中国联通沃商店UniPay统一账户SDK开发者帮助文档

V2.0.1.1

版权所有：中国联通沃商店游戏中心

修订日期：2015-1-28

版本修订记录

|  |  |  |
| --- | --- | --- |
| 版本号 | 更新时间 | 更新内容 |
| V2.0.0 | 2014-8-6 | 新建 |
| V2.0.1 | 2014-10-20 | 对应SDK2.0.1版 |
| V2.0.1.1 | 2015-1-28 | “切换账户”API改为必须接入的接口  详见 [切换账户API(必须接入)](#_切换账户API(必须接入))  “进入用户中心”API改为必须接入的接口  详见 [进入用户中心API(必须接入)](#_进入用户中心API(必须接入))  新增 6.服务端验证登录 |

# 目录

[版本修订记录 2](#_Toc1564)

[目录 3](#_Toc29799)

[1. 概述 6](#_Toc6144)

[2. 术语、定义和缩略语 6](#_Toc8828)

[2.1. 术语、定义 6](#_Toc8483)

[2.2. 缩略语 6](#_Toc4363)

[3. 开发者注册及集成账户能力获取 7](#_Toc27436)

[3.1. 使用规范 7](#_Toc22251)

[3.2. 开发者注册 7](#_Toc32313)

[3.3. 合作签约 8](#_Toc32221)

[3.4. Unipay账户SDK开发包获取 8](#_Toc16058)

[3.5. 新增作品 8](#_Toc7035)

[3.6. 申请统一账户能力 9](#_Toc19231)

[4. SDK接入指南 11](#_Toc16043)

[4.1. 配置工程 11](#_Toc8201)

[4.1.1. 将SDK包导入应用工程 11](#_Toc15770)

[4.1.2. 配置AndroidManifest 13](#_Toc32471)

[4.1.3. 混淆设置 14](#_Toc4227)

[4.1.4. 参数准备 14](#_Toc12742)

[4.2. 接入SDK初始化/销毁API 15](#_Toc2007)

[4.2.1. 初始化SDK 15](#_Toc25094)

[4.2.2. 销毁SDK 16](#_Toc24705)

[4.2.3. 设置账户状态变化事件通知回调 16](#_Toc5994)

[4.3. 接入登录和注销API 18](#_Toc3922)

[4.3.1. 登录 18](#_Toc30365)

[4.3.2. 注销 20](#_Toc9445)

[4.4. 切换账户API(必须接入) 21](#_Toc14852)

[4.5. 进入用户中心API(必须接入) 22](#_Toc30065)

[4.6. 接入可选API 23](#_Toc696)

[4.6.1. 悬浮工具栏 23](#_Toc19560)

[4.6.2. SDK屏幕方向设置 24](#_Toc19001)

[4.6.3. 其他API 24](#_Toc29141)

[4.7. 应用发布 25](#_Toc17095)

[5. API说明 26](#_Toc14289)

[5.1. 初始化SDK 26](#_Toc9097)

[5.1.1. 接口详细 26](#_Toc14477)

[5.1.2. 结果回调 27](#_Toc1771)

[5.1.3. 示例代码 27](#_Toc29123)

[5.2. 销毁SDK 29](#_Toc29908)

[5.2.1. 接口详细 29](#_Toc4233)

[5.2.2. 示例代码 29](#_Toc16291)

[5.3. 设置账户变化事件回调 30](#_Toc6361)

[5.3.1. 接口详细 30](#_Toc21020)

[5.3.2. 结果回调 31](#_Toc11857)

[5.3.3. 示例代码 31](#_Toc15783)

[5.4. 登录账户 33](#_Toc12277)

[5.4.1. 接口详细 33](#_Toc5004)

[5.4.2. 结果回调 34](#_Toc9975)

[5.4.3. 示例代码 34](#_Toc10516)

[5.5. 注销账户 36](#_Toc18644)

[5.5.1. 接口详细 36](#_Toc9657)

[5.5.2. 结果回调 37](#_Toc3913)

[5.5.3. 示例代码 37](#_Toc29642)

[5.6. 切换账户 39](#_Toc15590)

[5.6.1. 接口详细 39](#_Toc12073)

[5.6.2. 结果回调 40](#_Toc6988)

[5.6.3. 示例代码 40](#_Toc19218)

[5.7. 获取当前登录状态 41](#_Toc26717)

[5.7.1. 接口详细 41](#_Toc12050)

[5.7.2. 示例代码 41](#_Toc20966)

[5.8. 获取当前用户信息 43](#_Toc19589)

[5.8.1. 接口详细 43](#_Toc7044)

[5.8.2. 用户信息详细 44](#_Toc31075)

[5.8.3. 示例代码 44](#_Toc10459)

[5.9. 设置悬浮按钮可见性 45](#_Toc553)

[5.9.1. 接口详细 45](#_Toc13907)

[5.9.2. 示例代码 45](#_Toc13320)

[5.10. 设置SDK界面方向 46](#_Toc7419)

[5.10.1. 界面方向设置参数详细 46](#_Toc27100)

[5.10.2. 接口详细 46](#_Toc8768)

[5.10.3. 示例代码 47](#_Toc14698)

[5.11. 获得当前SDK界面方向设置 48](#_Toc6809)

[5.11.1. 接口详细 48](#_Toc3062)

[5.12. 进入用户中心 49](#_Toc7767)

[5.12.1. 接口详细 49](#_Toc1078)

[5.12.2. 示例代码 49](#_Toc19006)

[5.13. 查询用户已订购业务 50](#_Toc25898)

[5.13.1. 接口详细 50](#_Toc4493)

[5.13.2. 常见业务编码详细 51](#_Toc2144)

[5.13.3. 结果回调 51](#_Toc5535)

[5.13.4. 业务信息(UserOrderBiz)详细 51](#_Toc10157)

[5.13.5. 示例代码 52](#_Toc22354)

[6. 服务端验证登录 53](#_Toc7527)

[7. 常见错误码 54](#_Toc30925)

# 概述

本文描述应用开发者接入UniPay统一账户SDK的过程。

# 术语、定义和缩略语

## 术语、定义

本文使用的专用术语、定义见表2.1。

| 术语/定义 | 英文对应词 | 含 义 |
| --- | --- | --- |
| 开发者 | App Developer | 应用开发者 |
| 开发商 | App Provider | 应用开发商，也称机构开发者 |
| 线下 |  | 非Web形式如邮件、电话等交流方式 |

表-2.1

## 缩略语

本文使用的专用缩略语见表2.2。

| 缩略语 | 英文原文 | 中文含义 |
| --- | --- | --- |
| AppStore | AppStore | 应用商店 |
| WoStore | WoStore | 联通沃商店 |
| App | Application | 应用 |
| API | Application Program Interface | 应用程序接口 |
| SDK | Software Development Kit | 软件开发工具包 |

表-2.2

# 开发者注册及集成账户能力获取

## 使用规范

开发者需要严格按照如下规范要求进行使用：

注册成为沃商店开发者，完成签约和计费能力申请（参考3.2~3.3)

不得通过任何手段屏蔽用户下行短信通知，应用中不得包含除短信发送（SMS\_SEND)之外任何其他操纵短信的权限。

开发者需要严格按照本规范集成SDK，不得对SDK进行任何的修改。

## 开发者注册

渠道需要作为开发者（开发商）在中国联通沃商店开发者社区进行注册，并激活注册用户（URL:<http://dev.wo.com.cn/index.action>）



图-2.1

## 合作签约

开发者完成注册后，必须已经与联通沃商店完成开发者（开发商）合作签约（合作签约在联通开发者社区<http://dev.wo.com.cn/index.action> ）

## Unipay账户SDK开发包获取

开发者（开发商）与中国联通沃商店相关人员通过线下方式获取。

## 新增作品

开发者登录到沃商店开发者社区后，进入【我的社区】，选择导航菜单【作品管理】→【我的作品】，进入应用管理界面。



## 申请统一账户能力

单击【新增作品】或者【上传作品】按钮，进入作品上传第一步

如下图，在【权限能力】栏勾选【集成统一账户】，然后在新增的【集成统一账户】栏中点击【申请统一账户】按钮



上步如果成功，则【集成统一账户】栏出现【客户端ID】和【客户端秘钥】信息如下图。记录此信息



**获得客户端ID和密钥后，还需要线下通知支撑人员添加相关权限，联系方式：**

**赵国祥**

**电话：15618005945**

**QQ：1204314007**

**Email：zgx1990@gmail.com**

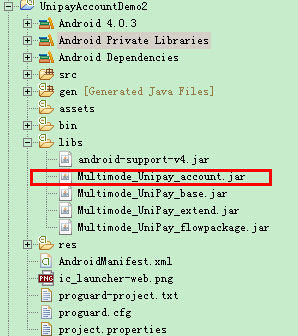
# SDK接入指南

## 配置工程

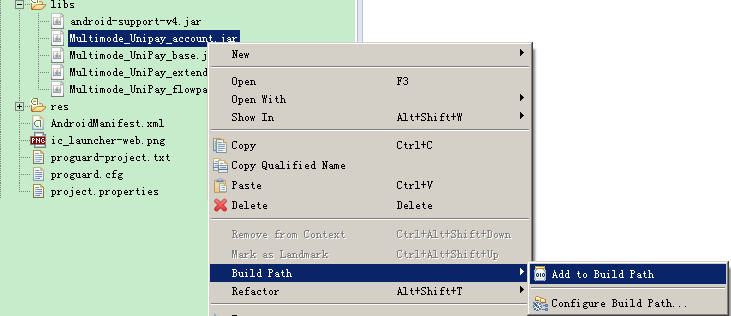
### 将SDK包导入应用工程

以下以Eclipse为开发IDE进行范例说明：

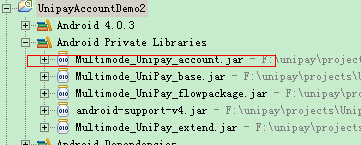
1. 将Multimode\_Unipay\_account.jar文件拷贝到工程libs(或lib)目录下，如图所示



1. 将Multimode\_Unipay\_account.jar加入编译路径中，如图所示



1. 在工程的Android Private Libraries(或Reference Libraries)目录下看到导入的jar包，则说明已成功将jar包加入编译路径



### 配置AndroidManifest

1. 在应用工程的AndroidManifest.xml文件中支持的最小安卓SDK版本必须设为9 (android:targetSdkVersion无特殊要求)，示例见下图：

|  |
| --- |
| <uses-sdk  android:minSdkVersion=**"9"**  android:targetSdkVersion=**"18"**/> |

1. 在应用工程的AndroidManifest.xml文件中加入以下权限：

|  |
| --- |
| <uses-permission android:name="android.permission.INTERNET"/>  <uses-permission android:name="android.permission.CAMERA"/>  <uses-permission android:name="android.permission.READ\_PHONE\_STATE"/>  <uses-permission android:name="android.permission.WRITE\_EXTERNAL\_STORAGE"/>  <uses-permission android:name="android.permission.MOUNT\_UNMOUNT\_FILESYSTEMS"/>  <uses-permission android:name="android.permission.INTERNET"/>  <uses-permission android:name="android.permission.ACCESS\_WIFI\_STATE"/>  <uses-permission android:name="android.permission.SEND\_SMS"/>  <uses-permission android:name="android.permission.SYSTEM\_ALERT\_WINDOW"/>  <uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE"/> |

1. 在Application节点下注册以下Activity/Service：

|  |
| --- |
| <activity  android:name="com.unipay.account.ui.generic.GenericActivity"  android:configChanges="orientation|screenSize"/>  <activity  android:name="com.unipay.account.ui.generic.GenericFragmentActivity"  android:configChanges="orientation|screenSize"/>  <service android:name="com.unipay.account.ui.generic.GenericFloatIconService" /> |

### 混淆设置

SDK本身已经被混淆过了，请在打包成APK的时候不要再混淆SDK，以免SDK被破坏。

如果使用Android SDK自带的proguard打包混淆的话，可在proguard.cfg中加入以下内容 ：

|  |
| --- |
| -ignorewarning  -libraryjars libs/\*.jar  # 或者  -keep class com.unipay.account.\*\*{\*;} |

### 参数准备

集成账户SDK接口时，需要用到如下参数，客户端ID，客户端密钥

|  |  |  |
| --- | --- | --- |
| 参数 | 参数名称 | 备注 |
| clientKey | 客户端ID | 和客户端密钥一起用于鉴定应用身份 |
| clientSecret | 客户端密钥 | 和客户端ID一起用于鉴定应用身份 |

以上参数获取方法参见[开发者注册及集成账户能力获取](#_开发者注册及集成账户能力获取_1)

## 接入SDK初始化/销毁API

### 初始化SDK

在使用SDK之前必须调用初始化接口，除了初始化接口以外，所有SDK接口必须在收到初始化结果回调之后调用，如下图所示：

|  |
| --- |
| protectedvoid onCreate**(**Bundle savedInstanceState**)**  {  **super.**onCreate**(**savedInstanceState**);**  **try**  **{**  // 传入的Context必须是当前活跃的Activity实例  UnipayAccountPlatform**.**init**(this,** CLIENT\_ID**,** CLIENT\_KEY**,**  **new** AccountAPI**.**OnInitResultListener**()**  **{**  publicvoid onResult**(**int code**,** String errmsg**)**  **{**  **if(**code **==** AccountAPI**.**CODE\_SUCCESS**)**  **{**  // 初始化完成后，应用应立即执行此接口注册账户状态变化回调  UnipayAccountPlatform**.**getInstance**().** setAccountStatusChangedListener**(**mOnAccountStatusChangedListener**);**  **}**  **else**  **{**  // 初始化失败  finish**();**  **}**  **}**  **});**  **}**  **catch(**BusyException e**)**  **{**  // 如捕获此异常则说明当前有SDK操作正在处理中  **}**  // ....其他操作  } |

SDK的其他接口均通过AccountAPI实例调用；实例可在SDK初始化完成后，通过如下方法取得

|  |
| --- |
| AccountAPI api **=** UnipayAccountPlatform**.**getInstance**();** |

### 销毁SDK

应用退出前必须调用SDK销毁接口，如下例所示：

|  |
| --- |
| protectedvoid onDestroy**()**  {  UnipayAccountPlatform**.**getInstance**().**exitSDK**();**  **super.**onDestroy**();**  } |

### 设置账户状态变化事件通知回调

SDK初始化完成后，应用必须立即注册账户状态变化事件通知回调，以及时处理用户手动操作或其他原因引发的状态变化。示例如下

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().** setAccountStatusChangedListener**(new** OnAccountStatusChangedListener**()**  {  @Override  publicvoid onLogout**()**  **{**  // 此处处理用户注销事件  **}**  @Override  publicvoid onAccountSwitched**()**  **{**  // 此处处理用户切换新账户事件  **}**  }); |

## 接入登录和注销API

### 登录

账户登录示例代码如下：

|  |
| --- |
| try  {  // 传入的Context参数必须是当前活跃的Activity实例  UnipayAccountPlatform**.**getInstance**().**login**(MainActivity.this,new** OnLoginResultListener**()**  **{**  @Override  publicvoid onLoginResult**(**int code**)**  **{**  **if(**code **==** AccountAPI**.**CODE\_SUCCESS**)**  **{**  // 登录成功  **}**  **else**  **{**  // 登录失败  **}**  **}**  **});**  }  **catch(**BusyException e**)**  {  // 捕获此异常则说明SDK当前正在执行登录或切换账户操作  } |

注：登录接口的Context参数必须是当前活跃的Activity实例

### 注销

账户注销示例代码如下：

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().**logout**(**MainActivity**.this,**  **new** OnLogoutResultListener**()**  {  @Override  publicvoid onLogoutResult**(**int code**,** String errmsg**)**  **{**  **if(**code **==** AccountAPI**.**CODE\_SUCCESS**)**  **{**  // 注销成功  **}**  **else**  **{**  // 注销失败，通常是因为此时SDK正在登录或切换账户中  **}**  **}**  }); |

## 切换账户API(必须接入)

由于某些手机系统会屏蔽应用的悬浮窗口，使得用户无法通过SDK内的悬浮工具栏切换账户，所以接入SDK的应用必须接入”切换账户API”，并在应用的界面上提供足够明显的入口供用户使用，否则应用提交测试时将不予通过。 接口详细说明参见 [切换账户](#_切换账户)

示例代码如下：

|  |
| --- |
| // 传入的Context参数必须是当前活跃的Activity实例  UnipayAccountPlatform**.**getInstance**().**switchAccount**(**MainActivity**.this,new** OnSwitchAccountResultListener**()**  {  @Override  publicvoid onLogout**()**  **{**  // 用户从当前账户注销，但未能成功登录新账户，此时应用可按用户注销账户处理  **}**  @Override  publicvoid onError**(**int code**,** String errmsg**)**  **{**  // 切换账户时出错，通常是因为此时SDK正在处理登录或切换账户操作  **}**  @Override  publicvoid onAccountSwitched**()**  **{**  // 用户从当前账户注销，并成功登录新账户  **}**  }); |

注：传入接口的Context参数必须是当前活跃的Activity实例

## 进入用户中心API(必须接入)

由于某些手机系统会屏蔽应用的悬浮窗口，使得用户无法通过SDK内的悬浮工具栏进入”用户中心”界面，所以接入SDK的应用必须接入”进入用户中心API”，并在应用的界面上提供足够明显的入口供用户使用，否则应用提交测试时将不予通过。接口详细说明参见 [进入用户中心](#_进入用户中心)

示例代码如下：

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().**enterUsercenter**(**MainActivity**.this);** |

注：传入接口的Context参数必须是当前活跃的Activity实例

## 接入可选API

除了以上必须接入的API外，SDK也提供了一些可选接口供应用根据本身需要酌情接入。

### 悬浮工具栏

用户登录账户成功以后，SDK会在屏幕前端显示悬浮工具栏，供用户查看或操作自己的账户信息，如下图所示：



SDK对应用提供接口用以控制悬浮工具栏的显示/隐藏状态，示例代码如下：

|  |
| --- |
| // 显示悬浮工具栏  UnipayAccountPlatform**.**getInstance**().**enableFloaticon**(true);**  // 隐藏悬浮工具栏  UnipayAccountPlatform**.**getInstance**().**enableFloaticon**(false);** |

注：

1. 本接口必须在账户登录成功以后调用，否则无效
2. 隐藏悬浮工具栏操作有效范围仅限于账户本次登录期间，当用下次账户登录或者切换账户后悬浮工具栏将重新处于显示状态

### SDK屏幕方向设置

应用可设置SDK界面的横竖屏方向以适应应用自身界面的横竖屏幕方向设置。示例代码如下：

|  |
| --- |
| // 设置SDK界面屏幕方向为自动  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_AUTO**);**  // 设置SDK界面屏幕方向为横屏  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_LANDSCAPE**);**  // 设置SDK界面屏幕方向为横屏，可随重力感应转换垂直方向  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_SENSOR\_LANDSCAPE**);**  // 设置SDK界面屏幕方向为竖屏  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_PORTRAIT**);**  // 设置SDK界面屏幕方向为竖屏，可随重力感应转换垂直方向  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_SENSOR\_PORTRAIT**);** |

注：当应用不设置屏幕方向或设置为自动时，SDK会根据初始化SDK和登录账户时传入的Activity的屏幕方向设置SDK自身的屏幕方向

### 其他API

详情参见[API说明](#_API说明)

## 应用发布

开发完毕后，请提交中国联通[沃商店开发者社区](http://dev.wo.com.cn/)发布，待中国联通沃商店测试中心测试通过后上线。

# API说明

## 初始化SDK

### 接口详细

本接口用于初始化SDK，创建AccountAPI实例；本接口应该在游戏开始时调用，且只能调用一次

注：

1. 本接口必须在UI线程下调用
2. 本操作为异步操作，只有应用接到【注销结果通知】后才能继续下一步账户操作
3. 除本接口外，SDK的任何其他接口都必须在调用本接口，并收到初始化成功的回调之后方可调用

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicstaticvoid init**(**  Context context**,**  String clientId**,**  String clientKey**,**  OnInitResultListener listener**)throws** BusyException | | |
| 参数名称 | 类型 | 说明 |
| context | Context | 必须为当前活跃的Activity 实例 |
| clientId | String | 客户端ID |
| clientKey | String | 客户端密钥 |
| listener | OnInitResultListener | 初始化结果回调 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| BusyException | | 当SDK此时正在处理初始化操作时抛出此异常 |

### 结果回调

初始化结果回调定义如下：

|  |
| --- |
| publicinterface OnInitResultListener  {  /\*\*  \* @param code  \* CODE\_SUCCESS: 成功;  \* 其他: 失败  \*  \* @param message  \* 错误描述  \*/  publicvoid onResult**(**int code**,** String message**);**  } |

### 示例代码

|  |
| --- |
| protectedvoid onCreate**(**Bundle savedInstanceState**)**  {  **super.**onCreate**(**savedInstanceState**);**  **try**  **{**  // 传入的Context必须是当前活跃的Activity实例  UnipayAccountPlatform**.**init**(this,** CLIENT\_ID**,** CLIENT\_KEY**,**  **new** AccountAPI**.**OnInitResultListener**()**  **{**  publicvoid onResult**(**int code**,** String errmsg**)**  **{**  **if(**code **==** AccountAPI**.**CODE\_SUCCESS**)**  **{**  // 初始化完成后，应用应立即执行此接口注册账户状态变化回调  UnipayAccountPlatform**.**getInstance**().** setAccountStatusChangedListener**(**mOnAccountStatusChangedListener**);**  **}**  **else**  **{**  // 初始化失败  finish**();**  **}**  **}**  **});**  **}**  **catch(**BusyException e**)**  **{**  // 如捕获此异常则说明当前有SDK操作正在处理中  **}**  // ....其他操作  } |

## 销毁SDK

### 接口详细

应用退出时必须调用本接口，否则可能会引起悬浮工具栏状态混乱

※本接口应尽量在UI线程下调用

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid exitSDK**();** | | |
| 参数名称 | 类型 | 说明 |
| 无 | 无 | 无 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| 无 | | 无 |

### 示例代码

|  |
| --- |
| AccountAPI api **=** UnipayAccountPlatform**.**getInstance**();**  if(api !=null)  {  api**.**exitSDK**();**  } |

## 设置账户变化事件回调

该接口应在调用初始化接口后调用。

在游戏过程中，如果手机用户使用SDK的用户中心做了注销登录或者切换账号操作，SDK会将相应事件通知到本接口指定的回调接口

注：

1. 本接口必须在UI线程下调用
2. 应用调用SDK【注销】接口时，注销结果将通知到参数指定的【注销结果回调】，不会通知到此【账户变化事件回调】中
3. 如应用在调用SDK【切换账户】接口时指定的结果回调非null，则本次操作中用户切换到新账户或注销的事将不会通知到此【账户变化事件回调】中

### 接口详细

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid setAccountStatusChangedListener**(**OnAccountStatusChangedListener listener**);** | | |
| 参数名称 | 类型 | 说明 |
| listener | OnAccountStatusChangedListener | 回调接口 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| 无 | | 无 |

### 结果回调

|  |
| --- |
| publicinterface OnAccountStatusChangedListener  {  /\*\*  \* 用户退出登录或切换账号失败  \*/  publicvoid onLogout**();**  /\*\*  \* 用户切换账号成功  \*/  publicvoid onAccountSwitched**();**  } |

### 示例代码

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().**setAccountStatusChangedListener**(**  **new** OnAccountStatusChangedListener**()**  {  @Override  publicvoid onLogout**()**  **{**  // 在此处理用户注销事件  **}**  @Override  publicvoid onAccountSwitched**()**  **{**  // 在此处理用户切换账户事件  **}**  }); |

## 登录账户

### 接口详细

注：

1. 本接口必须在UI线程下调用
2. 如调用接口时SDK已有账户登录，则SDK直接通知【登录成功】到【登录结果回调】

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid login**(**Context context**,** OnLoginResultListener listener**)throws** BusyException**;** | | |
| 参数名称 | 类型 | 说明 |
| context | Context | 必须为当前活跃的Activity 实例 |
| listener | OnLoginResultListener | 登录结果回调 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| BusyException | | 抛出此异常时说明此时SDK正在处理账户相关操作 |

### 结果回调

|  |
| --- |
| publicinterface OnLoginResultListener  {  /\*\*  \* 用户登录响应  \* @param code  \* CODE\_SUCCESS: 成功;  \* 其他: 失败  \* @param user 登录的用户信息, 仅当登录成功时有效  \*/  publicvoid onLoginResult**(**int code**);**  } |

### 示例代码

|  |
| --- |
| try  {  UnipayAccountPlatform**.**getInstance**().**  login**(**MainActivity**.this,new** OnLoginResultListener**()**  **{**  @Override  publicvoid onLoginResult**(**int code**)**  **{**  **if(**code **==** AccountAPI**.**CODE\_SUCCESS**)**  **{**  // 在此添加登录成功相关处理  **}**  **else**  **{**  // 在此添加登录失败相关处理  **}**  **}**  **});**  }  **catch(**BusyException e**)**  {  // 此时SDK正在处理账户相关操作，请稍后再试  } |

## 注销账户

### 接口详细

注：

1. 本接口必须在UI线程下调用
2. 本操作为异步操作，只有应用接到【注销结果通知】后才能继续下一步账户操作
3. 如果此时SDK没有账户登录，则SDK直接通知【注销成功】到【注销结果回调】
4. 调用此接口执行注销操作时，无论【注销结果回调】参数是否为null，都不会通知【注销事件】给【账户变化回调】

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid logout**(**Context context**,** OnLogoutResultListener listener**);** | | |
| 参数名称 | 类型 | 说明 |
| context | Context | 必须为当前活跃的Activity 实例 |
| listener | OnLogoutResultListener | 注销结果回调 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| 无 | | 无 |

### 结果回调

|  |
| --- |
| publicinterface OnLogoutResultListener  {  /\*\*  \* 用户注销响应  \* @param code  \* CODE\_SUCCESS: 成功;  \* CODE\_BUSY: sdk忙，请稍后再试  \* 其他: 失败  \*/  publicvoid onLogoutResult**(**int code**,** String message**);**  } |

### 示例代码

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().**logout**(**MainActivity**.this,new** OnLogoutResultListener**()**  {  @Override  publicvoid onLogoutResult**(**int code**,** String errmsg**)**  **{**  **if(**code **==** AccountAPI**.**CODE\_SUCCESS**)**  **{**  // 在此添加注销成功操作  **}**  **else**  **{**  // 在此添加注销失败操作  **}**  **}**  }); |

## 切换账户

本接口顺次完成以下功能：

1. 如果当前已有登录账户，则从该账户注销
2. 执行登录账户流程

### 接口详细

注：

1. 本接口必须在UI线程下调用
2. 如果此时SDK没有账户登录，则SDK直接通知【注销成功】到【注销结果回调】
3. 如果调用接口时传入的【切换账户结果回调】参数不为null，则切换账户操作引起的账户状态变化事件不会通知给【账户变化事件回调】
4. 如果调用接口时传入的【切换账户结果回调】参数为null，则切换账户操作结束时，SDK会根据最终的账户状态，发送【账户注销通知】或【切换新账户通知】给【账户变化事件回调】

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid logout**(**Context context**,** OnLogoutResultListener listener**);** | | |
| 参数名称 | 类型 | 说明 |
| context | Context | 必须为当前活跃的Activity 实例 |
| listener | OnLogoutResultListener | 注销结果回调 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| 无 | | 无 |

### 结果回调

|  |
| --- |
| interface OnSwitchAccountResultListener  {  // 切换账户结果：用户注销当前账户成功，但是登录新账户失败时  publicvoid onLogout**();**  // 切换账户结果：用户成功切换到新账户  publicvoid onAccountSwitched**();**  // 切换账户时出错，通常是因为调用"切换账户"接口时SDK正在处理账户相关操作  publicvoid onError**(**int code**,** String errmsg**);**  } |

### 示例代码

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().**switchAccount**(**  MainActivity**.this,new** OnSwitchAccountResultListener**()**  {  @Override  publicvoid onLogout**()**  **{**  // 此处添加"注销成功但登录新账户失败"情况下的操作  **}**  @Override  publicvoid onError**(**int code**,** String errmsg**)**  **{**  // 说明调用接口时SDK正在处理账户相关操作，此时不做任何处理为佳  **}**  @Override  publicvoid onAccountSwitched**()**  **{**  // 此处添加"成功切换到新账户"情况下的操作  **}**  }); |

## 获取当前登录状态

此接口用于获知当前SDK是否有账户登录

### 接口详细

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicboolean isLoggined**();** | | |
| 参数名称 | 类型 | 说明 |
| 无 | 无 | 无 |
| 返回值 | | |
| 类型 | | 说明 |
| boolean | | true:当前SDK已有登录账户  false:当前SDK没有登录账户 |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| 无 | | 无 |

### 示例代码

|  |
| --- |
| **if(**UnipayAccountPlatform**.**getInstance**().**isLoggined**())**  {  // 当前已有账户登录  }  else  {  // 当前没有账户登录  } |

## 获取当前用户信息

此接口用于获知当前SDK登录的账户信息

### 接口详细

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| public UserInfo getCurrentUserInfo**();** | | |
| 参数名称 | 类型 | 说明 |
| 无 | 无 | 无 |
| 返回值 | | |
| 类型 | | 说明 |
| UserInfo | | 当前有账户登录：当前账户信息  当前无账户登录：null |
| 抛出异常 | | |
| 异常名称 | | 说明 |
| 无 | | 无 |

### 用户信息详细

|  |  |
| --- | --- |
| 方法 | 含义 |
| String getUserId() | 获取用户id |
| String getImsi() | 获取用户imsi |
| String getNickname() | 获取用户昵称 |
| int getLevel() | 获取用户等级 |
| String getPortraitUrl() | 获取用户头像图片地址 |
| String getAccessToken() | 获取用户token |

### 示例代码

|  |
| --- |
| UserInfo uinfo **=** UnipayAccountPlatform**.**getInstance**().**getCurrentUserInfo**();**  if(uinfo !=null)  {  // 当前已有账户登录  // do sth with uinfo  }  else  {  // 当前没有账户登录  } |

## 设置悬浮按钮可见性

### 接口详细

注：

1. 该接口应在账号登录后调用，否则不起效
2. 本接口的设置仅对当前登录有效，当用户重新登录或者切换账户时，悬浮工具栏会重新处于可见状态

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid enableFloaticon**(**boolean enabled**);** | | |
| 参数名称 | 类型 | 说明 |
| enabled | boolean | true：设置悬浮工具栏为可见  false：设置悬浮工具栏为不可见 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |

### 示例代码

|  |
| --- |
| UnipayAccountPlatform.*getInstance*().enableFloaticon(**false**); // 隐藏悬浮工具栏  UnipayAccountPlatform.*getInstance*().enableFloaticon(**true**); // 显示悬浮工具栏 |

## 设置SDK界面方向

### 界面方向设置参数详细

|  |  |
| --- | --- |
| 参数 | 意义 |
| AccountAPI.SCREEN\_ORIENTATION\_AUTO | 自动，由重力感应和系统设置决定 |
| AccountAPI.SCREEN\_ORIENTATION\_PORTRAIT | 竖屏 |
| AccountAPI.SCREEN\_ORIENTATION\_LANDSCAPE | 横屏 |
| AccountAPI.SCREEN\_ORIENTATION\_SENSOR\_PORTRAIT | 竖屏，可随重力感应改变垂直方向 |
| AccountAPI.SCREEN\_ORIENTATION\_SENSOR\_LANDSCAPE | 横屏，可随重力感应改变垂直方向 |

### 接口详细

注：

1. 本接口的设置仅对SDK中的全屏界面有效，对对话框的方向无效

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid setScreenOrientation**(**int orientation**);** | | |
| 参数名称 | 类型 | 说明 |
| orientation | int | 见[界面方向设置参数详细](#_界面方向设置参数详细) |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |

### 示例代码

|  |
| --- |
| // 设置SDK界面屏幕方向为自动  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_AUTO**);**  // 设置SDK界面屏幕方向为横屏  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_LANDSCAPE**);**  // 设置SDK界面屏幕方向为横屏，可随重力感应转换垂直方向  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_SENSOR\_LANDSCAPE**);**  // 设置SDK界面屏幕方向为竖屏  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_PORTRAIT**);**  // 设置SDK界面屏幕方向为竖屏，可随重力感应转换垂直方向  UnipayAccountPlatform**.**getInstance**().**setScreenOrientation**(**AccountAPI**.**SCREEN\_ORIENTATION\_SENSOR\_PORTRAIT**);** |

## 获得当前SDK界面方向设置

### 接口详细

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicintgetScreenOrientation**();** | | |
| 参数名称 | 类型 | 说明 |
| 无 | 无 | 无 |
| 返回值 | | |
| 类型 | | 说明 |
| int | | 见[界面方向设置参数详细](#_设置参数详细) |

## 进入用户中心

打开当前登录账户的用户中心界面；该接口应在账号登录后调用

### 接口详细

注：

1. 本接口必须在UI线程下调用
2. 本接口仅在当前有账户登录时有效

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| publicvoid enterUsercenter**(**Context context**);** | | |
| 参数名称 | 类型 | 说明 |
| context | Context | 必须为当前活跃的Activity 实例 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |

### 示例代码

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().**enterUsercenter**(**MainActivity**.this);** |

## 查询用户已订购业务

本接口用于查询当前登录账户已订购的业务

### 接口详细

注：

1. 本接口必须在UI线程下调用
2. 如传入的业务代码参数为空(null或"")，则SDK将用户当前已订购的所有业务通过【查询结果】回调传给应用；无论查询到用户订购了几条(或0条)业务，传给【查询结果】回调的返回码都为CODE\_SUCCESS(0)
3. 如传入的业务代码不为空，则SDK将查询用户是否订购了指定的业务，如果已订购，则传给【查询结果】回调的返回码为CODE\_SUCCESS(0)，如未订购，则返回码为CODE\_USER\_ORDER\_BIZ\_NOT\_FOUND (-6)

|  |  |  |
| --- | --- | --- |
| 接口定义 | | |
| public void queryUserOrderBiz(  String bizCode,  OnQueryUserOrderBizResultListener listener); | | |
| 参数名称 | 类型 | 说明 |
| bizCode | String | 要查询的业务代码  常见业务编码见[常见业务编码详细](#_常见业务编码详细) |
| listener | OnQueryUserOrderBizResultListener | 查询结果回调 |
| 返回值 | | |
| 类型 | | 说明 |
| 无 | | 无 |

### 常见业务编码详细

|  |  |
| --- | --- |
| 业务编码 | 含义 |
| "0818" | 包月业务 |

### 结果回调

|  |
| --- |
| publicinterface OnQueryUserOrderBizResultListener  {  /\*\*  \*  \* @param code 返回码 CODE\_SUCCESS: 成功,如果请求时指定了要查询的业务代码  则表示用户已订购该业务, 如未指定业务代  码则表示已经获得用户订购的所有业务;  CODE\_USER\_ORDER\_BIZ\_NOT\_FOUND: 用户未订购请求指定的  业务;  其他 失败  \* @param message 错误描述  \* @param list 获得的订购业务列表, 仅在CODE\_SUCCESS时有效, 当请求指定了要查  询的业务代码时,list包含对应的那一条业务信息, 如未指定业务代码  则list包含用户当前订购的所有业务信息; 其他情况下为null;  \*/  publicvoid onResult**(**int code**,** String message**,** List**<**UserOrderBiz**>** list**);**  } |

### 业务信息(UserOrderBiz)详细

|  |  |
| --- | --- |
| 方法 | 含义 |
| String getBizCode() | 获取业务代码 |
| String getEffectDate() | 获取生效日期，格式为yyyy-MM-dd HH:mm:ss |

### 示例代码

|  |
| --- |
| UnipayAccountPlatform**.**getInstance**().**queryUserOrderBiz**(**  "0818"**,**  **new** OnQueryUserOrderBizResultListener**()**  **{**  @Override  publicvoid onResult**(**int code**,** String errmsg**,** List**<**UserOrderBiz**>** list**)**  **{**  **if(**code **==** AccountAPI**.**CODE\_SUCCESS**)**  **{**  // 用户已订购"0818"业务  **}**  **elseif(**code **==** AccountAPI**.**CODE\_USER\_ORDER\_BIZ\_NOT\_FOUND**)**  **{**  // 用户未订购"0818"业务  **}**  **else**  **{**  // 查询失败  **}**  **}**  **});** |

# 服务端验证登录

* **接口描述**  
  针对联网运营游戏三方验证的规范，CP Serve端做登陆验证的接口规范。Cp Server与账户server之间的验证。
* **接口地址**

https://open.wostore.cn/oauth2/auth/validate\_oauth2\_cp

* **请求方式**  
  POST
* **参数格式**  
  JSON
* **请求参数**

所有参数都通过request header传递：

| **元素名称** | **类型（长度）** | **标签元素** | **必须** |
| --- | --- | --- | --- |
| 客户端ID | String | client\_id | 是 |
| 客户端密钥 | String | client\_secret | 是 |
| 访问令牌 | String | access\_token | 是 |

* **返回结果**

成功:

| **元素名称** | **类型（长度）** | **标签元素** |
| --- | --- | --- |
| 用户ID | String | user\_id |

{

"user\_id":""

}

# 常见错误码

|  |  |
| --- | --- |
| 错误码 | 含义 |
| AccountAPI.CODE\_SUCCESS  0 | 成功 |
| AccountAPI.CODE\_FAILURE  -1 | 失败 |
| AccountAPI.CODE\_BUSY  -2 | 状态冲突，SDK正在处理相关操作中 |
| AccountAPI.CODE\_USER\_CANCELED  -4 | 用户取消操作 |
| AccountAPI.CODE\_NO\_ACCOUNT  -5 | 当前无登录账户 |
| AccountAPI.CODE\_USER\_ORDER\_BIZ\_NOT\_FOUND  -6 | 用户未订购指定业务 |
| AccountAPI.CODE\_ANDROID\_SDK\_VERSION\_TOOOLD  -7 | Android SDK版本过低 |