

# 基于 SpringBoot 与微信小程序的校园志愿者管理系统的设计与实现

朱晨星 颜硕 刘潇 唐星宇

上海工程技术大学管理学院, 中国·上海 201620

**摘要:** 随着移动互联网技术的快速发展, 微信小程序因其无需安装、即用即走的特性, 已成为校园应用的重要载体。本文设计并实现了一款基于 SpringBoot 框架与微信小程序的校园志愿者管理系统, 通过整合现代 Web 技术实现了志愿者服务全流程的数字化管理。系统采用前后端分离架构, 后端使用 SpringBoot 整合 MyBatisPlus 提供 RESTful API 接口, 前端采用 Uniapp 框架开发微信小程序, 使用 MySQL 作为数据存储方案。系统创新性地实现了基于地理位置的服务验证、多角色协同管理、实时消息通知等功能, 有效解决了传统志愿者管理中存在的效率低下、信息不透明、真实性难验证等痛点。实际应用表明, 该系统显著提升了志愿者管理的效率和服务质量, 为高校志愿服务工作的数字化转型提供了完整解决方案。

**关键词:** SpringBoot; 微信小程序; 志愿者管理; Uniapp; 数字化管理

## Design and Implementation of a Campus Volunteer Management System Based on SpringBoot

Zhu Chenxing, Yan Shuo, Liu Xiao, Tang Xingyu

School of Management Studies, Shanghai University of Engineering Science, China Shanghai 201620

**Abstract:** With the rapid development of mobile internet technology, WeChat mini-programs have become an important carrier of campus applications due to their feature of no installation required and immediate use. This paper designs and implements a campus volunteer management system based on the SpringBoot framework and WeChat mini-program, which achieves digital management of the entire process of volunteer services through the integration of modern web technologies. The system adopts a front-end and back-end separation architecture, with the back-end using SpringBoot integrated with MyBatisPlus to provide RESTful API interfaces, and the front-end using the Uniapp framework to develop WeChat mini-programs, using MySQL as the data storage solution. The system innovatively implements functions such as location-based service verification, multi-role collaborative management, and real-time message notification, effectively solving the pain points of low efficiency, information opacity, and difficulty in verifying authenticity in traditional volunteer management. Practical applications show that the system significantly improves the efficiency of volunteer management and service quality, providing a complete solution for the digital transformation of university volunteer service work.

**Keywords:** SpringBoot; WeChat Mini-Program; Volunteer management; Uniapp; Digital management

## 0 引言

在当前数字化校园建设的大背景下, 微信小程序凭借其轻量级、易传播、无需安装的特点, 已成为高校移动应用的首选平台。随着高校教育改革的深入推进, 志愿者活动已成为培养学生综合素质与社会责任感的重要载体<sup>[1]</sup>。传统的志愿者管理多依赖 Excel 登记、线下通知、纸质签到等方式, 存在管理效率低下、信息更新不及时、数据统计繁复、活动真实性难以核验等问题<sup>[2]</sup>, 已成为制约高校

志愿服务事业发展的瓶颈。

数字化转型是提升各领域管理效能的必然趋势<sup>[3]</sup>, 为解决上述痛点, 本文设计并实现了一款基于 SpringBoot 后端与微信小程序前端的综合管理系统。该系统旨在通过信息化技术, 重构志愿者活动的管理模式, 实现从活动发布、报名、参与、考核到激励的全流程线上化、智能化管理, 为提升高校志愿者工作的管理效率与服务质量提供一套完整的解决方案。

# 1 基于 SpringBoot 与微信小程序的校园志愿者管理系统的优势

## 1.1 SpringBoot 框架的技术优势

SpringBoot 作为 Java 领域的主流开发框架，具有配置简单、自动化配置等优点，通过内嵌 Tomcat 服务器和丰富的“starter”依赖，开发者可以快速搭建一套健壮的后端服务<sup>[4]</sup>。丰富的 starter 模块和良好的生态整合能力，使 SpringBoot 非常适合快速构建中小型管理系统。

SpringBoot 与 MyBatis-Plus 的结合，可以大幅减少数据库操作的代码量；与 Vue 的前后端分离架构，则使得项目分工更明确、耦合度更低，非常适合高校创新创业项目团队协作开发。相较于传统的 SSH/SSM 框架，SpringBoot 支持 RESTful API 设计，且减少了大量的 XML 配置，更加符合现代微服务架构的理念。

## 1.2 微信小程序技术选型优势

微信小程序具有庞大的用户基础，学生无需额外安装应用即可使用；其次小程序提供了丰富的 API 接口，包括地理位置、用户信息、消息推送等，能够满足志愿者管理的功能需求；小程序的审核发布机制相对简单，便于快速迭代和更新。所以选择微信小程序作为前端载体主要基于以下考虑。

Uniapp 框架的选择基于其出色的跨平台能力，使用 Vue.js 语法规则，学习成本低，开发效率高。同时，Uniapp 提供了丰富的插件市场，可以快速集成各种功能模块，如地图导航、文件上传、图片处理等。

## 1.3 校园志愿者管理系统的优势

相较于传统线下管理方式，基于 SpringBoot 与微信小程序的校园志愿者管理系统带来了多维度的提升：

### 1.3.1 高效率管理

系统全流程线上操作自动化，实现了志愿者任务的在线发布、申领、审核、打卡全流程管理，避免了人工统计的繁琐和错误，大大提升了工作效率。

### 1.3.2 信息透明度更强

学生可通过系统实时查看任务信息、申领状态、打卡记录等，管理人员可随时掌握志愿者活动开展情况，实现了信息的双向同步与透明化。

### 1.3.3 低运营成本

系统采用前后端分离架构，支持微信小程序访问，学生无需安装额外 APP，通过熟悉的微信环境即可参与志愿者活动，降低了推广和使用门槛。

### 1.3.4 数据安全性高

为了确保志愿者信息的安全性，系统采用 JWT+Sa-Token 双重认证机制，实现了严格的权限控制和数据保护。

### 1.3.5 真实可信

通过地理位置打卡等机制，有效确保了志愿服务数据的真实性，为公正的考核与评优提供了数据基石。

## 2 系统需求分析

### 2.1 业务需求分析

校园志愿者管理系统需要满足以下业务需求：支持学生在线注册和身份认证；发布志愿者任务、浏览和申领任务；全流程管理任务审核、执行和打卡；支持请假申请和流程审批；提供数据统计和生成报表，为管理决策提供数据支持；在这些基础上确保系统的安全性、稳定性和可扩展性。

### 2.2 功能需求分析

系统主要分为学生端和管理端两个部分，各自具有不同的功能需求。如图 1 所示。

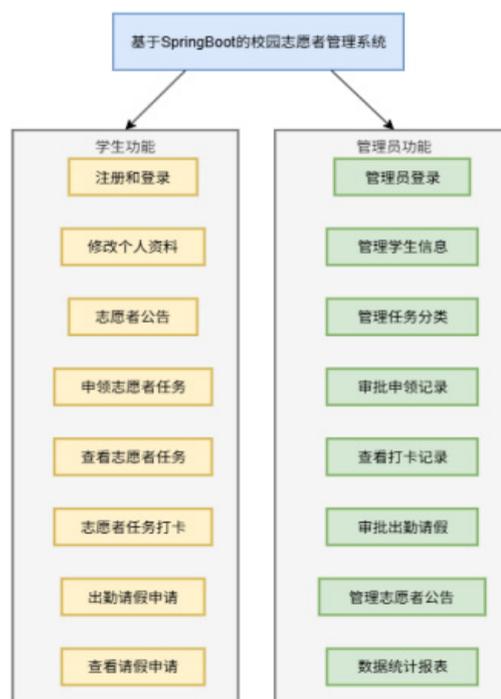


图 1 功能需求

学生端功能需求包括：注册登录功能，支持学生通过学号和信息认证完成注册和登录；个人信息管理，学生可以查看和修改个人基本信息；任务浏览和搜索，学生可以查看可申领的任务列表并进行条件筛选；任务申领功能，学生可以选择感兴趣的任务并提交申领申请；打卡签到功能，学生参与活动后可以通过地理位置进行打卡；请假申请功能，学生因故不能参加活动时可以提交请假申请；活

动记录查询, 学生可以查看个人参与活动的历史记录和统计信息。

管理端功能需求包括: 学生信息管理, 管理员可以查看和管理学生基本信息; 任务发布管理, 管理员可以发布、编辑和删除志愿者任务; 申领审批功能, 管理员可以审核学生的任务申领申请; 打卡记录管理, 管理员可以查看和验证学生的打卡记录; 请假审批功能, 管理员可以处理学生的请假申请; 数据统计报表, 管理员可以查看各类统计数据 and 生成报表; 系统权限管理, 管理员可以管理用户角色和权限设置。

### 3 系统设计与实现

#### 3.1 总体架构设计



图 2 总体架构设计

本系统采用前后端分离的架构设计, 如图 2 所示, 后端基于 SpringBoot 框架提供 RESTful API 接口, 前端采用 Uniapp 框架开发微信小程序。这种架构设计使得前后端可以独立开发、测试和部署, 提高了开发效率和系统的可维护性。

后端架构: 基于 SpringBoot 2.2 版本, 采用 MVC 设计模式, 分为前端展示层、业务逻辑层和数据访问层。前端展示层负责接收前端请求并返回响应; 业务逻辑层实现业务逻辑处理; 数据访问层使用 MyBatisPlus 操作 MySQL 数据库。同时集成 Sa-Token 进行权限认证, 使用 JWT 进行身份验证。

前端架构: 采用 Uniapp 框架开发微信小程序, Uniapp 支持使用 Vue.js 语法开发跨平台应用, 一套代码可发布到 iOS、Android、Web 以及各种小程序平台<sup>[5]</sup>。前端使用 Vue.js 进行组件化开发, 配合 ElementUI 组件库构建用户界面。

#### 3.2 数据库设计

系统使用 MySQL8.0 数据库, 设计了用户表、任务表、申领记录表、打卡记录表、请假表、公告表等数据表。

以志愿者任务表为例, 包含以下字段: 任务 ID、任务名称、任务类型、开始时间、结束时间、任务地点、所需人数、任务状态等。

### 3.3 后端核心功能实现

#### 3.3.1 登录认证模块

系统采用 JWT 令牌实现用户身份认证, 支持多种登录方式。JWT 是一种轻量级的身份认证协议, 由头部、载荷和签名三部分组成<sup>[6]</sup>。系统登录过程由以下五个步骤组成: 输入用户用户名和密码; 验证系统用户信息是否正确; 验证通过后生成 JWT 令牌并返回给客户端; 客户端在后续请求中携带 JWT 令牌进行身份验证。

除此之外, 为了提高系统的安全性, 还集成了 Sa-Token 框架进行权限控制。Sa-Token 提供了丰富的权限管理功能, 包括登录认证、权限验证、会话管理等功能。通过 Sa-Token 实现了细粒度的权限控制, 确保不同角色的用户只能访问其权限范围内的功能。后端系统登录界面如图 3 所示。



图 3 系统登录界面

#### 3.3.2 任务管理模块

管理员可以通过系统发布和管理志愿者任务, 主要包括任务信息的增删查改。任务发布过程包括以下四部分: 填写任务基本信息; 任务需求人数和相关要求设置; 提交任务发布申请; 通过系统审核后任务正式发布。

任务管理模块提供了查询和筛选功能, 管理员可以按状态、时间、类型等条件查询任务。除此之外, 该模块还提供了任务统计功能, 可以查看任务的参与情况、完成情况等统计信息。任务管理界面如图 4 所示。

ID	名称	发布时间	图片	发布者	截止时间	操作
1	植树造林	2023-09-28 10:00:00		管理员发布任务	2023-09-30 23:59:59	发布 删除
2	垃圾分类	2023-09-28 10:00:00		管理员发布任务	2023-09-30 23:59:59	发布 删除
3	社区志愿服务	2023-09-28 10:00:00		管理员发布任务	2023-09-30 23:59:59	发布 删除
4	环保宣传	2023-09-28 10:00:00		管理员发布任务	2023-09-30 23:59:59	发布 删除
5	敬老爱老	2023-09-28 10:00:00		管理员发布任务	2023-09-30 23:59:59	发布 删除

图 4 任务管理界面

#### 3.3.3 任务申领模块

学生用户可浏览所有可以申领的志愿者任务, 并在线

提交申领申请。任务申领过程包括以下五部分：学生浏览可申领任务列表；查看任务详细信息和要求；选择符合条件的任务并提交申领申请；等待管理员审核；审核通过后正式参与任务。除此之外，该模块还设置了申领限制，防止学生同时申领多个时间冲突的任务。

### 3.3.4 打卡签到模块

系统支持地理位置打卡的方式，确保志愿者活动的真实性。学生到达活动地点后打开打卡功能，系统获取当前地理位置信息，验证地理位置是否在活动范围内，通过验证后完成打卡。为了提高打卡的准确性和便捷性，系统采用了高德地图 API 进行地理位置服务，能够准确获取和验证用户的地理位置信息。同时，设置打卡时间限制，确保打卡时间与活动时间一致。

### 3.3.5 数据统计模块

系统提供多维度数据统计分析功能，支持图表可视化展示。包括志愿者参与情况统计、任务完成情况统计、服务质量评估统计、活跃度分析等统计功能。

通过数据可视化技术，将统计结果以图表形式展示，方便管理员直观了解志愿者活动的整体情况。同时，支持数据导出功能，可以将统计结果导出为 Excel 或 PDF 格式，便于进一步分析和存档。

## 3.4 微信小程序前端实现

### 3.4.1 小程序页面设计

微信小程序前端采用模块化设计，主要包含以下功能页面：首页展示志愿者活动列表和推荐活动；活动详情页显示活动详细信息并提供报名入口；个人中心页面管理用户信息和参与记录；打卡签到页面实现地理位置打卡功能；消息中心页面显示系统通知和活动提醒。小程序部分页面如图 5 所示。

小程序的页面布局采用响应式设计，适配不同尺寸的移动设备。通过使用 Flex 布局实现页面的自适应，确保在各种手机屏幕上都能显示良好。页面风格遵循微信小程序设计规范，采用简洁明了的界面设计，为用户提供流畅的体验。

### 3.4.2 小程序 API 集成

微信小程序前端与后端通过 HTTP API 进行数据交互，主要集成了以下类型的接口：

①用户相关接口：包括微信登录、用户信息获取、权限验证等。利用微信提供的登录能力，快速实现用户身份认证，获取用户唯一标识 OpenID。

②活动管理接口：实现活动列表获取、活动详情查

询、报名活动、取消活动等功能。通过合理的接口设计，确保传输数据的高效和安全。

③地理位置接口：集成微信小程序的地理位置 API，实现精确定位活动地点和打卡时的位置验证。通过 wx.getLocation 接口获取用户当前位置坐标，并与活动地点计算距离。

④消息模板接口：利用微信模板消息功能，实现活动提醒、结果审核通知、提醒打卡成功等消息推送服务，提高用户参与和系统互动。

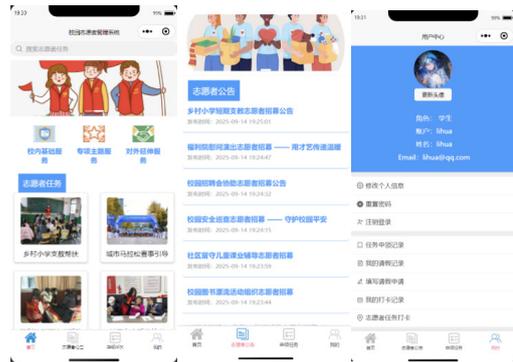


图5 小程序页面

## 4 结语

本文设计并实现的基于 SpringBoot 与微信小程序的校园志愿者管理系统，有效解决了传统志愿者管理方式中存在的效率低下、信息不透明等问题。系统稳定运行，操作简便，具有良好的实用性和推广价值。

未来将进一步优化系统的移动端体验，增强数据分析功能，为志愿者管理工作提供更加强大的技术支持。之后也会建立完善的志愿服务积分体系与商城，将志愿服务时长转化为可兑换的礼品、特权或荣誉勋章。形成“参与 - 积累 - 兑换”的正向激励循环，持续地激发学生动力。

### 参考文献：

- [1] 漆依林. 互联网 + 志愿服务：新时期青年志愿服务模式研究[J]. 青年发展论坛, 2019,29(4):9.
- [2] 朱江. 大学生青年志愿者管理系统的设计与实现[D]. 湖南大学[2023-08-10].
- [3] 康林冲, 姜晓轶, 吕懂憬等. 数字海洋应用服务系统构建及关键技术研究[J]. 应用海洋学学报, 2016,35(4):585-592.
- [4] 欧阳桂秀. 基于 Java 和 MySQL 的数据库管理系统的设计与实现[J]. 信息记录材料, 2022,23(9):240-242.
- [5] 陶力, 陈锦辉. 一种基于校园志愿者服务管理的一体化信息智能管理系统: CN202011162785.4[P]. CN112232782A[2023-08-10].

[6] 刘宇. 北京志愿服务信息管理系统的设计与实现 [D]. 北京工业大学, 2018.

基金项目: 上海工程技术大学大学生创新创业训练计划项目资助 (项目编号: cx2503041)。

作者简介: 朱晨星 (2005.03-), 男, 汉族, 河南许昌人, 本科, 管理学院 信息管理与信息系统专业 2023 级, 研究方向: 大数据与商务智能。