

基于Android的学生请假APP的设计与实现

张 升, 潘佳佳

(太原工业学院计算机工程系, 山西 太原 030008)

摘 要: 本文针对传统学生请假过程中流程繁琐耗时、记录不便管理等弊端, 设计开发一款基于Android的学生请假APP。该软件分为学生、教师、班主任等三个不同的角色, 在请假流程流转过程中对应不同的任务节点。本文首先简要介绍了系统开发的环境及工具, 然后根据不同角色分析了各自的功能需求, 并介绍了其设计及实现过程, 最后对系统进行了功能测试, 验证了其能够快捷高效的完成请假流程的流转及管理。

关键词: Activiti; Android; 学生请假

中图分类号: TP311.1 **文献标识码:** A

Design and Implementation of an Android-Based APP for Students Asking for Leave

ZHANG Sheng, PAN Jijia

(Department of Computer Engineering, Taiyuan Institute of Technology, Taiyuan 030008, China)

Abstract: In view of the disadvantages of complicated process, time consuming and inconvenient record management in the process of traditional students asking for leave, this paper designs and develops an application system based on Android. The three types of users in this application are students, teachers, and supervisors. Firstly, this paper introduces the environment and tools of the system development, then analyzes their functional requirements of different types of users, and introduces the design and implementation process. Finally, the results of functional test have proved the high efficiency of the APP.

Keywords: Activiti; Android; students asking for leave

1 引言(Introduction)

目前, 大学校园信息化正在逐渐完善, 为了便于全校师生的学习、办公、管理和生活, 教务管理系统、招生就业系统等各种管理信息系统纷纷开始使用。而对于学生请假业务目前没有相应的软件进行实现与管理, 还是按照线下方式完成。一般情况下, 学生需要提前填写请假申请表, 班主任首先进行审批, 并根据请假事由及天数, 判断是否要交由上级学生主管部门审批备案。但这种形式有许多缺点, 例如请假过程费时费力, 历史信息不便记录等。所以无论是学生还是教师都需要一个软件对学生请假业务进行实现和管理。

本系统的目的是设计实现一个基于Android^[1,2]的学生请假APP。此APP可以借助安卓移动设备, 不仅使学生可以随时随地发起请假申请, 还可以使老师及时处理请假申请, 方便记录、了解学生的请假情况。

2 系统开发环境及技术(System development environment and technology)

2.1 开发环境

系统使用Android Studio为开发环境。Android Studio是目前Android开发中使用人数最多的开发工具, 其软件布局大方得当, 开发人员使用十分顺手, 具有丰富的快捷方法, 流畅的代码提示功能, 其自带的布局编辑器能够使开发人员及时预览布局效果或直接拖拉控件。

2.2 系统架构及主要技术

系统采用MVP^[3]设计模式, 使Android开发中实现了更好的代码解耦。View层对应于视图, 负责页面控件的显示, 以及与用户进行数据交换, Presenter为了完成视图与数据间的连接, 通过Model层的对象获取数据, 然后将其传到View层, 而Model层主要作用于数据, 对其进行加工处理。MVP

架构图如图1所示。

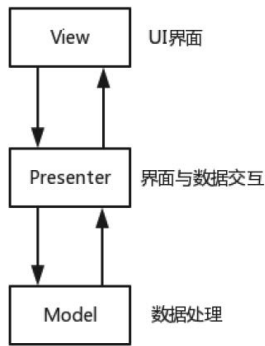


图1 MVP基本架构图

Fig.1 MVP architecture diagram

系统使用了Activiti^[4,5]业务流程管理框架，Activiti是基于Apache许可的开源BPM^[6]平台，覆盖了业务流程管理、工作流、服务协作等领域的一个开源的、灵活的、易扩展的可执行流程语言框架。本系统在此框架下进行请假业务流程建模，在此框架下进行请假流程的创建、流转、历史记录等操作。

在请假业务流程(图2)中，首先由学生发起请假申请，提交给班主任审批；班主任根据请假天数、是否离校等情况，经过互斥路由节点，选择是直接通知学生审批结果，还是转由上级分管领导审批；分管领导审批结束后，将审批结果通知学生；学生根据审批结果经过路由节点做相应处理，若通过则通知相关代课教师后结束流程，若没有通过则直接结束流程。

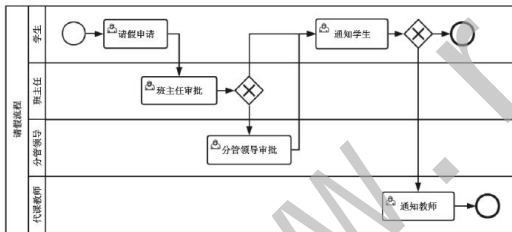


图2 请假流程

Fig.2 Ask for leave process

3 系统需求分析及功能设计(System requirement analysis and function design)

3.1 系统功能需求

本系统主要有学生、班主任/分管领导和代课教师三种不同的角色，分别有不同的功能需求。

(1)学生角色主要功能

①发起请假申请：学生发起请假申请，填写请假申请表中的请假起止时间、请假事由、是否离校外出及外出目的地等信息后，提交班主任审批。

②发起销假申请：学生对审批通过的请假申请，发起销假申请，提供图片等返校凭证后，提交班主任审核。

③历史查询：查询历史请假申请情况。

(2)班主任/分管领导角色主要功能

①请假审批：审核学生发起的请假申请表中的内容，填写审批意见后，进行拒绝、同意及转发操作。

②销假审批：审核学生发起的销假申请，查看返校凭证的图片，进行拒绝、同意及转发操作。

③历史查询：查询处理过的请假、销假申请。

(3)代课教师角色主要功能

代课教师角色功能较少，主要为请假信息查询：用来查看学生请假成功后转发过来的有效的请假信息。

3.2 系统数据库设计

本系统数据库名为leavesystem，里面包含34个表，其中23个表为Activiti框架自动生成的表，其余的数据表依次分别是本系统业务所需要的表。下面将对这些数据表分别介绍。

(1)Activiti框架相关表

Activiti框架相关表全部是以act_开头，分别有资源流程信息表、运行时信息表、历史数据表、用户信息表、公共数据表五类。

①资源流程信息表主要有：

act_re_deployment表：用来用于记录部署信息；

act_re_model表：用于记录流程设计模型部署数据；

act_re_procdef表：用于记录流程定义数据。

②运行时信息表主要有：

act_ru_execution表：用于记录运行时流程执行实例信息；

act_ru_identitylink表：用于记录运行时流程人员信息；

act_ru_task表：用于记录运行时任务节点信息；

act_ru_variable表：用于记录运行时流程变量数据；

act_ru_event_subscr表：用于throwEvent、catchEvent时间监听；

act_ru_job表：用于记录运行时定时任务数据。

③历史数据表主要有：

act_hi_actinst表：用于记录历史节点信息；

act_hi_attachment表：用于记录历史附件信息；

act_hi_comment表：用于记录历史意见；

act_hi_identitylink表：用于记录历史流程人员信息；

act_hi_detail表：用于记录历史详情；

act_hi_procinst表：用于记录历史流程实例信息；

act_hi_taskinst表：用于记录历史任务实例信息；

act_hi_varinst表：用于记录历史变量。

④用户信息表主要有：

act_id_group表：用于记录用户组信息；

act_id_info表：用于记录用户扩展信息；

act_id_membership表：用于记录用户与用户组对应信息；

act_id_user表：用于记录用户信息。

⑤公共数据表主要有：

act_ge_bytearray表：用于记录二进制数据；

act_ge_property表：用于记录整个流程引擎级别的数据。

(2)系统业务相关表

系统业务相关的表主要有：用户表(user)、老师表(teacher)、学生表(student)、专业表(major)、班级表(classes)、角色表(role)、角色—用户表(role_user)、老师—班级表(teacher_class)、假条表(leavecontent)、请假表(leavess)、销假表(back)等。

①用户表(user)：主要字段有用户id、用户名、用户账号、密码、邮箱等。

②老师表(teacher)：老师继承自用户，主要字段为id，且与用户表一致。

③学生表(student)：学生继承自用户，主要字段有id(与用户表一致)、班级id、专业id等。

④专业表(major)：主要字段有专业id、专业名称、专业说明、备注等。

⑤班级表(classes)：主要字段有班级id、班级名称、专业id、备注等。

⑥角色表(role)：主要字段有角色id、角色名称、说明等。

⑦角色—用户表(role_user)：主要字段有角色id、用户id，用来存放角色和用户的对应关系。

⑧老师—班级表(teacher_class)：主要字段有老师id、班级id、角色id，用来存放老师和班级的对应关系。

⑨假条表(leavecontent)：主要字段有id、流程id、学生id、开始时间、结束时间、请假原因、状态标志位等。

⑩请假表(leavess)：主要字段有id、流程id、假条id、老师id、申请时间、回复时间、回复内容、状态标志位、转发id等。

销假表(back)：主要字段有id、流程id、假条id、销假内容、时间、地址、照片路径、回复内容、回复时间、申请时间、状态标志位等。

4 系统实现(System implementation)

系统可分为Android端和服务端两部分。Android端采用MVP架构，其中View层为***Activity类，用来实现Android端界面，并使用ButterKnife进行View对象绑定；Presenter层为***Request类，负责View层和Model层的交互，被View层的***Activity类引用，同时引用Model层的数据模型，若需要服务端的支持通过Retrofit网络请求框架向服务端发送请求。服务端使用Servlet来处理Http请求；定义数据表对应的实体类；定义***Dao类用来对实体类进行数据访问操作。下面介

绍两个主要功能的实现过程。



图3 登录界面

Fig.3 Login interface



图4 请假界面

Fig.4 Ask for leave interface

(1)用户登录

登录界面(图3)中主要有账号、密码文本框(EditText)、登录按钮(Button)和logo图标(ImageView)等组件，由View层LoginActivity类实现该页面。打开登录页面时，在onCreate方法中为组件、引用的Request及相关数据进行初始化并使用ObjectAnimator属性动画以渐出效果呈现各个组件。当输入账号、密码后点击登录按钮时，在登录按钮OnClick事件方法中判断账号、密码是否填写完整；若没有填写完整，弹出对应的Toast消息框；若填写完整，调用Presenter层LoginActivityRequest类中的login方法向服务端发送请求，服务端使用LoginServlet类处理该请求，返回登录是否成功的响应结果。login方法中将根据请求的不同响应结果给出不同处理，若登录请求不成功，则弹出响应的信息；若登录请求成功，调用View层中success方法，利用SharedPreferences将用户信息记录在用户手机中，同时判断当前用户角色，若角色数大于1，则弹出选择角色窗口，由用户选择后，进入对应的界面，否则直接进入当前角色对应的界面。

(2)发起请假

发起请假界面(图4)主要有请假开始时间、请假结束时间、请假请求发送至及请假原因等组件，由View层WriteLeaveActivity类实现该界面。用户进入该页面后，点

击“开始时间”和“结束时间”将出现DatePickerDialog时间选择器,方便用户选择时间。若开始时间超过结束时间,在Toast提示框中,显示相应错误信息。点击“选择发送到”按钮,用户可选择发送请假请求的老师,列表中优先列出该学生的班主任。信息填写完成后,点击“完成”按钮,将通过对应的Presenter层WriteLeaveActiRequest类中的toLeave方法向服务器发出请求,服务器端ToLeaveServlet接受处理后,返回是否响应成功的结果。

5 结论(Conclusion)

系统针对高校学生请销假的实际需求出发,使用标准的BPM框架Activiti进行流程设计,实现了学生请假的Android App,能够方便快捷的进行请假流程的流转,能够实现请假信息的管理与统计。系统使用了目前Android开发中较流行的MVP架构,使用Retrofit框架实现HTTP请求,使用Servlet进行请求响应,页面设计采用Material风格。系统主要分学生、班主任/分管领导和代课教师等三种角色,实现了学生发起请销假请求、班主任/分管领导处理(批准或转发)请销假请求、学生和代课教师查看审批结果等流程流转功能,此外还实现了历史请假请求查询、统计等信息管理功能。当前系统中的请销假流程是使用Activiti框架预先设计好的,任务节点相对固定,虽然能满足当前请销假业务的需要,但不够灵活,若业务流程有较大变动,需重新设计修改编码。所以,系统在

后期迭代更新时,可考虑增加流程定义功能,能够在系统中定义使用流程。

参考文献(References)

- [1] Liu Yi.Mobile Network Application Engine Design Based on Android System[J].Applied Mechanics and Materials,2014(608):291-294.
- [2] Lin Deng,Jeff Offutt,Paul Ammann,Nariman Mirzaei.Mutation operators for testing Android apps[J].Information and Software Technology,2017,1(81):154-168.
- [3] 曾露.MVP模式在Android中的应用研究[J].软件,2016(06):75-78.
- [4] 孙勇.基于Activiti的考勤 workflows 系统设计与实现[J].计算机时代,2016(02):75-78.
- [5] 李修云.基于Activiti框架的在线审批流程应用研究[J].计算机科学,2016,43(6A):555-557.
- [6] Matthias Geiger,Simon Harrer,Jörg Lenhard,et al.BPMN 2.0:The state of support and implementation[J].Future Generation Computer Systems,2018,3(80):250-262.

作者简介:

张 升(1982-),男,硕士,讲师.研究领域:计算机应用。
潘佳佳(1995-),女,本科.研究领域:软件开发。

(上接第45页)

算法学习的输入,因此应深入研究乳腺癌X光片的病理特征和图像本身成像效果,进一步提高图像预处理和特征提取的质量。

(2)在模式识别的算法方面可以考虑改进参数的优化方法以提高算法的效率。

(3)在模式识别中训练模型的性能与参与训练的样本有直接的关系,因此提高训练样本的数量,以及样本的覆盖范围,能够有效提高最终的判断准确性。

参考文献(References)

- [1] 王欣,连臻强.中国乳腺癌筛查现状和评价[J/CD].中华乳腺病杂志(电子版),2015,9(3):159-162.
- [2] Fan L,Strasser-Weippl K,Li JJ,et al.Breast cancer in China[J].Lancet Oncol,2014,15(7):279-289.
- [3] 张欣.基于统计模型的乳腺癌微钙化病灶辅助检测研究[D].西北大学,2010.
- [4] 章永来,史海波,尚文利,等.面向乳腺癌辅助诊断的改进支持向量机方法[J].计算机应用研究,2013,30(8):2373-2376.
- [5] 郝冬冬.基于人机智能融合的乳腺癌辅助诊断方法[D].杭州

电子科技大学,2015.

- [6] 阳维.乳腺肿瘤的超声图像特征定量分析与良恶性识别[D].上海交通大学,2009.
- [7] 周悦.基于乳腺X线图像的计算机辅助诊断方法研究[D].兰州大学,2014.
- [8] 郝欣.基于乳腺癌X线肿块影像的计算机辅助诊断技术研究[D].浙江大学,2013.
- [9] 王迪.在线和分解支持向量机器学习算法研究[D].中国科学院大学,2012.
- [10] Cascio D,Fauci F,Magro R,et al.Mammogram segmentation by contour searching and mass lesions classification with neural network[J].IEEE Transactions on Nuclear Science,2006,53(5):2827-2833.

作者简介:

牛 琳(1978-),女,硕士,讲师.研究领域:数据挖掘,信息管理系统。

张雨薇(1995-),女,本科生.研究领域:信息管理系统。

张露馨(1995-),女,本科生.研究领域:信息管理系统。