Atitit.android webview h5运行环境总结

[1. WebView 的使用 1](#_Toc24116)

[2. Js调用java 1](#_Toc199)

[3. Js调用java 跟个swt的比较 2](#_Toc24368)

[3.1. Swt是BrowserFunction 机制,绑定了个自定义方法 2](#_Toc2365)

[3.1.1. nativeswing的实现 预绑定一个sendNSCommand方法 2](#_Toc13353)

[4. code 2](#_Toc3419)

[5. Webview code 4](#_Toc13398)

# WebView 的使用

# Js调用java

browExt.play();

webView.addJavascriptInterface(**new** browExtObj(**this**), "browExt");

webView.loadUrl(url);

实现模式是绑定一个可以自定义浏览器对象

作者::  ★(attilax)>>> 绰号:老哇的爪子 （ 全名：：Attilax Akbar Al Rapanui 阿提拉克斯 阿克巴 阿尔 拉帕努伊 ） 汉字名：艾龙，  EMAIL:1466519819@qq.com

转载请注明来源： http://blog.csdn.net/attilax

# Js调用java 跟个swt的比较

## Swt是BrowserFunction 机制,绑定了个自定义方法

**new** CallJavaPaa1(browser, "sendNSCommand");

BrowserFunction 是个非常有意思的类，它可以为 Browser 永久绑定一个 JavaScript 方法，它的构造函数是 BrowserFunction(browser:Browser, name:String)，其中 browser 代表 Browser 对象，而 name 则代表绑定该浏览器的 JavaScript 方法名，定义了该 BrowserFunction 对象以后，任何在 Browser 显示的网页，都可以访问名为 name 的 JavaScript 方法。

### nativeswing的实现 预绑定一个sendNSCommand方法

 sendNSCommand('play',video);

# code

package com.example.atiplat\_vodcp;

import java.io.File;

import java.io.PrintWriter;

import java.io.StringWriter;

import java.net.URLEncoder;

import android.content.Intent;

import android.net.Uri;

import android.os.Environment;

import android.util.Log;

import android.webkit.JavascriptInterface;

import android.widget.Toast;

public class browExtObj {

public static String getTrace(Throwable t) {

StringWriter stringWriter = new StringWriter();

PrintWriter writer = new PrintWriter(stringWriter);

t.printStackTrace(writer);

StringBuffer buffer = stringWriter.getBuffer();

return buffer.toString();

}

MainActivity mainActivity;

public browExtObj(MainActivity mainActivity2) {

mainActivity = mainActivity2;

}

@JavascriptInterface //sdk17版本以上加上注解 solu click btn ma fein ..

public void play() {

try {

String mv = "smb://192.168.2.106/e/非蓝光/大头儿子小头爸爸/新大头儿子和小头爸爸之秘密计划.mp4";

Log.v("::::mv in html ", mv);

Toast.makeText(mainActivity,

"play in html,mv:" + new File(mv).exists(),

Toast.LENGTH\_LONG).show();

String mv2 = Environment.getExternalStorageDirectory().getPath()

+ "/Test\_Movie.m4v";

mv="http://127.0.0.1:7788/?file="+URLEncoder.encode(mv,"utf-8");

// Log.v("URI html:::::::::", uri.toString());

Uri uri = Uri.parse(mv);

// 调用系统自带的播放器

Intent intent = new Intent(Intent.ACTION\_VIEW);

Log.v("URI html:::::::::", uri.toString());

// intent.setData(uri);

intent.setDataAndType(uri, "video/\*");

mainActivity.startActivity(intent);

System.out.println("--form\_load finish");

} catch (Throwable e) {

Log.i("::::::::::::exp", getTrace(e));

}

}

# Webview code

WebView webView;

public static String getTrace(Throwable t) {

StringWriter stringWriter= new StringWriter();

PrintWriter writer= new PrintWriter(stringWriter);

t.printStackTrace(writer);

StringBuffer buffer= stringWriter.getBuffer();

return buffer.toString();

}

@SuppressLint("SetJavaScriptEnabled")

public void form\_load() {

try {

String simple = PinyinX.getSimple( "Android 提示");

new AlertDialog.Builder(this).setTitle(simple)

.setMessage("这是一个提示,请确定").show();

webView = (WebView) findViewById(R.id.webView1);

// 设置WebView属性，能够执行Javascript脚本

webView.getSettings().setJavaScriptEnabled(true);

webView.loadUrl("http://139.196.164.150:8080/lime/cmsWechat/list.html");

} catch (Throwable e) {

Log.i("exp", getTrace(e));

}

}

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

// //加载需要显示的网页

// webview.loadUrl("http://www.51cto.com/");

// //设置Web视图

//

setContentView(R.layout.activity\_main);

form\_load();

// setContentView(webView);

}