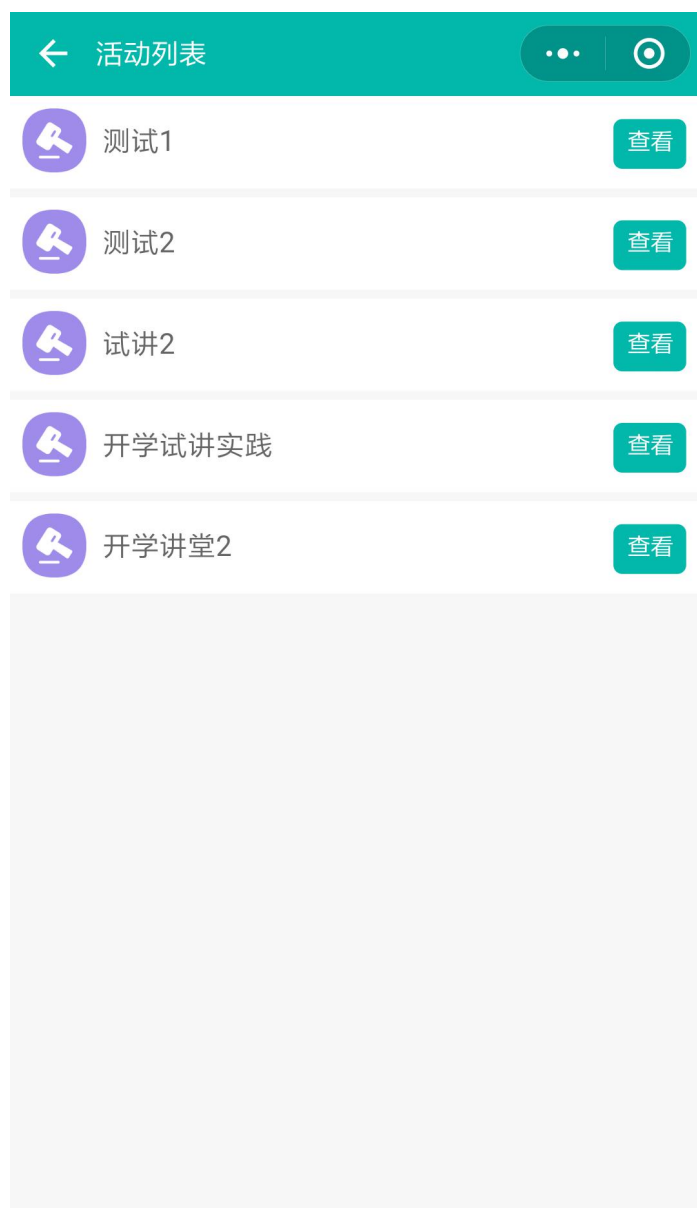


图 5-13 活动审核列表

在活动审核列表页面点击查看后，进入学生报名活动列表页面，在该页面对学生报名的活动进行审批，支持批量审核和单个审核，列表内容显示活动的名称，以及参加这个活动所对应学生的名称，如图 5-14 所示。

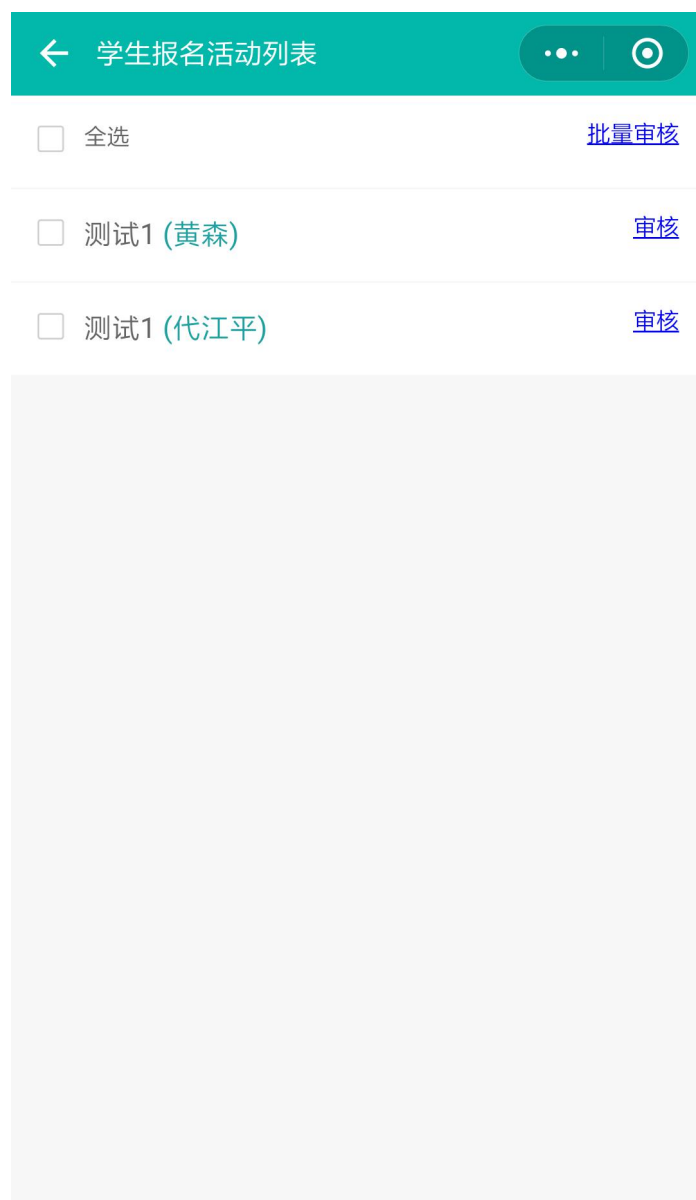


图 5-14 学生报名活动列表

在学生报名活动列表页面可以对已经报名的学生进行审核，如果要批量审核的话，可以勾选多个或者点击顶部全选按钮，选中所有列表，点击【批量审核】按钮，就会跳到批量审核页面，该页面会有审核意见输入框，勾选同意或驳回点击提交按钮即可对该批学生进行活动报名的审核，如图 5-15 所示，如果对单个的学生进行审核的话，点击对应学生列表后面所对应的审核按钮进行审核，点击【审核】后跳转到活动报名详情页面，显示活动的详细信息，如图 5-16 所示。



图 5-15 批量审核页面



图 5-16 单个审核页面

5.3.3.2 活动定位

点击【活动定位】进入活动定位管理页面，该列表显示所有已经发布的活动，列表显示活动的标题、活动的起止时间、活动说明，以及一个定位按钮和签退按钮，该页面主要的功能就是在活动开始的时候教师需要在这里点击与之对应的活动进行定位，定位成功之后学生才可以签到。签退按钮就是在活动结束的时候，点击与之对应的活动进行签退定位，定位成功之后，学生才能在手机端进行活动的签退。详细页面如图 5-17 所示。



图 5-17 活动定位管理页面

6 系统测试

为了提前预防系统的 BUG 出现，以及及时修复，系统测试也是一个重要的环节，通过系统测试可以及时发现软件在使用过程中会出现的问题，便于及时修复和完善，对软件质量起着至关重要的作用。

6.1 测试环境

处理器：2.5 GHz Intel Core i5

系统内存：8GB

操作系统：Windows

浏览器：Chrome 浏览器

数据库：MySQL

其他工具：微信开发者工具、IntelliJ IDEA

6.2 系统后端 API 测试

系统 API 测试使用 RestClient,专门用来测试接口的，也属于自动化测试的一种，可以自定义 URL,HTTP 头字段、请求方法。设置正文内容，即参数，点击发送按钮就会返回相应内容，以登录为例子,如果通过测试如图 6-1 所示



图 6-1 测试成功图

如果测试不通过，测试结果如图 6-2 所示：



图 6-2 测试失败图

6.3 系统功能模块测试

针对第二课堂学分管理系统中的各功能模块进行接口 A 测试，对比测试结果是否达到预期的结果，如果出现 BUG 应及时修复和完善。

（1）系统登录测试用例如表 6-1 所示

表 6-1 学生登录测试用例表

用例名称	第二课堂学分管理系统登录测试用例	
待测试模块	登录页面	
用例目的	测试输入账号和密码能否进行成功登录	
前提条件	服务器正常运行	
输入/动作	期望的输出/显示	测试结果
输入正确的账户和密码	系统提示登录成功，并跳转 到 首页。	测试成功
输入错误的账户和密码	系统提示用户名或密码错误提示	成功
输入账号不输入密码	系统提示请输入密码的提示	成功

（2）学生活动报名测试用例如表 6-2 所示

表 6-2 学生活动报名测试用例表

用例名称	学生活动报名测试用例	
待测试模块	活动报名页面	
用例目的	测试学生是否可以正常申请活动	
前提条件	服务器正常运行	
输入/动作	期望的输出/显示	测试结果
点击报名按钮	系统报名成功	测试成功

报名后再次点击报名按钮	系统提示已报名此活动，请勿重复报名	测试成功
当前日期不在活动日期之内	该条活动不在活动列表显示	测试成功

（3）活动签到测试用例如表 6-3 所示

表 6-3 活动签到测试用例表

用例名称	活动签到签退测试用例	
待测试模块	活动签到签退页面	
用例目的	测试是否能正常进行签到或签退	
前提条件	服务器正常运行	
输入/动作	期望的输出/显示	测试结果
老师在发起定位之后，学生在活动范围之内点击签到按钮	系统弹出扫二维码的界面，显示签到成功	测试成功
老师在发起定位之后，学生在活动范围以外点击签到按钮	系统弹出签到失败	测试成功
老师在发起签退定位之后，学生在活动范围之内点击签退按钮	系统提示签到成功	测试成功
老师在发起签退定位之后，学生在活动范围以外点击签退按钮	系统提示签到失败	测试成功

（4）活动公告测试用例如表 6-4 所示

表 6-4 活动公告测试用例表

用例名称	活动公告测试用例	
待测试模块	查看预约活动页面	
用例目的	测试是否能正常显示系统公告	
前提条件	服务器正常运行	
输入/动作	期望的输出/显示	测试结果
老师在发布活动之后，不进行通知公告的编辑	列表正常显示	测试成功
老师在发布活动之后，进行通知公告的编辑	列表左上角会出现一个红色小气泡	测试成功
老师在发布活动之后，进行通知公告的编辑，学生点击小气泡	系统提示活动公告的详细信息	测试成功

(5) 活动审核测试用例如表 6-5 所示

表 6-5 活动审核测试用例表

用例名称	活动审核测试用例	
待测试模块	学生报名活动列表页面	
用例目的	测试活动审核是否能成功	
前提条件	服务器正常运行	
输入/动作	期望的输出/显示	测试结果
点击全旋扭之后，再点击批量审核	跳转到批量审核页面	测试成功
不点击全旋扭，直接点击批量审核	系统弹出选择要操作的内容	测试成功
单个审核，点击审核按钮	跳转到审核详情页面	测试成功

7 结论

7.1 总结

本系统采用 SpringMVC、Spring、MyBatis 和微信小程序前端技术进行开发。开发遵循 RESTful API 的设计，后端提供数据接口，前端采用异步的形式请求后端资源。学生端完成了可以查看第二课堂活动信息、报名信息、学分等一系列相关信息，还可以实现在线活动申请、活动预约查看等功能，教师端完成了可以查看学生参加活动报名情况，可以对参加这些活动的同学进行活动报名的审核的功能。

在系统开发的过程中也遇到了不少的困难，比如对微信小程序的技术不熟悉，对一些概念理解不清等问题。通过不断的学习、研究，我也总结了以下几点作为经验积累：

- 1、学习一门新技术的时候，应该用思维导图工具列出需要学习的知识点，将大的概念拆成一个个小的技术点，逐一去攻克。不能一开始就把精通当成目标，应该在学习的过程当中不断摸索，记录所遇到的问题并将它们拆分开并逐一解决。

- 2、对于代码的管理应该是选择一个合适的版本控制管理工具，在第二课堂学分管理系统开发中使用 Git 版本管理工具，每次完成一个小功能的时候就要提交一次，避免因较大的错误导致一错到底的情况。当找到错误的时候可以进行版本回退，退回到上一次代码的提交，保证了每次提交的代码都是正确的。

- 3、系统的每一个功能都应该有相对应的接口测试，该系统使用 RestClient 进行接口的测试，保证每个功能的接口都能正常使用，遇到出错的接口，可以及时的调试，并给予解决和完善。

7.2 展望

该系统现在还处于测试阶段，很多地方还可以进行优化，学生端可以根据现实需求开发更多的功能，系统的性能还需要进一步的测试，之后可以尝试着用 wepy 小程序组件化框架来实现，支持组件化开发，支持加载外部 NPM 包，针对原生的 API 进行优化，希望自己可以在之后的日子当中抽出时间将该系统进一步完善，使他应用性更好、效果更佳。

致谢

毕业设计的开发和论文编写已接近尾声，在这里我要感谢在毕业设计过程当中帮助过我的老师和同学。

首先我要感谢我的导师张迪老师和任志宏老师，导师在我的毕业设计开题、任务书撰写、功能确定、中期论文修改等等每个环节都给予了很多帮助。导师工作认真负责、平易近人，在他的悉心指导下我也收获不少的学习方法和为人处世的道理。

同时要感谢信息工程学院的全体老师，老师们课堂上的教学激情洋溢，课堂下的教诲让我们受用一生。也要感谢工作室里的同学，在毕业设计过程中也给我了一些宝贵的意见，以及同学对于我的关心。

参考文献

- [1] 彭静. 论微信平台中的商标侵权行为及责任认定[J]. 法制博览, 2016
- [2] 李松佳, 徐飞. 基于微信公众号的高校微课平台的设计[J]. 价值工程, 2017
- [3] 贾士姣. 微信公众号第三方平台的发展现状和传播策略研究[D]. 郑州大学, 2016.
- [4] 李运莉, web 应用系统性能优化[M], 人民邮电出版社, 2016
- [5] 任杰, 徐树新. 内蒙古地区高校图书馆微信公众平台现状调查与分析[J]. 2014
- [6] 赖小芬. 微信公众平台的“互联网+”电力客户服务平台[J]. 农电管理, 2017.
- [7] 胡昊. 基于微信公众号的企业运营推广平台的设计与实现[D]. 北京交通大学, 2016.
- [8] 古丽敏. 餐饮服务类微信公众平台微交互设计研究[D]. 江南大学, 2016.
- [9] 马玲. 基于微信的大学英语微资源共享平台的开发[J]. 沈阳工程学院学报 (自然科学版), 2017,01:71-75.
- [10] 赖文渊. 微信时代下高校后勤服务系统开发运用[J]. 黑龙江科技信息, 2017,07:160.