**라인 크로스 프로모션 연동 가이드 : Android**

**MIH for LINE, ver\_1.1**

**Notice**

최종 업데이트 : 2016-04-28

|  |
| --- |
| **목차**   1. [기본연동](#기본연동) 2. [매체 연동](#매체연동)    1. [매체공통](#매체공통)    2. [오퍼월광고](#오퍼월광고)    3. [전면광고](#전면광고) 3. [광고주 연동](#광고주연동) |

**기본연동**

**필요조건**

1. 연동을 위해서 라인크로스프로모션 최신 SDK가 필요합니다.
2. 연동을 위해서 IGAWorks 앱키, 해시키가 필요합니다. (라인 기술 PM 담당자에게 문의해주세요.)
3. 연동을 위해서 Android 2.3(API Level 9) 이상이 필요합니다.

**SDK 설치**

1. 최신SDK 패키지를 다운로드합니다.
2. 다운로드 받은 파일의 압축을 해제합니다.
3. LineCrossPromotion~\*.jar 파일을 안드로이드 프로젝트의 libs 폴더에 복사합니다.
4. Gradle을 업데이트합니다.

**안드로이드 매니페스트 설정 및 기본 API**

AndroidManifext.xml 파일에 필수 설정을 추가합니다.

1. **앱키/해시키/리워드서버 타입 설정**

<application> 태그 안에 아래와 같이 앱키, 해시키와 리워드서버타입을 설정합니다.

|  |
| --- |
| <**application**>  <**meta-data android:name="igaworks\_app\_key" android:value="INPUT\_YOUR\_APP\_KEY"**/>  <**meta-data android:name="igaworks\_hash\_key" android:value="INPUT\_YOUR\_HASH\_KEY"**/>  <**meta-data android:name="igaworks\_reward\_server\_type" android:value="server"**/> </**application**> |

1. **필수 퍼미션 추가**

<manifest> 태그 안에 아래와 같이 필수 퍼미션을 추가합니다.

|  |
| --- |
| <**manifest**>  <**uses-permission android:name="android.permission.INTERNET"**/>  <**uses-permission android:name="android.permission.ACCESS\_NETWORK\_STATE"**/>  <**uses-permission android:name="android.permission.GET\_ACCOUNTS"**/>  </**manifest**> |

1. **필수 액티비티 추가**

<application> 태그 안에 아래와 같이 필수 액티비티를 추가합니다.

|  |
| --- |
| <**application**>  <**activity android:name="com.igaworks.adpopcorn.v2.activity.ApOfferWallActivity"  android:theme="@android:style/Theme.NoTitleBar"**/> <**activity android:name="com.igaworks.adbrix.line.view.InterstitialActivity"  android:configChanges="keyboardHidden|orientation|screenSize"  android:theme="@android:style/Theme.Translucent.NoTitleBar"**/> <**activity android:name="com.igaworks.adpopcorn.v2.activity.ApVideoAdActivity"  android:theme="@android:style/Theme.NoTitleBar"  android:screenOrientation="landscape"**/>  </**application**> |

1. **구글 인스톨 리시버 설정**

<application> 태그 안에 아래와 같이 구글 인스톨 리시버를 설정합니다.

|  |
| --- |
| <**application**>  *<!-- IGAWORKS INSTALL RECEIVER SETTING -->* <**receiver android:name="com.igaworks.IgawReceiver" android:exported="true"**>  <**intent-filter**>  <**action android:name="com.android.vending.INSTALL\_REFERRER"**/>  </**intent-filter**> </**receiver**>  </**application**> |

1. **startApplication API**

앱이 실행될 때 1번만 호출합니다. SDK를 초기화하고 DAU를 측정합니다.

Launcher 액티비티의 onCreate()에 추가하는 것을 권장합니다.

|  |
| --- |
| **public class** MainActivity **extends** Activity {  @Override  **protected void** onCreate(Bundle savedInstanceState) {  **super**.onCreate(savedInstanceState);  setContentView(R.layout.***activity\_main***);  *// Call StartApplication API* LineCrossPromotion.*startApplication*(**this**);  }  } |

1. **startSession API**

앱 세션 체크를 위해서 호출합니다. 앱 사용시간 측정, 트래킹링크 이용 등을 위해 필요합니다.

Launcher 액티비티를 포함한 최소한 1개 이상의 액티비티의 onResume()에 추가해야 합니다.

|  |
| --- |
| **public class** MainActivity **extends** Activity {  @Override **protected void** onResume() {  **super**.onResume();  *// Call StartSession API* LineCrossPromotion.*startSession*(MainActivity.**this**); }  } |

1. **endSession API**

앱 세션 체크를 위해서 호출합니다. 앱 사용시간 측정, 트래킹링크 이용 등을 위해 필요합니다.

startSession API가 호출된 액티비티의 onPause()에 반드시 추가해야 합니다.

|  |
| --- |
| **public class** MainActivity **extends** Activity {  @Override **protected void** onPause() {  **super**.onPause();  *// Call EndSession API* LineCrossPromotion.end*Session*(); }  } |

**매체 연동**

라인 크로스프로모션의 매체(publisher) 연동을 하기 위해서 다음의 가이드를 따릅니다.

**준비사항**

1. 기본연동이 완료되어 있어야 합니다.
2. setUserId API를 이용하여 라인유저식별값을 등록하여야 합니다.

**매체 연동 : 공통**

1. **setUserId API**

광고를 참여하는 유저를 식별하여 리워드 지급 정보에 사용하기 위해 유저식별값을 설정합니다.

\* 유저식별값 가이드라인은 라인의 가이드를 따릅니다.

\* 전면광고, 오퍼월광고를 가져오기 전에 설정하여야 합니다.

|  |
| --- |
| String encryptedUserId = **"ENCRYPTED\_LINE\_USER\_ID"**; **if**(encryptedUserId != **null**){  *//Set encryptedUserId before getting contents.* LineCrossPromotion.*setUserId*(encryptedUserId); }**else**{  *//Get userId information.* } |

1. **setLanguage API**

라인크로스프로모션 SDK는 디바이스의 language값을 사용하여 광고를 노출합니다.

만약 별도의 언어 설정을 원할 경우, 직접 언어를 설정할 수 있습니다.

LineCrossPromotion.LANGUAGE 클래스에 정의된 언어코드를 사용합니다.

|  |
| --- |
| *// Set Language*  LineCrossPromotion.*setLanguage*(**this**, LineCrossPromotion.LANGUAGE.***ENGLISH***); |

**[지원언어]**

|  |  |  |
| --- | --- | --- |
| **언어코드** | | **비고** |
| LineCrossPromotion.LANGUAGE | KOREAN | 한국어 |
| JAPANESE | 일본어 |
| CHINESE\_CN | 중국어(간체) |
| CHINESE\_TW | 중국어(번체) |
| ENGLISH | 영어 |

**매체 연동 : 오퍼월광고**

오퍼월 형태의 광고를 노출하기 위해서 다음의 가이드를 따릅니다.

**준비사항**

1. 매체 공통연동이 완료되어 있어야 합니다.
2. 애드스팟키가 발급되어 있어야 합니다. (라인 기술PM에게 문의해주세요.)
3. UserId가 설정되어 있어야 합니다.
4. **getOfferwall API**

오퍼월 광고 정보를 불러오기 위해 호출합니다.

|  |
| --- |
| *// Request offerwallAd Information with AdSpotKey* LineCrossPromotion.*getOfferwall*(MainActivity.**this**, **"INPUT\_YOUR\_AD\_SPOT\_KEY"**, iAdSpotListener); |

**주의사항\_**

* iAdSpotListener는 getOfferwall의 결과에 대한 이벤트 리스너입니다.
* getOfferwall에서 제공하는 광고 정보는 1회성 정보입니다. 새로운 광고정보를 얻기 위해서는

getOfferwall를 다시 호출하여야 합니다.

예) 오퍼월 노출 및 종료된 다음 다시 getOfferwall를 호출하여 새로운 광고 정보 불러오기.

1. **IgawAdSpotListener**

이벤트 리스너를 구현하여 광고 컨텐츠 요청 결과 확인, 신규 광고정보 요청 등의 액션을 정의할 수 있습니다.

각 이벤트 리스너로 전달되는 파라미터의 상세 내용은 [[이벤트리스너 파라미터 상세](#이벤트리스너파라미터상세)]를 참고해주세요.

1. **OnGetContentsSuccess**

getOfferwall API의 호출이 성공했을 때, 결과가 전달됩니다.

1. **OnGetContentsFailure**

getOfferwall API의 호출이 실패했을 때, 결과가 전달됩니다.

1. **OnShowContentsSuccess**

showOfferwall API의 호출이 성공했을 때, 결과가 전달됩니다.

1. **OnShowContentsFailure**

showOfferwall API의 호출이 실패했을 때, 결과가 전달됩니다.

1. **OnCloseContents**

오퍼월이 종료될 때, 결과가 전달됩니다.

이 때, 광고정보를 다시 불러오는 것을 권장합니다.

|  |
| --- |
| *// 3. Implement IgawAdSpotListener* IgawAdSpotListener iAdSpotListener = **new** IgawAdSpotListener() {  @Override  **public void** OnGetContentsSuccess(String adSpotKey, **int** contentsType, **int** adType) {  *//This method will be called when the request succeeded.*  }  @Override  **public void** OnGetContentsFailure(String adSpotKey, AdSpotError adSpotError) {  *//This method will be called when the request failed.*  }  @Override  **public void** OnShowContentsSuccess(String adSpotKey) {  *//This method will be called when the AdContent showing succeeded.*  }  @Override  **public void** OnShowContentsFailure(String adSpotKey, AdSpotError adSpotError) {  *//This method will be called when the AdContent showing succeeded.*  }  @Override  **public void** OnCloseContents(String adSpotKey) {  *//This method will be called when the AdContent closing.*  } }; |

**[****이벤트리스너 파라미터 상세]**

|  |  |  |
| --- | --- | --- |
| **파라미터** | **코드** | **비고** |
| adSpotKey | - | 애드스팟키 |
| contentsType | OFFERWALL (int:1) | 오퍼월 광고 |
| INTERSTITIAL (int:2) | 전면 광고 |
| VIDEO (int:3) | 비디오 광고 |
| NATIVEAD (int:4) | 네이티브 광고 |
| VIDEO\_MEDIATION (int:5) | 비디오 미디에이션 광고 |
| AdType | REWARD (int:1) | 리워드 광고 |
| NON\_REWARD (int:2) | 논리워드 광고 |
| AdSpotError.errorCode | ERROR (int:100) | 에러 |
| EXCEPTION (int:200) | 정의되지 않은 에러 |
| CAN\_NOT\_FIND\_AVAILABLE\_CAMPAIGN  (int:1000) | 송출가능한 광고없음 |
| MEDIA\_NOT\_EXIST (int:9999) | 앱키 오류 |
| NO\_AD (int:2000) | 광고 없음 |
| SHOW\_ADSPOT\_FAILURE(int:2030) | 광고 노출 실패 |
| INVALID\_SPOT\_KEY (int:2100) | 스팟키 오류 |
| INVALID\_CONTENTS\_KEY (int:2200) | 컨텐츠키 오류 |
| AdSpotError.errorMessage | - | 에러 메시지 |

1. **showOfferwall API**

불러온 광고 정보를 노출하기 위해 호출합니다.

|  |
| --- |
| *// Show Offerwall*  LineCrossPromotion.*showOfferwall*(MainActivity.**this**, **"INPUT\_YOUR\_AD\_SPOT\_KEY"**); |

**주의사항\_**

* showOfferwall로 노출된 광고 정보는 재사용이 불가합니다. 따라서 showOfferwall를 호출한 이후에는 반드시 새로운 광고 정보를 로드하기 위하여 getOfferwall를 호출하여야 합니다.

**매체 연동 : 전면광고**

전면광고 형태의 광고를 노출하기 위해서 다음의 가이드를 따릅니다.

**준비사항**

1. 공통연동이 완료되어 있어야 합니다.
2. 애드스페이스 키가 발급되어 있어야 합니다. (라인 기술PM에게 문의해주세요.)
3. UserId가 세팅되어 있어야 합니다.
4. **initInterstitialAd API**

전면광고 기능을 초기화하기 위해 호출합니다.

|  |
| --- |
| *// init InterstitialAd with InitInterstitialAd Event Listener*  LineCrossPromotion.*initInterstitialAd*(MainActivity.**this**, **"INPUT\_YOUR\_AD\_SPOT\_KEY"**, initInterstitialAdListener); |

**주의사항\_**

* initInterstitialAdListener는 initInterstitialAd의 호출결과에 대한 이벤트 리스너입니다.

1. **InitInterstitialAdEventListener**

이벤트리스너를 구현하여 initInterstitialAd 의 호출결과를 확인할 수 있습니다.

1. **OnInitSuccess**

initInterstitialAd 의 호출이 성공일 때, 호출됩니다.

1. **OnInitFail**

initInterstitialAd 의 호출이 실패일 때, 에러메시지와 함께 호출됩니다.

|  |
| --- |
| *// 2. Implement InitInterstitialAdEventListener* InitInterstitialAdEventListener initInterstitialAdListener = **new** InitInterstitialAdEventListener() {  @Override  **public void** OnInitSuccess() {  *//This method will be called when init succeeded.*  }  @Override  **public void** OnInitFail(String errorMsg) {  *//This method will be called when the init failed.*  }  }; |

1. **showInterstitialAd API**

전면광고를 노출하기 위해 호출합니다.

|  |
| --- |
| *// showInterstitialAd with InitInterstitialAd Event Listener*  LineCrossPromotion.show*InterstitialAd*(MainActivity.**this**, **"INPUT\_YOUR\_AD\_SPACE\_KEY"**, showInterstitialListener); |

**주의사항\_**

* showInterstitialListener 는 showInterstitialAd 의 호출결과에 대한 이벤트 리스너입니다.

1. **ShowInterstitialAdEventListener**

이벤트리스너를 구현하여 showInterstitialAd 의 호출결과를 확인할 수 있습니다.

1. **OnShowInterstitialSuccess**

showInterstitialAd 의 호출이 성공일 때, 애드스페이스키와 함께 호출됩니다.

1. **OnShowInterstitialFailure**

showInterstitialAd 의 호출이 실패일 때, 에러메시지와 함께 호출됩니다.

1. **OnCloseInterstitial**

전면광고가 종료될 때, 애드스페이스키와 함께 호출됩니다.

|  |
| --- |
| *// Implement ShowInterstitialAdEventListener* ShowInterstitialAdEventListener showInterstitialAdListener = **new** ShowInterstitialAdEventListener() {  @Override  **public void** OnShowInterstitialSuccess(String adspotkey) {  *//This method will be called when init succeeded.*  }  @Override  **public void** OnShowInterstitialFailure(String adspotkey, String errorMsg) {  *//This method will be called when the init failed.*  }  @Override  **public void** OnCloseInterstitial(String adspotkey) {  *//This method will be called when the interstitial is closed.*  }  }; |

**광고주 연동**

라인크로스프로모션의 광고주(advertiser) 연동을 하기 위하여 다음을 따릅니다.

(광고 집행 절차는 라인 기술 PM에게 문의해주세요.)

1. **회원가입 연동**

회원가입 절차가 완료되는 시점에 다음과 같이 unlock API를 호출합니다.

|  |
| --- |
| *// call below api when register process completed*  LineCrossPromotion.*unlock*(LineCrossPromotion.Event.***SIGN\_UP***); |

1. **로그인 연동**

로그인이 완료되는 시점에 다음과 같이 unlock API를 호출합니다.

|  |
| --- |
| *// call below api when login completed*  LineCrossPromotion.*unlock*(LineCrossPromotion.Event.***SIGN\_IN***); |

1. **딥링크 기능 추가**

기존 설치된 유저들이 광고에 참여할 때에 마켓 이동 없이 바로 앱 실행이 가능하도록 딥링크 기능을 추가해야 합니다.

다음을 참고하여 AndroidManifest.xml 파일에 딥링크 기능을 위한 새로운 intent-filter를 추가합니다.

|  |
| --- |
| <**activity  android:name=".MainActivity"  android:label="@string/app\_name"  android:theme="@style/AppTheme.NoActionBar"**>  <**intent-filter**>  <**action android:name="android.intent.action.MAIN"** />  <**category android:name="android.intent.category.LAUNCHER"** />  </**intent-filter**>  *<!-- Please add the new intent-filter for App Open with Deeplink -->* <**intent-filter android:label="@string/app\_name"**>  <**action android:name="android.intent.action.VIEW"** />  <**category android:name="android.intent.category.DEFAULT"** />  <**category android:name="android.intent.category.BROWSABLE"** />  *<!-- Accepts URIs that begin with "myAppName://default" -->* <**data android:scheme="myAppName" android:host="default"** />  </**intent-filter**> </**activity**> |

**참고\_**

딥링크 기능 추가 가이드는 구글의 가이드를 기준으로 작성되었습니다. [[구글 딥링크 가이드 바로가기](https://developers.google.com/app-indexing/android/app)]