# 개요

MiSDK는 CP사에게 편하게 제 3 자 마켓 SDK들을 적용 가능하도록 통합시킨 SDK로서, 단순한 SDK 통합 뿐만 아니라, 소셜 기능, 광고 플랫폼, 로그인 기능 등을 제공합니다. MiSDK는 그 자체가 SDK가 아니라, 중국의 수많은 SDK를 통합하기 위한 목적으로 태어났으며, AnySDK와 함께 사용하여, CP측에서 직접 결제 현황을 Miguyouxi 플랫폼에서 확인 가능하도록 하기 위해여 통합 준비되었습니다.

## 2014년 8월18일 업데이트 버전 v1.2.1

SDK 적용

기본 안드로이드 네이티브 프로젝트를 진행할 때, 본 Java SDK 를 사용하면 됩니다.

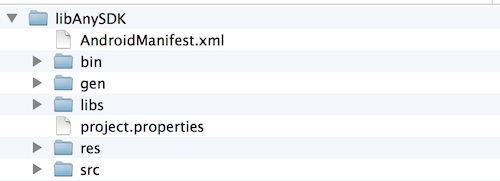
준비사항

링타이거/넷미고에서 제공하는 AnySDK용 AppKey와 AppSecret, PrivateKey 가 필요합니다.

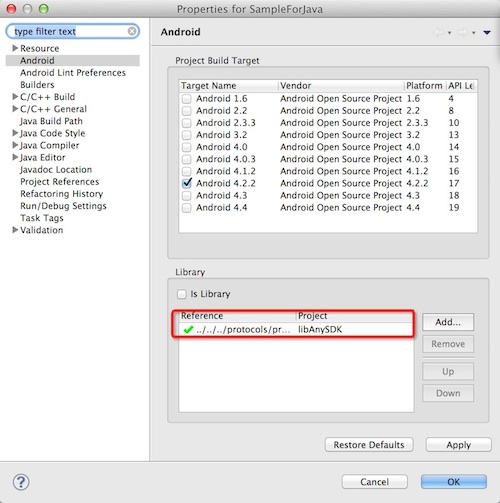
SDK 포함 방법

**제공되는 AnySDK\_Framework(Java) 압축을 푸시면,**

아래의 파일 구성입니다：

[](http://docs.anysdk.com/%E6%96%87%E4%BB%B6:Java%E7%9B%AE%E5%BD%95%E7%BB%93%E6%9E%84.jpg)

libs 폴더내의 libPluginProtocolForJava\_fat.jar、모든So관련 파일（armeabi、armeabi-v7a、x86 폴더）、res 폴더의 모든 파일을 사용하면 됩니다. API 레벨은 10 이상에서 사용 가능합니다.

[](http://docs.anysdk.com/%E6%96%87%E4%BB%B6:Java%E5%AF%BC%E5%85%A5%E5%B7%A5%E7%A8%8B1.jpg)

**AndroidManifest.xml 추가 필요 내용**

<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.RESTART\_PACKAGES" />

<uses-permission android:name="android.permission.KILL\_BACKGROUND\_PROCESSES" />

**게임 내 기본적인 삽입 코드**

一、메인 Activity의onCreate에서 AnySDK 를 초기화합니다.

String appKey = "AEE563E8-C007-DC32-5535-0518D941D6C2";

String appSecret = "b9fada2f86e3f73948f52d9673366610";

String privateKey = "0EE38DB7E37D13EBC50E329483167860";

String oauthLoginServer = "http://oauth.anysdk.com/api/OauthLoginDemo/Login.php";

AnySDK.getInstance().initPluginSystem(this, appKey, appSecret, privateKey, oauthLoginServer);

주의：appKey、appSecret、privateKey는 넷미고/링타이거에서 전달해드립니다.

二、Activity 유지를 위해 아래의 코드를 삽입합니다.：

@Override

protected void onActivityResult(int requestCode, int resultCode, Intent data){

super.onActivityResult(requestCode, resultCode, data);

PluginWrapper.onActivityResult(requestCode, resultCode, data);

}

@Override

protected void onResume() {

super.onResume();

PluginWrapper.onResume();

}

@Override

public void onPause(){

PluginWrapper.onPause();

super.onPause();

}

@Override

protected void onNewIntent(Intent intent) {

PluginWrapper.onNewIntent(intent);

super.onNewIntent(intent);

}

三、 onDestroy()에서 AnySDK를 정상적으로 종료시킵니다.

@Override

protected void onDestroy() {

AnySDK.getInstance().release();

super.onDestroy();

};

四、proguard 사용시:

-keep class com.anysdk.framework.\*\* {\*;}

-keep class com.anysdk.Util.SdkHttpListener {\*;}

결제 시스템 ( 현재는 AnySDK의 결제 부분만 사용 )

결제 결과 받기 위한 처리

아래와 같이 콜백 함수를 설정하여 결제에 대한 1차 처리를 합니다.

AnySDKIAP.getInstance().setListener(new AnySDKListener() {

@Override

public void onCallBack(int arg0, String arg1) {

switch(arg0)

{

case IAPWrapper.PAYRESULT\_SUCCESS://결제 성공 도착

// 이 부분에서 바로 처리하면, 해킹의 위험이 있으므로, 여기서 게임에 ‘결제 영수증 코드’를 전달하여, 게임에서 MiSDK 서버에 연결하여 최종 결제 확인

break;

case IAPWrapper.PAYRESULT\_FAIL://결제 실패 도착

break;

case IAPWrapper.PAYRESULT\_CANCEL://결제 취소 도착

break;

case IAPWrapper.PAYRESULT\_NETWORK\_ERROR://네트웍 문제 도착

break;

case IAPWrapper.PAYRESULT\_PRODUCTIONINFOR\_INCOMPLETE://결제 시간 초과 도착

break;

/\*\*

\* 결제 진행 중의 처리 예외 사항

\* 결제 진행 중 SDK자체에서 지연이 발생할 경우,

\* 개발자는 게임에서 결제 진행중인 화면을 보여주고, 랙이 걸린 상태일 경우, 취소 할 수 있게 해주고, 취소등의 결과를 받기 전까지는 다시 결제를 하지 않도록 조치를 해둬야 합니다.

\*/

case IAPWrapper.PAYRESULT\_INIT\_FAIL: // 결제 함수 관련 초기화 실패 도착

break;

default:

break;

}

}

});

결제 프로세스

**그림 설명**

[](http://docs.anysdk.com/%E6%96%87%E4%BB%B6:%E6%94%AF%E4%BB%98%E6%B5%81%E7%A8%8B.jpg)

**내용**

1、게임에서 AnySDK를 통해서 결제를 하면, AnySDK 서버에 결제 코드를 요청합니다.

2、AnySDK 서버가 결제 코드를 보내옵니다.

3、AnySDK 가 제 3 자 마켓의 게임 서버에 결제를 요청합니다. ( 3자 마켓 SDK내부처리)

4、3자 마켓 SDK 결제 과정을 마친 후, 결제를 완료 합니다. ( 게임에 결과 도착함 )

5、3자 마켓 서버가 AnySDK서버로 결제 결과를 반송해줍니다.

6、AnySDK 서버가 3자 마켓 서버에 결제 결과가 제대로 전달되었음을 보내줍니다.

7、AnySDK서버가 Miguyouxi 플랫폼 서버로 결제 성공을 알려줍니다.

8、Miguyouxi 플랫폼 서버가 AnySDK서버로 결제 성공이 전달되었음을 보내줍니다.

9、Miguyouxi 플랫폼 서버에 AnySDK부터 받은 결제 결과가 저장됩니다.

10、게임에서 Miguyouxi 플랫폼 서버에 4번에서 받은 결과를 확인 요청합니다. ( 수동적인 확인이므로, 결과를 제대로 받을 때까지 반복 확인 처리합니다. )

**실제 결제를 위한 처리**

**초기화 관련**

결제 관련된 부분은 결제 실행 함수를 진행하면, 자동 초기화 됩니다. 결제 초기화 성공 결과가 왔을때는 특별히 다른 처리할 필요가 없으며, 결제 초기화 실패가 왔을때, 그에 대한 처리를 하시면 됩니다.：

|  |  |  |
| --- | --- | --- |
| **내용** | **code** | **msg** |
| 초기화성공 | PAYRESULT\_INIT\_SUCCESS | 문자열 메시지 |
| 초기화실패 | PAYRESULT\_INIT\_FAIL | 문자열 메시지 |

**결제 요청**

void payForProduct(Map<String, String> info);

TProductInfo 구조체를 map，결제 아이템 관련 변수 내용을 채워서 사용：

주의：아래 내용 중, 캐릭터명, 캐릭터 아이디, 캐릭터 레벨 등은 일부 3자 마켓에서 필요로 하는 내용입니다. 만약, 해당 내용을 잘못 기입한 경우, 일부 플랫폼에서 등록이 거부될 수 있습니다.

|  |  |  |
| --- | --- | --- |
| **변수명** | **필수여부** | **설명** |
| Product\_Id | Y | 아이템 고유 id |
| Product\_Name | Y | 아이템 명 |
| Product\_Price | Y | 아이템 가격(숫자) |
| Product\_Count | Y | 아이템 갯수(여러 개의 아이템 구입시 사용 ) 사용 안하면 1 |
| Role\_Id | Y | 게임 캐릭터 id |
| Role\_Name | Y | 게임 캐릭터 명 |
| Role\_Grade | Y | 게임 캐릭터 레벨 |
| Role\_Balance | Y | 게임내 캐쉬 단위(보석, 골드 등) |
| Server\_Id | Y | 서버id，일반적으로 “1” |
| EXT | Y | Migu\_SDK 관련 내용 < 문서 별첨 참고 > |

Product\_Price는 반드시 정수( 1 위안 단위 )로 기입. 1.5위안 등은 불가

결제 결과 도착값

|  |  |  |
| --- | --- | --- |
| **내용** | **code** | **msg** |
| 결제 성공 | PAYRESULT\_SUCCESS | 문자열 메시지 |
| 결제 취소 | PAYRESULT\_CANCEL | 문자열 메시지 |
| 결제 실패 | PAYRESULT\_FAIL | 문자열 메시지 |
| 네트워크 관련 에러 | PAYRESULT\_NETWORK\_ERROR | 문자열 메시지 |
| 결제 정보 미완결 전달 | PAYRESULT\_PRODUCTIONINFOR\_INCOMPLETE | 문자열 메시지 |
| 결제진행중 | PAYRESULT\_NOW\_PAYING | 문자열 메시지 |

예제>

Map<String, String> mProductionInfo = new HashMap<String, String>();

mProductionInfo.put("Product\_Price", "1");

mProductionInfo.put("Product\_Id", "10");

mProductionInfo.put("Product\_Id", "monthly");

mProductionInfo.put("Product\_Name","gold");

mProductionInfo.put("Server\_Id", "13");

mProductionInfo.put("Product\_Count", "1");

mProductionInfo.put("Role\_Id","1");

mProductionInfo.put("Role\_Name", "1");

mProductionInfo.put("Role\_Grade", "1");

mProductionInfo.put("Role\_Balance", "1");

ArrayList<String> idArrayList = AnySDKIAP.getInstance().getPluginId();

if (idArrayList.size() == 1) {

AnySDKIAP.getInstance().payForProduct(idArrayList.get(0), mProductionInfo);

}

else {

ChoosePayMode(idArrayList);

}

**결제 항목 확인시 사용되는, 결제 영수 번호 얻기**

std::string getOrderId(string pluginId)

해당 함수로 얻은 결과를 MiguSDK 에 전달해서 서버와 결제 결과 확인

예제>

AnySDKIAP.getInstance().getOrderId(pluginId);

<별첨> MiSDK

MiSDK

1. 회원 가입 : <http://www.miguyouxi.com/>
2. 개발자 계정 등록 : 넷미고/링타이거에서 대행
3. 게임 추가 및 dev\_appid, dev\_appkey 얻기 : 넷미고/링타이거에서 전달
4. 게임 내에 ANYSDK 적용 – CP 적용 단계 시작
5. 아이템 결제 진행을 위해, 내용 기입시 EXT변수에 아래의 내용을 전달

전달내용：’dev\_appid’+ ‘|’+ ‘KEY’

KEY생성 방식：**KEY=md5(dev\_appkey.'com.miguyouxi')**

예> 아래의 아이템 결제 등록 부분 중…

mProductionInfo.put("Role\_Grade", "1");

mProductionInfo.put("Role\_Balance", "1");

String key = getMd5Hash( "d698114d83e0475f3c277b25395cce" + "com.miguyouxi" );

mProductionInfo.put("EXT", "9f35835099234baba8344|" + key );

1. 결제 성공 후 확인 과정：**SDK 확인 연결 주소：**[**http://op.miguyouxi.com/**](http://op.miguyouxi.com/)또는**<http://op.miguyouxi.com/index.php>**

* 필요 변수 ( POST 방식 )

appid：dev\_appid

key：5번에서 생성한 key

action: “queryAnyOrder”

order\_id：ANYSDK 결제 성공 후 얻게 되는 “결제 영수 코드”

* 리턴값 : JSON 방식

**\*\* “status” key**

* "1111" : 성공, "2222" : 실패

**\*\* “msg” key** : 성공 및 실패 등등 상세 메세지

**예> {“status” : “1111”, “msg” : “success payment”}**