

人人网开放平台 Windows Phone 7 SDK v2.0 说明文档

人人网开放平台

目录

一、产品特点.....	3
二、SDK接口介绍	3
RenrenAPI接口说明.....	3
三、基本功能使用说明.....	5
使用Windows Phone 7 SDK的准备工作	5
使用Windows Phone 7 SDK提供的页面授权方法	5
使用Windows Phone 7 SDK提供的用户名密码授权方式	6
使用Windows Phone 7 SDK调用人人API接口 （以取得用户信息为例）	6
使用Windows Phone 7 SDK完成一键上传照片	7
使用Windows Phone 7 SDK完成通用API接口的调用 （以发表日志为例）	7
使用Windows Phone 7 SDK提供的widget dialog （以发布状态为例）	8
使用Windows Phone 7 SDK退出授权状态	9
四、Demo示范.....	9

一、产品特点

- 全面支持人人网开放平台的 API 接口调用
- 提供客户端和用户名密码两种授权方式
- 对 SDK 进行优化和改进，使用方便和高效
- 授权页面方式提供 WEB 控件的组入方式，定制性更强

二、SDK 接口介绍

RenrenAPI 接口说明

序号	类型	功能	接口	状态
1	登录、授权与登出	授权页面方式登录(无权限设置)	<code>public void Login(PhoneApplicationPage page, LoginCompletedHandler callback)</code>	原有
2		授权页面方式登录	<code>public void Login(PhoneApplicationPage page, List<string> scope, LoginCompletedHandler callback)</code>	原有
3		用户名密码方式登录(无权限设置)	<code>public void Login(string username, string password, LoginCompletedHandler callback)</code>	变更
4		用户名密码方式登录	<code>public void Login(string username, string password, List<string> scope, LoginCompletedHandler callback)</code>	变更
5		判断用户授权状态	<code>public bool IsAccessTokenValid()</code>	新增
6		登出	<code>public void Logout()</code>	原有
7	一键发布	一键上传照片	<code>public void PublishPhotoSimple(BitmapImage photo, string imgPath, string caption)</code>	原有
8	常用功能	获取好友 id 列表	<code>public void GetFriendsID(GetFriendsIDCompletedHandler callback, int count=500, int page=1)</code>	原有
9		获取好友信息列表(可设置字段)	<code>public void GetFriends(GetFriendsCompletedHandler</code>	原有

			callback, List<string> scope, int count=500, int page=1)	
10		获取好友信息列表（返回默认字段）	public void GetFriends(GetFriendsCompletedHandler callback, int count=500, int page=1)	原有
11		获取 App 好友信息列表（可设置字段）	public void GetAppFriends(List<string> scope, GetAppFriendsCompletedHandler callback)	原有
12		获取 App 好友信息列表（返回默认字段）	public void GetAppFriends(GetAppFriendsIDCompletedHandler callback)	原有
13		创建相册	public void CreateAlbum(CreateAlbumRequest.CreateAlbumCompletedHandler callback, string name, string description=null, string location=null)	原有
14		获取当前用户相册列表	public void GetAlbums(RenrenSDKLibrary.GetAlbumsRequest.GetAlbumsCompletedHandler callback, int page = -1, int count = -1, string aids = null)	原有
15		上传照片	public void PublishPhoto(BitmapImage photo, string name, UploadPhotoCompletedHandler callback, string caption = null, int albumId = 0)	原有
16		上传照片	public void PublishPhoto(string fileName, UploadPhotoCompletedHandler callback, string caption = null, int albumId = 0)	原有
17		获取当前用户信息	public void GetCurUserInfo(GetCurUserInfoCompletedHandler callback)	原有
18		获取指定用户信息	public void GetUserInfo(string uid, List<string> scope, GetUserInfoCompletedHandler callback)	原有
19	通用方法	通用 API 接口调用方法	public void RequestAPIInterface(APIRequestCompletedHandler callback, List<APIParameter> param)	新增
20		通用 WidgetAPI 调用方法	public void WidgetDialog(PhoneApplicationPage page, string dialogType, List<APIParameter> param,	新增

			RenrenSDKLibrary.WidgetDialog.WidgetAPI RequestBS.DownloadStringCompletedHandle r_callback = null)	
21	辅助方法	读取 log	public string ReadLog()	原有
22		清空 log	public void Cleanlog()	原有

三、基本功能使用说明

使用 Windows Phone 7 SDK 的准备工作

1. 请将 RenrenSDKLibrary 文件夹复制到你的 Windows Phone 7 工程目录下
2. 在 Solution Explorer 选中你的 Solution，压下鼠标右键->Add->Existing Project->找到 RenrenSDKLibrary 中的 RenrenSDKLibrary.csproj 文件，点击打开。
3. 在你的工程中添加 RenrenSDKLibrary 的 Reference。
4. 在需要使用的使用人人 SDK 接口的文件中，使用 using RenrenSDKLibrary;

PS: 运行环境请使用 windows phone7.1 RC，7.0 版本不能引用

使用 Windows Phone 7 SDK 提供的页面授权方法

1. 首先，创建 RenrenAPI 的对象:

```
RenrenAPI api;
```

```
.....
```

```
api = new RenrenAPI("YOUR APP ID", "YOUR API KEY", "YOUR SECRET");
```

将上面的 YOUR APP ID、YOUR API KEY 和 YOUR SECRET 内容配置为你在人人开发平台上申请的应用信息。

2. **api.IsAccessTokenValid()**

这个方法可以判断当前用户是否是授权状态

3. **api.Login(this, renren_LoginCompletedHandler);**

通过这个方法，用户可以进行登录验证授权

this 是指当前 Page，renren_LoginCompletedHandler 是授权完成的处理函数，它的具体实现如下

```
public void renren_LoginCompletedHandler(object sender, LoginCompletedEventArgs e)
{
    if (e.Error == null)
        // 发生错误的处理
    else
        // 授权成功的处理
}
```

4. 如果需要授予多个权限的话，可以这样

```
List<string> scope = new List<string> { "publish_feed", "publish_blog", "read_user_album",
    "create_album", "photo_upload" };
```

```
api.Login(this, scope, renren_LoginCompletedHandler);
```

使用 Windows Phone 7 SDK 提供的用户名密码授权方式

1. 请先创建 RenrenAPI 的对象；

```
2. api.Login("User's Account", "User's Password", renren_LoginCompletedHandler);
```

其中第一个参数是用户的账号，第二参数是用户的密码，renren_LoginCompletedHandler 是授权完成的处理函数（具体接受请参看前面章节）

4. 如果需要授予多个权限的话，可以这样

```
List<string> scope = new List<string> { "publish_feed", "publish_blog", "read_user_album",
    "create_album", "photo_upload" };
```

```
api.Login("User's Account", "User's Password", scope, renren_LoginCompletedHandler);
```

使用 Windows Phone 7 SDK 调用人人 API 接口（以取得用户信息为例）

1. 请先创建 RenrenAPI 对象，并引导用户完成授权。

2. 设置需要取得的字段信息

```
List<string> scope = new List<string>();
scope.Add("uid");
scope.Add("name");
scope.Add("sex");
scope.Add("star");
scope.Add("work_history");
```

```
scope.Add("vip");
```

```
api.GetUserInfo("User's id", scope, renren_GetUserInfoCompletedHandler);
```

其中，第一个参数可以为多个用户 ID，中间用“,”间隔，用来取得多个用户的信息，例如“1234567,7654321,……”

最后一个参数是调用取得用户信息的回调函数，用于返回结果的处理。具体定义如下，

```
public void renren_GetUserInfoCompletedHandler(object sender, GetUsersCompletedEventArgs e)
{
    if (e.Error != null)
    {
        //错误处理
    }
    else
    {
        //对返回结果的处理
    }
}
```

使用 Windows Phone 7 SDK 完成一键上传照片

1.请先创建 RenrenAPI 对象，并引导用户完成授权。

2.完成授权后，创建 Image 的对象，

```
StreamResourceInfo info = Application.GetResourceStream(new Uri("Background.png",
UriKind.Relative));
BitmapImage bmp = new BitmapImage();
bmp.SetSource(info.Stream);
```

3. `api.PublishPhotoSimple(bmp, "Background.png", "照片描述");`

调用上面的方法即可完成上传照片的功能。

使用 Windows Phone 7 SDK 完成通用 API 接口的调用（以发表日志为例）

1.请先创建 RenrenAPI 对象，并引导用户完成授权。

2.完成授权后，创建参数列表

```
List<APIParameter> param = new List<APIParameter>();
```

```
param.Add(new APIParameter("method","blog.addBlog"));
param.Add(new APIParameter("title",tb_blogtitle.Text));
param.Add(new APIParameter("content",tb_blogcontent.Text));
```

诸如 access_token, call_id, sig, v 等参数信息, 开发者可以不用关注, 这些参数 SDK 已经帮你封装好了, 您只需要关注和接口密切相关参数就可以了。

3. *api.RequestAPIInterface(PublishBlogCompletedCallBack, param);*

调用上面的方法即可完成发表日志的功能。其中 PublishBlogCompletedCallBack 是 API 请求的回调方法, 用于处理请求结果, 具体实现如下

```
private void PublishBlogCompletedCallBack(object sender, APIRequestCompletedEventArgs e)
{
    if (e.Error == null)
    {
        //请求结果的处理, 解析JSON 数据
    }
    else
    {
        //错误处理
    }
}
```

使用 Windows Phone 7 SDK 提供的 widget dialog （以发布状态为例）

1. 请先创建 RenrenAPI 对象, 并引导用户完成授权。

2. 完成授权后, 创建请求 widget dialog 参数列表

```
List<APIParameter> param= new List<APIParameter>();
param.Add(new APIParameter("url","http://dev.renren.com"));
param.Add(new APIParameter("name", "人人网开放平台"));
param.Add(new APIParameter("action_name","访问我们"));
param.Add(new APIParameter("action_link","http://dev.renren.com"));
param.Add(new APIParameter("description","来自人人网 Windows Phone 7 SDK"));
param.Add(new APIParameter("caption","欢迎使用人人网 SDK"));
param.Add(new APIParameter("image","http://hdn.xnimg.cn/photos/hdn421/20090923/1935/head_1Wmz_19242g019116.jpg"));
```

3. *api.WidgetDialog(this, WidgetDialogType.FeedDialog, param, WidgetDialogComplete);*

调用上面的方法即可完成日志发表的功能。第一个参数是当前 Page, 第二个参数是 WidgetDialog 的类型, 最后一个参数是请求结果的回调函数, 其具体实现如下,


```
void WidgetDialogComplete(object sender,  
RenrenSDKLibrary.DownloadStringCompletedEventArgs e)  
{  
    if (e.Error != null)  
        // 错误处理  
    else  
        // 返回结果处理  
}
```

使用 Windows Phone 7 SDK 退出授权状态

1 在授权以后，通过调用下面的方法退出授权状态。

`api.LogOut();`

四、Demo 示范





人人网 Windows Phone 7 SDK 负责人：夏文海
E-mail:wenhai.xia@renren-inc.com
QQ:86852962