**pcs-note-demo介绍**

本文主要介绍如何使用百度PCS API开发基于android平台记事本（note）App的详细步骤，并对在代码中如何使用PCS API进行详细讲解。

pcs-note-demo完整代码下载地址：

<https://github.com/baidu-mco-evangelist>。

**一．首先介绍一下项目中主要文件夹和文件存放的内容和作用：**

**AndroidManifest.xml**

在该文件中申请使用PCS API所需的权限，并对项目中Activity进行配置。

**src/**

项目源码存放目录，该项目中包含两个包：com.baidu.pcs和com.baidu.pcsdemonote，具体内容如下：

com.baidu.pcs：定义了PCS 提供的15种 API接口，方便程序调用相应的接口实现具体的功能；

com.baidu.pcsdemonote：使用com.baidu.pcs中的接口实现note，并实现了各个Activity的具体功能。

**libs/**

libs文件夹中包含了百度官方提供两个jar包：httpmime-4.2.jar和oauth2\_api\_1.1.jar，开发者使用这两个jar包可实现与百度链接和获得服务器以及用户授权的功能。

**res/**

该目录下存放项目中需要使用的资源文件，包括image，layout，icon等，下面具体介绍各个目录的作用。

**drawable/**

存放实现点击button出现高亮效果的xml文件。

**drawable-hdpi/**

存放布局文件（Layout）中需要的image文件。

**layout/**

存放项目中登陆，内容显示，创建和编辑四个Activity的布局文件。

**values/**

String.xml定义了项目中使用到的字符串常量。方便系统的移植，适应不同的语言环境。

**二． 介绍使用PCS API实现note的具体步骤和对关键代码进行讲解。**1.修改AndroidManifest.xml文件，申请项目需要的权限

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

</uses-permission>

<uses-permission

android:name=*"android.permission.INTERNET"*>

</uses-permission>

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

</uses-permission>

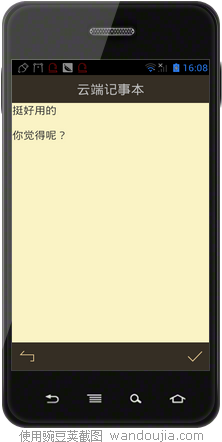
<uses-permission android:name=*"android.permission.READ\_PHONE\_STATE"*>

</uses-permission>

<uses-permission android:name=*"android.permission.ACCESS\_WIFI\_STATE"*>

</uses-permission>

2. 设计项目Activity的布局，本项目中设计了4个layout：登陆、内容显示、编辑和创建，实际效果图如下：

  
 登陆 内容显示 编辑 创建

3.在项目中创建libs文件夹，将httpmime-4.2.jar和oauth2\_api\_1.1.jar两个jar文件拷贝到该目录下，然后”右击libs目录” –>”Bulid path” –> “configure build path” -> “Libraries”->”Add JARs”，选择两个jar文件，点击“ok”，完成jar文件的导入。

4. 实现note的具体代码讲解，源代码放在“src/”目录下，该目录下有个包：（1）com.baidu.com定义PCS 15个API实现的接口，方便项目调用，也可以复用到其他的项目中；（2）com.baidu.pcsdemonote包是note的具体实现，包中主要包括6个文件，下面具体介绍各个文件的作用：

**PCSDemoInfo.java：**

该文件中定义了程序中需要用到的变量。

**BaiduPCSAction.java**

该文件中定义了登陆（login）、上传（upload）、获得文件列表（list）、删除文件（delete），下载文件（download）、保存文件（save），退出程序（exit）等方法，在具体的Activity中调用相应的方法实现具体的功能。

**PCSDemoNoteActivity.java：**

使用百度提供的oauth2\_api\_1.1.jar包实现与baidu连接的功能，获得access\_token。

**ContentActivity.java：**

内容显示Activity的具体实现，其中调用PCS list API实现显示PCS中的数据，创建记事本和刷新内容列表功能，调用 delete API实现文件和目录的删除功能。

**CreateActivity.java：**

创建记事本的Activity，在该Activity中使用upload API完成新记事本的创建，并将内容以txt的形式保存到PCS中。

**EditActivity.java：**

使用download API下载PCS上的文件，用户可对该文件进行编辑，编辑完成后上传到PCS。

下面具体介绍如何使用PCS API并对项目中的关键代码进行讲解。

（1）要想调用PCS API，首先使用百度账号与百度连接，通过用户用户授权获得access\_token，与百度连接获得access\_token是使用百度提供oauth2\_api\_1.1.jar中封装的BaiduOAuthViaDialog接口。具体方法如下：

首先创建BaiduOAuth的对象：

public BaiduOAuth bdOauth = null;

使用BaiduOAuthViaDialog接口，传入的参数值只有一个（百度开发者中心创建应用后分配的app\_key），具体实现方法如下：

bdOauth = new BaiduOAuthViaDialog(PCSDemoInfo.app\_key);

然后调用bdOauth的startDialogAuth方法：

bdOauth.startDialogAuth(context, new String[]{"basic", "netdisk"}, new BaiduOAuthViaDialog.DialogListener(){

public void onComplete(Bundle values) {

}

}

});

参数说明：

Context：获得当前的Activity；

String[]{“basic”,”netdisk”}：PCS使用权限申请，”basic”指的是基础权限，可以获取用户的基本信息；” netdisk”是获取用户在个人云存储中的数据；

BaiduOAuthViaDialog.DialogListener(){}：设置dialog的监听器；

onComplete(Bundle values){}：登陆成功后操作放在该函数内，values中存放授权成功后的响应信息，包括（access\_token、expires\_in、refresh\_token、scope等信息）

从values中获得access\_token:

PCSDemoInfo.access\_token = values.getString("access\_token");

到此完成了使用PCS的第一步，也是最重要的一步，因为后面所有API的使用都需要access\_token。

（2）在内容显示Activity中使用PCS 中的list和delete API，这两个API具体的使用方法在BaiduPCSAction.java文件中。两个API的作用如下：

list功能：获取个人云存储中文件（目录）列表；

delete功能 ：删除个人云存储中文件/目录（可以单个或者批量操作）

在调用PCS API之前需要为API设置access\_token，具体方法是先需要创建一个BaiduPCSAPI的对象，该类在com.baidu.pcs中有具体的定义。

BaiduPCSAPI api = new BaiduPCSAPI();

然后为api设置access\_token（由第一步中获取的access\_token）:

api.setAccessToken(PCSDemoInfo.access\_token );

list API的使用方法：首先设置需要显示文件列表的目录，当App申请使用PCS成功后，系统会自动在/apps目录为该App生成一个目录，该目录以App的名字命名，Demo App的应用名为“云端记事本”。

public final static String bdRootPath = "/apps/云端记事本/";

String path = PCSDemoInfo.bdRootPath;

配置完成后，开始使用list API：

final PCSActionInfo.PCSListInfoResponse listResponse = api.list(path, "name", "asc");

参数说明：

path：PCS上存储数据的目录；

name： list排序字段，可以选择按照 time、name、size排序

asc：排序顺序，asc（升序），desc（降序），缺省降序排序

响应信息存入listResponse中，listResponse是一个由PCSFileInfoResponse组成的list， 该list中PCSFileInfoResponse包含的主要内容介绍如下：

path：该文件在PCS上的绝对路径（包含文件名）

ctime: 文件在服务器上的创建时间

mtime：文件在服务器上的修改时间

size：文件的大小（byte）

isdir：0为文件，1为目录

下面介绍获取文件名和修改时间的方法，文件名包含在还回list中的path中，因为path是文件的绝对路径，所以从path中就可以获得文件名。

PCSFileInfoResponse info = i.next();

//Get the file name

String path = info.path; String fileName = path.substring(PCSDemoInfo.bdRootPath.length(),path.lastIndexOf("."));

从list中获得文件在服务器的修改时间方法：

Date date = new Date(info.mTime\*1000);

SimpleDateFormat formatter = new SimpleDateFormat("MM-dd HH:mm");

String dateString = formatter.format(date);

PCS上存储的mtime是以秒为单位的，而在Java中的系统时间是毫秒级的，所以首先要进行转换，再使用SimpleDateFormat的format()方法将date转化成相应的格式。

到此使用list API获得PCS上文件列表以及文件的详细信息的功能已经实现，哪在内容显示列表中如何删除PCS上的文件呢？实现这个功能就需要使用delete API，具体的过程如下。

首先需要设置要删除文件或者目录的在PCS上的位置（文件或者文件夹的绝对路径），这里files是list数据结构，可以批量的删除文件或者目录。

files.add(PCSDemoInfo.bdRootPath + PCSDemoInfo.fileTitle + ".txt");

然后调用delete API，该API中只传入一个参数，需要删除文件的list：

final PCSActionInfo.PCSSimplefiedResponse deleteResponse = api.deleteFiles(files);

通过访问deleteResponse.error\_code参数的值判断删除是否成功，如果为0则表示删除成功，不为0则表示删除失败。

（3）创建Activity中使用了upload API，使用upload的流程如下图：



注意：在使用upload上传文件时，一定要注意一点：PCS不支持文件的覆盖，当PCS上已存在该文件时，上传就会报错“文件已存在！”，这时的操作有两种选择：1.先删除PCS上的文件，然后上传；2.修改将要上传的文件名。

在应用中创建新note之前已经判断了文件名在PCS是否有重名，所以在创建界面保存文件时可以直接上传。具体方法如下：

首先将文件内容存入本地“data/应用程序包名/”目录下。

String saveFile = PCSDemoInfo.fileTitle+".txt";

FileOutputStream outputStream= context.openFileOutput(saveFile, Context.MODE\_PRIVATE);

outputStream.write(content)；

PCSDemoInfo.fileTitle：文件名 content为存储的文件内容

调用upload实现文件上传。

final PCSActionInfo.PCSFileInfoResponse uploadResponse = api.uploadFile(PCSDemoInfo.sourceFile, PCSDemoInfo.bdRootPath+PCSDemoInfo.fileTitle+".txt", new BaiduPCSStatusListener(){

public void onProgress(long bytes, long total) {

}

});

api.uploadFile( , , )参数说明：

第一个参数: 为文件在本地存储的绝对路径（带文件名）

PCSDemoInfo.sourceFile =

context.getFilesDir()+"/"+PCSDemoInfo.fileTitle+".txt";

第二个参数：PCS上存储的绝对路径

第三个参数：上传过程监控器，onProgress()中可添加过程处理操作

在uploadResponse中包括了PCS上成功创建文件后返回的信息，包括path、size、ctime、mtime、md5、fs\_id和isdir。

（4）在EditActivity中使用download、delete和upload实现文件的编辑和上传功能。流程图如下：

 在（3）中说明了PCS不支持文件的覆盖，所以编辑后的文件上传需要先删除PCS上相应的文件，然后再使用upload上传文件。

下面介绍如何使用download API实现文件下载的功能，首先需要定义文件在PCS上的绝对路径（sourceFile）和本地存储下载文件的绝对路径（target）。

PCSDemoInfo.sourceFile = PCSDemoInfo.bdRootPath + PCSDemoInfo.fileTitle+".txt";

PCSDemoInfo.target = context.getFilesDir()+"/"+

PCSDemoInfo.fileTitle+".txt";

设置完成后，使用download API下载文件。

final PCSActionInfo.PCSSimplefiedResponse downloadResponse = api.downloadFile(PCSDemoInfo.sourceFile, PCSDemoInfo.target, new BaiduPCSStatusListener(){

public void onProgress(long bytes, long total) {

}

});

参数说明：

第一个参数：文件在PCS上的绝对路径；

第二个参数：本地存储文件的绝对路径；

第三个参数：下载过程的监听器，在onProgress(long bytes, long total)中可加入进行过程处理操作

通过downloadResponse.error\_code的值判断是否下载成功，为0则表示下载成功，不为0则下载失败。当下载完成后将文件中的内容读入Activity中，用户编辑完成后使用点击“save”保存文件，先使用（2）中的delete删除PCS上相应的文件，再使用（3）中的upload上传文件，在这里就不在重复讲解了。

至此，note中使用到的PCS API讲解结束，本文主要讲解了如何使用百度PCS API搭建记事本（note）App的过程，并对如何使用百度PCS API做了详细的解析，关于android部分的代码可下载项目的完整代码查看，这里就不做详细解释，如果您想使用其他的API，可以添加src/com.baidu.pcs包到你的工程文件中，参考上面PCS API使用方法的讲解使用其他API开发自己的App。