# iOS接入指南 [在线查看](https://open.weixin.qq.com/cgi-bin/showdocument?action=dir_list&t=resource/res_list&verify=1&id=1417694084&token=&lang=zh_CN)

#### 1.向微信注册你的应用程序id

请到 开发者应用登记页面 进行登记，登记并选择移动应用进行设置后，将获得AppID，可立即用于开发。但应用登记完成后还需要提交审核，只有审核通过的应用才能正式发布使用。

#### 2.下载微信终端SDK文件

SDK文件包括 libWeChatSDK.a，WXApi.h，WXApiObject.h 三个。  
如选用手动集成，请前往“[资源下载页](https://open.weixin.qq.com/cgi-bin/showdocument?action=dir_list&t=resource/res_list&verify=1&id=open1419319164&lang=zh_CN)”下载最新SDK包

#### 3.搭建开发环境

#### 3.1 通过CocoaPods集成

#### 4.在代码中使用开发工具包

## [微信SDK Sample Demo源码](https://open.weixin.qq.com/zh_CN/htmledition/res/dev/download/sdk/WeChatSDK_sample_iOS_1.4.2.1.zip)

# Android接入指南 [在线查看](https://open.weixin.qq.com/cgi-bin/showdocument?action=dir_list&t=resource/res_list&verify=1&id=1417751808&token=&lang=zh_CN)

#### 1.申请你的AppID

请到 开发者应用登记页面 进行登记，登记并选择移动应用进行设置后，将该应用提交审核，只有审核通过的应用才能进行开发。

#### 2.下载SDK及API文档

Android Studio环境下：

在build.gradle文件中，添加如下依赖即可：

dependencies {

    compile 'com.tencent.mm.opensdk:wechat-sdk-android-with-mta:+'

}

或

dependencies {

    compile 'com.tencent.mm.opensdk:wechat-sdk-android-without-mta:+'

}

（其中，前者包含统计功能）

Eclipse环境下：

请前往“[资源下载页](https://open.weixin.qq.com/cgi-bin/showdocument?action=dir_list&t=resource/res_list&verify=1&id=open1419319167&lang=zh_CN)”下载最新SDK包。

#### 3.搭建开发环境

Android Studio环境下：

在Android Studio中新建你的工程，并保证网络设置可以成功从jcenter下载微信SDK即可。

#### 4.在代码中使用开发工具包

[微信SDK Sample Demo源码](https://open.weixin.qq.com/zh_CN/htmledition/res/dev/download/sdk/WeChatSDK_sample_Android.zip)

# APP移动应用微信登录开发

#### 准备工作

移动应用微信登录是基于[OAuth2.0协议标准](http://oauth.net/2/)构建的微信OAuth2.0授权登录系统。

在进行微信OAuth2.0授权登录接入之前，在微信开放平台注册开发者帐号，并拥有一个已审核通过的移动应用，并获得相应的AppID和AppSecret，申请微信登录且通过审核后，可开始接入流程。

1. 1、目前移动应用上微信登录只提供原生的登录方式，需要用户安装微信客户端才能配合使用。
2. 2、对于Android应用，建议总是显示微信登录按钮，当用户手机没有安装微信客户端时，请引导用户下载安装微信客户端。
3. 3、对于iOS应用，考虑到iOS应用商店审核指南中的相关规定，建议开发者接入微信登录时，先检测用户手机是否已安装微信客户端（使用sdk中isWXAppInstalled函数 ），对未安装的用户隐藏微信登录按钮，只提供其他登录方式（比如手机号注册登录、游客登录等）。





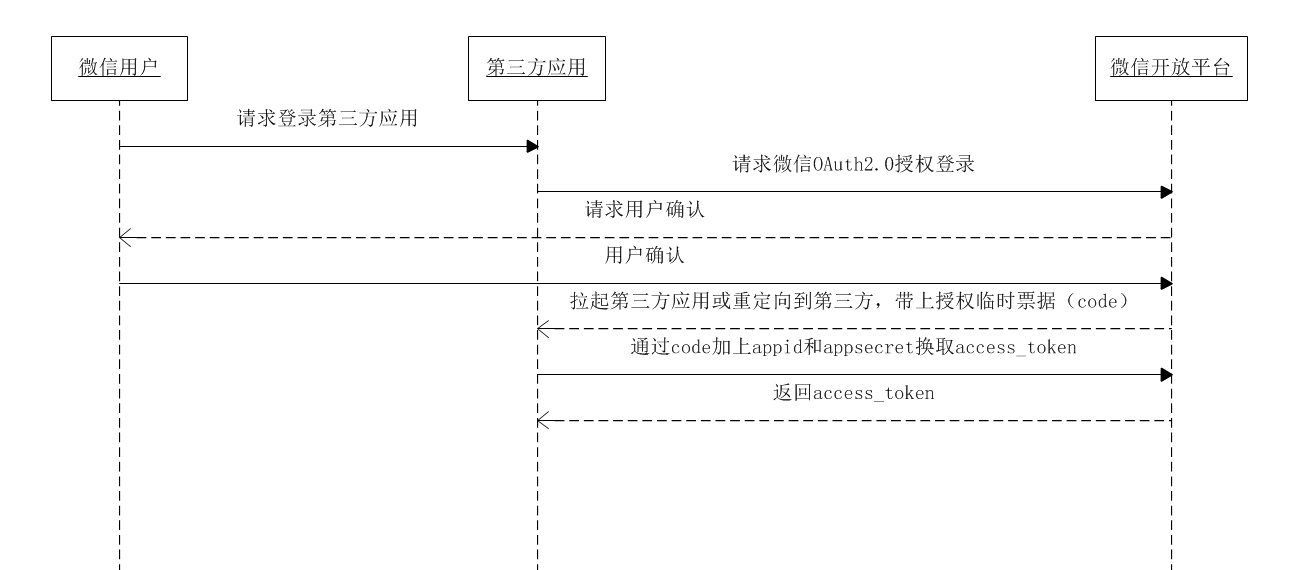
#### 授权流程说明

微信OAuth2.0授权登录让微信用户使用微信身份安全登录第三方应用或网站，在微信用户授权登录已接入微信OAuth2.0的第三方应用后，第三方可以获取到用户的接口调用凭证（access\_token），通过access\_token可以进行微信开放平台授权关系接口调用，从而可实现获取微信用户基本开放信息和帮助用户实现基础开放功能等。

微信OAuth2.0授权登录目前支持authorization\_code模式，适用于拥有server端的应用授权。该模式整体流程为：

1. 1. 第三方发起微信授权登录请求，微信用户允许授权第三方应用后，微信会拉起应用或重定向到第三方网站，并且带上授权临时票据code参数；
2. 2. 通过code参数加上AppID和AppSecret等，通过API换取access\_token；
3. 3. 通过access\_token进行接口调用，获取用户基本数据资源或帮助用户实现基本操作。

获取access\_token时序图：



##### 第一步：请求CODE

###### 移动应用微信授权登录

开发者需要配合使用微信开放平台提供的SDK进行授权登录请求接入。正确接入SDK后并拥有相关授权域（scope，[什么是授权域？](https://open.weixin.qq.com/cgi-bin/showdocument?action=doc&id=open1419317851&t=0.8273132876527127" \l "scope" \o "什么是授权域？" \t "_self)）权限后，开发者移动应用会在终端本地拉起微信应用进行授权登录，微信用户确认后微信将拉起开发者移动应用，并带上授权临时票据（code）。

iOS平台应用授权登录接入代码示例（请参考iOS接入指南）：

-(void)sendAuthRequest  
{   
    //构造SendAuthReq结构体   
    SendAuthReq\* req =[[[SendAuthReq alloc ] init ] autorelease ];  
    req.scope = @"snsapi\_userinfo" ;  
    req.state = @"123" ;  
    //第三方向微信终端发送一个SendAuthReq消息结构  
    [WXApi sendReq:req]; }

Android平台应用授权登录接入代码示例（请参考Android接入指南）：

{   
    // send oauth request   
    Final SendAuth.Req req = new SendAuth.Req();  
    req.scope = "snsapi\_userinfo";  
    req.state = "wechat\_sdk\_demo\_test";  
    api.sendReq(req);  
}

###### 参数说明

| 参数 | 是否必须 | 说明 |
| --- | --- | --- |
| appid | 是 | 应用唯一标识，在微信开放平台提交应用审核通过后获得 |
| scope | 是 | 应用授权作用域，如获取用户个人信息则填写snsapi\_userinfo（[什么是授权域？](https://open.weixin.qq.com/cgi-bin/showdocument?action=doc&id=open1419317851&t=0.8273132876527127#scope)） |
| state | 否 | 用于保持请求和回调的状态，授权请求后原样带回给第三方。该参数可用于防止csrf攻击（跨站请求伪造攻击），建议第三方带上该参数，可设置为简单的随机数加session进行校验 |

**返回示例：**

appid: wxd477edab60670232  
scope: snsapi\_userinfo  
state: wechat\_sdk\_demo

###### 返回说明

用户点击授权后，微信客户端会被拉起，跳转至授权界面，用户在该界面点击允许或取消，SDK通过SendAuth的Resp返回数据给调用方。

| 返回值 | 说明 |
| --- | --- |
| ErrCode | ERR\_OK = 0(用户同意) ERR\_AUTH\_DENIED = -4（用户拒绝授权） ERR\_USER\_CANCEL = -2（用户取消） |
| code | 用户换取access\_token的code，仅在ErrCode为0时有效 |
| state | 第三方程序发送时用来标识其请求的唯一性的标志，由第三方程序调用sendReq时传入，由微信终端回传，state字符串长度不能超过1K |
| lang | 微信客户端当前语言 |
| country | 微信用户当前国家信息 |

##### 第二步：通过code获取access\_token

获取第一步的code后，请求以下链接获取access\_token：

https://api.weixin.qq.com/sns/oauth2/access\_token?appid=APPID&secret=SECRET&code=CODE&grant\_type=authorization\_code

###### 参数说明

| 参数 | 是否必须 | 说明 |
| --- | --- | --- |
| appid | 是 | 应用唯一标识，在微信开放平台提交应用审核通过后获得 |
| secret | 是 | 应用密钥AppSecret，在微信开放平台提交应用审核通过后获得 |
| code | 是 | 填写第一步获取的code参数 |
| grant\_type | 是 | 填authorization\_code |

###### 返回说明

正确的返回：

{   
"access\_token":"ACCESS\_TOKEN",   
"expires\_in":7200,   
"refresh\_token":"REFRESH\_TOKEN",  
"openid":"OPENID",   
"scope":"SCOPE",

"unionid":"o6\_bmasdasdsad6\_2sgVt7hMZOPfL"

}

| 参数 | 说明 |
| --- | --- |
| access\_token | 接口调用凭证 |
| expires\_in | access\_token接口调用凭证超时时间，单位（秒） |
| refresh\_token | 用户刷新access\_token |
| openid | 授权用户唯一标识 |
| scope | 用户授权的作用域，使用逗号（,）分隔 |
| unionid | 当且仅当该移动应用已获得该用户的userinfo授权时，才会出现该字段 |

**错误返回样例：**

{"errcode":40029,"errmsg":"invalid code"}

##### 刷新access\_token有效期

access\_token是调用授权关系接口的调用凭证，由于access\_token有效期（目前为2个小时）较短，当access\_token超时后，可以使用refresh\_token进行刷新，access\_token刷新结果有两种：

1. 1. 若access\_token已超时，那么进行refresh\_token会获取一个新的access\_token，新的超时时间；
2. 2. 若access\_token未超时，那么进行refresh\_token不会改变access\_token，但超时时间会刷新，相当于续期access\_token。

refresh\_token拥有较长的有效期（30天），当refresh\_token失效的后，需要用户重新授权。

###### 请求方法

获取第一步的code后，请求以下链接进行refresh\_token：

https://api.weixin.qq.com/sns/oauth2/refresh\_token?appid=APPID&grant\_type=refresh\_token&refresh\_token=REFRESH\_TOKEN

###### 参数说明

| 参数 | 是否必须 | 说明 |
| --- | --- | --- |
| appid | 是 | 应用唯一标识 |
| grant\_type | 是 | 填refresh\_token |
| refresh\_token | 是 | 填写通过access\_token获取到的refresh\_token参数 |

###### 返回说明

正确的返回：

{   
"access\_token":"ACCESS\_TOKEN",   
"expires\_in":7200,   
"refresh\_token":"REFRESH\_TOKEN",   
"openid":"OPENID",   
"scope":"SCOPE"   
}

| 参数 | 说明 |
| --- | --- |
| access\_token | 接口调用凭证 |
| expires\_in | access\_token接口调用凭证超时时间，单位（秒） |
| refresh\_token | 用户刷新access\_token |
| openid | 授权用户唯一标识 |
| scope | 用户授权的作用域，使用逗号（,）分隔 |

错误返回样例：

{"errcode":40030,"errmsg":"invalid refresh\_token"}

##### 注意：

1、Appsecret 是应用接口使用密钥，泄漏后将可能导致应用数据泄漏、应用的用户数据泄漏等高风险后果；存储在客户端，极有可能被恶意窃取（如反编译获取Appsecret）；

2、access\_token 为用户授权第三方应用发起接口调用的凭证（相当于用户登录态），存储在客户端，可能出现恶意获取access\_token 后导致的用户数据泄漏、用户微信相关接口功能被恶意发起等行为；

3、refresh\_token 为用户授权第三方应用的长效凭证，仅用于刷新access\_token，但泄漏后相当于access\_token 泄漏，风险同上。

**建议将Appsecret、用户数据（如access\_token）放在App云端服务器，由云端中转接口调用请求。**

##### 第三步：通过access\_token调用接口

获取access\_token后，进行接口调用，有以下前提：

1. 1. access\_token有效且未超时；
2. 2. 微信用户已授权给第三方应用帐号相应接口作用域（scope）。

对于接口作用域（scope），能调用的接口有以下：

| 授权作用域（scope） | 接口 | 接口说明 |
| --- | --- | --- |
| snsapi\_base | /sns/oauth2/access\_token | 通过code换取access\_token、refresh\_token和已授权scope |
| /sns/oauth2/refresh\_token | 刷新或续期access\_token使用 |
| /sns/auth | 检查access\_token有效性 |
| snsapi\_userinfo | /sns/userinfo | 获取用户个人信息 |

其中snsapi\_base属于基础接口，若应用已拥有其它scope权限，则默认拥有snsapi\_base的权限。使用snsapi\_base可以让移动端网页授权绕过跳转授权登录页请求用户授权的动作，直接跳转第三方网页带上授权临时票据（code），但会使得用户已授权作用域（scope）仅为snsapi\_base，从而导致无法获取到需要用户授权才允许获得的数据和基础功能。

接口调用方法可查阅[《微信授权关系接口调用指南》](https://open.weixin.qq.com/cgi-bin/showdocument?action=dir_list&t=resource/res_list&verify=1&id=open1419317853&lang=zh_CN)

##### F.A.Q

###### 1. 什么是授权临时票据（code）？

答：第三方通过code进行获取access\_token的时候需要用到，code的超时时间为10分钟，一个code只能成功换取一次access\_token即失效。code的临时性和一次保障了微信授权登录的安全性。第三方可通过使用https和state参数，进一步加强自身授权登录的安全性。

###### 2. 什么是授权作用域（scope）？

答：授权作用域（scope）代表用户授权给第三方的接口权限，第三方应用需要向微信开放平台申请使用相应scope的权限后，使用文档所述方式让用户进行授权，经过用户授权，获取到相应access\_token后方可对接口进行调用。

###### 3.开放平台移动应用微信登陆目前是否收费？

答：“微信登录”和第三方网站共享微信庞大的用户价值，同时为微信用户提供更便捷服务和更优质内容，实现双向共赢，目前不收取任何费用。