

基于 Android 平台的校园志愿者 APP 的设计与开发研究

龚婷婷*

GONG Ting-ting

摘要

校园志愿者队伍是和谐校园建设的重要力量。随着志愿者队伍的壮大,志愿活动越来越多,为了便于随时随地管理志愿活动,一款相关的手机 APP 显得尤为需要。针对这样的需求,以 C# 作为开发语言,以 Visual Studio 2012 作为开发工具和 SQLite 内嵌数据库,开发了基于 Xamarin.Android 的校园志愿者 APP,并详细介绍了主要功能模块的实现方法。

关键词

校园志愿者; APP 设计与开发; Android 平台; Xamarin

doi: 10.3969/j.issn.1672-9528.2019.07.056

1 引言

校园志愿者秉承着“立足校区,走向社会”的宗旨,与学校周边所在社区结对开展一系列公益志愿服务活动,服务范围涉及公益事业的各个方面,包括关爱孤儿、探访独居老人、关爱残疾儿童、校区服务、公益社团交流等。^[1]学生参与志愿活动,对推动和谐校园的建设和培养学生综合素质起着非常重要的作用。随着志愿者队伍不断壮大,志愿活动也越来越多,显然手机 APP 应用可以方便随时随地管理志愿活动。目前 Android 依然是全球第一大的主流智能操作系统,可以为用户提供高质量的服务保障,在学生群体中认知度和占有率极高。^[2]因此,文本采用了 Xamarin.Android 技术开发了校园志愿者 APP。

2 志愿者 APP 的设计

2.1 角色

结合实际调研数据,进行此 APP 设计过程中,将校园志愿者 APP 一共划分为两种角色,分别是教师和志愿者成员。教师涉及的权限有登录、查看发布的活动、生成活动二维码、结束活动和个人信息管理;志愿者成员的权限有登录、查看招募活动、我的活动、历史活动记录、缺席活动、活动详细、扫码签到和个人信息管理。

2.2 功能模块

2.2.1 志愿者活动模块

教师用户登录系统后可以查看已发布的活动、活动的详

细信息。

志愿者成员用户登录后可以查看正发布的招募活动、活动的详细信息,对于感兴趣的可以申请参加;可以查看我当前参与的活动、参与的所有历史活动和缺席的所有活动。

2.2.2 活动签到模块

教师用户可以生成活动唯一的二维码供活动现场成员扫码签到,签到完成后在规定时间内结束活动。

志愿者成员用户可以扫码进行签到,累计服务时长。

2.2.3 个人信息管理模块

教师用户和志愿者成员用户可以查看个人信息,包括参与的活动和自己的服务时长。

2.3 系统功能结构设计

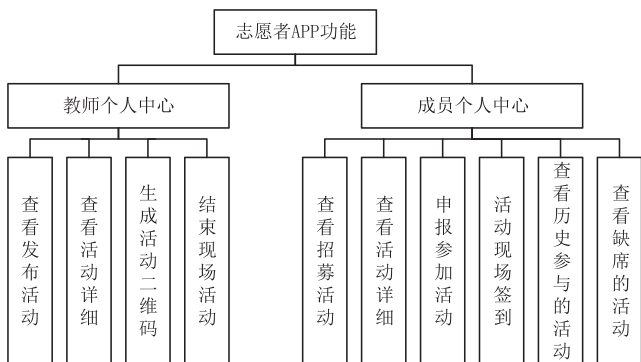


图 1 志愿者 APP 的功能结构

3 志愿者 APP 的开发

3.1 相关开发技术

3.1.1 Xamarin

* 镇江高等职业技术学校 江苏镇江 212016

Xamarin 的诞生旨在让移动开发变得难以置信的简单和迅捷。它是一个跨平台的开发框架。通过 Visual Studio 结合 Xamarin 利用 C# 语言来构建跨平台移动应用,是微软提供给移动开发者的跨平台开发解决方案^[3]。Xamarin 以 C# 作为开发语言,程序员直接用 C# 就可以开发。Xamarin 提供了整合开发工具 (IDE) Xamarin Studio,并针对 Microsoft Visual Studio 提供附挂套件。它还提供开发框架,支持跨平台的功能实作。它具有各平台的编译程序,能直接编译成 IOS 与 Android 平台的机器码。在使用 Xamarin 开发 Android 应用时,除提供 Android 开发环外,还需要其它的软件和工具。

3.1.2 Android

Android 是一种基于 Linux 的操作系统,主要应用于移动设备,优点是开放性,能满足大多数用户的使用需求。该系统由谷歌公司研发,在开放源码操作系统的支持下广泛应用于移动终端,并为第三方软件开发提供平台。^[4]Android 的框架结构主要有四个部分:Application Framework (应用程序框架)、Application (应用程序)、系统运行层和 Linux 核心层。在开发 Android 应用时,需要用到 Android 开发工具 ADT。ADT 不仅有 Android SDK Manager 工具,还有 Android 模拟器管理工具。

3.1.3 C# 语言

C# 是一种用于创建要运行在 .NET CLR 上的应用程序的语言,它从 C 和 C++ 演化而来,是微软专门为使用 .NET 平台而创建的。C# 是用于 .NET 开发的其中一种语言,但它却是最合适的一种语言。C# 能使用 .NET Framework 代码库提供的每种功能。C# 与 Xamarin 结合的好处有:具有更多的类型,但是输入却少了;C# 支持异步,异步编程保持应用程序响应;与 Java 相比,能简单地运用 lambda 表达式。

3.1.4 SQLite

SQLite 是一款采用 C 语言编写的开源的、轻量级的、跨平台的、嵌入式的关系型数据库。它支持大部分 SQL92 标准,支持弱类型数据存储。它的开发在省去了专用于关系型数据库管理系统的日常费用的同时,同样可以为应用软件提供方便的数据管理。^[5]它不占用很多资源,在嵌入式的设备中,只要几百 K 就可以了。它支持 Windows 和 Linux 等主流的操作系统,能和很多语言结合使用,如 C#、Java、php 等,比起一些数据库 DBMS,它的处理速度非常快。SQLite 有一个简洁的、模块化的体系结构,采用了独特的方法进行关系型数据库的管理。

3.2 数据库设计

根据 APP 功能模块的设计需求,建立了 11 个数据表,

即活动信息表 (ActivityInfo)、申请活动表 (ApplyInfo)、用户信息表 (UserInfo) 和用户角色信息表 (UserRoleInfo),如表 1 所示。以活动信息表为例,表结构如表 2 所示。

表 1 校园志愿者 APP 数据表清单

序号	表名 (中文)	表名 (英文)	数据库表说明
1	活动信息表	ActivityInfo	存放活动信息
2	申请活动表	ApplyInfo	存放申请活动信息
3	用户信息表	UserInfo	存放用户信息
4	用户角色信息表	UserRoleInfo	存放用户角色信息

表 2 活动信息表字段详细信息

序号	名称 (中文)	名称 (英文)	类型	说明
1	活动编号	ActivityID	String	主键
2	序列号	IndexId	Int	
3	活动名称	ActivityName	String	
4	组别名称	TeamName	String	
5	用户姓名	UserName	String	
6	活动开始时间	ActivityStartTime	DateTime	
7	活动结束时间	ActivityEndTime	DateTime	
8	活动地点	ActivityLocation	String	
9	活动简介	ActivitySummary	String	
10	活动状态	ActivityState	String	
11	活动人数	ActivityAttend	Int	
12	参加人数	JoinCount	Int	
13	时间戳	Tick	Long	
14	是否我的活动	IsMine	Bool	教师用户
15	是否申请参加活动	IsApply	Bool	

3.3 功能模块实现

3.3.1 APP 登录主界面



图 2 APP 登录界面图



图 3 活动发布列表页

用户打开 APP, 如第一次使用, 则呈现欢迎界面, 进而进入 APP 登录界面 (如图 2 所示)。登录时 APP 会检查移动设备的网络状态, 如若网络连接报错错误时, 会提示“登录失败, 请检查网络连接! ”。用户名或密码错误时, 会提示“登录失败”。登录成功时提示“登录成功!”。

3.3.2 志愿活动发布

(1) 活动发布列表

活动列表页 (如图 3 所示) 提供了查看教师发布的活动信息, 列表信息包括活动标题和开始时间。下拉页面可以查看新增的活动信息, 上拉页面可以加载更多活动信息。点击右箭头可以查看活动的详细信息。

(2) 活动详细

活动详细页 (如图 4 所示) 提供了查看活动详细信息的功能, 信息包括活动标题、活动地点、活动地点、组别名称、开始时间、参加人数和活动简历。点击生成二维码按钮, 跳转进入生成活动的二维码页面 (如图 5 所示)。参与活动的人可用手机扫一扫二维码进行签到, 在所有人完成签到后, 教师点击结束活动按钮后, 系统自动提示是否确定结束此活动, 如果点击确定, 则结束活动。活动一旦结束将不可撤销, 在活动结束之后, 若再扫二维码签到, 则属于签到无效。



图 4 活动详细页



图 5 活动签到二维码

3.3.3 学生参与志愿活动

(1) 招募活动

招募活动页 (如图 6 所示) 列出了正在招募的活动信息列表, 信息包括活动标题和开始时间, 下拉页面可以查看新增的招募活动。点击右箭头可进入活动详细页面。

(2) 我的活动

我的活动页 (如图 7 所示) 列出了用户个人所参加的活动信息列表, 信息包括活动标题和开始时间, 下拉页面可以查看新增的我参与的活动信息。点击右箭头进入活动详细页面。



图 6 招募活动页



图 7 我的活动页

(3) 结束活动

结束活动页面 (如图 8 所示) 列出了用户过往已经参与的活动信息列表, 信息包括活动标题、开始时间和结束时间。下拉页面可以查看新增的结束活动信息。点击右箭头进入活动详细页面。

(4) 缺席活动

缺席活动页面 (如图 9 所示) 列出了用户报名参与但缺席的活动信息列表, 信息包括活动标题、开始时间、结束时间, 下拉页面可以查看新增的缺席活动信息。点击右箭头进入活动详细页面。



图 8 结束活动页



图 9 缺席活动页

(5) 活动详细

活动详细页面 (如图 10 所示) 列出了活动详细信息, 信息包括活动地点、发起者、组别名称、开始时间、参加人数、活动简历。若由招募活动进入此页面, 则下方提供报名按钮, 若点击此按钮, 系统会根据报名设置人数提示是否报名成功。若已报名成功, 再次进入活动详细页面报名按钮则变为已报名, 并且按钮不能点击。若不是从招募活动页面进入, 则不提供按钮, 仅显示活动的详细信息。

(6) 签到

用户点击 APP 右上角的手机图标按钮将启动签到功能, APP 会自动打开照相机并出现扫描二维码的区域。将手机对准生成的二维码扫描, 系统会自动提示扫描结果信息 (第一次签到成功)。当活动结束后, 必须再次启动扫描二维码功能进行第二次签到, 扫描成功后, 系统会自动提示结果信息点击确定即可, 一个活动必须进行二维码扫描两次, 否则无效, 在活动结束前进行第二次扫描签到后系统将自动计算成员此次活动的服务时长, 如图 11 所示。



图 10 活动详细页



图 11 第二次签到结束活动

3.3.4 相关代码

活动网络传输连接

```
public class GetActivityRequest
{
    public int IndexId;
    public long Tick;
}

public class ActivityRepositoryNet :
IActivityRepository
{
    protected HttpUtility Http { get; set; }
    protected String BaseUrl { get; set; }
    public ActivityRepositoryNet(String baseUrl)
    {
        Http = HttpUtility.Instance;
        BaseUrl = baseUrl;
    }

    public ActivityInfos GetActivityInfos(int ActivityID, long tick)
    {
        Return Http.PostJson<GetActivityRequest, ActivityInfos>(String.Format("http://{0}/{1}", BaseUrl, "GetActivity"),
            new GetActivityRequest { IndexId = ActivityID, Tick = tick });
    }

    public PullDownResult EndActivity(int ActivityID)
    {
        return Http.PostJson<GetActivityRequest, PullDownResult>(String.Format("http://{0}/{1}", BaseUrl, "FinishActivity"),
            new GetActivityRequest() { IndexId = ActivityID });
    }

    public NSACL.PullToRefreshData.PtrResponse<ActivityDB> GetActivitiesData(PullDownRequest postData)
```

```
{
    return Http.PostJson<PullDownRequest, PtrResponse<ActivityDB>>(String.Format("http://{0}/{1}", BaseUrl, "GetActivitiesByPage"), postData);
}

public PullDownResult AddApply(string ActivityID)
{
    return Http.PostJson<ApplyInfo, PullDownResult>(String.Format("http://{0}/{1}", BaseUrl, "JoinActivity"),
        new ApplyInfo() { ActivityID = ActivityID });
}

public ScanCodeResult ScanCode(string activityid)
{
    return Http.PostJson<ScanCodeRequest, ScanCodeResult>(String.Format("http://{0}/{1}", BaseUrl, "ScanCode"),
        new ScanCodeRequest()
        {ScanRequest = activityid });
}

public ScanCodeResult TwoScanCode(string activityid)
{
    return Http.PostJson<ScanCodeRequest, ScanCodeResult>(String.Format("http://{0}/{1}", BaseUrl, "TwoScanCode"),
        new ScanCodeRequest
        {ScanRequest = activityid});
}

活动二维码生成
[Activity(Label = "活动二维码", Icon = "@drawable/ic_launcher")]
public class CodeResult : Activity
{
    protected ImageView image;
    protected Handler handler { get; set; }
    public const String Activity_ID_Key = "Activity_ID_Key";
    protected String ActivityID { get; set; }
    protected long Tick { get; set; }
    protected override void OnCreate(Bundle bundle)
    {
        base.OnCreate(bundle);
        SetContentView(Resource.Layout.CodeResult);
        ActivityID = Intent.Extras.GetString(Activity_ID_Key);
        ImageView view = FindViewById<ImageView>(Resource.Id.image);
```

浅析 RFID 技术在图书馆中的应用

代一平*

DAI Yi-ping

摘要

随着物联网技术的发展, RFID 技术在图书馆的应用越来越广泛。本文以 RFID 技术为背景, 简要介绍了基于射频识别技术的图书管理系统的应用现状以及建设内容, 并分析射频识别技术在图书馆中应用的优劣势, 以及目前存在的一些问题。在未来, 图书馆管理系统必将依附于射频识别技术。

关键词

图书馆; RFID; 管理系统

doi: 10.3969/j.issn.1672-9528.2019.07.057

0 引言

射频识别(Radio frequency identification, 简称 RFID)起源于 19 世纪中叶, 这种技术可以不通过机械或者光学接触, 利用无线电信号识别特定目标, 并且读取其中信息, 不需要识别系统与特定目标建立机械或者光学接触^{[1][2]}。传统的 RFID 系统主要由电子标签(Tag)、阅读器(Reader)、计算机网络等部分组成。在实际使用中, 电子标签附着在待

识别物体表面, 当 Tag 进入阅读器天线的覆盖范围内, 阅读器可通过电子标签发出的无线电波, 读取并识别射频标签中所保存的信息, 从而识别不同物体^[3]。

射频识别技术相对于传统的条形码, 具有穿透性强、体积小、形状多样化、抗污染能力强、读取距离更远、可重复使用、数据的记忆容量大、快速多个同时扫描、安全性高的优点。目前, 射频识别技术已被广泛应用在身份认证、药品销售、交通、物流等领域^{[4][5]}。RFID 技术已是物联网核心技术之一, 是市场最为关注的技术。中国的 RFID 市场规模正在迅速扩大。本文主要研究射频识别技术在图书馆中的应用。

* 河南工业贸易职业学院实验实训中心 河南郑州 450012

```

    UriImageViewHelper.SetUrlDrawable(view,
    "http://172.16.100.41:8080/PersonCenter/Public/
    CreatCode/" + ActivityID);
    }
}

```

4 结论

根据 APP 的流程测试其功能运行情况, 对系统数据在系统生命周期的各个阶段进行检测, 以尽可能全面完整地判断系统的运行状况。采用多种测试方法, 测试结果表明系统数据存储稳定无差错, APP 运行流畅, 各功能之间没有冲突, 软件性能良好, 流程简单明了。

参考文献

- [1] 刘春妍. 关于加强高职院校社会实践类社团创新建设的几点思考[J]. 智库时代, 2019, (22): 262-263.
- [2] 肖翀, 黄柯达, 曾魏伟, 张跃进. 基于 Android 平台的

校园助手 APP 设计研究[J]. 数字通信世界, 2019, (04), 165
DOI:10.3969/J-ISSN.1672-7274.2019-04-131.

[3] 尹孟征. 浅析几种 APP 开发平台[J]. 通信电源技术, 2016, (33): 158-159.

[4] 周世明, 董芮奇. 基于 Android 平台的手机游戏设计与研究[J]. 计算机产品与流通, 2019, (06), 141.

[5] 雷英栋. 基于 SQLite 的实时动力学仿真平台模型数据库[J]. 科技视界, 2018, (30), 209-210+215.

【作者简介】

龚婷婷(1985-), 女, 江苏南通人, 硕士研究生, 讲师, 研究方向为软件系统开发、课程教学、教育信息化以及工作室制人才培养模式。

(收稿日期: 2019-06-10)