**《课程名称》**

音乐播放器

|  |  |
| --- | --- |
| 学生姓名 | 王少凡，刘祥宇 |
| 学 号 | 202183290004，202183290006 |
| 所在学院 | 计算机学院，网络空间安全学院 |
| 专业名称 | 计算机学院与技术 |
| 班 级 | 计科5班 |
| 任课教师 | 凌妙根 |

二○二四年六月

目录

**[基于android的音乐播放器设计 3](#_Toc14845)**

**[第一章Android开发基础 4](#_Toc25736)**

[1.1 什么是Android 4](#_Toc2031)

[1.2 Android系统的四大组件 6](#_Toc26326)

[1.3 Android音乐播放器工程 7](#_Toc6207)

**[第二章 系统需求分析 10](#_Toc23900)**

[2.1系统功能需求 10](#_Toc30443)

[2.2系统性能需求 12](#_Toc20665)

**[第三章 详细设计及功能实现 12](#_Toc27722)**

[3.1 播放器界面的设计 12](#_Toc2393)

[3.2 播放动画的设计 13](#_Toc25310)

[3.3 播放\暂停模块 14](#_Toc1128)

[3.4 上一曲\下一曲模块 15](#_Toc5359)

[3.5 歌手模块 16](#_Toc17819)

**[第四章 软件测试 19](#_Toc1985)**

[4.1 软件的测试 19](#_Toc5121)

[4.2 软件的测试环境 19](#_Toc18843)

[4.3 测试 19](#_Toc25197)

[4.4 软件测试的评价 19](#_Toc20740)

[4.5 测试阶段 19](#_Toc26695)

[4.6 测试结果 19](#_Toc3802)

**[总 结 20](#_Toc21949)**

**[参考文献 21](#_Toc4656)**

音乐播放器

摘要：当今社会发展越来越快，人们对手机软件设备的用户体验要求越来越高，本文主要是对Android系统上对音乐播放器应用程序的设计与实现进行讨论，通过这一过程的设计和论证对手机平台播放器以后的研发有很大意义。木文的音乐播放器使用开源的Android平台，使用Java 语言和开发工具android studio对音乐播放器代码进行编写。通过设计能够实现音乐的播放、暂停、上一曲、下一曲、音乐列表、声音控制、歌手歌单、播放模式等功能，通过对基于Android 平台下的播放器的设计，了解音乐播放器的设计原理和实现的相关技术，掌握Android编程思想以及基木的应用组件，对以后从事Android开发有很人的帮助。研究表明，基于Android平台的应用软件开发快捷简单，用户体验良好，必将成为未来移动互联网时代的风向标。

关键词：android 音乐播放器 应用程序

第一章Android开发基础

1.1 什么是Android

1.1.1 Android简介

Android一词的本义指“机器人”，同时也是Google于 2007年11月5日宣布的基于 Linux 平台的开源手机操作系统的名称。该平台由中间件、操作系统、应用软件和用户界面组成，是首个为移动终端打造的真正开放和完整的软件平台和操作系统。目前最新Android系统版本为Android 13。

Android是基于Linux内核的软件平台和操作系统，早期由谷歌公司研发(在华注册商标名为“安致”)，后由开放于机联盟（Open Handsct Alliancc）开发。Android采用了软件堆层的架构，由三个部分组成。底层以Linux为操作系统内核，提供进程管理、存储管理等基本功能。其他各公司自行开发市场需求的应用软件，部分程序以java 编写。为了使这项技术得到推广，2007年11月，Google 正式发布 Android 操作系统并联合34家厂商成立了“开放手机联盟”。Android在未公布之前常被传闻为谷歌电话。业内大多数认为Google开发的是自有手机品牌产品，而不是一套开源的软件平台。到了2010年1月，Google才发表了白有手机品牌电话Nexus One。2011年10月，Android市场占有率超过48%，成为全球最受欢迎、用户交互.体验较好、占有率最高的智能手机平台。Android手机软件平台为Android Market，为用户提供下载和安装支持Android的第三方软件。截止到2010年4月15日，Android Market已经有3.8万多个应用程序。现在这个移动互联网时代，Android系统在智能手机和平板电脑市场均处于霸主地位。采用Android系统主要厂商包括HTC、SAMSUNG、LG、魅族、华为、中兴、联想、小米等。目前最新SDK版本为Android 13。

1.1.2 Android的基本框架

Android的基本框架由五个部分组成，应用层、应用框架层、类库、Android运行时环境和 Linux内核。我们程序开发人员只可以访问应用层和应用框架层，使用开发所需Library库。

(1 )Applications

Application Android会与一个核心应用的程序包一起发布，此应用程序包会包括SMS短消息程序，email客户端，地图,浏览器，联系人管理，日历程序等。使用JAVA语言编写的核心应用程序。

(2)Application FrameWork

允许开发人员完全访问核心应用程序所使用的API框架。此应用程序架构用来简化组件的重用;在遵循系统安全性的要求下任何应用程序都可以发布和使用的它的功能块。此应用程序的重用机制使得组建可以被用户替换。所有的应用程序都由一系列的服务和系统组成,包括:可扩展的视图(Views )、内容管理器(Content Providers )、资源管理器（Resource Manager)、通知管理器（Notification Manager)、活动类管理器( Activity Manager)。

(3) Libraries库

Android包含了一套C/C++函数库113。该库通过Android的 Application FrameWork为开发人员提供所需的底层服务。以下是一些核心Libraries库:系统C库就是系统C函数库（libc)，为基丁embedded linux 的设备专门定制，也是从BSD继承而来的。媒体库基于 PacketVideoOpenCore，这个库支持录和放，可以录制大多数比较流行的音频与视频格式，还有静态映像文件包括 MP3，MPEG4，AAC，H.264,AMR,JPG,PNG。Surface Manager―管理所显示子系统，并且为多个应用程序提供2D图层和 3D图层的无缝融合。LibWebCore - 一个最新用来支持Android浏览器和一个可嵌入web视图的web浏览器引擎。SGL- 一个内置的支持2D图形的引擎。3D libraries-该库可以使用高度优化的3D软加速或者使用硬件3D加速。FreeType-位图和矢量字体显示。SQLite -功能强劲的安卓用轻量级关系型数据库，所有应用程序均可使用。

(4) Android Runtime

Android包括了一个核心的Library，该库提供了编程中Java所封装的大多数类和方法。每一个Android应用程序都在它自己独立的线程中运行，都拥有一个相对独立的Dalvik虚拟机实例[15]。Dalvik 是同时针对于多个高效运行的VMs来实现的。Dalvik虚拟机执行.dex 的可执行文件，这一格式文件对最小内存的使用做出了较好的优化。该Dalvik 虚拟机是基于寄存器的,所有的类都需要经过JAVA编译,然后通过AndroidSDK 中的dx工具转化成.dex格式再由虚拟机执行。

(5) Linux 内核

Android的核心系统服务依赖于Linux 2.6内核，如安全性，网络协议栈，内存管理，驱动模型和进程管理。Linux 内核也同时作为硬件和软件堆栈之间的硬件抽象层。同所有Linux内核一样，Android内核是介于硬件层和软件组之间的一个抽象层次。

1.2 Android系统的四大组件

在Android系统中，提供了四人基本组件，每个组件都是一个视图，分为可见和不可见视图，我们常见的Activity，ContentProvider在应用程序中是能用肉眼看见的，而 Service和 BroadcastReceiver通常则是在后台运行的。下面来详细介绍下四大组件:

( 1 )Activity简介

在Android系统中Activity提供可视化的用户界面，一个Android应用通常由多个Activity组成。多个Activity组成了Activity栈（Stack )，当前活动的Activity 处于栈顶。Activity有自己的生命周期，由Android系统来控制。

(2)Service简介

Service其实就是在后台运行的一种服务程序，用户一般不需要与其接触，所以这样的一个Service类其实是比较简单的，我们自己所写的service方法只要继承Service类，实现所需要的方法就可以了。写好的Service需要在AndroidManifest.xml文件中注册，就是通过<service></service>声明才能使用。Service有自己的生命周期，我们可以调用startService()启动一个Service或者bindService()方法来绑定一个存在的Service。

(3)Broadcast Receiver简介

Broadcast Receiver字面是广播接收器的意思，它类似于事件处理机制，不过事件处理机制是程序组件级别的，而广播事件处理机制是系统级别的。我们可以使用Intent来启动一个程序组件，还可以通过使用sendBroadcast()方法来发起一个系统级别的事件广播来传递消息。我们可以在应用程序中实现 Broadcast Receiver来监听和响应这些广播的Intent。

(4)ContentProvider简介

Content Provider的应用使应用程序之间相互访问数据成为可能，目前它被用来保存和检索数据，也成为了Android系统提供的第三方应用数据的访问方案。它是跨应用程序共享数据的唯一方法。Android系统为常用的音视频、图片和文件等提供了大量的Content Provider。它们在android.jar包的 android.provider下面。当我们有适当的权限的时候，通过这些定义好的ContentProvider 可以极其方便的进行数据操作。我们也可以自己来制定ContentProvider用来共享我们的数据，以方便用户的访问。

1.3 Android音乐播放器工程

1.3.1 Android项目

Android的四大基本组件:Activity(活动)，Service(服务)，BroadcastReceiver(广播接受者),ContentProvider(内容提供者y20。各组件之间要进行切换需要一个Intent类，通过启动 StartIntent(方法来使各组件进行跳转。

Android的五大布局方式: LinearLayout(线性布局)、FrameLayout(帧布局)、TableLayout(表格布局)、AbsoluteLayout(绝对布局)、RelativeLayout(相对布局)。这五种布局方式一般都是需要相互嵌套使用的，各种漂亮的Android界面就是通过嵌套布局才能够实现的，但是这一般都需要一定的Android布局功底。

Android的线程,以及线程之间通信的中介:Handler类。

Android配置文件:AndroidManifest.Xml及res资源文件中各种属性、布局方式、引用的资源等。

1.3.2 AndroidManifest.xml文件

AndroidManifest.xml 文件用来启动所有的Android应用程序，当我们一个新建的工程，都会自动生成一个名叫AndroidManifest.xml的文件，这一配置文件是整个程序能够正常运行的核心，其重要性不言而喻，它里面明确声明了Android SDK的最低需求版本minSdk Version，程序运行时使用的Activity都需要在这里注册声明，当程序中任何一个操作触发事件后，系统都会自动的去AndroidManifest.xml中寻找对应的标识来启动相应的操作。任何一个Andro idManifest.Xml文件的格式都应为:

<?xml vers ion="1.0" encoding="utf-8"?>

<manifest >

<application >

<activity >

<intent- filter>

<action/><category/></ intent- filter>

</activity>

</application>

<uses-sdk /><uses-permission />

</manifest>

Android系统默认支持的编码方式是"utf-8"，当程序显示出现乱码时，我们必须将其编码方式转化为"utf-8".在 AndroidManifest.xml中会在第一行声明一句:<?xmlversion="1.0" encoding="utf-8"?>。它是该系统支持的版本和编码。在第二行定义了我们需要使用的架构，该架构来自:

xmlns:andro id="http ://schemas.andro id.com apk/res/andro id"。

xmIns(XML namespace)是一个命名空间，它就如同一个引导者，带领我们找到系统框架，然后找到这个架构，每一个配置文件必须要由xmIns引导，否则我们想引用的资源就会无法使用。在application元素我们定义androidicon-"@drawab le/music"和android:label="@string/app\_name"，icon是程序安装后在手机中显示的图标，我们运用@drawable/mus ic表示了对res/drawable目录下的music.png的引用。@string/app\_name表示了对res/value目录下string.xml中定义的app\_name对应值的引用，在 string.xml定义健值对的方式易于以后的修改。

<activity

android:label="@string/app\_name"

android:name="com.tianhaoxyz. music.MainActivity"><intent- filter >

<action android:name="android. intent.action.MAIN" >

<category android:name="android. intent.cate gory.LAUNCHER">

</intent-filter>

</activity>

这是系统确定的首先要启动的Activity是 MainActivity。

在 intent-filter 选项中，有 action和 category 属性，按照要求 action 的值为android. intent.action.MAIN，category的值为 android.intent.category.LAUNCHER，它们是系统的值，程序启动的时候解析到这些常用值就可以确定首先要启动的 Activity。在AndroidManifest.Xml中有一个根结点<uses-sdk>,它是我们需要定义的ADT版本，例如我们将其值设定为 android:minSdk Version="10"，则表示此Androd 工程所用的版木为2.3.3，Android还有1.6、2.1、2.2、4.0、4.1这几个版木。

如果我们需要对音频文件进行刻录，那我们就需要设置音频刻录权限，我们就需

要将其的值设置为android:name="android.permission.RECORD\_AUDIO,我们就可以进行音频刻录了。当然uses-permission还有别的权限，例如GPS定位权限，网络访问权限，后台执行权限等等。

当程序的任何一个组件要用到时，都必须在.AndroidManifest.xml 文件中声明。四大组件要使用前都需要声明，其卢明方式如下:

<service></service>

<provider></pro vider><activity</activity><receiver</receiver>

其中声明provider 时还需要声明其授权authorities。每个组件里面还有很多的属性，会根据程序的具体需要定义不同的属性。过滤器intcnt- filtcr有两种启动方式来启动界面:

(1)显示方式

显示方式是在代码中的直接设置需要启动组件，在 intent.setClass(纽件1，组件2)中确定从这个组件跳转到另一个组件,然后调用startIntent( intent)来启动指定的组件2。

(2)隐式方式

隐式方式是通过过滤器来过滤满足action和 category的条件来启动该组件。这些属性是在.AndroidManifest.Xml文件中声明的。两种启动方式都可以通过程序setAction(ACTION常量)和 startActivity ( intent）或配置文件两种方式来实现组件的跳转。

第二章 系统需求分析

2.1系统功能需求

根据项日的日标，我们可获得项日系统的基本需求，以下从不同角度来描述系统的需求，并且.使用用例图来描述，系统的功能需求，我们分成两部分来概括，即播放器的基本控制需要和播放友好性需求。以下分别描述:

播放用例

用例名称:播放

参与者:用户

日标:用户点击播放音乐列表中的歌曲前置条件:播放器正在运行

基本事件流: 1.用户单击列表中歌曲

2.播放器将播放列表中的点击的歌曲

暂停用例

用例名称:暂停

参与者:用户

日标:使得用户可以暂停正在播放的歌曲前置条件:歌曲正在播放且未停止和暂停基本事件流: 1.用户单击“暂停”按钮

2.播放器将暂停当前的歌曲

初曲/终曲用例

用例名称:初曲/终曲

参与者:用户

目标:使得用户可以返回音乐列表初曲或终曲

前置条件:歌曲正在播放或暂停

基本事件流: l.用户单击“初曲或终曲”按钮

2.播放器将播放音乐列表中初一曲或终一曲

上一首/下一首用例

用例名称:上一首/下一首

参与者:用户

目标:使得用户可以点播.上一首或下一首音乐前置条件:歌曲正在播放或歌曲暂停中

基本事件流: 1.用户单击“上一首或下一首”按钮

2.播放器将播放上一首歌曲或下一首歌曲

音乐列表用例

用例名称:音乐列表

参与者:用户

目标:使得用户可以进入音乐列表

前置条件:程序在运行

基本事件流: 1.用户单击“音乐”分区

2.播放器进入音乐列表

用例名称:播放模式

参与者:用户

目标:使得程序在播放面板设定播放状态前置条件:程序运行在播放器音乐播放界面中

基本事什流:1.用户单击“顺序、单曲、随机”按钮

2.播放器进入选中模式播放状态

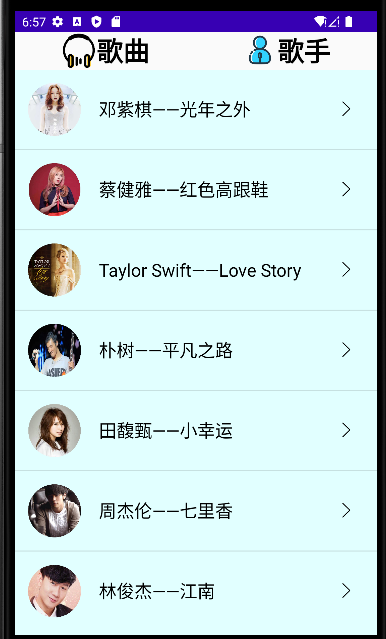
2.2系统性能需求

Android系统要求无响应时间为5秒，所以歌曲播放、暂停、上一曲/下一曲、初曲/终曲、进入播放器程序、进入音乐列表等等操作的系统响应时间最长均不能超过5秒。

第三章 详细设计及功能实现

3.1 播放器界面的设计

播放器的界面设计包括音乐播放界面、播放器主界面和播放器菜单托盘。音乐播放界面实现了声音进度条控制，歌曲信息获取，上一曲/下一曲，初曲/终曲，播放，暂停，单曲循环，顺序播放，随机播放，歌曲播放进度控制，播放进度时间获取。音乐播放界面整体是一个线性布局，里面是个大的相对布局，相对布局内部分布着三个线性布局依次排列。其中第一个线性布局获取显示歌曲信息、歌手信息，实现音量控制、播放模式选择;第二个线性布局实现歌词滚动播放;第三个线性布局实现播放器的基本功能、播放进度控制和播放进度时间的获取。其中播放按钮、暂停、上一曲、下一曲、播放模式、声音控制等功能在MusicActivity中实现，在按下和得到焦点的时候会做相应的改变。音乐播放界面的(包括各种控制按钮)布局代码具体见res\layout\music.xml文件中，播放界面如下图所示



播放器主界面包括四个部分，由五个activity 组成: MainActivity、ListActivity、ArtistsActivity、AlbumsActivity、SongsActivity。在resilayoutmain.xml中布局，通过MainActivity搭建好主界面框架，其它四个activity完成具体页面。由于本播放器页面布局是仿Android 原生播放器布局，主界面的四个模块分别为音乐、艺术家、专辑、最进播放。音乐模块获取音乐名称以及音乐持续时间列表排列;艺术家模块获取歌手、音乐名以及音乐持续时间列表排列;专辑模块获取专辑名、音乐名以及音乐持续时间列表排列;最近播放模块获取音乐名称以及音乐持续时间并且按照最近播放顺序列表排列。播放器菜单托盘的设计使用onCreateOptionsMenu方法和 onOptionsltemSelected方法，实现了播放器系统退出和软件版本信息查看。

3.2 播放动画的设计

在res\anim文件夹中专门放置动画文件，alpha\_z.xml文件在播放器音乐播放界面的歌词动态播放中使用。其调用动画关键代码为:

<alpha

xmIns:android="http://schemas.android.com/apk/res/android "android:interpolator="@android:anim/accclerate\_interpolator"

android:firomAlpha="0.0"

android:toAlpha="6.O"

android:startOfiset="700"

android:duration="l000"></alpha>

歌词动态显示时，随着歌曲播放进度歌词播放完毕后向上跳动一格变小，正在播放的歌词字体变最大且.颜色呈黄色。

3.3 播放\暂停模块

音乐播放界面能看到播放按钮，当点击播放按钮音乐播放，点击暂停按钮音乐暂停播放。当点击事件发生时，触发监听器执行处理操作，从而控制当前音乐的播放\暂停，其界面图如图4-1所示，关健处理机制如下所示:

//正在播放，点击暂停以后暂停播放

if (isPlaying == true){

Intent intent = new Intent(Music Activity.this,

MusicService.class);

intent.putExtra( "'play", "pause");startService( intent);

isPlaying = false;/l/播放状态改变为暂停

image BtnPlay.setImage Resource(R.dra wab le.playl );

replaying false;

}

//音乐哲停播放中，点击以后继续播放IfisPlaging == false ){

Intent intent = new Intent(MusicActivity.this,

MusicService.class);

intcnt.putExtra( "'play", "playing");intent.putExtra( "id", id);

startService( intent);//启动运行的服务isPlaying = true;/r播放状态改变为播放

imageBtnPlay.setImageResource(R.dra wable.pausel );

replaying-true;

}

3.4 上一曲\下一曲模块

当用户点击上一曲下一曲按钮时，监听器执行触发的事件，内部循环算法进行处理，从而播放上一曲\下一曲，其界面图如图4-1所示，关键处理逻辑如下所示:

//上一首的逻辑处理

//当前播放id-1播放

Int id=MusicService.\_id-1;i/id 比0大的时候id-1即可if(id>=lists.size(O)-1){

id=lists.size()-1;l/其它，id置0}

else if(id<=O){

id=O;}//下一首的逻辑处理//当前播放id+1播放

int id=MusicService.\_id+l;

if( id>=lists.size()-1){

id=lists.size()-1;

}else if(id<=0){

id=0;}



3.5 歌手模块

用户可以查看歌手信息，还有歌手的歌曲，其界面图如下图所示。

在播放器主界面点击艺术家图片按钮启动ArtistsActivity。在ArtistsAdapter.java文件中歌曲中的歌手名、歌曲名及音乐持续时间进行获取并且列表显示。艺术家模块包括歌手名、歌曲名以及歌曲时间。用户可以单击想要播放的音乐对其播放，跳转方式同音乐模块。关键代码如下:

//歌手名获取

TextView textMus icSingcr=(Text View) convert View.findViewByld(R. id.music\_item\_singer);

te xtMus icS inger.setText(m.getS inger());l音乐名获取

TextView textMusicName=(TextView) convertView.findViewByld(R. id.music\_item\_name );

textMusicName.setText(m.getName();l音乐持续时间获取

Text View textMusicTime=(TextView) convertView.findViewById(R. id.music\_item\_time);

textMusic Time.setText(to Time((int)m.get Time(O));



第四章 软件测试

4.1 软件的测试

一个程序并不可能达到十分完美，难免存在一些错误，就是我们常说的 bug。如果不能及时将其找出并修改，会造成系统崩溃并造成不必要的损失。通过软件测试，尽可能多的找出系统中的错误，以使于及时修改。

4.2 软件的测试环境

1、软件测试的性能要求

软硬件要求:系统环境 win7;此外对其他软件儿乎没有依赖性，程序健壮性较好。

4.3 测试

建立一张模拟的SD卡，上传歌曲文件;分别测试播放器的上/下一首、播放、暂停、初曲/终曲等功能用户体验是否良好。

4.4 软件测试的评价

软件功能评价:此音乐播放器软件各模块用户体验良好，基本符合设计要求。结论:此软件的设计和论证是可行的。

4.5 测试阶段

软件测试工程量大、过程复杂，在整个软件开发周期中占据着举足轻重的地位，需求调研和软什测试占据了软件开发三分之二时间。对于软什测试我们需要从不同的角度考虑，明确测试的目的，精细化测试的过程，对测试过程进行明确的分类，制定好测试计划，尽量做到细致以及全面的测试。按照软件测试的策略和过程分类，软件测试可分为单元测试、集成测试、确认测试、系统测试和验收测试。它们被依次顺序地执行，实验主要是进行功能测试。

4.6 测试结果

在课题后期进行了大力度的软件测试，按照测试的流程和规则，主要进行了功能测试。软件主要的 bug 表现为闪退、卡死等现象，在编辑器上观看log则能发现主要问题是运行时异常。我们主要对bug采取了异常拦截的方式进行调试完善，测试完成后软件稳定运行。

总 结

本文主要是对Android平台音乐播放器的分析设计，主要实现了音乐的播放、暂停、上一曲、下一曲、音乐列表、声音控制、退出菜单、播放模式等功能。在本文中第一章简要的介绍了选题依据背景、研究现状等;第二章中讲述了Android的开发基础;第三章进行了详尽的系统需求分析;第四章进行详细设计以及编码实现工作;最后的第五章进行了软件的测试工作，完善软件。本设计的音乐播放器软件能够稳定运行，体积小且反映灵敏，但还是有几点不足的地方，例如最近播放列表无法排序、代码编写冗余重复率高、图片素材布局不太美观等等。在以后我会改进软件的不足，以更优的编码水平，更高超的布局思维模型，并且学习更多新的知识重构这款软件。这款播放器软件还是比较好的，适用于低端智能手机市场，有良好的用户体验和反应速度。相信随着智能手机在发展中国家的普及，它们市场上会拥有庞大的用户群。

参考文献

[1] 赵世或，张盛，王玉辉，等．智能手机操作系统及其GoogleAndroid 上的软件开发[J]．煤炭技术，2011，30(04):197-198.

[2]朱婷婷,李惠.基于 Android的应用软件的综述[J].电脑与电信,2011,(01): 42-43.[3] LI Y,FENG G,Ll L,et al.Development and Research on Multimedia Application Based

on Android[J].Computer and Modernization,2011,(04):149-150.

[4]张正政，林耀荣.基于Android系统的影音播放器开发[J].现代电子技术，2011，

34(02):5-8.

[5]秦超,姜良君,蔡永香.基于Android的移动终端GPS测量系统的设计与开发[J].城

市勘测，2011，(01):64-67.

[6]刘昌平，范明钰，王光卫，等. Android手机的轻量级访问控制倡[J．计算机应用

研究，2010，27(07):20-25.

[7]刘仙艳．移动终端开放平台—Andro id[J].信息通信技术，2011,(04): 40-50.[8]党李成．基于GoogleAndro id智能手机平台的研究与应用[D]．安徽大学计算机科

学与技术学院，2010:9-12.

[9]贾理华，乐燕芬，施伟斌．基于Android平台简易即时通讯的研究与设计[J．常

州工学院学报，2010,(04):9-12.

[10]农丽萍，王力虎，黄一平.Android在嵌入式车载导航系统的应用研究[J].计算机工程与设计，2010，31(11): 2473-2476.