

Android开发与应用

项目开发报告

|  |  |
| --- | --- |
| 项目名称： | 任务管理器 |
| 院 （系）： | 计算机学院 |
| 专 业： | 计算机科学与技术 |
| 班 级： | 计科1407班 |
| 姓 名： | 张永健 |
| 学 号： | 04141086 |
| 指导教师： | 孟伟君 |
| 起止时间： | 2017年6月02日至2017年6月16日 |

**一. 实验目的及实验环境**

1. 实验目的

学会使用所学知识开发一个完整的APP项目。

2.实验环境

系统开发平 Android Studio

系统开发平台：Android

运行平台：Windows XP及以上

运行环境：Microsoft.NET Framework SDK 4.0

**二. 实验教材、组织方式、实验内容**

1.实验教材： Andorid开发与应用

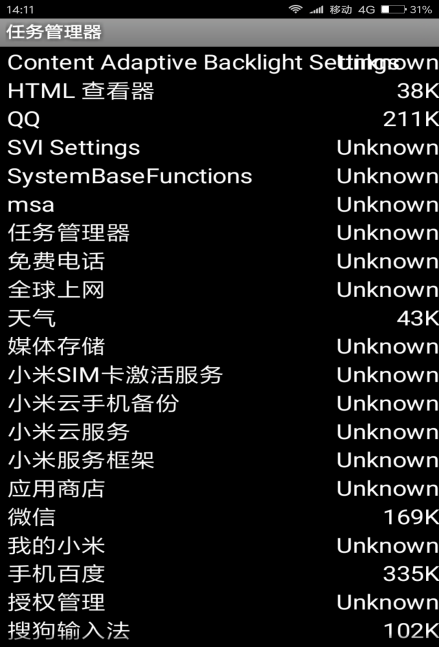
2.组织方式：个人独立完成

2.实验内容：

运用学过的知识完成一个任务管理器，可以对手机上的应用进行卸载，停止，查看，详细信息显示功能。

1. **运行结果**

1.初始界面

** **

2.切换到设置界面

** **

3.卸载界面

****

**四．总结**

这次实实验是对所学知识的全面的一个总结，让我对用Android设计完整APP的过程有了一个整体的认识，也巩固了之前所学的一些知识。在整个项目的设计过程中也遇到过很多的问题，但最终通过各种途径都一一解决了，增强了我分析问题解决问题的能力，并且对Android studio 的一些常用功能的操作更加熟悉，极大的额提高了我对APP开发的兴趣。

**五．附录：**源代码

**dataiProcess.java**

package com.xmobileapp.taskmanager;

import android.app.AlertDialog;

import android.app.Dialog;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.pm.PackageInfo;

import android.content.pm.PackageManager;

import android.content.pm.PackageManager.NameNotFoundException;

import android.net.Uri;

import android.widget.Toast;

public class MiscUtil {

public static final int MENU\_CANCEL = 0;

public static final int MENU\_SWITCH = 1;

public static final int MENU\_KILL = 2;

public static final int MENU\_DETAIL = 3;

public static final int MENU\_UNINSTALL = 4;

public static PackageInfo getPackageInfo(PackageManager pm, String name) {

PackageInfo ret = null;

try {

ret = pm.getPackageInfo(name, PackageManager.GET\_ACTIVITIES);

} catch (NameNotFoundException e) {

//TODO: 异常处理

}

return ret;

}

public static Dialog getTaskMenuDialog(final TaskManager ctx, final DetailProcess dp) {

return new AlertDialog.Builder(ctx).setTitle(R.string.operation).setItems(

R.array.menu\_task\_operation, new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

switch (which) {

case MENU\_KILL: {

ctx.am.restartPackage(dp.getPackageName());

if (dp.getPackageName().equals(ctx.getPackageName())) return;

ctx.refresh();

return;

}

case MENU\_SWITCH: {

if (dp.getPackageName().equals(ctx.getPackageName())) return;

Intent i = dp.getIntent();

if (i == null) {

Toast.makeText(ctx, R.string.message\_switch\_fail, Toast.LENGTH\_LONG)

.show();

return;

}

try {

ctx.startActivity(i);

} catch (Exception ee) {

Toast.makeText(ctx, ee.getMessage(), Toast.LENGTH\_LONG).show();

}

return;

}

case MENU\_UNINSTALL: {

Uri uri = Uri.fromParts("package", dp.getPackageName(), null);

Intent it = new Intent(Intent.ACTION\_DELETE, uri);

try {

ctx.startActivity(it);

} catch (Exception e) {

Toast.makeText(ctx, e.getMessage(), Toast.LENGTH\_LONG).show();

}

return;

}

case MENU\_DETAIL: {

Intent detailsIntent = new Intent();

detailsIntent.setClassName("com.android.settings", "com.android.settings.InstalledAppDetails");

detailsIntent.putExtra("com.android.settings.ApplicationPkgName", dp.getPackageName());

ctx.startActivity(detailsIntent);

return;

}

}

}

}).create();

}

}

**IntentList.java:**

package com.xmobileapp.taskmanager;

import java.util.ArrayList;

import com.xmobileapp.taskmanager.tools.RunScript;

import com.xmobileapp.taskmanager.tools.StrUtil;

public class ProcessInfo {

private ArrayList<PsRow> pslist;

private static String rootpid = null;

public ProcessInfo() {

ps();

}

private void ps() {

String ps = RunScript.runIt("ps");

String[] lines = ps.split("\n");

pslist = new ArrayList<PsRow>();

for (String line : lines) {

PsRow row = new PsRow(line);

if (row.pid != null) pslist.add(row);

}

}

public PsRow getPsRow(String cmd) {

for (PsRow row : pslist) {

if (cmd.equals(row.cmd)) {

return row;

}

}

return null;

}

public static class PsRow {

String pid = null;

String cmd;

String ppid;

String user;

int mem;

public PsRow(String line) {

if (line == null) return;

String[] p = line.split("[\\s]+");

if (p.length != 9) return;

user = p[0];

pid = p[1];

ppid = p[2];

cmd = p[8];

mem = StrUtil.parseInt(p[4]);

if (isRoot()) {

rootpid = pid;

}

}

public boolean isRoot() {

return "zygote".equals(cmd);

}

public boolean isMain() {

return ppid.equals(rootpid) && user.startsWith("app\_");

}

/\*\*

\* Constructs a <code>String</code> with all attributes in name = value format.

\*

\* @return a <code>String</code> representation of this object.

\*/

public String toString() {

final String TAB = ";";

String retValue = "";

retValue = "PsRow ( " + super.toString() + TAB + "pid = " + this.pid + TAB + "cmd = " + this.cmd

+ TAB + "ppid = " + this.ppid + TAB + "user = " + this.user + TAB + "mem = " + this.mem

+ " )";

return retValue;

}

}

}

**MiscUtil.java:**

package com.xmobileapp.taskmanager;

import android.app.AlertDialog;

import android.app.Dialog;

import android.content.DialogInterface;

import android.content.Intent;

import android.content.pm.PackageInfo;

import android.content.pm.PackageManager;

import android.content.pm.PackageManager.NameNotFoundException;

import android.net.Uri;

import android.widget.Toast;

public class MiscUtil {

public static final int MENU\_CANCEL = 0;

public static final int MENU\_SWITCH = 1;

public static final int MENU\_KILL = 2;

public static final int MENU\_DETAIL = 3;

public static final int MENU\_UNINSTALL = 4;

public static PackageInfo getPackageInfo(PackageManager pm, String name) {

PackageInfo ret = null;

try {

ret = pm.getPackageInfo(name, PackageManager.GET\_ACTIVITIES);

} catch (NameNotFoundException e) {

//TODO: 异常处理

}

return ret;

}

public static Dialog getTaskMenuDialog(final TaskManager ctx, final DetailProcess dp) {

return new AlertDialog.Builder(ctx).setTitle(R.string.operation).setItems(

R.array.menu\_task\_operation, new DialogInterface.OnClickListener() {

public void onClick(DialogInterface dialog, int which) {

switch (which) {

case MENU\_KILL: {

ctx.am.restartPackage(dp.getPackageName());

if (dp.getPackageName().equals(ctx.getPackageName())) return;

ctx.refresh();

return;

}

case MENU\_SWITCH: {

if (dp.getPackageName().equals(ctx.getPackageName())) return;

Intent i = dp.getIntent();

if (i == null) {

Toast.makeText(ctx, R.string.message\_switch\_fail, Toast.LENGTH\_LONG)

.show();

return;

}

try {

ctx.startActivity(i);

} catch (Exception ee) {

Toast.makeText(ctx, ee.getMessage(), Toast.LENGTH\_LONG).show();

}

return;

}

case MENU\_UNINSTALL: {

Uri uri = Uri.fromParts("package", dp.getPackageName(), null);

Intent it = new Intent(Intent.ACTION\_DELETE, uri);

try {

ctx.startActivity(it);

} catch (Exception e) {

Toast.makeText(ctx, e.getMessage(), Toast.LENGTH\_LONG).show();

}

return;

}

case MENU\_DETAIL: {

Intent detailsIntent = new Intent();

detailsIntent.setClassName("com.android.settings", "com.android.settings.InstalledAppDetails");

detailsIntent.putExtra("com.android.settings.ApplicationPkgName", dp.getPackageName());

ctx.startActivity(detailsIntent);

return;

}

}

}

}).create();

}

}

**PackagesInfo.java:**

package com.xmobileapp.taskmanager;

import java.util.List;

import android.content.Context;

import android.content.pm.ApplicationInfo;

import android.content.pm.PackageManager;

public class PackagesInfo {

private List<ApplicationInfo> appList;

public PackagesInfo(Context ctx) {

PackageManager pm = ctx.getApplicationContext().getPackageManager();

appList = pm.getInstalledApplications(PackageManager.GET\_UNINSTALLED\_PACKAGES);

}

public ApplicationInfo getInfo(String name) {

if (name == null) {

return null;

}

for (ApplicationInfo appinfo : appList) {

if (name.equals(appinfo.processName)) {

return appinfo;

}

}

return null;

}

}

**ProcessInfo.java:**

package com.xmobileapp.taskmanager;

import java.util.ArrayList;

import com.xmobileapp.taskmanager.tools.RunScript;

import com.xmobileapp.taskmanager.tools.StrUtil;

public class ProcessInfo {

private ArrayList<PsRow> pslist;

private static String rootpid = null;

public ProcessInfo() {

ps();

}

private void ps() {

String ps = RunScript.runIt("ps");

String[] lines = ps.split("\n");

pslist = new ArrayList<PsRow>();

for (String line : lines) {

PsRow row = new PsRow(line);

if (row.pid != null) pslist.add(row);

}

}

public PsRow getPsRow(String cmd) {

for (PsRow row : pslist) {

if (cmd.equals(row.cmd)) {

return row;

}

}

return null;

}

public static class PsRow {

String pid = null;

String cmd;

String ppid;

String user;

int mem;

public PsRow(String line) {

if (line == null) return;

String[] p = line.split("[\\s]+");

if (p.length != 9) return;

user = p[0];

pid = p[1];

ppid = p[2];

cmd = p[8];

mem = StrUtil.parseInt(p[4]);

if (isRoot()) {

rootpid = pid;

}

}

public boolean isRoot() {

return "zygote".equals(cmd);

}

public boolean isMain() {

return ppid.equals(rootpid) && user.startsWith("app\_");

}

/\*\*

\* Constructs a <code>String</code> with all attributes in name = value format.

\*

\* @return a <code>String</code> representation of this object.

\*/

public String toString() {

final String TAB = ";";

String retValue = "";

retValue = "PsRow ( " + super.toString() + TAB + "pid = " + this.pid + TAB + "cmd = " + this.cmd

+ TAB + "ppid = " + this.ppid + TAB + "user = " + this.user + TAB + "mem = " + this.mem

+ " )";

return retValue;

}

}

}

**TaskListAdapters.java:**

package com.xmobileapp.taskmanager;

import java.util.ArrayList;

import com.xmobileapp.taskmanager.ProcessInfo.PsRow;

import android.content.pm.PackageManager;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.view.View.OnClickListener;

import android.widget.BaseAdapter;

import android.widget.ImageView;

import android.widget.TextView;

public class TaskListAdapters {

public final static class ProcessListAdapter extends BaseAdapter {

private LayoutInflater mInflater;

private ArrayList<DetailProcess> list;

private TaskManager ctx;

private PackageManager pm;

public ProcessListAdapter(TaskManager context, ArrayList<DetailProcess> list) {

// Cache the LayoutInflate to avoid asking for a new one each time.

mInflater = LayoutInflater.from(context);

this.list = list;

this.ctx = context;

this.pm = ctx.getPackageManager();

}

public int getCount() {

return list.size();

}

public Object getItem(int position) {

return position;

}

public long getItemId(int position) {

return position;

}

public View getView(final int position, View convertView, ViewGroup parent) {

ViewHolder holder;

if (convertView == null) {

//通过LayoutInflater对象整合ListView与新的在list\_main.xml中定义的布局

convertView = mInflater.inflate(R.layout.list\_main, null);

holder = new ViewHolder();

holder.icon = (ImageView) convertView.findViewById(R.id.list\_icon);

holder.text\_name = (TextView) convertView.findViewById(R.id.list\_name);

holder.text\_size = (TextView) convertView.findViewById(R.id.list\_size);

// 设置TAG，以便读写其中各个控件的内容

convertView.setTag(holder);

} else {

// 获取TAG

holder = (ViewHolder) convertView.getTag();

}

// 获取进程列表

final DetailProcess dp = list.get(position);

convertView.setVisibility(View.VISIBLE);

String cmd = dp.getRuninfo().processName;

// 通过TAG赋值

holder.icon.setImageDrawable(dp.getAppinfo().loadIcon(pm));

holder.text\_name.setText(dp.getTitle());

PsRow row = dp.getPsrow();

if (row == null) {

holder.text\_size.setText(R.string.memory\_unknown);

} else {

holder.text\_size.setText((int) Math.ceil(row.mem / 1024) + "K");

}

// 添加点击事件处理机制，以支持弹出菜单

convertView.setOnClickListener(new OnClickListener() {

@Override

public void onClick(View view) {

MiscUtil.getTaskMenuDialog(ctx, dp).show();

}

});

return convertView;

}

}

private static class ViewHolder {

ImageView icon;

TextView text\_name;

TextView text\_size;

}

}

**TaskManager.java:**

package com.xmobileapp.taskmanager;

import java.util.ArrayList;

import java.util.Collections;

import java.util.List;

import com.xmobileapp.taskmanager.TaskListAdapters.ProcessListAdapter;

import android.app.Activity;

import android.app.ActivityManager;

import android.app.ActivityManager.RunningAppProcessInfo;

import android.content.BroadcastReceiver;

import android.content.Context;

import android.content.Intent;

import android.content.IntentFilter;

import android.content.res.Configuration;

import android.os.Bundle;

import android.view.Window;

import android.widget.ListView;

public class TaskManager extends Activity {

public static final boolean DEBUG = true;

public static final String TAG = "TaskManager";

private ProcessInfo pinfo = null;

ActivityManager am = null;

private PackagesInfo packageinfo = null;

//private PackageManager pm;

protected static final String ACTION\_LOADFINISH = "com.xmobileapp.taskmanager.ACTION\_LOADFINISH";

private ProcessListAdapter adapter;

private BroadcastReceiver loadFinish = new LoadFinishReceiver();

/\*\* Called when the activity is first created. \*/

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

requestWindowFeature(Window.FEATURE\_INDETERMINATE\_PROGRESS);

setContentView(R.layout.main);

am = (ActivityManager) getSystemService(Context.ACTIVITY\_SERVICE);

//pm = this.getApplicationContext().getPackageManager();

packageinfo = new PackagesInfo(this);

IntentFilter filter = new IntentFilter(ACTION\_LOADFINISH);

this.registerReceiver(loadFinish, filter);

}

private ListView getListView() {

return (ListView) this.findViewById(R.id.listbody);

}

public void refresh() {

setProgressBarIndeterminateVisibility(true);

Thread t = new Thread(new Runnable() {

@Override

public void run() {

// TODO Auto-generated method stub

pinfo = new ProcessInfo();

getRunningProcess();

Intent in = new Intent(ACTION\_LOADFINISH);

TaskManager.this.sendBroadcast(in);

}

});

t.start();

}

public ProcessInfo getProcessInfo() {

return pinfo;

}

public PackagesInfo getPackageInfo() {

return packageinfo;

}

@Override

public void onConfigurationChanged(Configuration newConfig) {

super.onConfigurationChanged(newConfig);

}

//从其他Activity切回当前Activity时，进程列表要刷新

@Override

protected void onResume() {

super.onResume();

packageinfo = new PackagesInfo(this);

refresh();

}

@SuppressWarnings("unchecked")

public void getRunningProcess() {

List<RunningAppProcessInfo> list2 = am.getRunningAppProcesses();

ArrayList<DetailProcess> list = new ArrayList<DetailProcess>();

for (RunningAppProcessInfo ti : list2) {

if (ti.processName.equals("system") || ti.processName.equals("com.android.phone")) {

continue;

}

DetailProcess dp = new DetailProcess(this, ti);

dp.fetchApplicationInfo(packageinfo);

dp.fetchPackageInfo();

dp.fetchPsRow(pinfo);

if (dp.isGoodProcess()) {

list.add(dp);

}

}

Collections.sort(list);

adapter = new ProcessListAdapter(this, list);

}

private class LoadFinishReceiver extends BroadcastReceiver {

@Override

public void onReceive(final Context ctx, Intent intent) {

TaskManager.this.setProgressBarIndeterminateVisibility(false);

TaskManager.this.getListView().setAdapter(adapter);

}

} }

**Main.xml:**  
<?xml version="1.0" encoding="utf-8"?>

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

android:orientation="vertical" android:layout\_width="fill\_parent"

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

<RelativeLayout android:orientation="vertical"

android:id="@+id/main\_title\_layer" android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" android:layout\_alignParentTop="true"

android:layout\_centerInParent="true">

<ListView android:id="@+id/listbody" android:layout\_width="fill\_parent"

android:layout\_height="fill\_parent" android:scrollbars="none|vertical"

android:layout\_alignWithParentIfMissing="true" />

</RelativeLayout>

</RelativeLayout>

**List\_main.xml:**

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

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

android:layout\_width="fill\_parent" android:layout\_height="wrap\_content">

<ImageView android:id="@+id/list\_icon" android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content" android:layout\_alignParentTop="true"

android:layout\_alignParentBottom="true" android:layout\_marginRight="5dip"

android:adjustViewBounds="true" android:maxWidth="24dip" android:src="@drawable/icon"

android:maxHeight="24dip" />

<TextView android:id="@+id/list\_name" android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content" android:layout\_toRightOf="@id/list\_icon"

android:layout\_alignParentRight="true" android:layout\_alignParentTop="true"

android:layout\_alignWithParentIfMissing="true"

android:textColor="#FFFFFF" android:textStyle="normal"

android:textAppearance="?android:attr/textAppearanceMedium"

android:gravity="center\_vertical" android:text="test" />

<TextView android:id="@+id/list\_size"

android:layout\_alignParentRight="true" android:layout\_width="wrap\_content"

android:textAppearance="?android:attr/textAppearanceMedium"

android:layout\_height="wrap\_content" android:text="12334" />

</RelativeLayout>

**Arrays.xml:**  
<?xml version="1.0" encoding="utf-8"?>

<resources>

ed in contact edit menus -->

<string-array name="menu\_task\_operation">

<item>返回</item>

<item>切换至该程序</item>

<item>结束进程</item>

<item>详细信息</item>

<item>卸载程序</item>

</string-array>

</resources>

**Strings.xml:**

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

<resources>

<string name="app\_name">任务管理器</string>

<string name="operation">操作</string>

<string name="menu\_switch">切换至该程序</string>

<string name="menu\_kill">结束进程</string>

<string name="menu\_detail">详细信息</string>

<string name="menu\_uninstall">卸载该程序</string>

<string name="cancel">返回</string>

<string name="memory\_unknown">Unknown</string>

<string name="message\_switch\_fail">切换失败！</string>

</resources>