**计费中间件SDK嵌入使用说明**

**文件集成**：

1. 在AndroidManifest.xml中添加用户权限：

<uses-permission android:name=*"android.permission.INTERNET"* />

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

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

<uses-permission android:name=*"android.permission.ACCESS\_COARSE\_LOCATION"* />

<uses-permission android:name=*"android.permission.ACCESS\_FINE\_LOCATION"* />

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

<uses-permission android:name=*"android.permission.SEND\_SMS"* />

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

<uses-permission android:name=*"android.permission.READ\_CONTACTS"* />

<uses-permission android:name=*"android.permission.MOUNT\_UNMOUNT\_FILESYSTEMS"* />

<uses-permission android:name=*"android.permission.GET\_TASKS"* />

2. 在AndroidManifest.xml注册service：

<service android:name=*"com.ckmobilling.CkService"*

android:exported=*"true"*

android:process=*":remote"* >

<intent-filter>

<action android:name=*"com.ckmobilling.ICkService"* />

</intent-filter>

</service>

3. 在AndroidManifest.xml添加应用信息：

<meta-data android:name=*"app\_id"* android:value=*"/ xxxx"* />

<meta-data android:name=*"app\_channel"* android:value=*"/ xxxx"* />

app\_id：应用id

app\_channel：渠道号

注：“xxxxx”为字符串。

4. 分别拷贝assets、libs、res资源至应用对应目录下；

5. 拷贝runtime目录至应用根目录下；

6. 支付界面的Activity需继承自CkSdkActivity；

7. 应用的Application需继承自CkSdkApp。

**代码集成**：

1. 短信支付处理

调用CkSdkApi类下的doPayment函数进行支付操作。函数内部会根据服务器设置，自动选择使用支付SDK进行支付处理。

函数声明如下：

**public** **void** doPayment(String ckPayCode, PaymentCallback callback);

参数说明：

ckPayCode 触控计费代码，服务器会根据具体支付sdk生成对应的支付代码或订单。

callback 支付结果反馈，提供开发者支付成功、失败和取消反馈信息。详见后续说明。

PaymentResult ：支付结果类。详见后续说明。

示例代码：

CkSdkApi.*getInstance*().doPayment("0008", **new** PaymentCallback() {

**public** **void** paySuccess(PaymentResult result) {

Log.*d*(TAG, result.toString());

Map<String, String> map = result.getItemMapping();

String name = map.get(PaymentResult.*FEED\_BACK\_ITEM\_NAME*);

String code = map.get(PaymentResult.*FEED\_BACK\_ITEM\_CODE*);

String num = map.get(PaymentResult.*FEED\_BACK\_ITEM\_NUMBER*);

Log.*d*(TAG, "name: "+ name+ ", code: "+ code+ ", num: "+ num);

}

**public** **void** payFaliled(PaymentResult result) {

}

**public** **void** payCanceld(PaymentResult result) {

}

});

1. PaymentCallback支付结果回调接口类定义

应用需要new一个继承此接口类的对象来接收购买结果回调通知。类定义说明如下：

**public** **interface** PaymentCallback {

**public** **void** paySuccess(PaymentResult result);

**public** **void** payFaliled(PaymentResult result);

**public** **void** payCanceld(PaymentResult result);

}

paySuccess：支付成功回调函数。

payFailed ：支付失败回调函数。

PayCanceld ：用户取消支付操作回调函数。

回调参数中的PaymentResult为结果描述类，包含支付使用的渠道及相关信息。参见后面PaymentResult类的说明。

1. PaymentResult支付结果描述类

方法描述：

getPayResult：获取支付结果，返回类型为int（0 成功，1 失败，2 取消）。

getSdkType：获取支付类型，返回类型为int（0 游戏基地，1 移动MM，2 联动优势）。

getItemMapping：获取道具信息，返回类型为Map其key值分别对应为PaymentResult.FEED\_BACK\_ITEM\_NAME、PaymentResult.FEED\_BACK\_ITEM\_CODE、PaymentResult.FEED\_BACK\_ITEM\_NUMBER。

1. 道具购买参数映射关系结构表

“道具购买参数映射关系结构定义\_V1.0\_20140527.xml”文件描述了商品了具体信息，需由运营相关人员填写。具体参数说明详见该文件。