

高校课堂微信扫码签到平台客户端设计研究

刘建芳, 马丽*, 张秋华

平顶山学院计算机学院, 河南 平顶山
Email: *maixnc@126.com

收稿日期: 2020年10月1日; 录用日期: 2020年10月16日; 发布日期: 2020年10月23日

摘要

针对目前高校课堂考勤存在的问题, 基于微信平台 and JavaEE, 采用移动设备作为签到终端, 设计课堂扫码签到平台。该平台基于B/S模式, 采用SSM五层架构, 设计签到平台架构、服务器端和客户端软件结构、数据以及关键问题解决方案。重点研究客户端微信公众平台对接、时效二维码生成、扫码、随机抽查、统计等关键模块的业务流程、类与接口以及功能的设计, 实现了高校课堂实时考勤、统计、查看等功能。经应用测试, 该平台考勤便捷、时效、高效, 较大程度杜绝了代签、伪签, 同时解决了其他课堂考勤软件存在的购买硬件设备、操作系统兼容、占用手机空间、开发时间长、维护成本高等问题, 具有一定的推广价值。

关键词

SSM, 高校课堂, 微信, 时效二维码, 扫码, 签到统计

Research on the Client Design of WeChat Scan Code Sign-In Platform in College Classroom

Jianfang Liu, Li Ma*, Qiu Hua Zhang

Pingdingshan University, Pingdingshan Henan
Email: *maixnc@126.com

Received: Oct. 1st, 2020; accepted: Oct. 16th, 2020; published: Oct. 23rd, 2020

Abstract

Aiming at the current problems of college classroom attendance, based on the WeChat platform and JavaEE, mobile devices are used as sign-in terminals to design a classroom scan code sign-in

*通讯作者。

platform. The platform is based on the B/S model and adopts the SSM five-tier architecture to design the sign-in platform architecture, server-side and client-side software structure, data, and key problem solutions. The focus is on the design of business processes, classes and interfaces and functions of key modules such as client WeChat public platform docking, time-effective QR code generation, code scanning, random check, statistics, etc., to achieve real-time attendance, statistics, and viewing functions in college classrooms. After application testing, the platform is convenient, time-efficient, and efficient attendance, which largely eliminates agent signing and counterfeiting, and at the same time solves problems in other classroom attendance softwares that exist in purchasing hardware equipment, operating system compatibility, occupying mobile phone space, development time and long and high maintenance costs, which has certain promotion value.

Keywords

SSM, College Classroom, WeChat, Time-Sensitive QR Code, Scan Code, Sign in Statistics

Copyright © 2020 by author(s) and Hans Publishers Inc.

This work is licensed under the Creative Commons Attribution International License (CC BY 4.0).

<http://creativecommons.org/licenses/by/4.0/>



Open Access

1. 引言

学风是一所大学的灵魂，学生课堂考勤工作是高校学风建设的根本。有效的课堂考勤，是保证高校教学秩序正常运行和各项集体活动顺利开展的重要手段。目前高校课堂考勤主要有：人工花名册点名、刷脸、刷卡、移动端 App 课堂等方式[1] [2] [3] [4]。人工点名考勤依然是高校课堂使用频率最高的方式，但占用课堂时间，且期末统计工作量较大；刷卡、刷脸考勤需投入大量的资金购买和维护软硬件设备且费时[5] [6] [7]；移动签到 APP 相比上述方式，签到、统计便捷，使教学管理更加科学和有效。但目前的移动签到 APP，存在下载安装软件占用手机空间、操作系统兼容和开发、更新成本高等问题[8] [9]。而微信是一款应用广泛的聊天软件，基于微信考勤，能较大程度避开上述考勤方式的弊端和不足。目前基于微信的签到软件多应用于举办活动，个别基于微信的课堂签到软件存在不能实时考勤、签到种类繁多、课堂签到操作繁琐等问题[10] [11] [12] [13] [14]。鉴于此，开发课堂微信扫码签到平台，优化高校考勤管理，提升课堂签到效率，提高教学质量。

2. 扫码签到平台功能需求分析

课堂微信扫码签到平台主要包括角色选择、微信授权、扫码签到、生成二维码、签到统计、随机抽查签到、管理员管理等业务。该平台分客户端(即微信移动端)和服务端。

移动端功能需求简要说明如下。

- 1) 微信公众平台对接。用户搜索微信公众号或扫码关注，成功后自动获取用户微信信息登录。
- 2) 身份选择。用户授权登录平台后，需先确认身份：教师或学生。
- 3) 个人信息设置。查看、设置个人信息。
- 4) 扫一扫签到。调用微信扫一扫，实现扫码签到。指定时间内不能重复签到。
- 5) 查看教学任务。教师登录后，可查看自己教学任务的详细信息。包含所教院系、班级、课程、课时、上课时间、地点等教学安排信息。
- 6) 统计缺勤情况。教师可在平台查看本节课程学生出勤及统计汇总情况。
- 7) 查看我的签到记录。学生可在平台直接查询本节签到信息。

8) 生成签到二维码。教师绑定个人信息后, 选择上课信息和教师信息生成随机时效二维码, 时效内不能重复签到, 超时将不能签到。

9) 签到统计。教师可查看每节、每天、每周、每学期课程签到情况。

10) 随机抽查。教师可按姓名当堂随机抽查学生签到并做相应处理。

服务器端功能主要是管理员对班级、课程、院系、用户、教学任务、签到等详细信息进行增、删、改、查、停用或启用等管理。并以报表形式统计每班、每学期每月每周签到情况。

3. 校园微信扫码签到平台设计

该签到平台开发基于 JavaEE, 采用 MVC 模式, 使用 SSM 框架处理业务、数据交互等, 数据存取采用 MySQL, 界面设计采用 MUI 框架和 Layer 插件, 使用 JavaScript、JQuery 和 Ajax 实现前后台交互。

3.1. 校园签到平台架构设计

该平台采用 SSM 框架五层架构: 数据层(Model 层)、持久层(Dao 层)、业务层(Service 层)、控制器层(Controller 层)和视图层(View 层)。Model 层映射数据表; Dao 层用来处理数据, 负责与数据库的交互; Service 层主要调用接口层的公共方法; Controller 层主要调用服务层的接口来控制业务流程。平台架构图如图 1 所示。该平台基于微信公众号开发, 用户通过自定义的安全域名, 绑定微信授权地址和回调地址请求微信服务器端, 微信端验证成功后发送响应信息给用户。

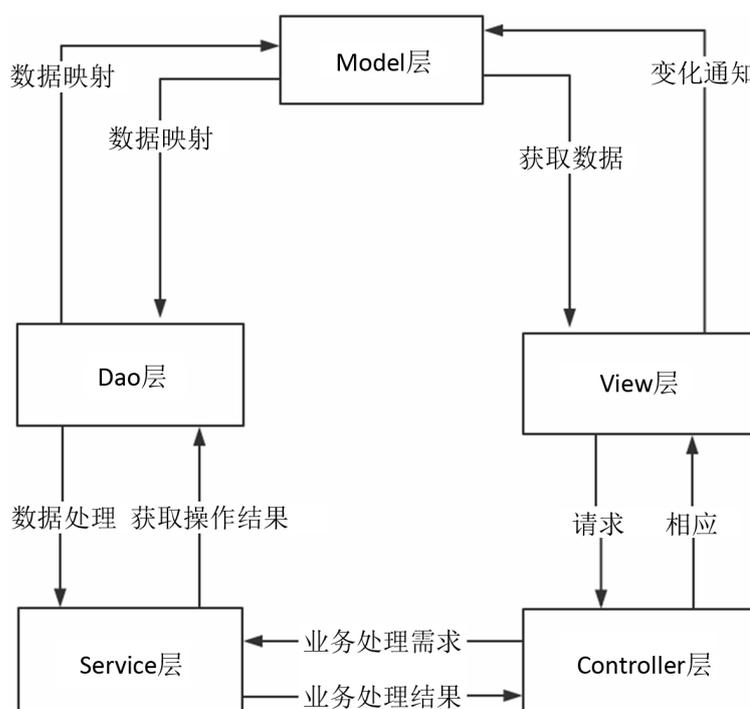


Figure 1. Platform architecture of Scanning code check-in

图 1. 扫码签到平台架构

3.2. 校园签到平台软件结构设计

根据上节平台的功能需求分析, 该平台分移动端和服务端, 根据“高内聚、低耦合”原则, 设计为基础、学生、教师、管理员等模块。该平台客户端、服务器端软件结构设计分别见图 2~3。

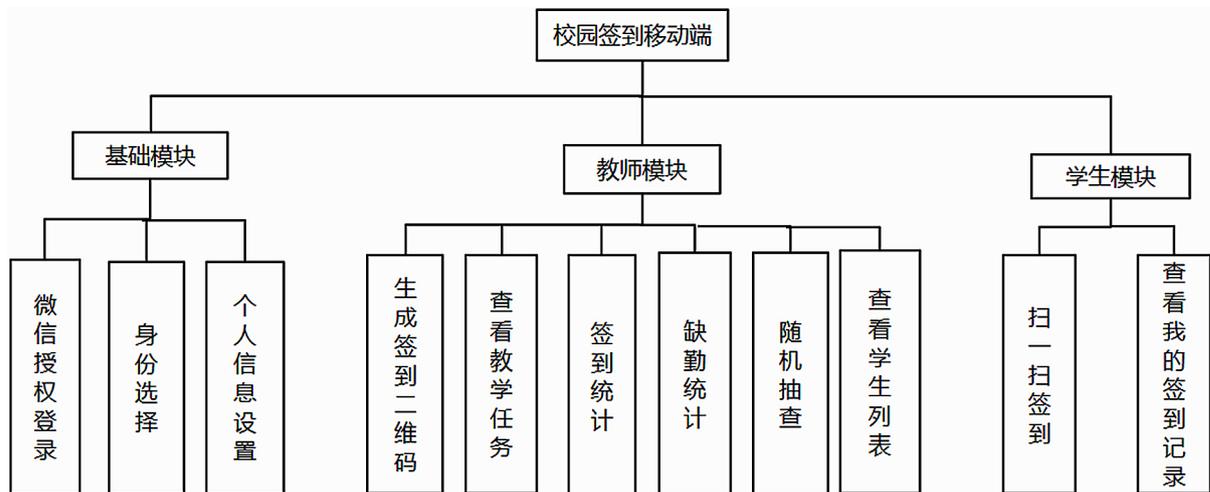


Figure 2. Software structure of client

图 2. 客户端软件结构

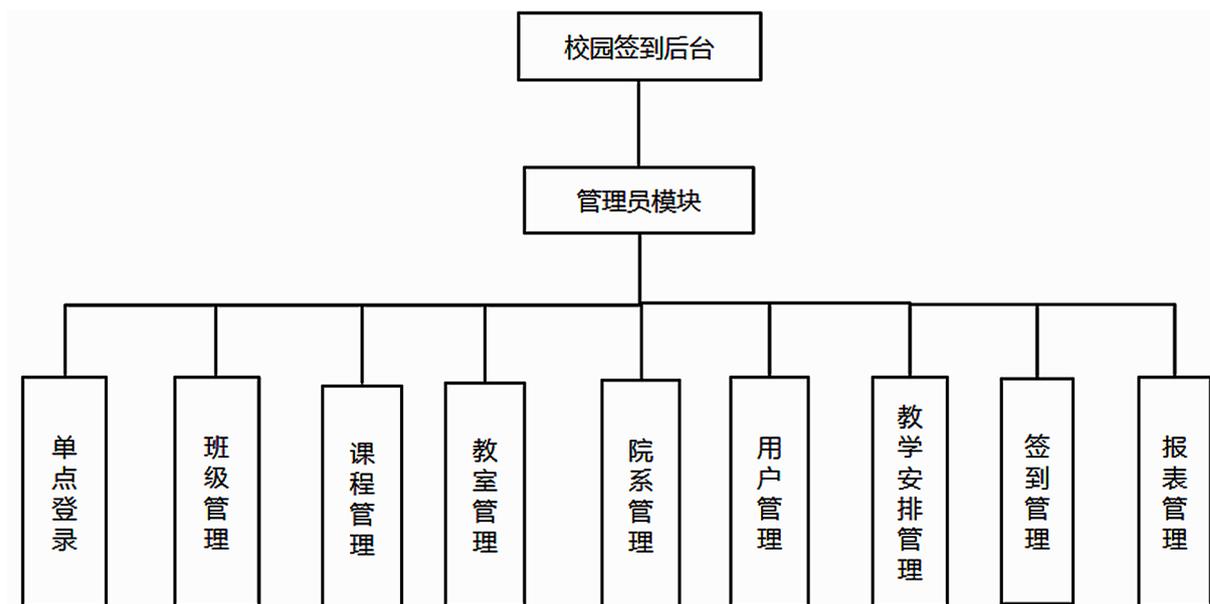


Figure 3. Software architecture of server

图 3. 服务器端软件结构

3.3. 数据设计

依据平台的需求，该平台主要包括用户、管理员、学生、教师、课程、班级、教室、二维码、院系和教学任务等实体。

用户实体属性：用户 ID、openId、昵称、用户头像、普通用户类型、用户生成时间、用户更新时间等；管理员实体属性：管理员 ID、登录名、登录密码、生成时间、更新时间；学生实体属性：学生 ID、用户 Id、班级 Id、学生名、学号、年级、院系、生成时间、更新时间等；班级实体属性：班级 ID、班级名称、班级生成时间、班级更新时间等；院系实体属性：院系 Id、班院系名称、生成时间和更新时间等；教师实体属性：教师 ID、用户 Id、教师姓名、教师年龄、电话、邮箱、生成时间、更新时间等；课程实体属性：课程 ID、课程名、课时、状态、生成时间、更新时间等；教室实体属性：教室 ID、教室名称、

教室地点、状态、生成时间、更新时间等；签到二维码实体属性：教师 Id、学生 Id、课程 Id、班级 Id、教学安排 Id 和生成时间等；教学任务实体属性：教学任务 Id、班级 Id、课程 Id、教室 Id、用户 Id、生成时间和更新时间等。

根据平台需求，上述实体间的联系见图 4。

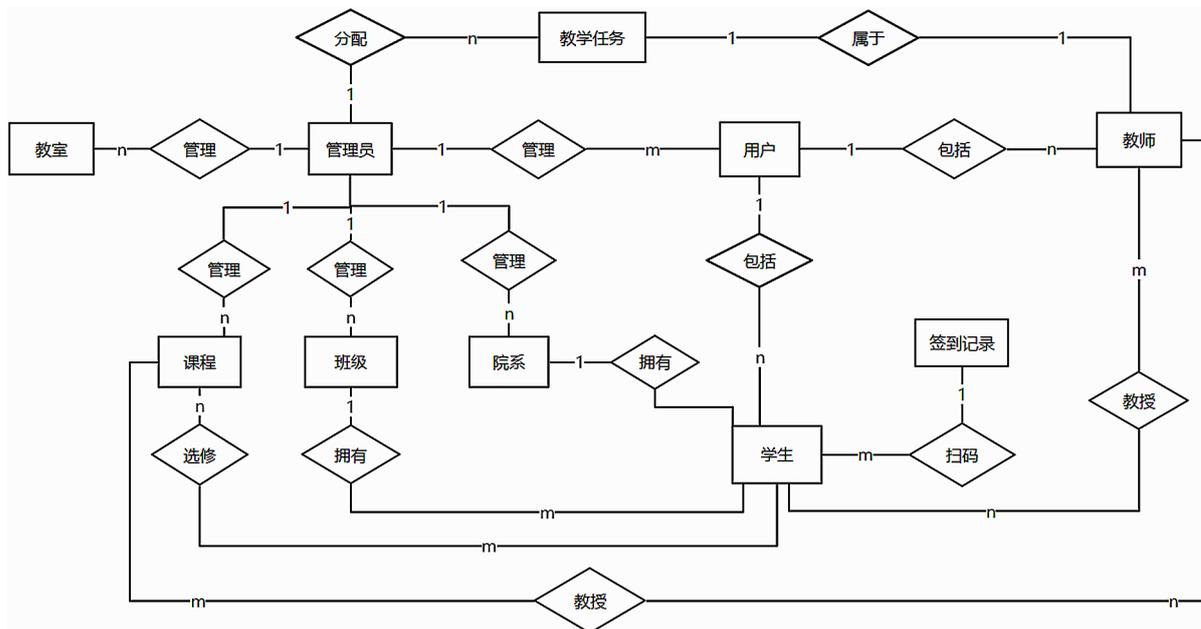


Figure 4. E-R diagram of the system
图 4. 系统 E-R 图

签到平台使用 MySQL 数据库存储数据，依据上面分析，在数据库中设计 user (用户表)、admin (管理员表)、student (学生表)、teacher (教师表)、class (班级表)、croom (教室表)、course (课程表)，arrange (教学安排表)，sign (签到表)等表。

3.4. 关键问题设计方案

本节将描述身份精确定位、自动获取周次和生成具有时效性的二维码等问题的设计思路。

1) 身份精确定位设计方案

用户首次进入平台个人中心需先确定身份，以便再次进入该平台时可以自动获取用户信息。用户授权登录成功，选择身份后，平台将该身份数据传入服务器端。服务器端分析传入的身份数据 1 或 0，进而确定身份信息，分析成功后入库。用户身份选择成功后，进入个人中心时，平台根据已录入的身份类型进行分析，自动进入到对应的个人中心页面。若用户再次进入个人中心，身份已确定。当用户处于签到公众号平台时，移动端可自动获取用户微信信息。

2) 自动获取周次设计方案

教师可在微信端查看任意周次签到记录。教师按授课时间选择周次，平台首先获取到事先录入到平台的开学时间，然后再调用日期接口获取当前时间戳，二者相减，得到时间差。将相减的结果向上取整后对 7 整除，动态获取本周及以前的周次(不会获取到超过本周的周次)，并在一级选择器中自动显示。

3) 生成具有时效性的二维码设计方案

生成教师拥有的时效性二维码，以便学生扫码签到。教师选择所教院系、班级、本节课程、地点等

信息生成实时的签到二维码，同时记录二维码生成时间。平台获取当前时间戳和生成二维码时间戳，二者相减，差值进行时间换算后，对 1 小时整除，得到有效的分钟数(比如 10 分钟)。二维码的时效性为 5 分钟，学生需在 5 分钟内扫码签到，否则，二维码无效，不能再进行签到。

4) 随机抽查设计方案

教师随机抽取课堂学生签到情况，核实学生是否代签，避免统计失误。平台根据教师、上课班级、课程名、地点等信息，调用随机函数随机显示学生姓名。教师点击停止菜单，口头核实界面显示学生签到信息，若有误，则会记为旷课并入库。

3.5. 客户端关键模块设计

客户端包括基础模块、普通用户模块。基础模块为用户共有操作，该模块设置微信公众平台对接、身份选择、个人信息设置三个子模块；普通用户模块包括教师和学生子模块。限于篇幅，本节只描述客户端关键模块的设计。

3.5.1. 微信公众平台对接模块的设计

对接模块连接平台与微信端，包括微信授权、身份选择和信息审核等功能。微信授权即基于微信公众平台获取用户基本信息。用户扫码关注公众号后，自动进入微信用户授权界面，待用户同意且选择身份后，平台自动获取保存该用户的微信信息。微信授权登录流程设计如图 5 所示。

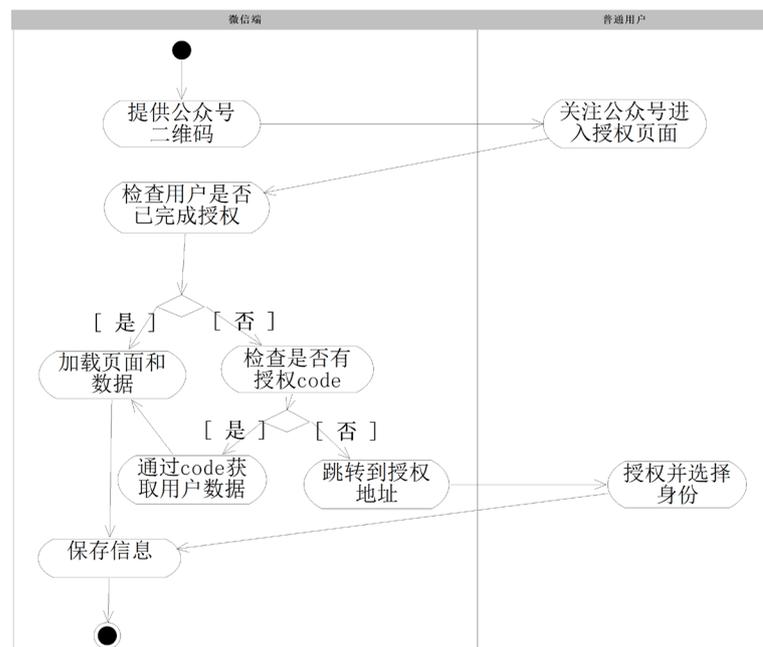


Figure 5. WeChat authorization process

图 5. 微信授权流程

1、类与接口设计

主要用到的类和接口有实体类 User、接口类 UserDao、用户业务逻辑类、用户信息控制类。

- 1) 微信对接控制类(UserController): 获取保存已关注公众号的微信用户信息;
- 2) 微信对接 Service 类(UserService): 处理微信对接的相关业务逻辑并调用 UserDao 层的方法;
- 3) 微信对接 Dao 类(UserDao): 负责数据库中微信用户信息的操作。

其他模块的 Service 类、Dao 类功能与此模块类似，后面不再描述，只描述控制类的设计。

2、微信授权功能设计

1) 用户关注公众号进入微信授权页, 平台返回授权地址, 用户确认授权登录后, 数据传到服务端请求 UseController 类。在控制层中调用 UseService 中的 getCode() 方法, 此方法调用接口 UserDao 中的 getCode() 方法, 平台根据该用户的 appId 和 secret 获取授权码 code。平台根据获取的授权 code 请求 UseController。在控制层中调用 UsrService 中的 getToken(String code) 方法, 调用接口 UsrDao 中的 getToken(String code) 方法, 获得授权登录 token。

2) 通过上步骤获得的 token 请求 UsrController, 调用 UseService 类中的 getUserInfo(String token) 方法, 此方法调用接口 UsrDao 类中的 getUserInfo(String token) 方法得到用户微信信息。

3) UseController 控制类调用 UserDao 接口类中的 insertUsers(Users us) 方法自动保存用户信息。

具体设计过程如图 6 所示。其他模块功能设计过程交互类似, 不再图示。

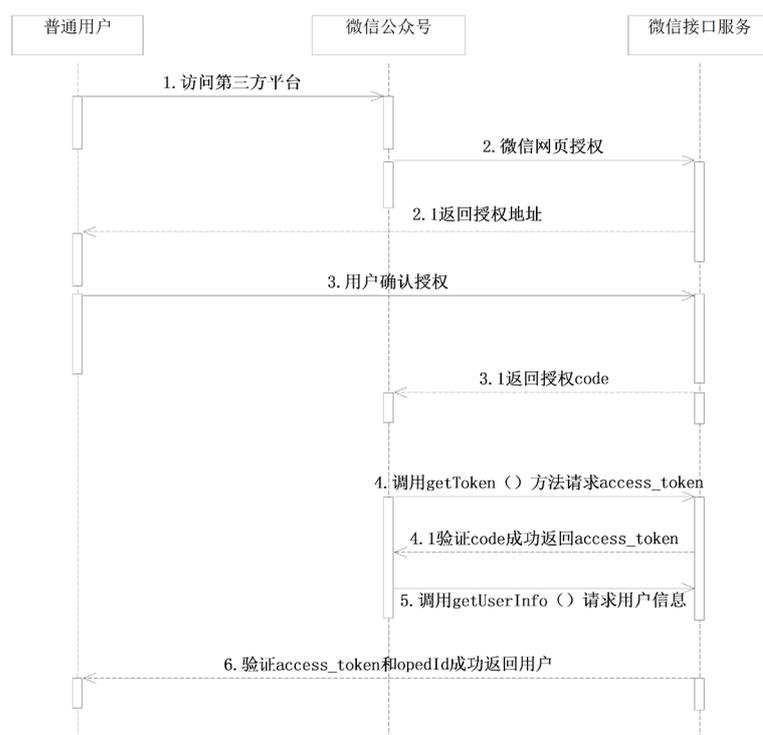


Figure 6. WeChat authorization design
图 6. 微信授权设计

3.5.2. 课堂考勤子模块的设计

该模块包含生成签到二维码和分配教学任务功能。教师进入个人中心, 跳转到生成二维码界面, 手动选择上课班级、地点和课程, 平台依据选择信息生成具有时效性的二维码。生成签到二维码的流程设计如图 7 所示。

1) 类与接口设计

该模块设计的实体类主要有课程 course、班级 class、教师 teacher 和二维码 code; 接口类有教师接口类 TeacherDao、教学安排实体类 ArrangeDao 类; 业务处理类有 TeacherServiceImpl 和 CodeServiceImpl; 控制类有 TeacherController 类和 CodeController 类。课堂考勤的控制类(CodeController): 生成签到二维码并调用 CodeService 中的方法。该类中定义 findTeacherByUserId() 方法获取教师信息; findTask() 方法查询教师的教学任务信息; GetCode() 方法根据课程 Id、班级 Id、教师 Id、教学安排 Id 等生成签到二维码。

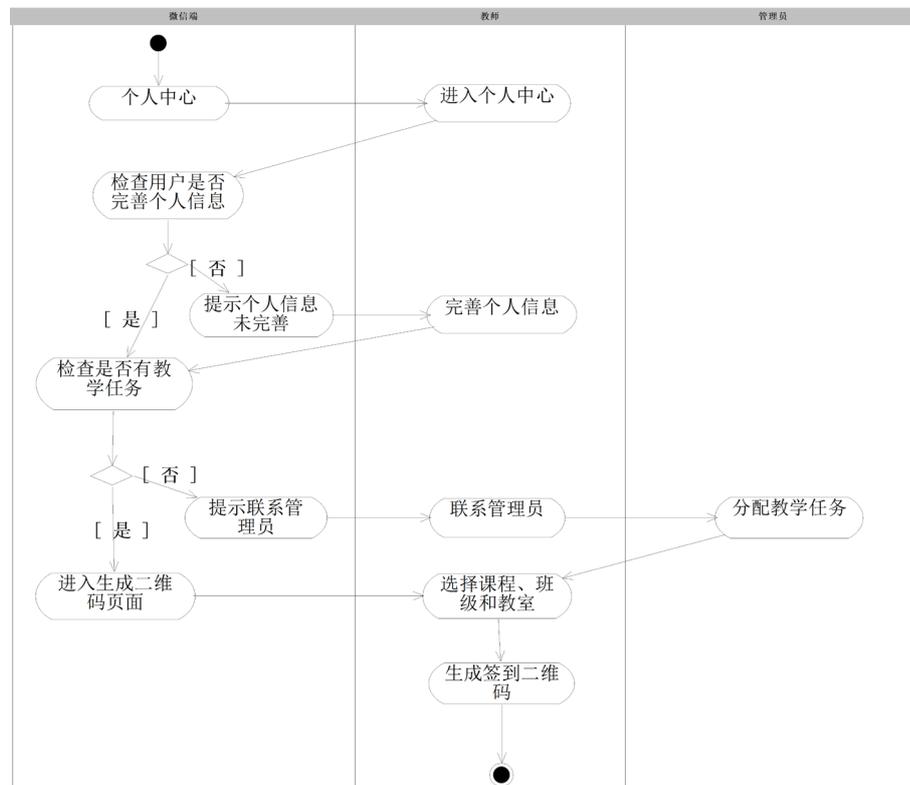


Figure 7. Process of generating check-in QR code

图 7. 生成签到二维码流程

课程实体类(Course): 数据库表结构对应的实体类, 封装课程自增 Id、课程名称(name)、课时(courseCount)、状态(status)、加入时间(createTime)、更新时间(createTime)等其属性。

2) 生成签到二维码设计

教师进入个人中心, 选择生成课堂签到二维码, 数据传到服务端请求 TechersController。在控制类中调用 TechersService 类中的 findTeacherByUserId() 方法, 此方法中调用 TechersDao 类中的 findTeacherByUserId()方法, 获取教师用户信息。

教师选择课程, 数据传到服务端请求 TecherController 类。在控制层中调用 TecherService 类中的 findTask()方法, 此方法中调用接 TecherDao 类中的 findTask()方法, 显示课程相关数据以及课程对应的班级和教室地点, 点击确定菜单生成具有时效性的签到二维码。

3.5.3. 扫码签到子模块的设计

学生在课堂扫描带有时效性的二维码签到, 包括微信扫一扫签到和查看签到记录功能。学生进入个人中心, 完善信息后扫描签到二维码, 扫码成功后保存本次签到信息。扫码签到流程设计见图 8。

1) 类与接口设计

该模块设计的类与接口有签到实体类 Sign、接口类 SignDao、业务类 SignServiceImpl 和控制类 SignController。扫码签到控制类(SignController): 扫码签到, 并调用 SignService 中的方法。该类定义 Scan() 方法, 调用微信扫一扫接口; 定义 SignAndAdd()方法, 获取学生扫码签到信息, 并分析入库; 定义的 findAllMySign()方法查询学生签到记录。扫码签到实体类(Sign): 数据库表结构对应的实体类, 封装教室关联 Id(croomId)、班级关联 Id(classId)、学生关联 Id(userStuId)、教师关联 Id(userTeachId)、课程关联 Id(courseId)、教学安排关联 Id(arrangeId)、签到时间(signtime)、加入时间(createTime)、更新时间(updateTime)等属性。

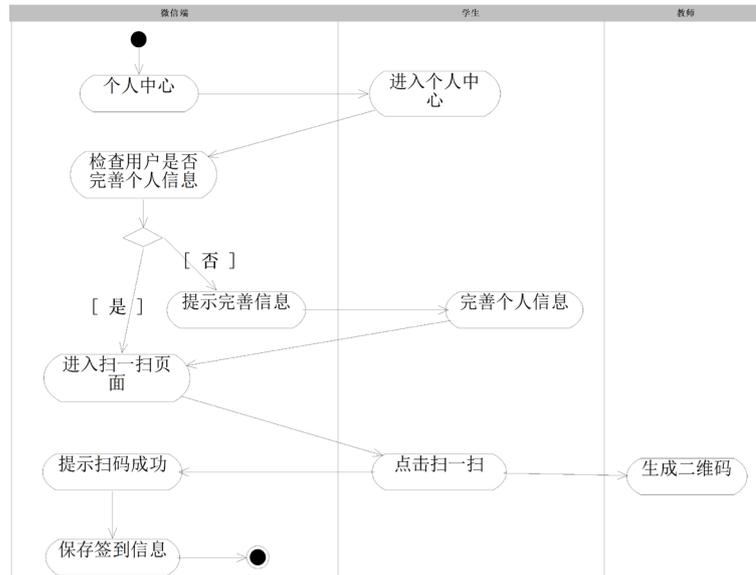


Figure 8. Scan code sign-in process
图 8. 扫码签到流程

2) 扫码签到功能设计

学生进入个人中心，选择扫一扫签到选择。数据传到服务端请求 StudentController。在控制类中调用 StudentService 中的 scan()方法，该方法中调用 StudentDao 中的 scan()方法，系统调用微信扫一扫接口扫描签到二维码。扫码成功后，系统提示签到成功。签到成功后，数据传到服务端请求 StudentController。在控制类中调用 StudentService 中的 signAndAdd(Sign sign)方法，此方法调用接 StudentDao 中的 signAndAdd(Sign sign)方法，学生签到后自动保存信息。学生点击签到记录，请求数据传到服务端请求 StudentController。在控制类中调用 StudentService 中的 findAllMySign()方法，该方法调用接 StudentDao 中的 findAllMySign()方法，查看签到记录。

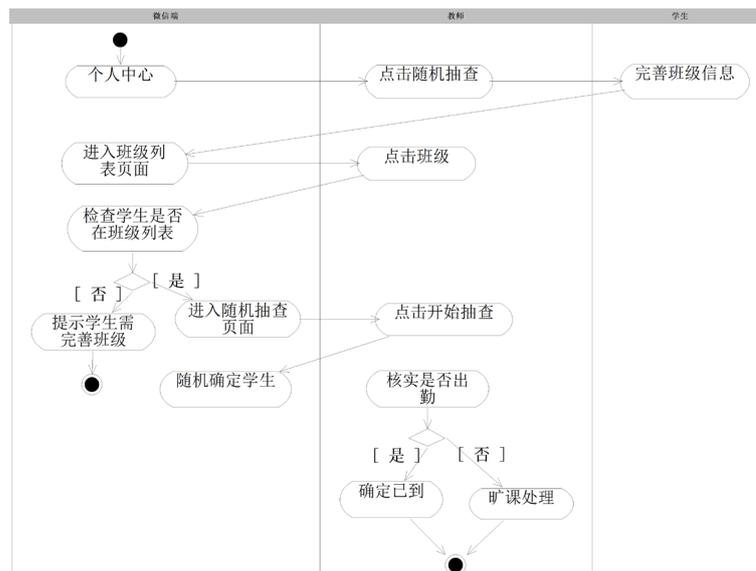


Figure 9. Random inspection process
图 9. 随机抽查流程

3.5.4. 随机抽查子模块的设计

教师随机抽查核实班级学生签到。教师进入平台，选择随机抽查后，随机显示学生姓名，当停止抽查时教师可看到该学生签到情况。随机抽查流程设计见图 9。

1) 类与接口设计

该模块涉及 Student 类、Teacher 类和 Sign 类。随机抽查控制类(SignController): 统计签到记录, 并调用 SignService 中的方法。SignService 类中 findAllStudents()方法查看本班所有的学生信息; findRandom()方法随机核实学生签到信息; findTeacherByUserId()方法查看教师信息。

2) 随机抽查功能设计

教师选择随机抽查, 选择班级, 数据信息传到服务端请求 StudetController 类。在控制类中调用 StudetService 类中的 findAllStudents()方法, 该方法调用接 StudetDao 类中的 findAllStudents()方法, 随机显示该班所有学生信息。教师随机选择停止, 将选择学生数据传到服务端请求 StudetController。在控制类中调用 StudetService 中的 findSignByRandom()方法, 该方法中调用 StudetDao 类中的 findSignByRandom()方法, 显示该学生签到情况。教师通过口头喊到方式对抽查到的学生进行核实, 并记录是否旷课。

3.5.5. 统计签到子模块的设计

教师进入个人中心, 可查看本节课、每日、周、学期签到统计和缺勤统计等。

1) 类与接口设计

该模块设计了 Sign 类、Student 类和 Teacher 类等实体类。设计了页面请求类 SignController, 在 Service 层设计 SignServiceImpl 类处理统计签到信息, 数据查询设计 SignDao 接口类。SignServiceImpl 类依赖 SignDao 接口类进行业务逻辑处理。统计签到控制类(SignController): 定义 findCurrent()方法, 查看本节课签到记录; findDaySign()方法查看日签到信息; findTermSign()方法用查看学期签到记录; findAbsent()方法查看本节课缺勤学生信息; findWeekSign()方法查看周签到信息; findClasses()方法查看班级列表。

2) 统计签到功能的设计

教师依据不同的签到统计, 选择不同的参数, 将数据传到服务端请求 SignController。在控制类中调用 SignService 中的不同方法(findCurrent()、findDaySign()、findTermSign()、findAbsent())方法, 该方法调用 SignDao 中的对应方法, 平台根据教师 Id、课程 Id、开始和结束时间等查询并分别显示、日签到、周签到、学期签到、本节未签到等统计信息。

4. 结束语

校园扫码签到平台采用 MVC 模式, 使用 SSM 框架开发。服务器端实现管理班级、课程、用户、安排教学任务等功能; 移动端绑定师、生信息, 实现扫码签到、签到统计、随机抽查等功能。经应用测试, 该平台能够满足高校课堂考勤需求。和目前已有相似的系统或平台相比, 该扫码签到平台通过制定相应信息编辑规则、时效二维码以及第三方微信平台登录等手段, 较大程度避免了代签问题; 使用 Ajax 异步访问, 提高了扫码签到的效率; 使用微信对接和扫一扫, 无须再购买硬件和下载软件, 只需关注微信公众号扫码签到, 提高签到平台的开发、维护效率和课堂签到、统计的时效性和便捷性; 使用 Layui 技术和全局异常处理机制, 提升用户体验。后续将定位引入平台, 并将平台融合学生课堂表现, 增加预警机制。该平台基于微信公众号, 使用便捷、可靠, 是未来高校考勤的主要手段之一, 值得期待。

基金项目

河南省科技厅科技发展计划科技攻关项目(立项编号: 172102210428)。

参考文献

- [1] 曹光辉. 基于人脸识别技术的考勤签到系统[J]. 电脑知识与技术, 2020, 16(10): 181-182.
- [2] 李文月. 基于智慧课室的无感知签到系统设计[J]. 信息通信, 2020(3): 114-116+118.
- [3] 赵从健, 雷菊阳, 李明明. 基于无监督学习的语音签到系统[J]. 软件, 2019, 40(12): 183-187.
- [4] 王青青, 周国祥, 戎皓, 石雷, 张翠翠. 基于互差跳跃算法的签到系统研究与实现[J]. 合肥工业大学学报(自然科学版), 2019, 42(7): 912-917.
- [5] 范哲瑞. 智能人脸识别会议签到系统的设计与实现[D]: [硕士学位论文]. 西安: 西安电子科技大学, 2019.
- [6] 牛亚运, 仲梁维. 融合人脸识别与数据分析的智能签到系统设计[J]. 软件导刊, 2019, 18(11): 30-35.
- [7] 石玲玲. AES 加密的 WiFi 安全智能签到系统的设计与实现[J]. 安徽理工大学学报(自然科学版), 2019, 39(4): 56-59.
- [8] 徐崧胜. 基于 Android 平台的学生签到系统设计和实现[J]. 电子技术与软件工程, 2020(2): 42-43.
- [9] 罗丹. 基于微信小程序的大学选修课考勤签到系统设计与开发[J]. 电子制作, 2020(1): 71-72.
- [10] 戴瑞婷, 莫李思, 黄祉衡. 一种基于扫码认证和人脸识别的签到系统设计与实现[J]. 通信技术, 2019, 52(7): 1773-1779.
- [11] 李敏, 陆莎, 常旺, 李晨波. 基于手机平台的快速学校签到系统的研究与设计[J]. 数字通信世界, 2020(7): 123-125.
- [12] 陈韶钰. 基于微信小程序的课堂考勤系统[J]. 内蒙古科技与经济, 2020(9): 59-60.
- [13] 王龙. 基于二维码的课堂签到管理系统设计与实现[J]. 软件工程, 2019, 22(6): 13-16.
- [14] 包蕾. 基于二维码的代签规避系统设计及研究[D]: [硕士学位论文]. 兰州: 兰州大学, 2018.