江西理工大学应用科学学院

手机APP用户手册

[项目名称]

**系 别： 信息工程系**

**班 级： 嵌入式16级**

**开发单位： 疯子传说**

**技术总监： 谢小云老师**

**时 间： 2019年6月22日**

文档编号 3.3

版本号 V1.0

文档名称 手机APP用户手册

项目名称 现代农业综合管理系统

项目负责人 郭凯

编写 文福金 年 月 日

校对 年 月 日

批准 年 月 日

开发单位 疯子传说  年 月 日

**版 本 历 史**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **版本/状态** | **作者** | **参与者** | **起止日期** | **备注** |
| V.00 | 文福金 |  | 2016/06/10-2016/06/22 |  |
|  |  |  |  |  |
|  |  |  |  |  |

**内容提要**

手机APP主要是访问服务器，调用服务器的相关功能，也主要涵盖农业生产管理、农业气象、大数据分析等模块。

（1）用户管理功能：具有用户注册、用户信息修改、设备注册与绑定等功能。

（2）数据查询与展示功能：向用户提供查询界面，可查询渔、林数据的信息。可以从服务器中获的设备相关的数据，实现与Web端完全相同的数据展示和应用功能。

（3）数据应用功能：与Web端类似，可基于存储的数据，可以完成简单的应用，如产量统计、能耗统计等。

（4）数据展示功能，实现各类数据的展示，实时刷新。

（5）个人信息展示，从服务器上获取个人信息并展示在手机界面上。

（6）远程控制，实现对硬件设备的远程控制。

**使用说明**

本手册使用PDF格式，用户可以安装可阅读PDF文档的相关工具软件，推荐安装Adobe Acrobat Reader。文档可通过Adobe Acrobat Reader等软件的“标签”进行主题的快速跳转。此外，请注意文档中所使用的以下符号：

**版本说明**

使用手册版本：V0.01

**版权申明**

本软件由XXXXXXX开发，任何个人和单位需要使用本软件，均应获得授权后才能使用，否则视为侵犯版权。

**目 录**

[1、系统功能特点与结构 2](#_Toc12400)

[1.1 主要功能 2](#_Toc4330)

[1.2 系统特点 2](#_Toc9915)

[1.3 系统结构 2](#_Toc9534)

[2、系统部署 3](#_Toc28911)

[Android 7.0 3](#_Toc10788)

[2.1 安装要求 3](#_Toc20983)

[2.1.1 硬件配置 3](#_Toc27029)

[2.1.2 系统软件配置 3](#_Toc6444)

[2.2 部署步骤 3](#_Toc31747)

[2.2.1安装方法 3](#_Toc16339)

[3、使用方法 3](#_Toc11223)

[3.1 用户界面结构 4](#_Toc3105)

[3.2 使用方法 7](#_Toc24109)

[5、出错处理和恢复 7](#_Toc12155)

[6、系统升级与维护 7](#_Toc17581)

[7、 代码部分 7](#_Toc2931)

[7.1、服务器前端Android APP部分的代码 7](#_Toc32516)

[1、清单文件：AndroidManifest.xml 7](#_Toc31706)

[2、build.gradle 9](#_Toc12586)

[3、公有类：common文件 10](#_Toc13938)

[（1）登录界面 10](#_Toc31967)

[（2）注册界面 21](#_Toc5368)

[（3）忘记密码界面 35](#_Toc2)

[4、数据展示模块:Weather\_Station.java 39](#_Toc1174)

[（1）数据展示 InitDatePage()： 39](#_Toc18096)

[（2）数据刷新setDataListenerProcess()： 42](#_Toc28178)

[（4） Activity\_weather\_station.xml 44](#_Toc22348)

[5、个人信息展示模块:PersonalData.java 57](#_Toc17449)

[（1）个人资料展示：init.class 57](#_Toc28644)

[（2）个人资料管理：save.class 60](#_Toc4592)

[6、 专家系统建议模块 62](#_Toc1926)

[（1）ExperSystem.java 62](#_Toc28993)

[（2）expert\_system.java 64](#_Toc31626)

[7、智能远程控制 67](#_Toc20851)

[8、服务器连接 69](#_Toc20442)

[9、Utils 81](#_Toc30961)

**1、系统功能特点与结构**

1.1 主要功能

(1) 登录

(2) 注册

(3) 修改密码

(4) 记住密码

1. 气象站模块的数据展示，实时刷新数据、数据折线图
2. 渔林模块的数据展示，实时刷新数据、数据折线图

(7) 个人信息的设置、修改与查看

(8) 智能远程控制

(9) 主页的相关网络信息展示

(10) 退出登录

1.2 系统特点

本APP针对大多数用户的需求，适应现代农业综合管理系统的发展，通过本app可查看与气象站的气象数据，以及渔林相关的所有可测量计算数据，并根据需求实时刷新数据信息。还可以对远程设备进行简单的控制。查看用户信息及设置。极大的实现了APP的智能化、自动化、及安全性、可靠性以及简单易用性。

1.3 系统结构

App主要包含了气象站、渔林数据展示、专家建议系统、个人信息、智能远程控制模块。如图所示。



图1.3.1 App的使用流程

**2、系统部署**

Android 7.0

2.1 安装要求

2.1.1 硬件配置

（1）需要系统存储权限

（2）具有网络功能

2.1.2 系统软件配置

无需。

2.2 部署步骤

2.2.1安装方法

(1) 下载APK文件

(2) 使用系统默认软件打开APK文件

(3) 选择安装

(4) 安装成功

**3、使用方法**

**3.1 用户界面结构**

(1) 登录界面

用户输入用户名，用户密码进行登录。为了

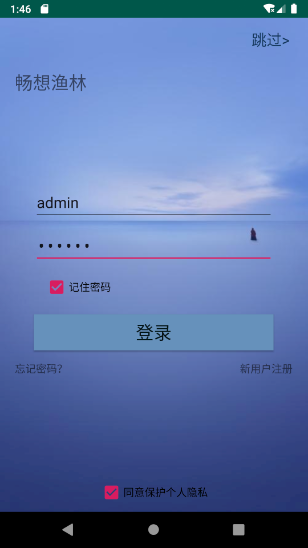


图3.1.1 用户登录界面

（2）用户忘记密码、注册界面

用户输入用户名、用户密码、性别选择，用户邮箱，并同意了协议条款的前提下进行注册。

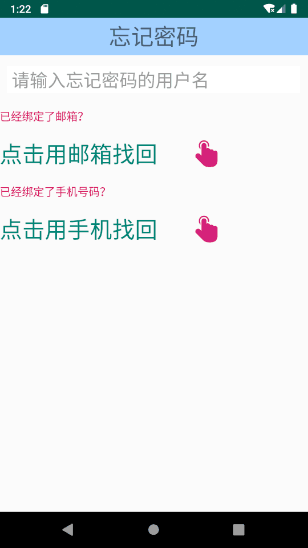
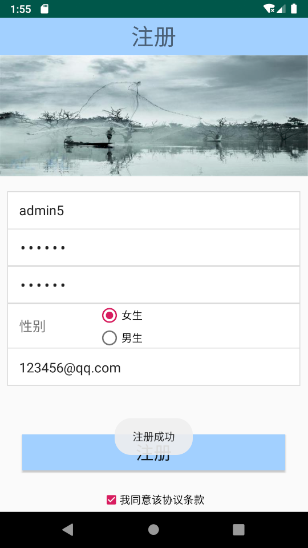
  

图3.1.2 用户忘记密码、注册界面

1. 数据展示页面

在该页面用户可以根据需要点击刷新按钮，立即刷新数据。

图3.1.3 数据展示页面

1. 首页浏览、个人信息展示

登录成功后，进入App首页，点击左上角图标显示系统功能。其中，在个人资料内，可对用户自己的信息进行管理。

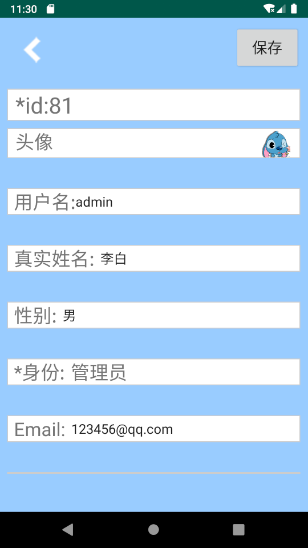
  

图3.1.4 首页浏览、个人信息展示

1. 专家系统建议

点击底部导航栏“数据”控件，进入专家系统页面，可对各项数据进行修改及设定。

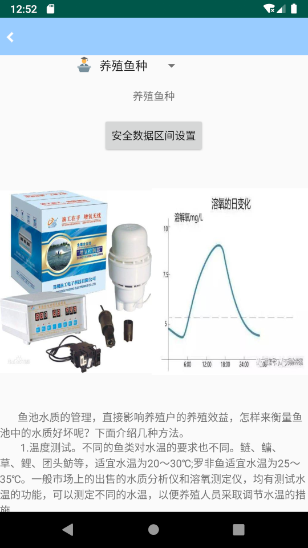
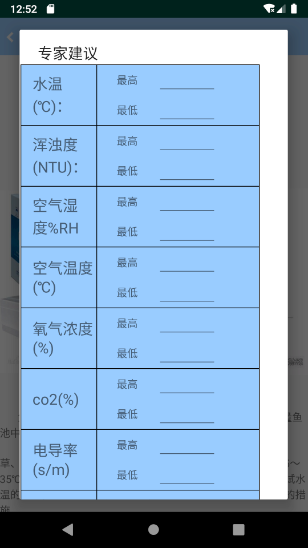
 

图3.1.5 专家系统建议

1. 智能远程控制

为了方便用户对设备的远程管理，在App上还提供了简单的远程控制界面，可以对设备进行简单的控制。当网络连接后，数据也自动联网，实时更新，用户可以在App中点击关闭设备，选择开启或关闭设备，远程控制设备的开关。下面是智能远程控制页面如图所示。

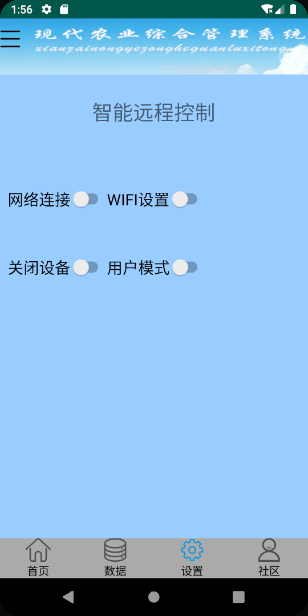


图3.1.6 智能远程控制

3.2 使用方法

(1) 用户在登录界面输入用户名和密码进行登录。

(2) 如果没有用户名和密码的可以在注册页面进行注册，然后再进行登录。

**5、出错处理和恢复**

（1）注册页面未反应：可能是按钮未点击到，请重新注册。或者是服务器状态，等待服务器重新开启。

（2）主页页面内容空白，可能是服务器未连接，等待服务器重新开启。

（3）数据点击刷新后，没有出现请求成功提示，可能是没有点击到或者网络连接异常。

**6、系统升级与维护**

介绍系统需要升级及进行日常维护的方法、注意事项等。

1. **代码部分**

**7.1、服务器前端Android APP部分的代码**

**1、清单文件：AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>

<manifest xmlns:android="http://schemas.android.com/apk/res/android"

package="com.mywork.myapplication">

<uses-permission android:name="android.permission.READ\_EXTERNAL\_STORAGE"/>

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

<uses-permission android:name="android.permission.INTERNET"/>

<!--允许网络通信-->

<uses-permission android:name="android.permission.INTERNET" />

<uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE" />

<uses-permission android:name="android.permission.CAMERA"/>

<!-- 往SDCard写入数据权限 -->

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>

<uses-permission android:name="android.permission.MOUNT\_UNMOUNT\_FILESYSTEMS"/>

<!-- 打电话权限 -->

<uses-permission android:name="android.permission.CALL\_PHONE"/>

<uses-permission android:name="android.permission.CAMERA" />

<uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE" />

<!-- 屏幕适配 -->

<supports-screens

android:largeScreens="true"

android:normalScreens="true"

android:anyDensity = "true"/>

<!--android:name=".weather\_station.myApplication"-->

<application

android:name=".weather\_station.myApplication"

android:allowBackup="true"

android:usesCleartextTraffic="true"

android:icon="@mipmap/icons"

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

android:roundIcon="@mipmap/test"

android:supportsRtl="true"

android:theme="@style/AppTheme">

<activity android:name=".login.LoginActivity">

<intent-filter>

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

<category android:name="android.intent.category.LAUNCHER" />

</intent-filter>

</activity>

<activity android:name=".login.RegisterActivity" />

<activity android:name=".Fishing\_Forest.Fishing\_ForestActivity" />

<activity android:name=".login.LostFindActivity" />

<activity android:name=".weather\_station.Weather\_StationActivity" />

<activity android:name=".weather\_station.ExpertSystem" />

<activity android:name=".weather\_station.PersonalDate" />

<activity android:name=".weather\_station.BaseActivity" />

<activity android:name=".weather\_station.ScanWeather\_Station" />

<activity android:name=".login.AgreementActivity" />

<activity android:name=".common.HeadActivity" />

<activity android:name=".common.book\_info" />

<activity android:name=".common.book\_list" />

<activity android:name=".common.search" />

<activity android:name=".setting.cancel" />

<activity android:name=".setting.setting" />

<activity android:name=".setting.contact" />

<activity android:name=".setting.createbook" />

<activity android:name=".weather\_station.UsernameChange" />

<activity android:name=".common.SystemSettingActivity" />

<activity android:name=".setting.photo" />

</application>

</manifest>

**2、build.gradle**

apply plugin: 'com.android.application'

android {

compileSdkVersion 28

defaultConfig {

applicationId "com.mywork.myapplication"

minSdkVersion 18

targetSdkVersion 28

versionCode 1

versionName "1.0"

testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"

}

buildTypes {

release {

minifyEnabled false

proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'

}

}

useLibrary 'org.apache.http.legacy'

}

dependencies {

implementation fileTree(dir: 'libs', include: ['\*.jar'])

implementation 'com.android.support:appcompat-v7:28.0.0'

implementation 'com.android.support.constraint:constraint-layout:1.1.3'

testImplementation 'junit:junit:4.12'

androidTestImplementation 'com.android.support.test:runner:1.0.2'

androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'

}

**3、公有类：common文件**

**（1）登录界面**

**逻辑代码：LoginActivity.java**

public class LoginActivity extends Activity {

private static boolean mBackKeyPressed=false; //记录是否首次点击按键

private String userName,psw,spPsw;//获取的用户名，密码，加密密码

private EditText et\_user\_name,et\_psw;//编辑框

private CheckBox rem\_pw;

private CheckBox checkBox2;

private TextView tv\_lgi\_hop;

private SharedPreferences sp;

public static String cookieVal="";

int flag;

@Override

protected void onCreate(Bundle savedInstanceState){

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

//设置此界面为竖屏

setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_PORTRAIT);

init();

}

//获取界面控件

private void init() {

//从main\_title\_bar中获取的id

//从activity\_login.xml中获取的

TextView tv\_register = (TextView) findViewById(R.id.tv\_register);

TextView tv\_find\_psw = (TextView) findViewById(R.id.tv\_find\_psw);

Button btn\_login = (Button) findViewById(R.id.btn\_login);

et\_user\_name= (EditText) findViewById(R.id.et\_user\_name);

et\_psw= (EditText) findViewById(R.id.et\_psw);

sp = this.getSharedPreferences("userInfo", Context.MODE\_PRIVATE);

rem\_pw = (CheckBox) findViewById(R.id.checkbox1); //获取记住密码框

checkBox2 = (CheckBox) findViewById(R.id.checkBox2); //获取记住密码框

tv\_lgi\_hop=(TextView) findViewById(R.id.lgi\_tv\_hop);

//案键 跳过 点击事件

tv\_lgi\_hop.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

Intent intent1=new Intent(LoginActivity.this, Weather\_StationActivity.class); intent1.putExtra("userkey","{\"user\":{\"face\":\"http://10.168.14.3:8080/headport/92eee6c96ffb4f0395b329f8faf017cf.jpeg\",\"identity\":\"试用身份\",\"sex\":\"男\",\"name\":\"李白\",\"id\":81,\"email\":\"123456@qq.com\",\"hobby\":null,\"username\":\"试用用户\"}}");

startActivity(intent1);

finish();

}

});

//监听协议多选框按钮事件

checkBox2.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

public void onCheckedChanged(CompoundButton buttonView,boolean isChecked) {

if (checkBox2.isChecked()) {

/\* System.out.println("协议已选中");

Intent intent=new Intent(LoginActivity.this, AgreementActivity.class);\*/

/\*Toast.makeText(LoginActivity.this,"记住密码已选中", Toast.LENGTH\_SHORT).show();\*/

sp.edit().putBoolean("ISCHECK1", true).commit();

/\* startActivityForResult(intent, 1);\*/

}else {

System.out.println("记住密码没有选中");

/\*Toast.makeText(LoginActivity.this,"记住密码没有选中", Toast.LENGTH\_SHORT).show();\*/

sp.edit().putBoolean("ISCHECK1", false).commit();

}

}

});

//判断记住协议多选框的状态

if(sp.getBoolean("ISCHECK1", false)) {

//设置默认是记录密码状态

checkBox2.setChecked(true);

}

//监听记住密码多选框按钮事件

rem\_pw.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

public void onCheckedChanged(CompoundButton buttonView,boolean isChecked) {

if (rem\_pw.isChecked()) {

System.out.println("记住密码已选中");

/\*Toast.makeText(LoginActivity.this,"记住密码已选中", Toast.LENGTH\_SHORT).show();\*/

sp.edit().putBoolean("ISCHECK", true).commit();

}else {

System.out.println("记住密码没有选中");

/\*Toast.makeText(LoginActivity.this,"记住密码没有选中", Toast.LENGTH\_SHORT).show();\*/

sp.edit().putBoolean("ISCHECK", false).commit();

}

}

});

//判断记住密码多选框的状态

if(sp.getBoolean("ISCHECK", false)) {

//设置默认是记录密码状态

rem\_pw.setChecked(true);

et\_user\_name.setText(sp.getString("USER\_NAME", ""));

et\_psw.setText(sp.getString("PASSWORD", ""));

}

//立即注册控件的点击事件

tv\_register.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

//为了跳转到注册界面，并实现注册功能

Intent intent=new Intent(LoginActivity.this, RegisterActivity.class);

startActivityForResult(intent, 1);

}

});

//找回密码控件的点击事件

tv\_find\_psw.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

startActivity(new Intent(LoginActivity.this, LostFindActivity.class));

}

});

//登录按钮的点击事件

btn\_login.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

//开始登录，获取用户名和密码 getText().toString().trim();

LoginActivity.this.userName = et\_user\_name.getText().toString().trim();

LoginActivity.this.psw = et\_psw.getText().toString().trim();

final ByteArrayOutputStream BufferedReader = new ByteArrayOutputStream();

new Thread() {

@Override

public void run() {

// TODO Auto-generated method stub

super.run();

String inputLine = null;

if (TextUtils.isEmpty(userName)) {

showToast(LoginActivity.this, "请输入用户名");

/\*Toast.makeText(LoginActivity.this, "请输入用户名", Toast.LENGTH\_SHORT).show();\*/

} else if (TextUtils.isEmpty(psw)) {

showToast(LoginActivity.this, "请输入密码");

// md5Psw.equals(); 判断，输入的密码加密后，是否与保存在SharedPreferences中一致

}else if (!checkBox2.isChecked()) {

showToast(LoginActivity.this, "首次见面，请确认协议");

}

else{

try {

HttpURLConnection connection = (HttpURLConnection) new URL("http://10.168.14.3:8080/auth/applogin").openConnection();

connection.setRequestMethod("POST");

connection.setDoInput(true);

connection.setDoOutput(true);

connection.setUseCaches(false);

connection.setConnectTimeout(3000);

connection.setRequestProperty("Content-type", "application/json;charset=UTF-8");

DataOutputStream outputStream = new DataOutputStream(connection.getOutputStream());

Map params = new HashMap();

params.put("username", userName);

params.put("password", psw);

outputStream.write(new JSONObject(params).toString().getBytes()); // POST

connection.connect(); // GET

if (connection.getResponseCode() == HttpURLConnection.HTTP\_OK) {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(connection.getInputStream()));

/\*Toast.makeText(LoginActivity.this,"登陆成功", Toast.LENGTH\_SHORT).show();\*/

while ((inputLine = bufferedReader.readLine()) != null) {

if (rem\_pw.isChecked()) {

// 记住用户名、密码

SharedPreferences.Editor editor = sp.edit();

editor.putString("USER\_NAME", userName);

editor.putString("PASSWORD", psw);

editor.commit();

}

// 跳转界面

Intent intent = new Intent(getApplicationContext(), Weather\_StationActivity.class);

Bundle bundle = new Bundle();

bundle.putString("userkey", String.valueOf(inputLine));

intent.putExtras(bundle);

finish();

startActivity(intent);

showToast(LoginActivity.this, "登录成功");

}

bufferedReader.close();

} else {

Log.i("reault","run");

showToast(LoginActivity.this, "输入的用户名和密码不一致");

}

connection.disconnect();

} catch (Exception e) {

e.printStackTrace();

showToast(getApplicationContext(), "连接网络失败");

}

}

}

}.start();

}

});

}

@Override

public void onBackPressed() {

if(!mBackKeyPressed) {

AlertDialog.Builder builder = new AlertDialog.Builder(LoginActivity.this);

builder.setTitle("提示!").setMessage("确定退出登陆！！！").setPositiveButton("确定",

new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

finish();

}

}).setNegativeButton("取消", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

}

}).show();

}

}

public boolean onTouchEvent(MotionEvent event) {

if(null != this.getCurrentFocus()){

/\*\*

\* 点击空白位置 隐藏软键盘

\*/

InputMethodManager mInputMethodManager = (InputMethodManager) getSystemService(INPUT\_METHOD\_SERVICE);

return mInputMethodManager.hideSoftInputFromWindow(this.getCurrentFocus().getWindowToken(), 0);

}

return super .onTouchEvent(event);

}/\*\*

\* 在线程中正常使用吐司

\*\*/

private static Toast toast = null;

public static void showToast(Context context, String text) {

Looper myLooper = Looper.myLooper();

if (myLooper == null) {

Looper.prepare();

myLooper = Looper.myLooper();

}

//设置toast的显示位置

WindowManager windowManager = (WindowManager) context.getSystemService(WINDOW\_SERVICE);

Point size = new Point();

windowManager.getDefaultDisplay().getSize(size);

toast = Toast.makeText(context, text, Toast.LENGTH\_LONG);

toast.setGravity(Gravity.BOTTOM, 0, size.y / 9);

toast.show();

if ( myLooper != null) {

Looper.loop();

myLooper.quit();

}

}

}

**配置文件:activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>

<!--登录界面,用LinearLayout-->

<android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:background="@mipmap/bg"

android:orientation="vertical">

<LinearLayout

android:id="@+id/title1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="35dp">

<ImageView

android:id="@+id/symbol"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" />

<TextView

android:id="@+id/app"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="20dp"

android:layout\_marginTop="35dp"

android:text="畅想"

android:textSize="24sp" />

</LinearLayout>

<ImageView

android:id="@+id/lgi\_iv\_logo"

android:layout\_width="80dp"

android:layout\_height="80dp"

android:layout\_centerHorizontal="true"

android:layout\_gravity="center"

android:layout\_marginTop="20dp"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintHorizontal\_bias="0.5"

app:layout\_constraintStart\_toStartOf="parent"

app:layout\_constraintTop\_toTopOf="@+id/et\_user\_name"

app:srcCompat="@mipmap/test" />

<!--输入框-->

<EditText

android:id="@+id/et\_user\_name"

android:layout\_width="320dp"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:layout\_marginTop="20dp"

android:hint="用户名 "

android:textSize="20sp" />

<!--输入框-->

<EditText

android:id="@+id/et\_psw"

android:layout\_width="320dp"

android:layout\_height="wrap\_content"

android:layout\_gravity="center"

android:layout\_marginTop="10dp"

android:hint="密码"

android:textSize="20sp"

android:inputType="textPassword"/>

<LinearLayout

android:id="@+id/linearLayout3"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignLeft="@+id/linearLayout2"

android:layout\_alignRight="@+id/linearLayout2"

android:layout\_below="@+id/linearLayout2"

android:layout\_marginTop="15dp"

android:orientation="horizontal" >

<CheckBox

android:id="@+id/checkbox1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/textview2"

android:layout\_marginLeft="60dp"

android:text="记住密码"

android:textColor="#000000" />

</LinearLayout>

<!--按钮-->

<Button

android:id="@+id/btn\_login"

android:text="登录"

android:background="#1E90FF"

android:textSize="24sp"

android:layout\_gravity="center"

android:layout\_marginTop="20dp"

android:layout\_width="320dp"

android:layout\_height="wrap\_content"/>

<RelativeLayout

android:layout\_marginTop="15dp"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content">

<TextView

android:id="@+id/tv\_register"

android:layout\_alignParentRight="true"

android:layout\_marginRight="20dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="新用户注册"/>

<!--layout\_weight="1" layout\_width="0dp"实现均分效果-->

<TextView

android:id="@+id/tv\_find\_psw"

android:layout\_alignParentLeft="true"

android:layout\_marginLeft="20dp"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="忘记密码？" />

</RelativeLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:layout\_centerHorizontal="true"

android:orientation="horizontal"

android:layout\_marginBottom="10dp">

<CheckBox

android:id="@+id/checkBox2"

android:layout\_width="wrap\_content"

android:layout\_alignParentBottom="true"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:text="同意保护个人隐私"

android:layout\_marginBottom="10dp"

/>

</RelativeLayout>

</LinearLayout>

<TextView

android:id="@+id/lgi\_tv\_hop"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="16dp"

android:layout\_marginEnd="24dp"

android:layout\_marginRight="24dp"

android:text="跳过>"

android:textSize="20dp"

android:textColor="#14395E"

app:layout\_constraintEnd\_toEndOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</android.support.constraint.ConstraintLayout>

**（2）注册界面**

**逻辑代码: RegisterActivity.java**

/\*注册\*/

public class RegisterActivity extends Activity {

//用户名，密码，再次输入的密码的控件的获取值

private String userName,psw,pswAgain,Email;

private RadioGroup Sex;

/\* private final String TAG = getClass().getSimpleName();

private IntentFilter intentFilter1;

private IntentFilter intentFilter2;

private final static String SENT\_SMS\_ACTION = "SENT\_SMS\_ACTION";

private final static String DELIVERED\_SMS\_ACTION = "DELIVERED\_SMS\_ACTION";\*/

private Handler handler;

//判断输入框内容

public static Pattern p =

Pattern.compile("\\w+([-+.]\\w+)\*@\\w+([-.]\\w+)\*\\.\\w+([-.]\\w+)\*");

private EditText et\_user\_name; // 输入用户名

private EditText et\_psw; // 输入密码

private EditText et\_psw\_again; // 确认密码

private EditText email; // 邮箱

public Button btn\_register; // 注册按钮

private CheckBox checkbox; // 协议同意框

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

//设置页面布局 ,注册界面

setContentView(R.layout.activity\_register);

//设置此界面为竖屏

setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_PORTRAIT);

// 初始化界面

setupViews();

handler = new Handler();

final ImageView imageview = (ImageView) findViewById(R.id.imgIcon);

checkbox = (CheckBox) findViewById(R.id.checkbox1);

// 给协议按钮添加监听

checkbox.setOnCheckedChangeListener(new CompoundButton.OnCheckedChangeListener() {

@Override

public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {

// TODO Auto-generated method stub

if (isChecked) {

btn\_register.setVisibility(View.VISIBLE);

imageview.setVisibility(View.GONE);

Intent intent = new Intent(RegisterActivity.this, AgreementActivity.class);

startActivity(intent);

} else {

btn\_register.setVisibility(View.INVISIBLE);

imageview.setVisibility(View.VISIBLE);

}

}

});

}

private void setupViews() {

//从activity\_register.xml 页面中获取对应的UI控件

btn\_register = (Button) findViewById(R.id.btn\_register);

et\_user\_name= (EditText) findViewById(R.id.et\_user\_name);

et\_psw= (EditText) findViewById(R.id.et\_psw);

et\_psw\_again= (EditText) findViewById(R.id.et\_psw\_again);

Sex= (RadioGroup) findViewById(R.id.SexRadio);

email=(EditText)findViewById(R.id.et\_eamil);

//点击事件

btn\_register.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

//获取输入在相应控件中的字符串

// 获取用户名和用户密码

final String username = et\_user\_name.getText().toString().trim();

final String psw = et\_psw.getText().toString().trim();

final String pswAgain = et\_psw\_again.getText().toString().trim();

/\*final String sex = et\_psw\_again.getText().toString().trim();\*/

final String Email = email.getText().toString().trim();

Sex= (RadioGroup) findViewById(R.id.SexRadio);

final int sex;

final int sexChoseId = Sex.getCheckedRadioButtonId();

switch (sexChoseId) {

case R.id.mainRegisterRdBtnFemale:

sex = 0;

break;

case R.id.mainRegisterRdBtnMale:

sex = 1;

break;

default:

sex = -1;

break;

}

//注册按钮

switch (v.getId()) {

case R.id.btn\_register:

if (pswAgain.equals(psw)) {

new Thread() {

@Override

public void run() {

// TODO Auto-generated method stub

super.run();

Object result;

//调用编写的isEmail()函数

//获取到输入的内容并转换为字符串形式，这样才可以进行匹配，且邮箱格式不能太长，字符串长度31以内

//按照实际需要可以添加不同的事件

if (TextUtils.isEmpty(username)) {

showToast(RegisterActivity.this,"请输入用户名");

} else if (TextUtils.isEmpty(psw)) {

showToast(RegisterActivity.this,"请输入密码");

} else if (psw.length() > 16 || psw.length() <5) {

showToast(RegisterActivity.this,"请输入6-15位密码");

}else if (TextUtils.isEmpty(pswAgain)) {

showToast(RegisterActivity.this,"请再次输入密码");

}else if (TextUtils.isEmpty(Email)) {

showToast(RegisterActivity.this,"请输入邮箱");

} else if (!(isEmail(email.getText().toString().trim()) && email.getText().toString().trim().length()<=31)){

showToast(RegisterActivity.this,"邮箱格式错误");

}else {

try {

HttpURLConnection connection = (HttpURLConnection) new URL("http://10.168.14.3:8080/auth/regist").openConnection();

connection.setRequestMethod("POST");

connection.setDoInput(true);

connection.setDoOutput(true);

connection.setUseCaches(false);

connection.setRequestProperty("Content-type", "application/json;charset=UTF-8");

DataOutputStream outputStream = new DataOutputStream(connection.getOutputStream());

Map params = new HashMap();

params.put("username", username);

params.put("password", psw);

params.put("sex", (sex == 0 ? "女" : "男"));

params.put("email", Email);

outputStream.write(new JSONObject(params).toString().getBytes()); // POST

connection.connect(); // GET

if (connection.getResponseCode() == HttpURLConnection.HTTP\_OK) {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(connection.getInputStream()));

String inputLine = null;

while ((inputLine = bufferedReader.readLine()) != null) {

Log.i("result", inputLine);Intent intent = new Intent(RegisterActivity.this, LoginActivity.class);

showToast(RegisterActivity.this,"注册成功");

startActivity(intent);

finish();

}

bufferedReader.close();

}

showToast(RegisterActivity.this,"用户名重复，请重新输入");

connection.disconnect();

} catch (IOException e) {

e.printStackTrace();

}

}

}

}.start();

} else {

Toast.makeText(RegisterActivity.this, "密码不一致，请重新输入！", Toast.LENGTH\_SHORT).show();

et\_psw\_again.setText(null);

}

}

}

});

et\_user\_name.setText("");

et\_user\_name.requestFocus();

et\_user\_name.addTextChangedListener(new TextWatcher() {

@Override

public void onTextChanged(CharSequence arg0, int arg1, int arg2, int arg3) {

// TODO Auto-generated method stub

refreshViews(arg0);

}

@Override

public void beforeTextChanged(CharSequence arg0, int arg1, int arg2, int arg3) {

// TODO Auto-generated method stub

}

@Override

public void afterTextChanged(Editable arg0) {

// TODO Auto-generated method stub

}

});

}

//验证函数优化版

public static boolean isEmail(String email){

if (null==email || "".equals(email)) return false;

Matcher m = p.matcher(email);

return m.matches();

}

public boolean onTouchEvent(MotionEvent event) {

if (null != this.getCurrentFocus()) {

/\*\*

\* 点击空白位置 隐藏软键盘

\*/

InputMethodManager mInputMethodManager = (InputMethodManager) getSystemService(INPUT\_METHOD\_SERVICE);

return mInputMethodManager.hideSoftInputFromWindow(this.getCurrentFocus().getWindowToken(), 0);

}

return super.onTouchEvent(event);

}

// 根据edittext的内容来判断是否应当出现“清除所有”的按钮x

private void refreshViews(CharSequence s) {

if (s.length() > 0) {

btn\_register.setEnabled(true);

} else {

btn\_register.setEnabled(false);

}

}

/\*\*

\* 在线程中正常使用吐司

\*\*/

private static Toast toast = null;

public static void showToast(Context context, String text) {

Looper myLooper = Looper.myLooper();

if (myLooper == null) {

Looper.prepare();

myLooper = Looper.myLooper();

}

//设置toast的显示位置

WindowManager windowManager = (WindowManager) context.getSystemService(WINDOW\_SERVICE);

Point size = new Point();

windowManager.getDefaultDisplay().getSize(size);

toast = Toast.makeText(context, text, Toast.LENGTH\_LONG);

toast.setGravity(Gravity.BOTTOM, 0, size.y / 9);

toast.show();

if ( myLooper != null) {

Looper.loop();

myLooper.quit();

}

}

}

**配置文件: activity\_register.xml**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout

xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical" >

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:background="@color/colorPrimary"

android:gravity="center\_vertical">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerInParent="true"

android:text="注册"

android:textSize="30sp"/>

</RelativeLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="20px"

android:background="@mipmap/beijing1"

android:orientation="vertical"

android:gravity="top" >

<ImageView

android:id="@+id/imgIcon"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerHorizontal="true"

android:layout\_marginTop="40dp"

android:src="@mipmap/test"

/>

</RelativeLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:background="@drawable/shap"

android:orientation="vertical" >

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:gravity="center\_vertical"

android:orientation="horizontal" >

<EditText

android:id="@+id/et\_user\_name"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:paddingTop="2dp"

android:paddingBottom="2dp"

android:paddingLeft="6dp"

android:hint="用户名"

android:singleLine="true"

android:maxLength="15"

android:background="#ffffff"

android:textSize="18sp" />

</LinearLayout>

<LinearLayout

android:background="@drawable/shap"

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:gravity="center\_vertical"

android:orientation="horizontal" >

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:paddingTop="2dp"

android:paddingBottom="2dp"

android:paddingLeft="6dp"

android:singleLine="true"

android:hint="请输入您的6~14位密码"

android:inputType="textPassword"

android:maxLength="15"

android:background="#ffffff"

android:id="@+id/et\_psw"

android:textSize="18sp" />

</LinearLayout>

<LinearLayout

android:background="@drawable/shap"

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:gravity="center\_vertical"

android:orientation="horizontal" >

<EditText

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:paddingTop="2dp"

android:paddingBottom="2dp"

android:paddingLeft="6dp"

android:singleLine="true"

android:hint="请再次输入您的6~14位密码"

android:inputType="textPassword"

android:maxLength="15"

android:background="#ffffff"

android:id="@+id/et\_psw\_again"

android:textSize="18sp" />

</LinearLayout>

<LinearLayout

android:background="@drawable/shap"

android:layout\_width="match\_parent"

android:layout\_height="60dp"

android:gravity="center\_vertical"

android:orientation="horizontal" >

<TextView

android:layout\_width="0px"

android:layout\_height="wrap\_content"

android:layout\_weight="1"

android:layout\_marginLeft="10dp"

android:paddingLeft="6dp"

android:gravity="left"

android:text="性别"

android:textSize="18sp" />

<RadioGroup

android:layout\_width="0px"

android:layout\_height="wrap\_content"

android:layout\_weight="2.6"

android:id="@+id/SexRadio"

android:paddingLeft="5dp">

<RadioButton

android:id="@+id/mainRegisterRdBtnFemale"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="女生"/>

<RadioButton

android:id="@+id/mainRegisterRdBtnMale"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="男生"/>

</RadioGroup>

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="50dp"

android:gravity="center\_vertical"

android:orientation="horizontal" >

<EditText

android:id="@+id/et\_eamil"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

android:paddingTop="2dp"

android:paddingBottom="2dp"

android:paddingLeft="6dp"

android:hint="邮箱地址"

android:singleLine="true"

android:maxLength="15"

android:background="#ffffff"

android:textSize="18sp" />

</LinearLayout>

</LinearLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="10dp"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp" >

<Button

android:id="@+id/btn\_register"

android:onClick="register"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_centerInParent="true"

android:layout\_marginLeft="20dp"

android:layout\_marginRight="20dp"

android:layout\_marginTop="40dp"

android:text="注册"

android:textSize="24sp"

android:background="@color/colorPrimary"

android:visibility="invisible"/>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:layout\_centerHorizontal="true"

android:orientation="vertical"

android:layout\_marginTop="65dp"

>

<CheckBox

android:id="@+id/checkbox1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:scaleX="0.70"

android:scaleY="0.70"

android:textColor="#000000"

android:textSize="20dp"

android:text="我同意该协议条款" />

</LinearLayout>

</RelativeLayout>

</LinearLayout>

**（3）忘记密码界面**

**逻辑代码LostandFind.java**

/\*找回密码\*/

public class LostFindActivity extends BaseActivity {

private EditText edit1; // 输入用户名

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_lost\_find);

//设置屏幕为竖屏

setRequestedOrientation(ActivityInfo.SCREEN\_ORIENTATION\_PORTRAIT);

// 初始化界面

setupViews();

}

public void setupViews(){

//lost\_find.xml 页面中获取对应的UI控件

final TextView email = (TextView)findViewById(R.id.email);

TextView phone = (TextView)findViewById(R.id.phone);

final TextView edit= (TextView)findViewById(R.id.edit1) ;

email.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

final String editing = edit.getText().toString().trim();

//按钮

if(TextUtils.isEmpty(editing)){

showToast(getApplicationContext(),"请输入用户名");

}else {

new Thread() {

@Override

public void run() {

// TODO Auto-generated method stub

super.run();

Object result;

//调用编写的isEmail()函数

//获取到输入的内容并转换为字符串形式，这样才可以进行匹配，且邮箱格式不能太长，字符串长度31以内

//按照实际需要可以添加不同的事件

if (TextUtils.isEmpty((CharSequence) edit1)) {

showToast(getApplicationContext(),"请输入用户名");

}else {

try {

HttpURLConnection connection = (HttpURLConnection) new URL("http://10.168.14.3:8080/auth/regist").openConnection();

connection.setRequestMethod("POST");

connection.setDoInput(true);

connection.setDoOutput(true);

connection.setUseCaches(false);

connection.setRequestProperty("Content-type", "application/json;charset=UTF-8");

DataOutputStream outputStream = new DataOutputStream(connection.getOutputStream());

Map params = new HashMap();

/\* params.put("username", username);

params.put("password", psw);

params.put("sex", (sex == 0 ? "女" : "男"));

params.put("email", Email);\*/

outputStream.write(new JSONObject(params).toString().getBytes()); // POST

connection.connect(); // GET

if (connection.getResponseCode() == HttpURLConnection.HTTP\_OK) {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(connection.getInputStream()));

String inputLine = null;

while ((inputLine = bufferedReader.readLine()) != null) {

Log.i("result", inputLine);

Intent intent = new Intent(getApplicationContext(), LoginActivity.class);

showToast(getApplicationContext(),"找回成功");

startActivity(intent);

finish();

}

bufferedReader.close();

}

showToast(getApplicationContext(),"网络连接错误");

connection.disconnect();

} catch (IOException e) {

e.printStackTrace();

}

showToast(getApplicationContext(),"服务器连接错误");

}

}

}.start();

}

}

});

phone.setOnClickListener(new View.OnClickListener(){

public void onClick(View v){

final String editing = edit.getText().toString().trim();

//按钮

if(TextUtils.isEmpty(editing)){

showToast(getApplicationContext(),"请输入用户名");

}else {

new Thread() {

@Override

public void run() {

// TODO Auto-generated method stub

super.run();

Object result;

//调用编写的isEmail()函数

//获取到输入的内容并转换为字符串形式，这样才可以进行匹配，且邮箱格式不能太长，字符串长度31以内

//按照实际需要可以添加不同的事件

if (TextUtils.isEmpty((CharSequence) edit1)) {

showToast(getApplicationContext(),"请输入用户名");

}else {

try {

HttpURLConnection connection = (HttpURLConnection) new URL("http://10.168.14.3:8080/auth/regist").openConnection();

connection.setRequestMethod("POST");

connection.setDoInput(true);

connection.setDoOutput(true);

connection.setUseCaches(false);

connection.setRequestProperty("Content-type", "application/json;charset=UTF-8");

DataOutputStream outputStream = new DataOutputStream(connection.getOutputStream());

Map params = new HashMap();

/\* params.put("username", username);

params.put("password", psw);

params.put("sex", (sex == 0 ? "女" : "男"));

params.put("email", Email);\*/

outputStream.write(new JSONObject(params).toString().getBytes()); // POST

connection.connect(); // GET

if (connection.getResponseCode() == HttpURLConnection.HTTP\_OK) {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(connection.getInputStream()));

String inputLine = null;

while ((inputLine = bufferedReader.readLine()) != null) {

Log.i("result", inputLine);

Intent intent = new Intent(getApplicationContext(), LoginActivity.class);

showToast(getApplicationContext(),"找回成功");

startActivity(intent);

finish();

}

bufferedReader.close();

}

showToast(getApplicationContext(),"网络连接错误");

connection.disconnect();

} catch (IOException e) {

e.printStackTrace();

}

showToast(getApplicationContext(),"服务器连接错误");

}

}

}.start();

}

}

});

}

**配置文件: activity\_register.xml**

**4、数据展示模块:Weather\_Station.java**

**（1）数据展示 InitDatePage()：**

//---------------页面数据显示-----------------------//

//获取气象站数据服务

public class ServiceThread extends Thread {

@Override

public void run(){

// TODO Auto-generated method stub

super.run();

String request = "";

list\_data = HttpRequest.resultJson(request, "GET", "dateWS");

if (result != null) {

} else {

runOnUiThread(new Runnable() {

@Override

public void run() {

}

});

}

}

}

//首页获取控件的方法

//data\_one\_page数据页面获取控件的方法

public void InitDatePage(){

img\_new = (ImageView) data\_set\_one.findViewById(R.id.img\_with\_new);

imggood = (ImageView) data\_set\_one.findViewById(R.id.imgGood);

imgExpert = (ImageView) data\_set\_one.findViewById(R.id.imgExpert);

fish = (ImageView) data\_set\_one.findViewById(R.id.fish);

tvwsid = (TextView) data\_set\_one.findViewById(R.id.tvId);

tvairtemper = (TextView) data\_set\_one.findViewById(R.id.textair\_temper);

tvsoiltemper = (TextView) data\_set\_one.findViewById(R.id.textsoil\_temper);

tvairtempla = (TextView) data\_set\_one.findViewById(R.id.textair\_templa);

tvsoiltempla = (TextView) data\_set\_one.findViewById(R.id.textsoil\_templa);

tvoxygen = (TextView) data\_set\_one.findViewById(R.id.textoxygen);

tvwaterlevel = (TextView) data\_set\_one.findViewById(R.id.textwaterlevel);

tvlight = (TextView) data\_set\_one.findViewById(R.id.textlight);

tvwind\_speed = (TextView) data\_set\_one.findViewById(R.id.textwind\_speed);

tvwind\_direction = (TextView) data\_set\_one.findViewById(R.id.textwind\_direction);

tvpressure = (TextView) data\_set\_one.findViewById(R.id.textpressure);

final String tvairtempers = tvairtemper.getText().toString().trim();

final String tvsoiltempers = tvsoiltemper.getText().toString().trim();

final String tvairtemplas = tvairtempla.getText().toString().trim();

final String tvsoiltemplas = tvsoiltempla.getText().toString().trim();

final String tvoxygens = tvoxygen.getText().toString().trim();

final String tvwaterlevels = tvwaterlevel.getText().toString().trim();

final String tvlights = tvlight.getText().toString().trim();

final String tvwind\_speeds = tvwind\_speed.getText().toString().trim();

final String tvwind\_directions = tvwind\_direction.getText().toString().trim();

final String tvpressures = tvpressure.getText().toString().trim();

System.out.println("显示气象站数据中");

// getinformation();

if(list\_data!=null) {

System.out.println("123456");

if ("false".equals(list\_data)) {

Toast.makeText(getApplicationContext(), "获取实时信息失败，请确定气象站设备已连接。。。", 0).show();

} else {

try {

tvwaterlevel.setText(list\_data.get(0).get("waterLevel").toString());

tvlight.setText(list\_data.get(0).get("light").toString());

tvwind\_speed.setText(list\_data.get(0).get("wind\_sp").toString());

tvwind\_direction.setText(list\_data.get(0).get("wind\_dir").toString());

tvoxygen.setText(list\_data.get(0).get("oxygen").toString());

tvairtemper.setText(list\_data.get(0).get("air\_temper").toString());

tvairtempla.setText(list\_data.get(0).get("air\_templa").toString());

tvsoiltemper.setText(list\_data.get(0).get("soil\_temper").toString());

tvsoiltempla.setText(list\_data.get(0).get("soil\_templa").toString());

tvpressure.setText(list\_data.get(0).get("pressure").toString());

} catch (Exception e) {

}

}

};

}

}

**（2）数据刷新setDataListenerProcess()：**

//data\_one\_page数据页面控件的监听绑定

public void setDataListenerProcess() {

imggood.setOnClickListener(clickListener1);

imgExpert.setOnClickListener(clickListener1);

fish.setOnClickListener(clickListener1);

img\_new.setOnClickListener(clickListener1); //获取刷新数据按钮

if (list\_data == null) {

System.out.println("服务器数据为空");

}

handler = new Handler() {

@Override

public void handleMessage(Message msg) {

tvwaterlevel.setText(list\_data.get(0).get("waterLevel").toString());

tvlight.setText(list\_data.get(0).get("light").toString());

tvwind\_speed.setText(list\_data.get(0).get("wind\_sp").toString());

tvwind\_direction.setText(list\_data.get(0).get("wind\_dir").toString());

tvoxygen.setText(list\_data.get(0).get("oxygen").toString());

tvairtemper.setText(list\_data.get(0).get("air\_temper").toString());

tvairtempla.setText(list\_data.get(0).get("air\_templa").toString());

tvsoiltemper.setText(list\_data.get(0).get("soil\_temper").toString());

tvsoiltempla.setText(list\_data.get(0).get("soil\_templa").toString());

tvpressure.setText(list\_data.get(0).get("pressure").toString());

super.handleMessage(msg);

}

};

}

View.OnClickListener clickListener1 = new View.OnClickListener() { //设置点击刷新按钮监听

@Override

public void onClick(View v) {

if (v.getId() == R.id.img\_with\_new) {

new Thread() {

@Override

public void run() {

// TODO Auto-generated method stub

super.run();

Log.i("test", "run");

String request = "";

list\_data = HttpRequest.resultJson(request, "GET", "dateWS");

System.out.println(list\_data.get(0).get("oxygen").toString());

Message msg = Message.obtain();

msg.what = 0;

handler.sendMessage(msg);

if (result != null) {

} else {

runOnUiThread(new Runnable() {

@Override

public void run() {

Toast.makeText(getApplicationContext(), "请求成功", Toast.LENGTH\_SHORT).show();

}

});

}

}

}.start();

}if (v.getId() == R.id.imgGood) {

Intent intent = new Intent(getApplicationContext(), ExpertSystem.class);

startActivityForResult(intent, CODE);

} else if (v.getId() == R.id.imgExpert) {

Intent intent = new Intent(getApplicationContext(), ExpertSystem.class);

startActivityForResult(intent, CODE);

} else if(v.getId() == R.id.fish){ //进入渔林按钮

Intent intent = new Intent(getApplicationContext(), Fishing\_ForestActivity.class);

startActivityForResult(intent, CODE);

}

else {

}

}

};

1. **Activity\_weather\_station.xml**

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context="com.mywork.myapplication.weather\_station.MySetActivity"

android:orientation="vertical"

>

<!-- 底下的内容都房子V4的包中 -->

<android.support.v4.widget.DrawerLayout

android:id="@+id/draw"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

><!-- 主界面 -->

<RelativeLayout

android:id="@+id/relativelayout1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" >

<FrameLayout

android:id="@+id/fltTop"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" >

<RelativeLayout

android:id="@+id/linear1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignStart="@+id/fltTop"

android:layout\_alignTop="@+id/fltTop"

android:background="@mipmap/pg4">

<ImageView

android:id="@+id/imghead1"

android:layout\_width="26dp"

android:layout\_height="53dp"

android:src="@mipmap/menu1" />

</RelativeLayout>

</FrameLayout>

<LinearLayout

android:id="@+id/linear2"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_below="@+id/fltTop"

android:orientation="vertical" >

<android.support.v4.view.ViewPager

android:id="@+id/vPager"

android:layout\_width="wrap\_content"

android:layout\_height="605dp"

android:layout\_gravity="center"

android:layout\_weight="1.0"

android:background="#000000"

android:flipInterval="30"

android:persistentDrawingCache="animation" />

<!-- 主页底部线性布局菜单 -->

<LinearLayout

android:id="@+id/linearLayout1"

android:layout\_width="fill\_parent"

android:layout\_height="51dp"

android:layout\_alignParentBottom="true"

android:background="@color/white">

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:layout\_margin="0dp"

android:background="@color/gray"

>

<ImageView

android:id="@+id/mainpage"

android:layout\_width="match\_parent"

android:layout\_height="30dp"

android:src="@mipmap/mainpage1"

android:gravity="center"

/>

<TextView

android:id="@+id/mainpage1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:gravity="center"

android:text="首页"

android:textColor="@color/black"

/>

<!--<TextView

android:id="@+id/mainpage"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_margin="0dp"

android:layout\_weight="1.0"

android:background="@mipmap/mainpage"

android:button="@null"

android:drawablePadding="4dp"

android:gravity="center"

android:scaleType="fitCenter"

android:text=""

android:textColor="#000000"

android:textSize="12sp" />-->

</RelativeLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:layout\_margin="0dp"

android:background="@color/gray"

>

<ImageView

android:id="@+id/data"

android:layout\_width="match\_parent"

android:layout\_height="30dp"

android:src="@mipmap/data"

android:gravity="center"

/>

<TextView

android:id="@+id/data1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:gravity="center"

android:text="数据"

android:textColor="@color/black" />

<!--<TextView

android:id="@+id/mainpage"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_margin="0dp"

android:layout\_weight="1.0"

android:background="@mipmap/mainpage"

android:button="@null"

android:drawablePadding="4dp"

android:gravity="center"

android:scaleType="fitCenter"

android:text=""

android:textColor="#000000"

android:textSize="12sp" />-->

</RelativeLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:layout\_margin="0dp"

android:background="@color/gray"

>

<ImageView

android:id="@+id/set"

android:layout\_width="match\_parent"

android:layout\_height="30dp"

android:layout\_alignParentTop="true"

android:layout\_marginTop="0dp"

android:gravity="center"

android:src="@mipmap/set" />

<TextView

android:id="@+id/set1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:gravity="center"

android:text="设置"

android:textColor="@color/black"

/>

<!--<TextView

android:id="@+id/mainpage"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_margin="0dp"

android:layout\_weight="1.0"

android:background="@mipmap/mainpage"

android:button="@null"

android:drawablePadding="4dp"

android:gravity="center"

android:scaleType="fitCenter"

android:text=""

android:textColor="#000000"

android:textSize="12sp" />-->

</RelativeLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_weight="1"

android:layout\_margin="0dp"

android:background="@color/gray"

>

<ImageView

android:id="@+id/community"

android:layout\_width="match\_parent"

android:layout\_height="30dp"

android:gravity="center"

android:src="@mipmap/personal01" />

<TextView

android:id="@+id/community1"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_alignParentBottom="true"

android:gravity="center"

android:text="社区"

android:textColor="@color/black"

/>

<!--<TextView

android:id="@+id/mainpage"

android:layout\_width="0dp"

android:layout\_height="match\_parent"

android:layout\_margin="0dp"

android:layout\_weight="1.0"

android:background="@mipmap/mainpage"

android:button="@null"

android:drawablePadding="4dp"

android:gravity="center"

android:scaleType="fitCenter"

android:text=""

android:textColor="#000000"

android:textSize="12sp" />-->

</RelativeLayout>

<!--<TextView

android:id="@+id/data"

android:layout\_width="1dp"

android:layout\_height="match\_parent"

android:layout\_margin="5dp"

android:layout\_weight="1.0"

android:background="@mipmap/date1"

android:gravity="center"

android:scaleType="centerInside"

android:text=""

android:textColor="#000000"

android:textSize="22.0dip" />

<TextView

android:id="@+id/set"

android:layout\_width="1dp"

android:layout\_height="match\_parent"

android:layout\_margin="5dp"

android:layout\_weight="1.0"

android:background="@mipmap/setable"

android:gravity="center"

android:scaleType="centerInside"

android:text=""

android:textColor="#D11C1C"

android:textSize="22.0dip" />

<TextView

android:id="@+id/community"

android:layout\_width="1dp"

android:layout\_height="match\_parent"

android:layout\_margin="5dp"

android:layout\_weight="1.0"

android:background="@mipmap/personal1"

android:gravity="center"

android:scaleType="centerInside"

android:text=""

android:textColor="#000000"

android:textSize="22.0dip" />-->

</LinearLayout>

</LinearLayout>

</RelativeLayout>

<!-- 侧滑界面 -->

<LinearLayout

android:id="@+id/linearlayout1"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_gravity="left"

android:background="#FFFFFF"

android:orientation="vertical" >

<!-- android:background="@drawable/bg\_border" -->

<RelativeLayout

android:id="@+id/rlt\_Bg"

android:layout\_width="match\_parent"

android:layout\_height="200dp"

android:background="@mipmap/beijing1"

android:orientation="vertical" >

<!-- 点击图片头像进行获取头像 -->

<!-- <ImageView

android:id="@+id/imgheadce"

android:layout\_width="70dp"

android:layout\_height="80dp"

android:layout\_centerVertical="true"

android:layout\_marginTop="10dp"

android:padding="3dp"

android:src="@mipmap/tou01">

&lt;!&ndash; <ImageView

android:id="@+id/imghead2"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_centerInParent="true"

android:src="@mipmap/tou01"

/>&ndash;&gt;

</ImageView>-->

</RelativeLayout>

<!-- 第一个框 -->

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp" >

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="40dp"

android:orientation="horizontal" >

<ImageView

android:id="@+id/imagepeople"

android:layout\_width="45dp"

android:layout\_height="45dp"

android:layout\_gravity="center"

android:background="@mipmap/personaldate" />

<TextView

android:id="@+id/personalData"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/imagepeople"

android:text="个人资料"

android:textSize="30dp" />

</RelativeLayout>

</LinearLayout>

<View

android:layout\_width="wrap\_content"

android:layout\_height="0.5dp"

android:background="#000000" />

<!-- 第二个框 -->

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp" >

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="40dp" >

<ImageView

android:id="@+id/imgMyfish"

android:layout\_width="45dp"

android:layout\_height="45dp"

android:background="@mipmap/add" />

<TextView

android:id="@+id/tvWeather\_station"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/imgMyfish"

android:text="添加设备"

android:textSize="30dp" />

</RelativeLayout>

</LinearLayout>

<View

android:layout\_width="wrap\_content"

android:layout\_height="0.5dp"

android:background="#000000" />

<!-- 第三个框 -->

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp"

>

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" >

<ImageView

android:id="@+id/changeuser"

android:layout\_width="45dp"

android:layout\_height="45dp"

android:background="@mipmap/setpeople" />

<TextView

android:id="@+id/changeUser"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/changeuser"

android:text="修改用户"

android:textSize="30dp" />

</RelativeLayout>

</LinearLayout>

<View

android:layout\_width="wrap\_content"

android:layout\_height="0.5dp"

android:background="#000000" />

<!-- &lt;!&ndash; 第四个框 &ndash;&gt;

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp" >

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" >

<ImageView

android:id="@+id/mysetup"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:background="@mipmap/setup" />

<TextView

android:id="@+id/tvSet"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/mysetup"

android:text="我的设置"

android:textSize="30dp" />

</RelativeLayout>

</LinearLayout>

<View

android:layout\_width="wrap\_content"

android:layout\_height="0.5dp"

android:background="#000000" />-->

<!-- 第5个框 -->

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="20dp" >

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" >

<ImageView

android:id="@+id/exit"

android:layout\_width="45dp"

android:layout\_height="45dp"

android:background="@mipmap/exit" />

<TextView

android:id="@+id/exitLogin"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/exit"

android:text="退出登录"

android:textSize="30dp" />

</RelativeLayout>

</LinearLayout>

<View

android:layout\_width="wrap\_content"

android:layout\_height="0.5dp"

android:background="#000000" />

</LinearLayout>

</android.support.v4.widget.DrawerLayout>

</LinearLayout>

**5、个人信息展示模块:PersonalData.java**

**（1）个人资料展示：init.class**

//个人信息界面初始化

@SuppressLint("WrongViewCast")

public void init() {

imageView = (ImageView) findViewById(R.id.imgLeft); //获取返回

btnsave =(Button) findViewById(R.id.btnSave); //获取保存用户信息

Sex =(EditText) findViewById(R.id.etSex); //获取用户性别

Username =(EditText) findViewById(R.id.etUsername); //获取用户名

Name =(EditText) findViewById(R.id.name); //获取姓名

/\*userpassword =(EditText) findViewById(R.id.etUserpass); //获取密码\*/

E\_mail=(EditText) findViewById(R.id.E\_mail); //获取e\_mail

Identify = (TextView) findViewById(R.id.identifys); //获取地址\*/

userid = (TextView) findViewById(R.id.Userid); //获取用户id

List<HashMap<String,Object>> allData = new ArrayList<HashMap<String,Object>>();

try {

allData = Analysis.AnalysisUserInfo(person);

} catch (JSONException e) {

e.printStackTrace();

}

Username.setText(allData.get(0).get("username").toString());

Sex.setText(allData.get(0).get("sex").toString());

E\_mail.setText(allData.get(0).get("email").toString());

Identify.setText(allData.get(0).get("identity").toString());

Name.setText(allData.get(0).get("name").toString());

userid.setText(allData.get(0).get("id").toString());

// /\*imgTitle.setImageURI(Uri.parse(allData.get(0).get("face").toString()));\*/

// imgTitle.setText(allData.get(0).get("face").toString());//获取图片字符串

handler = new Handler() {

@Override

public void handleMessage(Message msg) {

if(HttpRequest.bitmap != null){

Face = (ImageView) findViewById(R.id.imgTitle);

Face.setImageBitmap(HttpRequest.bitmap);

}

super.handleMessage(msg);

}

};

final List<HashMap<String, Object>> finalAllData = allData;

new Thread(){

public void run(){

String url =finalAllData.get(0).get("face").toString();

Log.i("we",url);

HttpRequest.getImage(url,"GET");

Message msg = Message.obtain();

msg.what = 0;

handler.sendMessage(msg);

}

}.start();

}

//点击性别选择

public void etSex(View sex){

AlertDialog.Builder builderSex= new AlertDialog.Builder(this); //自定义对话框

builderSex.setTitle("请选择性别");

builderSex.setSingleChoiceItems(sexArry, 0, new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) { //whitch是被选中的性别位置

Sex.setText(sexArry[which]);

Sex.setSelection(Sex.getText().length());

dialog.dismiss(); //对边点击一个item显示对话框，不用点击确认消息

}

});

builderSex.show(); //显示性别选择弹框

}

//验证是否是电话号的

public static boolean isMobile(String str) {

Pattern p = null;

Matcher m = null;

boolean b = false;

p = Pattern.compile("^[1][3,5,8][0-9]{9}$"); // 验证手机号

m = p.matcher(str);

b = m.matches();

return b;

}

//验证是否是邮箱验证

public static boolean isEmail(String strEmail) {

String strPattern = "^[a-zA-Z0-9][\\w\\.-]\*[a-zA-Z0-9]@[a-zA-Z0-9][\\w\\.-]\*[a-zA-Z0-9]\\.[a-zA-Z][a-zA-Z\\.]\*[a-zA-Z]$";

if(TextUtils.isEmpty(strPattern)) {

return false;

}else {

return strEmail.matches(strPattern);

}

}

**（2）个人资料管理：save.class**

//点击保存个人信息

public void save() {

imageView = (ImageView) findViewById(R.id.imgLeft); //获取返回

imageView1 = (ImageView) findViewById(R.id.imgTitle); //获取返回

btnsave =(Button) findViewById(R.id.btnSave); //获取保存用户信息

Sex =(EditText) findViewById(R.id.etSex); //获取用户性别

Username =(EditText) findViewById(R.id.etUsername); //获取用户名

Name =(EditText) findViewById(R.id.name); //获取姓名

/\*userpassword =(EditText) findViewById(R.id.etUserpass); //获取密码\*/

E\_mail=(EditText) findViewById(R.id.E\_mail); //获取e\_mail

Identify = (TextView) findViewById(R.id.identifys); //获取地址\*/

userid = (TextView) findViewById(R.id.Userid); //获取用户id

btnsave.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

//获取信息框内输入的内容

final String id = userid.getText().toString().trim();

final String sex = Sex.getText().toString().trim();

final String username = Username.getText().toString().trim();

final String name = Name.getText().toString().trim();

/\*final String password = userpassword.getText().toString().trim();\*/

final String e\_mail = E\_mail.getText().toString().trim();

final String identity = Identify.getText().toString().trim();

/\*final String address =Address.getText().toString().trim();\*/

if("".equals(name)||"".equals(e\_mail)){

Toast.makeText(getApplicationContext(), "真实姓名、邮箱不能为空，请重新输入！", Toast.LENGTH\_SHORT).show();

}else{

new Thread() {

@Override

public void run() {

// TODO Auto-generated method stub

super.run();

try {

HttpURLConnection connection = (HttpURLConnection) new URL("http://10.168.14.55:8080/auth/regist").openConnection();

connection.setRequestMethod("POST");

connection.setDoInput(true);

connection.setDoOutput(true);

connection.setUseCaches(false);

connection.setRequestProperty("Content-type", "application/json;charset=UTF-8");

DataOutputStream outputStream = new DataOutputStream(connection.getOutputStream());

Map params = new HashMap();

params.put("username", username);

params.put("name", name);

params.put("identity", identity);

params.put("id", id);

params.put("sex", sex);

params.put("email", e\_mail);

params.put("email", e\_mail);

outputStream.write(new JSONObject(params).toString().getBytes()); // POST

connection.connect(); // GET

if (connection.getResponseCode() == HttpURLConnection.HTTP\_OK) {

BufferedReader bufferedReader = new BufferedReader(new InputStreamReader(connection.getInputStream()));

String inputLine = null;

while ((inputLine = bufferedReader.readLine()) != null) {

Intent intent = new Intent(PersonalDate.this, LoginActivity.class);

showToast(PersonalDate.this, "请求成功");

startActivity(intent);

finish();

}

bufferedReader.close();

}else {

}

connection.disconnect();

} catch (IOException e) {

e.printStackTrace();

showToast(PersonalDate.this, "网络连接错误");

}

}

}.start();

}

}

});

}

1. **专家系统建议模块**

**（1）ExperSystem.java**

protected void onCreate(Bundle savedInstanceState) {

// TODO Auto-generated method stub

super.onCreate(savedInstanceState);

setContentView(R.layout.expert\_system);

spinner=(android.widget.Spinner) findViewById(R.id.spinner1); //获取下列列表控件

btnExpert=(Button) findViewById(R.id.btnExpertAdvice); //获取专家意见按钮控件

showText=(TextView)findViewById(R.id.TextShow); //获取显示列表选择的内容

scrollView=(ScrollView) findViewById(R.id.scrollView1); //添加滑动布局

scrollView.setVisibility(View.GONE); //设置滑动布局不可见

spinner.getSelectedItem(); //点击下拉列表方法

//给下拉列表添加监听，监听选择内容 ，并且将选择的内容显示到指定文本上

spinner.setOnItemSelectedListener(new OnItemSelectedListener() {

public void onItemSelected(AdapterView<?>parent,View arg1,int pos,long id) {

String result=parent.getItemAtPosition(pos).toString(); //获取选择项的值

showText.setText(result);

}

public void onNothingSelected(AdapterView<?>arg0) {

}

});

btnExpert.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

AlertDialog.Builder builder = new AlertDialog.Builder(ExpertSystem.this);

builder.setTitle("专家建议");

//引入专家建议布局

View view = LayoutInflater.from(ExpertSystem.this).inflate(R.layout.expert\_advice, null);

builder.setView(view); //设置我自定义的布局文件为弹出窗口

builder.show(); //显示弹窗

builder.setCancelable(true); //点击弹窗外，出发事件让弹框消失

//点击我的专家意见按钮后判断滑动布局是否显示

if(isVisible) {

isVisible=false;

scrollView.setVisibility(View.VISIBLE); //显示滑动布局内容

}else {

scrollView.setVisibility(View.GONE); //隐藏滑动布局内容

isVisible=true;

}

}

});

//使用字符串数组创建适配器

ArrayAdapter<CharSequence> adapter =ArrayAdapter.createFromResource(

this, R.array.ftype,android.R.layout.simple\_dropdown\_item\_1line);

//为适配器设置列表框下拉时的选项样式

adapter.setDropDownViewResource(android.R.layout.simple\_spinner\_dropdown\_item);

//给界面返回按钮添加监听

imageView=(ImageView) findViewById(R.id.imgLeft);

imageView.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

Intent intent=new Intent();

setResult(0x22,intent);

finish();

}

});

}

**（2）expert\_system.java**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical" >

<FrameLayout

android:layout\_width="fill\_parent"

android:layout\_height="50dp"

android:background="#99CCFF"

>

<ImageView

android:id="@+id/imgLeft"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="center\_vertical"

android:src="@mipmap/toleft"

>

</ImageView>

</FrameLayout>

<RelativeLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content" >

<ImageView

android:id="@+id/imgicon"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_alignParentLeft="true"

android:layout\_marginLeft="100dp"

android:src="@mipmap/zhuanjia" />

<Spinner

android:id="@+id/spinner1"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginTop="2dp"

android:layout\_toRightOf="@+id/imgicon"

android:entries="@array/ftype" />

<TextView

android:id="@+id/TextShow"

android:layout\_width="300px"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/spinner1"

android:layout\_centerInParent="true"

android:layout\_marginTop="50px"

android:gravity="center" />

<Button

android:id="@+id/btnExpertAdvice"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_below="@+id/TextShow"

android:layout\_centerInParent="true"

android:layout\_marginTop="50px"

android:text="安全数据区间设置" />

</RelativeLayout>

<ScrollView

android:id="@+id/scrollView1"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

>

<LinearLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:orientation="vertical">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="horizontal"

>

<ImageView

android:layout\_width="173dp"

android:layout\_height="wrap\_content"

android:src="@mipmap/qixiangzhan1" />

<ImageView

android:layout\_width="wrap\_content"

android:layout\_height="293dp"

android:src="@mipmap/timg" />

</LinearLayout>

</ScrollView>

</LinearLayout>

**7、智能远程控制**

**逻辑代码：set\_two.java**

//获取界面的控件id

public void init() {

imgBack=(ImageView) findViewById(R.id.imgLeft); //获取返回按钮图片控件

temperhigh= (EditText) findViewById(R.id.ethigh1);

temperlow = (EditText) findViewById(R.id.etlow1);

O2high = (EditText) findViewById(R.id.ethigh2);

O2low = (EditText) findViewById(R.id.etlow2);

lighthigh = (EditText) findViewById(R.id.ethigh4);

lightlow = (EditText) findViewById(R.id.etlow4);

setWeather\_StationConfig=(Button)findViewById(R.id.setFishConfig);

temperlow.setText(minTemper);

temperhigh.setText(maxTemper);

O2low.setText(minO2);

O2high.setText(maxO2);

lightlow.setText(minLight);

lighthigh.setText(maxLight);

}

//给界面控件添加监听

public void textViewOnClickListener() {

//给返回按钮添加监听

imgBack.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View v) {

// TODO Auto-generated method stub

Intent intent=new Intent();

setResult(0x22,intent);

finish();

}

});

}

//用户配置设置保存

public void setConfig(View v) {

minTemper =temperlow.getText().toString().trim();

maxTemper = temperhigh.getText().toString().trim();

minO2 = O2low.getText().toString().trim();

maxO2 = O2high.getText().toString().trim();

minLight = lightlow.getText().toString().trim();

maxLight = lighthigh.getText().toString().trim();

**配置文件：setup\_two.xml**

<LinearLayout

android:id="@+id/setPager"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="vertical"

android:layout\_alignParentBottom="true"

>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_marginTop="30dp"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:gravity="center\_horizontal"

android:text="智能远程控制"

android:textSize="26dp" />

</LinearLayout>

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:layout\_marginTop="80dp"

android:layout\_marginLeft="10dp"

android:layout\_marginRight="10dp"

>

<RelativeLayout

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

>

<Switch

android:id="@+id/switch1\_changeWater"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="网络连接"

android:textSize="20dp"

/>

<Switch

android:id="@+id/switch2\_addWater"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_toRightOf="@+id/switch1\_changeWater"

android:text="WIFI设置"

android:textSize="20dp"

/>

</RelativeLayout>

</LinearLayout>

**8、服务器连接**

**HttpRequest.java**

public static String send(String u, String request,String type){

ByteArrayOutputStream outStream = new ByteArrayOutputStream();

byte[] data = new byte[1024];

int len = 0;

try {

URL url = new URL(u);

HttpURLConnection conn = (HttpURLConnection) url.openConnection();

//请求类型

conn.setRequestMethod(request);

InputStream inStream = conn.getInputStream();

String cookieval = conn.getHeaderField("Set-Cookie");

if (cookieval != null && type == "login") {

session\_id = null;

session\_id = cookieval.substring(0, cookieval.indexOf(";"));//获取sessionid

Log.i("SESSION", "session\_id=" + session\_id);

}

while ((len = inStream.read(data)) != -1) {

outStream.write(data, 0, len);

}

inStream.close();

} catch (IOException e) {

e.printStackTrace();

}

System.out.println(new String(outStream.toByteArray()));

//通过out.Stream.toByteArray获取到写的数据

return new String(outStream.toByteArray());

}

/\*\*

\* fileName 文件名(不带后缀)

\* filePath 文件的本地路径 (xxx / xx / test.jpg)

\*/

public static void uploadFile(String fileName, String filePath) {

HttpURLConnection conn = null;

/// boundary就是request头和上传文件内容的分隔符(可自定义任意一组字符串)

String BOUNDARY = "\*\*\*\*\*\*";

// 用来标识payLoad+文件流的起始位置和终止位置(相当于一个协议,告诉你从哪开始,从哪结束)

String preFix = ("\r\n--" + BOUNDARY + "--\r\n");

String target="http://10.168.14.55:8080/upload/image";

URL url;

try {

// (HttpConst.uploadImage 上传到服务器的地址

url = new URL(target);

conn = (HttpURLConnection) url.openConnection();

conn.setConnectTimeout(5000);

conn.setReadTimeout(30000);

conn.setDoOutput(true);

conn.setDoInput(true);

conn.setUseCaches(false);

// 设置请求方法

conn.setRequestMethod("POST");

// 设置header

conn.setRequestProperty("Accept","\*/\*");

conn.setRequestProperty("Connection", "keep-alive");

conn.setRequestProperty("Content-Type",

"multipart/form-data; boundary=" + BOUNDARY);

// 获取写输入流

OutputStream out = new DataOutputStream(conn.getOutputStream());

// 获取上传文件

File file = new File(filePath);

// 要上传的数据

StringBuffer strBuf = new StringBuffer();

// 标识payLoad + 文件流的起始位置

strBuf.append(preFix);

// 这里只要把.jpg改成.txt，把Content-Type改成上传文本的类型，就能上传txt文件了。

strBuf.append("Content-Disposition: form-data; name=\"file\"; filename=\"" + fileName+".jpg" + "\"\r\n");

strBuf.append("Content-Type: image/jpeg" + "\r\n\r\n");

out.write(strBuf.toString().getBytes());

// 获取文件流

FileInputStream fileInputStream = new FileInputStream(file);

DataInputStream inputStream = new DataInputStream(fileInputStream);

// 每次上传文件的大小(文件会被拆成几份上传)

int bytes = 0;

// 计算上传进度

float count = 0;

// 获取文件总大小

int fileSize = fileInputStream.available();

// 每次上传的大小

byte[] bufferOut = new byte[1024];

// 上传文件

while ((bytes = inputStream.read(bufferOut)) != -1) {

// 上传文件(一份)

out.write(bufferOut, 0, bytes);

// 计算当前已上传的大小

count += bytes;

// 打印上传文件进度(已上传除以总大\*100就是进度)

Log.i("progress：",(count / fileSize \* 100) +"%");

}

// 关闭文件流

inputStream.close();

// 标识payLoad + 文件流的结尾位置

out.write(preFix.getBytes());

// 输出所有数据到服务器

out.flush();

// 关闭网络输出流

out.close();

// 重新构造一个StringBuffer,用来存放从服务器获取到的数据

strBuf = new StringBuffer();

// 打开输入流 , 读取服务器返回的数据

BufferedReader reader = new BufferedReader(new

InputStreamReader(conn.getInputStream()));

String line;

// 一行一行的读取服务器返回的数据

while ((line = reader.readLine()) != null) {

strBuf.append(line).append("\n");

}

// 关闭输入流

reader.close();

// 打印服务器返回的数据

Log.i("上传成功",strBuf.toString());

} catch (Exception e) {

Log.i("上传图片出错",e.toString());

} finally {

if (conn != null) {

conn.disconnect();

}

}

}

/\*\*

\* @Description 解析数据

\* @Param [http从服务器获取的数据]

\* @return java.util.ArrayList<java.util.HashMap<java.lang.String,java.lang.Object>>

\*\*/

private static ArrayList<HashMap<String,Object>> Analysis(String jsonStr) throws JSONException{

JSONArray jsonArray = null;

//初始化list对象

ArrayList<HashMap<String,Object>> list = new ArrayList<HashMap<String,Object>>();

JSONObject jsonObject = new JSONObject(jsonStr);

JSONObject data = jsonObject.getJSONObject("data");

System.out.println(jsonObject.getBoolean("success"));

System.out.println(data.getString("username"));

return list;

}

/\*\*

\* @Description //TOOD 适配数据，返回result

\* @return java.lang.String

\*\*/

public static List<HashMap<String,Object>> resultJson(String s, String r, String type){

try {

String url = "http://10.168.14.3:8080";

String path = null;

if(type.equals("login")){

path = "/fmall/user/login.do";

url = url + path;

}else if(type.equals("register")){ //注册事件

path = "/auth/regist";

url = url + path;

}else if(type.equals("dateWS")){ //气象站数据

path = "http://10.168.14.3:8080/app/data";

url = path;

}else if(type.equals("fish")){

path = "http://10.168.14.3:8080/app/data";

url = path;

}else if(type.equals("image")){

url = "/auth/regist";

}else if(type.equals("dateperson")){ //个人资料按钮

path="http://10.168.14.3:8080/auth/applogin";

url = path;

}else if(type.equals("getAllProduct")){

url = url + "/fmall/product/show\_product.do";

}else if(type.equals("createOrder")){

url = url + "/fmall/order/create.do";

}else if(type.equals("showCart")){

url = url + "/fmall/cart/show\_cart.do";

}else if(type.equals("addCollection")){

url = url + "/fmall/product/add\_collection.do";

}else if(type.equals("addCart")){

url = url + "/fmall/cart/add\_cart.do";

}

url = url + s;

String request = r;

Analysis analysis = new Analysis();

if(type.equals("login")){

allData = analysis.AnalysisLogin(send(url,request,type));

}else if(type.equals("register")){

String result = send(url,request,type);

allData = analysis.AnalysisRegister(result);

Log.i("allData", result);

}else if(type.equals("dateWS")){

allData = analysis.AnalysisDateInfo(send(url,request,type));

}else if(type.equals("fish")){

allData = analysis.AnalysisfishDateInfo(send(url,request,type));

}else if(type.equals("image")){

getImage(url,request);

}else if(type.equals("dateperson")){

allData = analysis.AnalysisGetApPerson(send(url,request,type));

}else if(type.equals("getAllProduct")){

allData = analysis.AnalysisGetAllProduct(send(url,request,type));

}else if(type.equals("createOrder")){

allData = analysis.AnalysisCreateOrder(send(url,request,type));

}else if(type.equals("showCart")){

allData = analysis.AnalysisShowCart(send(url,request,type));

}else if(type.equals("addCollection")){

allData = analysis.AnalysisWatch(send(url,request,type));

}else if(type.equals("addCart")){

allData = analysis.AnalysisWatch(send(url,request,type));

if(allData == null){

System.out.println("111111111111111111111111111111111111111111111111111");

}

}

} catch (JSONException e) {

e.printStackTrace();

}finally {

return allData;

}

}

**Analysis.java**

public static ArrayList<HashMap<String, Object>> AnalysisDateInfo(String jsonStr) throws JSONException {

//初始化list对象

ArrayList<HashMap<String, Object>> list = new ArrayList<HashMap<String, Object>>();

JSONArray jsonArray = new JSONArray(jsonStr);

JSONObject jsonObject = jsonArray.getJSONObject(0);

JSONObject jsonObject1 = jsonObject.getJSONObject("met\_atm");

JSONObject jsonObject2 = jsonObject.getJSONObject("met\_envir");

JSONObject jsonObject3 = jsonObject.getJSONObject("met\_soil");

JSONObject json = jsonArray.getJSONObject(1);

JSONObject jsonObject4 = json.getJSONObject("for\_soil");

JSONObject jsonObject5 = json.getJSONObject("fish");

JSONObject jsonObject6 = json.getJSONObject("for\_atm");

HashMap<String, Object> map = new HashMap<String, Object>();

//气象站

map.put("air\_temper", jsonObject1.getDouble("atm\_tem"));

map.put("air\_templa", jsonObject1.getDouble("atm\_humidity"));

map.put("oxygen", jsonObject1.getDouble("o2"));

map.put("pressure", jsonObject1.getDouble("atm\_pre"));

map.put("waterLevel",jsonObject2.getDouble("rain"));

map.put("wind\_sp",jsonObject2.getDouble("wind\_sp"));

map.put("wind\_dir",jsonObject2.getDouble("wind\_dir"));

map.put("light",jsonObject2.getDouble("illum"));

map.put("soil\_temper",jsonObject3.getDouble("soil\_tem"));

map.put("soil\_templa",jsonObject3.getDouble("soil\_humidity"));

// 渔林

map.put("soil\_nity",jsonObject4.getDouble("soil\_nity"));

map.put("soil\_tem",jsonObject4.getDouble("soil\_tem"));

map.put("soil\_hum",jsonObject4.getDouble("soil\_hum"));

map.put("conductivity",jsonObject4.getDouble("conductivity"));

map.put("water\_tem",jsonObject5.getDouble("water\_tem"));

map.put("turbidity",jsonObject5.getDouble("turbidity"));

map.put("co2",jsonObject6.getDouble("co2"));

map.put("atm\_tem",jsonObject6.getDouble("atm\_tem"));

map.put("smoke",jsonObject6.getDouble("smoke"));

map.put("atm\_humidity",jsonObject6.getDouble("atm\_humidity"));

list.add(map);

/\*JSONObject jsonObject1 = jsonObject.getJSONObject("met\_envir");

JSONObject jsonObject2 = jsonObject.getJSONObject("met\_oil");

JSONObject jsonObject3 = jsonObject.getJSONObject("met\_atm");

HashMap<String,Object> map = new HashMap<String,Object>();

map.put("waterLevel",jsonObject1.getDouble("rain"));

map.put("wind\_sp",jsonObject1.getDouble("wind\_sp"));

map.put("wind\_dir",jsonObject1.getDouble("wind\_dir"));

map.put("light",jsonObject1.getDouble("illum"));

map.put("soil\_temper",jsonObject2.getDouble("soil\_tem"));

map.put("soil\_templa",jsonObject2.getDouble("soil\_humidity"));

map.put("air\_temper",jsonObject3.getDouble("atm\_tem"));

map.put("air\_templa",jsonObject3.getDouble("atm\_humidity"));

map.put("oxygen",jsonObject3.getDouble("o2"));

map.put("pressure",jsonObject3.getDouble("atm\_pre"));

list.add(map);\*/

return list;

}

public static ArrayList<HashMap<String, Object>> AnalysisfishDateInfo(String jsonStr) throws JSONException {

//初始化list对象

ArrayList<HashMap<String, Object>> list = new ArrayList<HashMap<String, Object>>();

JSONObject jsonObject = new JSONObject(jsonStr);

JSONObject jsonObject1 = jsonObject.getJSONObject("envir");

JSONObject jsonObject2 = jsonObject.getJSONObject("soil");

JSONObject jsonObject3 = jsonObject.getJSONObject("atm");

HashMap<String, Object> map = new HashMap<String, Object>();

map.put("waterLevel", jsonObject1.getDouble("rain"));

map.put("wind\_sp", jsonObject1.getDouble("wind\_sp"));

map.put("wind\_dir", jsonObject1.getDouble("wind\_dir"));

map.put("light", jsonObject1.getDouble("illum"));

map.put("soil\_temper", jsonObject2.getDouble("soil\_tem"));

map.put("soil\_templa", jsonObject2.getDouble("soil\_humidity"));

map.put("air\_temper", jsonObject3.getDouble("atm\_tem"));

map.put("air\_templa", jsonObject3.getDouble("atm\_humidity"));

map.put("oxygen", jsonObject3.getDouble("o2"));

map.put("pressure", jsonObject3.getDouble("atm\_pre"));

list.add(map);

return list;

}

public static ArrayList<HashMap<String, Object>> AnalysisGetApPerson(String jsonStr) throws JSONException {

//初始化list对象

ArrayList<HashMap<String, Object>> list = new ArrayList<HashMap<String, Object>>();

JSONObject jsonObject = new JSONObject(jsonStr);

JSONObject jsonObject1 = jsonObject.getJSONObject("user");

/\* JSONArray jsonArray= null;\*/

HashMap<String, Object> map = new HashMap<String, Object>();

map.put("username", jsonObject1.getString("username"));

map.put("name", jsonObject1.getString("name"));

/\*jsonArray = jsonObject.getJSONArray("data");

for(int i = 0; i < jsonArray.length(); i++){

JSONObject jsonObject1 = jsonArray.getJSONObject(i);

HashMap<String,Object> map = new HashMap<String,Object>();

map.put("id",jsonObject1.getInt("id"));

map.put("name",jsonObject1.getString("name"));

list.add(map);

}\*/

return list;

}

public static ArrayList<HashMap<String, Object>> AnalysisGetAllProduct(String jsonStr) throws JSONException {

//初始化list对象

ArrayList<HashMap<String, Object>> list = new ArrayList<HashMap<String, Object>>();

JSONObject jsonObject = new JSONObject(jsonStr);

JSONObject jsonObject1 = jsonObject.getJSONObject("data");

JSONArray jsonArray = null;

jsonArray = jsonObject1.getJSONArray("list");

for (int i = 0; i < jsonArray.length(); i++) {

JSONObject jsonObject2 = jsonArray.getJSONObject(i);

HashMap<String, Object> map = new HashMap<String, Object>();

map.put("id", jsonObject2.getInt("id"));

map.put("userId", jsonObject2.getInt("userId"));

map.put("categoryId", jsonObject2.getInt("categoryId"));

map.put("livestock", jsonObject2.getInt("livestock"));

map.put("name", jsonObject2.getString("name"));

map.put("subtitle", jsonObject2.getString("subtitle"));

map.put("mainImage", jsonObject2.getString("mainImage"));

map.put("detail", jsonObject2.getString("detail"));

map.put("stock", jsonObject2.getInt("stock"));

map.put("status", jsonObject2.getInt("status"));

map.put("price", jsonObject2.getDouble("price"));

// map.put("price",jsonObject2.getBig)

list.add(map);

}

return list;

}

public static ArrayList<HashMap<String, Object>> AnalysisCreateOrder(String jsonStr) throws JSONException {

//初始化list对象

ArrayList<HashMap<String, Object>> list = new ArrayList<HashMap<String, Object>>();

JSONObject jsonObject = new JSONObject(jsonStr);

// JSONObject data = jsonObject.getJSONObject("data");

HashMap<String, Object> map = new HashMap<String, Object>();

map.put("status", jsonObject.getInt("status"));

map.put("data", jsonObject.getString("data"));

list.add(map);

return list;

}

public static ArrayList<HashMap<String, Object>> AnalysisWatch(String jsonStr) throws JSONException {

//初始化list对象

ArrayList<HashMap<String, Object>> list = new ArrayList<HashMap<String, Object>>();

JSONObject jsonObject = new JSONObject(jsonStr);

// JSONObject data = jsonObject.getJSONObject("data");

HashMap<String, Object> map = new HashMap<String, Object>();

map.put("status", jsonObject.getInt("status"));

map.put("msg", jsonObject.getString("msg"));

list.add(map);

return list;

}

**9、Utils**

**Streamtools.java**

import java.io.ByteArrayOutputStream;

import java.io.InputStream;

public class StreamTools {

public static String readInputStream(InputStream is ) {

try {

ByteArrayOutputStream baos = new ByteArrayOutputStream();

int len = 0;

byte[] buffer = new byte[1024];

while ((len=is.read(buffer))!=-1) {

baos.write(buffer,0,len);

}

is.close();

baos.close();

byte[] result = baos.toByteArray();

//试着解析result里面的内容

String temp = new String(result);

return temp;

} catch (Exception e) {

// TODO Auto-generated catch block

e.printStackTrace();

return "获取失败";

}

}

}

**BaseActivity**

import android.app.Activity;

import android.os.Bundle;

import android.widget.CompoundButton;

import android.widget.Toast;

public class BaseActivity extends Activity {

private myApplication application;

private BaseActivity oContext;

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

if(application == null) {

//得到Application对象

application=(myApplication) getApplication();

}

oContext =this; //把当前的上下文对象复制给BaseActivity

addActivity(); //调用添加Activity的方法

}

//添加Activity方法

public void addActivity() {

application.addActivity\_(oContext); //调用myApplication的添加Activity方法

}

//销毁当个Activity的方法

public void removeActivity() {

application.removeActivity\_(oContext);

}

//销毁所有的Activity

public void removeALLActivity() {

application.removeALLActivity\_();

}

//定义Toast方法，提示所执行的内容，可以重复使用

public void show\_Toast(String text) {

Toast.makeText(oContext, text, Toast.LENGTH\_SHORT).show();

}

public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {

// TODO Auto-generated method stub

}

protected void onActivityCreated(Bundle savedInstanceState) {

}

}